1. MATHMOD
VIENNA

1. MATHMOD
VIENNA

1. MATHMOD
VIENNA

1. MATHMOD VIENNA

Proceedings of the IMACS Symposium
on MATHEMATICAL MODELLING
during February 2-4, 1994
at Technical University Vienna, Austria

VOLUME 1 - 5

Edited by I. Troch and F. Breitenecker

ARGESIM Report 11

ASIM Mitteilung AM 40B

ISBN print: 978-3-901608-00-1
ISBN e-book: 978-3-901608-02-5

1. MATHMOD VIENNA

February 2-4, 1994
Technical University Vienna, Austria

Organizer:
Division for Mathematics of Control and Simulation (E114-5)
of Technical University Vienna

Sponsors:
IMACS (Int. Ass. for Maths. & Comp. in Simul.)
Division for Mathematics of Control and Simulation of Technical University Vienna
Technical University Vienna

Cosponsors:
IFAC (Int. Fed. for Automatic Control)
IAMCM (Int. Ass. for Math. & Comp. Modelling)
ASIM (German Speaking Simulation Group)
GAMM (Soc. Appl. Math. Mech., Germany)
VDI/VDE-GMA (Soc. for Meas. & Automation)
OCG (Austrian Computer Soc.)
ÖMG (Austrian Math. Soc.)

ARGESIM Report 11

ASIM Mitteilung AM 40B

ISBN print: 978-3-901608-00-1
ISBN e-book: 978-3-901608-02-5

1994

Preface

Applied Mathematics as well as simulation are based essentially on appropriate modelling of a given task. Such a model is intended to help for a better understanding of what is going on in the system and - last but not least - to assist in finding a good solution of the problem to be solved. There is a rather broad consent that modelling is of intrinsic importance. Moreover, most engineers and scientists know quite well that appropriate modelling is far from being easy and that the quality of the result depends strongly on the quality of the model. By now, this is accepted by practically all people involved in such tasks, no matter whether they work at a scientific institution or in an industrial environment.

However, one can observe that e.g. modelling approaches, methods for model simplification or for parameter estimation are developed and modified quite frequently and quite many things are discovered repeatedly. Therefore, many discussions during congresses and conferences showed the desire for a forum with mathematical modelling as its center whereas the solution of the underlying problem remains in the background is of peripheral interest only. Consequently, the IMACS symposium on Mathematical Modelling originated being devoted to the mathematical (or formal) modelling of all type of systems no matter whether the system is

- dynamic or static
- lumped parameter or distributed parameter
- deterministic or stochastic
- linear or nonlinear
- continuous or discrete
- or of any other nature.

Consequently, a wide variety of formal models is to be discussed and the term "mathematical model" includes classical models such as differential or difference equations, Markov processes, ARMA models as well as more recent approaches such as Bond graphs or Petri nets.

The written versions of the contributions to 1.MATHMOD Vienna are collected in these Proceedings. There are several tqpes of papers:

I ... invited lectures of experts presenting a survey

C ... contributed papers which were selected for presentation by a reviewing process in which the members of the IPC took part and which was based on submitted extended abstracts

O ... papers contributed upon invitation of a session organizer

L ... late papers for which the abstracts were submitted after the deadline and which were reviewed and selected by a the organizing committee

P ... posters

All these contributions were colleted and arranged in sessions according to their main thematic point. This was by no means easy because quite many contributions address several different aspects in a balanced manner. Therefore, the arrangement chosen for this volume follows rather closely the one of the conference were also time limitations had to be observed.

The editors wish to express their sincere thanks to all who have assisted them by making the idea of this symposium known within the scientific community by acting as sponsor or cosponsor, who have assisted them in the reviewing process and - last but not least - have done a good job by putting together special sessions devoted to one main theme.

Vienna, January 1994 I.Troch, F. Breitenecker

CONTENTS

VOLUME 1

Invited Lectures:

Physical Systems Modelling
Organizer: P.C. Breedveld (Cambridge, USA)

Bond Graph Modelling: Theory, Software, and Applications
Organizer: W. Borutzky (Gummersbach, D)

Modern Aspects of Modelling and Simulation
Organizer: A. Javor (Budapest, H)

Abbreviations:

C ... Paper, contributed upon invitation and under the responsibility of the organizer of the corresponding session.
R ... Regular paper; selected upon recommendation of the IPC.
L ... Late paper; selected by the NOC.

* ... Manuscript received too late to be printed together with the other papers of the session. If received by or prior to January 30, 1994, the paper is printed in the "Late Papers" volume.

Automation of Modelling and Related Software Techniques
Organizer: C. Maffezzoni (Milan, I)

Mechanistic Modelling of Industrial Processes and Plants
Organizer: K. Juslin (Espoo, SF)

Qualitative Modelling and Analysis of Dynamical Systems
Organizer: J. Lunze (Hamburg, D)

Qualitative Models and Neural Networks in Modelling
Organizer: R. Gorez (Louvain-la-Neuve, B)

Modelling of Distributed Parameter Systems

VOLUME 2

Order Reduction of Linear Systems 1
Organizer: P. C. Müller (Wuppertal, D)

Order Reduction of Linear Systems 2
Organizer: P. C. Müller (Wuppertal, D)

Simplification, Estimation and Compensation in Nonlinear Control Systems
Organizer: P. C. Müller (Wuppertal, D)

Parameter Uncertainties in Linear Systems
Organizer: P. C. Müller (Wuppertal, D)

Stochastic Systems Modelling

Neural Modelling
Organizer: F. Rattay (Vienna, A)

Biomathematics 1
Organizer: F. Rattay (Vienna, A)

Biomathematics 2
Organizer: F. Rattay (Vienna, A)

Modelling of Biotechnical Processes
Organizer: A. Munack (Braunschweig, D)

VOLUME 3

Modelling of Biochemical and Related Processes 1

Modelling of Biochemical and Related Processes 2

Modelling and Simulation of Waste Water Treatment Processes 1
Organizer: M. Köhne (Siegen, D)

Modelling and Simulation of Waste Water Treatment Processes 2
Organizer: J. P. Babary (Toulouse, F)

Simulation Models of Environmental Systems
Organizer: A. Sydow (Berlin, D)

Modelling of Control Systems 2
Organizer: D. J. Murray-Smith (Glasgow, UK)

Modelling of Control Systems 1
Organizer: D. J. Murray-Smith (Glasgow, UK)

Mathematical Modelling of Engineering Problems
Organizer: X. J. R. Avula (Rolla, USA)

Modelling and Applications in Engineering
Organizer: J. Ragot (Vandoeuvre, F)

Modelling for Simulation of Robot Dynamics
Organizer: G. Magnani (Milan, I)

VOLUME 4

AI, Modelling and Simulation in Robotics and Control 1
Organizer: S. Tzafestas (Athens, GR)

AI, Modelling and Simulation in Robotics and Control 2
Organizer: S. Tzafestas (Athens, GR)

Sophisticated Robots
Organizer: S. Tzafestas (Athens, GR)

Data Models for Mechanical System Simulations
Organizer: W. Schiehlen (Stuttgart, D)

Modelling of Mechatronic Systems
Organizer: J. Lückel (Paderborn, D)

Modelling of Electrical Systems

Modelling in Electronics
Organizer: N.R. Andronaty (Kishinev, Moldova)

VOLUME 5

Mobile Radio Networks
Organizer: B. Walke (Aachen, D)

Traffic Modelling – Theory and Applications
Organizer: K.-O. Praskawetz (Braunschweig, D)

Modelling of Economic and Socio Economic Systems

Teaching

Modelling Methodologies and its Application
Organizer: Bo Hu Li (Beijing, PRC)

Automation of Modelling and Software Support
Organizer: N. R. Andronaty (Kishinev, Moldova)

Numerical Methods in Natural Sciences
Organizer: B. Rybakin (Kishinev, Moldova)

Various Theoretical Aspects

CONTENTS

The following manuscript was not printed completely in Vol. 3, p. 481

Late Papers

Posters

ADVANCES IN BOND GRAPH MODELLING: THEORY, SOFTWARE, APPLICATIONS

1. MATHMOD VIENNA, 1994 IMACS Symposium on Mathematical Modelling, Vienna, Austria

W. Borutzky
Department of Computer Science
Cologne Polytechnic, 51643 Gummersbach, FRG

G. Dauphin-Tanguy
Laboratoire d'Automatique et d'Informatique Industrielle de Lille
Ecole Centrale de Lille, B. P. 48, 59651 Villeneuve d'Ascq, France

J. U. Thoma
Thoma Consulting, Bellevueweg 23, CH 6300 Zug, Switzerland
IMAGINE, Maison Productique, F 42300 Roanne, France

Abstract

Bond-graph modelling has been known for more than 30 years and is recognized to an increasing extend by academia and industries all over the world. By the same time considerable progress has been made with regard to the methodology, to the development of supporting software, and to applications in various fields of engineering.

The aim of this paper is to survey more recent advances in bond graph modelling, to discuss some topics of current research, and to briefly investigate emerging relationships to other disciplines.

1 Introduction

Bond-graphs were devised by H. Paynter at MIT in April 1959 (Paynter in [6]) and subsequently developed into a methodology together with Karnopp and Rosenberg. Early prominent promoters of bond-graph modelling techniques among others were J. Thoma, van Dixhoorn, and P. Dransfield. They contributed substantially to the dissemination of bond-graph modelling in Europe, Australia, Japan, China and India. Van Dixhoorn founded an IMACS Technical Committee on bond graph modelling chaired by J. Thoma for many years. Moreover, since the early days the methodology has been computer assisted by the well known ENPORT programs originated by Rosenberg. Nowadays, the success of bond-graph modelling reflects in a number of subconferences within IMACS conferences, a first international SCS conference explicitly dedicated to bond graph modeling in 1993 in San Diego, invited plenary session papers on bond graph modelling (Cellier [11]), the 1990 second edition of the well known textbook of Karnopp and Rosenberg (Karnopp et al. [15]) and the new textbook of Thoma [23] that was published also in 1990, seminars in industry, and an increasing number of companies using bond graphs, especially in France. Last but not least, bond graph modelers organized into national associations like the *Bondgraafclub* in the Netherlands, or the *Club de Bondgraphistes* in France.

Besides ENPORT and the TUTSIM, formerly THTSIM program developed in the 70's numerous other programs with windows and menu oriented graphical user interfaces and powerful symbolic manipulation and numerical capabilities have emerged supporting bond graph modelling, analysis, and simulation. Relations to other disciplines like object oriented programming, artificial intelligence, or parallel processing are investigated and exploited. Nevertheless, in some cases adequate, proper bond-graph modelling of details of real-world physical systems respectively processes can be a challenge, however with the prospect of a gain in physical insight.

Before discussing some state-of-the-art issues and topics of current research the concept of bond graph modelling is briefly outlined in the next section.

2 Brief Outline of the Principles of Bond Graph Modelling

As it is common in other modelling methodologies first of all complex physical systems are partitioned into subsystems which in turn are decomposed further hierarchically top down to components of which the dynamic behavior is known or down to elements that represent physical processes. In bond-graph modelling that decomposition is guided by the view that subsystems, components, elements interact by exchanging power which is intuitive and essential in the bond graph modelling approach. Power while flowing through a system is distributed, passed on, stored, converted into other forms. An important consequence is that true bond graph modelling displays conservation of energy (1st law of thermodynamics) and of quantities like momentum, matter, electrical charge, etc. Thus, bond graph modelling applies basically to continuous processes and systems, but virtually instantaneously changes can be included as discussed in section 5 below. Adopting the abstraction of spatially concentrated properties the fundamental physical processes of energy storage, energy transformation, and energy dissipation as well components and subsystems are represented by the vertices of a bond-graph while the edges called bonds denote the power exchange between through so-called ports. (Nodes with several ports are called multiports.) Bond-graphs are directed graphs in the sense that a half arrow attached to a bond indicate a reference orientation of the power flow represented by that bond. Since in every energy domain the amount of power is given by the product of two conjugate variables, in contrast to block diagrams, edges in a bond-graph are associated with two generalized variables *effort* and *flow*. Unified mnemonic codes for the basic physical processes of energy delivery, storage, transportation, and dissipation make bond graphs applicable to all kind of physical systems and best suited for multidisciplinary physical systems in which different forms of energy and their interactions are involved. Moreover, since the exchange of power between components of a system or between subsystems is mostly bound to real-world interconnections like mechanical shafts, hydraulic pipes, or electrical wires the physical structure of a system is a strong help in the development of bond graph models. In many cases a bond graph, before it is simplified, shows a close topological affinity to a schematic system representation.

A final important issue of bond graph modelling that needs to be mentioned in this brief outline of fundamentals is the concept of causality. One aim of bond graph modelling is to come up with a concise graphical representation from which, either manually or automatically, a mathematical model can be derived. To that end the generalized variables *effort* and *flow* associated with each bond of a bond-graph are viewed as two signals with opposite directions. Considering a port of an element or of a multiport one signal must be an input while the other is an output. The direction of the effort signal is depicted by a so-called causal stroke at one end of that bond, perpendicular to the bond. The important point is that from a complete causally augmented bond graph information concerning the form of the mathematical model to be constructed can be obtained prior to considering any equations. It suffices to know whether port characteristics of vertices are linear or not. That is, generation of the equations of a mathematical model in general is the final step in the systematic step by step approach of bond graph modelling. This is an important characteristic of the bond graphs methodology and will be discussed in more detail later on. Another feature of causal bond graphs with respect to control applications is that they allow for investigation of structural properties like observability and controllability (Sueur and Dauphin [21]).

3 Multibond Graphs

A remarkable feature of the advanced bond graph modelling methodology is the concept of multibond graphs not appreciated by all bond graph modelers in the same way. Originating from an idea of Paynter and used in the beginning by Bonderson [2] for description of one-dimensional distributed systems, who called them vector bonds, they were systematically developed into the concept of multibonds by P. Breedveld and exploited in modelling 3D-mechanical multibody systems by Bos [5]. In a multibond graph representation ordinary bonds are collected into multidimensional bonds which in turn can be collected into arrays of multibonds. By consequence the associated generalized power variables effort and flow become vectors. The advanced abstraction of multibond graphs results a compact notation that conveniently supports a top down modelling approach. Its full power however, deploys in modelling large 3D-mechanical multibody systems. The result is a clear and compact representation from which 3D Newton-Euler equations can be derived. In modelling e. g. robots with several degrees of freedom (DOF's) standard multibond graphs of a single freely moving rigid body containing a transformation from a reference frame to a body-fixed reference frame are connected according to the joints between the bodies allowing to account for damping and compliance in the joints (Bos [5]). An important step in the progress of bond graph modelling is that modelling and simulation environments like for instance, CAMAS (Broenink et al. in [6]) or BONDYN (Vera et al. in [14] were developed that do not only support graphical input of multibond graphs but can also generate the corresponding systems equations. Moreover, since solvers for Differential Algebraic Systems (DAE's) like the DASSL code (Brenan et al. [8]) are used, algebraic constraints which frequently occur in models of rigid multibody systems, in general are no problem for the simulation software. That is, in principle there is no need to either modify an initial bond graph model for instance, by introducing additional small compliances or to reformulate equations by explicitly using a symbolic manipulation package. Nowadays multibond graphs are widely used, in particular in the Netherlands and in France.

4 Bond-Graph Modelling of Systems with Convection

Hydraulic and pneumatic systems are commonly described in an Eulerian frame, that is with reference to a control volume appropriately defined . On the other hand the difficulty of constructing true bond graph models for open thermodynamic systems in which enthalpy is carried across boundaries into and out of a control volume by means of matter flows has been well known. By consequence, due to the complexity of the problem a number of approaches have emerged. One attractive approach due to Karnopp [15] is to represent a gas filled control volume by a so-called accumulator with a true mechanical bond and two pseudo bonds. (A pseudo bond is a bond for which the product of effort and flow is not power.) From a practical point of view it is important to note that the accumulator although not a true capacitor with a stored energy function from which its constitutive relations could be derived, nevertheless can be connected (with some care) to a true bond graph representation of the mechanics of a system while thermal aspects are computationally conveniently and more intuitively modelled by a pseudo bond graph.

Another approach proposed by Brown in [7] has been to introduce so-called convection bond graphs in which two efforts, enthalpy h and pressure p are assigned to bonds while the flow variable variable is chosen to be the matter flow $\dot{m}$, and in which standard bond graph elements are adapted to that particular bond. The development of convection bond graphs was influenced by previous work of Thoma and Atlan [24] who also coined the term *convection bond*. An important feature of that remarkable approach is that the concept of two bilateral signal flows associated with the power bonds in conventional bond graphs can be retained such that causality can be assigned consistently allowing to derive state equations. However, at least to the authors knowledge, up to now there is yet no software that directly supports convection bonds.

On the other hand Beaman and Breedveld showed by investigating a number of idealized case studies that for open systems in principle, true bond graphs without ad hoc elements nor even controlled sources can be developed (Beaman and Breedveld [1]). This was done for some real-world systems under simplifying modelling assumptions. For instance, Willson and Traver carried out a control volume analysis for components of a 2-stroke internal combustion liquid piston pump and came up with a true bond graph model in which heat transfer is not taken into account (Willson and Traver [25]). Borutzky in [6] used a control volume approach for an incompressible, isothermal fluid to propose a true bond graph model of the fluid mechanical interaction in spool valve control orifices which is close to formulae commonly used for practical investigation of hydraulic control systems. Nevertheless, the authors feel that the development of true bond graph models for thermodynamic systems in general will remain a difficult task and give rise to further modelling attempts. For instance, in 1992 Thoma proposed another approach to true bond graph modelling of thermofluid systems using an Eulerian frame and 3-dimensional multibonds of which the conjugate variables are pressure and volume flow, temperature and entropy flow, chemical potential and matter flow (Thoma [22]). Excluding phase changes the chemical potential μ is dropped turning the corresponding bond into an activated bond that only represents matter flow. An important point with regard to practical engineering is that the output flow variables of a basic multiport R element representing restrictors and the output variables of a multiport C element accounting for fluid storage can be computed by using real data tables. Here efforts, flows and power variables (enthalpy flux $\dot{h}$ and heat flow $\dot{Q}$) are used alternately , in order to take advantage of their respective conservation properties. As demonstrated by Thoma the multiport R element can be extended into a pseudo bond graph representation of heat exchangers and moreover, of thermal fluid machines by adding a pseudo bond for conduction with conjugate variables temperature T and heat flow $\dot{Q}$ and a mechanical bond. Cf. Fig. 1 (Thoma in [18]). Note, that multibonds are distinguished from normal bonds by adding a little circle.

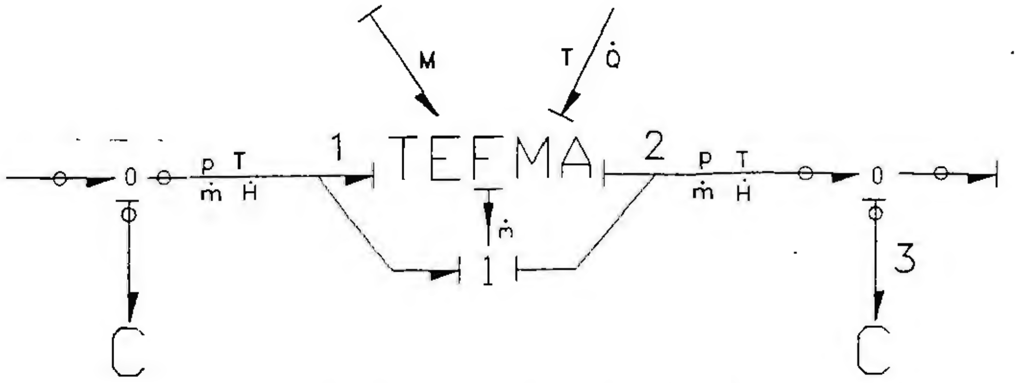


Figure 1: Pseudo bond graph of thermal fluid machine

5 Treating Discontinuities in a Bond Graph Modelling Frame

Conventional bond graph model represent power flows as well as storage and conversion of energy in a physical system. These fundamental processes are continuous with respect to time (and space). Hence, bond graphs cannot be expected to easily support the useful abstraction of state transitions taking place virtually instantaneously in a macroscopic time scale. While simulation programs like for instance ACSL, ESACAP, or SIL accept event driven continuous models a true bond graph modelling approach forces either a microscopic view or to look for simplifying assumptions in order to ensure that the model does not violate conservation principles for physical quantities. Since this is not satisfactory with regard to investigation of real-world physical systems dynamics for which not the beauty of the model is of major concern but the results obtained from simulation of the model, the question of how to treat discontinuities in a bond graph modelling frame is one of the topics currently discussed by bond graph modelers. In fact, two 1993 international conferences dedicated a special session exclusively to that topic. Considering electrical diodes, thyristors, or hydraulic check valves recently several authors proposed to incorporate the concept of an ideal switch into the bond graph language and to introduce either another basic bond graph element with variable causality (Strömberg et al. in [14]) or to model the switch supposed non ideal by means of a transformer modulated by a Boolean signal and a small ON-resistance that is, using standard elements (cf. Fig 2). In that case, the choice is made to assign a unique and fixed causality

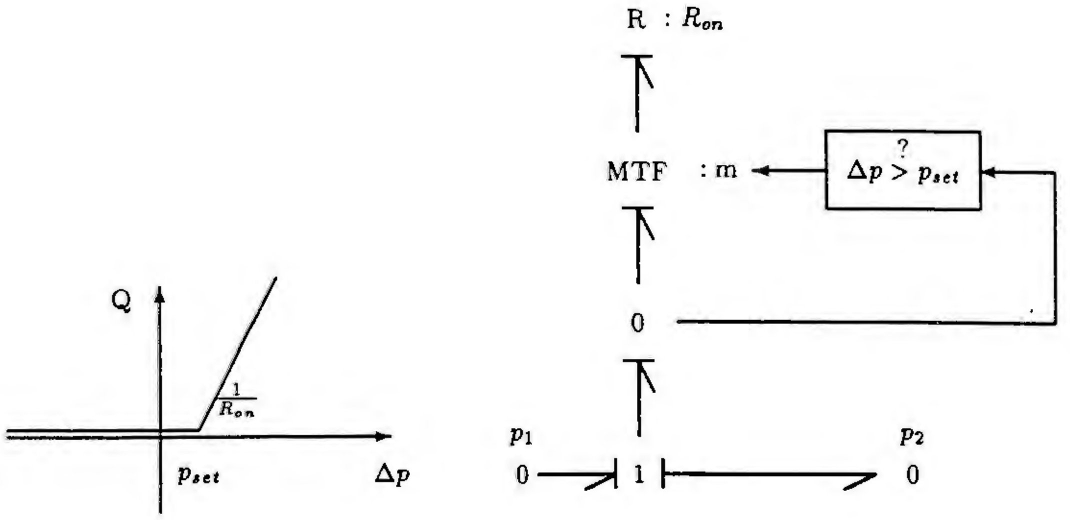


Figure 2: piecewise linear characteristic and bond graph model of a hydraulic check valve

to the association MTF - R element whatever the switch state is, which is physically correct because of the hypothesis of a non ideal switch (Dauphin et al. in [18]). It leads to a unique mathematical model valid not only for simulation but also for control design (Ducreux et al. in [14]). Other approaches are to represent energy interactions that depend on conditions by so-called switched bonds and to use a Finite State Machine representation (Broenink and Wijbrans in [14]) in addition to the bond graph, or last but not least, to identify major modes of operation in a system, to represent states and state transitions in an overall Petri net and to develop for each mode of operation taken into account a bond graph or a set of disjoint bond graphs composed of standard elements (Borutzky and Thoma [3]). An attractive feature of switches is that one single bond graph can be used for all modes of operation. Parts of the bond graph are simply switched off and on. Moreover, one single set of system equations can be generated in which Booelan expressions account for the different modes. A disadvantage is that causality changes on switching elements partly affect the rest of the bond graph if not prevented by introducing parasitic elements (causality resistors). Again, a drawback of these causality resistors is that they can lead to widely separated time constants in the model. If causality resistors are not used causality switching may entail a change in the dimension of the state vector and produce Dirac pulses in the time history of some variables. For linear systems theses pulsed values can be calculated in order to compute the initial values for the new mode (Buisson [9]). Bond graph representation that employ switched bonds do not explicitly indicate which bonds are switched on and off at a given time point. (With n switched bonds at most 2^n combinations of switch states are possible.) The third approach has the advantage that conventional bond graphs do not have to be extended in some way. The important point is that a bond graph is only used as long as the system is in one mode of operation. The problem is the same as with bond graphs using switched bonds. Theoretically the number of different system modes may be 2^n. Obviously, in case of a large number of modes it is not attractive to develop sets of conventional bond graphs for each mode even if the individual bond graphs are small. The different approaches have been discussed in more detail by Borutzky in [19].

6 Analysis and Symbolic Processing of Causal Bond Graphs

An important well known feature of causal bond graphs is that information about the form of a mathematical model to be generated can be obtained prior to writing any equation. For instance, if there storage elements that have a port with derivative causality or/and causal paths between the ports of elements with algebraic constitutive relationships occur, it can be concluded that the mathematical model in general will be a set of Differential and Algebraic Equations (DAE's). In the past such a result of causal analysis was of more importance than today because many simulation programs had very limited capabilities to solve models of that category forcing the user to modify the initial model appropriately. With the advent of solvers like DASSL which is mostly used, the need for producing assignment statements has relaxed. By consequence, since DAE's can be generated straightforward from causal bond graphs automatic generations of system equations extends to the very general case of bond graphs containing any kind of so-called Zero-order Causal Paths (ZCP's) (van Dijk and Breedveld [13]). In case the user has access to the equations of an automatically generated model an advantage of DAE's over the explicit state-space form is that they directly reflect underlying fundamental physical principles and constraints that is, modelling assumptions are more explicit. Nevertheless, presently not all bond graph preprocessors for continuous systems simulation programs have been adapted for generation of DAE's, while in the meantime the algorithms of DASSL have been incorporated into the widely used ACSL program, and a number of bond graph modelling and simulation programs are able to generate and to solve DAE's. An alternative to the well known Sequential Causality Assignment Procedure (SCAP), especially for mechanical multibody systems is the Lagrange causality approach proposed by Karnopp [16]. However, at least to the author's knowledge derivation of Lagrange equations from bond graphs up to now is not supported by modelling and simulation tools.

On the other hand in a recent theoretical investigation van Dijk and Breedveld in [14] showed that the system of DAE's in the general case is of semi-state space form and statements about its structure can be deduced from structural properties of the causal bond graph. Moreover, they give topological criteria under which a state-space form can be derived from the linearized semi-state space equations. A practical implication is that system equations derived from a bond graph after linearization by means of formula manipulation may be analyzed by programs like MATLAB in the context of control engineering applications.

The significance of causality in bond graphs reflects in the development of tools that emphasize not (only) on simulation but on symbolic and numerical analysis. In fact, an acausal bond graph can be viewed as a symbolic core representation from which specific representations can be automatically derived. For instance, programs like ARCHER (Azmani and Dauphin in [6]), or the Bond Graph Toolbox (Nolan in [7]) automatically assign causality, perform causal analysis (determination of causal paths, algebraic loops), and generate mathematical models in symbolic form. Obviously, automatic causal augmentation of an acausal bond graph in combination with re-arrangement of constitutive relations is of particular importance for models that are hierarchically composed of sub-models and in deriving a mathematical model of reduced order from a bond graph. For linear systems, transfer functions or transfer matrices can be derived in addition to formal linear state-space equations, allowing for an analysis of system properties like stability that relates to model parameters. In the Bond Graph Toolbox symbolic manipulation is carried out by MATHEMATICA. A remarkable analysis capability of ARCHER is that a causal bond graph can be investigated with regard to structural controllability and observability (Sueur and Dauphin [21]). Lastly, both analysis tools provide a link to numerical simulation. Archer can transform the symbolic form of a model into a notation accepted by ACSL or MATLAB, while in the Bond Graph Toolbox MATHEMATICA is used to output FORTRAN code that can be compiled and linked to an appropriate solver from a mathematical library.

7 Applications

Although the various engineering disciplines have their own traditional modelling techniques bond graphs have been used for physical systems modelling in many fields of applications and numerous application related papers have been published (Breedveld, Rosenberg, Zhou in [7]).

Presently there is a remarkable growing interest in using bond graphs and multibond graphs for modelling multibody and mechatronic systems in robotics and automation, especially in the French automobile industry. This can be seen for instance, from the OLMECO project initiated by PSA Peugeot Citroën and a European consortium that aims at building an open library of reusable bond graph based models of mechatronic components (Rault in [6]) and e. g. by the fact that in the 1993 IEEE conference on Systems, Man and Cybernetics an organized special session on mechatronics in car industry was held in which most papers were related to bond graphs.

In a related field of applications, bond graphs have been used intensively in industrial projects for modelling electrohydraulic and pneumatic systems for instance by IMAGINE, an exceptional consultant company of a team of bond graph modelers (Moulaire in [19]). Details of such systems like an energetically consistent two-port C element representation of the variable volume chambers in hydraulic cylinders still have been subject of current research (Borutzky in [19], Scavarda [20]).

Furthermore, presently bond graphs have gained considerable attraction in modelling power electronic systems and electrical machines in research as well as in industry (Dauphin and Rombaut in [18]). In that context switching elements and their implications for causality are subject of ongoing research.

Last but not least, recently some (distinct) bond graph approaches to modelling of complex processes in distillation columns with a two phase fluid and chemical reactors have been presented (Brooks and Cellier in [14]). For dynamics and temperature control of simple chemical reactors modelled by bond graphs see Thoma et al. in [14]. The authors feel that due to the complexity of the problems including convection and phase changes bond graph modelling of these phenomena will be subject of remain subject of ongoing research, however, as Brooks and Cellier state: "bond graphs do point the way towards the nature of the requisite research" [14].

Moreover, it is professor Paynter's vision that bond graphs will play an important role in understanding and modelling processes in such fields as chemistry, electrochemistry and biochemistry [18]. As early as 1973 Oster et al. [17] used bond graphs for modelling chemical reaction systems and in his recent textbook on *Continuous System Modeling* F. Cellier [10] devoted a voluminous chapter to that topic.

8 Relationships of Bond-Graph Modelling to Other Disciplines

Frequently it is appropriate to represent the control system by a standard block diagram while the system to be controlled may be described by a bond graph. With such mixed representations it should be kept in mind that causality is not as easily traced as in pure bond graphs. Computational dependencies may be somewhat hidden and are only revealed by careful examination [12]. Combinations of bond graphs and block diagram are supported by a novel modeling system BondGraph-80 developed by Reid (Reid and Rosenberg in [6]). An essential feature of this preprocessor is that object oriented programming (OOP) techniques were used in its implementation that allows for appropriate capturing of objects, attributes, and interrelations of objects in physical systems modelling. The fundamental concepts of OOP, namely encapsulation, inheritance and polymorphism are the key to the re-use and modification of submodels that have proven useful and to the definition of generic submodels. The object oriented paradigm supports hierarchical modelling and provides a new kind of flexibility in customizing and extending the capabilities of the software.

In regard to physical system models containing discontinuities one approach proposed by Borutzky et al. in [19] already mentioned above has been to combine standard bond graphs with a Petri net. This idea has been pursued also by Bloch et al. [18] in a recent paper in modelling hybrid systems.

Furthermore, bond graph modelling relies on the abstraction of lumped parameters. Nevertheless, besides the well known modal analysis investigated by Margolis, as well as by Lebrun, recently Pelegay et. al. proposed to combine the benefits of bond graph modelling and Finite Element techniques by representing the element stiffness matrix and the element mass matrix resulting from a finite element approximation of distributed parameter subsystems by means of an energy conservative C field, respectively I field in an overall bond graph of the system (Pelegay et al. in [14]).

Moreover, some advanced bond graph based modelling environments like the mechatronic modelling environment MAX (van Dijk et al. in [6]) even allow for using application dependent graphical symbols in the development of so-called Ideal Physical Models (IPM's) such that bond graphs can be viewed as an intermediate format in the transformation of an initial graphical system description into a mathematical model. By that way bond graphs may be transparent to users who like to prefer a representation other than bond graphs for some reasons. The other way round, users who are familiar with the bond graph language may start with an initial bond graph model and request for an application dependent schematic description. Now, if one of the both representations of the same model is manipulated by means of a graphical editor these modifications are automatically and consistently transformed to the other representation(s). This management of multiple representations is achieved by maintaining an internal core model.

Finally, recently bond graphs have attained some interest in artificial intelligence, where they are viewed as a conceptual framework and as a knowledge representation that can be exploited for qualitative reasoning that requires deep-level knowledge models, and for qualitative simulation. A number of corresponding software tools have emerged in this presently highly advocated research area among which QREMS (Linkens et al. in [14]) is an environment that supports modelling and simulation of physical systems by combining qualitative reasoning with bond graphs.

9 Conclusions

In this paper we aimed at giving a survey of the state-of-the-art and of trends presently pursued in bond graph modelling by focusing the discussion on some topics which we think are of actual relevance. However, the world of bond graph modelling has much more facets and the interested reader is invited to have a look at the voluminous and ever increasing bibliography (Breedveld, Rosenberg, Zhou in [7]) and to study original papers related to his/her field of specialization.

By concluding, there is no doubt, advances in theory and software development have contributed significantly to the power and potential the bond graph modelling approach shows today. Nevertheless, human expertise and creativity will still remain the key to a successful modelling process.

References

[1] *Beaman, J. J.; Breedveld, P. C.:* Physical Modeling With Eulerian Frames and Bond Graphs, J. Dyn. Sys. Meas. Control, June 1988, Vol 110, pp. 182-188

[2] *Bonderson, L. S.:* Vector Bond Graphs Applied to One-Dimensional Distributed Systems, J. Dyn. Sys. Meas. Control, March 1975, pp. 75-82

[3] *Borutzky, W.; Thoma, J. U.:* On representing hydraulic control system models with discontinuities in the bond graph framework, these proceedings

[4] *Borutzky, W.:* The Bond Graph Methodology and Environments for Continuous Systems Modelling and Simulation, Proc. of the 1992 European Simulation Multiconference, York, U. K., 1992, pp. 15-21

[5] *Bos, A. M.:* Modelling Multibody Systems in Terms of Multibond Graphs; with application to a motorcycle, PhD thesis, University of Twente, Enschede, 1986

[6] *Breedveld, P. C.; Dauphin-Tanguy, G.* (Editors): Bond Graphs for Engineers, North-Holland, 1992

[7] *Breedveld, P. C.* (Guest Editor): Current Topics in Bond Graph Related Research, J. Franklin Inst., Vol. 328 No. 5/6, 1991

[8] *Brenan, K. E.; Campbell, S. L.; Petzold, L. R.:* Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations, North-Holland, 1989

[9] *Buisson, J.:* Analysis of Switching Devices with Bond Graphs, J. Franklin Inst., Vol. 330, No. 6, pp. 1165-1175, 1993

[10] *Cellier, F.:* Continuous System Modeling, Springer-Verlag, 1991

[11] *Cellier, F.:* Hierarchical Nonlinear Bond Graphs - A Unified Methodology for Modeling Complex Physical Systems, Proc. European Simulation Muliconference, Nürnberg, F. R. G., 1990, pp. 1-13

[12] *Cornet, A.; Lorenz, F.:* Equation ordering using bond graph causality analysis, MODELLING AND SIMULATION of systems, J. C. Baltzer AG, Scientific Publication Co., pp. 55-58, 1989

[13] *van Dijk, J.; Breedveld, P. C.:* Simulation of System Models Containing Zero-Order Causal Paths - I. Classification of Zero-order Causal Paths, J. Franklin Inst., Vol. 328, No. 5/6, 1991, pp. 959-979

[14] *Granda, J. J.; Cellier, F.:* ICBGM'93, Proc. of the 1993 Western Simulation Multiconference, La Jolla, CA, SCS, Simulation Series, Vol. 25, Number 2, 1993

[15] *Karnopp, D. C.; Margolis, D. L.; Rosenberg, R. C.:* SYSTEM DYNAMICS: A UNIFIED APPROACH, Wiley & Sons, 1990

[16] *Karnopp, D. C.:* Alternative Bond Graph Causal Patterns and Equation Formulation for Dynamic Systems, ASME J Dyn Syst Meas Control, Vol. 105, No. 2, 1983, pp. 58-63

[17] *Oster, G. F.; Perelson, A. S.; Katchalsky, A. K.:* Network Thermodynamics: Dynamic Modelling of Biophysical Systems, Q. Rev. Biophys., Vol. 6, No. I, pp. 1-134, 1973

[18] Proc. IEEE/SMC'93, Le Touquet, France, 1993

[19] Proc. of the 1993 European Simulation Multiconference, Lyon, June 1993

[20] *Scavarda, S.:* , these proceedings

[21] *Sueur, C,; Dauphin-Tanguy, G.:* Structural Controllability/Observability of Linear Systems Reprensented by Bond Graphs, J. Franklin Inst., Vol. 326, No. 6, pp. 869-883

[22] *Thoma, J. U.:* Thermofluid Systems by Multi-bondgraphs, J. Franklin Inst., Vol. 329, No. 6, pp. 999-1009, 1992

[23] *Thoma, J. U.:* Simulation by Bondgraphs, Springer-Verlag, 1990

[24] *Thoma, J. U; Atlan, H.:* Network thermodynamics with entropy stripping, J. Franklin Inst., Vol. 303, No. 4, pp. 319-328, 1977

[25] *Willson, B.; Traver, A. E.:* The Use of Control Volume Analysis and Non-potenital Junction Concepts to Model Liquid Piston Engine Dynamics, Proc. of the American Control Conference, Vol. 2, pp. 1436-1443, 1987

ENHANCED ENVIRONMENTS FOR THE DEVELOPMENT AND VALIDATION OF DYNAMIC SYSTEM MODELS

D.J. Murray-Smith
Department of Electronics and Electrical Engineering
University of Glasgow
Glasgow G12 8QQ
Scotland

Abstract. Many modern continuous system simulation tools provide good facilities for the efficient implementation of simulation models and for interactive experimentation using a simulation model. In most cases, however, the environment lacks some important features which can greatly assist the user in tackling efficiently some of the broader issues which arise in the model development process, such as model testing and validation. This paper reviews the problems which arise in the validation of dynamic system models and discusses the implications in terms of the type of computing environment within which model development can be carried out efficiently.

1. INTRODUCTION

The user-friendly nature of modern simulation packages and the very powerful facilities for model implementation within such software products have contributed greatly to the range of applications which now exist for continuous system simulation methods. This growth means, of course, that simulation tools are being used more and more by non-specialists and the risk of errors through misunderstandings and misuse is therefore becoming steadily more significant.

The objectives of computer simulation studies differ considerably according to the application area, but some overall classification is possible in terms of the intended use of the model. Many simulation models are developed to provide insight about the dynamic behaviour of a complex system, or to allow design decisions to be made, or to provide a basis for predictions of system behaviour.

As part of the model development process it is very important to be able to test a simulation program to establish that it is an accurate implementation which incorporates the equations of the chosen mathematical model. It is also vital to test the mathematical model to ensure that, in the context of the intended application, the mathematical description is itself appropriate.

Although the significance of both of these aspects of testing appears to have been recognised in many texts on modelling (e.g. [1]) it is disturbing to find that most applications papers pass over the question of simulation testing in a superficial fashion, or make no mention of it at all. Those applications studies in which some discussion of validation is included seldom provide enough detail of the methods used. The processes of model testing and evaluation too often appear to be regarded as an afterthought, rather than as a central part of the model development process. Validation and testing should be an integral part of the iterative process of development of the mathematical model and the associated simulation program. Their importance should be reflected in the provision of specialised facilities available for model validation and testing within the computing environment.

2. THE PROCESSES OF SIMULATION MODEL VALIDATION.

While validation appears to be a somewhat neglected topic in many application areas, there are some safety-critical applications in which the importance of model validation is already fully recognised. The aerospace industry provides many good illustrations of this, since accurate externally-validated models are needed for the design of aircraft flight control systems, for the design of simulators for pilot training and for other activities such as aircraft handling-qualities investigations. The nuclear power industry provides other examples in which safety requirements make it essential to have a high level of confidence in simulation models and in the results of simulation experiments.

Guidelines on terminology were published in 1979 by the Technical Committee on Model Credibility of the Society for Computer Simulation [2]. Although these have not been adopted as a standard by everyone using simulation and modelling techniques, the committee has provided very useful definitions. One key recommendation of the Committee was that there a strong distinction should be drawn between the words "verification" and "validation". This has since been extended slightly to associate the word "internal" with "verification" and "external" with "validation"[3,4]. Internal verification is defined as the process of proving that a computer simulation is consistent with the underlying model, to a specified degree of accuracy, while external validation involves demonstrating that the mathematical or conceptual model has an acceptable accuracy over the range of conditions relevant for the application.

2.1 Internal verification.

Criteria for internal verification involve the following two aspects:

(i) Internal consistency of the simulation program with the mathematical model on which it is based. The program and the model must be shown to involve no contradictions in terms of mathematics, logic or concepts.

(ii) Algorithmic validity of the simulation program so that all the numerical algorithms and associated software routines are shown to be appropriate and provide solutions having a specified numerical accuracy.

Internal verification is important at every stage of model development. All changes within a model must be considered carefully in terms of internal verification, however minor they may be. This must involve very careful line-by-line checks of code which form the simulation program. In addition there are a number of additional checks and comparisons that can be helpful. Checks of well-understood special cases which can sometimes even be derived using pencil-and-paper calculations are particularly appropriate and these can be divided conveniently into checks of static (or equilibrium) states and dynamic checks.

2.2 External validation.

Criteria for external validation involve assessment of the accuracy and suitability of a model in the context of the intended application. They may include a number of different aspects, the most important of which are:-

(a) Theoretical validity, in the sense that the model shows overall consistency with accepted theories or is based upon a satisfactory theoretical foundation.

(b) Empirical validity, with adequate agreement shown between the behaviour of the model and that of the real system represented by the model.

Assessment of theoretical validity involves checking that the chosen mathematical description does not, in any way, violate important physical laws or principles. Many situations can arise in which mathematical statements have no physical relevance, or have restricted validity in terms, say, of the frequency range over which they are applicable.

Empirical validity is concerned, in the broadest sense, with comparisons between the behaviour of the model and the behaviour of the real system. This can include comparison of chosen system and model variables under equilibrium conditions, comparison of stability limits and comparisons in terms of the dynamic response of selected variables to chosen input perturbations. Parameter sensitivity analysis and system identification techniques have been found to be particularly important tools for the investigation of empirical validity.

Models are not unique and in most situations of practical importance there will be a number of candidate models which give an adequate match for a given set of experimental responses. Models should therefore be assessed for a variety of experimental data sets and, before being accepted for the intended application, should be shown to be capable of matching the experimental results to an acceptable level of accuracy for all of the test conditions.

3. THE COMPUTING ENVIRONMENT.

It can be seen that the tasks involved in simulation model validation go far beyond the technical processes of constructing a simulation model and the performing of simulation experiments. They may include analysis of linearised descriptions derived from a more general nonlinear model, storage, retrieval and comparison of simulation results for a wide range of conditions, comparisons with measured responses obtained from the real system, system identification and parameter estimation, sensitivity analysis, experiment design, post-processing, graphical presentation of results and documentation of simulation models and experiments. What is of vital importance is to have a properly integrated set of software tools covering the continuous system simulation requirements, together with database software for experimental and simulation model response records, facilities for visualisation and well-designed user-friendly software for all of the analysis and optimisation tasks mentioned above.

The computer-based support system should ideally be wide in scope but closely integrated in terms of the ways in which its different parts relate. It should be easily comprehensible and tolerant of mistakes on the part of the user, so that the effects of user errors should be capable of being undone without undue difficulty. It should be an adaptable environment with powerful help facilities providing rapid and efficient access to relevant information for both novice and experienced users. Self documentation is also an important feature in the highly interactive situations which can arise in validation applications.

Recent experience with the development and application of validation methods for helicopter flight mechanics models [5-8] suggests that currently available simulation tools do not yet provide an environment which is ideal. For example, one approach to the empirical validation of these flight mechanics models involves comparison of linear models identified from flight test data with equivalent models obtained from a theoretical nonlinear description by linearisation. The comparisons of the theoretical and identified models must be made for a wide range of operating conditions over the complete flight envelope of the aircraft. In the planning of external validation experiments of this kind it is, of course, important to make provision for a number of separate tests at each flight condition. For example, careful consideration of linearity is of importance for reliable estimation of parameters in a linear description. Test inputs should therefore be applied to the aircraft for a number of different amplitudes and for different directions for each test point. For each test condition it is also helpful to carry out tests with different forms of test input signal, selected to cover different parts of the frequency range. Repeated tests are desirable for each situation to reduce the risks of data degradation, due for example to external disturbances such as atmospheric turbulence. In the case of a conventional helicopter a single test input shape applied on each of four controls with two repeats at two amplitudes in each direction of control movement would give a total of 48 experiments at each test point. The requirement for several input test signal shapes and a range of different operating points within the flight envelope further magnifies the task. Each experiment is likely to lead to the creation of a large data file involving dynamic response data relating to a number of measured variables associated with the fuselage, main rotor and tail rotor. When using data of this kind for model validation it is likely that one will also generate a large number of files relating to simulation experiments. Unless the computing environment is designed to allow users to handle both test data and simulation model response data efficiently the whole validation task can easily become unmanageable.

4. DISCUSSION.

It is possible that something useful can be learned about the integration of simulation software with other modules by considering the experience which has been gained in the development of facilities for the computer-aided design. MacFarlane, Grübel and Ackermann have provided [9] an interesting review of future possibilities in terms of control engineering design environments. MacFarlane [10] has more recently expanded upon this in a broader context through a discussion paper on computer environments appropriate for engineering design and for education. These thought-provoking papers bring together many useful ideas, some of which should be equally relevant for those concerned with the development of environments for system modelling and simulation.

One interesting current development, which might provide a basis for a powerful support environment for modelling and simulation activities, is the ANDECS system being developed at DLR-Oberpfaffenhofen [11]. Although developed primarily as an environment for computer-aided design for control engineering, it has been suggested that ANDECS could also be of considerable value as an environment for more general simulation and modelling activities [12]. ANDECS already has features which support modelling and simulation, together with integrated database, manipulation and visualisation facilities. The ANDECS system is modular in nature and through its multi-objective programming system could allow clear distinctions to be made between the processes of model creation, execution and experimentation. Other interesting features of ANDECS include a translator for the commercial simulation language ACSL and code generation by the Dymola modelling language. A module has also been developed for the system to allow MATLAB routines to be called from ANDECS. Other environments have, of course, been developed elsewhere for control systems design and for other engineering design applications. A review of the type of facilities available within these might well be a revealing and useful exercise

5. CONCLUSIONS.

Support environments for simulation model development are not yet ideal in terms of the integration of facilities needed for the handling of experimental response data, the application of system identification and parameter estimation methods and the application of analysis and optimisation tools. The type of support environment required should not only be designed to meet these needs, which are specific to the model development cycle, but should also have some more general attributes defined by MacFarlane [10] for other intensively-supportive environments. These attributes include facilities to allow the user to build, analyse, browse, search, compare and evaluate, reason and hypothesise, synthesise, design, manipulate and modify, experiment, catalogue, store and retrieve. Experience in the helicopter application suggests that further enhancement of the computing environment, especially in terms of the man-machine interface and the proper integration of simulation facilities with other features such as databases, could make the processes of model validation more straightforward and thus less likely to be neglected.

References

[1] Spriet, J.A. and Vansteenkiste, G.C. "Computer-Aided Modelling and Simulation", Academic Press, London, 1982.

[2] S.C.S. Technical Committee on Model Credibility, Terminology for model credibility, Simulation, 32, pp103-4, 1979.

[3] Murray-Smith, D.J., A review of methods for the validation of continuous system simulation models, in Nock, K.G. (editor), "UKSC '90: Proceedings of the 1990 UKSC Conference on Computer Simulation", United Kingdom Simulation Council, pp108 -111, 1990.

[4] Murray-Smith, D.J., Problems and prospects in the validation of dynamic models, in Sydow, A (ed.), "Computational Systems Analysis 1992", Elsevier, Amsterdam, pp21-27 1992.

[5] Bradley, R., Padfield, G.D., Murray-Smith, D.J. and Thomson, D.G., Validation of helicopter mathematical models, Transactions of Institute of Measurement and Control, 12, pp186-196, 1990.

[6] Gray, G.J. and Murray-Smith, D.J., The external validation of nonlinear models for helicopter dynamics, in Pooley, R. and Zobel, R "UKSS '93: Proceedings 1993 Conference of the United Kingdom Simulation Society", United Kingdom Simulation Society, pp143-147, 1993.

[7] Padfield, G.D. and DuVal, R.W., Application areas for rotorcraft system identification: simulation model validation, AGARD Lecture Series Vol. LS-178 ("Rotorcraft System Identification"), Paper 12, AGARD, 1991.

[8] Murray-Smith, D.J., Modelling aspects and robustness issues in rotorcraft system identification, AGARD Lecture Series Vol. LS-178 ("Rotorcraft System Identification"), Paper 7, AGARD, 1991.

[9] MacFarlane, A.G.J., Grübel, G. and Ackermann, J., Future design environments for control engineering, Automatica, 25, pp165-176, 1989.

[10] MacFarlane, A.G.J., Interactive computing: a revolutionary medium for teaching and design, Computing and Control Engineering Journal, 1, pp149-158, 1990.

[11] Grübel, G., Joos, H.-D., Finsterwalder, R. and Otter, M., The ANDECS design environment for control engineering, Technical Report TR R79-92, DLR -Oberfaffenhofen, D-8031 Oberfaffenhofen, Germany, July 1992.

[12] Jopling, C.P. and Grübel, G., The value-added simulation facilities provided by the ANDECS control design environment, Institution of Electrical Engineers Colloquium, London, 3rd. November 1993, (IEE Colloquium Digest No. 1993/201, pp.2/1-2/6).

PETRI NETS AND AI IN MODELING AND SIMULATION

András JÁVOR

KFKI Research Institute for Measurement and Computing Techniques
H-1525 Budapest, P.O.Box 49., Hungary

Abstract. The application of Petri nets and AI to enhance the effectivity of modeling and simulation is envisaged. The combined use of both disciplines enabling the dynamic control of simulation by intelligent demons as well as the realization of mobile knowledge bases attached to the tokens by the introduction of Knowledge Attributed Petri Nets are outlined. The principles described are illustrated on practical examples.

1. INTRODUCTION

In recent decades both the disciplines of artificial intelligence and Petri nets (PN) have acquired considerable attention among people engaged in R&D activities. It is interesting to note that their origins in time are very close to each other. The term *artificial intelligence* (AI) was introduced by Minsky in 1961 [1]. The origin of Petri nets, on the other hand date back to 1962 when C.A. Petri introduced the concept in his Ph.D. dissertation [2].

Since that time both disciplines have developed extensively and rather independently of each other. The directions of investigations in each of the fields are rather diverse ranging from strictly theoretical to those aiming at practical applications [3] [4]. Considering the problems dealt with; AI is intended for the solution of tasks for which no predetermined sequence of problem solving operations exist, such as: theorem proving, games, robotics, vision, natural language processing, knowledge engineering. In the case of Petri nets - beyond the field of formal language theory - their main application has been in modeling and simulation of various complex systems [3] as e.g.: flexible manufacturing systems, computer networks and communication protocols, computer systems (hardware and software), logistics, transportation, management, etc.

It is interesting to note, that both AI and PN - although from different starting points - aim at the achievement of the most general problem solving tools possible equivalent to human problem solving capabilities. AI is intended to implement methods enabling computers to do the job where human intelligence has been required. On the other hand the R&D work in the field of PN resulted in numerous different *High Level Petri Nets* (as e.g. colored, delayed, stochastic, numerical, object, knowledge attributed Petri nets) as a further develop-ment of the original idea [5] [6]. Among high level Petri nets there are such that possess the modeling power of Turing machines [7] that - as it was stated in the Theorem of Church - are able to solve problems as generally as human thinking procedures can undertake it.

In recent years artificial intelligence as well as Petri nets have had a great impact on the simulation tools and methodologies. It is interesting to have a look on the "cross fertilization" of them.

2. BASIC WORLD VIEWS AND METHODS

Let us have a quick glance on Petri nets - including those high level versions that have been successfully applied in simulation - without going into details. The keyword for describing a PN is *directed graph* (see Fig. 1/a) providing for the *structural description* of real world models where *parallel events and processes* occur and have to be simulated. The nodes of the graph are *places* (depicted as circles) and *transitions* (depicted as bars) interconnected in an alternating way, i.e., no two nodes of the same type can be interconnected directly.

The state of a PN is characterized by its *marking,* i.e., the distribution of *tokens* (depicted as dots in Fig. 1/a) in the places. The trajectory of a PN in the *time-state space* is described by the sequence of markings called the *token play*. Changes in the markings are caused by the *firing* of the transitions. A transition fires if it is *enabled* in case the required conditions supplied by the states of the places (i.e., the required numbers of tokens) connected to it are fulfilled. Firing destroys tokens in the places connected to the inputs of the transition and creates new ones in the places connected to its output. Depending on the type of PN it may be that;

- Different types (colors) of tokens may be in the places. Accordingly the firing conditions may include prescribed numbers of various types that have to be available in the respective input places. This may also apply to the numbers of destroyed and created tokens.
- The tokens may have attributes that can be tested as firing conditions by the transition of which input places they are, and modified in the newly created tokens in case of firing.
- The places may have limited capacities for each token type that has to be considered as an additional condition for firing (i.e., whether the token creating operation can be executed).
- Complex logic and arithmetic conditions can be assigned to the transitions as firing conditions.
- Deterministic or stochastic time delays can be assigned to the nodes (in some cases to the transitions in others to the places).

In general it can be stated that the *places* contain the *states* as conditions and consequences of the *actions* performed by the *transitions*.

An extremely important property of PNs is that they enable an easy and natural mapping of real world systems preserving not only their behavioural properties but also their structure on an abstract model level. Thereby more insight can be given into the operation and interaction of the subsystems of the model as a whole.

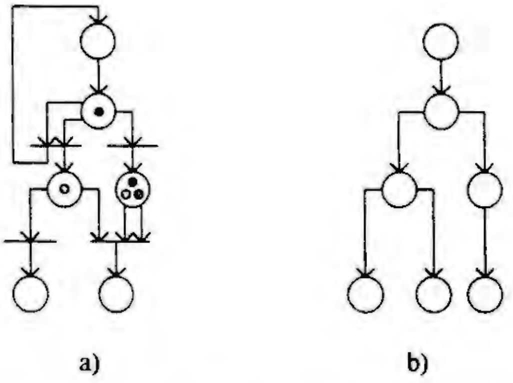


Figure 1   Graph representations of   a) Petri nets, b) knowledge bases

It is very interesting to have a brief look on some tools and methods of AI. Let us consider the aspects of knowledge bases, search procedures and inferencing.

With regard to *knowledge bases* it is interesting to note that one of the most important forms of knowledge representation is in the form of *frames*. Knowledge can be stored in graphs that have in many cases tree structure (see Fig. 1/b). Information can be stored and manipulated in the frames located in the nodes of the graph and the search for solving the problems can be undertaken by searching in the tree using various techniques as depth first, width first, hill climbing, best first strategies, etc. [4]. This representation of knowledge can be regarded as a *model* of the knowledge that shows an interesting analogy to the PN graph model representations. The search during problem solving in AI is performed in the state space while the trajectory of the simulation is in the time-state space. The basic difference is that - usually - there is no time dimension involved in the knowledge space.

Another important aspect of AI is *inferencing*. A well-known way for solving this is the application of *production rules*. Here a pattern of facts is given to the rules and if the expressions in the rules are fulfilled; they "fire" and a conclusion (a new fact) is created that may be used in further inferencing by other production rules. This resembles directly to the operation of Petri nets. It is also easy to model them directly. The production rules can be mapped into transitions while the places may represent the facts the fulfilment of which being signalized by the presence of tokens. As it is well-known most - AI based - expert systems can also provide some measure of certainty of their decision. This can also be easily solved by using high level Petri nets as, e.g., *numerical PNs* where the token attributes may carry this information.

3. COMBINATION OF ARTIFICIAL INTELLIGENCE AND PETRI NETS

Let us envisage how the two tools can be combined to increase the effectivity of simulation. In the following, two solutions shall be dealt with which can also be applied together.

3.1. Intelligent Demon Controlled Simulation [8] [9]

In conventional programs the procedures are activated by other procedures calling them. Demons on the other hand are monitoring the situation continuously and in case a certain special predefined pattern occurs; they are activated and perform their activity. Demons are used in various fields including AI [4]. Let us now use special intelligent *demons* equipped with knowledge bases and inference engines. The goal to be achieved is to automate the - usually iterative - process of simulation. The application of simulation can generally be regarded as an iterative process where a series of simulation runs with modified model parameters, structures and/or experimental conditions are undertaken until the goals are reached. These goals may be the determination of the appropriate model corresponding to given requirements or the behaviour of it. The efficiency of this process can be increased considerably by applying intelligent demons to supervise the trajectory of the simulation experiment, evaluate it and execute the necessary interventions (i.e., modification of model structure and parameters or the experimental conditions respectively). As can be seen in Fig. 2 not only a single intelligent demon can be used to control the experiment. This distributed control of the experiment can be more effective than a centralized one, as has already been shown empirically.

The realization of this principle can be undertaken easily in an object oriented way meaning that the model consists of a network of interconnected modules and the demons themselves can also be implemented as objects. Petri nets lend themselves naturally to be used as such models. The nodes (i.e., places and transitions) can be implemented as objects and the demons can easily change the structure and parameters of the model.

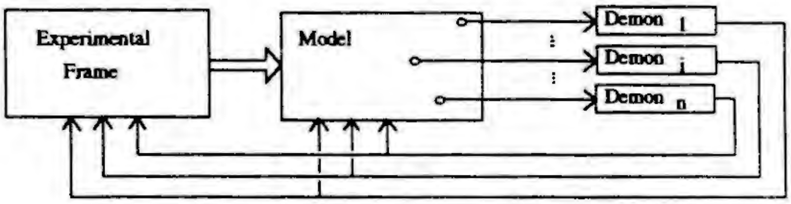


Figure 2 Principle of demon controlled simulation

3.2. Knowledge Attributed Petri Nets [6]

As it was mentioned already earlier in high level Petri nets, attributes can be assigned to the tokens. In Knowledge Attributed Petri Nets (KAPN) the tokens may also have *knowledge bases* as their attributes. This enables the inclusion of *mobile knowledge bases* in the models. The knowledge bases can be implemented using object oriented techniques and in many cases in form of frames proposed by Minsky [10]. A frame is a data-structure for representing a stereotyped situation. Attached to each frame are several kinds of information. Some of this is about how to use the frame. Some is about what one can expect next. Some is about what to do if these expectations are not confirmed. We can think of a frame as a network of nodes and relations. It is also possible that a set of tokens may use the same knowledge base frame as their attribute that can be accessed by individual pointers that can point to various parts of the frame network attached to the individual tokens.

4. EXAMPLES FOR UTILIZING PN WITH AI

The principles outlined above are illustrated on examples run on the CASSANDRA (Cognizant Adaptive Simulation System for Applications in Numerous Different Relevant Areas) simulation system version 3.0 where the demon control of simulation experiments has been implemented. The models in the system are built from Knowledge Attributed Petri Nets internally and user modules can be built from their subnetwork that can be stored in libraries and used in various fields [11] [12] [13].

In Fig. 3 a simple producer-consumer model is shown where the "producer" (formed from p6-t6) supplies products to the "market" (p7) to which a set of consumers (p1-t1 through p5-t5) can be connected. Demon D1 monitors the contents of the market and in case the average contents increases (i.e., there is an overproduction) it connects more "consumers" to the market. In case the average contents decreases, the demon disconnects "consumers". Thus the equilibrium state can be determined, i.e., how many consumers can be satisfied. Here - as can be seen - the demon undertakes structural modifications of the model. It is clear that similar equilibrium could also be reached if it would be given the task to modify production or consumption rates by changing delay values i.e. parametrical changes.

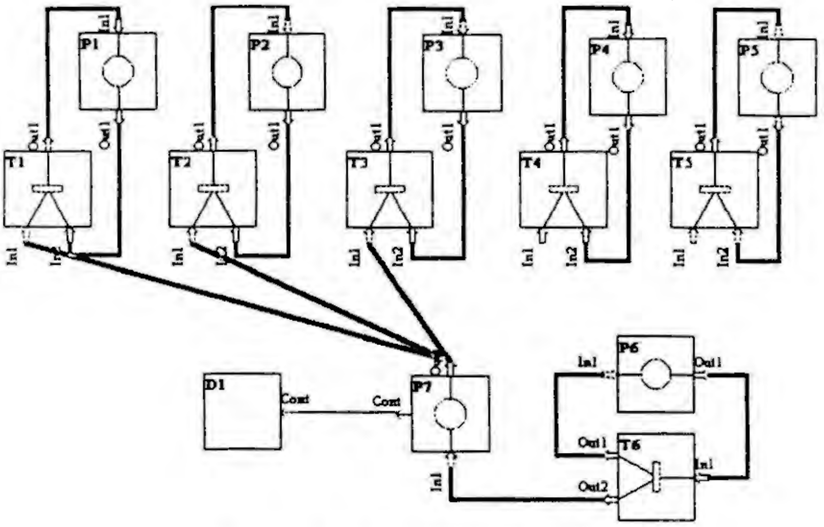
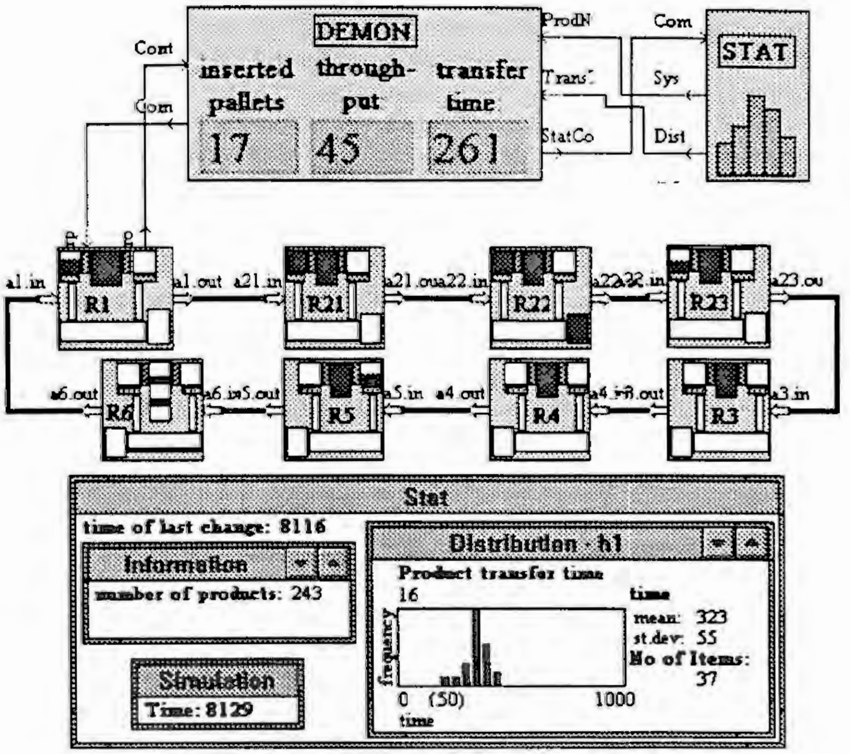


Figure 3 Producer-consumer model

Another model can be seen in Fig. 4 [14]. Although it is not shown in the user level display of the animated run, internally the building blocks are built from KAPNs and the workpieces are represented by Knowledge Attributed Tokens (KAT). Here a flexible manufacturing system (FMS) is simulated with eight workstations, each consisting of a conveyor belt for advancing pallets and a robot, with an input and output buffer before and after it. If the arriving workpiece on the pallet does not have to be processed by the given robot, or the input buffer is full, it proceeds further on the conveyor belt and leaves the station. Otherwise the pallet is shifted to the input buffer, and after the required operation has been performed by the robot, it proceeds to the conveyor belt through the output buffer. (The FMS system we have investigated has been described in EUROSIM Simulation News Europe, No. 1.) The building blocks represent the workstations together with their respective conveyor belt segments. In each station the conveyor belt segment is represented by a horizontal rectangle, the branching conveyor belt segments by vertical rectangles and the input buffer, robot and output buffer are shown by rectangles between the two branching conveyor belt segments. The mentioned rectangles representing the various parts of the building elements are colored white if empty and they become red to the extent of their relative occupation as a result of animation during the dynamic simulation run.

The eight robots perform various processes and have different parameters. Robot1 places unprocessed workpieces on empty pallets and removes the finished ones from the pallets. Robot21, Robot22 and Robot23 can perform the same process (processA), while Robot3, Robot4 and Robot5 can undertake different processes (processB, processC and processD). Robot6 is a special robot, which can substitute any of Robot3, Robot4 or Robot5.

The technological prescription for manufacturing in the given example was the following: an unprocessed workpiece can either be processed first by processA then in arbitrary order, or processA has to be the last after the other three. The workpieces on the pallets are circulating in the system until they are not completed, then Robot1 replaces them with unprocessed ones. The technological prescription is described in a knowledge base together with the current state of manufacturing attached to the individual workpieces in the system. The frame describing the technological prescription is shown in Fig. 5. The manufacturing system performance is characterized by the total throughput and the average transfer time of the workpieces. The goal is to optimize the number of the pallets in the system.



Stat

time of last change: 8116

Information

number of products: 243

Simulation

Time: 8129

Distribution - h1

Product transfer time

16

time

mean: 323

st.dev: 55

No of Items: 37

Figure 4 FMS model

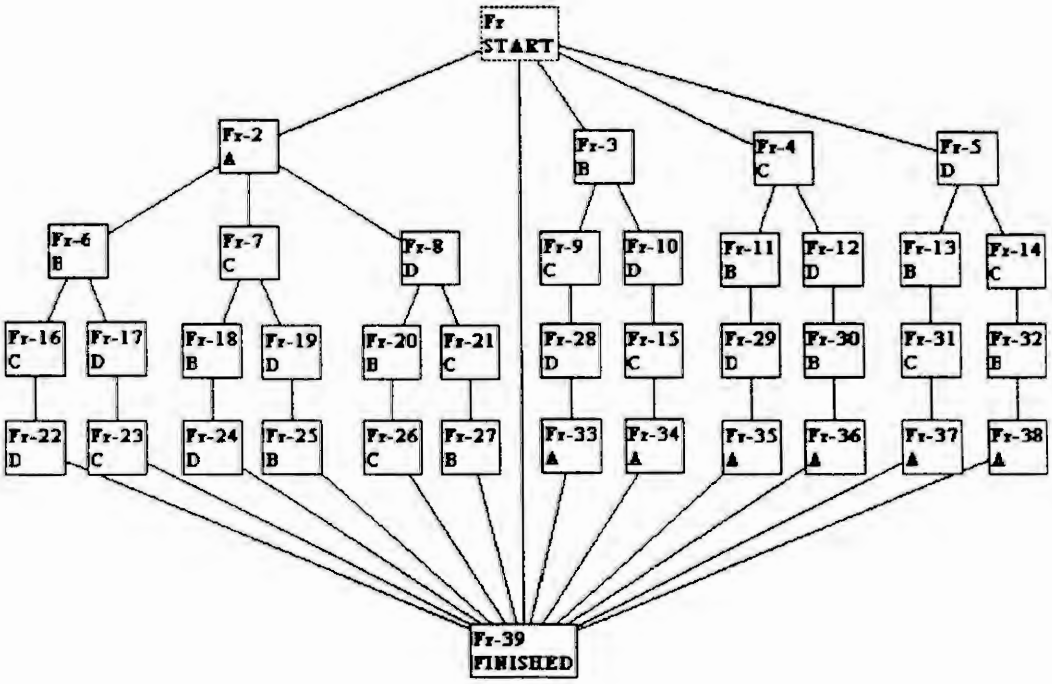


Figure 5 Technological description stored in knowledge base

Using CASSANDRA 3.0 optimization can easily be performed by applying demons for intelligent experiment control. In this case the optimization is fast and direct. Fig. 6 shows the process of such an optimization, where the optimization rule given to the demon was the following: find the number of pallets that gives the maximum total throughput, then decrease the number of pallets until reaching the 90% of the maximum in order to decrease the average transfer time. In this case the optimum number of pallets was 13.

This "optimum" is obviously an arbitrary one and in general the demon needs weighting factors (weighting the total throughput against the transfer time, e.g.), telling the demon what is to be regarded as an optimum and the demon provides the optimum number of pallets automatically.

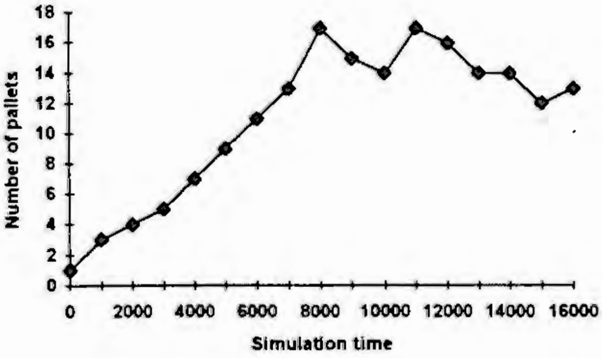


Figure 6 Optimization of FMS performance

5. CONCLUSIONS

It is a usual fact that in case principles and methods of different origins can be combined successfully their positive effects may be more than just as could be expected by adding up the components. It may be that instead of addition the results can be the multiplication of the effects, and this might hopefully also be expected in case of combining AI and PN.

6. REFERENCES

[1] Minsky, M., Steps towards Artificial Intelligence. M.I.T. 1961.

[2] Petri, C., Kommunikation mit Automaten. Ph.D. Dissertation, University of Bonn, Germany, 1962.

[3] Peterson, J.L., Petri Net Theory and Modeling of Systems. Prentice Hall, 1981.

[4] Yoshiaki Shirai, Jun-ichi Tsui, Artificial Intelligence, Concepts, Techniques and Applications. Iwanami Shoten, Publishers, Tokyo, 1982.

[5] Jensen, K., Rozenberg, G., High-Level Petri Nets. Springer Verlag, 1991.

[6] Jávor, A., Knowledge Attributed Petri Nets. Systems Analysis, Modelling, Simulation, 13(1993)1/2, 5-12.

[7] Billington, J., Wheeler, G.R., Wilburham, M.C., PROTEAN: A High-level Petri Net Tool for the Specification and Verification of Protocols. IEEE Transactions on Software Eng. 14(1988)3, 301-316.

[8] Jávor, A., Demons in Simulation: A Novel Approach. Systems Analysis, Modelling, Simulation 7(1990)5. 331-338.

[9] Jávor, A., Demon Controlled Simulation. Mathematics and Computers in Simulation. 34(1992), 283-296.

[10] Minsky, M., A Framework for Representing Knowledge in the Psychology of Computer Vision. ed.: Winston, P.H. McGraw-Hill New York 1975.

[11] Jávor, A., AI Controlled High Level Petri Nets in Simulating FMS. Fourth Annual Conference: AI, Simulation, and Planning in High Autonomy Systems, Tucson, Arizona, USA, September 20-22, 1993. 302-308.

[12] Jávor, A., Demon Controlled Simulation Models. IMACS Conference on Modelling and Control of Technological Systems, May 7-10, 1991, Lille, France, 122-127.

[13] Jávor, A., AI and Petri Nets in the Simulation of Flexible Manufacturing Systems. EUROSIM'92, Capri, 1992. 459-464.

[14] Jávor, A., Benkõ, M., Leitereg, A., Moré, G., AI Controlled Simulation of Complex Systems. Computing & Control Engineering Journal (in publication)

PROPERTIES OF MIXED SYSTEMS ARISING FROM BOND GRAPH MODELS

B.M. MASCHKE *, M. VILLARROYA**

*Lab. d'Automatisme Industriel, Conservatoire National des Arts et Métiers
21 rue Pinel, F-75013 PARIS, FRANCE
e-mail: maschke@ensam-paris.fr

**DE RENAULT Service 805 BM2
67 rue des Bons Raisins
92508 RUEIL MALMAISON CEDEX, FRANCE

ABSTRACT:

In this paper we first discuss the different assumptions used for the graphical analysis of dynamical systems arising from bond graph models. Secondly we propose a descriptor formulation of such dynamical systems defined by the constitutive relations of the elements and a basis set of constraint relation at the ports of the Simple Junction structure of a linear bond graph model. Finally we state that these systems verify the two fundamental properties: the rank identity and the dynamical degree identity.

1. INTRODUCTION

The graphical analysis of the properties of linear dynamical systems such as solvability, controllability, observability and structure at infinity [1] [2][3] [4] [5] [6] [7], relies on the assumption that the matrices defining the system are structured, i.e. that their coefficients are algebraically independent[3] [6]. In this paper we first shall discuss the different assumptions underlying the proposed graphical analysis when the dynamical systems arise from bond graph models [8] [9] [10] [11] [12]. Secondly we propose a descriptor formulation of such dynamical systems, differentiating between real valued coefficients defining constitutive relations and integer valued coefficients defining constraint relations at the ports of Simple Junction Structures, and discuss its properties in relation with mixed matrices [6] [7].

2. REAL DESCRIPTOR SYSTEMS ARISING FROM BOND GRAPH MODELS

The definition of structured systems and their associated graphical representations relies on the definition of structured real matrices [3] splitting its coefficients into fixed zeroes and non-zero reals in the following way.

Definition 1:

***A structured real matrix M** is a matrix for which some coefficients are fixed to zero and the remaining non-zero coefficients are supposed to be algebraically independent reals.*

In the frame of classical control theory, one considers then structured explicit systems defined as follows [3] .

Definition 2:

*A **structured explicit system** is defined by the the following system:*

$$\begin{cases} \dot{x} = A\,x + B\,u \\ y = C\,x \end{cases} \tag{1}$$

where: $u \in \mathbb{R}^p$, $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$ and A, B, C are structured real matrices.

To this system different graphs may be associated, representing exhaustively the structure of the system and different algorithm for determining the dynamical structural properties were proposed [2] [3] . However for systems arising from physical systems, the assumption on the (algebraic) independence of the non-zero coefficients of the matrices A, B, C is almost always false as will be illustrated on the example in figure 1 .

On figure 1 .a all the energy-storage elements are given integral causality yielding a causal conflict on the greyed 1-junction. Hence, associated with this causal conflict [13], there is a relation among the energy-variables:

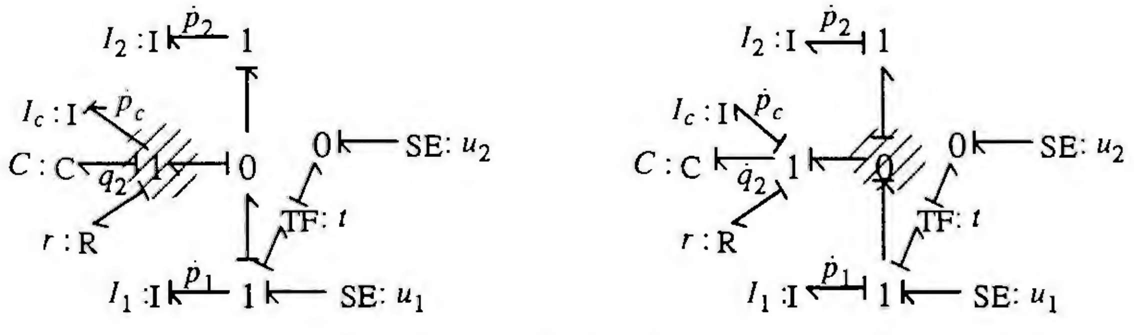
$$\frac{p_c}{I_c} = \frac{p_1}{I_1} - \frac{p_2}{I_2} \qquad . \tag{2}$$

Thus the order of the system is 2 and one may choose for instance $(q,\ p_1,\ p_2)'$ as state vector. Moreover the outputs are taken as the momenta: $(p_1,\ p_2)'$. Then the associated structured explicit system is defined by the following structured matrices:

$$A = \begin{pmatrix} 0 & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \qquad B = \begin{bmatrix} 0 & 0 \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{bmatrix} \qquad C = \begin{pmatrix} 0 & a_{c12} & 0 \\ 0 & 0 & c_{23} \end{pmatrix} \tag{3}$$

Assuming that the non-zero coefficients of these matrices are independent implies to overlook that the rank of A is generically 2 as is shown on figure 1 .b: indeed assigning differential causality to all energy storage elements yields the causal conflict on the greyed 0-junction, i.e. to the following dynamical invariant: $\dot{p}_1 + \dot{p}_2 = 0$ (4). In the same way one deduces that the rank of the decoupling matrix BC is 2 although it is actually 1: graphically the I-element labelled I_1 is a separation node on the causal connection from the 2 I-elements labeled I_1, I_2 to the input sources [11].

In order to properly take into account such structural algebraic constraints, one may consider dynamic systems described by a coupled set of differential and algebraic equations, and called singular or descriptor systems [14]. Consequently structured descriptor systems were considered and defined as follows [11].



(a) Fig. 1 . Bond graph model (b)

Definition 3:

A structured descriptor system is defined by the the following system:

$$\begin{cases} F\,\dot{x} = A\,x + B\,u \\ y = C\,x \end{cases} \qquad (5)$$

where: $u \in \mathbb{R}^p$, $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$ *and* F, A, B, C *are structured real matrices.*

Different choices of the semi-state variable x were proposed: the complete set of energy-variables [10] [11] following the usual expression of the systems equations [15] or or the complete set of energy variables and port variables at the ports of the energy storage and dissipative elements [12] however without writing the system explicitly in descriptor form of eq. (5). On the example we shall criticize the first choice of variables, similar arguments holding for the second one.

Considering the example in figure 1 . and writing the state-vector as: $x = (q,\ p_1,\ p_2,\ p_c)^t$, the associated structured descriptor system is given by:

$$F = \begin{bmatrix} f_{11} & 0 & 0 & 0 \\ 0 & f_{22} & 0 & f_{24} \\ 0 & 0 & f_{33} & f_{34} \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad A = \begin{bmatrix} 0 & a_{12} & a_{13} & 0 \\ a_{21} & a_{22} & a_{23} & 0 \\ a_{31} & a_{32} & a_{33} & 0 \\ 0 & a_{42} & a_{43} & a_{44} \end{bmatrix} \qquad B = \begin{bmatrix} 0 & 0 \\ b_{21} & b_{22} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \qquad C = \begin{pmatrix} 0 & c_{12} & 0 & 0 \\ 0 & 0 & c_{23} & 0 \end{pmatrix} \qquad (6).$$

Now the rank of the decoupling matrix is seen to be 1, indeed: $rank \left(\dfrac{\partial \dot{p}_i}{\partial u_j}\right)_{ij} = rank\ CB = 1$ (7)

which may also be seen graphically as the I-element labelled I_1 is a separation node on the causal connection from the 2 I-elements with energy variables p_1, p_2 to the input sources [11]. However the dynamic invariant is still not seen as the dynamical degree of the system: $deg_s\ det(sF - A)$ is 3 assuming the independence of the non-zero coefficients.

3. MIXED SYSTEMS ARISING FROM BOND GRAPH MODELS

Therefore to take account all the algebraic relations expressed in a bond graph model, we propose consider on one side the constitutive relations defined by real valued parameters and on the other side the constraint relations on the port variables of the Simple Junction Structure and eventually some coupling elements like the Symplectic Gyrator. This has the major consequence to distinguish among the coefficients between real-valued and numbers defined on a subfield of R. This correspond to consider so-called mixed matrices [6] and accordingly mixed descriptor systems. With such systems different graphs may be associated to analyze graphically their structural dynamic properties such as solvability, controllability/observability [6] and structure at infinity [4].

Definition 4:

*Let K be a subfield of a field F. A **mixed matrix** D with respect to F/K is matrix which may be decomposed additively as follows: $D = Q + F$* (8)

where Q is a matrix over the field K and T is a matrix over the field F such that its non-zero coefficients are algebraically independent over K.

Definition 5:

*A descriptor systems defined by eq.(5) is called **structured mixed descriptor system** if the matrices F, A, B, C are mixed matrices.*

Then it may be shown that two classes of (linear) bond graph models generate a mixed descriptor systems [16]: the first class corresponds to models of L, C, R electrical networks or 1-dimensional mechanical systems, and the second class corresponds to (linearized) chemical reactions.

Proposition 1:

*Consider a bond graph consisting of linear I, C, R elements connected by a Simple Junction Structure (respectively linear C, R elements connected by a Weighted Junction Structure with moduli being integer). Choosing as semi-state variables **the energy-variables and the port-variables** of all the elements: I, C, R, output variables and source input-variables, **the set of constitutive relations and a basis set of constraint relations** at the port of the Simple Junction Structure (respectively Weighted Junction Structure), define the equations of motion as a mixed descriptor system defined with respect to $\mathbb{R}/\{-1,\ 0,\ -1\}$ (respectively $\mathbb{R}/\mathbb{Q}$ where $\mathbb{Q}$ denotes the field of rational numbers).*

If the bond graph model contains transformers or gyrators with real-valued moduli, the set of constitutive relations of the elements and a basis set of constraint relations at the port of the Simple Junction Structure leads to a descriptor system which may decomposed additively according to eq.(8) with respect to $\mathbb{R}/\{-1,\ 0,\ -1\}$, but is not a structured mixed system. Indeed, consider the example in figure 1 ., choosing as semi-state vector:

$$x = (q,\ p_1,\ p_2,\ p_c,\ e_c,\ f_{i1},\ f_{i2},\ f_{ic},\ e_R,\ e_{1T},\ f_{2T},\ f_c\cdot f_R,\ f_{1T},\ e_{i1},\ e_{i2},\ e_{ic},\ e_{2T},\ u_1,\ u_2) \tag{9}$$

the descriptor system is defined by:

$$F = \begin{pmatrix} -s\,I_4 & 0_{4\times 16} \\ 0_{16\times 4} & 0_{16\times 16} \end{pmatrix}\ A = \begin{bmatrix} 0_{4\times 4} & 0_{4\times 4} & 0_{4\times 3} & \mathcal{P} & 0_{4\times 2} \\ Z & I_{4\times 4} & 0_{4\times 3} & 0_{4\times 7} & 0_{4\times 2} \\ 0_{3\times 4} & 0_{3\times 4} & I_{3\times 3} & \mathcal{L} & 0_{3\times 2} \\ 0_{7\times 4} & & & \mathcal{J} & \end{bmatrix}\ B = \begin{pmatrix} & 0_{18\times 20} & \\ 0_{2\times 2} & I_{2\times 2} & 0_{2\times 16} \end{pmatrix}\ C = \begin{bmatrix} & 0_{16\times 2} \\ 0_{20\times 16} & I_{2\times 2} \\ & 0_{2\times 2} \end{bmatrix} \tag{10}$$

where: Z defines the parameters of the energy-storage elements: $Z = diag\left(-\frac{1}{C},\ -\frac{1}{I_1},\ -\frac{1}{I_2},\ -\frac{1}{I_c}\right)$ (11),

$\mathcal{P}$ identifies the rate variables (i.e. time derivative of the energy-variables) with some power variables, $\mathcal{L}$ defines the constitutive parameters of the R elements and the coupling elements TF and GY and $\mathcal{J}$ defines the independent set of constraint relations on the port variables of the Simple Junction Structure:

$$\mathcal{P} = \begin{bmatrix} 1&0&0&0&0&0&0 \\ 0&0&0&1&0&0&0 \\ 0&0&0&0&1&0&0 \\ 0&0&0&0&0&1&0 \end{bmatrix},\ \mathcal{L} = \begin{pmatrix} 0&-r&0&0&0&0&0 \\ 0&0&0&0&0&0&-t \\ 0&0&-t&0&0&0&0 \end{pmatrix},\ \mathcal{J} = \begin{bmatrix} 0&1&-1&0&0&0&0&1&0&0&0&0&0&0 \\ 0&1&-1&0&0&0&0&0&1&0&0&0&0&0 \\ 0&-1&0&0&0&0&0&0&0&1&0&0&0&0 \\ 0&1&-1&1&0&0&0&0&0&0&0&0&0&0 \\ -1&0&0&0&-1&0&0&0&0&0&1&0&-1&1 \\ 1&0&0&0&1&0&0&0&0&0&0&1&1&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&1 \end{bmatrix} \tag{12}$$

It is clear that the matrix $\mathcal{L}$ has 2 identical coefficients corresponding to the two adjoint constitutive relations of the transformer. Hence the real coefficients of A are not independent (over any subfield of $\mathbb{R}$) and the descriptor system is not defined on mixed matrices.

Therefore it might seem that the analysis proposed in [6] [4] fails for descriptor systems arizing for such bond graphs. However it may be proven that the two basic properties underlying the whole analysis proposed in [6] [4] still remain true [16].

Proposition 2:

*Consider a bond graph consisting of linear I, C, R, TF, GY elements connected by a Simple Junction Structure. Choosing as semi-state variables **the energy-variables and the port-variables** of the elements: I, C, R, TF, GY, output variables and source input-variables, **the set of constitutive relations and a basis set of constraint relations** at the port of the Simple Junction Structure, define the equations of motion as a descriptor system (5) where the matrices F, A, B, C may be decomposed additively according to (8) with coefficients in the field $\mathbb{R}$ or in the field $\{-1,\ 0,\ -1\}$.*

*Any of these matrices or matrix composed of these matrices, denoted by M, denoting the additive decomposition with respect to $\mathbb{R}/\{-1,\ 0,\ -1\}$ by: $M = Q_M + T_M$ verifies **the rank identity**:*

$$rank\ M = \max\{r(Q_M[R\backslash I,\ C\backslash J]) + t(T_M[I,\ J])\ \mid\ I \subset R,\ J \subset C\} \tag{13}$$

where r is the rank function of matrices, t is the term-rank of a matrix, R is the set of rows and C is the set of *columns of the matrix M and the* ***dynamical degree identity****:*

$$\delta(M) = \max\{\delta(Q_M[R\backslash I,\ C\backslash J]) + \delta(T_M[I,\ J]) \mid I \subset R,\ J \subset C\} \tag{14}$$

where δ denotes the dynamical degree of a matrix: $\delta(M) = deg_s\ det(M)$.

4. CONCLUSION

We have proposed here a descriptor formulation defined by the constitutive relations of the elements and a basis set of constraint relation at the ports of the Simple Junction structure of a linear bond graph model. We have shown that although this system may be decomposed additively with respect to the fields $R/\{-1,\ 0,\ -1\}$, but is not not a mixed descriptor system as soon as it contains transformers or gyrators with real moduli. This is due to the two adjoint constitutive relations of these elements. Finally we have stated that the two fundamental properties (the rank identity and the dynamical degree identity), allowing a graphical analysis of the properties of such systems, are still verified for the proposed descriptor formulation.

These results allow first to apply the algorithms proposed in [6] [7] order to investigate the properties, such as solvability, controllability and structure at infinity, of dynamical systems arising from bond graph models and secondly to derive and give an exact basis to the causal algorithm applied directly to the bond graph model [16].

Finally this formulation may also be extended to bond graphs containing multiports subject to causality restriction at their ports by defining the appropriate rank functions and the extension to nonlinear systems will also be treated in next future continuing [10] [11].

REFERENCES

[1] C.T.Lin, "Structural controllability", IEEE Trans. on Automatic control, Vol.19, n°3, pp.201-208, June 1974

[2] A.Linneman, "Decoupling of structured systems", Systems and Control Letters, Vol.1, n°2, August 1981

[3] K.J.Reinschke, "Multivariable control: a graph-theoretic approach", Lecture Notes on Control Information Sci., Springer-Verlag, 1988

[4] J.W. van der Woude, "On the structure at infinity of a structured system", Report BS-R8918, Center for Mathematics and Computer Science, Amsterdam, 1989

[5] C.Commault, J.-M.Dion and A.Perez, "Disturbance rejection for structured systems", IEEE Trans. on Automatic Control, Vol.36, n°7, July 1991

[6] K.Murota, **"Systems Analysis by Graphs and Matroids - Structural Solvability and Controllability"**, Algorithms and Combinatorics, Vol.3, Springer Verlag, 1987

[7] K.Murota and J.W. van der Woude, "Disturbance decoupling and structure at infinity of structured descriptor systems", Report 89605-OR, Institut fur Okonometrie und Operations Research, Universitaet Bonn, 1989

[8] N.Suda and T.Hatanaka, "Structural properties of systems represented by bond graphs", in "Complex and Distributed Systems: Analysis, Simulation and Control", S.G.Tsafestas and P.Borne eds., pp.73-80, Elsevier, Amsterdam, 1986

[9] A.Zeid, "Some bond graph structural properties: eigenspectra and stability", Trans. ASME, J. Dynamic Systems, Meas. and Control, Volo. 111, pp.382-388, 1989

[10] B.M.Maschke, A.Yazman, *"Graphical tools to analyze nonlinear controlled systems: Bond-Graphs and System-Graphs"* in **"Modelling and Simulation of Systems"**, P.Breedveld, G.Dauphin-Tanguy and P.Borne (eds.), Trans. of 12th IMACS World Congress on Scientific Computation (July 1988, Paris, France), J.C. Baltzer AG, Scientific Publishing Co., 1989, pp. 35-40

[11] B.M.Maschke, *"Bond-Graphs for the structural dynamic decoupling problem"*, in **"Bond graphs for Engineers"**, G.Dauphin-Tanguy and P.C.Breedveld eds., Proc. 13th IMACS World Congress on Computational and Applied Mathematics, Dublin, Ireland, 22-26 July 1991 and the IMACS Conference on Modelling and Control of Technological Systems, Lille, France, 7-10 May 1991

[12] C.Sueur and G.Dauphin-Tanguy, "Bond graph approach for the structural analysis of MIMO linear systems", J. of the Franklin Institute, vol.328, n°6, pp. 55-70, 1991

[13] Bidard C., "Displaying Kirchhoff's invariants in simple junction structures", in **"Bond graphs for Engineers"**, G.Dauphin-Tanguy and P.C.Breedveld eds., Proc. 13th IMACS World Congress on Computational and Applied Mathematics, Dublin, Ireland, 22-26 July 1991 and the IMACS Conference on Modelling and Control of Technological Systems, Lille, France, 7-10 May 1991

[14] L.Dai, "Singular control systems", Lecture Notes in Control and Inf. Sciences, n°118, Springer Verlag, 1989

[15] R.C. Rosenberg, "Exploiting bond graph causality in physical systems models", Trans. ASME, J.Dyn.Syst.Meas and Control, Vol. 109, pp. 378-383, 1987

[16] M.Villarroya, "Analyse structurelle graphique des propriétés des systèmes dynamiques générés par la modélisation en graphes de liaison", Mémoire d'Ingénieur, CNAM, Paris, 1993

MULTIBOND-GRAPH REPRESENTATION OF LAGRANGIAN MECHANICS: THE ELIMINATION OF THE EULER JUNCTION STRUCTURE

Peter BREEDVELD† and Neville HOGAN††

†University of Twente, Electrical Engineering Department, Control Laboratory,
P.O. Box 217, 7500 AE Enschede, NL, tel.: + 31 53 89 27 92, fax: + 31 53 34 00 45, email: brv@rt.el.utwente.nl
††Massachusetts Institute of Technology, Department of Mechanical Engineering, 77 Massachusetts Avenue,
Cambridge, Mass. 02139, USA, tel.: + 1 (617) 253 2277, fax: + 1 (617) 258 5802, email: neville@mit.edu

Abstract. For a multibond graph representing the rotation of a rigid body we show that transformation of both the 3-port inertia and the 3-port nonlinear gyrator ('Euler Junction Structure') over the modulated transformer representing the coordinate transformation from generalized coordinates, like Euler, Cardan or Bryant angles, to body-fixed coordinates (e.g. the principle axes of the body), results in a proper 6-port storage element for the kinetic energy, of which 3 ports have an 'I-nature' and the other 3 ports a 'C-nature'.

1. INTRODUCTION

Several ways to describe rigid-body mechanics in terms of (multi-)bond graphs have appeared in the literature [9, 1, 2, 12]. These descriptions generally represent a Newton-Euler approach. In this paper we show that the various formalisms to describe mechanisms and rigid-body systems in particular, such as the formalisms of (Newton-)Euler, (Euler-)Lagrange and Hamilton, can all be represented in multibond graph form [7] and that their relationships are relatively simple multibond graph operations. Especially the rotation of a rigid body will be elaborated.

This again illustrates that bond graphs are *not* some alternative or even competitive (modeling) technique, as they are often depicted, but a graphical model description formalism, able of representing various techniques to describe physical models and the relationships between these techniques. As a consequence, this representation may increase not only insight in the structure of the physical system and its behavior(s), but also, by virtue of its representation of computational causality, support an efficient choice of numerical computation algorithms.

2. A BOND GRAPH 'PREVIEW'

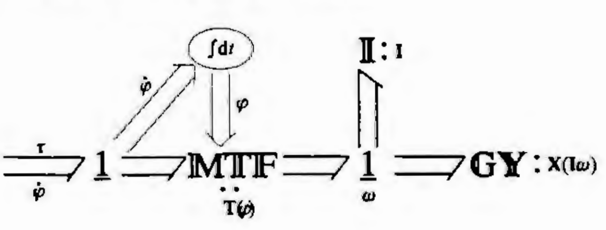


Figure 1: Multibond graph representation of Euler's equations for a rotating rigid body

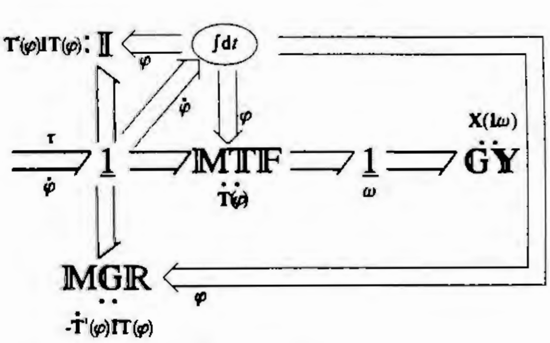


Figure 2: Virtual inertia and gyristor

We start with a frequently used multibond graph model of the spatial rotation of a rigid body that corresponds to the (Newton-)Euler formalism (figure 1) [2]. It consists of a 3-port I-element characterized by the inertia tensor I of the body and the corresponding 3-port nonlinear gyrator ('Euler Junction Structure' or EJS) which represents the gyroscopic effects. This 3-port GY is also, but in a nonlinear way, characterized by the inertia tensor. Both elements are connected to a 1-junction array representing the rotational velocity ω in body-fixed coordinates. In order to be able to represent or compute the position of the rigid body, it is required that the rotational velocity ω is transformed from the body-fixed coordinates to generalized coordinates φ, like the Euler angles, Cardan angles, Bryant angles, etc., in order to obtain an integrable form of the rotational velocity:

$$\omega = \mathbf{T}(\varphi)\frac{\mathrm{d}\varphi}{\mathrm{d}t} = \mathbf{T}(\varphi)\dot{\varphi} \qquad (1)$$

The modulated multiport transfomer in figure 1 represents this coordinate transformation of the velocities. It is state-modulated due to $\mathbf{T}$ being dependent on φ. These angles can be obtained by integrating the rotational velocity $\dot{\varphi}$ expressed in generalized coordinates.

Starting from this bond graph we show that transformation of the 3-port I-element over the 2x3-port MTF leads to a virtual inertia I^* and a 3-port gyristor MGR as described by Allen [1], both modulated by φ (figure 2). The nonlinear 3-port GY (or EJS) in figure 1 has been described as a modulated multiport gyrator (3-port MGY) due to the presence of 2-port modulated gyrators in one of its decompositions (figure 3) [2]. The EJS was even introduced in this immediate canonical decomposition [6] — hence the terminology *junction structure* — [9], because the concept of the multiport gyrator had not yet been introduced in the literature at the time of introduction of the EJS [4].

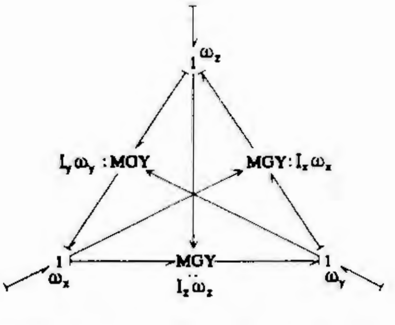
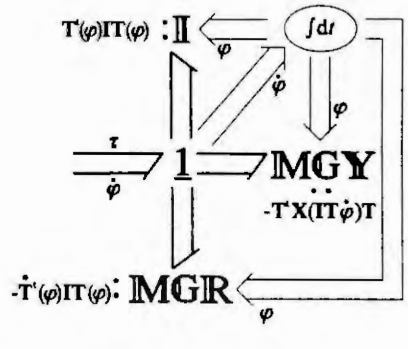


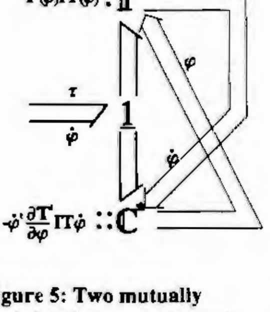
Figure 3: Decomposition of the Euler Junction Structure



$\mathbf{T}'(\varphi)\mathbf{I}\mathbf{T}(\varphi)$:II
$\int dt$
φ
$\dot{\varphi}$
φ
τ
1
MGY
$\dot{\varphi}$
$-\mathbf{T}'\mathbf{X}(\mathbf{I}\mathbf{T}\dot{\varphi})\mathbf{T}$
$-\dot{\mathbf{T}}'(\varphi)\mathbf{I}\mathbf{T}(\varphi)$: MGR
φ

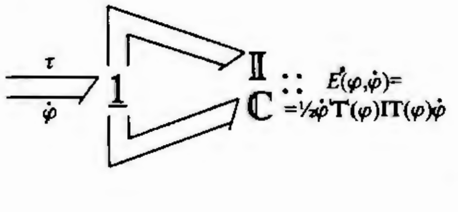
Figure 4: Transformation of the EJS

Transformation of this nonlinear 3-port GY over the state-modulated MTF representing the coordinate transformation leads to a state-modulated 3-port MGY (figure 4) [2]. The crucial step presented in this paper is that the combined port behavior of the MGY and the MGR is equivalent with the port-behavior of a flow-modulated 3-port C-element. This means that the resulting elements are a 3-port I, modulated by the state variable of this 3-port C, which, in turn, is modulated by the output of the 3-port I-element (Figure 5). The best way to eliminate these modulations that violate the nature of a storage element, is to consider both elements together as a 6-port storage element with a mixed nature (Figure 6). Use of a symplectic gyrator array as proposed in the Generalized Bond Graph notation to represent the coupling between the kinetic and the potential domains explicitly, results in one 6-port C-element, of which the constitutive equations obey the Maxwell reciprocity conditions (Figure 7), because the energy stored in this element corresponds to a proper energy function, the Hamiltonian [5,11].



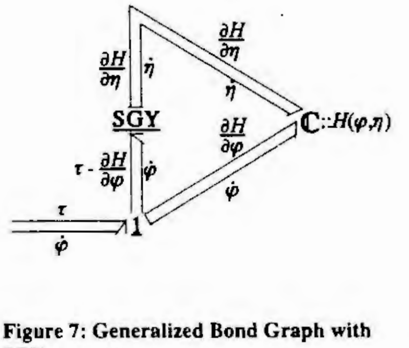
$\mathbf{T}'(\varphi)\mathbf{I}\mathbf{T}(\varphi)$: II*
φ
τ
1
$\dot{\varphi}$
$\dot{\varphi}$
$-\dot{\varphi}'\frac{\partial\mathbf{T}'}{\partial\varphi}\mathbf{I}\mathbf{T}\dot{\varphi}$ ::C*

Figure 5: Two mutually modulated storage elements



τ
1
$\dot{\varphi}$
II
C
$E(\varphi,\dot{\varphi})=$
$=\frac{1}{2}\dot{\varphi}'\mathbf{T}(\varphi)\mathbf{I}\mathbf{T}(\varphi)\dot{\varphi}$

Figure 6: Nonlinear 6-port storage element



$\frac{\partial H}{\partial \eta}$
$\frac{\partial H}{\partial \eta}$
$\dot{\eta}$
$\dot{\eta}$
SGY
$\frac{\partial H}{\partial \varphi}$
C::$H(\varphi,\eta)$
$\tau - \frac{\partial H}{\partial \varphi}$
$\dot{\varphi}$
$\dot{\varphi}$
τ
1
$\dot{\varphi}$

Figure 7: Generalized Bond Graph with SGY array

In the next section we show this process in terms of the equations, both to support the conclusions, but also to illustrate that in the above discussion we made one silent assumption that is not always satisfied: the regularity of the transformation matrix $\mathbf{T}(\varphi)$. This transformation becomes singular in one particular point, which correspond to a physically interpretable situation called gimbal-lock, i.e. two of the three coordinate-axes line up.

3. THE UNDERLYING EQUATIONS

Using the basic relation (e.g. [8,10]):

$$\left(\frac{\mathrm{d}\mathbf{a}}{\mathrm{d}t}\right)_\mathrm{s} = \left(\frac{\mathrm{d}\mathbf{a}}{\mathrm{d}t}\right)_\mathrm{b} + \omega \times \mathbf{a} = \left(\frac{\mathrm{d}\mathbf{a}}{\mathrm{d}t}\right)_\mathrm{b} + \mathbf{X}(\mathbf{a})\omega = \left(\frac{\mathrm{d}\mathbf{a}}{\mathrm{d}t}\right)_\mathrm{b} - \mathbf{X}(\omega)\mathbf{a} \tag{2}$$

where **a** is an arbitrary vector, ω the rotational velocity of the body-fixed frame and the subscripts s and b stand for spatial and body-fixed coordinates respectively, the time derivative of $\mathbf{T}(\varphi)$ can be found:

$$\left(\frac{\mathrm{d}\mathbf{T}}{\mathrm{d}t}\right)_\mathrm{b} = \left(\frac{\mathrm{d}\mathbf{T}}{\mathrm{d}t}\right)_\mathrm{s} - \mathbf{X}(\mathbf{T})\omega = \frac{\partial \mathbf{T}}{\partial \varphi}\frac{\mathrm{d}\varphi}{\mathrm{d}t} - \mathbf{X}(\mathbf{T})\omega = \frac{\partial \mathbf{T}}{\partial \varphi}\dot{\varphi} + \mathbf{X}(\omega)\mathbf{T} = \frac{\partial \mathbf{T}}{\partial \varphi}\dot{\varphi} + \mathbf{X}(\mathbf{T}\dot{\varphi})\mathrm{T} \tag{3}$$

or

$$\left(\frac{\mathrm{d}\mathbf{T}}{\mathrm{d}t}\right)_\mathrm{b} - \mathbf{X}(\mathbf{T}\dot{\varphi})\mathbf{T} = \frac{\partial \mathbf{T}}{\partial \varphi}\dot{\varphi} \tag{4}$$

hence

$$\left(\frac{\mathrm{d}\mathbf{T}^\mathrm{t}}{\mathrm{d}t}\right)_\mathrm{b} - \mathbf{T}^\mathrm{t}\mathbf{X}^\mathrm{t}(\mathbf{T}\dot{\varphi}) = \left(\frac{\mathrm{d}\mathbf{T}^\mathrm{t}}{\mathrm{d}t}\right)_\mathrm{b} + \mathbf{T}^\mathrm{t}\mathbf{X}(\mathbf{T}\dot{\varphi}) = \dot{\varphi}^\mathrm{t}\frac{\partial \mathbf{T}^\mathrm{t}}{\partial \varphi} \tag{5}$$

If (2) is applied to **I**ω, where **I** is the inertia tensor, we obtain Euler's equation for the rotation of a rigid body:

$$\left(\frac{\mathrm{d}\mathbf{I}\omega}{\mathrm{d}t}\right)_\mathrm{s} = \left(\frac{\mathrm{d}\mathbf{I}\omega}{\mathrm{d}t}\right)_\mathrm{b} + \omega \times \mathbf{I}\omega = \left(\frac{\mathrm{d}\mathbf{I}\omega}{\mathrm{d}t}\right)_\mathrm{b} + \mathbf{X}(\mathbf{I}\omega)\omega = \left(\frac{\mathrm{d}\mathbf{I}\omega}{\mathrm{d}t}\right)_\mathrm{b} - \mathbf{X}(\omega)\mathbf{I}\omega \tag{6}$$

Figure 1 shows the bond graph interpretation of (6). The effort relation of the multiport transformer characterized by $\mathbf{T}(\varphi)$ is (compare (6)):

$$\tau = \mathbf{T}^\mathrm{t}\left(\frac{\mathrm{d}\mathbf{I}\omega}{\mathrm{d}t}\right)_\mathrm{s} = \mathbf{T}^\mathrm{t}\left(\frac{\mathrm{d}\mathbf{I}\omega}{\mathrm{d}t}\right)_\mathrm{b} - \mathbf{T}^\mathrm{t}\mathbf{X}(\omega)\mathbf{I}\omega = \mathbf{T}^\mathrm{t}\left(\frac{\mathrm{d}\mathbf{I}\mathbf{T}\dot{\varphi}}{\mathrm{d}t}\right)_\mathrm{b} - \mathbf{T}^\mathrm{t}\mathbf{X}(\mathbf{T}\dot{\varphi})\mathbf{I}\mathbf{T}\dot{\varphi} \tag{7}$$

or

$$\tau = \mathbf{T}^\mathrm{t}\left(\frac{\mathrm{d}\mathbf{I}\omega}{\mathrm{d}t}\right)_\mathrm{s} = \left(\frac{\mathrm{d}\mathrm{T}^\mathrm{t}\mathbf{I}\mathbf{T}\dot{\varphi}}{\mathrm{d}t}\right)_\mathrm{b} - \left(\frac{\mathrm{d}\mathbf{T}^\mathrm{t}}{\mathrm{d}t}\right)_\mathrm{b}\mathbf{I}\mathbf{T}\dot{\varphi} - \mathbf{T}^\mathrm{t}\mathbf{X}(\mathbf{T}\dot{\varphi})\mathbf{I}\mathbf{T}\dot{\varphi} = \left(\frac{\mathrm{d}\mathbf{T}^\mathrm{t}\mathbf{I}\mathbf{T}\dot{\varphi}}{\mathrm{d}t}\right)_\mathrm{b} - \left[\left(\frac{\mathrm{d}\mathbf{T}^\mathrm{t}}{\mathrm{d}t}\right)_\mathrm{b} + \mathbf{T}^\mathrm{t}\mathbf{X}(\mathbf{T}\dot{\varphi})\right]\mathbf{I}\mathbf{T}\dot{\varphi} \tag{8}$$

The bond graph interpretation of (8) is that of figures 2 and 4. The 3-port I is transformed over the 3x3-port MTF, resulting in a virtual I^* characterzed by $\mathbf{T}^\mathrm{t}\mathbf{I}\mathbf{T}$ and a gyristor MGR characterized by $-\frac{\mathrm{d}\mathbf{T}^\mathrm{t}}{\mathrm{d}t}\mathbf{I}\mathbf{T}$ (figure 2) [1,3]. Transformation of the nonlinear 3-port GY over this MTF results in a state-modulated 3-port MGY characterized by $-\mathbf{T}^\mathrm{t}\mathbf{X}(\mathbf{T}\dot{\varphi})\mathbf{I}\mathbf{T}$ (figure 4).

Combining (5) and (8) we obtain

$$\mathbf{T}^\mathrm{t}\left(\frac{\mathrm{d}\mathbf{I}\omega}{\mathrm{d}t}\right)_\mathrm{s} = \left(\frac{\mathrm{d}\mathbf{T}^\mathrm{t}\mathbf{I}\mathbf{T}\dot{\varphi}}{\mathrm{d}t}\right)_\mathrm{b} - \dot{\varphi}^\mathrm{t}\frac{\partial \mathbf{T}^\mathrm{t}}{\partial \varphi}\mathbf{I}\mathbf{T}\dot{\varphi} \tag{9}$$

This means in bond graph terms that the effect of the latter MGY and the MGR are combined into one term. We recognize this term as the result of the (Euler-)Lagrange equation with the kinetic ' co-energy $E^*(\varphi,\dot{\varphi}) = \frac{1}{2}\dot{\varphi}^\mathrm{t}\mathbf{T}^\mathrm{t}\mathbf{I}\mathbf{T}\dot{\varphi}$:

$$\begin{aligned}&\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial E^*(\varphi,\dot{\varphi})}{\partial \dot{\varphi}}\right) - \frac{\partial E^*(\varphi,\dot{\varphi})}{\partial \varphi} = \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial\left(\frac{1}{2}\dot{\varphi}^\mathrm{t}\mathbf{T}^\mathrm{t}\mathbf{I}\mathbf{T}\dot{\varphi}\right)}{\partial \dot{\varphi}}\right) - \frac{\partial\left(\frac{1}{2}\dot{\varphi}^\mathrm{t}\mathbf{T}^\mathrm{t}\mathbf{I}\mathbf{T}\dot{\varphi}\right)}{\partial \varphi} = \\ &= \frac{\mathrm{d}}{\mathrm{d}t}\left(\mathbf{T}^\mathrm{t}\mathbf{I}\mathbf{T}\dot{\varphi}\right) - \frac{1}{2}\dot{\varphi}^\mathrm{t}\frac{\partial \mathbf{T}^\mathrm{t}}{\partial \varphi}\mathbf{I}\mathbf{T}\dot{\varphi} - \frac{1}{2}\dot{\varphi}^\mathrm{t}\mathbf{T}^\mathrm{t}\mathbf{I}\frac{\partial \mathbf{T}}{\partial \varphi}\dot{\varphi} = \frac{\mathrm{d}}{\mathrm{d}t}\left(\mathbf{T}^\mathrm{t}\mathbf{I}\mathbf{T}\dot{\varphi}\right) - \dot{\varphi}^\mathrm{t}\frac{\partial \mathbf{T}^\mathrm{t}}{\partial \varphi}\mathbf{I}\mathbf{T}\dot{\varphi} = \tau = \mathbf{T}^\mathrm{t}\left(\frac{\mathrm{d}\mathbf{I}\omega}{\mathrm{d}t}\right)_\mathrm{s}\end{aligned} \tag{10}$$

This shows that the bond graphs of the figures 5 and 6 are correct: the combination of the MGY and MGR have the nature of a 3-port C and it is more proper to combine this 3-port C with the 3-ports of the modulated virtual inertia into a 6-port IC-element.

So far we did not consider causality in the bond graphs. Inspection of the form of the equations and the fact that the Lagrangian $E^*(\varphi,\dot{\varphi}) = \frac{1}{2}\dot{\varphi}^t\mathbf{T}^t\mathbf{I}\mathbf{T}\dot{\varphi}$ characterizes this 6-port storage element, shows that we have been using differential causality of the 3-port I-element. To elaborate on the latter argument: The Lagrangian does not represent the energy stored in the 6-port storage element, but a Legendre transformation of this energy with respect to the momentum $\eta = \mathbf{T}^t\mathbf{I}\mathbf{T}\dot{\varphi}$. The proper energy is the Hamiltonian $E(\varphi,\eta) = \frac{1}{2}\eta^t\mathbf{T}^{-1}(\varphi)\mathbf{I}^{-1}\mathbf{T}^{-t}(\varphi)\eta = H(\varphi,\eta)$. This Legendre transformation replaces η by $\dot{\varphi}$ as independent variable corresponding with a change of causality of the I-ports from integral (preferred) to differential causality (figure 8). However, as well the causality of the MTF in the bond graph (figure 8a) as the form of the Hamiltonian (figure 8b) indicate that the transformation matrix **T** has to be inverted. In case of the rotation of a rigid body any choice of generalized coordinates (Euler, Cardan, Bryant or other angles) will display the effect of lining up of two of the axis in a certain situation as a singularity of the matrix **T**. In bond graph terms this singularity restricts the choice of integral causality: reaching the singular point (or approximating it in the case of numerical simulation), requires a change of causality. The alternative is to use differential causality all the time, but this has serious disadvantages in case numerical solution (integration) techniques are applied (figures 8c and 8d).

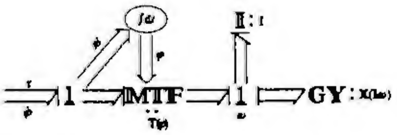


Figure 8a: 'Newtonian' Multibond Graph in integral causality

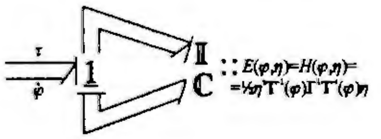


Figure 8b: 'Hamiltonian' Multibond Graph in integral causality

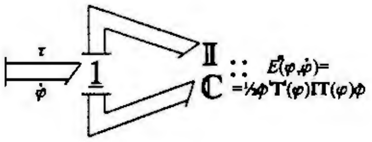


Figure 8c: 'Lagrangian' Multibond Graph in differential causality

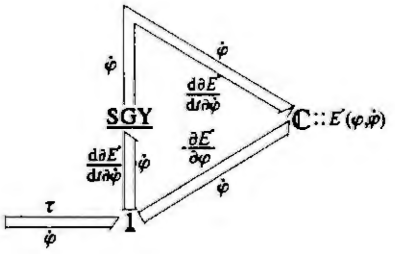


Figure 8d: 'Lagrangian' Generalized Bond Graph in differential causality

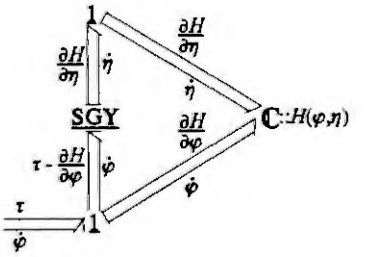


Figure 9: 'Hamiltonian' Generalized Bond Graph in preferred causality

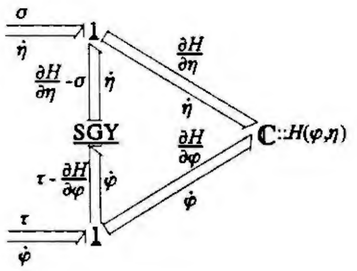


Figure 10: Brouwer's extension in Multibond Graph form

Finally, the Generalized Bond Graph notation of figure 8b as shown in figure 9 represents Hamilton's equations explicitly in the form of an array of symplectic gyrators (SGY array) in combination with the 1-junction array:

$$
\begin{aligned}
\dot{\varphi} &= +\frac{\partial H(\varphi,\eta)}{\partial \eta} \\
\dot{\eta} &= -\frac{\partial H(\varphi,\eta)}{\partial \varphi} + \tau
\end{aligned}
\tag{11}
$$

Both (11) and figure 9 display the asymmetry of the regular Hamiltonian representation. Figure 10 shows how Brouwer's extension resolves this asymmetry:

$$
\begin{aligned}
\dot{\varphi} &= +\frac{\partial H(\varphi,\eta)}{\partial \eta} - \sigma \\
\dot{\eta} &= -\frac{\partial H(\varphi,\eta)}{\partial \varphi} + \tau
\end{aligned}
\tag{12}
$$

where σ can be interpreted as an external velocity source analogue, or rather dualogue, to the external torque (generalized force) τ.

It deserves attention that the figures 8b through 10 represent *any* system characterized by a Lagrangian or a Hamiltonian. In many cases the matrix **T** can be replaced by the Jacobian **J** of the relation **L** between the coordinates **x** and the generalized coordinates φ:

$$\begin{aligned} \mathbf{x} &= \mathbf{L}(\varphi) \\ \dot{\mathbf{x}} &= \frac{\partial \mathbf{L}(\varphi)}{\partial \varphi}\dot{\varphi} = \mathbf{J}(\varphi)\dot{\varphi} \end{aligned} \tag{13}$$

Equation (13) shows that also in this more general case the matrix **J** can be singular, especially if the dimension of **x** is larger than the dimension of φ, which is often the case. In that case the kinetic ports must have differential causality. This explains the use of the Lagrangian in these situations.

4. CONCLUSION

In this short paper we showed that (multi-)bond graphs can be used successfully to display several representations of the dynamics of rigid body rotation: Newton-Euler, (Euler-)Lagrange, Hamilton and even Brouwer's extension of Hamilton's equations, which is almost suggested by the asymmetry of the Generalized Bond Graph representation. The bond graphs show that the equations can have integral causality except for the situation of gimbal-lock, where the inner gimbal is not influenced by torque on the outer gimbal. The only way to drive the system out of this situation is to apply a velocity source, i.e. to change the causality of the external port, which changes the causality of the I-ports and of the MTF representing the coordinate transformation from spatial to body-fixed coordinates.

5. REFERENCES

[1] Allen, R.R., *Multiport Representation of Inertia Properties of Kinematic Mechanisms*, J. Franklin Inst., Vol. 308, No. 3, pp. 235-255, 1979.

[2] Bos, A.M., *Modelling Multibody Systems in Terms of Multibond Graphs*, Ph.D. Thesis, University of Twente, Enschede, Netherlands, ISBN 90-9001442-X, 1986.

[3] Breedveld, P.C., *Comment on "Multiport representations of inertia properties of kinematic mechanisms"*, J. Franklin Inst., Vol. 309, pp. 491-492, 1980.

[4] Breedveld, P.C., *Thermodynamic Bond Graphs and the problem of thermal inertance*, J. Franklin Inst., Vol. 314, No. 1, pp. 15-40, July 1982.

[5] Breedveld, P.C., *Physical Systems Theory in terms of Bond Graphs*, ISBN 90-9000599-4, Enschede, 1984 (distributed by the author).

[6] Breedveld, P.C., *Decomposition of Multiport Elements in a Revised Multibond Graph Notation*, J. Franklin Inst., Vol. 318, No. 4, pp. 253-273, October 1984.

[7] Breedveld, P.C., *A definition of the multibond graph language*, in "Complex and Distributed Systems: Analysis, Simulation and Control", Tzafestas, S. and Borne, P., eds., Vol. 4 of "IMACS Transactions on Scientific Computing", pp. 69-72, North-Holland Publ. Comp., Amsterdam, 1986.

[8] Goldstein, H., *Classical Mechanics*, Addison-Wesley, New York, Second Edition, 1980.

[9] Karnopp, D.C., *Bond Graphs for Vehicle Dynamics*, Vehicle System Dynamics, Vol. 5, No. 3, pp. 171-184, 1976.

[10] Crandall, S.H., Karnopp, D.C., Kurtz, E.F., and Pridmore-Brown, D.C., *Dynamics of Mechanical and Electromechanical Systems*, Krieger Publishing, Malibar, Florida, 1968.

[11] Maschke, B.M., Schaft, A.J. van der, and Breedveld, P.C., *An intrinsic Hamiltonian formulation of network dynamics: non-standard Poisson structures and gyrators*, J. Franklin Inst., Vol. 329, No. 5, pp. 923-966, 1992

[12] M.J.L. Tiernego and A.M. Bos, *Modelling the dynamics and kinematics of mechanical systems with multibond graphs*, J. Franklin Inst., Vol. 319, No. 1/2, pp. 37-50, Jan./Feb. 1985, ISBN 0-08-03259-39.

TAXONOMY AND MODELLING OF VARIABLE IMPEDANCE ACTUATORS

ERNEST D. FASSE

MIT Department of Mechanical Engineering, Cambridge, MA 02139, USA

Abstract. A variable impedance actuator is a device which behaves like a modulated impedance. There is no theory of such devices. Elements useful in such a theory are presented. A number of concepts are introduced to distinguish different kinds of actuators, most importantly: *intrinsically variable impedance, near-passivity*, and *postural stability*. By means of example it is shown that the bond graph formalism is adequate to model both internal function and mechatronic function of variable impedance actuators.

1. INTRODUCTION

A variable impedance actuator is a device which behaves like a modulated impedance; for example animal muscle, which behaves like a modulated spring. There is no existing theory of such devices, that is: (1) There is no vocabulary that adequately distinguishes their essential properties and construction from other actuators. (2) There are no intuitive models that describe their internal functional, which is useful for detailed design and analysis. (3) There are no essential, functional models appropriate for incorporation in models of mechatronic systems such as robots, powered orthoses, and virtual environment simulators. A theory of variable impedance actuators would (1) discourage existing misconceptions about the functionality of existing actuators, namely animal muscle, and (2) encourage the development of new devices.

Elements useful in such a theory are presented in the sequel. Section 2 defines a number of concepts, most importantly: *intrinsically variable impedance, near-passivity*, and *postural stability*. Section 3 shows by means of example that the bond graph formalism is adequate to model both internal function and mechatronic function of variable impedance actuators.

2. VARIABLE IMPEDANCE, PASSIVITY, AND POSTURAL STABILITY

It is assumed that the reader is familiar with the multibond graph formalism [1]. The following set of definitions generate a basic taxonomy of actuators.

Variable impedance. An *actuator* is an element with one or more energetic ports, and one or more non-energetic signal inputs (Figure 1). An actuator is thus a modulated junction element. What distinguishes actuators from arbitrary, modulated junctions is that the modulating signals must be steering (control, command) inputs from some intelligent device. The signals must be associated with physical communication channels, and cannot be merely abstract flows of information. This broad definition encompasses such mechatronic components as variable transmissions, servoamplifiers, servomotor systems, hydraulic servovalves, and shock absorbers with variable sized orifices.

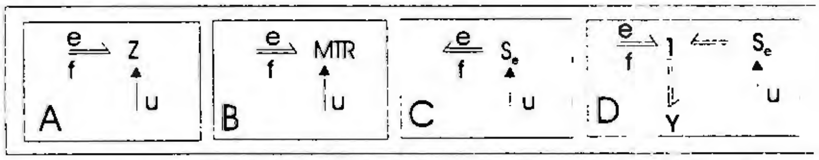


Figure 1. (A) An actuator is a mechatronic component corresponding to a modulated impedance. (B and C) Examples of simple actuators are (B) a variable transmission and (C) a modulated effort source, such as a torque servomotor. (D) Example of a Thevenin-degenerate variable impedance actuator.

Some actuators have simple functional descriptions. It is useful to distinguish these actuators from others with more complex dynamic behavior. *Simple actuators* are modulated junction structures (weighted junction structures, [1]), i.e., variable transmissions; and modulated flow and effort sources. The distinction is assumed to be based on the ideal functional behavior of the device, not the actual behavior. Thus a variable transmision with significant internal dynamics is still a simple actuator; the internal dynamics are considered to be parasitic. Similarly, a DC torque motor and servoamplifier is a simple actuator; the fact that the output torque is not independent of speed is an undesired, parasitic effect, and not essential.

Actuators that are not simple will be said to be *variable impedance actuators*. They have a functional, non-parasitic, non-trivial impedance. Animal muscle is an example of such an actuator. Most mechanical actuators are designed to be simple.

Let us temporarily restrict our attention to linear systems. The impedance of an energetic port is a linear dynamic relation between conjugate power variables of the port. A network with a given impedance can always be replaced by Norton or Thevenin equivalent networks of identical impedance: that is, by a flow source in parallel with an impedance, or an effort source in series with an impedance.

Two degenerate cases of a variable impedance actuator are (1) a modulated effort source in series with an unmodulated impedance (Figure 1), and (2) a modulated flow source in parallel with an unmodulated impedance. Such actuators will be referred to as *Thevenin-degenerate* and *Norton-degenerate actuators*. These terms are introduced primarily to facilitate the following definition of inherently variable impedance.

A variable impedance actuator will be said to be an *inherently variable impedance actuator* if it satisfies two criteria: (1) It is neither Thevenin- nor Norton-degenerate. (2) Neither its Norton nor Thevenin equivalent decompositions are easily identifiable with a physical decomposition. The first criterion is conceptual. The second is constructural; it requires that the energy source and the impedance be physically indistinguishable. An inherently variable impedance actuator is at some level of description monolithic.

Passivity. The impedance of an interaction port is passive if it is possible to define an available energy function [4]. Loosely, passive systems can store or dissipate energy, but cannot produce it. Passive systems are necessarily stable, and have robust interactive stability properties. A system that consists of two coupled passive subsystems is itself stable and passive. For this reason, passivity is a desirable property of the actuators of interactive machines such as robots, powered orthoses and virtual object simulators.

Another desirable property of actuators is that they be able to supply energy indefinitely, i.e., that they be active! It is useful then to define near-passivity, a less stringent notion of passivity. The set of conjugate power variables at an interaction port, *e* and *f*, are related by a modulated, possibly nonlinear, dynamic impedance operator *Z(u)*, where *u* is the steering input(s). Power flow, $\langle e,f\rangle$, is assumed positive into the port.

An actuator is *passive* if there exists an available energy function. The available energy function is a function of internal state, *x*. The following, acausal definition is a modification of Definition 3 of [4].

$$S_a(x) = \sup_{\substack{u(t) \\ e,f \text{ consistent with } Z(u(t))}} -\int_0^t \langle e,f\rangle dt \qquad (1)$$

In words, the available energy of an actuator with internal state, *x*, seen through the interaction port, is the supremum of the energy taken from the system over all steering inputs, *u(t)*, and effort and flows, *e(t)* and *f(t)*, consistent with the impedance of the port, *Z(u(t))*, starting from initial condition *x*. An actuator is *active* if it is not passive (i.e., if the supremum does not exist).

An actuator cannot both supply infinite power and be passive. An actuator is *nearly passive* if it is active, but passive for arbitrary, constant steering inputs, *u(t) = c*. An actuator can both supply infinite power and be nearly passive. An actuator is *strictly active* if it is neither passive nor nearly passive.

Postural stability. The final distinction is useful in classifying actuators of mechanical linkages (e.g., robotic manipulators). Such an actuator will have at least one mechanical port. A configuration (displacement, posture) can be associated with the integrated flow variable of this port. An actuator is *posturally stable* if for each constant, steering input u there is a stable configuration, *x*.

The resultant taxonomy is useful in classifying variable impedance actuators. For example, it is now possible to articulate the essential characteristics of muscle. Muscle can be thought of as a posturally stable, nearly passive, intrinsically variable impedance actuator. Conventional DC torque servomotors can be thought of as posturally instable, strictly active actuators.

3. A FUNCTIONAL MODEL OF A VARIABLE IMPEDANCE, ELECTROMECHANICAL ACTUATOR

This section shows that the bond graph formalism is flexible enough to model the function of a particular, reasonably complicated (posturally stable, nearly passive, intrinsically-) variable mechanical impedance, electromechanical actuator. Details of this actuator have not yet been published, although a relatively complete description of the design is given in Appendix A of [2]. This actuator behaves like a modulated, mechanical spring and damper. A model is presented which conceptually separates the compliant and resistive subsystems of the actuator. This detailed functional decomposition, which is unneccessary to describe the behavior of the device, gives more insight into the internal function. This detailed model is thus useful for design. This model is then simplified, resulting in a model adequate to describe the mechatronic function of the device.

The first model of the VZ actuator is the schematic diagram shown in Panel A of Figure 2, which shows the essential geometry of the actuator. It represents a cylindrically symmetric, electromechanical machine. The machine has a moving rotor inside a stationary stator, with an air gap between the two. Both the rotor and stator are wound with independent windings of sinusoidally varying conductor density, as is common on electric machines. The rotor has three such windings, the stator two. Each winding is represented by a pair of conductors at the points where the conductor density is highest. This representation is similar to that used in [3] and other texts on electric machines. Three windings are driven by current sources. The other two are short-circuited, either directly or via external resistances as shown. This actuator is connected to some mechanical system via a shaft. The schematic diagram is appealing because, although quite abstract, it has an obvious spatial interpretation. It does not represent the function of the device.

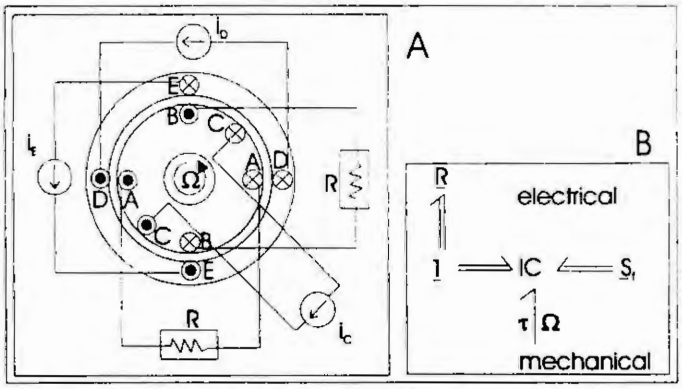


Figure 2. (A) Schematic diagram of VZ actuator. (B) Simple bond graph of VZ actuator.

Panel B shows what could be called the simplest bond graph corresponding to the device. The electromagnetic energy storage is modelled as a six-port IC field, which looks like an inertance to each of the five electrical ports and a capacitance to the mechanical port. Two ports are connected to a 2D array of resistances via a 2D array of one junctions. The remaining electrical ports are connected to a 3D array of flow sources. The mechanical port is not connected. The coenergy of the IC field is:

$$E^* = \tfrac{1}{2} i' L i = \tfrac{1}{2} i' \begin{bmatrix} L_{rr} & 0 & L_{ac} & L_{sr}\cos(\alpha+\beta) & L_{sr}\sin(\alpha+\beta) \\ 0 & L_{rr} & L_{bc} & -L_{sr}\sin(\alpha+\beta) & L_{sr}\cos(\alpha+\beta) \\ L_{ac} & L_{bc} & L_{cc} & L_{cs}\cos(\beta) & L_{cs}\cos(\beta) \\ L_{sr}\cos(\alpha+\beta) & -L_{sr}\sin(\alpha+\beta) & L_{cs}\cos(\beta) & L_{ss} & 0 \\ L_{sr}\sin(\alpha+\beta) & L_{sr}\cos(\alpha+\beta) & L_{cs}\sin(\beta) & 0 & L_{ss} \end{bmatrix} i. \qquad (2)$$

The mutual inductance between pairs of windings varies cosinusoidally withβ, the angle between the rotor and the stator. The angle between windings A and C is$\alpha = \pi/4$. The inductance parameters L_{jk} are constant, and current $i' = \{i_a \quad i_b \quad i_c \quad i_d \quad i_e\}$. Although elegant, this model also does not give insight into the function of the device, due to the complex IC field. Function could be explained by a detailed analysis of the dynamic equations, as in [2], but it is possible to describe the function by elaborating the bond graph. It is useful to break the IC field up into an equivalent network of four I fields and two IC fields, as in Panel A of Figure 3.

The windings can be grouped into three functional groups, {A,B}, {C}, and {D,E}. This functional division is reflected by the three arrays of one junctions. I fields I_{AB}, I_C, and I_{DE} are the inductances associated with each winding group, and are nondegenerate (full-rank). I field $I_{AB,C}$, and IC fields $IC_{AB,DE}$ and $IC_{C,DE}$ are mutual inductances between functional groups. These fields are degenerate, with self inductances (diagonal elements) of zero, and must have differential causality on the inertial ports.

Fields I_C and I_{DE} also have differential causality because they are in series with flow sources. Because the flows on the inertial ports of field are fixed by the sources, the field looks like a modulated capacitance from the mechanical port. This is accounted for in the simplified functional graph of Panel B.

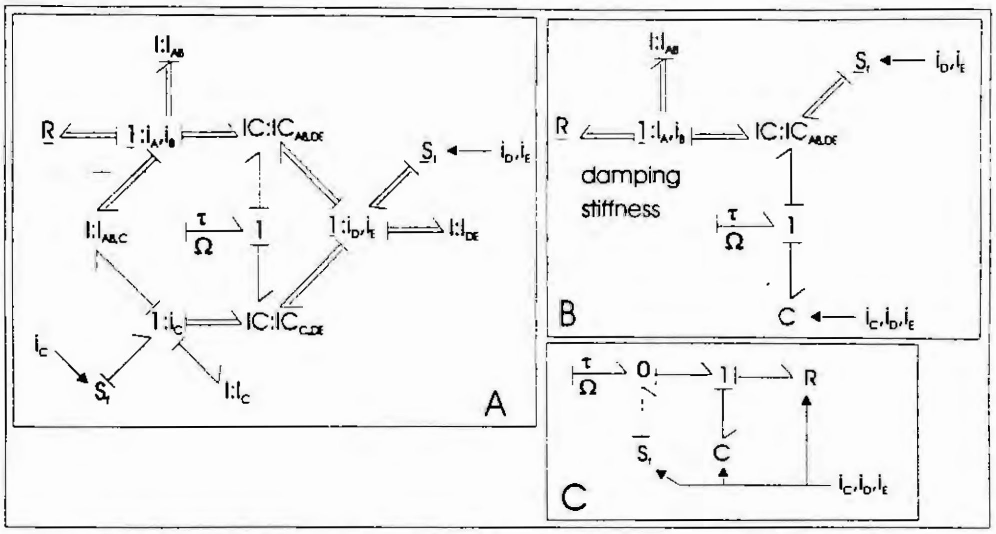


Figure 3. (A) Internal functional graph. (B) Simplified graph. (C) Mechatronic functional graph.

Field $I_{AB,C}$ (coupling between different functional groups on the rotor) is functionally undesirable, although it cannot be eliminated from this particular design. Its effects are negligible if i_C changes quasi-statically. For this reason it is not present in the simplified graph. What is left is a division of the graph into two functional parts. The first functional part behaves like a modulated capacitance, as measured. The second part behaves like a modulated damper. This cannot obviously be shown with additional bond graph manipulations, but the two things have been gained: (1) The subgraph is much easier to analyse than the original graph. (2) The subgraph is equivalent to that of a two-phase induction motor [3] being excited by servoamplifiers, instead of sinusoidally-varying voltage sources. Quasi-static analysis of this graph [2,3] shows that the subgraph bdhaves like a modulated resistance. Equation 3 gives the resultant impedance seen by the mechanical port.

$$\tau = i_C\sqrt{i_D^2 + i_E^2}\,L_{CS}\sin\left(\beta - \arctan\left(\frac{i_D}{i_E}\right)\right) + \frac{L_{SR}^2}{R}\left(i_D^2 + i_E^2\right)\frac{\Omega}{1 + \left(\frac{L_{RR}}{R}\Omega\right)^2} \tag{3}$$

The actuator behaves like a modulated spring and damper, the stiffness, resistance and equilibrium postion of which are modulated by currents i_C, i_D, and i_E. The mechatronic functionality of the actuator is described by the graph shown in Panel C of Figure 3.

4. RESULTS

The preceding discussion gave only a superficial description of the actuator; its primary purpose was to illustrate the use of bond graphs to model both internal and mechatronic functionality of actuators. Although it was obvious that dynamic behavior could be modelled by a bond graph (Panel B of Figure 2), it was not obvious that function could be modelled by a bond graph (Panel A of Figure 3). The bond graph formalism, tqgether with the actuator taxonomy presented, are useful elements of a theory of variable impedance actuators.

5. ACKNOWLEDGEMENTS

E. Fasse is currently a US NSF-NATO postdoctoral fellow with the Department of Electrical Engineering, Control, Systems and Computer Engineering Group, Control Laboratory, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands. Work on the VZ actuator was performed with support of a Fairchild Fellowship together with Neville Hogan in the Newman Biomechanics Laboratory of MIT.

6. REFERENCES

[1] Breedveld, P.C., Multibond-graph elements in physical systems theory in terms of bond graphs, J. of Franklin Inst., 319, No. 1/2 (1985), 1-36.

[2] Fasse, E.D., On the Use and Representation of Sensory Information of the Arm by Robots and Humans, Ph.D. thesis, MIT Dept. of Mechanical Eng., Cambridge, MA, (1993), 287-299.

[3] Slemon, G.R. & Straughen, A., Electric Machines, Addison Wesley, Reading, MA, 1980.

[4] Willems, J.C., Dissipative Dynamical Systems, Part 1: General Theory, Arch. Rational Mech. Anal., (1972), 321-351.

An interpretation of the Eulerian Junction Structure in 3D Rigid Bodies

S.Stramigioli and P.C.Breedveld
University of Twente, Electrical Engineering Department
Control Lab. and Mechatronics Research Centre Twente
P.O. Box 217, 7500 AE Enschede, The Netherlands
e-mail: smi@rt.el.utwente.nl

Abstract

This paper emphasises that the gyroscopic effects represented by the Eulerian Junction Structure in the rotational domain of a rigid body, are due to centrifugal phenomena, resulting from the non-inertiality of the coordinate frame in which the angular velocity is expressed. It is in fact common in literature to say that the EJS incorporates Coriolis effects. It will be shown that these are not present at all in the rotational domain, but are indeed present in the translational one.

1 Introduction

In the context of modeling of spatial mechanisms by means of bond graphs, the power continuous gyrator called Eulerian Junction Structure plays an important role [6]. This element is a very expressive representation of the gyroscopic effects that can appear in a rotating rigid body. Unfortunately, it is often stated that this element is representing also Coriolis forces in the rotational domain but, as it will be shown, this idea is not correct. In fact, the gyroscopic effects that the EJS represents in the rotational domain are only due to centrifugal phenomena.

2 Basics

2.1 Centrifugal and Coriolis apparent forces

The second law of dynamics states that any acceleration of a point mass observed from an inertial frame, is justifiable by means of a force applied to it. If we consider two reference frames e and c in which e is inertial and c is rotating respect to e with an angular velocity ω, the acceleration that an observer in c would see of a particle will be correlated to the one that an inertial observer will see by the kinematic relation:

$$A = a - \dot{\omega} \times r - \omega \times (\omega \times r) - 2\omega \times V \tag{1}$$

where A is the acceleration of the particle that the non-inertial observer will see, a is the acceleration that the inertial observer will see, ω is the angular velocity of c with respect to e, r is the position of the particle with respect to c and V is the velocity that the non-inertial observer will see . If the particle under consideration has a mass m the forces which will be attributed to these accelerations will be for the inertial observer e just ma which will be a real force. For the non-inertial observer instead, there will be three supplementary forces, two of which will respectively be $m(\omega \times (\omega \times r))$ and $m(2\omega \times v)$. Both of them are apparent forces and they are called respectively *centrifugal* and *Coriolis* forces.

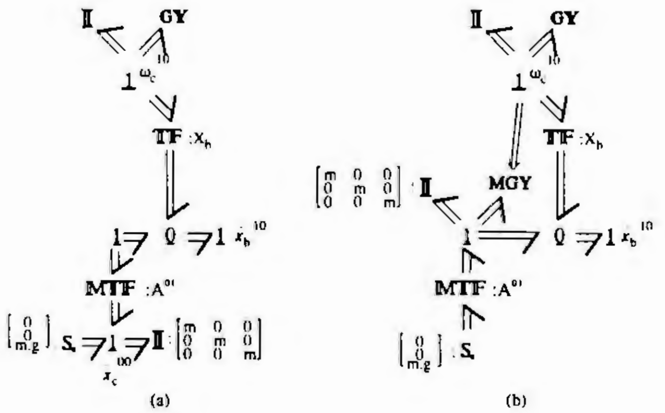


Figure 1: A Rigid body

2.2 Rigid body rotations.

Let us consider the third law of Newton in implicit tensor notation [4]:

$$M_j = \frac{d}{dt}(I_{ji}\omega^i) \tag{2}$$

where ω^i is the angular velocity of the body respect to an inertial reference, I_{ji} is the inertia tensor and M_j is an applied torque. If we express (2) in a base which is static respect to the inertial reference, the derivative with respect to time of the base element is zero and therefore due to the linearity of the operator d/dt , the numerical representation gets the same form:

$$^e\mathbf{M} = \frac{d}{dt}(^e\mathbf{I}^e\omega) \tag{3}$$

The latter form[1] is similar to the tensor form of (2) and corresponds to an I-element in bond graph terminology. The form of the same equation in a base c that is not static is different and can be calculated in the following way.

Let us indicate with $\mathbf{A}$ the jacobian matrix from coordinates c to coordinates e and suppose for simplicity that we are working with orthonormal bases. In this case $\mathbf{A}$ is orthonormal and therefore $\mathbf{A}^{-1} = \mathbf{A}^T$. If c is moving with respect to e the matrix $\mathbf{A}$ will be clearly time-dependent. Using the change of coordinates for the tensor under consideration, and the hypothesis of orthonormality of $\mathbf{A}$ we obtain:

- $^e\omega = \mathbf{A}^c\omega$
- $^e\mathbf{I} = \mathbf{A}^{-T c}\mathbf{I}\mathbf{A}^{-1} = \mathbf{A}^c\mathbf{I}\mathbf{A}^{-1}$
- $^e\mathbf{M} = \mathbf{A}^{-T c}\mathbf{M} = \mathbf{A}^c\mathbf{M}$

Substituting the last identities in (3), we obtain

$$\mathbf{A}^c\mathbf{M} = \frac{d}{dt}(\mathbf{A}^c\mathbf{I}^c\omega)$$

Now $\mathbf{A}$ is time dependent and therefore we obtain:

[1] The superfix e indicates that it is the numerical rappresentation of the tensor in the base e.

$$\mathbf{A}\,{}^c\mathbf{M} = \dot{\mathbf{A}}\,{}^c\mathbf{I}^c\omega + \mathbf{A}\frac{d}{dt}({}^c\mathbf{I}^c\omega)$$

The matrix $\mathbf{A}$ being orthonormal yields that $\dot{\mathbf{A}} = {}^e\Omega_c^e\mathbf{A}$, where ${}^e\Omega_c^e$ is a skew-symmetric matrix[2], corresponding to the angular velocity ${}^e\omega_c^e$ of the frame c respect to e in coordinate of e.

The general form of the numerical expression of (2) can be obtained using the identity $\mathbf{M}^T\Omega\mathbf{M}\mathbf{p} = (\mathbf{M}^T\omega) \times \mathbf{p}$ and is therefore of the form:

$$ {}^c\mathbf{M} = {}^c\,\omega_c^e \times ({}^c\mathbf{I}^c\omega) + \frac{d}{dt}({}^c\mathbf{I}^c\omega) \tag{4}$$

where ${}^c\omega_c^e$ is the vector corresponding to the anti symmetric matrix ${}^c\Omega_c^e$ which is used to have the vector product.

If c is rigidly connected to the rigid body, ${}^c\omega_c^e = {}^c\omega$, and the usual form of the Euler equation in body coordinates is obtained.

Let us now pre multiply (4) for ${}^c\omega^T$. It can be seen that if ${}^c\omega = {}^c\,\omega_c^e$ the first term of the right side of (4) vanishes but in general if they are different, this is not the case. This means that in the representation of (4) in rigid body coordinates, there is no flow of power to the EJS, in other words it is power continuous, in correspondence with the nature of a gyristor.

3 The Bond Graph of a Rigid Body.

In the context of modeling of 3D mechanism, one of the most used representations of a rigid body is the one reported in Fig.1a, [6]. The gyrator in the top of the figure represents the gyroscopic effects in the rotational domain corresponding to the first term on the right side of (4) as also shown in [2, 3]. By bringing the translation inertia through the lower MTF, another gyrator appears (Fig.1b). These two gyrators, are called Eulerian Junction Structures and are representations of non-inertial effects in the rotational and translational domain [2, 3].

Since the m_{ij} tensor representing the mass is a scalar matrix of the form $\delta_{ij}m$ and it is isotropic[3] [4], its numerical representation is the same also after the transformation through the MTF.

3.1 The EJS in translation domain.

The EJS that appears by transforming the inertia through the lower MTF, is a representation of non-inertial effects. If the body is also translating with respect to the inertial reference used, the EJS will indeed represent all the non-inertial forces introduced in (1).

3.2 The EJS in rotational domain and the centrifugal effects.

A rigid body is nothing else that an infinite set of infinitesimal masses rigidly connected in which the mutual distance does not change with time. Therefore, a rigid body could be in principle studied by studying each of the single point masses which constitute it and then integrating over the whole volume. To facilitate this process, the concept of the inertia-tensor has been introduced. Nevertheless, concepts like Coriolis and centrifugal acceleration are defined for points of masses and to see how they are correlated to rigid bodies is important. To do so, we can analyze the rotational dynamics starting by considering the definition of angular momentum for a point p of mass m.

The angular momentum for a point p of mass m at a position r with velocity v is defined as follows:

[2] This can be also seen in a coordinate independent system by considering $\mathbf{A}$ an element of the Lie group $SO(3)$ and Ω as an element belonging to the Lie algebra tangent at the identity to the Lie group that is the right translation of $\dot{\mathbf{A}}$. See [1] or [5] for details.

[3] It can be shown that it is the only isotropic second order tensor.

$$k_k = (r\times)^j_k m_{ji} v^i$$

which can be expressed in an inertial coordinate frame e as:

$${}^e\mathbf{k} = m^e\mathbf{r} \times {}^e\mathbf{v}$$

By indicating with $\mathbf{A}$ the matrix of the change of base from c to e and differentiating we obtain:

$${}^e\dot{\mathbf{k}} = m(\dot{\mathbf{A}}\,{}^c\mathbf{r}) \times {}^e\mathbf{v} + m(\mathbf{A}\,{}^c\mathbf{r}) \times {}^e\mathbf{a}$$

Expressing $\dot{\mathbf{A}} = {}^e\mathbf{\Omega}^e_c\,\mathbf{A}$, using the identity (1) expressed in inertial coordinates e and the fact that the velocity of the point of mass in the system c is zero, after some tedious calculations we obtain:

$${}^c\dot{\mathbf{k}} = m({}^c\mathbf{r} \times ({}^c\dot{\omega} \times {}^c\mathbf{r})) + m({}^c\mathbf{r} \times ({}^c\omega \times ({}^c\omega \times {}^c\mathbf{r}))) \tag{5}$$

The term corresponding to the Coriolis term has clearly disappeared because the point of mass under consideration is not moving in its own frame, but the centrifugal term is still present.

By considering now a rigid body and by using (5) with instead of m, dm, and with instead of $\mathbf{k}$, $d\mathbf{k}$ we obtain:

$${}^c\dot{\mathbf{k}} = \int_V \rho({}^c\mathbf{r} \times ({}^c\dot{\omega}\times{}^c\mathbf{r}))dv + \int_V \rho^c\mathbf{r} \times ({}^c\omega\times({}^c\omega\times{}^c\mathbf{r}))dv \tag{6}$$

Since we have that $\mathbf{r} \times (\omega \times (\omega \times \mathbf{r})) = \omega \times (\mathbf{r} \times (\omega \times \mathbf{r}))$ and by the definition of the inertia tensor, it is possible to see that the first term on the right side corresponds to ${}^c\mathbf{I}^c\dot{\omega}$ and the second one to ${}^c\omega \times ({}^c\mathbf{I}^c\omega)$ which proves that the EJS represented by the last term is a consequence of centrifugal effects and not Coriolis ones.

4 Conclusions

By means of some simple calculations, it has been shown that the expression representing Euler equations in body-fixed coordinates gives rise to two terms represented in bond graphs by means of the so called Eulerian Junction Structure and an I-element. It has been shown that differently from what is normally stated in literature, the Eulerian Junction Structure of the rotational domain represents just centrifugal effects since Coriolis ones are not present.

References

[1] Arnold, V.I. *Mathematical Methods of Classical Mechanics*, Second Edition, Springer-Verlag 1989.

[2] Karnopp, D. C. *Bond Graphs for Vehicle Dynamics*, Vehicle Systems Dynamics (1976), Vol 5, pp. 171-184, 1976.

[3] Karnopp, D. C. *The Energetic Structure of Multibody Dynamic Systems*, Journal of the Franklin institute, Vol 306, pp. 165-181, August 1978.

[4] Dubrovin, B.A. *Modern Geometry - Methods and Applications, Part 1*,pp.157,Springer-Verlag 1984.

[5] Olver, P.J. *Applications of Lie Groups to Differential Equations*, Springer-Verlag1986.

[6] Bos, a.M. *Modeling Multibody Systems in Terms of Multibond Graphs*, Ph.D. dissertation, University of Twente, the Netherlands, 1986.

ENERGETICALLY PROPER MODELING OF A SIMPLE THROTTLING PROCESS

Peter BREEDVELD[†] and Neville HOGAN[††]

[†]University of Twente, Electrical Engineering Department, Control Laboratory,
P.O. Box 217, 7500 AE Enschede, NL, tel.: + 31 53 89 27 92, fax: + 31 53 34 00 45, email: brv@rt.el.utwente.nl
[††]Massachusetts Institute of Technology, Department of Mechanical Engineering, 77 Massachusetts Avenue,
Cambridge, Mass. 02139, USA, tel.: + 1 (617) 253 2277, fax: + 1 (617) 258 5802, email: neville@mit.edu

Abstract. Many attempts to construct energetically correct network models of the dynamics of compressible fluid systems have been reported, many of them in bond graph — usually pseudo-bond graph — terms [10,11,14,13,12]. In this paper we show that the use of the specific enthalpy and the (molar) mass flow as power-conjugate variables for energetically correct models of mass flow processes does not lead to a proper result. The simplest way to demonstrate this is by the example of a simple throttling process between two gas-filled tanks with different pressures and temperatures. The result is that the convected entropy flow has to be modeled *explicitly* and that the irreversible effect of the throttling has to be described by as well a constitutive relation between the pressure drop and the volume flow as the entropy production in this process.

1. INTRODUCTION

Suppose we want to model the transient behavior of pressure and temperature in two gas-filled tanks coupled by tubes via a valve (figure 1), when this valve is opened. The process is assumed to be adiabatic, i.e. no energy is exchanged with the environment. This assumption holds due to either proper thermal isolation or a sufficiently fast process. The choice of the flow variable for the gas flow usually causes no difficulty: the mass flow $\frac{\mathrm{d}m}{\mathrm{d}t} = \dot{m}$ or the molar flow $\frac{\mathrm{d}N}{\mathrm{d}t} = \dot{N}$ are used, which are related by

$$\dot{m} = \dot{N}\sum_{i=1}^{n} c_i M_i \tag{1}$$

with $c_i = \frac{N_i}{N}$ the molar fraction or concentration, $N = \sum_{i=1}^{n} N_i$ the total number of moles, N_i the molar number per species i, n the number of species and M_i the molar mass of species i. We suppose in the sequel that n=i=1, which means that m=MN. This means that the molar flow and the mass flow differ only up to a proportionality constant M, such that the more commonly used mass flow can be chosen to describe this process without any conceptual difficulty. The total power due to the material flow is equal to the enthalpy flow. The classical way to interpret this is to say that the change of energy dE in a tank consists not only of the change of internal energy dU but also of the work dW done by the fluid, i.e.

$$\mathrm{d}E = \mathrm{d}U + \mathrm{d}W = \mathrm{d}U + p\mathrm{d}V = \mathrm{d}H \tag{2}$$

where p is the pressure, dV the 'cubic displacement' of the fluid (note that the total volume of the tanks, the valves and the fluid lines is constant!) and dH the change of enthalpy. Another way to look at this is to observe that the volume V has been used as a boundary criterion. This means that the volume is not a property that is convected by a flow of matter like the other stored properties. This in turn means that the change of energy due to exchange of matter has to be reduced with a term $\frac{\partial U}{\partial V}\mathrm{d}V$:

$$\mathrm{d}E = \mathrm{d}U - \frac{\partial U}{\partial V}\mathrm{d}V = \mathrm{d}U - (-p)\mathrm{d}V = \mathrm{d}U + p\mathrm{d}V = \mathrm{d}H \tag{3}$$

The latter interpretation results in a more systematic (bond graph) representation [2,3,8], but this seems to have stayed unobserved in the last decade [9,7].

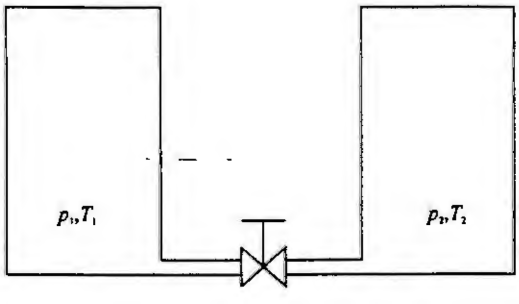


Figure 1: Two gas-filled tanks coupled by a valve

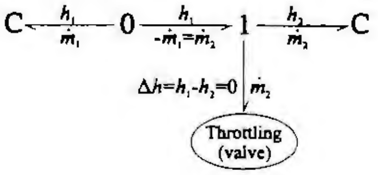


Figure 2: Bond graph with improper powerconjugate variables

2. PROPER POWERCONJUGATE VARIABLES

The first interpretation suggests that the (mass-)specific enthalpy $h=\dfrac{H}{m}$ is the proper conjugate power variable (effort) to describe this process:

$$\mathrm{d}E=\mathrm{d}H=h\mathrm{d}m \tag{4a}$$

or

$$P=\frac{\mathrm{d}E}{\mathrm{d}t}=\frac{\mathrm{d}H}{\mathrm{d}t}=h\frac{\mathrm{d}m}{\mathrm{d}t}=h\dot{m} \tag{5b}$$

This suggests that we need to describe the throttling process in the valve by a specific enthalpy difference Δh and a mass flow $\dot{m}$ (figure 2). However, Δh is typically zero during throttling, such that there is no 'driving force' for the mass flow $\dot{m}$. A first indication of how this paradox can be resolved is that the power $(\Delta h)\dot{m}$ is not equal to the entropy production rate times the temperature, i.e. the produced thermal energy ('heat') in such a resistive process. This suggest that, in order to model the irreversible process of throttling in a proper way, the whole entropy bookkeeping has to be made explicit. For reasons of simplicity we will assume that no other extensive properties than the entropy S, the amount of matter m and the (in this case constant) volume V characterize the state of the gas. In other words the internal energy U is a function of these three extensities: $U=U(S,V,m)$. In bond graph terms this means that the tanks are both described by a 3-port C element of which the constitutive equations have the form (due to the preferred integral causality):

$$\begin{aligned}&p=p(V,S,m)\\&T=T(V,S,m)\\&\mu=\mu(V,S,m)\ (=\mu(p,T))\end{aligned} \tag{5}$$

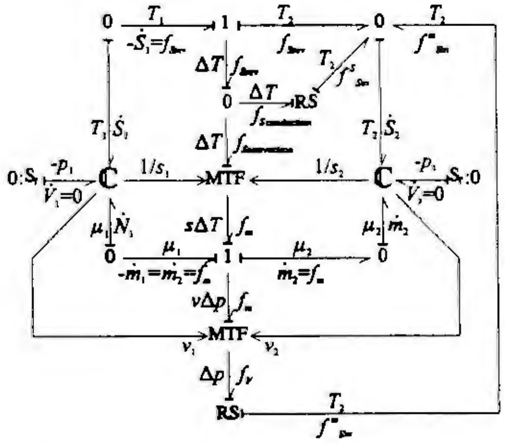


Figure 3: Bond graph with proper variables and explicit entropy bookkeeping

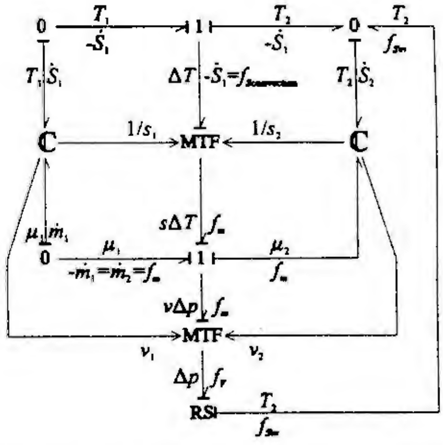


Figure 4: Simplified bond graph without conduction

Both for a monatomic ideal gas and for a monatomic van-der-Waals gas the constitutive equations of the gas ('state equations') can be analytically derived [6]. In other cases these relations may have to be found experimentally. Note that these three equations are not independent: The number of independent intensive states (p, T) is one less than the number of independent extensive states (V, S and m) due to the fact that the internal energy is a first-order homogeneous function of V, S and m [3].

Since the volume V serves as boundary criterion and cannot be convected, the only property to be convected by $\dot{m}$ is the entropy S. This means that the bond graph in figure 3 represents the throttling process much better than the one in figure 2: the entropy flow between the tanks is modeled explicitly. Apart from the convected entropy also the entropy exchanged by conduction is shown, including the entropy generated in this process. The effort relation of the 1-junction which represents the mass flow $\dot{m}$ is

$$\Delta\mu = -s\Delta T + v\Delta p \tag{6}$$

where $\Delta\mu$ is the difference in (total) material potential between the tanks with $\frac{\partial U}{\partial m} = \mu$, ΔT the temperature difference with $\frac{\partial U}{\partial S} = T$, Δp the pressure difference with $\frac{\partial U}{\partial V} = -p$, $s = \frac{S}{m}$ the (mass-) specific entropy and $v = \frac{V}{m} = \frac{1}{\rho}$ the (mass-) specific volume, where ρ is the mass density. We recognize (6) as (a discrete version of) the Gibbs-Duhem relation, which also expresses the dependence of μ, p and T and thus is also a result of the first-order homogeneity of the (internal) energy. The state-modulated transformer (MTF) representing the entropy convection is modulated by s^{-1}. Although the volume is not convected, it remains possible to convert the bond with variables $v\Delta p$ and $\dot{m}$ into a bond with Δp and f_V using an MTF modulated by v. The flow f_V represents the material flow as a volume rate. Note that it is not a change of change of volume $\dot{V}$. Therefore, it is not connected to the 'volume-ports' of the three-port C-elements. The relation between Δp and f_V may be linear ($\Delta p = Rf_V$) or, as will be often the case, nonlinear ($\Delta p = \Delta p(f_V)$). It describes the resistive behavior of the valve (in the linear case is R the resistance). Since entropy is modeled explicitly, the valve has to be described as an irreversible transducer RS. The entropy production rate $f_{S_{\mathrm{irr}}}$ in this process is given by

$$f_{S_{\mathrm{irr}}} = \frac{(\Delta p)\cdot f_V}{T} \tag{7}$$

In case of a linear resistive relation $f_V = \frac{\Delta p}{R}$ (form corresponding with the causality shown in figure 3)

$$f_{S_{\mathrm{irr}}} = \frac{(\Delta p)^2}{RT} \tag{8}$$

or in case of a quadratic flow relation $f_V = sgn(\Delta p)\frac{\sqrt{|\Delta p|}}{R}$:

$$f_{S_{\mathrm{irr}}} = \frac{|\Delta p|\sqrt{|\Delta p|}}{RT} \tag{9}$$

As the volume of the tanks is constant, the rate of change of the volume is zero. This explains the two zero-flow sources (Sf) connected to the 3-port C element representing the tank. Although this means that the power is zero and the bond can be omitted in principle (figure 4), it is necessary to leave this port in the model as long as one is interested in the pressure in each of the tanks. Otherwise only the pressure difference over the valve would be known. Another phenomenon represented in figure 3 that has been omitted in figure 4 is heat conduction. As shown in figure 3, the reversible part of the entropy exchange between the tanks may consist of a convective part, represented by the MTF and a conductive part, represented by the resistive port of an irreversible transducer. The other port of this transducer represents the irreversibly produced entropy in the process of heat conduction. In the model of figure 4 it has been assumed that the heat conduction can be neglected in order to be able to focuss on the modeling of the convection process. The causality of both models emphasizes that the irreversible transducer which represents the throttling process (in the valve) is crucial for obtaining a dynamic model.

What causality also shows, in combination with the modulating signals, are the following conditions to prevent positive feedback in the model via the modulations is the following:

Given that

1) the intrinsic stability conditions of the gas require that $\frac{\partial T}{\partial S} > 0$, $\frac{\partial \mu}{\partial m} > 0$ and $\frac{\partial p}{\partial V} < 0$ (the latter condition plays no role, since the corresponding port is not connected to the junction structure)

2) due to the nature of S, m and V: $s>0$, $v>0$ (in case of numerical integration it is even desirable to exclude the pathological case that the tanks are almost empty in order to prevent small time constants: $s>s_{min}>0$, $v>v_{min}>0$)

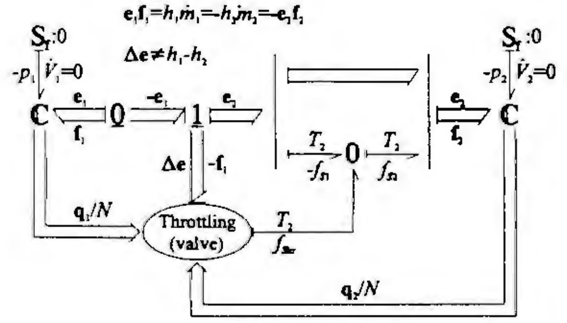


Figure 5: Multibond graph of the throttling process

it is required that:

$s = s_1$ if $f_V > 0$; $s = s_2$ if $f_V < 0$ and $v = v_1$ if $f_V > 0$; $v = v_2$ if $f_V < 0$

In other words: 'the origin of the modulation has to be the origin of the flow of matter', which, in case of convection of properties, makes good sense.

3. CONCLUSION

Finally, figure 5 shows the main argument of this paper, viz. the fact that the specific enthalpy h cannot be used as an effort variable to 'drive' the mass flow during throttling is shown in multibond graph terms [4,5]. This representation elucidates even more that 1) the whole process is powercontinuous, 2) explicit representation of the entropy production is crucial to be able to model the resistive process and 3) any extensive variable (property) can be convected by the matter flow in the same manner as the entropy, even momentum [2,3,1].

4. REFERENCES

[1] Beaman, J.J., Breedveld, P.C., *Physical Modeling with Eulerian frames and Bond Graphs*, Trans. ASME, J. of Dyn. Syst., Meas. & Control, Vol. 110, No. 2, pp. 182-188, June 1988.

[2] Breedveld, P.C., *Thermodynamic Bond Graphs: A new Synthesis*, Int. J. Modelling Simulation, Vol. 1, pp. 57-61, 1981.

[3] Breedveld, P.C., *Physical Systems Theory in terms of Bond Graphs*, ISBN 90-9000599-4, Enschede, 1984 (distributed by the author).

[4] Breedveld, P.C., *Multibond Graph Elements in Physical Systems Theory*, J. Franklin Inst., Vol. 319, No. 1/2, pp. 1-36, 1985.

[5] Breedveld, P.C., *A definition of the multibond graph language*, in "Complex and Distributed Systems: Analysis, Simulation and Control", Tzafestas, S. and Borne, P., eds., Vol. 4 of "IMACS Transactions on Scientific Computing", pp. 69-72, North-Holland Publ. Comp., Amsterdam, 1986.

[6] Breedveld, P.C., *An alternative formulation of the state equations of a gas*, Entropie, Vol. 164/165, pp. 135-138, 1991, ISSN 0013 9084.

[7] Brown, F.T., *Convection Bonds and Bond Graphs*, J. Franklin Inst., Vol. 328, No.:5/6, pp. 871-886, 1991.

[8] Diller, K.R., Beaman, J.J., Montoya, J.P., Breedveld, P.C., *Network Thermodynamic Modeling With Bond Graphs for Membrane Transport During Cell Freezing Procedures*, Trans. ASME, J. of Heat Transfer, Vol. 110, pp. 938-945, November 1988.

[9] Engja, H., *Pseudo Bond Graph Representation of Unsteady State Heat Condition*, Proc. 11th IMACS Cong. Systems Simulation and Scientific Computation, Oslo, Vol. 4., pp. 301-304, 1985.

[10] Karnopp, D.C., *A Bond Graph Modeling Philosophy for Thermofluid Systems*, ASME Trans., J. Dynamic Syst. Measure. Control, Vol. 100, No. 1, pp. 70-75, 1978.

[11] Karnopp, D.C., *State Variables and Pseudo Bond Graphs for Compressible Thermofluid Systems*, Trans. ASME, J. Dynamic Syst. Measure. Control, Vol. 101, No. 3, pp. 201-204, 1979.

[12] Margolis, D.L., *Modeling of Two-Stroke Internal Combustion Engine Dynamics Using Bond Graph Technique*, ASAE Trans., pp. 2263-2275, 1976.

[13] Rietman, J., *New System Variables for the Flow of Thermal Energy Based on the Concept of Energy*, in "Physical Structure in System Theory", edited by J.J. van Dixhoorn and F.J. Evans, pp. 35-37, Academic Press, New York, 1974.

[14] Thoma, J.U., *Entropy and Mass Flow for Energy Conversion*, J. Franklin Inst., Vol. 299, pp. 89-96, 1970.

BOND GRAPH MODELLING AND INVARIANT SUBSPACES

C. SUEUR and G. DAUPHIN-TANGUY

LAIL URA CNRS D1440
Ecole Centrale de Lille
Cité scientifique
BP 48
59651 Villeneuve d'Ascq Cedex France
Tel: (33) 20 33 54 02 Fax: (33) 20 33 54 18

Abstract. Explicit interrelations are established between the geometric approach for linear system control synthesis and the structural study of bond graph models. This allows us to have a better understanding of bond graph models directly from a graphical approach in view of control synthesis.

1. INTRODUCTION

Control synthesis for multivariable linear, time invariant systems is an active area of current research. The so called "geometric approach" is proposed by Wonham [11,12] within the state space framework and the linear algebra. The two fundamental concepts are the (A,B) invariant subspaces and the controllability subspaces included in one specific subspace. The geometric approach has given complete answers to important synthesis questions, such as the so called Morgan's problem (Descusse et al [4]).

A jointly use of geometric and frequency domain concepts has led to the resolution of particular structural control synthesis. These synthesis are based on the structural properties of the matrix triplet (C,A,B) which remains invariant under various transformation groups. Some invariant lists of integers, such as the infinite zero orders, (Morse[5]), or the essential orders, (Commault et al [3]), have been found.

In this context, the study of the structural Controllability/Observability properties, (Sueur et al [8]), and the Controllability/Observability subspaces for pole assignment, (Sueur et al [9]), have been proposed from a bond graph approach. Another work concerns the study of poles and zeros, (Sueur et al [10]).

The object of this paper is to characterize some structural properties of multivariable systems modeled by bond graphs. These properties are proposed for the state space equations derived from the bond graph model and are directly pointed out from causal manipulations on the bond graph model. We characterize some invariant subspaces for bond graph models and we give procedures to formally calculate them. We show up the simplicity of the procedures and the accuracy of the results.

2. BASIC CONCEPTS

2.1 (A,B) invariant subspaces.

Let us consider a dynamical linear time-invariant system (A,B,C) described by equation (1) where $A \in \Re^{nxn} = \chi$, $B \in \Re^{nxm}$, $C \in \Re^{pxn}$, and let us denote $\mathcal{K}$ the kernel of C and $\mathcal{B}$ = ImB.

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx \end{cases} \tag{1}$$

Definition 1.

A subspace $V \subset \Re^n$ is (A,B) invariant if there exists a map $F \subset \Re^{mxn}$ such that $(A + BF)V \subset V$.

V has the property that if the initial state $x(0) \in V$, then there exists a control $u(t)$, $t \geq 0$, such that $x(t) \in V$, for all $t \geq 0$. It follows the property $AV \subset V + \mathcal{B}$.

Now let $\mathcal{I}$(A,B; $\mathcal{K}$) denotes the subclass of (A,B) invariant subspaces contained in $\mathcal{K}$. $\mathcal{I}$(A,B; $\mathcal{K}$) has a unique largest or supremal element V^*. A well-known algorithm, the limit of which is V^*, named the Invariant

Subspace Algorithm, proposed in the literature [11,12} is recalled in (2). A procedure for the numerical computation of V^* by matrix manipulation is proposed in [12].

$$\begin{cases} V_0 = \chi \\ V_\mu = K \cap A^{-1}(B + V_{\mu-1}) \end{cases} \qquad (2) \qquad\qquad \begin{cases} R_0 = 0 \\ R_\mu = V^* \cap (B + AR_{\mu-1}) \end{cases} \qquad (3)$$

Definition 2.

A subspace $\mathcal{R} \subset \Re^n$ is a controllable subspace of the pair (A,B) if there exists a map $F \subset \Re^{m \times n}$ such that $\mathcal{R} = \prec A + BF|\mathrm{Im}B \succ = \sum_{k \geq 0} (A+BF)^k \,\mathrm{Im}B$.

The controllability subspace $\mathcal{R}$ is characterized by the fact that for every $x \in \mathcal{R}$, there exists a continuous control $u(t)$ such that every state x is reachable from the origin along a controlled trajectory that is wholly contained in $\mathcal{R}$.

Now write $\mathcal{C}(A,B;\mathcal{K}) = \{\mathcal{R}/\mathcal{R} \in \mathcal{C}(A,B), \mathcal{R} \in \mathcal{K}\}$. $\mathcal{C}(A,B;\mathcal{K})$, the set of controllability subspaces contained in $\mathcal{K}$ has a unique largest or supremal element $\mathcal{R}^*$. It is named the supremal controllability subspace of $\mathcal{C}(A,B;\mathcal{K})$. It is the limit of the algorithm (3). Other invariant subspaces, such as the conditionnaly invariant subspace containing ImB can be characterized. We do not recall them.

2.2 Characterization of the invariant subspaces.

The previous subspaces emphasize the structure of the system, and allow us to solve fundamental problems using the structural properties of the system. Particular lists of integers which are specific invariants of the system have been characterized in a paper of Morse [5]. There are a list of polynomials and three lists of integers. These lists emphasize the structure of (C,A,B) under a group of transformations and are closely related to the invariant subspaces.

Some specific relations between the invariant subspaces are recalled, as well as the properties on their dimension. They simplify their computation.

$$\mathcal{R}^* \subset V^* \subset \mathrm{Ker}C = \mathcal{K} \qquad \& \qquad \dim V^* = n - \Sigma n_i' \quad \& \quad \dim V^* = q + \dim \mathcal{R}^* \qquad (4)$$

with $\dim \mathcal{R}^* = 0$ if $m \leq p$. q is the number of invariant zeros that is the zeros [7] of the system matrix *P(s)*, and $\Sigma n_i'$ is the sum of the order of the infinite zeros.

It means that ($\dim V^*$) modes will be hidden from the output and this is the maximal number of modes with this property. $\mathrm{Dim}\mathcal{R}^*$ of these modes are arbitrarily assignable. In case of single output systems, $V^* = \bigcap_{k=1,\ldots,n_i} \mathrm{Ker}CA^k$.

2.3 Finite and infinite structure

By inspecting relations (4), it is concluded that the finite and infinite structures of linear systems point out the dimension of the (A,B) invariant subspaces. The study can be decomposed into several steps. The first one concerns the invertibility of the system, by studying the rank of the system matrix, for example. In a second step, the number of finite zeros and infinite zeros is found. The first ones are the zeros of the system matrix and the second ones are characterized by the Smith McMillan form at infinity of the transfer matrix.

The (A,B) invariant subspaces are then calculated by implementing the previous algorithms (4), by taking into account this information.

2.4 Graph theoretic approach.

A graph theoretic approach is proposed in [1] for sparce systems. The subspaces are no more characterized by vectors but by a set of nodes which are associated with state variables. Some well-known problems are solved. Other problems are solved in [6] once more from a graph theoretic approach but not necessarily for sparce systems.

3. BOND GRAPH APPROACH

3.1 Structural study

The geometric concept for the study of multivariable linear systems requires a state space representation. From a bond graph model, it is possible to find a state space representation. In [8], it is shown that structural properties such as the structural rank of the state matrix or the controllability matrix can be obtained directly from the bond graph model by causal manipulations. In [10], the finite and infinite structure of bond graph models are highlight. The controllability and observability subspaces for bond graph models are calculated in [9]. In this paper we propose a new characterization of the invariant zeros from a bond graph approach:

These different properties allow us to characterize the (A,B) invariant subpaces for bond graph models. Some results proposed from a graph theoretic approach are used.

3.2 Invariant zeros

The Smith form of the system matrix $P(s)$ points out the number of invariant zeros. For square invertible systems, the number of invariant zeros is superior or equal to the number of dymamical elements which remain on the bond graph model when the dynamical elements contained in the shorter different causal paths between all the outputs and the inputs are removed.

For non square systems, the maximal number of different causal paths between the inputs and the outputs can not be obtained. The determinant of $P(s)$ cannot be calculated, therefore some minors of maximal dimension have to be calculated. The common polynomials highlight the invariant zeros. The minors are obtained by removing some columns of $P(s)$ when $m>p$. From another point of view, it is enough to add m-p rows with 0 elements everywhere except for one position per row corresponding the column which has to be removed. Then, the determinant of the augmented matrix can be calculated.

From a bond graph point of view, it is enough to consider m-p dynamical elements at a time, as m-p output detectors and to calculate the number of invariant zeros of the new bond graph model as for square models.

For a sake of place, the methodologies are proposed on simple examples.

3.3 Examples

a) Non observable system

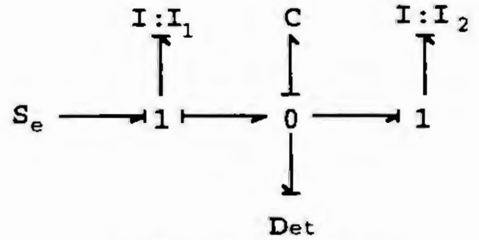


Fig. 1 Non observable system

The BG-rank of the state matrix is 2, the BG-rank of the controllability matrix is 3 and the BG-rank of the observability matrix is 2, [8]. The system is then not state observable. The minimal length of the causal path between the output detector and the input source is 2. The order of the infinite zero is then 2 (for monoviariable systems, it is the difference between the degree of the denominator and the numerator of the transfer function). When the dynamical elements in the previous causal path are removed, the second inertial element stays alone. It means that there is an invariant zero at $s=0$.

Equation (4) induces that the dimension of V^* is equal to 3-2=1 and the dimension of $\mathcal{R}^*$ is 0 because $m=p=1$. At most, $V^* = \text{Ker}C \cap \text{Ker}CA = \text{Ker}\begin{bmatrix} C \\ CA \end{bmatrix}$ which is in that case the non observable subspace. It is directly characterized on the bond graph model by implementing a derivative causality assignment [9]. It comes $V^* = \text{Span}\left\{\begin{bmatrix} I_1 & I_2 & 0 \end{bmatrix}^t\right\}$. In that case, the inplementation of algorithms (2) and (3) is not required.

b) Non observable system with two inputs.

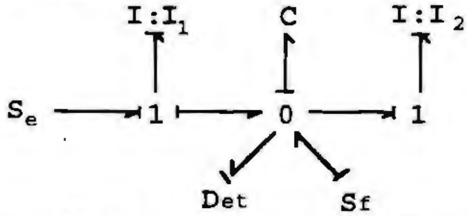


Fig. (2) Non observable system with two inputs

The bond graph model of figure (2) has two input sources and one output detector. The system matrix is not square. The order of the infinite zero is 1. It is the shorter path length between the output and the two inputs. The dimension of V^* is equal to 3-1=2. Then, $V^* = \mathrm{Ker}C = \mathrm{Span}\left\{\begin{bmatrix}1\\0\\0\end{bmatrix},\begin{bmatrix}0\\1\\0\end{bmatrix}\right\}$. These vectors are directly obtained from the bond graph model.

As $m=2$ and $p=1$, one dynamical element is replaced by an output detector. When I_1 is replaced by an output detector, s can be factorized in the corresponding determinant and when the two other dynamical elements are considered as output detectors, the system is no more invertible, that is the minors are null. Then $s=0$ is a structural invariant zero. From (4), it is concluded that $\dim\mathcal{R}^*$=2-1=1. From (3), it comes $\mathcal{R}^*=\mathrm{Span}\left\{\begin{bmatrix}1\\0\\0\end{bmatrix}\right\}$.

4. CONCLUSION

In this paper, we have provided a new characterization of bond graph models in terms of geometric tools. The graph theoretic, geometric or algebraic tools are shown to be relevant to solve problems in linear multivariable control synthesis directly from the bond graph model. These remarks are proposed on two simple examples. A synthesis will be made in a future work.

5. REFERENCES

[1] N. Andrei, "Sparse systems: Digraph approach of large scale linear systems theory". ISR, Verlag TÜV Rheinland, 1985.

[2] P.J. Antsaklis, T.W.C. Williams, "On the dimension of the supremal (A,B)-invariant and controllability subspaces". I.E.E.E. Trans. on Automatic Control, Vol. AC-25, n°6, pp. 1223-1225, 1980.

[3] C. Commault, J. Descusse, J.M. Dion, J.F. Lafay, M. Malabre, "New decoupling invariants: the essential orders". Int. J. Control, Vol.44, n°3, pp.689-700, 1986.

[4] Descusse, J.F. Lafay, M. Malabre, "Solution to Morgan's problem". IEEE Trans. on Aut. Control, Vol.AC-33, n°8, pp.732-739, 1988.

[5] A.S. Morse, "Structural invariants of linear multivariable systems". SIAM J. of Control & Opt., Vol.11, n°3, pp.446-465, 1973.

[6] K.J. Reinschke, "Multivariable control. A graph-theoretic approach". Lecture Notes in Cont. and Inf. Sc., Vol.108, 1988.

[7] C.B. Schrader, M.K. Sain, "Research on system zeros: a survey". Int. J. Control, Vol.50, pp.1407-1433, 1989.

[8] C. Sueur, G. Dauphin-Tanguy, "Bond graph approach for structural analysis of MIMO linear systems". J. Franklin Inst., Vol.328, n°1, pp.55-70, 1991.

[9] C. Sueur, G. Dauphin-Tanguy, "Bond Graph determination of controllability subspaces for pole assignment". IEEE SMC, Inter. Conf. on Systems, Man and Cybernetics, Systems Engineering in the Service of Humans, Le Touquet, France, pp. ,oct 1993.

[10] C. Sueur, G. Dauphin-Tanguy, "Poles and Zeros of multivariable linear systems: a bond graph approach". Bond Graphs for Engineers, P.C. Breedveld and G. Dauphin-Tanguy (Editors), Elsevier Science Publishers BV (North Holland) pp. 211-228, 1992, IMACS.

[11] W.M. Wonham, "Geometric state-space theory in linear multivariablecontrol: a status report". Automatica, Vol.15, pp.5-13, 1979.

[12] W.M. Wonham, "Linear multivariable control : a geometric approach". Spinger-Verlag, Applications of mathematics, Second Edition, 1979.

ON REPRESENTING HYDRAULIC CONTROL SYSTEM MODELS WITH DISCONTINUITIES IN THE BOND GRAPH FRAMEWORK

1. MATHMOD VIENNA, 1994 IMACS Symposium on Mathematical Modeling, Vienna, Austria

W. Borutzky
Department of Computer Science
Cologne Polytechnic, 51643 Gummersbach, FRG
Fax: +49.2261.8196.15 E-mail: bj020@rs1.rrz.Uni-Koeln.de

J. U. Thoma
Thoma Consulting, Bellevueweg 23, CH 6300 Zug, Switzerland
IMAGINE, Maison Productique, F 42300 Roanne, France

Abstract

In modeling hydraulic control systems it is convenient and common to describe phenomena like stops of pistons or devices like check valves discontinuously. On the other hand, their is still an ongoing discussion how to treat adequately discontinuities in the bond graph framework. Following an approach proposed in [1] in this paper a hydraulic-mechanical damping network in which check valves of a rectifier bridge and the limitation of spool displacement in a spool valve give rise to a partly discontinuous description is represented both by means of a small Petri net and three bond graph models. Each bond graph model corresponds to one of the system's (major) modes of operation. Different modes of operation and transitions between them are given by a Petri net. The model was manually coded in ACSL. During simulation it is switched from one initial value problem to another using values from the previous model for initial values in the subsequent model.

1 Introduction

In the bond graph framework modeling starts by considering energy flows in a system which are associated with an exchange of physical quantities like matter, momentum, electrical charge etc. between subsystems. This exchange of quantities happens continuously in time, and physical modeling must obey corresponding fundamental conservation principles. By consequence, the abstraction of state transitions taking place virtually instantaneously in a macroscopic time scale cannot be expected to represented adequately in a bond graph with standard elements. Energy that passes through a system is stored temporarily to some extent and it takes a finite time to change the content of energy in a subsystem. That is, a mathematical model contains Ordinary Differential Equations (ODEs). In principle, ODEs may also describe instantaneous changes if variables are allowed to be generalized functions. Moreover, since the energy exchange between system components is bound to permanent interconnections like mechanical links, electrical wires, or hydraulic pipes in most cases, bond graph representations implicitly adhere to a time invariant system topology. However, when a valve spool reaches a stop and an energy exchange between the spool and the valve casing takes place during a short time interval, then there is a change in the problem structure. Such cases lead to energy links that do not exist permanently but depend on conditions.

On the other hand it is common practice to neglect the dynamics of certain state transitions with respect to the overall dynamic behavior of engineering systems containing devices like for instance, thyristors or hydraulic check valves. Hence, in graphical representations like circuit diagram switches are used. In hydraulic networks the model associated with the standard symbol of a check valve may be discontinuous, and block diagrams may represent any functional relations.

Extending the standard bond graph language in order to cope with models containing discontinuities has been subject of growing interest recently. Common to several approaches is the introduction of an ideal switching element and the aim of using one single bond graph of fixed structure. In contrary, Borutzky,

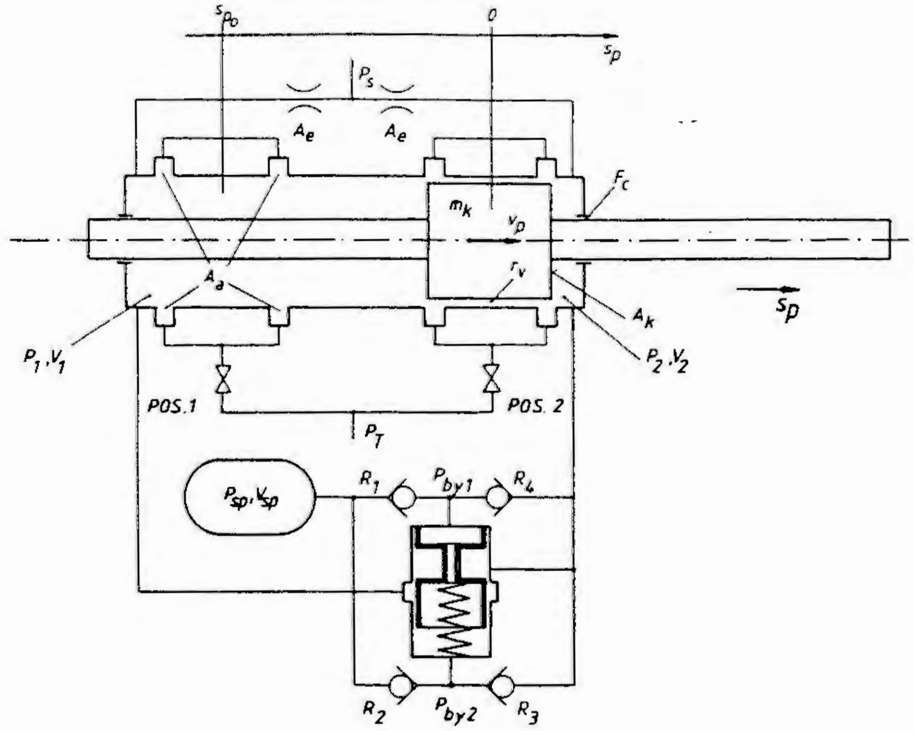


Figure 1: hydraulic positioning drive with damping network

Broenink, and Wijbrans proposed in [1] a graphical description that uses a Petri net or a Finite State Machine (FSM) to display (major) states or modes of operation in a system and the different possible state transitions, while for each mode of operation a bond graph or a set of (disjointed) bond graphs describes the dynamic behavior of the system. That is, the bond graphs exclusively composed of standard elements are used only during a time window of the overall simulation interval. Both approaches have their advantages and disadvantages which are discussed by Borutzky in [1]. In this paper a combination of both approaches is applied to a hydraulic-mechanical damping network.

The need for switching elements was cited by Thoma in [8], where they were called time dependent junctions (tdj). Such elements can be reticulated as modulated resistors, or as resistors connected to an MTF [4], similarly as used here for check valves. The aim of the present work is to make the bond graph computationally efficient.

2 The system

The system we shall consider in this paper is the hydraulic positioning system shown in Fig. 1. It comprises the hydraulic drive in the upper part of Fig. 1 with a piston of mass m_K in target position 2 and a control valve for each position, and a damping network with a bypass valve depicted in the lower part of Fig. 1. For reasons of clarity only two of four possible positions are shown. The purpose of the damping network is to improve the dynamic behavior of the hydraulic positioning drive by establishing dynamically a bypass leakage between the chambers in the actuating cylinder only when the mechanical load is oscillating around a final position, while fast movement from one position to another must not be affected significantly. Even if overshoots at a final position can be tolerated, badly damped oscillations must be avoided because they decrease the stiffness of the hydraulic positioning drive in a target position, or in other words, they increase the sensitivity to changes of the load. To achieve that goal a high pass filter is connected to chamber 2 of the hydraulic positioning drive. It sensors pressure rates of frequencies above its natural frequency in that chamber that are due to an oscillation of the actuating piston around its final position. That is, the system has a non symmetric structure. The pressure drop across the orifice of the high pass filter controls the opening of a bypass valve connecting the chambers of the hydraulic positioning drive. Since the bypass valve can open only in one direction against the spring the pressure drop acting on its spool is rectified by means of a bridge with check valves. Hence, the frequency of the pressure drop across the spool in the bypass valve is twice the frequency of the pressure drop across the actuating piston of the positioning drive. To reduce sensitivity of the damping network with regard to low pressure drops, e. g. due to noise the spool of the bypass valve is biased by means of a spring. In addition, the maximum opening of the bypass valve can be adjusted by means of a screw (not shown in Fig. 1. Such damping networks were designed and tested by K. Engelsdorf [5]. A bond graph model of the system based on some simplifications was given by Borutzky in [3]. A more elaborated model is proposed in this paper.

3 A Model containing Discontinuities

The damping network was manufactured as a small scaled integrated device in which check valves with a low set pressure of .5 bar were inserted. Hence, it is justified to neglect their fast transients between on- and off-mode, and to model them discontinuously as a non-ideal switch with a small on-resistance controlled by the pressure drop across the check valve. Following G. Dauphin-Tanguy and her co-workers [4] the check valves are represented in the bond graph by means of a modulated transformer and a resistor (Fig.2).

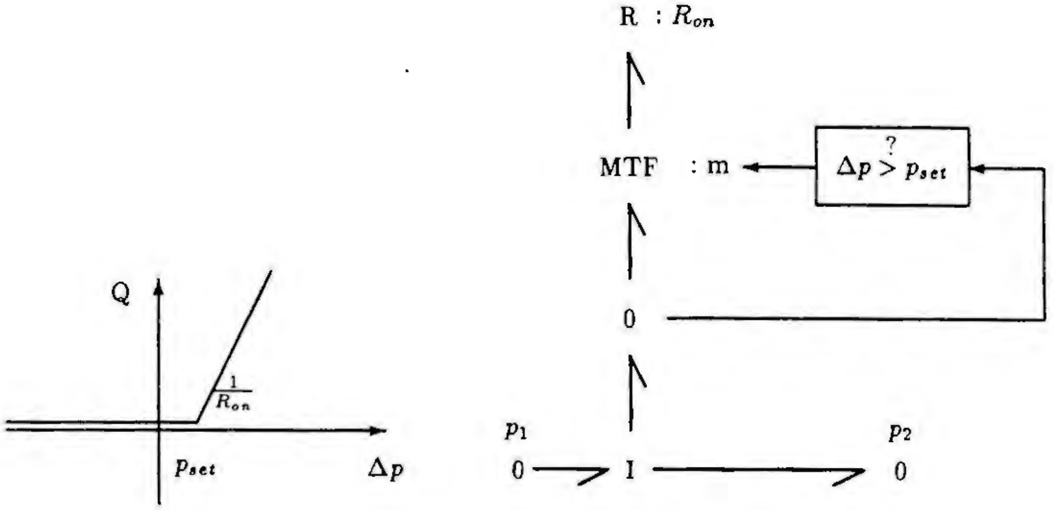


Figure 2: piecewise linear characteristic and bond graph model of a check valve

Now, consider the spool of the bypass valve. Since the valve only opens in one direction the spool hits the valve casing when it closes the valve. Another stop is introduced by the screw that limits the opening of the bypass valve. Both stops may be represented in a bond graph in a conventional manner by means of a capacitor of very low capacitance and a resistor of rather high resistance. That is, the model would account for strongly damped oscillations of high frequency when the valve spool is in contact with a stop. A drawback of this straightforward approach is that the mathematical equations are stiff. Of course, this is not a problem for the Backward Differentiation Formula (BDF) method. However, if in addition other discontinuities like Coulomb friction have to be modeled properly a stiff stable multistep method is not very efficient since it must restart at low order after the discontinuity has been located. Moreover, such oscillations are of no concern for the overall dynamic behavior in most cases. Hence, it is natural to assume that the kinetic energy of the valve spool is lost instantaneously when the stop is reached. (The valve spool acceleration reverses if the sign of the net force acting on the it changes.) Of course, as discussed above such an idealized model cannot be represented by means of a standard bond graph. (A storage element would empty instantaneously without causing a transient.) To cope with the abstraction of an instantaneously absorption of kinetic energy a bond graph model for each mode of operation is introduced. From a mathematical point of view this means that computation of the system behavior switches from one initial value problem (IVP) to another by using values from the previous model as initial values of the subsequent model.

Concerning the corresponding overall Petri net three different modes can be distinguished (cf. Table 1). Either the bypass valve is closed (state 0), then there is no bypass leakage between the chambers of the positioning drive, or it is fully open (state 2), then the spool of the bypass valve is at the stop introduced by the screw, or the spool is moving from one stop to the other (state 1) that is, we have a mechanical mass-spring oscillator in the bypass valve. In contrast to the check valves of the rectifier bridge the values of the valve spool inertia and the biasing spring are such that the valve in the bypass leakage cannot be considered as a switch neglecting its dynamic behavior. Hence, all state transitions take place through state 1. The Petri net of the damping network is shown in Fig. 3. An attribute c_{ij} at a transition denotes the condition under which a transition from state i to state j takes place. For instance, if the pressure across the spool predominates the biasing spring the bypass valve can open, and the damping network changes from state 0 (no bypass leakage) to state 1 (valve partly open). If the valve is fully open it remains open until the net force acting on the spool reverses its sign (condition c_{21}. Now, consider state 1 in which the spool of the bypass valve is moving between both stops. In that case a flow force is acting on the valve spool which tends to close the valve. Taking that force into account by using the bond graph representation proposed by Borutzky in [2] a bond graph for that mode of operation can be derived from the system schematic 1 by application of the standard procedure. The result is shown in Fig. 4. In fact, in the equations of the mathematical model only the static component of the flow force was taken into account. In the opinion of the first author such a simplification cannot be represented in a true bond graph without activated bonds [2]. In case the bypass valve is closed only the high pass filter connected to chamber 2 of the positioning drive is active. On the other hand, as long as the valve is fully open we have an orifice of fixed cross section area in the bypass leakage and the high pass filter sensoring pressure oscillations in chamber 2. For both modes of operation the corresponding bond graph is a special case of that depicted in Fig. 4. Therefore, due to the lack of space not all bond graphs of the system are shown.

0	:	valve closed	:	no bypass leakage
1	:	valve partly open	:	moving valve spool
2	:	valve fully open	:	valve spool at stop

Table 1: Modes of operation

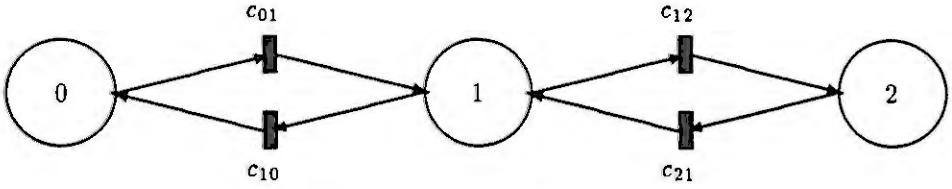


Figure 3: Petri net of the damping network

4 Simulation results

For validation of the damping network model proposed above its equations were formulated manually in ACSL. For each mode of operation the dynamic equations were derived manually from the corresponding bond graph and collected in a block of an IF THEN ELSE statement in the DERIVATIVE section. Calculation of the conditions for state transitions that is, for switching between the different dynamic models within the DERIVATIVE section is accompanied by SCHEDULE statements which allow for locating an event. The abstraction that the kinetic energy is lost instantaneously when the valve spool hits the stop can be expressed at block diagram level more easier by means of the double limited integration

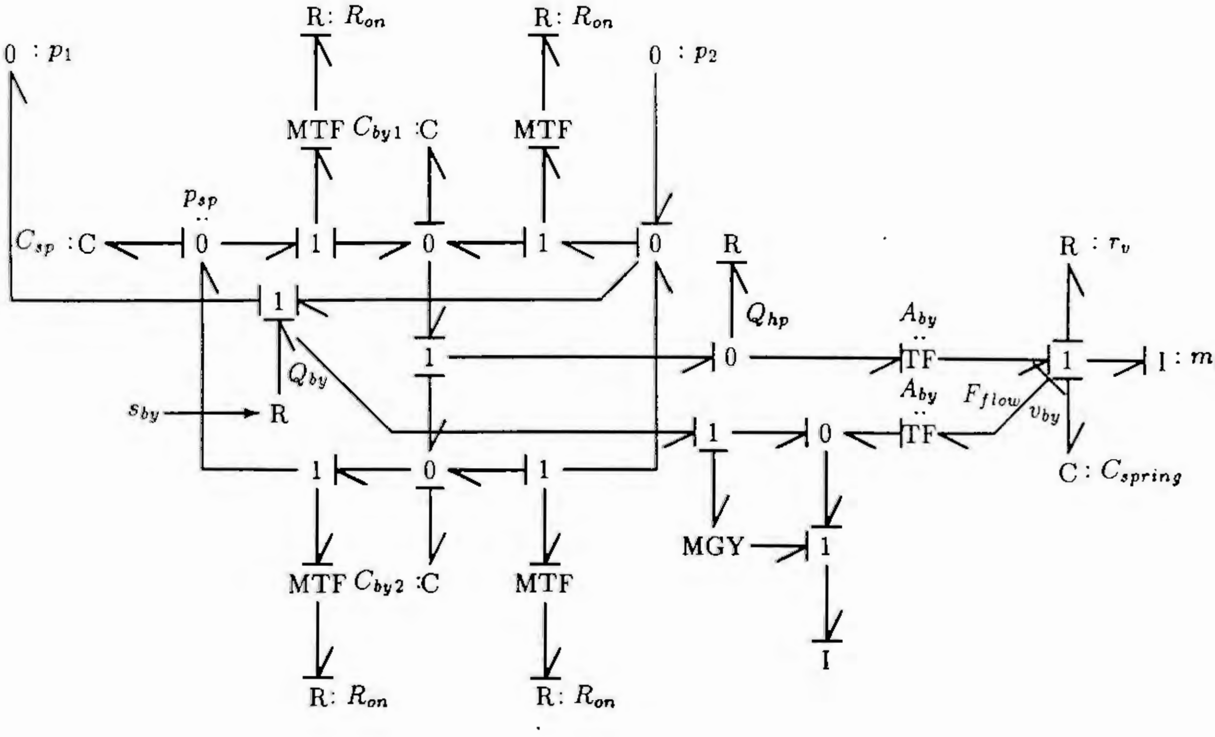


Figure 4: bond graph of the damping network in mode 1

operator DBLINT provided by ACSL. That functional block is very convenient for description of the problem. The displacement is limited to the lower or upper bound, and the velocity remains zero until the net force acting on the valve spool changes in sign. However, such an operator does not fit well in the bond graph language unless a combination of bond graphs and block diagrams is used as it is common if the entire feedback loop of a system and its control system is to be modeled.

The damping network was not simulated as a system in its own but connected to the hydraulic positioning drive to be controlled. However, due to the lack of space and since we want to focus on that part of the overall system model containing discontinuities the conventional bond graph of the hydraulic positioning drive is not shown. The latter is stimulated by opening the orifice corresponding to the target position (position 2) according to a step function. The dynamic response of the system is shown in Fig. 5. The upper curve in Fig. 5 shows the velocity v of the piston of the positioning drive. As can be seen, after opening the orifice of position 2 the actuating piston moves towards that position with an increasing velocity. After about 25ms it closes the first outlets of the target position. By consequence, it is slowed down in order to avoid too large overshoots. At about 50ms the actuating piston totally closes the orifice and enters into a damped oscillation around position 2.

The lower curve in Fig. 5 shows the displacement sby of the valve spool. As expected, the valve remains fully open, allowing for a maximum bypass leakage when the actuating piston of the positioning drive is oscillating around its target position. More biasing of the spring would result in a periodically closing of the bypass valve which reduces the damping effect of the network on the positioning drive.

There is one undesired effect that is related to the principle of operation of the damping network and hence, cannot be prevented. After opening of the orifice corresponding to the target position, chamber 2 of the hydraulic drive as well as the volume of the high pass filter discharge rapidly through that orifice. By consequence, there is a high gradient of the pressure drop across the actuating piston of the positioning drive. This gradient corresponds to high frequencies sensored by the high pass filter and thus, the valve in the bypass leakage opens. The time for which the valve remains open as well as the time for closing cannot be reduced witbout affecting the opening of the valve when the actuating piston of the positioning drive is oscillating around the target position. By consequence, due to that undesired bypass leakage the

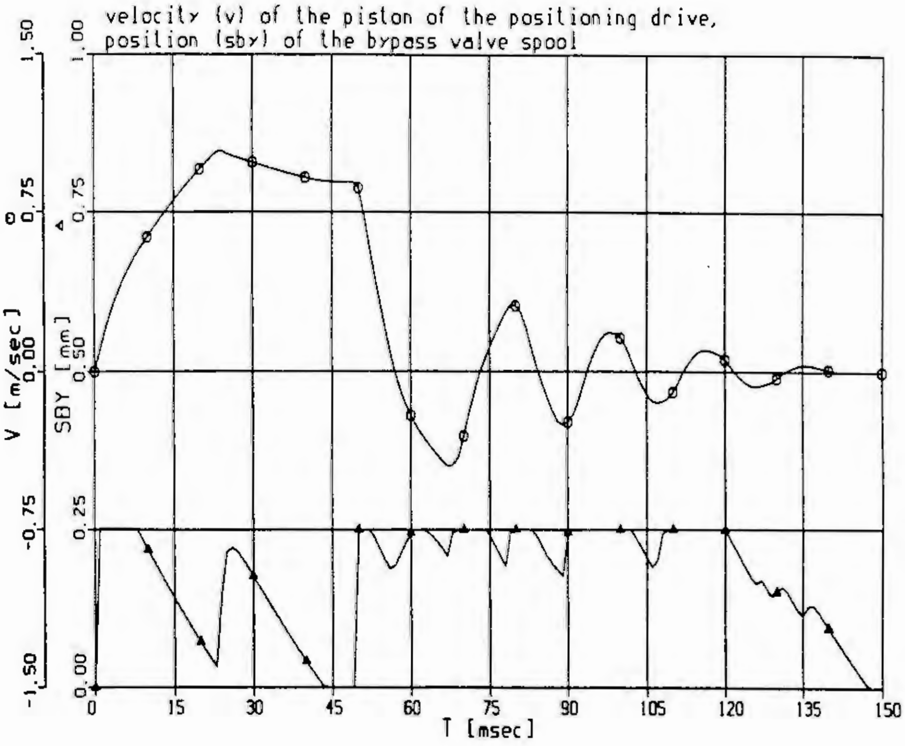


Figure 5: Simulation results

maximum velocity the actuating piston can attain when moving towards the target position is somewhat decreased. The time needed to close the bypass valve after the oscillations of the positioning drive have been damped, depends on the hydraulic capacity of the high pass filter which in turn determines the natural frequency of the filter. During that time for closing the positioning drive still remains sensitive to changes of the load. Nevertheless, as a major result, the simulation shows that the considered damping network fulfills its purpose of efficiently damping the oscillations of the positioning drive. Moreover, the simulation results agree well to measurement results of K. Engelsdorf [5].

5 Conclusions

By considering a dedicated hydraulic mechanical control system it was shown that a model containing discontinuities can be represented in the bond graph framework without introducing a switch element. The underlying idea presented in [1] is to develop different bond graphs for different modes operation by applying standard bond graph procedures and to represent the different modes by means of a Petri net. Of course, if we would use the abstraction of n binary switches and one set of switch states defines one state of the system we would have to consider a large number of up to 2^n possible system states. Therefore, it is important to concentrate on major modes of operation. Therefore, we did not consider the different states of the check valves in the bridge but adopted the model of an electrical diode proposed by G. Dauphin-Tanguy and her co-workers [4]. This is also reasonable because the check valves permanently interconnect system components, although there is not an energy exchange through them at all times. (In contrary, the valve spool and its stops are not always connected.) The result is a small set of three bond graphs for the damping network corresponding to the major modes of operation listed in Table 1. The aim of the approach in this paper has been to avoid contradictions to the fundamentals of the bond graph language without restricting a pragmatic development of realistic models. From a mathematical point of view the approach means that a (small) set of Initial Value Problems (IVPs) must be solved for one single system.

References

[1] *Borutzky, W.; Broenink, J. F.; Wijbrans, K. C. J:* Graphical Description of Physical System Models Containing Discontinuities, Proc. of the 1993 European Simulation Multiconference, Lyon, June 1993, pp. 203-207

[2] *Borutzky, W.:* A Dynamic Bond Graph Model of the Fluid Mechanical Interaction in Spool Valve Control Orifices, Bond Graphs for Engineers, North-Holland, 1992, pp. 229-236

[3] *Borutzky, W.:* Ein Beitrag zur Bondgraphmodellierung in der Feinwerktechnik am Beispiel miniaturhydraulischer Geräte, Dr.-Ing. Thesis, Technische Universität Braunschweig, Germany, 1985

[4] *Durcreux, J. P.; Castelain, A.; Dauphin-Tanguy, G.; Rombaut, C.:* Power electronics and electrical machines modeling using bond-graphs, Bond Graphs for Engineers, North-Holland, 1992, pp. 121-133

[5] *Engelsdorf, K.:* Auslegung hydraulischer Positionierantriebe mit Dämpfungsnetzwerken, Dr.-Ing. Thesis, Technische Universität Braunschweig, Germany, 1984

[6] *Guillon, M.:* Hydraulische Regelkreise und Servosteuerungen, Hanser-Verlag, Munich, 1968

[7] *Thoma, J.U.:* Simulation by Bondgraphs, Springer-Verlag, 1990

[8] *Thoma, J.U.:* Introduction to Bondgraphs, Pergamon Press, 1975

AUTOMATED SIMULATION OF BOND GRAPH MODELS

Bruno ARNALDI and Bénédicte EDIBE
*IRISA */ INRIA †*
Campus de beaulieu
35042 Rennes Cedex
France
Tel: (33) 99.84.72.61 Fax: (33) 99.38.38.32 E-mail: edibe@irisa.fr, arnaldi@irisa.fr

Abstract

This paper presents our work on automated modeling and simulation of physical systems with bond graphs. A symbolic computation based method performing the derivation of dynamic equations from a bond graph model has been realized, then extended to multibond graphs. We focus now on the automation of multibody systems modeling. The systematic modeling of joints is organized by using joint coordinate systems and adapting a generic joint bond graph model to each type of joint. A standard library of joint bond graph models allows the automated modeling of a large number of articulated mechanical systems.

1 Introduction: context of our work

The program we outline here is developed within the context of previous and actual researches of our team (SIAMES [1]) on simulation of mechanical systems [ADH89], [ADH91]. A simulation program based on the principle of virtual work was conceived. This system performs automatically the simulation of mechanical systems described in terms of objects, joints, constraints and initial conditions. Figure 1 shows the main phases of the automatic process. The different stages from the interactive geometric modeling are respectively: extraction of the mechanical properties of the object (center of gravity, inertia matrix, principal coordinate system of inertia); symbolic derivation of motion equations; solving of these equations. The operations are carried out as far as possible with algebraic computation.

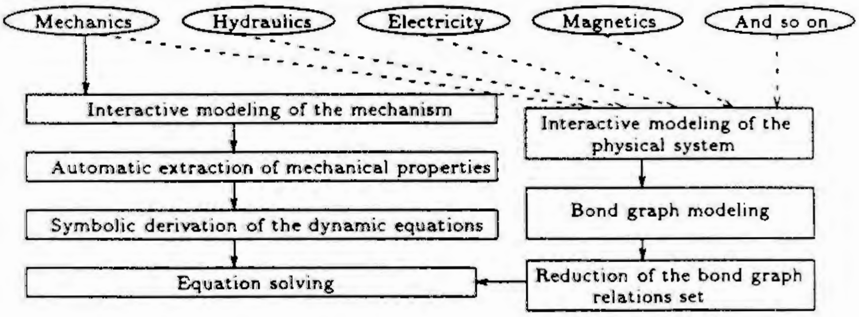

Mechanics
Hydraulics
Electricity
Magnetics
And so on
Interactive modeling of the mechanism
Automatic extraction of mechanical properties
Symbolic derivation of the dynamic equations
Equation solving
Interactive modeling of the physical system
Bond graph modeling
Reduction of the bond graph relations set

Figure 1 : Simulation of mechanical systems and simulation of bond graph models

In the continuity of this work, the bond graph approach should allow us to broaden our application field. We would like to be able to simulate pluridisciplinary physical systems. The stages of our simulation program based on bond graph modeling are the following: interactive physical modeling of the system; translation to the bond graph model of the system; derivation of dynamic equations; solving of these equations. The different steps of this program are represented on the right side of figure 1 (dotted lines). The two stages we are working on are the automation of the bond graph modeling and the derivation of dynamic equations from a bond graph model.

*Institut de Recherche en Informatique et Systèmes Aléatoires.
†Institut National de Recherche en Informatique et Automatique.
[1] Synthèse d'Image, Animation, Modélisation et Simulation

2 Symbolic reduction of the initial bond graph relations set

2.1 Overview of the method

The dynamic equations of a physical system modeled with bond graphs are derived by reducing the mathematical model linked to the bond graph model. Our reduction method consists in an ordered series of symbolic substitutions within the bond graph constitutive relations set. The set of equations obtained may be a non linear implicit DAE system. Derivative causalities and algebraic loops, which are handled at the solving level, do not condition the reduction operation. A prototype of the reduction process was devised with the symbolic computation system MAPLE. We presented a detailed description of this reduction method in a previous article [EA93]. Since then, the algorithmic organization of the substitutions has been optimized and the method has been extended to multibond graphs.

2.2 example

The following example shows a result of reduction. Figure 2 represents a small mechanical system and a bond graph model of this system (example taken from [RK83]). The set of initial bond graph relations and the reduced set of dynamic equations are listed below. The example, chosen as an illustration, is very simple but it must not conceal the fact that the reduction method is efficient when applied to much more complex systems. The idea we want to underline here is that this reduction is realized by performing a bond graph specific series of symbolic substitutions.

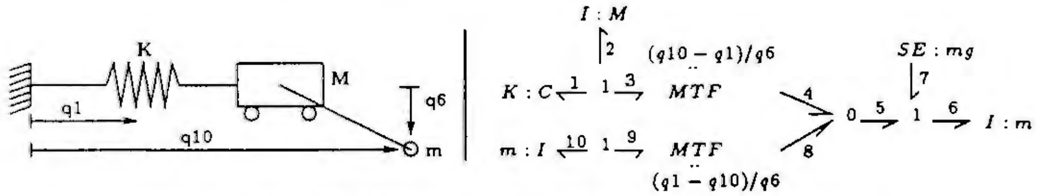


Figure 2 : Mechanical system and bond graph model of this system

$$\text{rel1} := \dot{p}_1 = K * q_1;$$
$$\text{rel2} := -\dot{p}_1 - \dot{p}_2 - \dot{p}_3 = 0;$$
$$\text{rel3} := \dot{p}_2 = M * \ddot{q}_2;$$
$$\text{rel4} := \dot{p}_3 = (q_{10} - q_1)/q_6 * \dot{p}_4;$$
$$\text{rel5} := \dot{q}_4 = (q_{10} - q_1)/q_6 * \dot{q}_3;$$
$$\text{rel6} := \dot{q}_4 + \dot{q}_8 - \dot{q}_5 = 0;$$
$$\text{rel7} := \dot{p}_5 + \dot{p}_7 - \dot{p}_6 = 0;$$
$$\text{rel8} := \dot{p}_6 = m * \ddot{q}_6;$$
$$\text{rel9} := \dot{p}_7 = m * g;$$
$$\text{rel10} := \dot{p}_9 = (q_1 - q_{10})/q_6 * \dot{p}_8;$$
$$\text{rel11} := \dot{q}_8 = (q_1 - q_{10})/q_6 * \dot{q}_9;$$
$$\text{rel12} := -\dot{p}_9 - \dot{p}_{10} = 0;$$
$$\text{rel13} := \dot{p}_{10} = m * \ddot{q}_{10};$$
$$\text{rel14} := \dot{q}_2 = \dot{q}_1;$$
$$\text{rel15} := \dot{q}_3 = \dot{q}_1;$$
$$\text{rel16} := \dot{p}_8 = \dot{p}_4;$$
$$\text{rel17} := \dot{p}_5 = \dot{p}_4;$$
$$\text{rel18} := \dot{q}_5 = \dot{q}_6;$$
$$\text{rel19} := \dot{q}_7 = \dot{q}_6;$$
$$\text{rel20} := \dot{q}_9 = \dot{q}_{10};$$

$$\text{equ1} := (q_{10} - q_1)/q_6 * \dot{q}_1 + (q_1 - q_{10})/q_6 * \dot{q}_{10} - \dot{q}_6 = 0;$$
$$\text{equ2} := 1/(q_{10} - q_1) * q_6 * (-K * q_1 - M * \ddot{q}_1) + m * g - m * \ddot{q}_6 = 0;$$
$$\text{equ3} := -K * q_1 - M * \ddot{q}_1 - m * \ddot{q}_{10} = 0;$$

3 Automated bond graph modeling of multibody mechanical systems

The second part we are working on is the bond graph modeling process. We are planning to automate it as much as possible. As a first step, we are focusing on the automated modeling of multibody mechanical systems.

3.1 Bodies and connections

Our work on automation of such a modeling is related to researches on modeling of multibody systems in terms of multibond graphs [Bos86], [TB85], [BT85]. The systematic modeling is based on modularity which is a fundamental property of bond graphs. The bond graph submodel of each body of the system is constructed by adapting to the represented body a generic submodel; this generic submodel corresponds to the bond graph model of a single freely moving body with one or more hinge points (see figure 3). The notation used is the following one: a velocity (linear or angular) $v_k^{i,j}$ is the velocity of point k, with respect to base j; the coordinates of this vector are expressed in base i. For further details on the body model, refer to [Bos86].

The systematic modeling of joints is considered next.

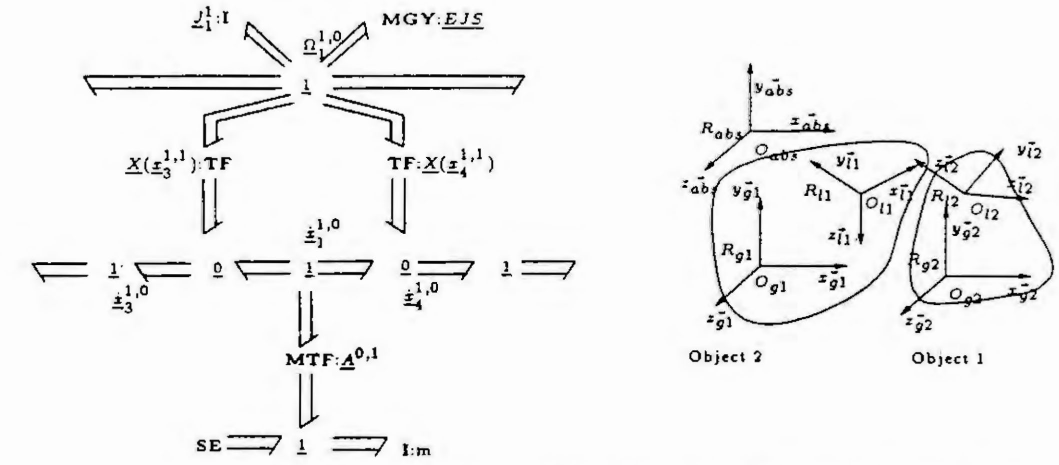


Figure 3 : Body bond graph model and description of joint reference frame

3.2 Generic bond graph model of a joint

The automation of the bond graph modeling of multibody systems is organized by using joint coordinate systems which are defined during our geometric modeling of the mechanism (see figure 3). These joint coordinate bases are helpful since the joint is not supposed to be necessarily in the orientation of the principal axes of inertia of the bodies. For each type of joint, the directions of the two joint bases axes are chosen in order to describe in the most adequate way the articulation.

Each hinge point is fixed on the body it belongs to, it is not able to move with respect to the body. The motion between the bodies is entirely specified by the rotation and translation transformations between the two joint coordinate bases involved in the articulation.

Figure 4 represents the generic bond graph model of a joint involving 3-dimensional rotation and translation. Numbers 0, 1, 1', 2', 2 denote respectively the absolute reference base, the body 1 inertia reference base, the body 1 joint base, the body 2 joint base and the body 2 inertia reference base.

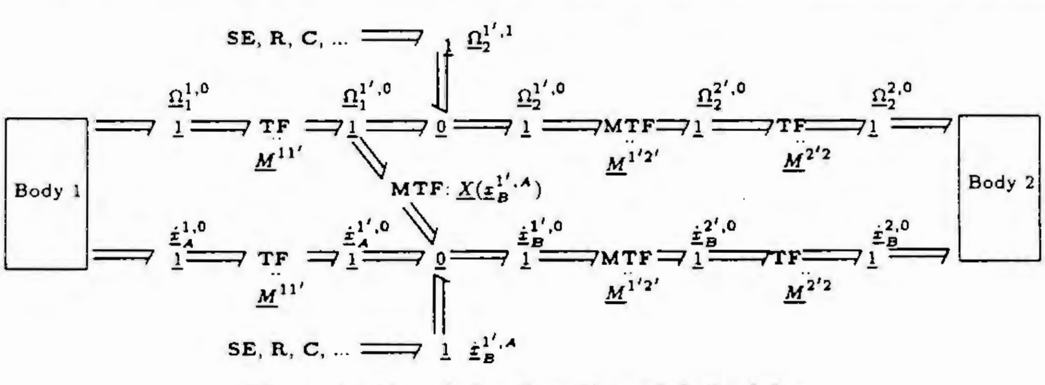


Figure 4 : Generic bond graph model of a joint

This generic bond graph model of a joint shows the different base transitions required to express the geometric transformation from hinge point A on body 1 to hinge point B on body 2. The " TF path" between the rotation level and the translation level exists as soon as there is a translation degree of freedom in the joint.

Beyond the geometrical type of a considered joint, we have to take into account the efforts applied to the joint degrees of freedom (due to springs, absorbers, motors,..). When an effort is applied to a degree of freedom, the bond graph element which "gives" this effort is assigned non zero parameter values.

The use of joint bases is recognized as a sensible way to model systematically the joints. The use of "attachment frames" in the bond graph modeling of joints was proposed yet in another paper [BL88] which establishes the feasibility of a local description of joints and mentions the possibility of elaboration of articulations libraries. Beyond the feasibility of an automatic bond graph modeling of joints, we

4/4

demonstrate here a realization of this automatic process. A library of joint bond graph models is presented in the following. Indeed, the automation of the modeling of multibody systems implies the elaboration of such a library.

3.3 Library of joint bond graph models

The basic library presented on table 1 allows the automated modeling of a large set of articulated systems. To each type of joint corresponds a bond graph model. The joint specific degrees of freedom are expressed with appropriate direct sum decomposition of vector bonds, so that the 0 junctions which exist at the translation level and at the rotation level are effective only along the degrees of freedom axes. The only specifications a joint submodel needs regard the type of efforts applied to the joint degrees of freedom. Everything else is ready for assembly with the bodies submodels.

Nu.	Joint	Rot	Trans	DOFs	Nu.	Joint	Rot	Trans	DOFs
1	Embedding	0	0	0	7	Ball and socket	3	0	3
2	Pin	1	0	1	8	Cyl. on plane	2	2	4
3	Sliding	0	1	1	9	Ball in a cyl.	3	1	4
4	Twist sliding	1	1	1	10	Ball on plane	3	2	5
5	Cylindrical	1	1	2	11	Flying object	3	3	6
6	Plane on plane	1	2	3					

Table 1 : Joint library.

4 Conclusion

Our current purpose is to devise an automatic process from interactive geometric modeling of a multibody system to its dynamic equations. The simulation part was perfected. The original method allowing the derivation of dynamic equations from a bond graph model by performing specific symbolic substitutions has proved its efficiency for the treatment of multibond graphs. The automatic modeling part is under development. Joint coordinate systems are used to describe methodically the joints and specify straightforwardly the rotation and translation degrees of freedom for each type of joint. In this context, a generic joint bond graph model was conceived. It gives the global structure of the models which constitute our library of joint bond graph models. Although a complete automation of bond graph modeling is not always possible, the process we set out in this paper is still applicable to a wide variety of multibody systems.

References

[ADH89] B. Arnaldi, G. Dumont, and G. Hégron. Dynamics and unification of animation control. *The Visual Computer*, (5):22–31, March 1989.

[ADH91] B. Arnaldi, G. Dumont, and G. Hégron. Animation of physical systems from geometric, kinematic and dynamic models. In Springer-Verlag, editor, *Modeling in Computer Graphics*, pages 37–53, IFIP Working conference 91 (TC 5/WG5. 10), apr 1991.

[BL88] M. Beauwin and F. Lorenz. Representation of 3-d mechanisms using 6-dimensional multibonds. In *12th World Congress on Scientific Computation*, pages 27–30, IMACS, 1988.

[Bos86] A. M. Bos. *Modelling multibody systems in terms of multibond graphs, with application to a motorcycle.* PhD thesis, Twente University, 1986.

[BT85] A.M. Bos and M.J.L. Tiernego. Formula manipulation in the bond graph modelling and simulation of large mechanical systems. *Journal of the Franklin Institute*, 319(1/2):51–65, jan/feb 1985.

[EA93] B. Edibe and B. Arnaldi. Automated symbolic reduction of constitutive relations from a bond graph model. In *International Conference on Bond Graph Modeling ICBGM'93, San Diego (USA)*, pages 11–16, SCS The Society for Computer Simulation, jan 1993.

[RK83] R.C. Rosenberg and D. Karnopp. *Introduction to physical system dynamics. mechanical engineering*, McGraw-Hill Book Company, 1983.

[TB85] M.J.L. Tiernego and A.M. Bos. Modelling the dynamics and kinematics of mechanical systems with multibond graphs. *Journal of the Franklin Institute*, 319(1/2):37–50, jan/feb 1985.

AI CONTROLLED SIMULATION EXPERIMENTS*

András JÁVOR

KFKI Research Institute for Measurement and Computing Techniques
H-1525 Budapest, P.O.Box 49., Hungary

Abstract. A methodology for controlling dynamic simulation experiments using distributed AI is outlined. The principles presented have been implemented in the CASSANDRA 3.0 simulation system where knowledge bases and inference engines have been included both in form of demons and in mobile entities of the models. A few examples for applying the methodology are mentioned.

1. INTRODUCTION

The primary aim of AI controlled simulation experiments has been to dedicate the task of the iterative process of *model building - simulation - evaluation - experiment modification* (conventionally performed by the human experimenter) to agents of AI. It has been proposed [1] to apply *demons* monitoring the trajectory of the dynamic simulation procedure and using their knowledge bases and inference engines for modifying the model structures and parameters to obtain a model conform to the expected behaviour which in certain cases could be termed as optimal according to certain criteria. The same methodology can be applied to adjust the experimental frame in order to obtain certain characteristics of the model investigated.

Assuming an object oriented approach where both the conventional building blocks of the model as well as the demons are represented as objects communicating with each other in form of a network, the introduction of the intelligent agents to control the simulation experiments has revealed that their inclusion into the models themselves provides a convenient means to build models of systems of high autonomy. On the one hand the intelligent demons mentioned above can be included as elements of the model [1]. On the other hand *mobile* knowledge bases and intelligence can be implemented into the model structure by their assignment to the mobile entities in the system. A possible and effective solution to this end is provided by the introduction of *Knowledge Attributed Petri Nets (KAPN)* [2], where knowledge bases can be used as attributes of the tokens. The inclusion of AI into the models themselves makes them adaptive and self modifying to which the above mentioned methodologies [1] [2] provide convenient solutions.

*This work was partly sponsored by the National Scientific Research Fund. Contract number: 1847

2. EXPERIMENTAL RESULTS

The theoretical-methodological principles have been tested in different fields, where simulation had to be applied to investigate complex systems.

§a The first experiment has been undertaken in the field of digital logic circuit design. The problem that had to be solved was the determination of the limit speed with which a circuit designed already could operate according to its functional specifications. The input sequences to the model have been tuned to determine the limit speed of operation. The test conditions in various parts of the circuit where the states could be observed were compared to the patterns stored in the knowledge base and production rules were used for a binary search of the input sequence frequencies (see Fig. 1). The investigation was undertaken using an extended version of the LOBSTER MPC simulation system where not all aspects described in the introduction have been implemented only a centralized version of the experimental frame control [3].

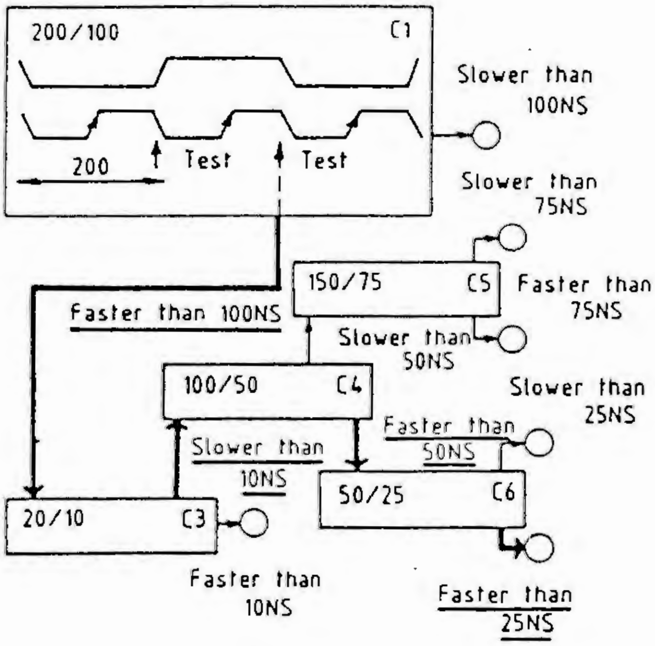
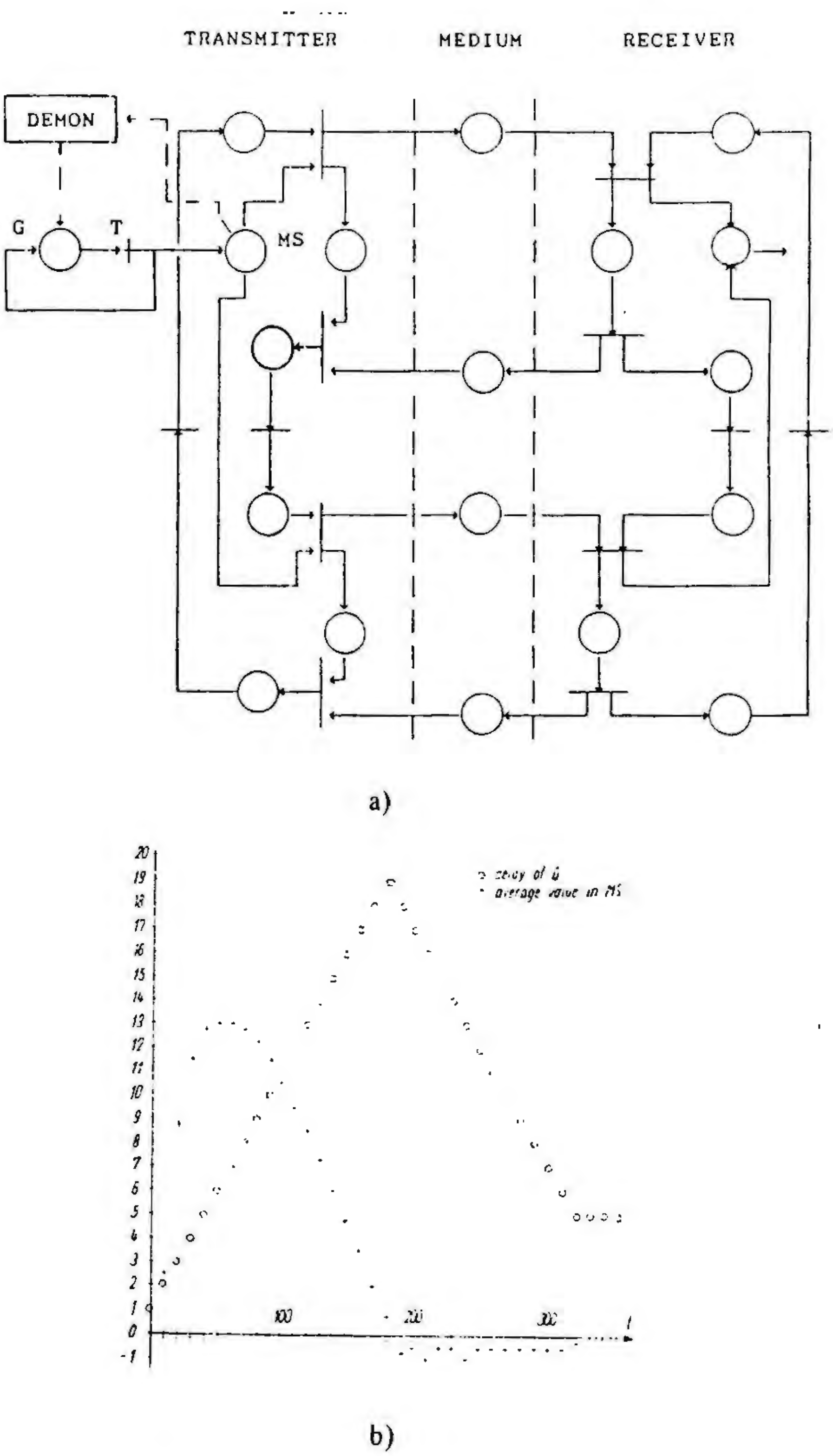


Figure 1 Experiment control for determining the speed of a digital logic circuit

§b A system level investigation of a communication protocol determining its throughput speed was undertaken by the version 1.4 of the CASSANDRA (Cognizant Adaptive Simulation System for Applications in Numerous Different Relevant Areas) simulation system where the speed of the input generator implemented in form of a delayed Petri Net was regulated by a demon [4] (see Fig. 2). In this system the principles of using demons as a means of distributed AI control have already been fully implemented in an experimental form.

§c A number of experiments have been undertaken to prove the efficient applicability of the method using CASSANDRA 1.4 [1]. They included - beyond parametrical - also structural modifications of the model. As the models have been built in form of various types of Petri Nets in an object oriented way the topology of the network has been modified by the demons according to their inferencing based on the trajectory of the models in the time-state space and the information stored in their knowledge bases.



a)

b)

Figure 2 a) Petri Net model of the communication protocol, b) result of the simulation

Investigations were performed to determine the number of customers that could be assigned to a given market getting products at a given production rate to avoid both deficiency and overproduction.

In an other example the demon applied has become an integral part of the model itself. The problem solved here was the dynamic assignment of a reserve resource to the subsystem where it was desired most. So the relative state of both subsystems (that may have been interpreted as computing or manufacturing systems as well) had to be evaluated. In this experiment two other demons undertaking parametrical modifications in parallel with the structural resource assignment have been used demonstrating the advantage of distributed control. In these cases already secondary parameters as average values of state variables postprocessed by the demons had to be used.

§d We have considered as a highly significant field of application the simulation and design of flexible manufacturing systems [5] which beyond their importance can serve as an excellent testbed for other applications. Here - beyond applying the knowledge based methods to the experimental frame and the static model structure and parameters - another possibility has also been investigated i.e. dealing with the mobile entities in the model.

* The application of KAPNs [2] to describe and monitor the process according to the technological plan as well as other features of the workpieces.

* Controlling their life cycles (number and possible destination in accordance with the state of the model itself).

These have been implemented and investigated in CASSANDRA 3.0 a completely new version of the system and are to be reported on in the next future.

3. CONCLUSIONS

The experimental verification of the methodology proposed has proven its advantageous applicability in various fields encouraging further methodological research as well as their adaptation in many fields.

4. REFERENCES

[1] Jávor, A., Demon Controlled Simulation. Mathematics and Computers in Simulation, 34(1992), 283-296.

[2] Jávor, A., Knowledge Attributed Petri Nets. Systems Analysis, Modelling, Simulation (in publication).

[3] Jávor, A., Rõmer, M., Benkõ, M., Knowledge Base Controlled Simulation. Proc. of the 12th IMACS World Congress on Scientific Computation, July 18-22 1988, Paris, France, Vol.2. 749-751.

[4] Jávor, A., An AI Supported Tools for Simulation in Informatics. Systems Analysis, Modelling, Simulation, 8(1991)4/5, 273-278.

[5] Rozenblit, J.W., Experimental Frame Specification Methodology for Hierarchical Simulation Modeling. Int. J. General Systems, 19(1991), 317-336.

Process Graph and Tool for Performance Analysis of Parallel Processes *

Roman Blasko
Department of Software Technology and Parallel Systems
University of Vienna, Brünner Str. 72, A-1210 Vienna, Austria
tel: +431-392647-226, fax: +431-392647-224
email: roman@par.univie.ac.at

Abstract

We have defined a Process Graph for an abstract representation of parallel programs, which are generated by the supercompiler VFCS. We have developed a technique for automatic generation of the Process Graph model and a tool PROGAN for evaluation of the detailed and global performance characteristics of the parallel models. This technique and tool is integrated in VFCS, but can be used also separately for other types of parallel systems.

1 Introduction

Parallel processing is one of the main techniques how to improve the performance of various technical or production systems. Our global aim is to develop tools for the automatic parallelization of the Fortran programs. The program is parallelized by Vienna Fortran Compilation System (VFCS) [7], which translates Fortran programs into an explicit parallel program with message passing statements. An initial sequential version of the program is written in Fortran'77 or Vienna Fortran programming language [7]. The considered target computer is the distributed memory computer system iPSC/860. The performance analysis for VFCS is supported by the tools as Weight Finder [4], PPPT [5] and PEPSY [2] developed in our team and MEDEA [3] and PARMON [6] developed independently.

The technique and tool described in this paper was developed for the performance prediction to be done before generation of the final code for the target compiler of the parallel machine i.e. before a real run on the machine. The performance results should be used for a comparison of several variants of the parallel program to be possible to find the optimal parallelization strategy made by VFCS. This implies that our focus is to develop the technique suitable for evaluation of relative differences between several program versions and not to concentrate to absolute values e.g. the precise run time values usually measured on the real machine.

There are several abstract description tools (e.g. Petri nets based) for parallel systems, where the common problem is the size of the model by their application for real systems.

*The work described in this paper is being carried out as a part of the research project " Performance Prediction and Expert Adviser for Parallel Programming Environment", funded by the FWF, Vienna, Austria, under the grant number P9205-PHY. The author assumes all responsibility for the contents.

The model based tools use the model created by the user, what is not feasible way in our case. The performance evaluation tools based on the measurements cannot be used before the real run of the parallel program. We tried to avoid these disadantages by the following approach. We have defined the *Process Graph* (PG) (chap.2), for an abstract description of the parallel program. The PG model of the program is generated automatically from its internal representation in VFCS. We have developed PROGAN (chap.3), as a tool for performance analysis of the Process Graph models. This technique is now implemented as a component of VFCS (chap.4).

2 Process Graph for Abstract Representation of Parallel Programs

We have designed the abstract form called *Process Graph* for representation of sequential and parallel Fortran programs on the statement level.

Def.: The **Process Graph (PG)** is defined as a directed graph $PG = (N, A)$, where N is a set of nodes and A is a set of the directed arcs $A \subset N \times N$. The *PG node* is defined by the following set of *node attributes*:

$$n = \{nn, nt, ni, no, dt_1, is, os, ap\}$$

where nn is the node number, nt is the node type, ni is the number of node inputs, no is the number of node outputs, dt_1 is the delay time, is (os) is the set of nodes identified by nn and connected to the node inputs (outputs), i.e. input (output) set and ap represents one or more additional parameters for the specific node type.

We have defined the **set of node types** for the Process Graph with the following semantics:

- The **node type OA** (Or-And, $nt = 1$) with OR input and AND output logicis activated by the control flow comming to any of its inputs and after the specified delay (processing) time, all nodes connected to its outputs are activated in parallel.

- The **node type AA** (And-And, $nt = 2$) with AND input/output logic, usually with two inputs and two outputs, is activated by the control flow present on all its inputs simultaneously and the output signals are sent to all its outputs after the node delay time.

- The **node type OOI** (Or-Or-Iteration, $nt = 3$) has the OR input/output logic in general and one additional parameter $ap = nit$, where nit is the number of iterations. After the initial activation from the second input the first output from the node (to the loop body) is activated. The next activation comes from the first input, the node counts down the number of iterations and after the last one the second node output is activated.

- The **node type OOT** (Or-Or-True, $nt = 4$) has OR input/output logic and the additional parameter for the probability of the first (True) output $ap = P_t$ or the number of cases for the first (second-False) output $ap = N_t(N_f), N_t > 0, (N_f < 0)$.

- The **node type OAD** (Or-And-Delay time, $nt = 5$) with OR input and AND output logic has two possibilities for its delay time. The additional parameter $ap = \{P_t\ (N_t), dt_2\}$ specifies the probability P_t, $P_t \in [0:1]$, for the first variant of the delay time $dt = dt_1$, otherwise $dt = dt_2$, or the number of times N_t when the delay time $dt = dt_1$ consecutively and then $dt = dt_2$.

- The **node type OAAA** (Or-And-And-And, $nt = 6$) has OR input and AND output logic for the "main" inputs and outputs and AND input/output logic for additional (communication) inputs and outputs. After receiving the signal on any of OR inputs, all communication outputs (AND logic) are activated after the delay time for sending messages. Then the node has to wait for all communication inputs (AND logic) and after receiving of all communication inputs (synchronized) and the delay time for receiving the messages all outputs (AND logic) of the node are activated. The node has a set of additional parameters $ap = \{nco, nci, cos, cis, pc, dt_2\}$ where nco (nci) is the number of communication outputs (inputs), cos (cis) is the set of communication outputs (inputs), ps is the probability of communication and dt_2 is the communication time.

3 Performance Analyzer PROGAN

The performance analyzer **PROGAN** (PROcess Graph ANalyzer) is based on the event-driven simulation technique and developed for evaluation of the dynamic behavior of parallel processes represented by the Process Graph. The input of PROGAN is the Process Graph model of the parallel program to be analyzed and the output of PROGAN is a set of performance data characterizing the dynamic behavior of the process graph i.e. modeled parallel program. PROGAN has the following performance analysis facilities implemented by several optional monitoring modes. An **activity diagram** shows all active phases of nodes in the graph activated sequentially or in parallel. A **waiting messages** protocol gives the numbers of waiting messages on inputs of the PG nodes. A **detailed parallelism degree** protocol shows the number of simultaneously activated PG nodes in the graph. All these three protocols are event driven protocols illustrating exactly all events in the graph and suitable also for its debugging. The other monitoring modes are based on a statistical processing of the PG behavior and give a more comprehensive view on the PG performance. Those are based on evaluation of the **utilization** or activity of the j-th node u_j, $u_j \in [0 : 100]\%, j \in [1 : n]$ and the **parallelism degree** p_d, $p_d \in [0 : n]$ in PG, where n is a number of the nodes in PG. Both of these parameters are evaluated for steady and transient states by sampling the values.

Going up for more comprehensive view on PG behavior we have developed the monitoring mode for evaluation of the activity of PG nodes grouped into the specified size of the group. This mode gives the data about the **activity (utilization) of every group** of nodes, its **communication activity** and **waiting for communication** separately. The PG performance is analyzed for the specified time period, for whole graph or for the specified set of the nodes and documented in the numerical or graphical form of results. Additional facilities of PROGAN include an output of the PG structure, connectivity test of PG, complete output of PG specification, evaluation of the PG run time and others.

PROGAN can be used also separately for performance analysis of any PG stored in the input file generated automatically or written by an editor. Then PG behavior may be interpreted for any parallel system modeled by PG.

4 Application for Parallel Fortran Programs

The defined six node types cover all Fortran77 statements after processing by the front-end of VFCS and statements in Message Passing Fortran generated by VFCS. This node set is open with regards to the development of the VFCS and the Vienna Fortran language [7]. The application of the defined PG nodes for the abstract representation of the program statements is the following [1]. The control statements as *DO-loop head, conditional GOTO* and *logical IF* are represented by the node types OOI, OOT and OAD respectively. All other labeled or not labeled statements of Fortran77 are represented by the node type OA. The special statements from the Message Passing Fortran [7] are represented as follows. The masked statement $OWNED$ by the node type OAD, simplified $SEND$ and $RECEIVE$ by OA and AA respectively, and the complex communication statement $EXSR$ (EXchange-Send-Receive) by $OAAA$. The PG model is generated automatically by the PG generator **PROGEN** [2]. This abstract representation of the program is used for the next processing by the performance analyzer **PROGAN**. By the first experience in performance analysis of the concrete PG models generated automatically, we can say that this tool is very effective for practical performance analysis giving both the detailed and global view on the PG behavior.

5 Conclusions

We have defined the Process Graph for the abstract representation of the parallel Fortran programs generated by the supercompiler VFCS. The Process Graph model of the parallel program is generated automatically. We have developed the performance analysis tool PROGAN for the Process Graph models providing the detailed and global performance characteristics of the parallel system to be analyzed. This tool is interconnected with VFCS, but can be used also separately for the analysis of other Process Graph based models.

Our future steps will be devoted to the development of the more compact representation of the parallel program generated automatically and more precise performance analysis during program design and improvement process.

References

[1] Blasko R.: "Parameterization and Abstract Representation of Parallel Fortran Programs for Performance Analysis", Proc. of the AICA'93 Conference, Gallipoli (Italy), Sept. 1993, pp.75-91.

[2] Blasko R.: "Performance Prediction for Parallel Programs Embedded in Supercompiler", Dept. of Software Technology and Parallel Systems, University of Vienna, October 1993, p.20. (to appear).

[3] Calzarossa M., Massari L., Merlo A., Pantano M., Rossaro P.: "Techniques and Tool for the Analysis of Parallel Programs", Rech. Rep. R3/98, Progetto Finalizzato C.N.R. "Sistemi Informatici e Calcolo Parallelo", October 1992, (in Italian).

[4] Fahringer T., Huber C.: "Der Weight Finder. Ein Profiler Fuer sequentielle Fortran77 Programme", Rech. Report, Inst. for Statistics and Computer Science, University of Vienna, Sept. 1992.

[5] Fahringer T., Zima H.P.: "A Static Parameter based Performance Prediction Tool for Parallel Programs", Proc. of the 7-th ACM Int. Conf. on Supercomputing 1993, Tokyo, July 1993.

[6] Lenzi P., Serazzi G.: "PARMON: Parallel Monitor", Tech. Rep. R3/95, Progetto Finalizzato C.N.R. "Sistemi Informatici e Calcolo Parallelo", October 1992, (in Italian).

[7] Zima H.P. et al.: "Vienna Fortran - A Language Specification, Version 1.1", NASA Contract Report. ICASE Langley Research Center, Hampton, Virginia, March 1992, 84p.

A DISTRIBUTED OPEN SYSTEM ENVIRONMENT FOR A REAL-TIME SIMULATORS

L.Castiglioni, M.De Chirico - S.d.I.Automazione Ind.-via Winckelmann,1 Milano, ITALY

F.Pretolani-ENEL C.R.A.- via Volta,1 - Cologno , Milano, ITALY

Abstract. This paper presents the problems tackled and the solutions adopted in creating a distributed scheduling process for the construction of real-time simulators. This product is used by ENEL for training and engineering simulators within the context of industrial processes for the production of electric energy. The scheduling process is for co-ordinating of a certain number of processes, each of which deals with the simulation of the different subsystem of a power plant constituting a simulator

1. Introduction

Within the LEGOCAD [1] system, simulators are constructed in a first phase grouping together pre-built library modules to create a modelling task describing a single plant subsystem. After developing and optimising the various modelling tasks, they have to be interconnected in order to optimise models further and build the simulator. A scheduling process is then required to manage and synchronise the individual modelling tasks and their mutual interactions.

The functions performed and the user services provided by this process are the following:

functions:

- activation of the modelling tasks forming the simulator;
- synchronised exchange of the variables of interconnection between the various modelling tasks;
- maintaining real time (where the calculation resource allows it).

user services:

- periodical filing on disk of a certain (configurable) number of measurements of special interest;
- saving and loading of snapshots of the simulator data base;
- maintaining an updated and coherent data base of the simulator during transients;
- management of a queue for receiving disturbances on the free inputs of the simulator;
- management of a queue for receiving simulation commands and information on the status of the simulator.

As listed, synchronisation of the various modelling tasks and management of the variables of interfacing between the different parts of the plant which they model are handled by the scheduling process. In the case of large simulators or those with highly sophisticated calculations, it may be useful to be able to distribute the modelling tasks over several computers connected by a network so as to exploit fully the calculation parallelism. Synchronisation of the tasks on different machines and the correct exchange of interface variables is performed by the scheduling processes present on each machine. For this purpose a distributed scheduling process has been produced which, in addition to the functions and services mentioned previously, allows modelling tasks to be allocated to several, possibly dissimilar, machines.

In the following sections the needs which led to the creation of a distributed scheduling process, which enables the various calculation processes to be allocated to different machines, are underlined.

The guidelines followed when developing the project are also given, above all the choice of the hardware and software platforms supported by the system. The scheduling algorithm is described in detail. Finally reference is made to the building of an engineering simulator for training purposes (simulator of a 320 MW unit at the Priolo power plant) in order to check the effectiveness of the project performed.

2. The distributed scheduling process

The reasons which, in certain conditions, led to the need to distribute the simulation environment over several machines connected in a network are now discussed.

This need normally arises with real-time simulators, a term which indicates occurrence of the following circumstances:

- the simulated system evolves over the same periods of time as the real system, even in special load conditions of the system;
- the operator can interact with the simulator using the services of actuation and/or command from external units, achieving reaction times which are comparable to those of the real plant.

The scheduling algorithm deals with activating the tasks modelling a process at specific frequencies which differ for each task (we refer to the interval of activation of the individual task as the integration time step). So that the evolution of the system occurs in real time, each task must take for calculation a period of time which is equal to or shorter than the integration time assigned to it. Normally the tasks completes the calculation phase within the time limit. The scheduler, after having managed the exchanges of variables between the tasks, waits for the set time to elapse before re-activating them.

For the regulation tasks the time taken for calculation can be determined beforehand and is independent of the type of transient underway. The process tasks, using an iterative algorithm for the calculation, will however require calculation times proportional to the number of iterations necessary for an integration step.

The iterative tasks are therefore responsible for introducing a certain degree of non-determination into the necessary calculation time and in extreme cases for causing deceleration in simulation. If a task does not complete the calculation phase in good time, the other tasks also have to wait before being able to exchange interface variables, introducing a delay into the simulation as a whole (it is in fact the actual introduction of the delay, linked to synchronisation between the tasks, which ensures the numerical accuracy of the simulation). Often the delay condition only occurs in particular phases of simulation in which some tasks are forced to iterate repeatedly. In a later phase the tasks will probably return to a less critical working condition; in this case the scheduler benefits from this situation by making up the accumulated delay until it relinks with the clock which marks real time.

The second condition introduced into the definition of real- time simulator means that the time taken for a command (sent by the console or by an MMI service) to cause a reaction by the simulated system comes within times of the order of 400 msec. Sending a command causes a variation in a simulation variable which, being part of a remote control or regulation task, on the basis of the coded logic in the remote control itself, produces a variation in another variable which acts as feedback for the operator. The time taken to perform the process described is directly linked to the time of integration of the remote control task to which the relevant variables belong (in the worst hypothesis the integration time will contribute in the total calculation for a time equal to double its value). In order to achieve acceptable interaction times, the integration times for these tasks will normally be short (200 msec) and shorter than those taken for iterative tasks (typically 1200 msec).

The possible deterioration in performances in terms of operator-simulator interaction time is however almost always caused by situations in which the iterative tasks accumulate a delay. In this case the regulation tasks are also slowed down, leading to deceleration in the responses of the system which is apparent to the operator.

A further CPU load source for the system is the MMI services which, for interfacing with the user, adopt advanced graphics environments which, if the system consists of a single workstation, could bring about mutual disturbance situations.

The solution identified therefore is to create real processing parallelism, using a system composed of several workstations interconnected in a network. This solution enables part of the calculation potential to be reserved for absorbing load peaks which occur in the simulation of particularly fast transients which give rise to a high number of iterations of the process tasks. During normal transients the excess calculation power can be used for developing simulation in fast-time mode, i.e. in a shorter space of time than those of the real plant.

3. Project guidelines

One of the essential HW and SW requirements of the simulation environment is the high portability of the end product. The simulation environment is currently available on UNIX (ULTRIX, AIX, SCO-UNIX), VMS, HELIOS platforms.

This restraint has heavily influenced the decisions made when defining the project, leading to the exclusion of those technologies, both HW and SW, which are excessively restrictive in terms of portability.

Attention has been turned to operating system which are widely used in the various applications, also present on HW platforms which differ in terms of calculation power supplied or graphical representation capability. UNIX, by now widely used as a standard both for workstations and personal computers, fulfils these requirements. Furthermore the ease of integration via the network between HW platforms which support standards such as TCP/IP [3] is an additional advantage.

These considerations indicate that, by using simple IPC mechanisms under UNIX, migration under these other operating systems is not particularly complex (obviously the situation changes if advantage is to be gained from special mechanisms only existing in particular working environments).

In order to facilitate the portability of the SW, the ANSI C programming language was chosen. Moreover efforts were made to maintain the unique nature of the source code also for the later extensions made to make the simulation environment also available in VMS and HELIOS. Therefore on the one hand differentiated coding of basic libraries was used, and on the other conditional compiling. Modifications were limited to where the different operating systems required the use of different system services, maintaining general software structure uniformity and above all conformity in the use of SW interfaces between the different services.

For VMS the structure of the simulator is identical to the UNIX case, whereas modifications were made for the transputers to exploit the parallelism potential where several processors co-operate in performing calculation. Having maintained the same SW structure, and above all the same interfaces of dialog and interfacing, a distributed simulation environment can also be created by using heterogeneous machines.

As regards the network protocol to be used, TCP/IP[3] was chosen, a protocol which on the one hand is emerging as a standard and which on the other has been available on the target platforms for the project for some time.

4. Scheduling algorithm

The processes working together for simulation are allocated to different workstations and dialog with each other via the network. Approximately 90% of the network traffic consists of data exchange and the remaining part of commands for driving simulation.

Since numerical accuracy of the simulation is an essential characteristic, the protocol to be used for the exchange of data must be reliable.

From the range of TCP/IP protocols a choice had to be made between TCP (stream mode) and UDP (datagram mode). The TCP mode provides a reliable protocol while the UDP mode requires the addition of a suitable acknowledge mechanism.

Tests were performed to assess, in a situation similar to the real one, the performances of the TCP protocol compared to those of the UDP protocol with the addition of a suitable acknowledge mechanism. The network traffic was estimated at approximately 600 bytes every 100 ms per task[1].

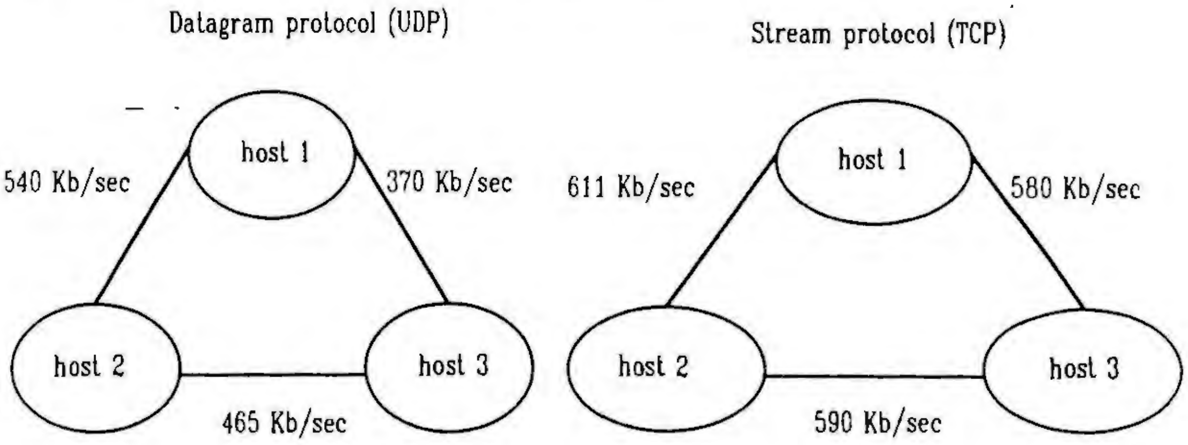


Fig. 1 - Result obtained with UDP and TCP protocol

Fig. 1 gives the results obtained using 3 DEC-Station 500/200's and with the hypothesis of having 10 tasks resident on each workstation (host number 3 is not on the same physical network). Given the results of the tests it was decided to use the stream mode which, although involving greater complexity due to the initial work of setting up a connection between the various hosts, has higher performance levels than the UDP, managing the acknowledge mechanisms efficiently inside the TCP protocol itself.

Having defined the protocol of communication between the workstations, attempts were made to identify the architecture which best combines features of simplicity and linearity with the performances within the limits of resource consumption and fast time. The choice was between a centralised architecture and a distributed one.

In the first solution one of the workstations has to be elected which, while continuing to contribute to the numerical integration of simulator tasks, also performs the role of MASTER. This means that all the data of the simulator are sent by the other machines (SLAVE) to the MASTER machine, in this way centralising the whole dynamic simulation data base. The MASTER machine has to deal with management of the variables of interface between the various tasks, sending their updated value to the SLAVE machines.

In the second solution, all the machines working together for simulation are at the same level: each individual workstation sends the data directly to the workstation which requires them.

In the attempt to find an architecture which is the simplest possible without jeopardising the system performances too much, a hybrid one was chosen (Fig. 2): management of the exchange of data between the individual tasks is distributed, while performance of services (simulator data base management, snapshots, continuous recordings etc.) is centralised. Centralised management of services, in the case of the simulator data base, also allows greater ease of interfacing with external client processes.

In the solution chosen, one of the machines is referred to as MASTER and has the task of managing all the simulation services as well as activating and stopping simulation, while as far as exchange of data between the tasks during normal operations is concerned, all the machines are at the same level.

Thus an architecture was produced which, during normal operation, focuses on performances, decentralises communications and simplifies the functioning of services by centralising them on the MASTER machine.

[1] This value was obtained by estimating that on average the exchange of data between two tasks is approximately 150 data at activation, then considering the dimension of 4 bytes per each item of data (float) and since 100 ms is the minimum activation time of a task, the data shown is obtained.

Each simulation step [2] depends on the individual scheduling algorithms: the MASTER machine, where the MASTER scheduling process is resident, is the machine dedicated to receiving commands [4] from the outside. Having receive the simulation start command, it sends it to the SLAVE machines involved in simulation, according to the centralised model discussed previously.

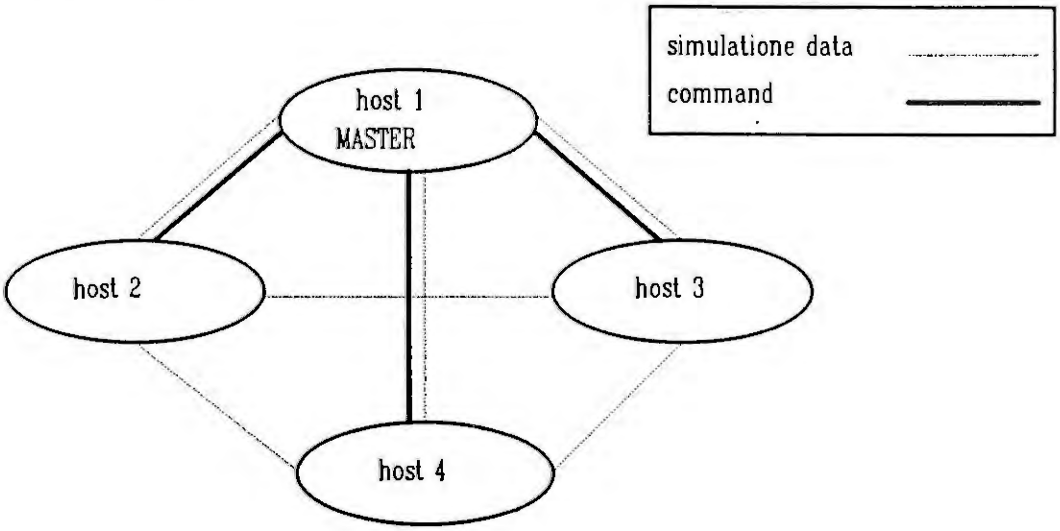


Fig. 2 - distribuited architecture

Later the distributed management phase begins in which each machine dialogues, at the level of data exchange, with any other machine according to needs. The destination and origin of each item of data is defined in special tables resident on each machine and containing the following information:

a. times of activation of each resident task (e.g. integration step 0.4 seconds, simulation time step 1.2 seconds => activation times 0.0 - 0.4 - 0.8);
b. the list of output data for each task with the relative destination tasks;

c. the list of input data for each task with the relative origin tasks

On the basis of these tables the scheduling processes, both MASTER and SLAVE, at the start of the simulation step, activate all the resident tasks and then await the conclusion of the calculation of the tasks with an integration step of 100 ms. The latter, having completed the calculation phase, send a calculation end message to the scheduler on which they depend. The scheduler, on the basis of table b), will send the output variables to the appropriate machines, then will hang up to await the arrival, on the basis of table c), of all the data coming from the other machines.

If the time taken to perform these operations is less than 100 ms, the task will hang up for a time equal to the difference between 100 ms and the time taken, otherwise the delay will be stored and recovered later if possible, avoiding hang-ups.

Scheduling proceeds in a similar manner on each machine, with steps of 100 ms, up to the end of the simulation step which marks the return to centralised management: each SLAVE machine sends all the simulation data relating to the resident tasks, forming a kind of local data base, to the MASTER machine, on which the global data base relating to all the tasks of the simulator is resident, and then hangs up to await a new step start message.

The MASTER scheduler, after having carried out the periodical recordings on files, checks that other simulation commands are not present, then, in the absence of other requests, sends the step start

[2] Simulation time step is defined as the minimum common multiple between all the individual integration steps of the tasks forming the simulator.

message to the SLAVE machines, launching a new simulation step.

5. Calculation distribution strategies

For the simulator to function properly, the various modelling tasks have to be allocated, with a certain amount of care, to the machines making up the calculation engine. First of all the features have to be indicated (in addition to accuracy of calculation) on the basis of which different possible solutions are to be compared:

- maintaining of real time in the various operating conditions of the simulated system;
- achieving the maximum fast-time level in relation to real time.

Due to the complexity of the problem and the large number of cases and hence of possible solutions, it is impossible to trace a general theory for distribution of the calculation load, therefore only some guidelines will be set out.

First of all it is to do analysed the case in which the machines set of the calculation engine is homogeneous (in respect of HW and SW), then additional considerations will be made in the case of a heterogeneous HW/SW environment.

The first operation to be performed is that of checking, in quantity terms, the consumption of each modelling task which makes up the simulator, bearing in mind that, unlike the regulation and automation tasks which require a constant level of calculation resource, the process tasks, using an iterative calculation algorithm, have consumption which varies according to the operating conditions of the simulated plant.

These variations in calculation time can also be appreciable, even reaching peaks of over 100%. Generally however it is fairly unlikely that all the process tasks of a simulator encounter this problem simultaneously. This means that real time can be achieved even without excessively overdimensioning the calculation system and advises the choice of a structure composed of many small tasks rather than a few large ones. An objection may be made to this by saying that breaking down into many tasks may cause problems of numerical stability of the integration algorithm, which ensures stability within each task, but which cannot offer the same guarantees for an aggregation of tasks. This clearly indicates that breaking down the process to be simulated into tasks is an operation of some importance which has to make a compromise between various needs.

As regards distribution of tasks on the calculation machines, those configurations which use less network resources have to be identified from the various ones possible. In this way a large share of machine resources is available to be dedicated to calculation and an increase in the upper fast time limit is achieved. This limit in fact depends mostly on saturation of network resources.

Generally, in order to achieve adequate performances, the subdivision of the plant (process, automation, regulation) merely has to be copied for the machines making up the simulator. For a quantitative analysis it must be considered that normally the quantity of data exchanged between individual tasks does not pose a problem while, during a simulation step, the number of packages of data exchanged has to be restricted as much as possible.

It is unlikely that a single solution can be found to the problem and in any case a certain number of tests will have to be carried out.

Having decided on the distribution of the tasks, one of the calculation machines has to be elected as simulation MASTER. Generally, since the MASTER machine has to perform some additional services, it is advantageous to choose the less loaded machine in terms of calculation, without however forgetting that at the end of each simulation step the SLAVE machines transfer via the network the local data base to the global one resident on the MASTER machine, hence from the network standpoint the optimum solution is that of using as Master the machine where the tasks relating to the larger local data base are resident.

Since for communication between machines involved in simulation, use was made of messages with a format independent of the HW platform and the operating system used, the possibility remains of using a heterogeneous group of machines. In this case, when determining the distribution of tasks on the available platforms, the following hints must be taken into consideration:

- calculate the load required of each machine as a percentage of the maximum available, in that the machines in general will have different calculation capacities;
- restrict the load due to the possible conversions of format for the data, necessary for example in the case of the joint presence of UNIX and VMS machines (e.g. G- FLOAT VMS format with IEEE format).

6. Results

To date, the most complete simulator achieved (see Fig. 3) is the one of a 320 MW unit at the Priolo power plant, used by the ENEL operator training school in Piacenza [5].

As shown in the illustration the simulator can be divided schematically into three sections: calculation engine, instructor's desk, operator console.

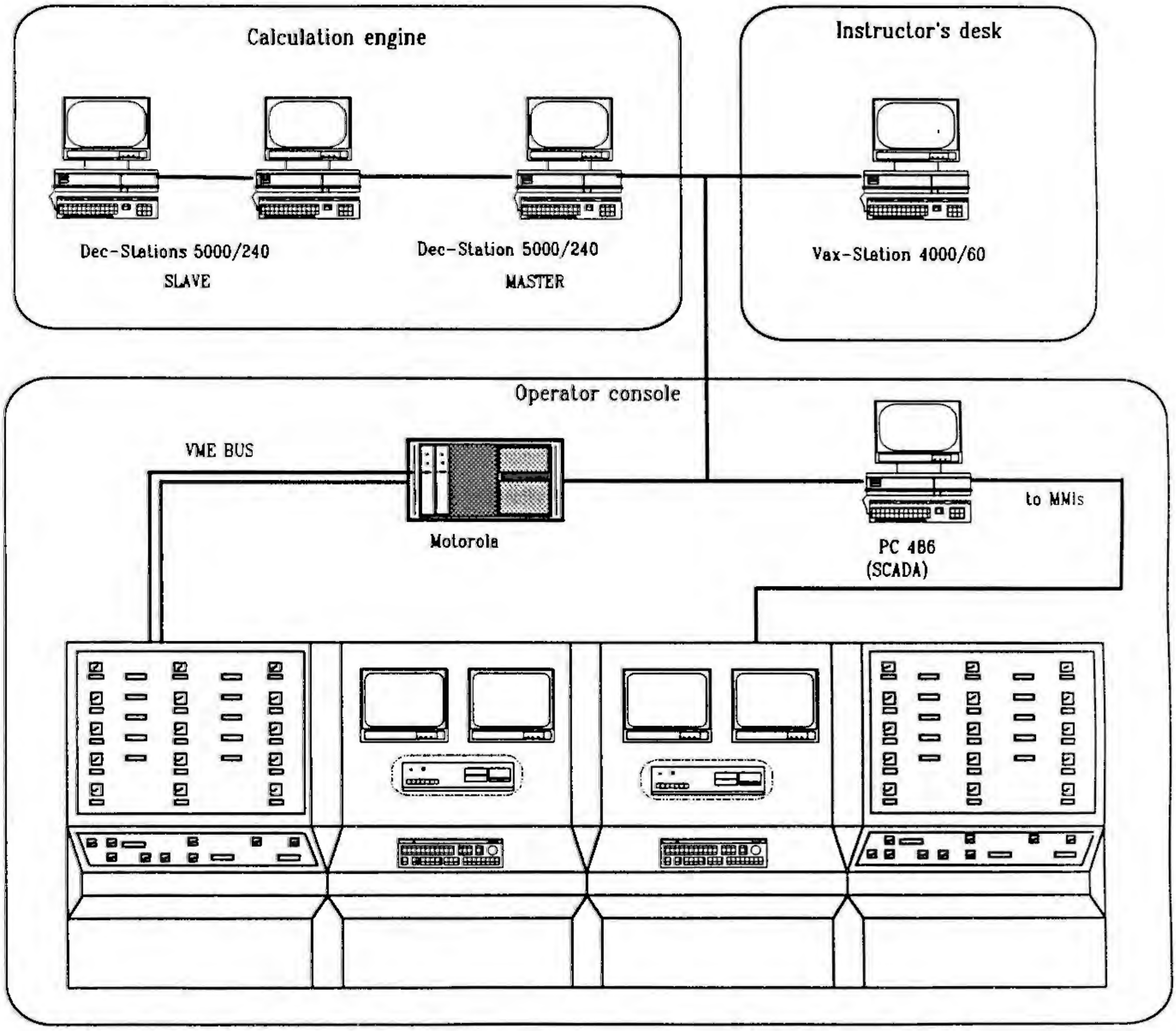


Fig. 3 - Priolo power plant training simulator architecture

The calculation engine has been built by distributing the calculation load between three DECStation 5000/240's. In distributing the modelling tasks over the three workstations, we referred back, as far as possible, to the real subdivision present within the plant: process, regulation, automation. This subdivision, restricting most interactions among the models inside the same machine, enabled the network traffic between the three workstations forming the calculation engine to be reduced to a minimum.

The use of three machines, operating together, means that the three workstations always maintain a good calculation reserve. The presence of this reserve enables, even during the most critical transients, from the calculation point of view, real time is to be maintained [5], and also enables the use of fast time during the transients of less importance from the training standpoint. The following tables give the CPU occupancies recorded in a test aiming at establishing the load of the workstations (in normal conditions) during normal and fast-time operation. For the instructor's desk a VAXStation 4000/60 was used.

real time	host 1	host 2	host 3
% user	40%	38%	39%
% system	8%	4%	17%
% idle	52%	58%	44%

fast-time (1.6)	host 1	host 2	host 3
% user	48%	66%	51%
% system	14%	20%	7%
% iddle	38%	14%	42%

The operator console contains two subsystems seen as clients by the simulator: the S.d.I. supervision system (SCADA on PC-486) and the Motorola board (MVME 147 SA-1) which handles the I/O cards of the operator console. For the first of the two clients, as mentioned previously, centralised interfacing was used. From the tests performed this interfacing was found to have a minimal effect both on the MASTER workstation and on the network load.

For interfacing with the Motorola board, given the realism requirements, distributed management had to be adopted with obvious increases in consumption of the calculation resources of the workstations forming the calculation engine. On average 15% of the resources of each workstation was consumed. In order to assess the response times in quantity terms, a modelling task with a step lasting 100 ms was used and the space of time between pressing a pushbutton and later lighting of a led was measured: the average time obtained was approximately 200 ms, considered satisfactory for realistic operation of the console.

Bibliography

[1] R.CORI, S.SPELTA "Il sistema LEGO dedicato alla modellistica assistita da calcolatore per la simulazione di processi complessi" Convegno annuale ANIPLA, Bari 1988.

[2] Rebecca Thomas, Lawrence R. Rogers, Jean L. Yates "Advance Programmer's Guide to UNIX SYSTEM V", McGraw-Hill, Inc. 1986

[3] Douglas Comer "Internetworking with TCP/IP - Principles, protocols, and architecture", Prentice-Hall, Inc. 1988.

[4] G.DEMICHELI, L.GIORGIUTTI, V.VANNELLI, G.B.GARBOSSA, F.PRETOLANI, S.SPELTA "Linee progettuali di un simulatore di impianto di produzione termoelettrica da 320 MW con caldaia a circolazione assistita - Automazione 1992, 36th Annual Conference - Novembre 1992, Genova.

[5] G.Garbossa, M.Favaretto, L.Castiglioni, M.De Chirico "NUOVE ARCHITETTURE MODULARI HARDWARE E SOFTWARE PER I SIMULATORI DI ADDESTRAMENTO", Convegno internazionale BIAS 1993 "Automazione 1993"

Object-Oriented Modelling and Simulation of Batch Plants

Konrad Wöllhaf and Sebastian Engell
Lehrstuhl für Anlagensteuerungstechnik
Fachbereich Chemietechnik
Universität Dortmund
D-44221 Dortmund
Germany

Abstract

In this paper, the modelling and simulation of recipe driven multipurpose chemical or biochemical plants using an object oriented approach is described. A data model is presented which enables the description of all important aspects of batch plants: the model of the plant (reactors, tanks, pipes, separation units etc.), the batches of material and the dynamics of their transformation, and the recipe driven sequential production process. Batch plants are complex hybrid systems (Clark and Joglekar, 1992) because many different components with associated continuous dynamics are involved, the production sequence is event-driven, continuous and discrete state variables must be used to describe the production process, and the structure of the system and the number of state variables change over time.

1. INTRODUCTION

Multipurpose batch plants are used for the economic production of chemical products in small quantities. Different products can be made at the same plant, eventually even in parallel at the same time. Flexible batch production, formerly mainly used in certain specialised branches of the chemical industry as e.g. the production of pharmaceutics and paints, is extended to other products like polymers and replaces continuous production processes as production to order becomes more and more important.

A multipurpose batch plant consists of different vessels (tanks and reactors), auxiliary equipment and pipes to transport the products. Recipes define the production process of each product, i.e. which substances have to be put together in which quantities and under which conditions. They also contain instructions for the operator or the process control system on temperature and pressure values or curves, variables to be monitored, stopping conditions, etc. In recent years, rapid progress in the automation of batch processes has been made such that in newly equipped plants the task of the operator is not any more the proper execution of the recipes, but the choice of the production sequences and the scheduling of the equipment. For this task, it is not sufficient to take the availability of raw materials and other resources and the production orders into account, but the complete state of the batches of substances and the equipment and their future development must be considered.

Purely discrete models of the production process as e.g. Petri nets are not sufficient to model such plants because the thermal and chemical processes have continuous, often variable, dynamics, which must be taken into account to predict the development of the plant, e.g. for scheduling purposes. The aim of our work is the development of a tool for the simulation of the production processes in a recipe-driven multipurpose batch plant including all important aspects of such plants, in particular the continuous and the discrete dynamics, in order to support the supervision of the plant and the scheduling of production tasks.

The first step towards simulation is the development of a formal model of the system which is to be simulated. In our case, models of the production plant (equipment with basic control units), of the recipes and of the batches of material are developed in a form which allows the free combination of elements of each class. So these are the atoms of the models which can be combined flexibly at run time to describe the actual processes in the plant which depend on the scheduling decisions.

2. ELEMENTS OF THE MODEL

The representation of the production processes in a batch plant is based on the concept of recipe controlled operations, which has been developed by the major chemical companies in Germany (Uhlig 1987, NAMUR

1992) and is used in the automation of batch plants by most suppliers of process control systems. The structure of the model consists of the description of the multipurpose plant, the basic recipes, the production schedule, and the executable control recipes. In this paper, mainly the plant model and the control recipcs are discussed.

The plant model consists of all main elements of the plant, the auxiliary equipment, the connections to exchange energy and material, and the control components. The plant model must be detailed enough to generate information about the duration of the production processes and quality parameters of the products. Therefore the states of the batches of matcrial such as their pressurcs, temperatures, and compositions must be included. The state variables of the batches arc changed by the unit operations. They can either be operations enforced by process control (or manually) such as heating, cooling, mixing, discharging, or chemical or physical processes such as chcmical reactions, condensation, solution or crystallisation of a component. The enforced operations are called technical functions. The technical functions are implementations of the basic operations used in the recipe to describe the production steps. In fact, the two types of operations can be treated in the same way, so the difference is one of semantics.

A basic recipe is a sequence of instructions for the production of a batch of product of standard size. It describes, without reference to a particular plant or particular equipment, the sequential or parallel production steps. The sequence of the production steps is controlled by events which either arc created by clocks, the time events, or by continuous process variables reaching predefined thresholds, the state events. Sequential function charts are used to represent the recipes in a formal manner. In the actual production process, the basic recipe is specialised to the control recipe which makes reference to specific equipment assigned to the particular batch of a certain size which can differ from the standard size. The basic operations of the basic recipe are replaced by technical functions of the real plant, and the process variables which occur in the conditions for the transitions, are identified with measurement devices. The control recipe activates resp. deactivates the unit operations as triggered by the evolution of the state of the plant. Thus the mathematical model of the continuous processes in the plant may change with every stan of a new production step and must be generated during a production resp. simulation run depending on the actual dynamics and decisions.

3. THE OBJECT ORIENTED DATA STRUCTURE

In this section, the object oriented data structure of the most important parts of the model is described. We give no general description of the object oriented approach because there is ample literature about object oriented design and programming languages, e.g. (Martin and Odell, 1992).

3.1 The class structure

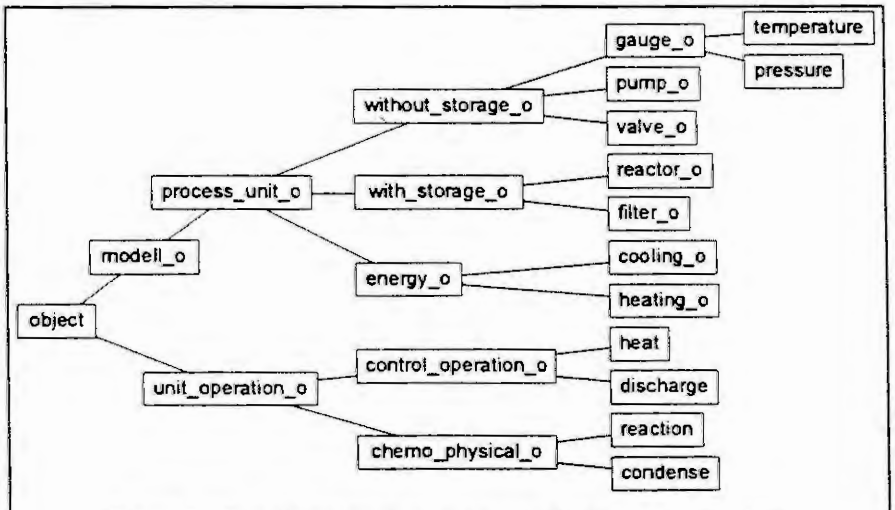


Figure 1: class inheritance of process units and unit operations

The inheritance of data and functions is a powerful element of the object oriented approach. It allows the design of basic elements and the stepwise refinement of the model objects. Similar to the manner how natural objects are classified by types and supertypes, e. g. plant, tree, broad-leaved tree, apple-tree, the class structure for multipurpose batch plants can be defined. The definition of such an inheritance structure depends on the point of view on the system, i.e. which properties are important, which properties are common to several objects, which features should they have. In figure 1, the class structures of process units and of unit operations are shown.

The class "object" is used as a common list object. All elements of the models are kept in lists. This allows for the management of an arbitrary number of components, which is only limited by the memory size. The basic functions for list handling are defined in container classes. A container class defines objects that in turn contain other objects, such as linked lists and binary trees. The objects in the class "modell_o" are nodes of a graph, connected by flow elements of different types. Functions to define and to save model parameters are provided. The class "process_unit_o" defines general process units. All process units are derived from this class. The next level of inheritance defines the classes "without_storage_o", process units which cannot store material, "with_storage_o" which provides properties to store material, and the class "energy_o" for which the basic features of sources (or sinks) of energy are defined. The classes on the right are non-abstract classes, only for these objects memory can be allocated.

The class "unit_operation_o" is the superclass of the classes "control_operation_o" for control operations and "chemo_physical_o" for chemical or physical unit operations, which cannot be controlled directly by external inputs. The class "unit_operation_o" provides a virtual function to evaluate the equations of the unit operations. The implementation of this function is realized in the non-abstract classes, e.g. "heating" or "reaction". The definition of virtual functions enables the definition of basic functions in a program, without knowing on this level, how they are realized. But somewhere in the derived classes, the function must of course be defined completely. One advantage of this mechanism is that new models can easily be added to a model library.

3.2 Relations between objects

The hierarchical data structure allows the definition of classes which have specific properties on the different levels. Object are instances of the classes. They are used to construct the actual data model by defining the relations between the model objects. Model objects may contain a list of other objects or may have special relations to other objects. As the mathematical models change dynamically due to the structural changes in the system, the relations between the model objects must be dynamic as well. A multipurpose batch plant consists of many process units and their connections. The relation of the objects can be defined on different levels of the data structure. In complex systems there are many relations between the objects, therefore they are described from different viewpoints. A way to represent the relations between objects are entity relationship models.

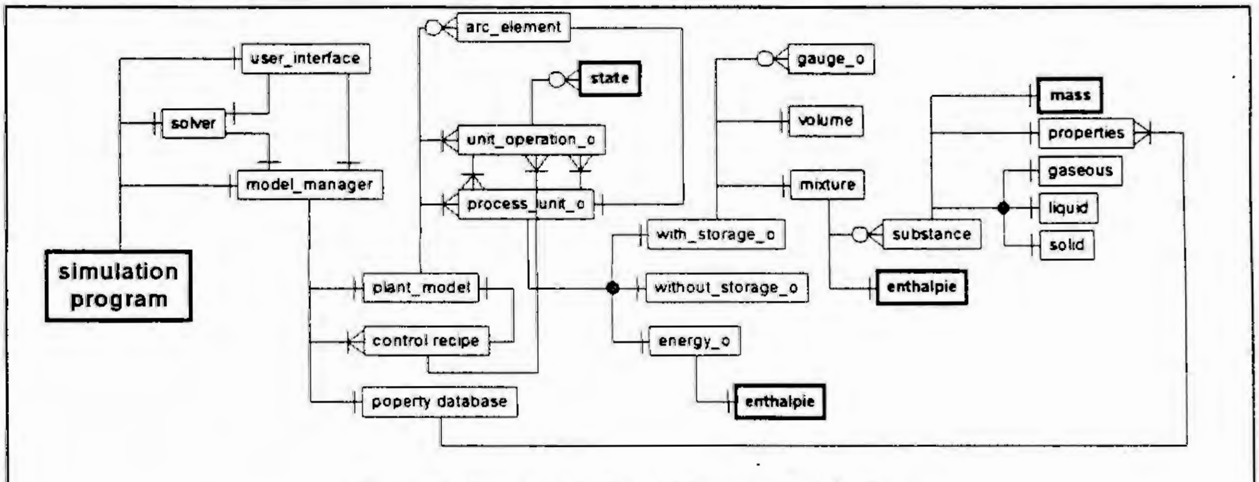


Figure 2: data-model of a multipurpose batch plant

Figure 2 shows a part of the data structure of the model. Here the Crows Foot Notation (Martin and Odell, 1992) is used. The plant model, one or several control recipes and the properties database are contained in the object "model_manager". The plant model consists of one or several unit operations, process units and connecting elements. The unit operations may define state variables which appear in a differential equation. A process unit is an object with storage or without storage, or an energy object. A process unit with storage contains a mixture of substances, has a specified volume, and can have several measurement transducers. The mixture is characterized by its enthalpy and a list of substances. Each substance is characterized by its mass and its state of aggregation and has access to its physical and chemical properties stored in a database. The state variables of a mixture, its enthalpy and the masses of every substance can be changed by the unit operations.

4. THE MODEL EQUATIONS

The change of the state of the plant is caused by the active unit operations. The differential equations are formulated by explicit equations such that every active unit operation gives an additive input to the derivatives of the state variables. Every object in figure 2 that contains a state variable (bold letters) also includes its derivative which is the left-hand side of an explicit differential equation. The use of implicit differential equations is avoided because the numerical solution is computationally expensive and DAE's with high index can be created. If the solution of implicit equations is necessary this is done locally in every process unit by a special solver object. This approach can also be generalized to systems which have to be described by DAE's.

This representation allows for a free combination of active unit operations. It is particularly useful to describe the dynamics of multipurpose batch plants, because different unit operations can be active simultaneously.

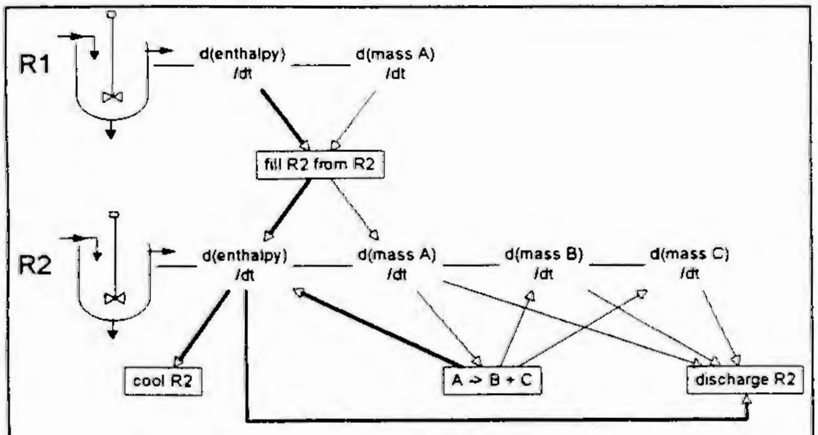


Figure 3: The interaction of different unit operations and process units

As an example, figure 3 shows two reactors, four unit operations and the derivatives of the state variables enthalpy and mass. The unit operations "fill R2 from R1", "cool R2", "chemical decomposition of A to B and C" and "discharge R2" are active at the same time. Reactor R1 contains substance A, reactor R2 the substances A, B, and C. While the unit operation "fill R2 from R1" is active, d(mass A)/dt and d(enthalpy)/dt in R1 are negative and the values are added to the derivatives of the state variables in reactor R2. The process units themselves are passive components. After all unit operations are executed in a simulation step, the derivatives are returned to the ODE solver. Only the solver changes the state variables.

5. SUMMARY

This paper describes a modular mathematical model of multipurpose batch plants using the object oriented approach. The basic data structure of the model which consists of the class structure, the relations between the objects, and the way the model equations are formulated was presented. The resulting model describes a hybrid system that contains the continuous dynamics of the processes and the discrete dynamics of the production control (Cellier et al, 1993). The data structure is used in a prototype of a simulation system (Engell and Wöllhaf, 1993) that contains the data structure described here, implementations of basic models, algorithms for the solution of differential and algebraic equations, a database for physical and chemical properties of the most common substances, a graphical user interface to define the models, and an interface to other programs to evaluate and visualise the results of the simulation. Mechanisms to support automatic step size control, the detection of state events, the switching between models of different structure, and the handling of the global state vector are taken into account in the prototype of the program.

6. ACKNOWLEDGEMENTS

The research was supported by the Ministerium für Wissenschaft und Forschung of the Province of Nordrhein-Westfalen. This support is very gratefully acknowledged.

7. REFERENCES

[1] Cellier F. E., H. Elmqvist, M. Otter, J.H. Taylor: Guidelines for modeling and simulation of hybrid systems, Proceedings of 12th IFAC World Congress, vol. 8, page 391-397, Sydney Australia, 1993.

[2] Clark, S., and Joglekar, G. S.: Features of Discrete Event Simulation. NATO Advanced Study Institute, Antalya, Turkey, May 1992.

[3] Engell S. K. Wöllhaf. Dynamic simulation of batch plants, Proc. European Symposium on Computer Aided Process Engineering-3 (ESCAPE-3), Graz, 439-444, 1993.

[4] Martin J., J. J. Odell: Object-Oriented Analysis and Design, Prentice-Hall, Englewood Cliffs, 1992.

[5] NAMUR-Empfehlung: Anforderungen an Systeme zur Rezeptfahrweise (Requirements for Batch Control Systems). NAMUR AK 2.3 Funktionen der Betriebs- und Produktionsleitebene, 1992.

[6] Uhlig, R. J. Erstellen von Ablaufsteuerungen für Chargenprozesse mit wechselnden Rezepturen (Development of sequential controls for batch processes with variable recipes). Automatisierungstechnische Praxis 29, 17-23, 1987.

A NEW APPROACH TO DATABASE MODELLING

Rudolf Keil
Joanneum Research
A-8010 Graz Steyrergasse 17

Abstract: In this paper a database model is defined to be the structure of some real world model. The real world model is a model in the sense of natural science that means it consists of a set of hypothetical assumptions simplifying some real world area to enable a representation of the information about this area in a database system. Such an approach to database modelling makes it possible to give a mathematical treatment of database models and to give mathematical definitions of database terms.

1. INTRODUCTION

A database model is usually considered to be something like a user interface of a database system or a database management system respectively. For example, this spirit is expressed in the definition given in [5], where a database model is described in the following way ' A database model is used as the overall framework for describing how data and manipulations of it are to be presented to database users'.

This function of a database model is certainly the most evident one, but there is another aspect of a database model which can be called the real world aspect. This real world aspect was recognized with the coming up of the semantic database models. The spirit of these semantic database models is best reflected in a statement given in [4]: 'Every database is a model of some real world system.'. In recent days, this spirit has a rennaissance in the object-oriented paradigm, this can be seen e. g. in [7], where it is stated: 'A data model is a logical organization of the real world objects, constraints of them, and relationships among objects'

Both of these database aspects are valuable and have their own right. This means that a database model has a bridge position bridging the real world and the computer world (world of data).

The development of a database model can thus start from each of these two sides. Usually, the development of a database model starts from the computer side. The hierarchical model, the network model, and the relational model are examples for such a procedure.

In contrast to these approaches, I want to present an approach from the real world side. The starting point of my approach is a real world model which is described by a set of hypotheses and which acts as a framework for ordering the knowledge about some real world area. Such a model is comparable with a model used in natural science, e. g. the model of an ideal liquid, where simplifying assumptions are made to enable a mathematical treatment. Natural science usually has to deal with dynamic systems. Therefore, the adequate mathematical description is given by differential equations. In our case, we have to deal with a static structure. Therefore we proceed to give a formulation in a first order language where the hypotheses of the real world model are represented by a set of first order formulas, describing a mathematical structure which is a model (in the sense of mathematical logic) of these formulas (axioms).

This structure, which is a topological structure, can then act as a database model. This means it can be used as a framework for logical interfaces to database systems. According to this theory, the general rule for developing a database model is given by the following steps:

(I) Formulate the real world model by a set of hypotheses.

(II) Represent these hypotheses by a set of well formed formulas in a first order language.

(III) Give a set theoretic discussion of the structure given by these formulas.

(IV) Find those structural properties which are important for using the structure as a database model and derive set theoretic definitions of those terms commonly used in database modelling.

The merits of such a procedure are that it is a straightforward procedure, that it yields consistent and sharp definitions of database terms, and that it offers a sound theoretical base for database modelling.
Usually, the terms of database modelling are introduced as socalled basic concepts in a rather ad hoc fashion. They are defined by verbal descriptions and explained by examples. This is one of the reasons why a comparison and a systematic treatment of database models have not been possible yeL
Now we are able to derive these terms from an overall structure und to define them in a mathematical notation as special structural properties. In the following chapter we give an example for such a procedure:

2. THE PROPERTY-ORIENTED MODEL: A PARADIGM

As an example for a general rule (i. e. a paradigm) for developing a database model, we present a special real world model which I call the property oriented model [6]. It was designed to reveal the basic assumptions underlying the 'flat' database models, including the relational model.
In this model, the real world model is taken to be given by the following hypotheses:

1. There exists a set of **property values** which enable us to describe a certain real world area, e.g. {(colour red), (length 1 m) .....}.
2. Each such property value specifies exactly one **property (type)**. e.g. (length 1 m) and (distance 1 m) are regarded to be different property values. Because the one is describing a length and the other is describing a distance.
3. Each property value is atomic. This means that a property value cannot be described or replaced by any other property value(s) taken from the set of property values required in No. 1.
4 There exists a set of objects which potentially can belong to the real world area. We will call them **potential objects**.
5. To each object there belongs a non empty set of property values. Such a set is the '**description of the object**'.
6. Each object is described uniquely by its description. Which means that there are no two objects with the same description and different descriptions refer to different objects.
7. At each time t a real world area is in a special '**real world situation**' which is defined by a set of potential objects actually belonging to the real world area at that time. The potential objects belonging to a real world situation are called '**actual objects**' and the set of actual objects is called a '**state**'. 'This means that each change of a real world situation is simply given by inclusion and exclusion of potential objects from the real world state.
8. Each real world state is well defined. This means that for every object it is decidable whether it is actual or not.

To illustrate the meaning of these hypotheses, let us consider a small example: The real world area could be an enterprise. The potential objects for this enterprise might be buildings, cars, and employees. The property values available for descriptions might consist of inventory numbers, names of employees, social security numbers, licence numbers for the cars, values for describing loading space and transportation capacity.

A special situation of this real world area would be if the enterprise actually comprises three cars, two buildings, and ten employees, each described by some of the available property values.

Let us now proceed to the formal representation of the hypotheses given above. The alphabet of our first order language is given by;

(I) the set of basic logical symbols G := $\{ ,, (,), \neg, \exists, \forall, \wedge, \vee, \Rightarrow, \Leftrightarrow, \equiv \}$.

(II) the set of variables VS := $\{x, x', y, y', z, z', \ldots\}$.

(III) the predicate symbols RS := $\{A, B, D, Z\}$.

A and *B* are binary predicates and *D* and *Z* are unary. Thus the set of terms is given by VS and the atomic formulas are $D(x)$, $B(x,y)$, $A(x,y)$, $Z(x)$, $x \equiv y$. The rules for constructing well formed formulas are the usual syntactic rules of first order languages [3].

We now introduce the interpretation (U, ρ). U is the universe given by $U := \mathrm{D} \cup \Omega$ where Ω is the set of '**identifyers**' identifying the potential objects and D is the set of the '**descriptors**' d representing the property values. ρ is the mapping which assigns a relation over U^n to each n-ary predicate symbol. ρ is given by

(1) $$\rho : D \mapsto \mathrm{D} \qquad \rho : Z \mapsto \Omega^t \qquad \rho : A \mapsto \alpha \qquad \rho : B \mapsto \beta$$

where Ω^t is the set of identifyers identifying the actual objects, and α and β are given by

(2) $\alpha \subseteq \mathrm{D} \times \mathrm{D}$, $\alpha(d,d') \Leftrightarrow$ d and d' are descriptors for values of the same property type
$\beta \subseteq \mathrm{D} \times \Omega$, $\beta(d, \omega) \Leftrightarrow$ d is a descriptor belonging to the description of ω

Taking this interpretation of symbols, the first order language formulation of the above hypotheses is given by the following formulas:

(3) $\forall x\, \forall y\, [B(x,y) \rightarrow (D(x) \wedge \neg D(y))]$

(4) $\forall y\, \exists x\, [\neg D(y) \rightarrow B(x, y)]$

(5) $\forall x\, \exists y\, [D(x) \rightarrow B(x, y)]$

(6) $\forall x\, \forall x'\, [\forall y\, (B(y, x) \leftrightarrow B(y, x')) \leftrightarrow x \equiv x']$

(7) $\forall x\, \forall y\, [A(x, y) \rightarrow (D(x) \wedge D(y))]$

(8) $\forall x\, [D(x) \leftrightarrow A(x, x)]$

(9) $\forall x\, \forall y\, [A(x, y) \rightarrow A(y, x)]$

(10) $\forall x\, \forall y\, \forall z\, [(A(x, y) \wedge A(y, z) \rightarrow A(x, z))\,]$

(11) $\forall x\ [Z(x) \rightarrow \neg D(x)]$

(3) says that objects are described by descriptors (property values) only and no descriptor can be described by other descriptors (hypothesis 3). (4) and (5) say that every descriptor is used in some description and every object is described by at least one descriptor (hypothesis 5). (6) reflects the statement of hypothesis 6 and (7), (8), (9), and (10) are the axioms for an equivalence relation defining a partition over the set of descriptors where each class represents a property (type) (hypothesis 2). (11) states that the state is made up by objects only (hypotheses 7 and 8). The hypotheses 1 and 4 simply state the existence of D and Ω.

3. DISCUSSION OF THE STRUCTURE

The formulas given above are closed formulas without any function symbols. Therefore, the model of these formulas is a topological (relational) structure. For a discussion of this structure, let us introduce some new terms and symbols: The equivalence classes of α are called **attributes** and will be denoted by a_i with $i \in I_u$ (I_u being an arbitrary set of indices) and the quotient set $\mathrm{D}/\alpha =: \mathrm{U}$, is called the **set of attributes** (sometimes called the **universe**). The potential objects shall be denoted by integers $i \in I_O$ and ω_i is the identifyer for the i-th object. i. e. $\Omega := \{\omega_i \mid i \in I_O\}$. With these assumptions, we define

$$(12) \qquad e_i := \{d \mid \beta(d,\omega_i)\} \qquad i \in I_O$$

to be the **entity** representing (the description of) the i-th object and $O := \{e_i \mid i \in I_O\}$ to be the **entity set**.
For each entity e_i, $i \in I_O$, we have a set of **associated attributes** A_i and for each attribute a_j, $j \in I_u$, we have a set of **associated entities** E_j. They are given by

$$(13) \qquad A_i := \{ a_j \mid e_i \cap a_j \neq \emptyset, j \in I_u\} \quad i \in I_O \qquad E_j := \{ e_i \mid a_j \cap e_i \neq \emptyset, i \in I_O \} \quad j \in I_u$$

The associated attributes A_i and A_j need not satisfy $A_i \neq A_j$ for $i \neq j$ and the same is due to the dual situation with E_i and E_j. Thus we can define two further relations τ_O and τ_u given by

$$(14) \qquad \tau_O(e_i, e_j) \Leftrightarrow A_i = A_j \qquad \tau_u(a_i, a_j) \Leftrightarrow E_i = E_j .$$

It is evident that τ_O and τ_u are equivalence relations over O and U. We will call them the **entity type relation** and the **attribute type relation** respectively. The quotient set $Q_O := O/\tau_O$ is called the **set of entity types** and the elements of Q_O are the **entity types** (or simply **types**) denoted by O_i, $i \in I_A$ where $I_A \subseteq I_O$ is the index set of the representatives of the equivalence classes.The dual notions for τ_u are given in the same way: The quotient set $Q_u := U/\tau_u$ is called the **set of attribute types** and the elements of Q_u are the **attribute types** denoted by U_i, $i \in I_E$ where $I_E \subseteq I_u$ is the index set of the representatives of the equivalence classes. In contrast to the notion of an entity type, the notion of an attribute type is usually not used in database theory. It has been introduced to point out the dual nature of this notion to the notion of entity type. An attribute type corresponds best to what is called a class in object oriented databases. The entity types play an important role in relational database design. Because their relational representations are usually taken as starting relations (before normalization) in relational databases [2].
A type O_i is said to be a **subtype** of O_j if for the associated attributes $A_i \supset A_j$ holds. In a similar way we define U_i to be a **subclass** of U_j if for the associated entities $E_i \subset E_j$ holds. Conversely, U_j is called a **super class** of U_i. According to a general principle (the Galois Correspondence), the attributes of a superclass are also attributes in all its subclasses. This is what in object-oriented modelling is called **inheritance**.
For all these definitions there exists an underlying binary relation which is of really fundamental importance for database modelling. This is the relation ϕ, given by

$$(15) \qquad \phi \subseteq U \times O \qquad \phi(a_i, e_j) \Leftrightarrow a_i \cap e_j \neq \emptyset \qquad \text{for } i \in I_u \text{ and } j \in I_O.$$

A similar relation Φ connecting classes and types is of special importance for database design. Φ is defined by

$$(16) \qquad \Phi \subseteq Q_u \times Q_O \qquad \Phi(U_i, O_j) \Leftrightarrow a_i \cap e_j \neq \emptyset \qquad \text{for } i \in I_E \text{ and } j \in I_A.$$

The characteristic function χ of Φ (the 0-1-matrix with $\chi_{ij} = 1$ iff $\Phi(U_i, O_j)$) gives a good picture of the overall structure of a schema (that is the model applied to a special application). Such a characteristic function can be viewed as an incidence matrix for a hypergraph. The representation of this hypergraph in the form of directed or bipartite graphs leads to the well known graphical design tools in database modelling (e. g. Bachmann Diagram [2], Entity-Relationship-Diagram [1], Functional-Data-Model-Graph [8]).

Let us now turn back to the small example introduced in the last section. The enterprise was assumed to be made up of cars, buildings, and emplyees. Let us assume further that there are three types of cars, limousines, lorries, and busses, each having a specific set of associated attributes. The form of the characteristic function $\chi(\Phi)$ could be

$$\chi(\Phi) = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \tag{17}$$

Here we have assumed that $O = \{O_1,O_2,O_3,O_4,O_5\}$ where O_1 stands for buildings, O_2 for limousines, O_3 for lorries (which is a subtye of limousines), O_4 for busses, and O_5 for the employees. The classes are $U = \{U_1,U_2,U_3,U_4,U_5,U_6\}$. The attribute type U_1 contains those attributes which lorries and limousines have in common. The attribute type U_2 can be interpreted to contain the attributes describing the class of cars containing the entity types O_2, O_3, and O_4. U_3 are the attributes applied to busses only, U_4 are the attributes concerning the employees, U_5 might be some inventory attributes applying to buildings and lorries and U_6 are the attributes applicabe to lorries only.

4. CONCLUSION

We have shown how a database model can be derived from a real world model. The procedure is not really new, because it is the classical procedure of natural science. But , as far as I know, it has never been described in the literature for database modelling and no model has been derived in such a way. It is a straightforward method of deriving database models. The terms are formulated and defined in a mathematical style giving them a clear meaning and embedding them in an overall structure. The symmetries and dualities of this structure reveal a deep insight into the essentials of database modelling.

Acknowledgement: The work on this subject has been done in close cooperation with Prof. R. Albrecht of the "Institut für Informatik" at the University of Innsbruck. He has encouraged me to give a set theoretic formulation of a structure which is implicitly contained in most of the database models.

5. REFERENCES

[1] P. P. Chen: The Entity-Relationship Model - Toward a Unified View of Data. ACM Trans. on Database Systems 1 (1976) 9 - 36

[2] C. J. Date: An Introduction to Database Systems. Vol. 1. Addison Wesley (1986)

[3] H.-D. Ebbinghaus, J. Flum, W. Thomas: Einführung in die mathematische Logik. Bibliographisches Institut Mannheim (1992)

[4] M. Hammer, D. McLeod: Database Description with SDM: A Semantic Database Model. ACM Trans. on Database Systems 6 (1981) 351 - 386

[5] R. Hull, Chee K. Yap: The Format Model: A Theory of Database Organization. J. of the ACM 31 (1984) 518 - 537

[6] R. Keil: The property orientd model: A set theoretic approach.CIS-91 Conference on Intelligent Systems (1991) 181 - 191

[7] Won Kim: Object-Oriented Databases: Definitions and Research Directions. IEEE Trans. on Knowledge and Data Engineering 2 (1990) 327 - 341

[8] E. H. Sibley, L. Kerschberg: Data architecture and data model considerations. Proceedings of the AFIPS Conference 46 (1977) 85 - 96

State of the art in parallel discrete event simulation

Rassul Ayani
Department of Teleinformatics, Computer Systems Laboratory
Royal Institute of Technology (KTH)
Stockholm, Sweden

Abstract

This paper surveys various approaches to executing discrete event simulation programs on a parallel computer. Parallelization of discrete event simulation programs requires an adequate synchronization scheme. We review several synchronization schemes that have appeared in the literature in recent years and discuss future trends in parallel simulation.

1. Introduction

Simulation of a system may have several objectives, including: (i) understanding the behavior of a system; (ii) obtaining estimates of the performance of a system; (iii) guiding the selection of design parameters and (iv) validation of a model. Simulation has been used in many areas, including manufacturing lines, communication networks, computer systems, VLSI design, design automation, air traffic and road traffic systems.

Two separate classes of methodologies, called *continuous* time and *discrete* time simulation, have emerged over the years and are widely used for simulating complex systems. As the terms indicate, in a continuous simulation changes in the state of the system occur continuously in time, whereas in a discrete simulation changes in the system take place only at selected points in time. Thus, in a discrete-event simulation (DES) events happen at discrete points in time and are instantaneous. One kind of discrete simulation is the fixed time increment, or the *time-stepped* approach, the other kind is the *discrete-event* method.

The conventional DES is sequential and may consume several hours (and even days) to run a practical simulation on a sequential machine. Parallel computers are attractive tools that are used to reduce execution time of such simulation programs. In this paper, we discuss parallel discrete event simulation (PDES) where several processors of a multiprocessor system cooperate to *execute a single simulation program*.

The common approach to PDES is to view the system being modeled, usually referred to as the *physical system*, as a set of *physical processes* (PPs) that interact at various points in simulated time. The simulator is then constructed as a set of *logical processes* (LPs) that communicate with each other by sending timestamped messages. In this scenario, each logical process simulates a physical process. Each LP maintains its own logical clock and its own event list. The logical process view requires that the state variables are statically partitioned into a set of *disjoint states* each belonging to an LP. This view of PDES as a set of communicating sequential simulators is used by all of the simulation methods reviewed in this paper.

It can be shown that no *causality* errors occur if each LP processes events in *non-decreasing* timestamp order [9]. This requirement is known as *local causality constraint.*

Two main paradigms have been proposed for PDES: *conservative* and *optimistic* methods.

2. Conservative and Optimistic approaches to PDES

Conservative approaches strictly avoid the possibility of any causality error ever occurring. Several conservative approaches to PDES have been proposed in the literature. These approaches are based on processing safe events. The main difference between these methods, as discussed in this section, lies in the way they identify safe events.

Chandy and Misra proposed one of the first conservative PDES algorithms [9]. Several researchers have proposed window based conservative parallel simulation schemes (e.g., see [3], [7], [10]). The main idea behind all these schemes is to identify a time window for each logical process such that events within these windows are safe and can thus be processed concurrently. The basic constraint on such schemes is that the events occurring within each window are processed sequentially, but events within different windows are independent and can be processed concurrently.

The performance of the window based schemes depends heavily on how the windows are assigned to the processors. Several scheduling schemes have been proposed and evaluated in [2]. As discussed in [3], the number of non-empty windows and the size of each one depends on features of the system being simulated, e.g. message population, network topology, and network size.

Several researchers have studied the performance of the conservative schemes. The most extensive performance result has been reported by Richard Fujimoto [5]. According to Fujimoto, performance of the conservative algorithms is critically related to the degree to which logical processes can look ahead into their future simulated time. Ayani and Rajaei [3] present an intensive performance study of the conservative time window scheme on shared memory multiprocessors.

Optimistic approaches to PDES, as opposed to conservative ones, allow occurrence of causality error. These protocols do not determine *safe* events; instead they *detect* causality error and provide mechanisms to *recover* from such error.

The Time Warp mechanism proposed by Jefferson and Sowizral is the best known optimistic approach. The Time Warp mechanism allows an LP to execute events and proceed in its local simulated time, called local virtual time (LVT), as long as there is any messeage in its input queue. This method is optimistic because it assumes that message communications between LPs arrive at proper time, and thus LPs can be processed independently. However, it implements a rollback mechanism for the case where the assumption turns out to be wrong, i.e. if a message arrives at a node in its past. The method requires both time and space to maintain the past history of each node, and to perform the rollback operation whenever necessary.

Several schemes for *undoing* side effects caused by erroneous messages have appeared in the literature. In the *aggressive cancellation* mechanism, when a process rolls back *antimessages* are sent immediately to cancel erroneous messages. In *lazy cancellation*, antimessages are not sent immediately after rollback. Instead, the rolled back process resumes execution of events from its new LVT. If the re-execution of the events regenerates the *same* message, there is no need to cancel the message. Under aggressive cancellation, a process may send unnecessary antimessages. Under lazy cancellation there are no unnecessary antimessages. However, lazy cancellation may allow erroneous computation to spread further because antimessages are sent later.

The lazy cancellation mechanism may improve or degrade performance of the time warp depending on features of the application. Most of the performance results reported in the literature suggest that lazy cancellation improves performance. However, one can construct cases where lazy cancellation is much slower than aggressive cancellation .

Several researchers have reported successes in using Time Warp to speedup simulation problems. For instance, Fujimoto [5] has reported significant speedup for several queueing networks.

3. Hybrid Approaches

The deadlock handling is the main cost factor in conservative methods. In optimistic approaches, the detection and recovery of causality errors require state saving and rollback. State saving may require a considerable amount of memory if system state consists of many variables that must be saved frequently.

It seems reasonable to combine the advantages of these two approaches in a hybrid protocol. The issue of combining the two approaches has received considerable attention in recent years, since the limitations of each paradigm are better understood. It is believed that the future PDES paradigm will be a hybrid one! There are three general categories of hybrid approaches:

(i) To add optimism to a conservative approach. For instance, in the speculative simulation method proposed by Horst Mehl [8] whenever an LP is to be blocked, it optimistically simulates the events in its event list, but keeps the result locally until it becomes committed.

(ii) To add conservatism to an optimistic approach. One may try to bound the advancement of LVTs in Time Warp. This technique reduces rollback frequency and the rollback distance in general. However, it tends to reduce the degree of available parallelism as well. The main problem with this category of schemes is how to determine a boundary for limiting the optimism. The bounded time warp (BTW) proposed by Turner and Xu [13] divides the simulation duration time interval into a number of equal intervals and all events within an interval are processed before the next one is started. The local time warp (LTW) proposed by Rajaei [11] partitions the system into a set of clusters each containing a number of LPs. The LPs within each cluster are synchronized by Time Warp, whereas the inter-cluster synchronization is based on the conservative time window scheme described in [3].

(iii) Switching between Optismism and Conservatism. Some researchers, e.g. [1], suggest to switch between the conservative and the optimistic schemes. This approach is attractive, especially when the behavior of the application changes dynamically.

4. Conclusions

The state of the art in PDES has advanced very rapidly in recent years and much more is known about the potential of the parallel simulation schemes. In particular, the extensive performance studies conducted by several researchers have identified strengths and weaknesses of the parallel simulation schemes. In this paper, we attempted to provide an insight into various strategies for executing discrete event simulation programs on parallel computers and to highlight future research directions in this field. The implementation of the event-list and its impact on performance, though important, was not covered in this paper (interested readers are referred to [12], [6]).

Conservative methods offer good potential for certain classes of problems where application-specific knowledge can be applied to exploit look ahead. Optimistic methods have had significant success in a wide range of applications, however, reducing the state saving costs is still a research problem. The issue of combining the two approaches has received considerable attention in recent years. It is believed that the future PDES paradigm will be based on hybrid approaches.

Acknowledgment Our research project on parallel simulation (PARSIM) supported by the Swedish National Board for Industrial and Technical Development (NUTEK) was the driving force behind this study.

References

[1] K. Arvind and C. Smart. Hierarchical parallel discrete event simulation in composite elsa. In 6^{th} *Workshop on Parallel and Distributed Simulation*, volume 24, pages 147–158. SCS Simulation Series, January 1992.

[2] R. Ayani and H. Rajaei. Event scheduling in window based parallel simulation schemes. In *Proceedings of the Fourth IEEE Symposium on Parallel and Distributed Computing*, Dec 1992.

[3] R. Ayani and H. Rajaei. Parallel simulation based on conservative time windows: A performance study. *To appear in Journal of Concurrency*, 1993.

[4] R. Felderman and L. Kleinrock. Two processor Time Warp analysis: Some results on a unifying approach. In *Advances in Parallel and Distributed Simulation*, volume 23, pages 3–10. SCS Simulation Series, January 1991.

[5] R. M. Fujimoto. Parallel discrete event simulation. *Communications of the ACM*, 33(10):30–53, October 1990.

[6] D. W. Jones. An empirical comparison of priority-queue and event-set implementations. *Communications of the ACM*, 29(4):300–311, Apr. 1986.

[7] B. D. Lubachevsky. Efficient distributed event-driven simulations of multiple-loop networks. *Communications of the ACM*, 32(1):111–123, Jan. 1989.

[8] H. Mehl. Speedup of conservative distributed discrete-event simulation methods by speculative computing. In *Advances in Parallel and Distributed Simulation*, volume 23, pages 163-166. SCS Simulation Series, January 1991.

[9] J. Misra. Distributed-discrete event simulation. *ACM Computing Surveys*, 18(1):39–65, March 1986.

[10] D. M. Nicol. Performance bounds on parallel self-initiating discrete-event simulations. *ACM Transactions on Modeling and Computer Simulation*, 1(1):24–50, January 1991.

[11] H. Rajaei, R. Ayani, and L.-E. Thorelli. The local time warp approach to parallel simulation. In 7^{th} *Workshop on Parallel and Distributed Simulation*, January 1993.

[12] R. Ronngren, R. Ayani, R. Fujimoto, and S. Das. Efficient implementation of event sets in time warp. In *Workshop on Parallel and Distributed Simulation (PADS)*, volume 23, pages 101–108. SCS Simulation Series, May 1993.

[13] S. Turner and M. Xu. Performance evaluation of the bounded Time Warp algorithm. In 6^{th} *Workshop on Parallel and Distributed Simulation*, volume 24, pages 117–128. SCS Simulation Series, January 1992.

Simulation of Object-Oriented Continuous Time Models

Sven Erik Mattsson

Department of Automatic Control, Lund Institute of Technology
Box 118, S-221 00 LUND, Sweden

Abstract. Simulation of object-oriented continuous time models requires solution of differential-algebraic equations. This paper discusses use of symbolic analysis and manipulation for detection of incompletenesses and inconsistencies, and for reduction and transformation of the user's problem into a form well-suited for numerical solution.

1. Introduction

The aim of object-oriented modeling is to support model development and facilitate reuse so a model can be used to solve various problems and so model components easily can be modified to describe similar systems. Object-oriented modeling exploits modern concepts for model structuring to implement this. Models can be hierarchically decomposed with well defined interfaces to describe interaction with other components. Inheritance and specialization support easy modification. We have developed a new general object-oriented modeling language called Omola and an environment, called OmSim, of tools supporting modeling and simulation of Omola models. Omola and OmSim are described in [7], where references to other object-oriented modeling languages also are given.

The explicit state space form, $\dot{x} = f(t,x)$, required by today's most used general-purpose continuous simulation tools (ACSL, CSMP, EASY5, Simnon, SIMULINK etc.) is not general enough to support natural model decompositions and flexible reuse of models. For continuous time modeling it is necessary to allow behaviour to be described by differential-algebraic equation (DAE) systems; $g(t,\dot{x},x,v) = 0$, where x and v are vectors of unknown dynamic respectively algebraic variables to be solved for. It is often a significant effort to transform a problem into explicit state space form. Use of symbolic manipulation to support automatic generation of state-space models for special classes of systems are discussed in [2].

There are two major approaches to solving a simulation problem: simultaneous or modular solution. In the simultaneous approach, all equations are collected and solved as one global problem. The idea of modular solution is to calculate the behaviour locally for each component and then include the effects of interaction by iteration or by external modification of the states at each step of integration. The modular approach has been very successful in object-oriented discrete-event simulation, where the objects are actors and communicate by means of message passing during a simulation run. It is not a good idea to use message passing for continuous-time simulation since interaction means that variables should be equal for all times. Efficiency is lost, since it is necessary to communicate frequently. Both approaches are of practical interest, but here we will focus on the simultaneous approach.

There are numerical DAE solvers which treat the problem, $g(t,\dot{x},x) = 0$, if they are provided with a routine that calculates the residual, $\Delta = g(t,z,x)$, for given arguments. A good numerical DAE solver, which is available in public domain, is DASSL by Petzold [1]. DASSL implements a multi-step method. It converts a DAE system into a nonlinear algebraic equation system by approximating derivatives with backward differences of order up to 5. The simplest discretization is backward Euler, $\dot{x}_{n+1} \approx (x_{n+1} - x_n)/h$, where h is the step size. It turns the DAE problem into the implicit recursion $g(t_n, (x_n - x_{n-1})/h, x_n) = 0$. A drawback is that discontinuities force multi-step methods to restart with low-order approximations. One-step discretizations of the Runge-Kutta type avoids this problem. One such available high-quality solver is RADAU5 [5].

Unfortunately, today's numerical DAE solvers are not able to solve all mathematically well-defined problems. For example, they fail to solve high-index problems. In this paper we will discuss use of symbolic analysis and manipulation to support and facilitate solution of DAE systems.

2. Structural analysis

Since it is important that the methods to support solution of DAE systems are able to handle large problems, it is advisable to first analyze the structure to detect structural defects and to decompose the problem into a sequence of subproblems, which can be analyzed in turn.

To explain the idea, let us first consider the nonlinear algebraic equation system, $h(x) = 0$, and focus on which variables that appear in each equation rather than how they appear. This information can be represented by the "structure" Jacobian, where each element i,j, is zero if x_j does not appear in the expression h_i, otherwise it is one. The idea is now to permute unknowns and equations to make the structure Jacobian become Block Lower Triangular (BLT). A BLT partitioning reveals the structure of a problem. It decomposes a problem into subproblems which can be solved in sequence. There are efficient algorithms (cf. [3, 6]), which can be implemented in a few pages of Pascal or C++ code, for constructing BLT partitions with diagonal blocks of minimum size.

Structural singularities

A basic step of a BLT partitioning is to permute the equations to make all diagonal elements of the structure Jacobian non-zero. We can also view this step as a procedure which assigns to each variable x_i to a unique equation $h_j = 0$ such that x_i appears in h_j. If it is impossible to pair variables and equations in this way then the problem is structurally singular.

If a problem is singular, it is desirable to give the user some hints what is wrong. If any variable have not been assigned an equation, we can produce some hints by assigning each unassigned variable to a fictitious equation in which all unknowns appear and make a BLT partitioning. The fictitious equations will all be collected in the last block. Thus the variables of the other blocks can (at least structurally) be determined, whereas the variables of the last block are only partially constrained and the equations to be added must at least include one of these variables. Unfortunately, it is often the case that many variables end up in the last block, thus giving little hint. To produce some hints for the dual problem when there are redundant equations, we create a fictitious variable for each redundant equation and make them appear in each equation and make a BLT partitioning. To remove the overdeterminacy, equations of the first block should be removed.

To check for structural singularities of DAE systems we should not distinguish between the appearance of x_i and the appearances of its derivatives; the appearances of $\dot{x}_i$, $\ddot{x}_i$, etc., are considered as appearances of x_i. Thus to check if the first-order DAE problem $g(t, \dot{x}, x) = 0$ is structurally singular, we check if the algebraic problem $g(t, z, z) = 0$ is structurally singular with respect to z.

It is advisable to check for structural singularities since it is a simple and efficient way to catch many bad models early.

Time-invariant parts

It is simple to use the BLT-partition to find out if variables in fact are constant or time invariant, i.e., the variables only change their values when parameters change values.

By analyzing the variability of the subproblems in turn and by calculating variables that are implicitly constant the size of the problem can be reduced. By also calculating constant subexpressions of the equations and exploiting zeros it is possible to reduce the complexity of the equations. It may be useful to make a new BLT-partitioning of each simplified block, since it may now decompose into several subproblems.

The DAE index

The solution of a DAE problem with given initial values may involve integration, but also differentiation. The needs to differentiate are classified by the DAE *index*, which is defined as the minimum number of times that all or part of $g(t, \dot{x}, x) = 0$ must be differentiated with respect to t in order to determine $\dot{x}$ as a continuous function of x and t [1]. An ODE, $\dot{x} = f(t, x)$, is index 0. The problem $\dot{x} = f(t, x, y)$ and $g(t, x, y) = 0$ is index 1 if the Jacobian $\partial g / \partial y$ is nonsingular.

Available numerical DAE solvers may solve some index 2 problems, but they fail when index is greater. High index problems are difficult to solve because they involve differentiations. Available DAE solvers are designed to integrate, i.e., calculate x from $\dot{x}$, but if the index is greater than 1, they also should differentiate, i.e., calculate $\dot{x}_i$ from x_i for some components, which is quite another numerical problem. In numerical integration it is assumed that the errors can be made arbitrarily small by taking sufficiently small steps. In numerical differentiation it is well-known that the step cannot be taken infinitely small. When the DAE solver estimates the errors by taking steps of different sizes and comparing the results, it gets error estimates that behave irregularly. Today's DAE solvers cannot handle this situation, but they have to exit with some error message.

High index problems are natural in object-oriented modeling, since we want to build general model components. When connecting them to make a model of a system we may introduce algebraic constraints on dynamic variables. For example, when building a model library for mechanical components, it is natural to model the components as moving freely in a three dimensional space. When the components are connected to build a model of mechanical system, the motion of each component are constrained by geometric constraints which implicitly define reaction forces and torques.

The algorithm by Pantelides [9] finds the minimum number each equation has to be differentiated to make the problem index 1. This algorithm is easy to implement. It is an extension of the algorithms for assigning equations to variables. It is not necessary to perform any differentiations. To find out the orders of the derivatives appearing in a an equation that is differentiated m times, we need only know the orders of the derivatives in the original equation and increment with m.

3. Reduction of size and complexity

The possibilities to reduce the size and the complexity of the problem that needs to be solved by the numerical DAE solver will now be considered. Elimination of invariant parts was indicated above.

Index reduction

To facilitate the numerical solution, it is of interest to reduce the index to 1. If we differentiate the equations according to the results of Pantelides's algorithm we get the so called differentiated index-1 problem. It is often less satisfactory to solve this problem because its set of solutions is larger. The algebraic relations of the DAE are only implicit in the differentiated problem as solution invariants. Unless linear, these invariants are generally not preserved under discretization. As a result, the numerical solution drifts off the algebraic constraints. We have developed a combined symbolic and numeric index reduction method which constructs an index-1 problem having the same solution set as the original problem; see [8], where also other techniques are surveyed.

In the following we will for simplicity assume that the problem is index 1 and that it is BLT-partitioned with respect to the highest appearing derivative of each unknown variable.

Tearing and Hiding of Algebraic Variables

A model developer often introduces algebraic variables for display and plotting purposes. It means that the associated equation for such a variable is in fact a simple assignment and that the value is not needed when solving for the dynamic variables. It is easy to sort out such variables by analyzing the BLT-partitioned problem and calculate their values after having solved the dynamic problem.

To use a numerical DAE solver we need only produce a routine that calculates the residuals for given values of the variables and their derivatives. Algebraic variables, which are easy to calculate when the values of the dynamic variables and their derivatives are known need not be introduced as unknowns to the numerical DAE solver. Variables that can be hidden for the numerical DAE solver are found by analyzing the BLT-partitioned problem. An algebraic variable, v, which belongs to a scalar block, where it is possible to solve for v analytically is a good candidate. It is not advisable to eliminate v by substitutions since that could mean that the residual routine "calculates v"several times. Auxiliary variables are often introduced to denote an expression that appears several times. Note that also when using numerical ODE solvers it is common to have a lot of auxiliary algebraic variables that in fact are hidden for the numerical solver.

The idea can be extended to blocks of algebraic variables. Furthermore, the idea can also be used when a block has both algebraic and dynamic variables. In more general terms, the variables and equations are divided into two sets so that it is easy to solve for the variables in the first set if the variables of the other set is known. This kind of partitioning is called tearing. Here the intention is to hide the variables of the first set and let the DAE solver treat the variables of the second set. There are many algorithms for tearing. Unlike the situation for BLT partitioning, there are no clear winners. We refer to the textbooks [3, 6] for algorithms and further discussion.

Condition numbers and pivoting are important concepts when solving equation systems. By restricting manipulations to be local to each block, much is gained since it is, in general, a bad idea to use an equation to solve for a variable belonging to another diagonal block. It is not a good approach to use Gauss elimination to triangularize a block, since it is easy to make the problem ill-conditioned and it may also happen that the triangularization causes division by zero. For small or sparse linear equation systems it is feasible to use Cramer's rule and calculate the inverse by calculating the determinant and minors [2].

Efficient numerical solution

Symbolic analysis can be used to support selection of numerical solver. It is easy to find out if there are time delays, discontinuities, sampled subparts etc. If we have an object-oriented implementation of numerical solvers as proposed in [4] it would be possible to use such information to costumize the solver. A simulation is often repeated several times with small variations in input data or parameter values. It is then useful to make a retrospective analysis of the first results to decide if another solution method should be used.

As indicated above the numerical solvers convert the DAE problems to algebraic equation systems. The symbolic methods discussed above can be applied to these algebraic problems. Symbolic analysis can be used to provide information about the structure of the Jacobian to avoid calculation of zeros or to exploit sparsity or special band structures. Symbolic and automatic differentiation [10] can be used for calculation of Jacobians.

4. Conclusions

In this paper we have indicated needs and use of symbolic manipulation to facilitate numerical solution of DAE-problems, which is an important issue in object-oriented modeling. A good illustration to what can be achieved is modeling of the motion of mechanical systems. To get a flexible model library, it is natural to model components objects that can move freely in a three dimensional space, but when the components are connected to build a model of mechanical system such as an industrial robot, the motion of each component are constrained drastically. By use of symbolic methods outlined above, the size of the problem to be solved numerically can typically be reduced to one tenth of the original problem.

The methods outlined above are to a large extent supported by OmSim. The economical support to the development of OmSim from the Swedish National Board for Industrial and Technical Development (NUTEK) is gratefully acknowledged.

5. References

[1] K. E. BRENAN, S. L. CAMPBELL, and L. R. PETZOLD. *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations.* North-Holland, Amsterdam, 1989.

[2] F. CELLIER and H. ELMQVIST. "Automated formula manipulation supports object-oriented continuous-system modeling." *IEEE Control Systems,* **13:2**, pp. 28–38, April 1993.

[3] I. S. DUFF, A. M. ERISMAN, and J. K. REID. *Direct Methods for Sparse Matrices.* Clarendon Press, 1986.

[4] K. GUSTAFSSON. "An object oriented implementation of software for solving ordinary differential equations." In *OON-SKI '93, Proceedings of the First Annual Object-Oriented Numerics Conference, Sunriver, Oregon,* pp. 318–330. SIAM, 1993. To be published in Scientific Programming.

[5] E. HAIRER, C. LUBICH, and M. ROCHE. *The Numerical Solution of Differential-Algebraic Systems by Runge-Kutta Methods.* Lecture Notes in Matematics No. 1409. Springer-Verlag, Berlin, 1989.

[6] R. S. H. MAH. *Chemical Process Structures and Information Flows.* Butterworths, 1990.

[7] S. E. MATTSSON, M. ANDERSSON, and K. J. ÅSTRÖM. "Object-oriented modeling and simulation." In LINKENS, Ed., *CAD for Control Systems,* pp. 31–69. Marcel Dekker, Inc., 1993.

[8] S. E. MATTSSON and G. SÖDERLIND. "Index reduction in differential-algebraic equations using dummy derivatives." *SIAM Journal of Scientific and Statistical Computing,* **14:3**, pp. 677–692, May 1993.

[9] C. PANTELIDES. "The consistent initialization of differential-algebraic systems." *SIAM Journal of Scientific and Statistical Computing,* **9**, pp. 213–231, 1988.

[10] L. B. RALL. *Automatic Differentiation — Techniques and Applications.* Lecture Notes in Matematics No. 120. Springer-Verlag, Berlin, 1981.

GUIDELINES FOR AN OBJECT ORIENTED ADVANCED MMI FOR ENEL SIMULATORS

L.Castiglioni-S.d.I.Automazione Ind.-via Winckelmann,1 Milano, ITALY

F.Pretolani-ENEL C.R.A.- via Volta,1 - Cologno , Milano, ITALY

Abstract. An object oriented methodology is proposed for the development of the specifications of an Advanced Man Machine Interface for power plant engineering simulators. The most important benefits of this new approach are described and applied to a project in progress at ENEL.

1. Introduction

This paper describes some guidelines for the specifications of an Advanced Man Machine Interface (named LEGOMMI) for power plant engineering simulators in development at the Automatica Research Centre of the Italian Electricity Company (ENEL CRA). This class of simulators has a wide and important use in ENEL especially in the fields of the operators training, plant dynamic project assessment, control systems design and optimisation and in the factory trial of the hardware of the actual control systems.

These specifications arise from the ones of the MMI software package used at present in ENEL for its engineering simulators extending performances in terms of portability and object oriented architecture.

As for the portability, the need to easily migrate to future state-of-the-art hardware platforms leads to the choice of "Open System" software environments for their easy transportability on most of the hardware platforms and in particular to the choice of X-Window and OSF-Motif graphical environments.

The object oriented approach seems to be very interesting in environments where the application requirements may evolve quickly: a system designed using an object-oriented methodology is robust for functionality changes while, using a traditional function oriented methodology, a requirement change may ask for massive restructuring.

In the following sections after an overview of the functional and structural requirements of the MMI system, the description of the objects constituting the system and their relationships is proposed; this view of the system is called *object model* and may be represented in an expressive form via *object diagrams* All the steps constituting the system development life cycle (analysis, design, implementation) are based on the object model and can be described using the same concepts and notation conventions: they only gain additional implementation details after each design step.

Next the object model is integrated with the description of the *system architecture* where the system is organised in subsystems having a client-server relationship.

A further chapter contains a brief discussion on the implementation details of major interest concerning the graphic objects and the use of the data base.

2. Functional requirements

LEGOMMI is designed to provide a MMI for engineering simulators developed in ENEL using the LEGOCAD [1] modelling environment.

LEGOCAD is based on a FORTRAN library of computational modules describing single components of power plants (LEGO modules library). All information concerning the LEGO modules, such as topologic information and geometric or physical data, are located inside the LEGO modules library.

The mathematical models, representing in general portions of the entire power plant, are obtained aggregating instances of LEGOCAD modules (named blocks); at a higher hierarchical level the aggregation of one or more mathematical model gives rise to a simulator.

The main functional requirements for LEGOMMI can be summarised as follows:

1) LEGOCAD engineering simulators are used in different fields and for different purposes, from thermodynamics studies to control systems design, to operators training: LEGOMMI must be satisfactory for all these working environments. This leads to a MMI including a wide variety of services such as mimics, alarms treatment, historical data trends visualisation and control room instrumentation emulation typical of an industrial plant supervisory MMI.

2) when configuring a mimic the user must be helped in establishing the desired connections between the graphic dynamic objects behaviour in animation and the related simulation variables. The use of information derived from the topology of the underlying mathematical models is considered to be basic for this purpose. This feature should however be optional when the simulator model has not yet been created or completed. In this case the links between the simulator variables and the graphic dynamic objects placed on the mimics will be resolved in a following phase called MMI installation.

3) navigation through simulator related information must be provided. The user can retrieve model's informations of plant components selecting the related graphic object on a MMI page. These information can include geometric and physical data and input/output block variables. The variables list can then be used to request particular run-time services such as input variable value manipulation or graphical presentations. This requirement may be achieved using queries to a relational data-base where simulator's topologic information reside.

4) LEGOMMI has to manage correctly context information. Installed MMI pages are correct only in connection with a particular simulator, that relies on a particular LEGOCAD library. If the simulator model structure and/or the modules library have been modified after the installation of the MMI pages, the system should force the user to reinstall the pages out of order automatically fixing eventually changed links between the graphic components and the simulation variables.

5)LEGOMMI graphic ability should also be used in the future graphic construction of the simulator mathematical models. The design of the object classes should be made so that subclasses with additional methods and attributes could be used for the mathematical model construction. This solution will allow a simple automatic generation of animated MMI pages using the same graphic pages assembled for the mathematical model construction where object with topological information will be substituted with their simpler parents.

6)Particular performances are needed in run-time: a)user interactions, such as changing the current MMI page, should occur within 1 second; b) page refresh with updated simulation variables should occur every 1 second; c) control room instrumentation emulation should allow delay times not greater then 300msec between the user action and the feedback on the emulated instrumentation panel; d) less restrictions are imposed for the time required to retrieve model related information in the data-base: standard SQL data base committing time is considered acceptable.

3. Architectural Requirements.

At present the LEGOCAD simulation environment is composed of a series of tools and utilities that co-operates to the simulator's construction: LEGOMMI must be integrated in this simulation environment having the same basic architectural characteristics.

The simulation environment has flexibility characteristics and can work on a cluster of co-operating workstations connected via Ethernet; it is implemented in accordance to international de facto standards [2] such as UNIX as operating system, X-Window (XWS) and OSF/Motif as graphics windowing system, TCP/IP as networking protocol.

The hardware platforms supported at present are: DECStation 5000 s.o. ULTRIX, VAXStation 4000 s.o. VMS, IBM RISC6000 s.o. AIX, PC386/486 s.o. SCO UNIX.

The simulation environment and LEGOMMI will co-operate sharing information stored in a relational database. Database queries will be conform to SQL standard, granting system functionality on different SQL compliant databases.

4. Object Model

The Object Model production is the first step in the development of specifications using an object oriented approach; this phase gives rise to the description of the objects constituting LEGOMMI and their relationships. An OMT (Object Modelling Technique) notation derived from [2] and [3] is adopted, a brief description of the notation and its meaning is summarised in App.1.

In Fig.1 the object diagram proposed for LEGOMMI is shown: objects that are not properly part of LEGOMMI (simulator, LEGOCAD modules and variables) but strictly connected with it are also represented.

The class *Context* has a single instance that provides a context granting consistency amoung the MMI's plant representation and the *Simulator* or the *Animated Icons Library*. *Context* has a great importance during installation and simulation, while at configuration time consistency may not be granted meaning that the links between models' variables and graphic MMI may not be solved in this phase, so allowing MMI configuration before the simulator models have been completed.

The *Animated Icons Library* include and organize a series of *Animated Icons*. These are rectangular windows containing a schematic representation of a plant component. *Animated Icons* own dynamic graphic attributes and are connected with one or more simulator variables supplying the related components states. These variables values can be represented into the *Animated Icons* as numerical values called displays or can cause movement or colour modifications in parts of the icon's draw according to the *Animation Table*.

The *Animation Table* contains one row for every simulation variable specifying the ranges of the variable and the associated corresponding colours; during the simulation the actual values of the variables are compared with the ranges thresholds and the icon's draw is eventually refreshed.

During the configuration of the *Animated Icons* it is optionally available a function to facilitate the retrieving of the simulation variables to be associated to the icons. This is obtained through the implementation of the *Component* object of class. This class specifies for each *Component* the names of the associated LEGOCAD modules and for each of them gives the default animation variables.

In LEGOMMI the collection of MMI graphical *Pages* can be divided into two different subclasses: *Configured Pages* that are typical MMI pages configured by the user before the simulation phase, and *Dynamically Configurable Pages* that provide services where the links with simulator's variables are specified during the simulation according to the user needs. *Configured pages* include background draw elements defined by the class *Elementary Draw Elements* and objects of the class *Active Objects*.

All *Active Objects* reference to simulation variables for representing and/or modifying graphically (only for input variables) their values. Variables values, kept up-to-date by the simulator, are represented as instances of the class *Dynamic Simulation Variables*.

Some *Active Objects* subclasses *(Graphics* and *Input to Model)* are also used as components of the *Dynamically Configurable Objects* class. This last class provides, during the simulation, a dialogue facility requesting variables names to be associated to its objects; in this phase additional modellistic information can be retrieved through queries to a relational data base managing all simulator topologic related information included in the object class *Simulator Topology Database*.

5. System Architecture

The whole LEGOMMI system will be structured as a group of subsystems with client-server dependencies. This system decomposition is affected by the particular phase considered. Three phases characterize LEGOMMI use: configuration, installation and simulation. Figs. 2,3 describe the corresponding decomposition diagrams where arrows connect client to server subsystems.

During the configuration phase the Page Editor is used to include graphic objects on the current page. These graphic objects can then be configured using the Attributes Configurator. This configurator presents to the user the editable list of attributes that can be defined for the graphic object in use. Some attributes are simply configurable inserting a line of text in editing fields or selecting an item from an option menu, others are configurable using Simple Attribute Configurators such as colours and fonts configurators. For animated icons a dedicated editor is provided (Animated Icon Configurator) that makes use as servers of the Draw and Animation Table Editors. The Draw Editor can also be used directly by the Page Editor for background drawings.

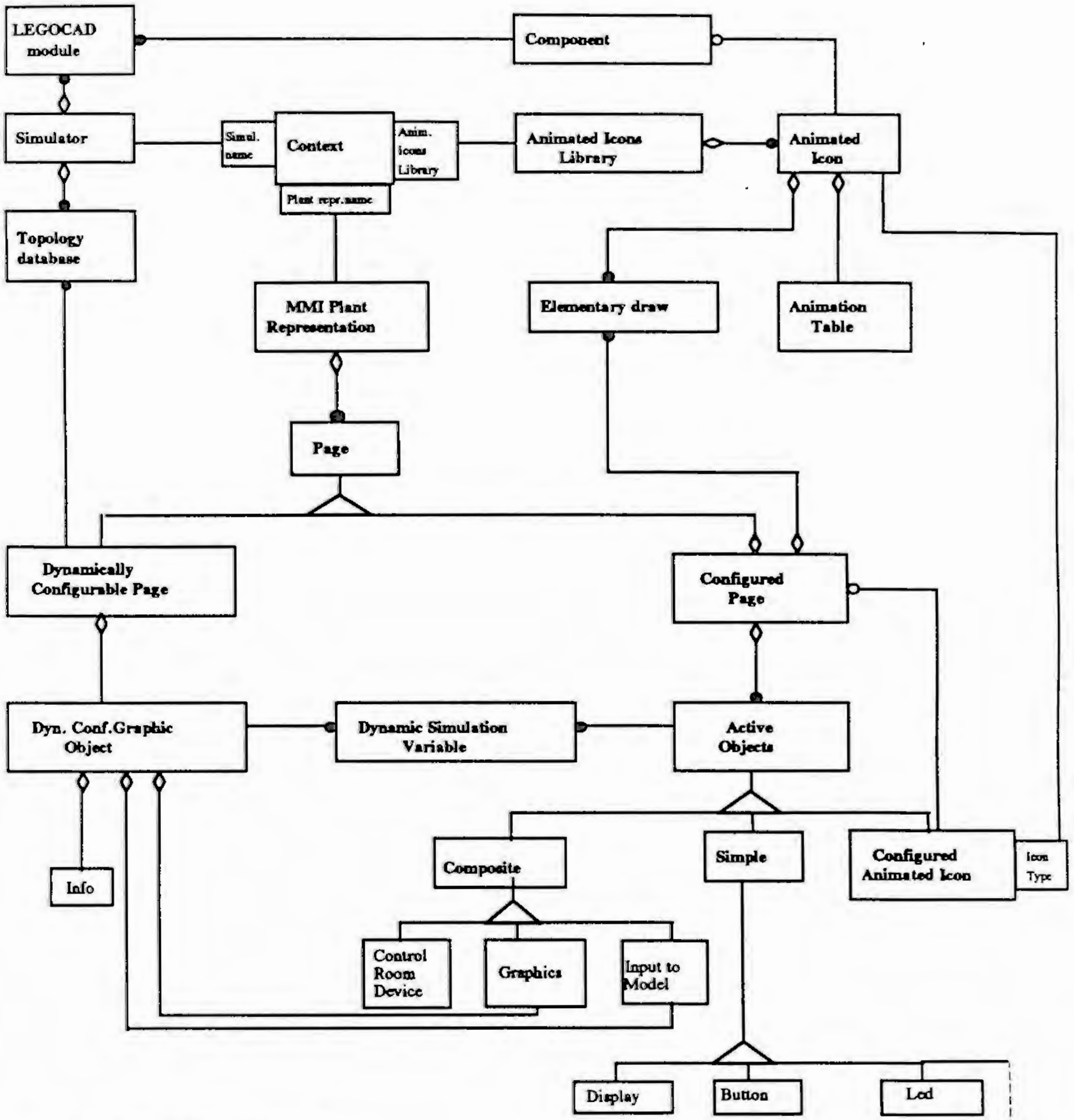


Fig.1 - Object Model for LEGOMMI

Animated Icons are configurable directly using the Attributes Configurator when editing a page or by means of the Icons Librarian. This last organizes a series of animated icons and optionally integrates them with LEGOCAD modellistic information by means of the Components Editor. Animated icons selected from the librarian may have, when associated to LEGOCAD components, additional capabilities of modellistic data-base guided configuration.

At installation phase all references to simulation variables names are solved by the Page Compiler in terms of pointers to the dynamic Simulation Topology Database. Also page links are verified.

During the simulation the system is composed by a Pagination Subsystem that activates/deactivates the requested graphic pages. It asks to the Updating System for the updating of the dynamic objects included in the visualised page and acts as a client with respect to the Run-Time Services.

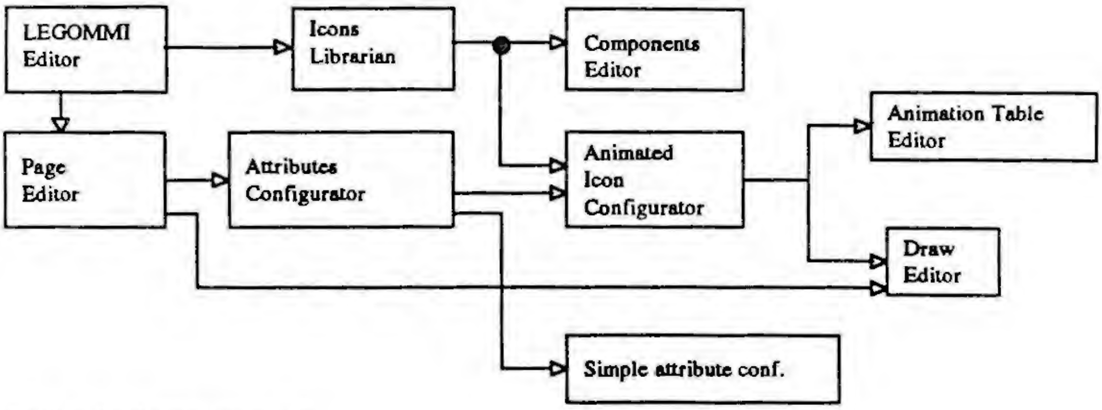


Fig.2 - Configuration phase

These services are fulfilled by means of Dynamically Configurable Graphic Objects (see previous section); they are started selecting an Active Object and through queries to the simulator topology database they are capable of interacting with the simulation variables. Finally the simulation's variables are obtained from the simulator by the Updating System (generally via an Ethernet connection).

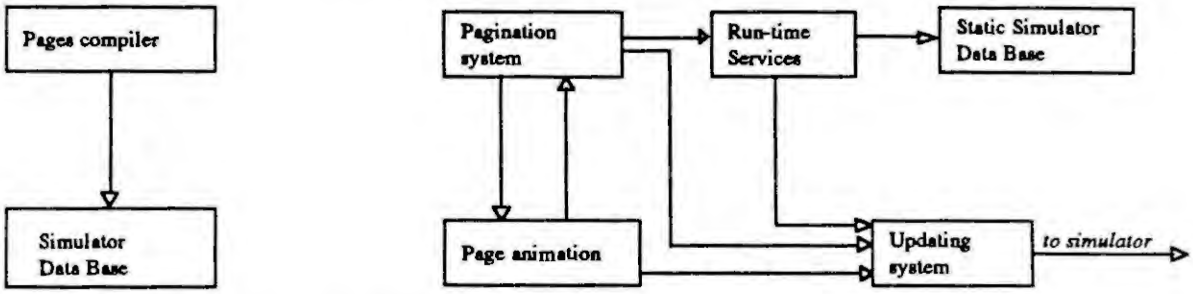


Fig.3-Installation phase (left) and simulation session (right).

6. Implementation guidelines

Architectural requirements have an important impact on implementation details in particular for what concerns the choose of the graphical environment and the requirement of portability on different operating systems. Although the use of an object oriented language for development (such as C++) should be desirable, portability requirements lead to consider more convenient at present to develop the system in C language. The choice of OSF/Motif graphic environment gives the opportunity of using an object oriented system (the Xtoolkit) for the implementation of the graphic objects. Neverheless the Motif Toolkit has a lack in widget classes specific for plant simulation (such as animated icons, graphics, bar charts) or for control room instrumentation emulation (analogue display, switches, led-button etc..). In some cases Motif widgets should also be manipulated through attribute values to obtain widgets useful for this scope. In this case the creation of new and more dedicated classes, presenting only the attributes and call-backs useful for LEGOMMI, is desirable.

Configuration of graphic objects is a matter of specifing particular user configurable widget resources. A general resources configurator requesting to a generic widget the list of its user configurable resources (listed by type of resource, description and default value) and submitting them to the user via editable lists is to be implemented. More complex resources such as animated draws can be edited by means of specialized editors.

Another important implementation choice concerns the data format and memorization policies. Data used by LEGOMMI can be classified in graphic data (drawings), object related information (attributes of objects) and LEGOCAD simulation model data (model topology data and pointers to dynamic simulation data base). The direct use of a standard relational data-base for retrieving graphical information has been evaluated too much time consuming to allow for instance a switching time between configured pages of 1 second: a better solution should be to store the graphical and object related information in data-base flat files where pages description information will be managed as complete entities.

Queries to data base during simulation are performed only by Run-Time Services for retrieving general information related to the LEGOCAD mathematical models to select variables for display or actuation. In conclusion, the relational data base is used with different granularity depending on the data types treated, as represented in table 1.

Information type	Minimum treated entity
Graphic Data	Animated page
Animated Icons Library	Icon draw Component blocks Proposed block's animation variables Colours and ranges of animation table
Model's related information	Model variable Variable attributes (alarm thresholds, var type, alarm treatments, units of measurement ..)

Page refresh is achieved by a client-server connection between LEGOMMI and an updating process located on the simulator side. Variables' pointers to dynamic simulator data base are obtained during page compilation and are available in the page description files during simulation; when the user calls a new page a message is send to the updating process containing the new variable pointers and the required refresh time. This mechanism provides the minimum load for the data transfer (via Ethernet connection) and the possibility of differentiating the refresh frequency of the control room instrumentation emulation from that of the conventional MMI pages.

7. Further developments

The use of inheritance should be useful to extend classes developed for LEGOMMI to create classes for future applications such as a graphic LEGOCAD mathematical models builder. In this case the main functionality, not needed in LEGOMMI's classes, will be the treatment of model topological information. Animated Icon object class should be provided by a series of input and output *ports*to allow a mechanism for graphic connection of different icons by means of ports. Connection between ports should give rise to changes in the data of the object port so that the topology of the whole scheme drawn by the user should be reconstructed in a further activity called compilation of a model.

The requirements briefly described should be satisfied by the implementation of a new composite object class: *icon with ports* Animated Icons attributes and methods will be reused in the simulation phase creating directly a MMI page from the same topological scheme used to produce the mathematical model.

8. Conclusions

A representation of the whole MMI system in terms of connected object (OMT object model) give a first view of dependencies between different components of the whole system and indication on the characteristics of interfaces between subsystems that may be most convenient. This leads to a decomposition of the system in subsystems that have each other client server relationships.

The two representations proposed (OMT object model and subsystem decomposition) may be considered the first step for a further detailed analysis to describe, on one side, the attributes and methods of the single objects and on the other, the functions and data types that constitute the subsystems interfaces.

Object design can start with the definition of all operations that an object may support. For graphics object polymorphism should be widely used: every operation is implemented for every single object class as a particular methods. Hierarchy and class attributes are then described in details.

Subsystem interfaces should be specified in terms of interactions requested to communicate data amoung the applications (clipboard's cut & paste, drag & drop) followed by the software mechanisms accmplishing this function. Data type format used in data passing should also be specified.

9. References

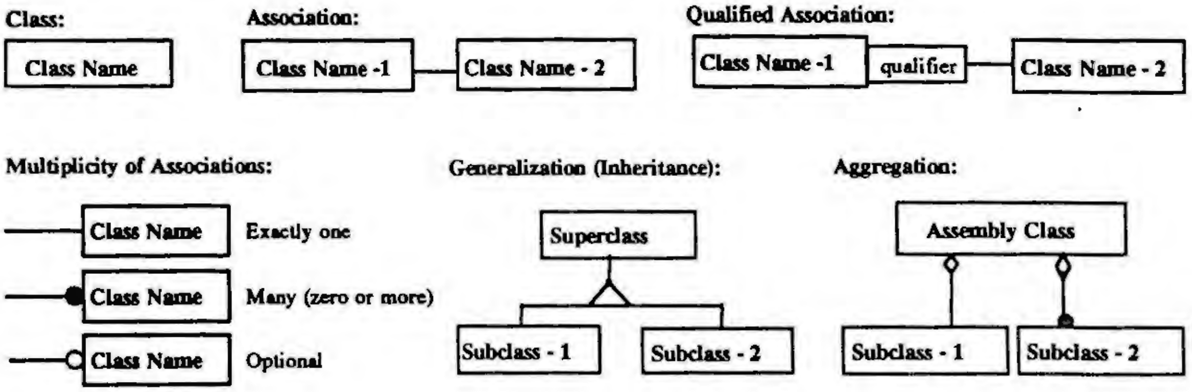
[1] R.Cori, S.Spelta, G.A.Guagliardi, F.Pretolani, P.Maltagliati, F.Persico, M.Sommani "La Workstation LEGOCAD per lo sviluppo di modelli dinamici complessi", 90 Riunione Annuale AEI, Lecce.

[2] H.A. Barker, Min Chen, P. W. Grant, C. P. Jobling, P.Townsend: "Open Architecture for Computer-Aided Control Engineering"; IEEE Control Systems, April 1993.

[3] J.Rumbaugh, M.Blaha,W.Premerlani,F.Eddy,W.Lorensen: "Object-Oriented Modeling and Design", Prentice-Hall, Inc. 1991.

[4] M.E.S.Loomis, A.V.Shah, J.E.Rumbaught.: "An object modeling tecnique for conceptual design",Lecture Notes in Computer Science, 276, Springer Verlag.

Appendix - OMT notation



Generating efficient computational procedures from declarative models

Claudio Maffezzoni
Politecnico di Milano
Dipartimento di Elettronica e Informazione
Piazza L. da Vinci 32, 20133 Milano Italy
email maffezzo@ipmel2.polimi.it

Petrika Lluka
CEFRIEL
via Emanueli 15, 20126 Milano Italy
email petrika@mailer.cefriel.it

Abstract

This paper describes a methodology and the related algorithms for the automatic generation of efficient computational procedures from models in declarative form, being the latter very suitable for a fast and easy model building. Design and implementation issues of a software tool based on object-oriented programming paradigm are also presented.

Keywords: modeling, simulation, procedural and declarative form, DAE system, symbolic manipulation, object-oriented database.

1 Introduction

Computer modeling and simulation of industrial plants very often is a complicated and time-consuming task. A drastic reduction in model development work can be obtained if a modular model building and a high degree of module reusability are allowed by the modeling and simulation environment [11, 3].

Informally speaking, building a model in a modular fashion means that the model of a physical system is built putting together the models of its components following connection rules that are independent of the modules content. There are two main approaches for the description of a modeling module. The model of a physical component can be expressed like an algorithm for computing the "outputs" of the component model when its "inputs" are given. A model of a physical component expressed in this way is said to be in a *procedural form*. Most of the existing simulation packages use this approach. But this approach gives rise to some critical problems. The most relevant is *low reusability*. The user must define the model inputs and outputs at the module level. A model module depends on the context in which it is used and can be reused only when it appears in the same context. On the other hand, the procedural approach yields a ready-to-use software for model simulation.

The model is in a *declarative form* if one software module is associated to one physical component independently of the context in which it is used [3, 12]. The software module is a set of structured data and not a coded procedure. A module specifies the behaviour of physical component using directly mass, momentum, energy balances or other first principle equations without restrictive assumptions on the possible boundary conditions. The declarative form allows a complete model reusability, because it is not necessary to specify model inputs and outputs and it guarantees a one-to-one correspondence between physical components and their software representation. Powerful software environments as object-oriented databases [14] can be used to support easy and fast model building [3]. However, this form of model representation cannot directly be used for simulation, so it is necessary to compile it, generating the procedural form [13]. This paper just addresses the problem of deriving most efficient procedural forms in an automatic way and presents a prototype software where such operations have been implemented.

2 Efficient computational procedures for model simulation

If models are written based on first principles equations, their simulation often consists in solving a system of differential/algebraic equations (DAE) [12]. A model can be simulated only if its boundary conditions[1]

[1] Boundary conditions represent the model interaction with the "rest of the world".

are specified. Such a model is called *closed*. A closed model is *consistent* if the number of DAE system equations, is equal to the number of model variables. Of course, the model consistency is only a necessary condition for the DAE system solvability.

Generally a DAE system has the form (1):

$$\mathbf{F}(t, y, \dot{y}, u, p) = 0 \tag{1}$$

where y is the unknown variables vector, u is the input variables vector, p is the parameters vector and t is the independent variable time.

Especially for complex plants, the DAE system is of very large dimensions, so, its solution would require excessively long computation times. Therefore, an order reduction of the DAE system is recommanded; this can be done by means of various expedients.

First, equations of form $y_i = \pm y_j$ can be eliminated substituting one of the variables with the other. The DAE system consistency is preserved, because for every eliminated unknown variable also an equation is eliminated.

Second, the equations introduced by boundary conditions are of the form $y_i = f(t)$. The variable y_i can be considered as a known variable. So it is possible to have a further reduction of the DAE system order.

Third, a drastic system order reduction is achieved if the DAE sytem is split into implicit equations and assignments. Let consider the unknown variables set y as two subsets, x the state variables (i.e. those variables which are present in (1) together with their derivatives) and z the algebraic variables. It is possible to rewrite (1) in the following form:

$$\mathbf{F}(t, x, \dot{x}, z) = 0 \tag{2}$$

ommitting u and p for the sake of simplicity. If any equation of system (2) can be solved with respect to some algebraic variable, say z_m, it is possible to write:

$$z_m \cdot g_1(t, \bar{x}, \dot{\bar{x}}, \bar{z}) + g_2(t, \bar{x}, \dot{\bar{x}}, \bar{z}) = 0 \tag{3}$$

where $\bar{z}$ and $\bar{x}$ are subsets of x and z respectively and $z_m \notin \bar{z}$. Once the variables of $\bar{z}$ and $\bar{x}$ are known, if $g_1(t, \bar{x}, \dot{\bar{x}}, \bar{z}) \neq 0$, the variable z_m can be computed as an assignment. Let $\tilde{z}$ be the set of those algebraic variables with respect to which no equation can be solved, and let $z_1 \ldots z_k$ be the remaining algebraic variables. Then, if possible, the DAE system (2) can be transformed into the following one:

$$\begin{aligned}
z_1 \cdot g_{11}(t, x, \dot{x}, \tilde{z}) + g_{12}(t, x, \dot{x}, \tilde{z}) &= 0 \\
z_2 \cdot g_{21}(t, x, \dot{x}, \tilde{z}, z_1) + g_{22}(t, x, \dot{x}, \tilde{z}, z_1) &= 0 \\
&\vdots \\
z_k \cdot g_{k\,1}(t, x, \dot{x}, \tilde{z}, z_1, ..z_{k-1}) + g_{k\,2}(t, x, \dot{x}, \tilde{z}, z_1, ..z_{k-1}) &= 0 \\
\tilde{\mathbf{G}}(t, \dot{x}, x, \tilde{z}, z_1, \ldots, z_k) &= 0
\end{aligned} \tag{4}$$

The problem is finding of the subset $\tilde{z}$ and the set of assignments.

The efficiency may also be improved, if it is first possible to reorder the DAE system (2) into a block lower triangular (BLT) form [12, 4], then, splitting every subsystem into assignments and implicit equations.

After substituting of variables $z_1 \ldots, z_k$ into $\tilde{G}$, the problem is reduced to the solution of the DAE system $\mathbf{G}(t, \dot{x}, x, \tilde{z}) = 0$. To do this, a function is required to compute the residual vector [12]:

$$\Delta = \mathbf{G}(t, x, \dot{x}, \tilde{z}) \tag{5}$$

when all arguments of $\mathbf{G}$ are known.

Remark: It is known [7, 12], that the theory for deciding when a DAE system of form (2) has a unique solution is incomplete. A useful trace for the solvability, is the DAE system index, that is the minimum number of times that all or part of (2) must be differentiated with respect to t in order to transform it into an explicit ODE form [7]. All known implicit DAE solvers have troubles with solving problems of index greater than one [12, 8]. In particular, it is recognized that one of principal causes for the lack of DAE system solvability is the presence of algebraic constraints on state variables.

3 Generating procedural forms from declarative models

An approach to generating efficient procedural forms that treats the majority of problems exposed in the previous section, is proposed in the rest of the paper. This approach is quite general, but it requires that a declarative model building environment be available. Such an environment, having the GemStone object-oriented database [10] as support, is realized at the center CEFRIEL in collaboration with the Politecnico di Milano [3]. All the data structures and algorithms shown in this paper are implemented in this environment, since using an object-oriented database as support, most operations that require information about model constitution, are implemented in a nice and secure way.

3.1 Model's declarative form

In tbe above mentioned model building environment, every model is represented as an aggregation of submodels connected together by means of physical ports [9] or control signals. Tbe connections representing physical ports are called *physical terminals* and the connections representing control signals are called *control terminals* [3]. It is possible also a hierarchical aggregation where every submodel can in turn be represented as aggregation of several submodels. Models that cannot be decomposed into submodels are called *simple models*. A simple model is desribed as a set of *variables*, *parameters*, *equations*, *terminals* and *variable-terminal* relationships.

If the hierarchical aggregation is reduced to a one-level aggregation, an overall plant model can be represented as a set of simple models connected together by means of physical and control terminals. Every connection between two physical terminals introduces equations of type $e_1 = e_2$ for *effort* variables and $f_1 = -f_2$ for *flow* variables [9], while a connection between two control terminals introduces an array of equations of the type $v_1 = v_2$. The equations generated by connections between models terminals are called *binding equations*. The system of equations describing the plant model is obtained merging the equations of all simple models and all binding equations.

There are two principal types of simple models: continuous-time and discrete-time. In every plant it is possible to distinguish two sets of models containing continuous-time and discrete-time models respectivly. The interaction between these two sets of models is realized only by means of control terminals, because only information (and not power) is exchanged between them. There are not physical terminals in tbe discrete-time part of the plant model.

Given the simplicity of discrete-time models simulation, only the continuous-time part of the plant model is considered in the following, according to the lines defined in the previous section.

3.2 Symbolic manipulation of model equations

The generation of efficient procedural form is based on the symbolic manipulation of the equations system representing the overall plant model. In order to make it possible, the following information is needed:

- if a variable is a state variable or an algebraic variable
- if a variable takes part in an equation
- if an equation containing an algebraic variable can be solved with respect to that variable

Moreover, the following operations on the equations must be performed:

- solving an equation with respect to one of its variables (if it is possible)
- transformation of an equation into the implicit form $g(t, y) = 0$
- substitution of a variable with another one
- recognition of forms: $y_i = \pm y_j$ and $y_i = f(t)$

3.3 Data structures for equation representation

In order to allow tbe required symbolic manipulations, an equation is described as an entity holding the data concerning its form and content and the set of allowed operations on those data. An object-oriented approach [5, 2] is adopted for the representation and storage of equations as data structures. This, for two principal reasons:

1. Object-oriented programming environments are very suitable for the declarative model building, specially, to meet modularity and reuse requirements [3]. A simple model defined as an object containing other objects, must contain also its equations. So, it is natural to define the equations as objects.

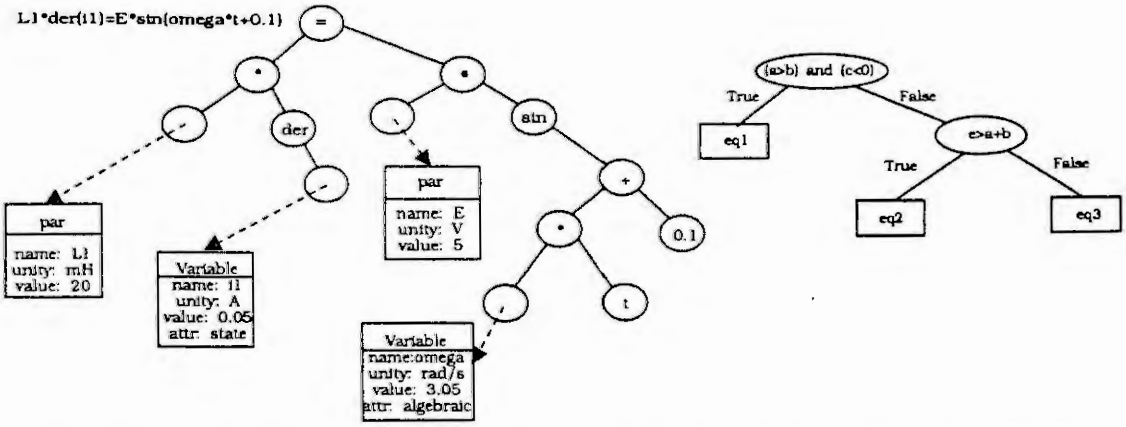


Figure 1: Binary trees representing: an equation (left) and a conditional equation (right)

2. It is of great convenience to define a class `Equation`, which can be seen as a set of encapsulated data and operations defined on them. So, the information associated to each equation can be encapsulated, while all tests and operations on it (belonging tests, solvalibity, etc.) are defined as methods applicable to equation's data.

The symbolic manipulation of equations is much easier if an equation is represented as a ***binary tree*** (see figure 1). To tree nodes that are not *leaves* there are associated atomic symbols called ***terminal symbols*** [1]. Typical terminal symbols are (=, +, -, *, /, der, sin, cos, sca, exp, ln, ...). To the leaves are associated variables, parameters or numerical constants. Atomic symbols and numerical constants are ***string*** objects, while variables and parameters are compound objects (name, unity of measure and value are some of their components).

Moreover, since every equation, variable or parameter is part of some simple model, relationships can be defined between simple models, equations, variables and parameters. A fundamental rule is established: if an equation is defined in a simple model, it must contain only variables and parameters defined in that model. The independent variable t is an exception, it takes part implicitly in every simple model (so it can be used in every equation) and it does not require a representation as a compound object.

On the other hand, an equation is created as a text string, so operations that implement conversions between the two forms of equation representation are needed. More involved is the conversion string-to-tree, because it must be preceded by a lexical analysis that splits in *tokens* [1] the equation string, passing these tokens to a *tree constructor* that builds the binary tree, providing also a parsing (syntax analysis) of the equation expression. As a consequence, specifications of regular expressions for the lexical analysis and a context-free grammar for parsing, are needed [1]. Tests about tokens that are not terminal symbols are needed, too. These tokens must be names of variables or parameters defined in the same model to which the equation belongs.

Based on the above considerations, the following structure for the class `Equation` is proposed:

```
Class:  Equation
Instance variables:
name                             instanceOf  String
equationAsString                 instanceOf  String
equationAsBinaryTree             instanceOf  BinaryTree
variablesBelonging               instanceOf  SetOfVariables
variablesInDiffFormBelonging     instanceOf  SetOfVariables
explicitableVariables            instanceOf  SetOfVariables
Class variables:
symbols                                      {=, +, -, *, /}
functions                                    {der,sin,cos,exp,...}
```

A special case in the description of equations defined in a simple model, is that of the so-called *conditional equations* which are equations taking on different forms in different conditions specified via `if-then-else`

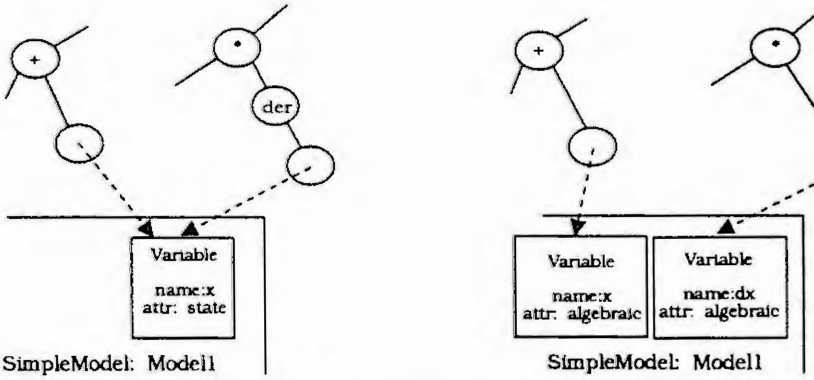


Figure 2: Data structure transformation while treating an algebraic constraint

constructs. A typical case is the interaction of some moving object (for example a robotic arm) with a stiff surface. The equations describing forces exerted on the object during the free motion are different from the equations describing forces during the interaction. Generally, in such a case, an equation form is associated to a boolean expression. If the boolean expression returns **True**, the corresponding form is selected as the current one. Considering boolean expressions as nodes and equation forms as leaves, the binary tree structure can be used also for the representation of conditional equations (see figure 1).

3.4 Preprocessing of the DAE system

The first step toward DAE order reduction is the elimination of equations of form $y_1 = \pm y_2$ (binding equations are among them). Such equations are eliminated substituting one of variables with the other. The substitution of variables consists in moving all leaf pointers (see figure 1) from the substituted variable to the substituting variable.

Algebraic constraints on state variables are then treated. An algebraic constraint on state variables causes a reduction by one of system's degrees of freedom. In fact, it defines one of the state variables as function of only other state variables. Such variable has to be replaced by a pair of two algebraic variables, the variable itself and its derivative. As a consequence, the DAE system lacks its consistency. To obtain consistency, an equation is added differentiating the algebraic constraint with respect to time .

The above procedure can be implemented by suitable expedient in the modular model formulation. Based on information associated to each equation, it is possible to identify equations that contain only state variables not in differential form[2]. To this aim, in a model holding an algebraic equation which potentially may give rise to algebraic constraints on state variables, also the differentiated form of that equation is added. Than, when building the closed model, one of the state variables taking part in the algebraic constraint is substituted by two algebraic variables (see figure 2).

3.5 Splitting into equations and assignments

In this section, an algorithm for the trasformation of the form (2) in the form (4) using symbolic manipulation of equations is presented. This algorithm is executed after the above preprocessing of the DAE system and operates on the following sets of objects:

- set of equations to be scanned: **EQS**
- set of unknown variables: **VARS**
- set of variables to be computed by the DAE solver: **SVARS**
- set of equations left in implicit form: **SEQS**
- set of assignments: **ASS**

Initialization: The sets are initialized as follows: **SEQS** = set of conditional equations, **EQS** = set of overall model equations - **SEQS**, **VARS** = set of overall model variables, **SVARS**=$\emptyset$, and **ASS**=$\emptyset$.

[2] Hidden algebraic constraints may occur, also. For example, an algebraic constraint between x_1, x_2 and x_3 is defined by the form $f_1(z_1, z_2) = 0$; $z_1 = f_2(x_1, z_3)$; $z_2 = f_3(x_2)$; $z_3 = f_4(x_3)$. Hidden algebraic constraints are not treated in the practical implementation.

Step 1: Search for the variables that cannot be computed as assignments, i.e. the algebraic variables with respect to which no equation $\in$ **EQS** can be solved, and the state variables. These variables are moved from **VARS** to **SVARS**.

Step 2: The following iterations are repeated until **VARS**=$\emptyset$ or **EQS**=$\emptyset$ or substeps a) and b) are unsuccessful:

a - Scan **EQS** until equations of form $f(z) = 0$ where z algebraic variable is the only unknown variable (z $\in$**VARS**), are found. If the equation is solvable with respect to z, then it is solved, and moved in **ASS**. The variable z is cancelled from **VARS**.

b - Scan **EQS** until equations of form $f(z, y) = 0$ where z and y are unkown algebraic variables ($z, y \in$ **VARS**), are found. If the equation is solvable with respect to one of variables, say z, then it is solved, and moved in **ASS**. The variable z is cancelled from **VARS**, while the variable y is moved in **SVARS**.

Step 3: The remained variables are moved from **VARS** to **SVARS**, and the remained equations are moved from **EQS** to **SEQS**.

The sets **SEQS** and **ASS** obtained from the algorithm are said *split form* of the system, while the algorithm is said *split algorithm.* Note that the equation data structure contains all the necessary information required for the construction and updating of all sets. This information is largely used in all steps of the split algorithm.

3.6 Manipulating vector equations

Some application domains require vector variables and equations, also. This is the case of 3D modeling and simulation, very useful in robotic applications. There are not conceptual differences with the scalar case, since the data structures for equation representation does not depend on the variables type. The processing of vector equations consists in the following sequence of operations:

- After the preprocessing of the DAE system, all vector equations are collected in a set and the algorithm for splitting into implicit equations and assignments is applied on it.
- Every vector equation is expanded into scalar ones. Then, the scalar equations obtained by implicit vector equations expansion are merged with the rest of model equations giving the **EQS** set, while the scalar equations obtained by vector assignments expansion are put in the **ASS** set.
- Finally, the split algorithm for the scalar equations is applied.

3.7 Coding the efficient procedural form

If a DAE solver is chosen, a software module that computes the residuals vector is needed. Thus, the set of equations and assignments (as data structures) representing the efficient procedural form must be transformed in a source code which can be compiled and linked together with solver modules.

Generally, the residuals computing module is required in the form of a procedure or a function to which the vector of variables computed by solver y, the vector of their derivatives ypr, and the vector of parameters par are passed(this is the case of DASSLRT [7], the adopted solver, but few variations from this scheme are possible). The variables computed as assignments are not members of vector y, so it is necessary to pass them as part of par vector. As a consequence, the par vector contains plant parameters, control variables[3] and variables computed as assignments. Algebraic variables are passed to the solver as derivatives, because this is the way to allow step variations of these variables.

In an object-oriented programming environment, the correspondences between `Variable` objects and y, ypr and par vectors components can be represented as an instance of the `Dictionary` class, that is a non ordered collection containing associations between *keys* and *values* [5, 15]. In the *dictionary of correspondences* a key is a string representing one element of y, ypr or par vectors and a value is an object that is an instance of classes `Variable` or `Parameter`. Such instances belong to various plant simple models that are instances of the class `Model`.

In the practical implementation [3] the module computing the residuals vector is generated as a C language function, given the high machine-independence of C code. Every equation tree is converted to a `string` form putting at every leaf the variable or parameter key found in the *dictionary of correspondences.*

[3] Are variables computed by the discrete part of the plant

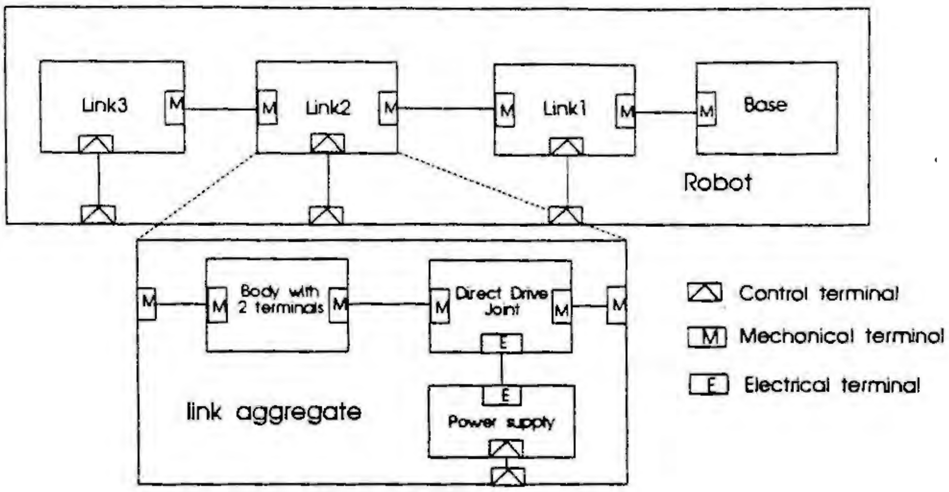


Figure 3: Schematic represention of 3-links robot model

The role of *dictionary of correspondences* is crucial because it allows not only the generation of the simulation code, but also the association between simulation results and plant variables.

Note that in the C function that computes the residuals vector there may appear built-in functions. Some of them as `sin()`, `exp()`, etc. can be part of the C language mathematical functions library, while other functions as `step()`, `ramp()`, etc. must be included in some user-built library.

4 An example

An application to the robotic modeling and simulation is reported. In figure 3 the model of a three-link direct-drive robot is shown. There are two types of physical terminals, mechanical terminals that export linear and angular positions, velocities and accelerations, forces and torques, and electrical terminals that export voltage and current. The robot is composed by three aggregate models *link1*, *link2*, *link3* and a simple model *base*. A link is an aggregate of a rigid body, a Direct Drive Joint (DDJ), i.e. a revolute joint equipped with a dc motor, and a voltage-controlled power supply. The rigid body is described by 13 vector equations and the DDJ by 9 vector equations and 15 scalar equations. After the preprocessing, i.e. the elimination of equations of form $y_i = \pm y_j$, the overall DAE system reduces to 117 scalar equations, while its split form reduces to 12 implicit equations. The split form corresponds to the Newton-Euler equations of mechanical manipulators [6]. The simulation has been executed on a DECStation 5000/200. The simulation times are shown in the following table:

DAE system	simulated time	CPU time
Not split	100 seconds	1500 seconds
Split	100 seconds	14 seconds

The drastic reduction of the computing time is due to the DAE system order reduction.

5 Conclusions and future directions

The declarative approach is very useful in model building for simulation aims. To simulate declarative models, a plant model compilation is needed in order to generate its procedural form. The generation of procedural form can be done taking into account its efficiency in therms of simulation execution times and numerical robustness. A drastic order reduction of DAE system is achieved if binding equations are eliminated and the remaining system is split into assignments and equations. This is realized by means of symbolic manipulation of equations. An object-oriented approach in model variables and equations representation has resulted of a great benefit in designing and implementing their symbolic manipulation. A software package that allows to generate efficient procedural forms from declarative models and embedded into the modular model building environment [3] has been developed in the GemStone object-oriented database. Further extensions of software package will implement the possibility of the DAE system partitioning into several subsytems.

References

[1] Aho A.V., Sethi R., and Ullman J. V. *Compilers: Principles, Techniques and Tools.* Addison-Wesley, 1986.

[2] Mayer B. *Object-Oriented Software Construction.* Prentice Hall, 1988.

[3] Bellasio F., Benvenuti A., Groppelli P., Lluka P., and Maffezzoni C. A modular simulation environment based on object-oriented database technology. *Proceedings of Second European Control Conference,* June 1993.

[4] Elmqvist H. *A Structured Model Language for Large Continous Systems.* Sigma-Tryck, Lund, Sweden, 1978.

[5] Cox J.B. *Object-Oriented Programming: An Evolutionary Approach.* Addison-Wesley, 1986.

[6] Luh J.U.S., Paul R.C.P., and Walker M.W. On-line computation scheme for mechanical manipulators. *ASME, J.on Dynamic Systems, Measurement and Control,* 102(2), 1980.

[7] Brenan K.E., Campbell S.L., and Petzold L.R. *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations.* North-Holland, 1989.

[8] Petzold L.R. Differential-algebraic systems are not ODE's. *SIAM Journal on Scientific and Statistical Computing,* September 1982.

[9] Wellstead P.E. *Physical System Modeling.* Accademic Press, 1979.

[10] Bretl R., Maier D., Otis A., Penney J., Schuchardt B., Stein J., Williams E. H., and Williams M. The GemStone data management system. In Kim W. and Lochovsky H., editors, *Object-Oriented Concepts, Databases and Applications.* Addison-Wesley, 1989.

[11] Mattsson S.E. On model structuring concepts. *4th IFAC Symposium on CAD in Control Systems,* August 1988.

[12] Mattsson S.E. On modeling and differential-algebraic systems. *Simulation,* January 1989.

[13] Mattsson S.E., Andersson M., and Åström K.J. Object-Oriented Modeling and Simulation. In Linkens D.A., editor, *CAD for Control Systems.* Marcel Dekker, Inc., 1993.

[14] Kim W. *Introduction to Object-Oriented Databases.* The MIT Press, 1992.

[15] LaLonde W.R. and Pugh J.R. *Inside Smalltalk.* Prentice-Hall, 1990.

EXPLOITING THE POWER OF MODELLING IN LOGISTICS NETWORK ENGINEERING (LNE)

Jyrki Ingman
Veli-Pekka Mattila
Antti Hovi

Laboratory of Electrical and Automation Engineering, Technical Research Centre of Finland
Otakaari 7 B, FIN-02150 Espoo, Finland

Abstract

Logistics Network Engineering (LNE) is a discipline connected with the Supply Chain Management approach. Modelling is a fundamental element in analysis, design and re-design of value chains and networks. Logistics networks are usually too large to be modelled in detail. With modelling elements and a hierarchical approach the task becomes manageable. Exploiting modelling and simulation in re-structuring logistics networks helps a company concentrate on essential problems and with that achieve competitive advantage through logistics.

1. BACKGROUND

During the past decades manufacturing research focused on productivity and efficiency of individual manufacturing operations. As companies managed to squeeze down set up times etc. the hidden profit potential in the material flow at the factory floor level was discovered. Then the focus was shifted towards achieving continuous material flow within the factory walls. The scope was also expanded to include the first line supply and demand customers.

Constant new pressures and rapid changes in the business environment drive companies to set their focus on improving total value chains and logistics networks instead of re-engineering individual business operations. World class performance at company level does not mean that the company achieves competitive advantage automatically. Its survival depends more or less on the ability of the total value chain to cost effectively meet the requirements of the end customers. Good and effective performance can easily be wasted along the value chain. Therefore, the companies can not afford to be involved with an ineffective value chain.

In the 1990's the research scope is set one step further - total business logistics. This is also the focus of the logistics research programme of VTT, Technical Research Centre of Finland [2]. The total volume of the project is 50 man years including work done by participating companies representing the major areas of industry and trade. The term total business logistics or integrated logistics has various definitions and VTT has adopted the holistic definition presented by the European Logistics Association ELA:

> The organization, planning, control and execution of the goods flow from development and purchasing through production and distribution to the final customer in order to satisfy the requirements of the market at minimum costs and capital use.

2. LOGISTICS NETWORK ENGINEERING

Logistics networks are large and complex. Supply Chain Management directs the scope to the importance of striving towards a global optimum instead of towards a group of local optimas. It helps the companies cooperatively improve the total performance of supply chains and logistics networks. One should first focus on doing the right things before doing things right. The hierarchial top-down approach guides the re-designing process so that the data involved remains manageable and resources are used on the essentials. Without re-structuring the network first the risk of solving problems that should not exist in the first place is huge. Improving the performance of the existing network could easily lead to a local optimum in spite of trying to avoid it. There is no optimal structure valid for all the different scenarios. Instead of seeking an optimal network design one should seek a design that is the best possible solution in relation to the current knowledge and estimate concerning the future.

The steps required for gaining continuous improvement of value chains and logistics networks are the following

1. selection of the structure that within a foreseen future will guarantee the best performance as well as the desired flexibility of the logistics network,
2. elimination of the non-value-adding operations to the extent possible without worsening the flexibility and
3. improvement of the operations of individual business processes connected to the network.

The logistics concept is characterized by analysis and synthesis of systems, process and cross functionality. Pfohl [3] says that all the logistics activities together form a system that is a coherent set of interacting elements or variables. The logistics network flow is often described with its three components: the physical material flow, the information flow and the capital flow. Actually this separation is only conceptual and the components are just different sides of the same triangle. They cannot be treated separately. Methods and tools for all of them must be developed simultaneously. The same applies to the different time horizons, the basic solutions for planning and modelling at the strategic level and at the operational level must be uniform.

The rapidly increasing importance of logistics networks, and the development of the methods and tools for logistics organization and planning make the introduction of a new concept of Logistics Network Engineering (LNE) appropriate. The same progress took place when Software Design matured into Software Engineering. Savolainen and Mattila [4] define the LNE concept as follows:

> *Logistics Network Engineering LNE* is the discipline for organization and planning of the structure of the goods and information flow from development and purchasing through production and distribution to the final customer in order to satisfy the requirements of the market at minimum cost and capital use.

The charter of the discipline of LNE is to provide the methods and tools to be used by logistics network engineers to organize and plan the goods and information flows. The aim is to assist continuous improvement of the logistics networks both from the structural and the operational point of view. LNE can be seen as an extension to the concurrent engineering concept. One does not only have to design a new product at the same time with planning of the production process, the logistics network bas to be designed simultaneously as well. The general principle of LNE concept is not to emphasize individual variables but rather on how they interact as a whole.

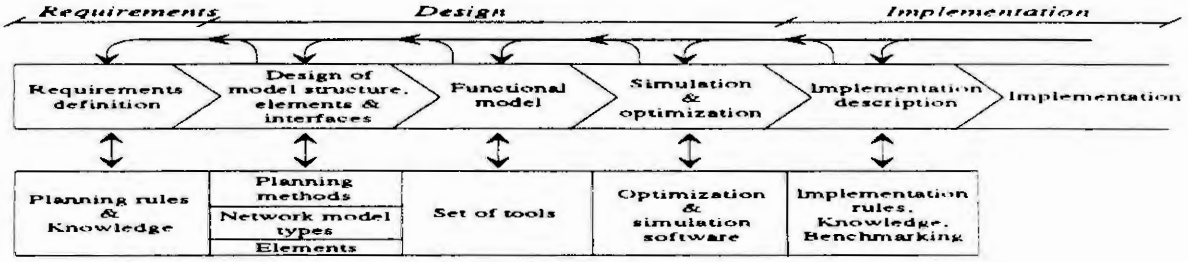

Requirements
Design
Implementation
Requirements definition
Design of model structure, elements & interfaces
Functional model
Simulation & optimization
Implementation description
Implementation
Planning rules & Knowledge
Planning methods
Network model types
Elements
Set of tools
Optimization & simulation software
Implementation rules, Knowledge, Benchmarking

Figure 1. The area of Logistics Network Engineering.

3. MODELLING OF LOGISTICS NETWORKS

Modelling is a key element in analyzing and designing complicated systems like logistic networks. Different types of models are required in understanding the behaviour of existing logistic networks and in improving their performance as well as creating new logistic networks. Modelling technology is normally used for analyzing systems and for estimation and optimization of alternative solutions. The models may also be used as an instrument to check whether the requirements are being met or to define the requirements to be met.

In the field of logistics, modelling has been applied to individual operations. Unfortunately major effort has been wasted in solving problems that are of minor importance (or totally avoidable) from the point of view of tbe total logistics chain. Most of the modelling of logistics operations has been company-oriented. In many cases detailed modelling has resulted in precise solutions but in a highly constrained solution space. Design of the model has usually been so big an effort that the real utilization including re-design and updating of the model has been neglected.

Modelling of logistics networks has some special features. The networks are large and it is impossible to get detailed data from every organization or from every operation. Design and re-design of the structure cannot only be based on existing companies and other operational units. Also units not belonging to the existing value chains must be included whenever needed. Furthermore elements without any connections to real life are useful tools when striving for an optimal network structure.

4. HIERARCHICAL ELEMENT-BASED STRUCTURE

All the elements of a logistics network cannot be modelled in the same level of detail. The elements along the current critical path in relation to logistics performance measurements are the most important. The critical path can also move due to the changes in demand or due to the re-design of the network. In that case new details must be introduced and the modelling structure has to support this.

The hierarchical element-based modelling structure leads into an object-oriented approach. When taken into the model an element must be assigned an *identifier* that will be used during the planning phases. It has also *attributes* that describe the properties (capacity, lead time etc.). The interaction between elements consists of *service requests* and *services*. The element is capable of carrying out *internal actions* (manufacturing, transportation, etc.) that can be requested by other elements and it may need internal actions of other elements in order to fulfil the task. For example, an element representing warehouse receives a request (order) that initiates the appropriate internal action (picking) and a service request (transportation). [1]

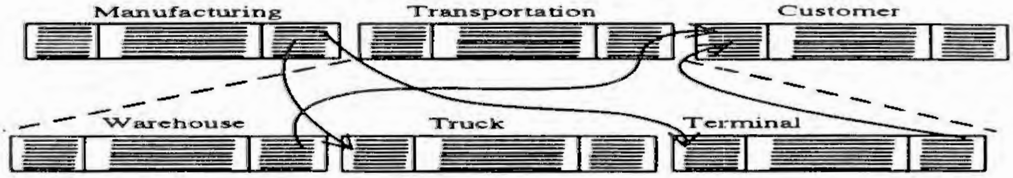


Figure 2. Modelling hierarchy.

5. SIMULATION AND CALCULATIONS

Looking from the point of view of the logistics flow the functions of an element can be divided into three categories: *receiving, processing and dispatching*. The performance of a network is measured by time related, cost related and customer service related factors. Simulation of a logistics network is controlled with demand patterns for various products and product types. An order is sent to an element "connected" to the end customer and that creates additional orders. A *service request* always creates demand for *service*. For example, if an element like Warehouse (W) receives an order (service request sr1) from customer C it first checks whether it has the requested amount of the ordered product. If the answer is yes, a delivery (service s1) is made and if not, the W has to make an order (service request sr2) to its Supplier (X). After X has fulfilled sr2 by delivering s2 to W, then W can respond to sr1 by delivering s1. (see figure 3)

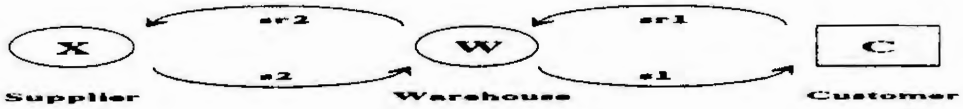


Figure 3. Interaction between elements via service request and service.

Although the modelling can show that the problems within a network are caused by an individual element the real opportunities for improvement are hidden in the interaction of the elements. The interface between two elements can contain operations needed only for matching of the dispatching and receiving operations. There are usually delays between elements due to lack of balance. The key question of LNE is to find the network structure that in the best way meets the requirements of the end customers. The non-value-adding operations must be separated from the value-adding ones in order to find the critical points of the network. For the calculation three main performance indicators have been defined: *element burden (EB)*, *interface burden (IB)* and *flow burden (FB)*. The element burden consists of all the appropriate factors connected with receiving, processing, dispatching and waiting. The interface burden consists of the factors describing, receiving and dispatching. The flow burden consists of all the delays caused by timing and balancing of the flow. Equations 1 to 3 describe the main structure of the indicators.

$$EB_j = \left[\sum_i rf_{ji} + pf_j + wf_j + \sum_k dc_{jk} \right] \tag{1}$$

$$IB_{ij} = [\ rf_{ij} + df_{ji}\ ] \tag{2}$$

$$FB_j = [\ wf_j\ ] \tag{3}$$

rf represents any of the time, cost or customer service related factors when receiving, *pf* any of the time, cost or customer service related factors when processing, *df* any of the time, cost or customer service related factors when dispatching and *wf* any of the time, cost or customer service related factors when waiting. *I* represents the sending element, *j* the element in focus and *k* the receiving element. Note that all the indicators are vectors.

6. IMPLEMENTATION

This paper is related to the research project on business logistics in different branches of industry in Finland. The project focuses on development of decision support systems both for strategic and operational use, modelling and simulation of logistic operations and development of technical solutions for the physical material flow along the whole logistics chain. Modelling elements have been developed in close co-operation with the companies. The elements build up an object library for the hierarchical modelling system and software based on an object-oriented process simulator software APROS (Advanced Process Simulator) by VTT.

7. REFERENCES

[1] Hovi, A. and Mattila, V.-P., Object-oriented Modelling for Logistics Network Engineering, DIISM '93 Workshop Proceedings, (1993)

[2] Mattila, V.-P., et al, Development of Research in Logistics (in Finnish), (Technical Research Centre of Finland 1991) 80 pp. ISBN 951-38-3869-2

[3] Pfohl, H.-C., Logistiksysteme: Betribswirtshaftliche Grundlagen. Berlin 1990

[4] Savolainen, T. and Mattila, V.-P., Models Required for Logistics Network Engineering (LNE), 3rd International Conference FAIM 93, (1993)

DYNAMIC SIMULATION OF OFFSHORE PROCESSES

Jørgen Nielsen, Steinar Sælid & Hilde Lien
Norsk Hydro a.s
Postbox 200, Stabekk, Norway

Norsk Hydro has performed a dynamic study of an offshore separation process with the aim of verifying the design's ability to handle process disturbances, -slugs. The modelling and the simulation study was carried out during the Engineering phase of the project by the use of the CADAS simulation tool. CADAS is being developed in joint effort between Simrad Albatross, SINTEF, Norsk Hydro, Statoil, Aker and Kværner Engineering.

1. INTRODUCTION

Norsk Hydro operates several platforms in the Norwegian sector of the North Sea. The newest offshore platform, to be operated by Norsk Hydro when commissioned, will be a floating concrete construction with a complete oil/gas process installed. The production will exclusively be from a number of sub-sea production wells some 350 meters below the surface of the North Sea. The connection between the sub-sea well-head clusters and the floating production platform will be established via flexible risers. The geometry of well-stream risers and the choke valves may introduce dynamic production upsets, known as terrain slugs in the offshore terminology.

The modelling of the Platform's separation train has been executed as a multi-discipline project where personnel from both the Control Engineering and the Process Technology departments in Norsk Hydro has been involved.

A detailed model of the separation process has been established. The model includes wells, risers, manifolds, pipes, control valves, measurement elements, controllers, water separating cyclones, two & three phase separators, an electrostatic coalescer, heat exchangers, oil export pumps and an export pipeline to the shore. Correlations have been established for all relevant fluid properties and equilibrium quantities.

Controller settings have been optimized to handle variations in flow with a minimum negative impact on process performance. Further a controller module implementing fuzzy logic for multi variable control structures has been developed. A comparison between PID and FUZZY control has been made.

2. CADAS

CADAS (Computer Aided Design, Analysis and Synthesis of industrial processes) is a software product for simulation and analysis of process plants and the associated control and logic structures of the plant.

Construction of chemical processes involves a variety of engineering disciplines.

Examples are the process and control engineering disciplines. In large projects, such as the realisation of oil production plants in the North Sea, several distinct groups work in sequence on their respective tasks in order to complete the plant. Upon completion of a sub-task, the groups are demobilised while the results are used further downstream in the project. Consequently, information flows from conception to start-up of the plant and rarely the opposite way. This means that every task should be correctly solved the first time.

With this in mind, it is important to invest an optimal effort in analysing the process- and control design early in the project. It is much cheaper to find a design flaw at an early stage, than during the commissioning and start-up phase.

2.1 The module concept

The module is the basic building block for configuration of a CADAS simulator. A module may be a control element like a controller or a logic module, it may be a small model of a process unit like a pump, or it may be a model of a complex process unit like a distillation column or a fluid catalytic cracker.

The CADAS user interface is based on the process flow sheet (PFS). This is a drawing of the plant and the control and logic system on the computer screen. The total PFS in defined as a large virtual drawing. The user can view parts, or the whole, of this drawing on the screen at any time. The building blocks of a PFS are icons representing modules, and graphical lines representing pipes or signal interconnections between modules. The user interacts with CADAS using menus, a mouse and keyboard input. Addressing modules in the PFS is done by clicking at them with the mouse.

The PFS is a true mapping of the real CADAS configuration. Each module in CADAS is represented by a corresponding icon in the PFS.

The configuration of a simulator is done by selecting modules from a library, and putting an instance of the module (actually the module icon) into the PFS. When this is done, a copy of the modules data structure is automatically created in the CADAS database. If the user makes a graphic connection between the terminals of two modules in the PFS, CADAS automatically creates a reference between the modules so that automatic data transfer between the modules is done when the simulator is running.

3. SYSTEM CONTROL DIAGRAMS

Norsk Hydro has developed a high level, graphical descriptive language for design and documentation of process control systems. The basic building block of the System Control Diagrams, SCD, is the function block module, which comprises process modules as well as a range of control and logic function blocks.

Many process control systems are configurated by the use of IEC standard functions blocks. For this reason the selection of function blocks or modules as a basic unit in the simulation language CADAS was preferred. The benefits are several. It provides a graphical interface for the designer/operator of the module, while in addition engineers familiar with IEC programming of process control systems needs a minimum of training in order to use CADAS as a design tool.

4. THE SEPARATION PROCESS

The purpose of the separation system is to separate the incoming well fluid into a stabilized oil product for export water for cleaning and disposal and gas for injection or export. The separation is carried out in three stages, with an inter stage heater between the first and second separator stage.

In the first stage separator, the pressure is controlled by the speed of the downstream high pressure compressor train. Free water and most of the gas is separated out in the first stage separator. The flow of oily water is regulated on interface level control and is routed to the hydro cyclone package manifold for oily water treatment, the control valve being located downstream of the hydro cyclone. Emulsified oil with a water content possibly up to 40 (vol) % is heated before entering the second stage separator. The fluid is heated in order to break the water/oil emulsion and to meet the oil export vapour pressure. Level control in the first stage separator is done by control of the oil outlet upstream of the oil heater and water flow through the hydro cyclones.

The second stage separator is a three phase separator located downstream of the oil heater. The pressure in the separator is controlled by means of a control valve in the gas outlet. The water phase is routed to the manifold of the hydro cyclone package for oily water treatment, while the oil phase is routed to the third stage separator. Level control in the second stage separator is through control of the oil outlet.

The third stage separator is a two phase separator operating at a low pressure and a temperature determined by the inter stage heater. The gas is routed to the first stage compressor and the liquid flows to the electrostatic coalescer by gravity. The pressure is controlled by modulation of the speed of the recompression train. The level is controlled by modulating the speed of the oil export pumps.

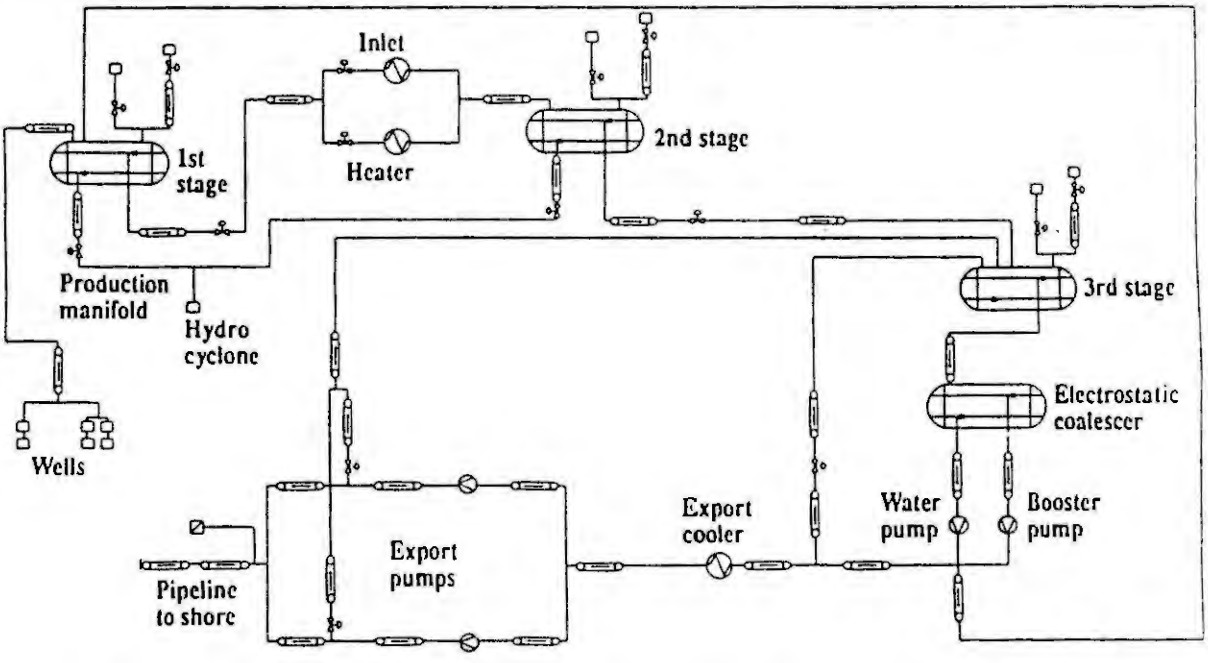


Figure1: A CADAS Process Flowsheet of the separation process.

Separation is a single train operation, hence any shutdown of a separator will cause a total process shutdown. Following the third stage separator, the oil is finally dehydrated in an electrostatic coalescer. The coalescer is designed to remove the remaining water from the produced oil to the specification level. The water separated out in the coalescer is pumped back to the first stage separator by the produced water pump, whose speed is controlled by the coalescer interface controller.

The oil export system comprises the oil booster pumps, the oil coolers and the oil export pumps. The export oil is then transferred to shore in an export pipeline. A CADAS PFD of the process is shown in the following figure.

5. PROCESS SIMULATION

Three different automatic control algorithms have been applied during the simulation of the oil separation process, traditional PID controllers, FUZZY rule based controllers and compensation of process time delay by application of Smith compensators. All three types are defined as function blocks in CADAS, with input terminals for connection to transmitter function blocks and output terminals for connection to the control valves. The purpose of simulating the process dynamics with PID versus FUZZY controllers was to evaluate the overall system performance against severe process disturbances.

A CADAS FUZZY controller was designed to handle level and pressure constraints and the multivariable nature of the process. Using this controller with 4 inputs and 3 outputs as a replacement for the Cascade PID controller for the liquid level, a substantial increase in performance was obtained. The primary process upsets to an oil separation train are slugs consisting of gas and liquid. In a multiphase transport system, pockets of gas will frequently occur in the well stream. In severe cases this will lead to trip of the process due to excessive level or pressure in the separation tanks. The multivariable FUZZY controller demonstrated during the simulation a 50 percent better capability to handle slugs compared with the PID cascade controller.

A large amount of thermal energy is often needed in order to increase the temperature of the wellstream between first and second separation tank. In order treduce CO_2 emission from the production platform, waste heat for this purpose is recovered from the gas turbines, which drives the electrical generators on board. In this system a relative large dead time occurs in the temperature control loop relative to the time constant of the process. In the CADAS model of this waste heat recovery system a comparison was made between tradition PID control and the combination of a Smith compensator and a PID controller. The task of the Smith predictor module is to compensate for the time delay. By applying a Smith predictor the gain of the PID controller can be increased beyond what is possible for a non compensated control loop. The simulation showed that an increase of 100 percent for the controller gain is possible, when a Smith predictor is used. This provides a far better control of the temperature, without sacrificing the stability of the process.

EXPERINENCE ON MECHANISTIC MODELLING OF INDUSTRIAL PROCESS DYNAMICS WITH APROS

K. JUSLIN

Technical Research Centre of Finland (VTT),
P.O.Box 34, SF-02151 Espoo

Abstract. Dynamic simulation involving thermohydralics of fluids of multiple components in several phases has been considered as a task for specialists in mathematics and computer programming. A high level model development tool, based on specification of physical mechanisms, not equations, is described. We have experienced that even a process engineer can build up a dynamic model of a large process with this tool. The code is optimized for modern workstation computers, enabling real-time simulation studies.

1. INTRODUCTION

As industrial process dynamics have become extremely complex, it has become more and more important to understand how they work and react in very variable conditions. The traditional methods of calculation, experiment and measurement are no longer sufficient nor economic; they need to be supplemented or replaced by computer-based simulations. The complexities of real-life include scheduled and unscheduled interruptions even to basically steady-state processes. Dynamic simulation is needed to complement the steady state calculations.

The Advanced Simulation Environment (APROS), has made it affordable to construct computerized models of complete process plants, including automation systems and electrical power distribution [1]. The systematic model specification, also producing the documentation needed, is made very easily by operating with the mouse on the workstation screen, choosing unit operation symbols and filling in query forms. The user doesn't have to write equations or be familiar with computer programming.

All information on process parameters and flowsheet connections, is stored into the APROS object oriented real-time database. The user can define new unit operations by combining physical mechanisms with the APROS process specification language, or by the graphics interface. He can also combine unit operations to subprocesses and make his own symbols. APROS is a computer-aided engineering (CAE) tool for the assistance of designers in the gaining of full knowledge and understanding of the dynamics of both small- and large-scale chemical processes. Very fast and efficient solution methods have been developed, enabling the designer to use his own workstation for the simulation runs. The system operates very interactively. The user can start a simulation run, and include a new unit operation into the flowsheet, even without stopping the simulation.

2. MODEL SPECIFICATION

The user may specify the simulation experiment, including process models and automation system models, using the APROS Specification Language on an alphanumeric terminal. The specification may concern control of the simulation experiment, or the unit operations on Technological level or on Mechanistic level. On a graphical work station the specification can be made using the mouse, simply combining symbols on the screen and filling in query forms.

A synthesis tool for automatic construction of common unit operation models has been included. Based on easily available user specified performance criteria on Technological level it generates elementary process structures on Mechanistic level, which provide data for the equation solvers involved.

The user can also build up a library of own specific unit operation models by combining suitable elements from the elementary process structures library, without computer programming, or knowledge of differential equations or numerical solution methods. Further, the user can combine unit operations to sub-processes, and combine sub-processes to larger entities, as well.

The elementary flow structures are composed of control volumes, and streams representing flows between the control volumes. Streams are at present available for different accuracy levels, as single phase flow, homogeneous mixture of possible gas or liquid phases, or separated phase flow enabling different velocities and temperatures for the fhases involved. The direction of the flow can change during a simulation run. The control volumes are either of phase separating or mixing type. The possible continuation of moment flux through a control volume can be specified. The heat structures can contain several layers of different materials and can be shaped as a plane, a cylinder or a sphere. Elementary heat sources, automation system elements, pumps and valves can be connected to the flow structures.

A separate set of interconnected flow structures forms a flow department. For each isolated flow department the user can specify the physical properties estimation method used, as well as the expected flow components. Chemical reactions can be specified as well.

3. INTERACTIVE SIMULATION

When analyzing the modelled process with the simulator, the user can interactively change the flowsheet structure or the parameters of the unit operation models, and continue with the simulation, all in the same session, without recompiling and linking procedures. The total number of specific process components need not be specified in advance. The real time database used is automatically reorganized for optimal operation. The calculated results can be displayed in graphical form on-line during the simulation, and can be compared with results from previous runs or measurements. The simulation time step is automatically chosen between user defined limits. The modelled situations can be stored as snap-shots for later use; the full snapshot comprises all the specification data including the object names, the run-time snapshot comprises all data needed for running simulation of a specific process, and the small snap-shot comprises only state variables. All the user operations during a simulation run are recorded, which enabling replay of the simulation run.

Different users can work on their own subprocesses on separate workstations. Specified model data can be written in text form to specification batch files, which can easily be transported and combined to larger models. The computer platform includes UNIX workstations as HP, Sun, Alfa, and Silicon Graphics, equipped with X-Windows software. It also runs in network environment. The calculation program can thus be situated on an efficient network server and the graphics interface on a separate workstation. It is also possible to operate the graphics interface from an X-terminal or from a PC emulating an X-Terminal.

4. GOVERNING EQUATIONS

The parameters and boundary values needed by the governing equations for mass, momentum, chemical species and energy, are mainly derived from process structures and dimensions, material properties and empirical correlations [2]. The general conservation equations for mass, momentum and energy are partial differential equations with respect to time and space. The one-dimensional conservation equations for mass (1), momentum (2) and energy (3) are as follows:

$$\frac{dA\rho}{dt}+\frac{dA\rho u}{dz}=S_1 , \tag{1}$$

$$\frac{dA\rho u}{dt}+\frac{dA\rho u^2}{dz}+\frac{Adp}{dz}=S_2 \tag{2}$$

and

$$\frac{dA\rho h}{dt}+\frac{dA\rho uh}{dz}=S_3 . \tag{3}$$

In equations (1) to (3) it is assumed that the properties are averaged over the cross-section. The symbols are as follows: A is the flow cross section, ρ is the density, u is the velosity and p is the pressure. In equation (3) h represents the total entalphy of the mixture including also the kinetic energy. The right side terms S_1 ,S_2 and S_3 are the sources terms of mass, momentum and energy, respectively.

The kinetics of chemical reactions can be described by a set of ordinary differential equations [3]. The mass change due to reactions is described for each component by the equation

$$\frac{dc_j}{dt}=V_R\sum_{k=1}^{n_R}r_j^{(k)}(T,p,c_1,\ldots,c_{n_c}) , \tag{4}$$

where c_j is the concentration of component j, V_R the reaction volume, r_j the reaction rate of component j due to reaction k and n_R the number of independent reactions, n_c is the number of independent concentrations, T the temperature and p the pressure.

The thermodynamic equilibrium of a mixture containing several components in two or more phases is described by algebraic equations,

$$\mu_j^{(1)}=\mu_j^{(2)}=\ldots=\mu_j^{(n_f)} , \tag{5}$$

where $\mu_j^{(i)}$ is the chemical potential of component j in phase i and n_f is the number of phases. The thermodynamoc properties needed are calculated as functions of pressure, specific entalphy of the mixture h, and mass fractions X_j.

$$\rho, \frac{d\rho}{dp}, X_1^{(v)}, \ldots, X_{c_n}^{(v)}, X_1^{(l)}, \ldots, X_{c_n}^{(l)} h^{(v)}, h^{(l)}, \rho^{(v)}, \rho^{(l)} \\ = f(p, h, X_1, \ldots, X_{c_n}) \ . \tag{6}$$

The verification of the simulation system can be done using pilot plant data. Uncertain physical parameters and correlations can be measured and corrected. After that the simulation model can be scaled up to represent a best estimate model of a full scale plant. Finally, when the full scale plant has been taken into use, the design model can be validated, updated if needed, and used for optimization studies of the operation of the plant.

5. NUMERICAL SOLUTION

The differential equations have to be discretized both with respect to space and time. The staggered grid approach is used both for heat and flow structures. A nearly implicit integration method is used. This offers more stable calculation and larger time steps than explicit methods.

The solution phase starts by the discretization of equations (1) to (3) and the linearization of non-linear terms. The most important linearisation is done in the mass equation where the density is expressed as follows

$$\rho^k = \rho + \frac{d\rho}{dp}(p^k - p) \ , \tag{7}$$

where the superscript k refers to the new iteration step. Most other nonlinear terms have been linearised in a similar fashion. In the space discretization scheme used we define V_i and V_j as two control volumes under consideration, L_{ij} as the lenght between the middle points of the volumes, and K_{ij} as the flow resistance, the equations (1) to (3) can be discretized and linearized as follows:

$$\frac{V_i}{\Delta t}(\rho_i - \rho_i^{t-\Delta t}) + \frac{V_i}{\Delta t}\frac{d\rho_i}{dp_i}(p_i^k - p_i) = -\sum_j^{j \to i} w_{ij}^k + S_1 \ , \tag{8}$$

$$\frac{L_{ij}}{A_{ij}} \frac{(w_{ij}^k - w_{ij}^{t-\Delta t})}{\Delta t} - p_i^k + p_j^k + \frac{1}{2} K_{ij} w_{ij} |w_{ij}| + K_{ij} |w_{ij}| (w_{ij}^k - w_{ij}) - \frac{w_{hi}^2}{\rho_{hi} A_{hi}^2} + \frac{w_{ij}^{k\,2}}{\rho_{ij} A_{ij}^2} = S_2 \ , \tag{9}$$

and

$$V_i \frac{\rho_i h_i^k - \rho_i^{t-\Delta t} h_i^{t-\Delta t}}{\Delta t} - \sum_j^{j \to i} w_{ij}^k h_j^k + \sum_j^{j \to i} w_{ij}^k h_i^k = S_3 \ . \tag{10}$$

The linearization of the momentum flux term in the equation (9) and the source terms has not been shown in detail. In equations (8) to (10) w is the mass flow which can be calculated as

$$w_{ij} = A_{ij} \rho_{ij} u_{ij} \ . \tag{11}$$

Separate equation systems are solved for control volume pressures, stream total flows, control volume temperatures, stream component flows, chemical reactions, temperatures in heat structures and automation system dynamics. A quasi Newton method is used to correct the errors caused by the interaction between the separate equation systems and by the linaerization of the nonlinear terms. After that the thermodynamic properties and their derivatives are updated. The remaining error is taken into account during the next time step. If the error is too large the previous iteration is automatically repeated with a smaller time step.

The advantages gained by the tearing of the large equation system into separate smaller subsystems are obvious. The resulting matrices can be made fairly diagonally dominant. Their sparsity can be preserved during the solution using optimized sparse matrix methods. The computer code is optimized for RISC architecture processors. Also hardvare capable of parallel or vector prosessing can be used efficiently; the calculation of the matrics element values of the sparse matrices can benefit from vector processing, and the component flows may be solved in parallel.

6. APPLICATION EXAMPLES

The flow sheets representing the following different application models have been plotted by post-script routines on a laser printer. The graphical specification of a combined cycle power plant model is shown in Figure 1. The diagram includes components related to the water, steam and fluegas flow, and components related to the electrical and control systems. A distillation plant model diagram is shown in figure 2, including a model of a sieve tray column and a model of a packed column. The models are capable of simulating the pressure dynamics; this is essential especially when simulating vacuum columns and columns where floating pressure control can cause large changes in operating pressure. The flow sheet of a batch pulping process for craft cooking is shown in figure 3. The model is used for testing of cooking control and energy control concepts. In figure 4 is shown the model specification of a black liquor evaporation plant used in the chemical recovery process in pulp and paper industry.

7. CONCLUSIONS

The developed simulation software can be used in a wide range of applications, for instance for decisions making in process design, automation system design and the planning of operational procedures. Comprehensive disturbance and safety analysis can be made by experimenting with the dynamic simulation model - not with the real plant. The software is also intended for different training purposes, basic training of students, more advanced training of design engineers, and training of plant operators.

8. REFERENCES

[1] E. Silvennoinen, K. Juslin, M. Hanninen, O. Tiihonen, J. Kurki, K. Porkholm, The APROS software for process simulation and model development, VTT publications 618, Espoo, 1989.

[2] S.V. Patankar, Numerical heat transfer and fluid flow, McGraw-Hill, 1980.

[3] K. Juslin, S. Kaijaluoto, B. Kalitventzeff, A. Kilakos, E. Lahdenperä, An equation oriented software package for dynamic simulation of chemical processes, COPE-91, Barcelona, Spain, 1991.

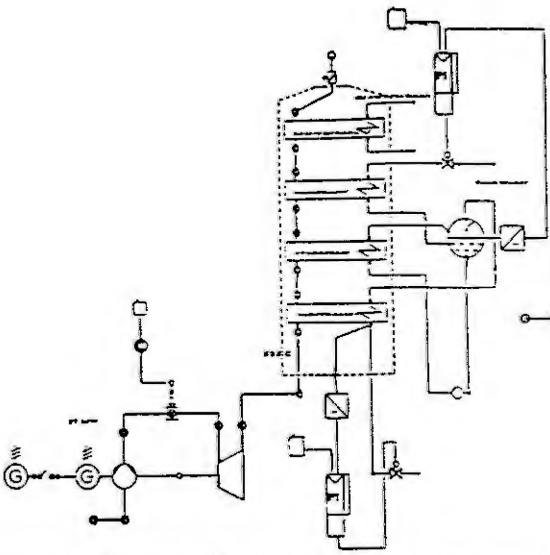


Figure 1. Combined cycle power plant model.

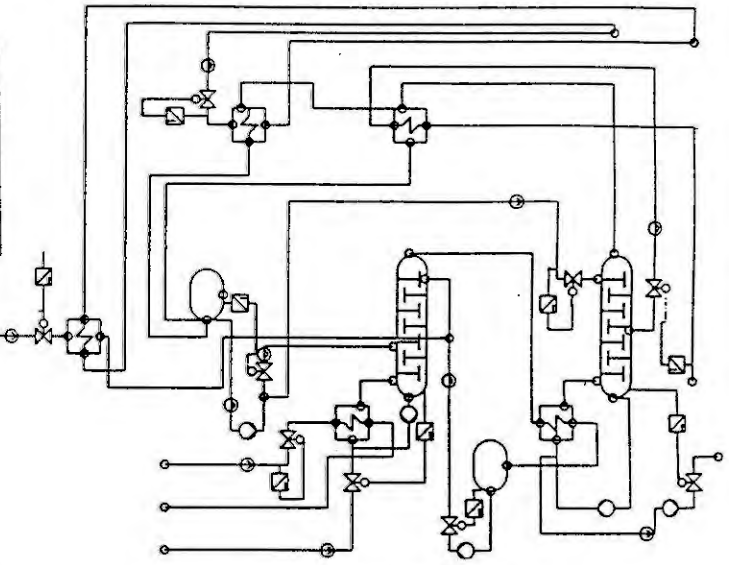


Figure 2. Distillation plant model.

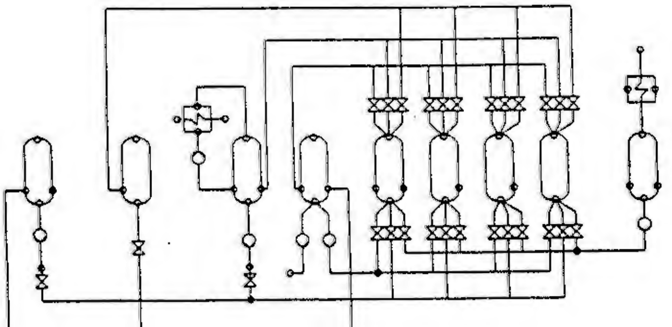


Figure 3. Batch pulping process model.

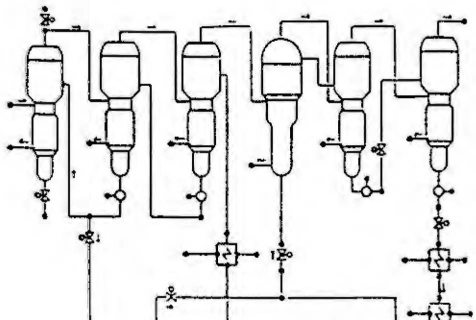


Figure 4. Black liquor evaporation process model.

Simulation of Energy Systems - Guidelines and Pitfalls

by
Niels Houbak, Laboratory for Energetics,
Technical University of Denmark, DK-2800 Lyngby, DENMARK

Abstract.

This paper is a survey paper on simulation of energy systems. It covers important aspects of the different phases of modeling an energy system. It also gives a short introduction to robust numerical methods which it is strongly recommended to use. Finally, 3 short examples illustrates some of the problems a modeler still has to face despite the developments in modern simulation software.

The Modeling Process.

Simulation of a system, i.e. in this paper an energy system, is a task that normally has to be divided into several subtasks. Each of these tasks are equally important because the final difference between the physical reality and the simulated solution is the sum of the errors from each subtasks.

The first task to be performed is to build a physical model of the real system. This model is typically a drawing of the system. The boundaries for the system are specified, and neglected processes and processes accounted for are determined. These approximations to the nature of the problem can cause a difference between the actual behavior of the system and the computed solution.

The second task is to make a mathematical model from the physical model. This is done by applying some of the fundamental laws. Conservation of mass and energy, definition of heat exchanger efficiency, the ideal gas equation, and the heat conduction law being examples on such fundamental laws. Some of the laws do have limitations; i.e. an ideal gas need not be ideal under the current conditions (pressure and temperature). This may influence the accuracy of the solution.

The third task is to apply a numerical method to the mathematical model in order to obtain a numerical model. Finding the most appropriate numerical method depends on the type of the problem: algebraic equation system (static model), differential equation system (dynamic model), differential algebraic equation system (DAE model), or partial differential equation system (PDE model). Further, the choice of method may also depend on the actual behavior of the problem: stiff / non-stiff, highly oscillatory or discontinuous being some important properties. It is obvious that applying the numerical method introduces errors to the solution; these are iteration error, integration error and/or discretization error.

The fourth task is to implement the numerical model in a program, to run the program, and to present the results. In this process the computer introduces rounding errors into the solution.

It is good modeling practice to use stepwise refinement of the model. Always start with the simplest and least complicated model of the system, even if the results are known on beforehand to be too erroneous. Having verified the correct general behavior of the simple model, new components can be added (one by one).

Mathematical Models of Energy Systems.

Often an energy systems (for example a power stations) is modeled as a network of components, turbine, furnace, heat exchanger, or pump. The connections between components are assumed to be without losses. This component approach requires access to a large library of precompiled components but also that the user can define his own components. The behavior of a component is normally given by equations and preferably they should be residual equations. Complex component models may include several table lookups in order to get correct water/steam-, gas-, or other material properties. These lookups can be relatively expensive and should be avoided whenever possible. Other components contain chemistry; i.e. chemical reaction equations that have to be satisfied.

It has been much used to make electrical analogies of various systems. For energy systems it is obvious that massflow is current and pressure is voltage, but the temperature (enthalpy) of the massflow has no electrical equivalent. In some situations, the chemical composition of a massflow varies; this has no equivalent either.

The component oriented approach usually generates many equations, but it is very user friendly. For a given component only a few parameters have to be specified, almost independent of the complexity of the underlying equations. An alternative approach for the user, would be to generate all the equations himself. Here, the number of equations can be reduced dramatically by more or less trivial substitutions.

Many energy system models are steady state (static) models. Mathematically they can be described in the following way

$$(1) \qquad \underline{0} = \underline{g}(\underline{z}, \underline{p}) \ .$$

The vector $\underline{z}$ contains all the unknown variables, $\underline{p}$ represents the parameters and the function g describes all the residual equations from the components.

Sometimes, one or more of the unknown variables are determined by a differential equation. These variables are called $\underline{y}$, their time derivative $\underline{y}'$ and mathematically the system can be written as

$$(2) \qquad \begin{array}{lll} \underline{y}' = \underline{f}(t, \underline{y}, \underline{z}, \underline{p}) \ , & 0 <= t <= T \ , & \underline{y}(0) = \underline{y}_0 \\ \underline{0} = \underline{g}(t, \underline{y}, \underline{z}, \underline{p}) \ , & & \underline{g}(0, \underline{y}_0, \underline{z}(0), \underline{p}) = \underline{0} \end{array}$$

t is the model time. These equations must be solved simultaneously for $\underline{y}$ and $\underline{z}$. Systems of type (2) are referred to as DAEs in semi explicit form.

Numerical Methods.

Having a mathematical formulation of the system as in equation (1), the most robust method for the numerical solution is Newton iteration. This well-known method can be described as follows

$$(3) \qquad \begin{array}{ll} \underline{\underline{J}} \, \Delta\underline{y}_s = - \underline{g}(\underline{y}_s, \underline{p}) & \qquad \underline{\underline{J}} = \dfrac{\partial}{\partial \underline{y}} \underline{g}(\underline{y}, \underline{p}) \\ \underline{y}_{s+1} = \underline{y}_s + \Delta\underline{y}_s & \end{array}$$

Subscript s is the iteration count. $\underline{\underline{J}}$ is the Jacobian matrix. The first equation in (3) is a linear system of equations that has to be solved in each iteration. It is assumed that a good initial approximation $\underline{y}_0$ to the solution exists. Specially for large systems an approximation to the exact Jacobian matrix is used because the cost of an iteration is dominated by generating the matrix and solving the linear system. There are (at least) 3 different ways of approximating the Jacobian;

1) Numerical differentiation (difference approximation),
2) rank one updates (Broyden update), and
3) assume the Jacobian matrix is constant for several iterations.

For very large problems the non-linearities are often located in relatively few equations; combining 1) and 3) may be advantageous. The iteration is converging as long as the residual ($\underline{g}(\underline{y}, \underline{p})$) is decaying in each component. As long as this decay is satisfactory, there is no need to update the iteration matrix.

The ODE part of (2) is normally written as

$$(4) \qquad \underline{y}' = \underline{f}(t, \underline{y}) \ , \quad 0 <= t <= T \ , \quad \underline{y}(0) = \underline{y}_0$$

and is referred to as an initial value problem. The numerical solution is only computed at special points called steppoints. From the initial solution the solutions at the steppoints are generated sequentially. By interpolation the analytical solution is approximated between steppoints. The distance between two steppoints is called the stepsize (denoted h). A good, simple, and popular ODE method is the classical 4'th order Runge-Kutta.

$$(5)\qquad \begin{aligned} \underline{K}_1 &= \underline{f}(t_n \quad , \underline{y}_n \quad) & \underline{K}_3 &= \underline{f}(t_n + h/2, \underline{y}_n + h/2{*}\underline{K}_2) \\ \underline{K}_2 &= \underline{f}(t_n + h/2 \, , \underline{y}_n + h/2{*}\underline{K}_1) & \underline{K}_4 &= \underline{f}(t_n + h \, , \underline{y}_n + h \ {*}\underline{K}_3) \\ \underline{y}_{n+1} &= \underline{y}_n + h/6{*}(\underline{K}_1 + 2{*}\underline{K}_2 + 2{*}\underline{K}_3 + \underline{K}_4) \end{aligned}$$

Subscript n is the step number. Some ODEs are stiff: Initially a steep transient dramatically varies the solution; after the transient has died out, the solution is slowly varying. The stepsize strategy will select a small stepsize during the transient and increase the stepsize when the transient has died out. Strategies for stepsize control are found in [3]. The previous Runge-Kutta method is explicit and is not well suited for solving stiff systems. A simple implicit method is backward Euler

$$(6)\qquad \underline{y}_{n+1} = \underline{y}_n + h\,\underline{f}(t_{n+1}, \underline{y}_{n+1}) \,.$$

The method is implicit in $\underline{y}_{n+1}$. It is rewritten in (7), which is of the same form as (1). It is solved using the quasi-Newton method previously described

$$(7)\qquad \underline{0} = \underline{F}(\underline{y}_{n+1}) = \underline{y}_n + h\,\underline{f}(t_{n+1}, \underline{y}_{n+1}) - \underline{y}_{n+1} \,.$$

A type of implicit ODE methods are the BDF methods. For constant stepsize they can be derived from (6) by adding a linear combination of backward information

$$(8)\qquad c_k\underline{y}_{n-k} + c_{k-1}\underline{y}_{n-k+1} + \ldots + c_0\underline{y}_n + \underline{y}_{n+1} = b\,h\,\underline{f}(t_{n+1}, \underline{y}_{n+1}) \,.$$

In each step, the system of non-linear equations to be solved is almost identical to (7). b and c_0 to c_k are determined in order to achieve the appropriate order (k + 1). These methods are normally implemented using the Nordsieck formulation (a Taylor expansion), but this does not change the form of the system of non-linear equations to be solved in each step. Coefficients for several of these methods can be found in [2].

Also implicit Runge-Kutta methods exists. A 2 stage, 2'nd order Diagonally Implicit Runge-Kutta (DIRK) is given in (9). The coefficients can be found in [4].

$$(9)\qquad \begin{aligned} \underline{K}_1 &= \underline{f}(t_n + b_1\,h, \underline{y}_n + h\,\gamma\,\underline{K}_1) & \gamma &= 1 - 1/\sqrt{2} & b_1 &= \gamma \\ \underline{K}_2 &= \underline{f}(t_n + b_2\,h, \underline{y}_n + h\,a_{21}\,\underline{K}_1 + h\,\gamma\,\underline{K}_2) & a_{21} &= 9 - 28{*}\gamma & b_2 &= a_{21} + \gamma \\ \underline{y}_{n+1} &= \underline{y}_n + h\,(c_1\,\underline{K}_1 + c_2\,\underline{K}_2) & c_1 &= (43 + 10{*}\gamma)/62 & c_2 &= 1 - c_1 \end{aligned}$$

By substituting $\underline{Y} = \underline{y}_n + h\,\gamma\,\underline{K}_1$ (or $\underline{K}_1 = (\underline{Y} - \underline{y}_n)/(h\,\gamma)$) into the first stage of (9) we get an equation of the form (7) with $\underline{Y}$ being the iteration variable.

The DAE problem (2) can be solved in one of two ways. If an implicit ODE method is used, equation (7) from the ODE part is coupled with the algebraic part of the system. For explicit methods it is possible to apply a quasi-Newton method for solving the algebraic part, each time the derivatives have to be calculated. DAE problems are covered in [1].

What can go wrong?

With a good mathematical model of a system and using the best numerical methods for solving the problem, can anything go wrong? The answer is yes. First of all, there is the problem that a non-linear model may have 0, 1, and several solutions. Second, it is possible to make small errors with major effect in each of the 3 modeling stages. This is illustrated by the following examples.

The first example is the problem of a discontinuous behavior of a model. An on-off control, say, can cause serious troubles for the stepsize strategy in an ODE solver, as can a linearly interpolated table of data. A discontinuity can only be passed with a very small stepsize. The process of both reducing the stepsize and locating the point of discontinuity will cost many rejected steps.

The second example illustrates the problem of linear dependencies in the model. This has 3 aspects.
1) It is sometimes difficult to find all the equations of a system. One can combine 2 of the equations in the model to get an extra equation. Usually, this is trapped by the linear equation solver routine, saying the

matrix is singular. It is very rare to get information on which equation is superfluous.
2) Energy systems can be both open loop and closed loop systems. For closed loop models there are always one surplus mass balance equation per separate closed loop; it must be removed.

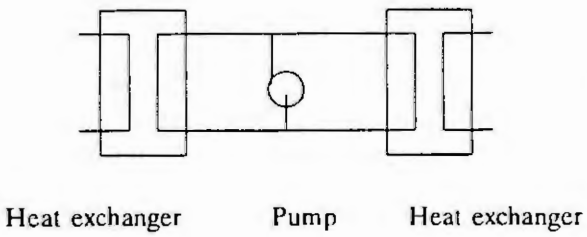


Figure 1.

3) Too simple models may also cause linear dependencies. In figure 1, there is a pump and two heat exchangers. The pump has a characteristic connecting the pressure increase and the massflow. If the heat exchangers are assumed to have the same constant pressure drop, the system is singular! Any mass flow can circulate between the two heat exchangers without influencing the pump pressure. With only one heat exchanger there is no problem.

The third example is a gas turbine with a recuperator. The exhaust gas from a gas turbine is normally very hot (500 °C) and with the recuperator some of this heat is used for preheating the compressed air before the combustion camber. Figure 2 shows this schematically.

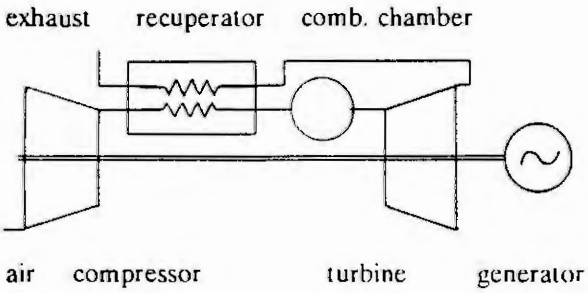


Figure 2.

The system can relatively easy be modeled. Reducing the load on the gas turbine reduces the temperature of the exhaust gas. This temperature may drop below the temperature of the compressed air. Consequently, the exhaust gas is heated! The problem is the recuperator, that works equally well both ways. In this small example it is easy to see the problem, but simulating a large system, it is important to check 'trivial' conditions for all components.

Conclusion.

When modeling large and complex systems it is necessary to have the right tools. This can be either a set of basic numerical solution routines or a high level simulation language. Even with the right tools, it is still neeessary to be very careful when building and validating a model.

The modeling process can be divided into 4 separate stages. It is important to avoid mixing models from the different stages; i.e. don't insert a difference approximation to a derivative when developing a mathematical model, this gives a build-in Euler method, without any chance of changing numerical method later.

Equally important is the step by step refinement of a model. Develop and test submodels before they are glued together to form a complete system model. Modern simulation tools have many facilities that are very useful for making good models and validating them, but these features can never take the responsibility for the quality of a model away from the modeler.

References.

[1] Brenan, K.E., Champbell, S.L., and Petzold, L.R.: "Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations". North-Holland. 1989.
[2] Gear C.W.: "Numerical Initial Value Problems in Ordinary Differential Equations". Prentice-Hall. 1971.
[3] Gustafsson K.: "Control of Error and Convergence in ODE Solvers". Dept. of Automatic Control, Lund Institute of Technology, Lund, Sweden. 1992. (Ph.D. thesis)
[4] Nørsett, S.P.: "Semi explicit Runge-Kutta Methods". Mathematics and Computation. Vol 6/74, NTH, ISBN 82-7151-009-6. 1974.

Software Supporting Modelling Automation for Optimal Control Problems

R. Mehlhorn and G. Sachs
Institute of Flight Mechanics and Flight Control
Technische Universität München
Arcisstr. 21
80290 München, Germany

Abstract. A method is proposed which is aimed at reducing modelling effort and increasing productivity when dealing with optimal control problems. Part of the method is a dedicated language consistent with the LATEX-language providing automatic differentiation in order to generate necessary derivative information. The compiler built for this language produces efficient FORTRAN77 subroutines with a standard interface to a multiple point boundary value problem solver being capable to solve the transformed optimal control problem numerically. The efficiency of the automatically generated subroutines is shown.

1. Introduction

Modelling is an important task in engineering control problems when considering complex systems in a realistic way. The modelling effort is significantly increased for optimization techniques (*Minimum Principle*) which make use of adjoint systems being of the same complexity as the original physical system. The modelling effort is further increased when changes in the structure of a physical system occur in the course of a process.

The proposed method is aimed at reducing modelling effort and increasing productivity by providing a *small programming language* that is consistent with LATEX [1], a language widely used for typesetting of scientific publications.

With the programming language developed especially for solving optimal control problems using the Minimum Principle the problems can be formulated in a very compact manner which is furthermore directly correlated to theoretical approaches to the addressed problems, e.g. [2, 3]. This is basically due to the introduction of differentiation operators which are evaluated using the technique of automatic differentiation and some even more sophisticated operators being capable of computing the optimal control at a given point of time. Basic knowledge of the structure of optimal control problems gives rise of detecting not only lexical and syntactical but also some semantic errors.

The consistency with typesetting software ensures the identity between problem formulation and documentation. Therefore semantic errors during the modelling process of optimal control problems can be further reduced.

An automatic transformation process is performed from the proposed programming language to an existing computer language with the use of a dedicated compiler. The compilation procedure includes lexical and syntactical analysis and code generation. Creating such small languages as proposed is considerably aided by some standard tools provided by UNIX-Systems [4].

2. Optimal Control Problem Description

The optimal control problems under consideration are of *Bolza* type

$$\min_{\mathbf{u}, t_1, \ldots, t_m} \mathcal{J} := \Phi\left(\mathbf{x}(t_m), \mathbf{p}\right) + \int_0^{t_m} \mathcal{L}\left(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p}\right) \, dt \tag{1}$$

where $\mathbf{x}(t)$ denotes state variables, $\mathbf{u}(t)$ controls and $\mathbf{p}$ system parameters. The dynamics of the system is to be modelled by a set of differential equation systems

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} := \mathbf{f}^{(j)}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p}), \qquad t \in (t_{j-1}, t_j), \qquad j \in [1, m] \tag{2}$$

each of which is valid in a certain phase j and is assumed to be sufficiently smooth during this phase in order to use higher order integration methods.

The problem of finding the optimal control history $\mathbf{u}(t)$, the optimal phase separation points $t_1, \ldots, t_{m-1}$ and the optimal final point t_m can be reduced to a multiple point boundary value problem and solved by the appropriate numerical methods [5].

This reduction step includes the modelling of the adjoint dynamic system

$$\begin{aligned} \frac{\mathrm{d}\lambda_{\mathbf{x}}}{\mathrm{d}t} &:= -\frac{\partial \mathcal{L}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p})}{\partial \mathbf{x}} - \lambda^T(t)\,\frac{\partial \mathbf{f}^{(j)}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p})}{\partial \mathbf{x}}, \qquad t \in (t_{j-1}, t_j), \qquad j \in [1, m] \\ \frac{\mathrm{d}\lambda_{\mathbf{p}}}{\mathrm{d}t} &:= -\frac{\partial \mathcal{L}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p})}{\partial \mathbf{p}} - \lambda^T(t)\,\frac{\partial \mathbf{f}^{(j)}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p})}{\partial \mathbf{p}}, \qquad t \in (t_{j-1}, t_j), \qquad j \in [1, m] \end{aligned} \tag{3}$$

which can be performed automatically by the proposed method applying results concerning automatic differentiation.

The proposed language also allows for specification of nonlinear boundary conditions as well as interior point and jump conditions which complete the formal problem statement of the optimal control problem.

3. Modelling System Dynamics

An important modelling task is to describe the dynamics of the system properly. There are some features of the proposed language that reduce the effort for this task when compared with general purpose programming languages.

As usual, auxiliary variables may be introduced in order to get a modular view of the system. In terms of the proposed language these auxiliary variables are called modelling functions. They may be composed by the use of state variables $\mathbf{x}$, controls $\mathbf{u}$, parameters $\mathbf{p}$ as well as adjoint variables $\lambda_{\mathbf{x}}$ and $\lambda_{\mathbf{p}}$, switching variables $\varsigma_{j=1,\ldots,m}$ and previously defined symbolic constants and modelling functions.

One of the above mentioned features of the proposed language is the possibility to describe piecewise smooth functions by the use of switching variables ς_j which are implicitly defined variables correlated to the integration phases j. They are defined as

$$\varsigma_j(t) := \begin{cases} 1 \text{ iff } t \in (t_{j-1}, t_j) \\ 0 \text{ iff } t \notin (t_{j-1}, t_j) \end{cases}, \quad j \in [1, m] \tag{4}$$

Another feature of the proposed language concerns the transcription of results obtained by previous modelling efforts to the formal problem statement. Modelling procedures often lead to polynomial approximations of tabular or otherwise given data which may be derived from theoretical considerations or experiments. In order to enable a direct transcription of the modelling procedure to the problem formulation, a super-elementary operation for the evaluation of polynoms has been implemented. Introducing super-elementary functions such as polynoms, scalar products or matrix-vector operations not only allows for compact problem formulations but also give rise to more sophisticated algorithms concerning their evaluation and automatic differentiation. This especially holds for higher order derivatives.

The maybe most sophisticated operation implemented in the current version of the language is the automatic modelling of the optimal control law. Concerning regular controls a necessary condition which usually defines the control variables is the system of algebraic conditions

$$\frac{\partial \mathcal{L}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p})}{\partial \mathbf{u}} + \lambda^T(t)\,\frac{\partial \mathbf{f}^{(j)}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p})}{\partial \mathbf{u}} = 0, \qquad t \in (t_{j-1}, t_j), \qquad j \in [1, m] \tag{5}$$

Applying automatic differentiation to Eq. (5) yields a system of equations linear in the rate of the controls $\mathrm{d}\mathbf{u}/\mathrm{d}t$ from which the rate of the controls can be determined explicitly. Together with pointwise evaluations of Eq. (5) this procedure leads to a fully automatized modelling of the optimal control.

In the case of controls linearly entering systems dynamics, the controls take on either values on the boundary of the control domain or become singular. Since from a practical point of view singular controls of first order are very important, a special operator for modelling singular controls was also implemented which not only evaluates the optimal control law, but also counteracts inherent numerical instability effects existing with this kind of controls.

4. Automatic Differentiation

Automatic differentiation is used to efficiently deal with all operations requiring differentiation. Neglecting rounding errors it yields the exact value of the desired derivative. Applied to evaluating the gradient of a scalar function it has been shown [6] that the derivative information is available in less than five times the evaluation of the underlying scalar function using the technique of automatic differentiation in reverse mode. Experience in several realistic applications has shown that it is presumable to obtain a factor of about two.

Application of automatic differentiation particularly concerns modelling the adjoint differential equation systems (3) which are of the same size and complexity as the original engineering system. Modelling the adjoint system and the necessary algebraic conditions (5) concerning optimality of control can be considered as a gradient evaluation of the underlying *Hamiltonian* $\mathcal{L} + \lambda^T \mathbf{f}^{(j)}$ with respect to the state variables, controls and system parameters. Therefore the advantages of automatic differentation can be fully exploited.

Automatic differentiation provides also a particular advantage for optimal control problems showing singular arcs and problems with state constraints of higher order. Usually, evaluating and programming the optimal control law for singular arcs and for arcs with active state constraints requires a great effort, is very time consuming and also a major source of errors. In these cases evaluation of gradients can be combined with specific propoerties of the structure of optimal control problems so that the efficiency can be enhanced also for automatically computing higher order time derivatives. Additionally methods for increasing numerical stability concerning the resulting index reduced differential algebraic systems are automatically applied [7].

Since the numerical solution is obtained using the Multiple Shooting Method [5], further derivative information is required which concerns the dependency of the result of the integration of an initial value problem with respect to the initial values. The automatic differentiation technique is also applied to the computation of these transition matrices yielding full accuracy.

5. Results

The proposed method has been applied to a variety of optimal control problems related to flight trajectories. Two applications for which reference data of usual solution techniques are available are considered in the following.

The first example, referenced as nonplanar ascent in Table 1 deals with the minimization of fuel consumed during the ascent of an orbital stage. The ascent of the rocket propelled orbital stage of a two stage system begins at a hypersonic flight condition (Mach number 6.8, altitude 31 km) at a rather small flight path angle.

The second example, referenced as singular control in Table 1 considers range maximization for a propeller driven aircraft [8]. The resulting periodic solution of the optimal control problem shows arcs of singular control concerning throttle setting.

	Optimal Control Problem	
Task	Nonplanar Ascent	Singular Control
Hamiltonian	126 Basic Operations	246 Basic Operations
RHS of ODE	285 Basic Operations	265 Basic Operations
Jacobian	1590 Basic Operations	2071 Basic Operations

Table 1. Complexity of produced FORTRAN77-Code

Table 1 shows the number of basic operations for the two applications considered. Basic operations are understood as FORTRAN77 floating point operations or polynom evaluations.

The number of basic operations necessary to compute the Hamiltonian can be considered as a measure for the complexity of the differential equations defined by the original system. The right hand side of the ordinary differential equations shown in the second row of Table 1 (RHS of ODE) takes also into account the automatically modelled adjoint system and the optimal control law. The last row of Table 1 shows the complexity of evaluating the complete system of differential equations together with the associated Jacobian matrix which is used to compute the transition matrices.

	Optimal Control Problem	
Method / Task	Nonplanar Ascent	Singular Control
Programming FORTRAN77, Differentiation by Hand	1044.2 sec	63.0 sec
Programming LaTeX, Automatic Differentiation	1683.4 sec	58.3 sec
LaTeX to FORTRAN77 Compilation Time	13.4 sec	14.6 sec

Table 2. Comparison of computational costs

Table 2 shows a comparison between the usual technique of deriving the right hand side of the adjoint system and the optimal control law by hand and the proposed method of using a programming language especially built for dealing with optimal control problems. The example concerning the nonplanar ascent which was obtained by applying a continuation method proves that the computational costs for program execution may increase. Experience with the above mentioned optimal control problems shows that this slow down factor can be expected to be less than two. However, it must be taken into account that the programming effort of modelling the adjoint system and the optimal control law may be considerably more time consuming. As far as computational costs for program execution are concerned, Table 2 shows it may as well be possible to outperform handwritten code.

6. Conclusions

Experience with the proposed method in the field of flight trajectory optimization has shown that the method provides an efficient means for reducing modelling effort and increasing productivity when dealing with optimal control problems.

This particular holds for realistic engineering problems resulting in an complex modelling of the dynamics of the system and for problems where state constraints or singular controls occur.

The proposed method also increases efficiency when modelling changes are considered and their effects are investigated.

In addition to a theoretical treatment of the method, numerical results including a comparison with usual techniques have been presented in order to show quantitatively the improvements in efficiency.

7. References

[1] L. Lamport. *LaTeX- A Document Preparation System.* Addison-Wesley Co., Inc., Reading, MA, 1985.

[2] A.E. Bryson and Y. Ho. *Applied Optimal Control.* Hemisphere Publishing Corporation, Halsted Press, New York, 1972.

[3] H. Seywald. Optimal Control Problems with Switching Points. NASA Contractor Report 4393, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, 1991.

[4] A.V. Aho, R. Sethi, and J.D. Ullmann. *Compiler-Principles, Techniques and Tools.* Addison-Wesley, New York, 1986.

[5] P. Hiltmann. *Numerische Lösung von Mehrpunkt-Randwertaufgaben und Aufgaben der optimalen Steuerung mit Steuerfunktionen über endlichdimensionalen Räumen.* Dissertation, Technische Universität München, 1990.

[6] A. Griewank. *Mathematical Programming: On Automatic Differentiation in Mathematical Programming - Recent Developements and Applications.* Kluwer Academic Publishers, Boston, 1989.

[7] R. Mehlhorn, K. Lesch, and G. Sachs. A Technique for Improving Numerical Stability and Efficiency in Singular Control Problems. In *AIAA Guidance Navigation and Control Conference Proceedings,* pp. 388-396, 1993.

[8] G. Sachs and K. Lesch. Optimal Periodic Trajectories of Aircraft with Singular Control. Report No. 273, DFG-Schwerpunktprogramm Anwendungsbezogene Optimierung und Steuerung, 1991.

Evolutionary Modelling: a structured approach to model construction

Jan Top and Hans Akkermans

Netherlands Energy Research Foundation ECN
P.O. Box 1, NL-1755 ZG Petten (NH), The Netherlands
E-mail: top@ecn.nl, akkermans@ecn.nl

Abstract

Our objective is to provide a basis for automated modelling of energetic-dynamic systems. We have developed the *evolutionary modelling approach*, which defines a framework for organizing *structural* and *behavioural* assumptions. In this paper we summarize the main features of this approach by discussing the origins of assumptions and their ontological distinctions. Separating different ontologies is crucial for a knowledge-based approach to modelling support systems, in particular with respect to the organization of generic component libraries.

1 Introduction

In the development of technical devices and processes, modelling is a serious bottleneck. Indeed, modelling is often considered to be an *art*, suggesting an unorganized creative process that cannot be automated at all. However, here we describe a structured, knowledge-based approach to physical model construction called 'Evolutionary Modelling'. This approach is based on Artificial Intelligence concepts and methodology for knowledge modelling and representation. It has been implemented in an experimental knowledge-based modelling and simulation environment called QuBA.

We take the view that modelling is a form of design. Similar to the design task [3], we can decompose the modelling task into *specification of assumptions, construction of the model* and *assessment of the model with respect to the assumptions*. The specification subtask is the most important subtask: it is the incremental process of formulating the exact question which the model is supposed to answer. Hence, this subtask cannot be fully automated. However, a knowledge-based approach to the construction and assessment subtasks supports the modeller in posing the right question for the given circumstances. This is possible because generic knowledge can be made available in the form of reusable library components and domain theories.

Since modelling can be viewed as the incremental specification of explicit assumptions, a structured approach necessarily calls for an orderly way of maintaining these assumptions. We begin with distinguishing between *behavioural* aspects and *structural* aspects of a model. Structural properties are those characteristics of the model that need to be given in order to implement it. For example, the structure of a mathematical model is defined by the set of variables and parameters and the relations between them. The behavioural properties are those features that emerge from this structure. For a mathematical model these could be stability, eigenfrequencies, asymptotes etc. Modelling requires specification of a structure such that the observed or intended behaviour is realized. To explain our approach we will deal with the following questions:

- Where do modelling assumptions come from?
- How can modelling assumptions be organized?

These issues will be covered in the next two sections in the context of *energetic-dynamic* systems. In section 4 we will shortly describe the consequences for model libraries. The approach summarized in this has been described in detail in [6]. An example of the evolutionary modelling approach is presented in [7].

2 Sources of modelling assumptions

There are various potential sources of assumptions which can help the modeller to focus on relevant issues. First, the *modeller* supplies initial, partial assumptions by asking a question. For example, by mentioning *'central heater'* and *'temperature'* in a central heating problem a number of assumptions have already been explicated, thus narrowing down the modelling context. The analysis of the original *query* is often considered as an important element of the modelling task [1, 4]. Second, *heuristics* derived from similar circumstances and applications can be used to suggest 'default' assumptions. For example, 20 degrees Celcius is usually viewed as an *agreeable* temperature. Another heuristic could be that from experience we know that the given type of heating system will reach its setpoint within one hour under normal conditions. Finally, the most general sources of modelling assumptions are *domain theories*. Theories provide a basis for rationality, giving confidence to models without the need for exhaustive validation. For example, if a model does not violate the law (assumption) of energy conservation, there is one reason less for scepticism.

How can this knowledge be introduced into the modelling process? Heuristic and theoretical knowledge can be stored and retrieved if we can find proper representations. Query analysis seems to require interpretation of natural language, which is a separate problem of its own. But the modeller can be given guidance by making selections in lists and editing graphical models proposed by the system rather than having to formulate a query from scratch. We consider the following elements to be essential carriers of modelling knowledge:

1. *Generic components.* For example, a generic model for a class of central heating systems can be proposed. The introduction of such a generic model component will add to the explicit assumptions underlying the model, to be accepted or rejected by the modeller. For example, it introduces the assumption that there is an electrical pump, a heater, pressure gauge, etc. Generic components can be accessed through part-of (decomposition) and kind-of (taxonomy) hierarchies.
2. *Formal representation languages.* For lumped parameter dynamic systems the mathematics of ordinary differential equations is an adequate representation. However, it is very general and allows non-physical constructs. An example of a more specific formalism is the bond-graph language [2], which is based on power continuity.

Modelling tools can exploit the knowledge from generic components and formal languages. In the case of physical systems, this can be done in a way set out in the next section.

3 Organization of modelling assumptions: four ontologies

A major activity in engineering is to design generic physical components with more or less ideal functions, such that the overall design task given a specific problem can be reduced to a configuration task, if necessary supplemented with some local tuning. These functions are typically shown in *engineering drawings*. However, the engineering drawing is not an unambiguous representation. On the one hand it is typically used for configuration of a system out of device components. On the other hand, the drawing is also interpreted as a representation of more or less ideal functions or processes: an initial abstraction step is implied. Hence, it depends on the modeller how an engineering drawing is going to be interpreted.

Therefore, a more precise representation of engineering knowledge is needed, and it is necessary to distinguish between the different aspects or *ontologies* that are used to describe a model. This reduces the amount of information that has to be handled at any given time during modelling. In the domain of *energetic-dynamic* systems we distinguish the following ontologies:

Functional components Components are the basic starting points for decomposition, just because components are designed to perform well-defined functions. Therefore, we define the initial abstraction level in terms of *functional components*, which are subsystems expressing two aspects of the observed system. First, we have the *interface* between a subsystem and its environment. As far as its energetic-dynamic behaviour is concerned, a component is completely defined by the set of *energy ports* through which it exchanges energy with the environment. The ports are characterized by a physical subdomain (eg. mechanical, electrical, hydraulic, etc.). Second, a *label* is used to indicate a class of engineering functions. For example, there are AC-motors, pumps, pipes etc. This is merely a (heuristic) way to provide additional organization of models. We note that the taxonomies of functional components are domain dependent.

Physical processes The second abstraction level describes the actual physical processes occurring in the considered system. A proper way to describe these processes is the bond graph. The external ports of bond graphs refer to the ports of the associated component model. The basic difference between the component view and the process view is the fact that the bond graph represents abstract *processes* or *mechanisms* rather than devices.

Mathematical relations The third level of description is that of *signals* rather than energetic links: it describes the mathematical structure of the model. The process level and the mathematical representation are frequently fused, but differentiation between these levels is certainly sensible, both from a theoretical and from a practical point of view. They deal with essentially different types of knowledge and they carry with them different methods (*e.g.* causal analysis *vs.* symbolic algebra).

Model data The fourth level is that of *model data*, describing the quantities of interest in the model, and if possible, their scope and accuracy. The latter specify the conditions under which the model is valid. The *scope* of a quantity (parameter or variable) generalizes the notion of 'possible value' or 'applicability condition'. Any instantiation of the quantities of interest that complies with the ranges specified here and the associated set of mathematical equations, is assumed to be valid. Hence, a state variable may be initialized at any value within its *scope*. This level provides the necessary data for performing simulation runs.

4 Model libraries

As we stated before, there are two main sources for modelling knowledge that can be incorporated in support systems. In *evolutionary modelling* the bond graph language is used to restrict the modeller to create only models that are conform the elementary laws of physics. The second source, *generic components*, requires the availability of a library of submodels. Our approach provides the organizational framework required for structuring and maintaining such a library. If engineering models are organized around the four mentioned engineering ontologies, the modeller can be supplied with *reasonable alternatives* [5]. The availability of sensible alternatives is important in any design activity because it provides potential answers for the questions summarized as 'What if ...?' A library restricts the infinite number of possible alternatives to a limited set that has proved to be useful in practice.

Each of the ontological levels is stored as a separate library entry, as depicted in Fig. 1. This figure shows how the four ontologies provide a framework for structuring reusable physical models. This framework is used in practice as a basis for organizing the library to be developed in the Esprit-III OLMECO project (P6521), which is concerned with mechatronic component design. This figure shows the actual contents of a composed model for a given situation. A complete model — i.e., a model for a particular device within a certain task environment — is composed by selecting a library component from each level, while keeping the references to possible alternatives.

In addition to the entries for the actual energetic-dynamic model, important information is stored in the *observation data* section of the library. The entries in this section contain behavioural and structural descriptions together with the conditions under which they where obtained. Structural data are geometrical and material specifications, behavioural data are observations of dynamic quantities.

When retrieving a model from the library, indexes of alternative models are obtained simultaneously, suggesting alternative decompositions. Further, alternative instantiations can be obtained for the component, process or signal level in terms of respectively bond graphs, mathematical relations or data. Moreover, identical bond graphs may occur in different components, in particular if the domains are not required to be equal.

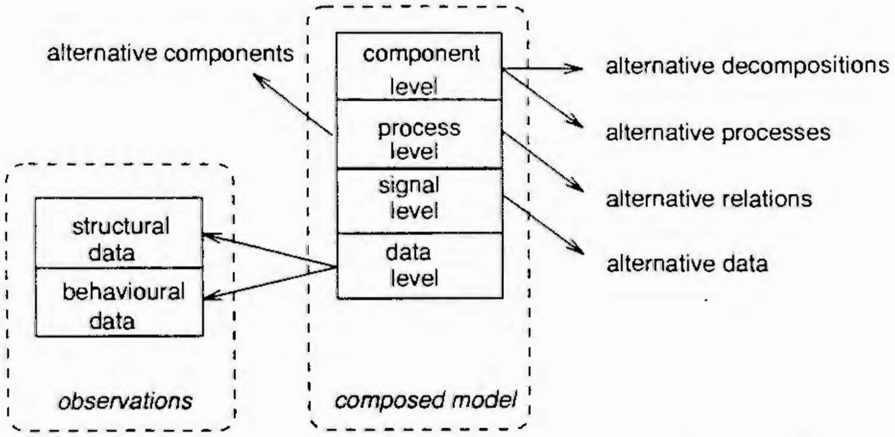


Figure 1: Composed model, retrieved from a library based on evolutionary modelling. The arrows denote links between different entries in the library. The data level can refer to a set of observational data, containing device structure (geometry or material) and behavioural data (experiments).

5 Conclusion

We have briefly sketched the *evolutionary modelling* approach for energetic-dynamic systems. This approach emphasizes the role of the specification subtask, and employs knowledge-based model construction and assessment. The latter two tasks support the modeller in formulating his or her query. In order to reduce the amount of information that has to be handled we have introduced four ontological levels: functional components, physical processes, mathematical relations and model data. Whereas the latter two levels comprise the *core model*, defining the required behavioural properties, the former two represent essential support knowledge that is more or less domain specific. The evolutionary modelling approach forms the basis for our automated modelling system called QuBA. More information on our approach can be found in refs. [6, 7, 8]

Acknowledgements This work has partially been supported by the CEC Esprit-III project 'OLMECO' (P6521) and by the Netherlands Ministry of Economic Affairs/Senter in the framework of the 'EBIB' Telematics Guidance project.

References

[1] B. Falkenhainer and K.D. Forbus. Compositional modeling: Finding the right model for the job. *Artificial Intelligence*, 51:95–143, 1991.

[2] D.C. Karnopp, D.L. Margolis, and R.C. Rosenberg. *System Dynamics: A Unified Approach.* John Wiley & Sons, New York, 1990. Second Revised edition.

[3] M.L. Maher. Process models for design synthesis. *AI Magazine*, pages 49–58, 1990.

[4] G. Pahl and W. Beitz. *Engineering Design—A Systematic Approach.* The Design Council, London, 1988. Springer-Verlag.

[5] R.C. Rosenberg. The bond graph as a unified data base for engineering system design. *Journal of Engineering for Industry*, 97(4), 1975.

[6] J.L. Top. *Conceptual Modelling of Physical Systems.* PhD thesis, University of Twente, Enschede, The Netherlands, September 1993.

[7] J.L. Top and J.M. Akkermans. Layered modelling of physical systems. In *Proceedings of the 1993 Winter Annual Meeting*. ASME, 1993. To appear.

[8] J.L. Top and J.M. Akkermans. Tasks and ontologies in engineering modelling. In *Proceedings of the International Knowledge Workshop KAW '94*, Banff, Alberta, February 1994. To appear.

Qualitative Reasoning Based on Temporal Abstraction

Peter Struss
Technical University of Munich, Computer Science Dept.
Orleansstr. 34, D-81667 Munich, Germany
struss@informatik.tu-muenchen.de

Abstract Abstraction is a fundamental element of qualitative reasoning about physical systems and crucial for coping with complexity of real problems. Temporal abstractions form an important class that comprises a variety of different transformations. We show how an important subclass can be formalized in a theory of relational behavior models of physical systems.

1 Introduction

Transforming representations of problems into new ones is often crucial for finding solutions efficiently, sometimes for finding them at all. An important class of such transformations are abstractions. Abstraction is a conceptual generalization obtained by eliminating individual and arbitrary characteristics while maintaining those distinctions only that are essential in a particular context. It is frequently, and often unconsciously, applied in human reasoning and problem solving and promises to be a good means for coping with complexity in automated reasoning and problem solving.

Whilst many AI researchers agree upon this, the attempts to formalize abstraction are quite diverse, sometimes reflecting different understanding of the nature of abstraction. They are also not comprehensive. In particular, we are lacking a theory of a quite distinctive and important type of abstraction: temporal abstraction.

Abstraction is a **process of generalization**. It steps **from individual objects to concepts** of these objects that **capture their essence** but **eliminate their individual and incidental properties**. More technically speaking, it creates equivalence classes of objects (see also [Hobbs 87]) or of existing concepts to build more abstract concepts. Of course, what is considered essential can depend on the context, task, or perspective.

One kind of abstract, qualitative description of behaviors that we want to obtain in qualitative reasoning can be described in the following way: Even though one may consider system variables and parameters to change continuously over time, only certain changes are significant (say, for instance, melting). Hence, qualitative reasoning tries to make **the essential distinctions only**. Often, this is done by identifying "landmarks" in the continuum (such as melting point and boiling point) and by collapsing the intervals between adjacent landmarks into single "qualitative values". Thus, the discretization of the domain of characteristic variables induces a discretization of time (in contrast to other methods, e.g. in numerical simulation, where a discretization of time enforces a discretization of the variable domain). For each behavior, only time points corresponding to essential changes are represented, and the intervals in between are summarized in single instances of time.

This kind of representation allows us, for instance, to characterize a large class of behaviors as "oscillations" ignorant of their particular shape, amplitude, and frequency: if we make 0 the only landmark for some variable x, then any function that is transformed into a sequence (..., 0, +, 0, -, 0, +, ...) by the above steps is an oscillation.

This is something we intuitively consider as a temporal abstraction, and our goal is to formalize this part of qualitative reasoning. The use of abstraction has been investigated and formalized in other areas of AI, such as theorem proving and planning (see, for instance, [Giunchiglia-Walsh 92]). The strength of this work is that they use formal logic as a framework. Abstraction is analyzed as a mapping between logical theories. Our approach is dual in the sense that, rather than viewing the problem from the perspective of **theories and proofs**, it analyzes different transformations applied to the set of **models** (in the logical sense).

Section 2 summarizes then our theory of multiple relational models that introduces a notion of abstraction whose form is adapted to specific representations in qualitative reasoning about physical systems. In section 3, we demonstrate that the theory presented provides a foundation for formalizing at least some of them.

Although we are aware of the fact that results presented in this paper are still preliminary, we believe they may stimulate the discussion of this important area.

2 Relational Models and their Transformations

In ths section, we summarize the formalism for relational models we developed in [Struss 92] for structuring sets of multiple models and, in particular, for using this structure in model-based diagnosis.

Different **representional spaces** for the behavior of a physical system, which may be an atomic constituent (component) or some aggregate, are given by different vectors of local variables v_i:

$\underline{v} = (v_1, v_2, ..., v_k).$,

and one or more domains of $\underline{v}$:

$\mathrm{DOM}(\underline{v}) := \mathrm{DOM}(v_1) \times \mathrm{DOM}(v_2) \times ... \times \mathrm{DOM}(v_k)$.

A behavior of the system is described by specifying the set of possible values of $\underline{v}$, i.e. by a relation $R \subseteq \mathrm{DOM}(\underline{v})$. As a logical formula, the respective behavior model can be regarded as the statement that R contains (exactly) the values that can be observed in real situations:

Definition 2.1 (Strong Behavior Model, Complete Behavior Model)

A relation $R \subseteq \mathrm{DOM}(\underline{v})$ specifies a strong behavior model by

$B(R) \Leftrightarrow \forall \underline{v}_0 \in \mathrm{DOM}(\underline{v}) \quad ((\exists s \in \mathrm{SIT}\ \mathrm{Val}(s, \underline{v}, \underline{v}_0)) \Leftrightarrow \underline{v}_0 \in R\ )$,

and a complete behavior model of C by

$M(R) \Leftrightarrow \forall \underline{v}_0 \in \mathrm{DOM}(\underline{v}) \quad ((\exists s \in \mathrm{SIT}\ \mathrm{Val}(s, \underline{v}, \underline{v}_0)) \Rightarrow \underline{v}_0 \in R\ )$.

(In [Struss 92], we focus on complete models, because they suffice for consistency-based diagnosis). Here, $\mathrm{Val}(s, \underline{v}, \underline{v}_0)$ means that $\underline{v}$ has the value $\underline{v}_0$ in the situation s, and if $\underline{v}_0 = (v_{01}, v_{02}, ..., v_{0k})$, then

$\mathrm{Val}(s, \underline{v}, \underline{v}_0) \Leftrightarrow \bigwedge_i \mathrm{Val}(s, v_i, v_{0i})$

holds. Note that we do not postulate that the value be unique; from

$\exists s \in \mathrm{SIT}\ (\ \mathrm{Val}(s, \underline{v}, \underline{v}_0) \wedge \mathrm{Val}(s, \underline{v}, \underline{v}_1)\ )$

we cannot infer $\underline{v}_0 = \underline{v}_1$. This does not only allow us to handle multiple domains for $\underline{v}$. Even if $\underline{v}_0$ and $\underline{v}_1$ are from the same domain, they may be different. For instance, in the domain of intervals of real numbers

$\mathrm{Val}(s, x, (1, 3)) \wedge \mathrm{Val}(s, x, (2, 5))$

is perfectly consistent, if $\mathrm{Val}(s, x, (a, b))$ means

$\exists r \in (a, b) \subseteq \mathbf{R}\ \ \mathrm{Val}(s, x, r)$.

(In this case, we may want to infer a relation weaker than equality, namely that the two values have a non-empty intersection).

New behavior models can be obtained from an existing one in two ways:

- by modifying the relation $R \subseteq \mathrm{DOM}(\underline{v})$ to some relation R' in the same representation. Of course, if this transformation is not the identity, the property of a strong behavior model is lost, while a complete model property may survive,

- by transforming the representation space:
  τ: $\mathrm{DOM}(\underline{v}) \rightarrow \mathrm{DOM}'(\underline{v}')$
  and, thus, obtaining a relation $R' = \tau(R) \subseteq \mathrm{DOM}'(\underline{v}')$ from $R \subseteq \mathrm{DOM}(\underline{v})$.

The former allows us to directly express a variety of common modifications of descriptions of system behaviors, such as linear approximation (replacing R by the graph of a piecewise linear function) and introducing tolerances (by expanding R). The latter provides the basis for exploiting other, equally frequently applied, mappings between different representations, for instance switching from Cartesian to polar coordinates. In [Struss 92], we illustrated such mappings also by structural aggregation (by dropping internal variables) and mapping domains into equivalence classes (e.g. real numbers to intervals between landmarks) using a real world example.

Again, the question is raised what preconditions ensure the preservation of behavior model properties. A quite obvious and basic conditions turns out to suffice, namely that the Val-predicate is preserved in both directions:

Definition 2.2 (Representational Transformation)
A mapping
τ: $\mathrm{DOM}(\underline{v}) \rightarrow \mathrm{DOM}'(\underline{v}')$
is a representational transformation, iff
$\mathrm{Val}(s, \underline{v}, \underline{v}_0) \wedge \underline{v}_0 \in \mathrm{DOM}(\underline{v}) \quad \Rightarrow \quad \mathrm{Val}(s, \underline{v}', \tau(\underline{v}_0))$
and
$\mathrm{Val}(s, \underline{v}', \underline{v}'_0) \wedge \underline{v}_0 \in \mathrm{DOM}'(\underline{v}') \quad \Rightarrow \quad \exists \underline{v}_0 \in \tau^{-1}(\underline{v}'_0)\ \mathrm{Val}(s, \underline{v}, \underline{v}_0)$.

Intuitively speaking, the second condition states that the target domain does not introduce values that are not grounded in the original one (Actually, this definition becomes slightly more complicated for domains that allow for multiple values, such as intervals, see [Struss 94]).

Theorem 2.1
If τ: $\mathrm{DOM}(\underline{v}) \rightarrow \mathrm{DOM}'(\underline{v}')$
is a **representational transformation**, then the **image of a strong model is a strong model**:
$B(R) \vdash B(\tau(R))$.

(For the preservation of complete behavior models the second condition of Def. 2.2 is sufficient; see proof in [Struss 94]). Obviously, we are getting closer to our concept of an abstraction transformation.

Definition 2.3 (Abstraction)
An abstraction is a **representational transformation** that is
- **surjective** and
- **not injective**.

So far, in previous presentations of our theory and in this paper, we used static views on physical systems to illustrate the concepts. It is now time to ask whether modeling dynamic systems and formalizing temporal abstraction can be handled within this framework.

Unfortunately, there seem to be quite different methods to be subsumed under the heading of temporal abstraction. Describing behaviors as sequences of states (as done in qualitative simulation), viewing a slow process as providing constant conditions from the perspective of a faster one ([Kuipers 87], [Iwasaki 92]), identifying a behavior as cyclic ([Weld 86]), and characterizing a signal as changing at least once in an interval ([Hamscher 91]) appear to be quite different operations, but still share that some aspect of time has been "abstracted away". Here, we focus on the kind of abstraction" discussed in the introduction (others are discussed in the long version of this paper).

3 Behavioral Abstraction of Relational Models

As a matter of fact, the properties and their transformations we considered in the previous section are not very specific for time, but could be related at least to any variable that is considered to have the real numbers as the ultimate ground representation. So, one might be tempted to simply introduce time as another variable in the vector $\underline{v}$:
$\underline{v} = (v_1, v_2, \ldots, v_k, t)$

and describe the behavior over time by a relation

$R \subseteq \mathrm{DOM}(\underline{v}) := \mathrm{DOM}(v_1) \times \mathrm{DOM}(v_2) \times \ldots \times \mathrm{DOM}(v_k) \times T$.

In order to demonstrate that this does not provide a satisfactory solution, we go back to the problem of an abstraction transformation that maps different oscillations into a general qualitative concept represented by the sequence (..., 0, +, 0, -, 0, +, ...), or, as a relation,

$$R_{Q_3} = \{(0, t) \mid t = 2n, n \in N_0\} \cup \{(+, t) \mid t = 4n+1, n \in N_0\} \cup \{(-, t) \mid t = 4n-1, n \in N_0\}$$
$$\subseteq \{0, +, -\} \times N_0 = Q_3 \times N_0 = \mathrm{DOM}'((x, t)) .$$

Let us assume that two possible behaviors are described in the ground representation

$\mathrm{DOM}((x, t)) = [-1, 1] \times \mathbf{R}_0^{+}$

by

$R_R = \{(\sin t, t) \mid t \in \mathbf{R}_0^{+}\} \cup \{(\sin 2t, t) \mid t \in \mathbf{R}_0^{+}\}$

(Fig. 3.1).

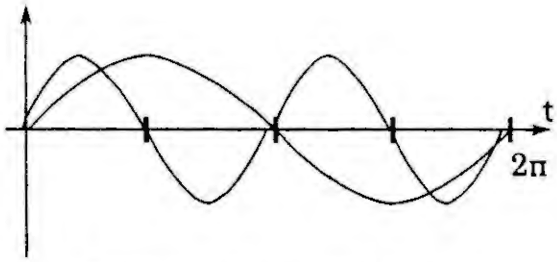


Figure 3.1 (sin t, t) and (sin 2t, t)

We notice that it is impossible to find a mapping

$\tau_t: \mathbf{R}_0^{+} \rightarrow N_0$

of the temporal spaces such that, in conjunction with the qualitative domain abstraction

$\tau_q: [-1, 1] \rightarrow Q_3$,

it forms a representational transformation

$\tau = (\tau_q, \tau_t): [-1, 1] \times \mathbf{R}_0^{+} \rightarrow Q_3 \times N_0$.

For instance, the tuple (0, 2π)∈DOM((**x**, t)) is shared by both oscillations. But for sin t , it has to be mapped onto (0, 2), whereas sin 2t reaches 0 for the fourth time, and, hence, (0, 2π) corresponds to (0, 8). In other words, the temporal transformation is not a unique one, but may be specific for each single behavior. again, we see that the kind of temporal abstraction we are aiming at cannot be achieved by treating time merely as an additional parameter. The reason is that what we want is **behavioral abstraction**, and it might seem that we have to drop our claim that abstraction is to be considered as a general mapping between representational spaces rather than tied to particular behaviors.

However, for the relational models, there are no restrictions imposed on the choice of the domains for the variables. They need not be sets of real numbers, integers, or qualitative values, as before. They can also be **sets of functions over time**. So, if T is some temporal universe, and $\mathrm{DOM}_v(v_i)$ denotes the set of possible values v_i can take in principle at each time instance, let

$F(T, \mathrm{DOM}_v(v_i)) = \{ f \mid f: T \rightarrow \mathrm{DOM}_v(v_i) \}$

be the set of functions in $\mathrm{DOM}_v(v_i)$ over time and

$F(T, \mathrm{DOM}_v(\underline{v})) = F(T, \mathrm{DOM}_v(v_1) \times \mathrm{DOM}_v(v_2) \times \ldots \times \mathrm{DOM}_v(v_k))$.

If s_t is the “time slice” corresponding to $t \in T$ in situation $s \in \mathrm{SIT}$, then

$\forall f \in F(T, \mathrm{DOM}_v(\underline{v})) \quad (\mathrm{Val}(s, \underline{v}, f) \Leftrightarrow \forall t \in T \ \mathrm{Val}(s_t, \underline{v}, f(t)))$.

Then we can describe behaviors over time in the representational space

$\mathrm{DOM}(\underline{v}) = F(T, \mathrm{DOM}_v(\underline{v}))$,

and an abstraction will be a transformation of the function spaces. (Note that we assume the same temporal representation for each variable). This allows us to correctly handle some kinds of temporal abstraction, such as the “oscillation abstraction”.

As a first step, we show that the “traditional” domain abstraction

$\alpha: \quad \mathrm{DOM}_v(\underline{v}) \quad \rightarrow \quad \mathrm{DOM}'_v(\underline{v})$

induces a behavioral abstraction.

We can map a function

$f: T \to DOM_v(\underline{v})$

onto its composition with α:

$\tau_\alpha: \; DOM(\underline{v}) = F(T, DOM_v(\underline{v})) \; \to \; DOM'(\underline{v}) = F(T, DOM'_v(\underline{v}))$,

where

$\tau_\alpha(f) := \alpha \circ f: T \to DOM'_v(\underline{v})$

and

$\alpha \circ f(t) := \alpha(f(t))$.

τ_α collapses functions that differ only in values that are mapped to the same image, into one function on the abstract domain. More generally:

Lemma 3.1

If $\alpha: DOM_v(\underline{v}) \to DOM'_v(\underline{v}')$

is an abstraction, then

$\tau_\alpha: \; DOM(\underline{v}) = F(T, DOM_v(\underline{v})) \; \to \; DOM'(\underline{v}') = F(T, DOM'_v(\underline{v}'))$

is an abstraction.

For instance, all functions k*sin t, $k \in (0, 1]$, are mapped to the same function which has the value + over $(0, \pi)$, 0 at π, - over $(\pi, 2\pi)$ etc. (see first mapping in Fig. 3.2). So far, no property of time has been lost. But temporal abstraction may now be applied by collapsing maximal time intervals in which no changes in values occur into single time instances t' of a new temporal space, T'_f, which is still specific for each function f. Formally, we define

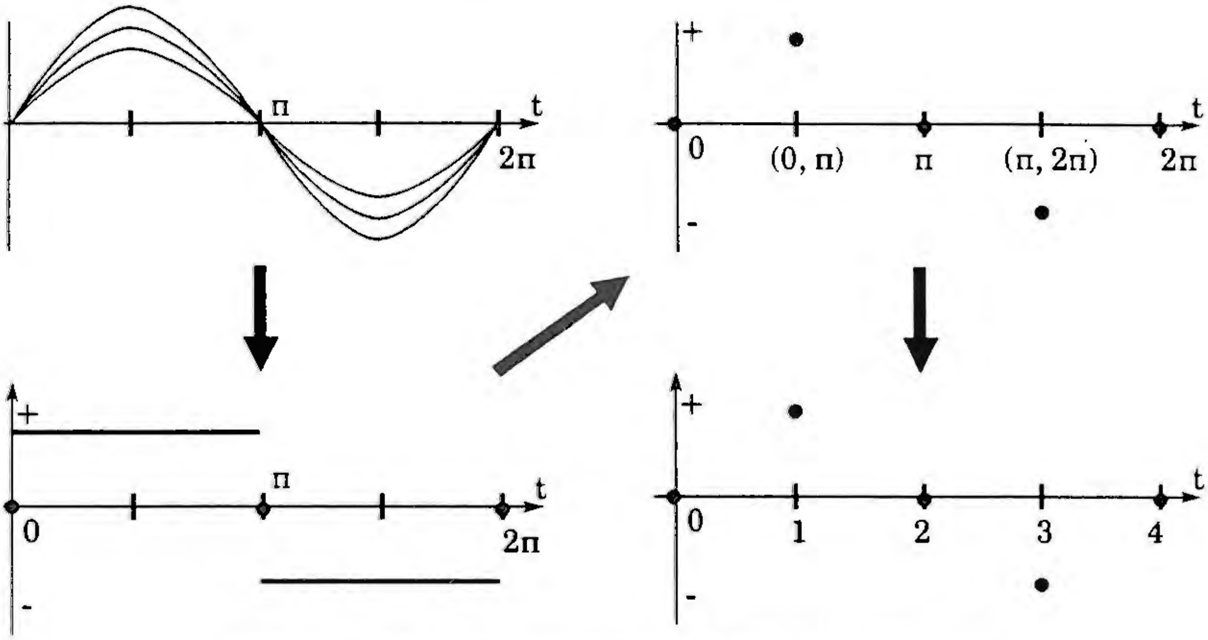


Figure 3.2 The steps in constructing the oscillation abstraction

$T'_f := \{ i \in I(T) \mid \exists \underline{v}'_0 \in DOM'_v(\underline{v}') \; f(i) = \{\underline{v}'_0\} \wedge (i' \supsetneq i \Rightarrow f(i') \neq \{\underline{v}'_0\}) \}$,

where I(T) is the set of intervals of T. Then the function

$f': \; T'_f \to DOM'_v(\underline{v}')$

is well-defined by

$f'(t') := f(t)$ for some $t \in t'$.

This is the second mapping illustrated in Fig. 3.2. The spaces T'_f still contain the metric information of T. But under certain reasonable assumptions (that exclude pathological cases such as sin 1/t), there exists a subset $\mathbf{Z}'_f \subseteq \mathbf{Z}$ of the integers and a bijective mapping (the third one in Fig. 3.2)

$\tau_f: \; \mathbf{Z}'_f \to \mathbf{T}'_f$

for each f. Defining

$\tau'(f) = f' \circ \tau_f: \; Z' \to DOM'_v(\underline{v}') = [-1, 1]$

as a (partial) function on $\mathbf{Z}' = \cup \mathbf{Z}'_f \subseteq \mathbf{Z}$, we can obtain the desired representational transformation

τ': $F(\mathbf{R}_0^+, [-1, 1]) \rightarrow F(N_0, Q_3)$

that maintains the ordering, eliminates metric properties, and performs the "oscillation abstraction". It is also the basis for other temporal abstractions, e.g. used in XDE ([Hamscher 91]). For instance, the predicate that counts the number of changes, maps f to $|\mathbf{Z}'_f|$ -1.

More generally, we have

Lemma 3.2

If α: $DOM_v(\underline{v}) \rightarrow DOM'_v(\underline{v}')$

is an abstraction, and f', $\tau_{f'}$, and τ' are well-defined as above, then

τ': $DOM(\underline{v}) = F(T, DOM_v(\underline{v})) \rightarrow DOM''(\underline{v}') = F(N_0, DOM'_v(\underline{v}'))$

is an abstraction.

4 Conclusions

Our theory of relational descriptions of system behavior allows us to express common transformations of models and representations easily. It can also be used to formalize at least some kinds of temporal abstraction if we introduce spaces of functions over different universes of time as domains that can be subject to transformations.

All this is but a first step towards a systematic analysis of problems and techniques of temporal abstraction. Also, while the theory of multiple relational models has already helped to tackle practical problems involving models of static devices, this remains to be explored for the dynamic case. This is true especially when the behavior description of components is not a priori given as sets of functions over time, but in terms of (ordinary or qualitative) differential equations, as is the case in most qualitative physics systems.

Temporal abstraction is too important as a means for complexity reduction, and we cannot afford to neglect it or treat it too generally in the ongoing discussion about multiple modeling and automatic model generation.

Acknowledgements

Thanks to Oskar Dressler, Hartmut Freitag, Gerhard Friedrich, Roy Leitch, Wolfgang Nejdl, partners in the ARTIST project, and the reviewers for comments on this paper and related ones. This work was supported in part by BMFT (ITW 8506 E4, ITW 9001 A9), CEC (P5143, 3249), and CISE.

References

[Giunchiglia-Walsh 92] Giunchiglia, F., and Walsh, T., *A Theory of Abstraction,* In: Artificial Intelligence, 57(2-3), 1992

[Hamscher 91] Hamscher, W., *Modeling Digital Circuits for Troubleshooting,* in: Artificial Intelligence, 51(1991)

[Hobbs 85] Hobbs, J.R. , *Granularity,* IJCAI-85

[Iwasaki 92] Iwasaki , Y., *Reasoning with Multiple Abstraction Models.* In: Faltings, B. and Struss, P. (eds.), Recent Advances in Qualitative Physics, 1992

[Kuipers 87] Kuipers, B., *Abstraction by Time-Scale in Qualitative Simulation.* AAAI-87

[Struss 92] Struss, P. *What's in SD? Towards a Theory of Modeling for Diagnosis.* In: Hamscher, W., de Kleer, J., and Console, L. (eds.), Readings in Model-based Diagnosis, San Mateo, 1992.

[Struss 94] Struss, P. *Multiple Models of Physical Systems - Modeling Intermittent Faults, Inaccuracy, and Tests in Diagnosis.* To appear in: Annals of Mathematics and Artificial Intelligence, 1994.

[Weld 86] Weld, D.S., *The Use of Aggregation in Qualitative Simulation.* In: Artificial Intelligence 30(1), 1986

[Weld-Addanki 92] Weld, D.S. and Addanki, S., *Task-Driven Model Abstraction.* In: Faltings, B. and Struss, P. (eds.), Recent Advances in Qualitative Physics, 1992

Qualitative Modelling based on Rules, Petri Nets, and Differential Equations

M. Kluwe, V. Krebs[1]
University of Karlsruhe
Institut für Regelungs- und
Steuerungssysteme

J. Lunze, H. Richter[2]
Technical University Hamburg–Harburg
Control Engineering Department

1. Introduction

Increasing the degree of automation for complex technical processes is an important challenge of today's system and control engineering. The tasks in view are not only closed-loop and open-loop control but also the automated start-up and shut-down phases of the process as well as supervision and fault diagnosis. Thus knowledge-based components known as *model-based consulting systems* or *decision support systems* have to be added to conventional process control equipment.

The modelling of complex systems usually refers either to quantitative or to qualitative knowledge. However this yields disadvantages:

On the one hand, the representation of the plant by exact quantitative models is often generally feasible only within a small environment of its working point. Furthermore quantitative modelling causes high costs or is even impossible. On the other hand, using only qualitative descriptions of the plant always results in consiberable uncertainties that may be too large for analysis and control design.

In order to achieve an overall optimum, both kinds of models are combined in the *three-layer-model* proposed in this paper. Not only its function and implementation but also an example for its technical application will be described in more detail in the further sections.

2. The Three-Layer-Model

The *structure of the model* is shown in Figure 1. It is subdivided into a quantitative part represented by the lower layer and a qualitative part consisting of the intermediate and upper layer. Arrows indicate the possible interactions between the different layers of the model and the external inputs and outputs. In accordance with the different characteristics of the model layers the corresponding inputs and outputs vary.

The qualitative layer I is represented by *structured Petri nets* which have been used so far only for the modelling of discrete-event systems and the design of open-loop controllers [1], [2]. Our interpretation of Petri nets results in a *qualitative modelling of continuous-time systems*: Each place in the Petri net is assigned to a specific quality of the considered plant. If a place is marked by a token, this quality is part of the plant's current state. Hence the entire marking represents the qualitative state variables. A

[1] Prof. Dr. Ing. Volker Krebs, Dipl.-Ing. Mathias Kluwe, Universität Karlsruhe (T.H.), Institut für Regelungs- u. Steuerungssysteme, Kaiserstr. 12, D-76128 Karlsruhe Tel. (+721) 608-3180. Fax -2707 e-mail: JC17@ibm3090.rz.uni-karlsruhe.de

[2] Prof. Dr. Ing. Jan Lunze, Dipl.-Ing. Henrik Richter, Technical University Hamburg–Harburg, Control Engineering Department, D-21071 Hamburg Tel. (+40) 7718-3215, Fax -2573, e-mail: Lunze@tu-harburg.d400.de, Richter@tu-harburg.d400.de

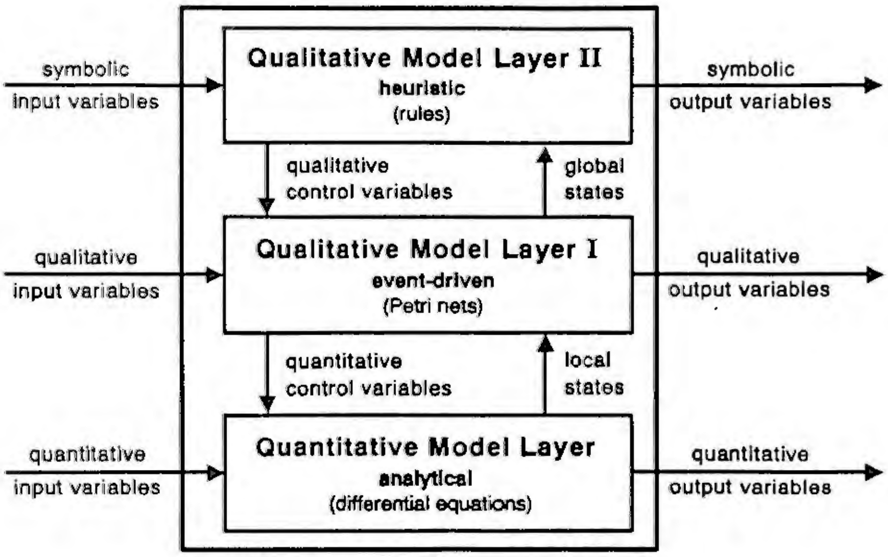


Figure 1: Three-layer-model

subset of them can be considered as the qualiative output variables. All possible changes of the qualitative state are modelled by the transitions. Their external activation-conditions constitute the qualitative input variables of this layer.

The **quantitative model layer** consists of a set of *differential equations* used as analytical descriptions of well-known properties of the plant. Generally they can be of nonlinear type but commonly linear equations will be used:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad \mathbf{x}(0) = \mathbf{x}_0 \quad , \\ \mathbf{y} &= \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \quad . \end{aligned} \tag{1}$$

As usual, $\mathbf{x}$ indicates the state variables, whereas the input and output vectors $\mathbf{u}$ and $\mathbf{y}$ correspond to the input and output variables of the quantitative model layer in Figure 1.

The **upper qualitative layer II** of the model is to represent heuristic knowledge about the plant as a set of *rules* like

`IF antecedent THEN consequence.`

The antecedents of the rules reflect a more global description of the plant. The symbolic input variables are determined by its interaction with both the environment (measured variables) and the operator. The consequences express the more strategic state changes which can not be derived immediately from the lower quantitative and qualitative layers of the model. They are equivalent to the symbolic output variables.

The three-layer-model ist characterized first by the *close interactions* among the three completly different parts of the model:

The Petri net as central part of the model is influenced by different inputs linked to its transitions. These inputs, external or coming from the other layers, are represented by Boolean expressions. Depending on the truth value of the expressions, the token flow and the resulting marking of the Petri net is controlled. This internal control of the model layers is necessary to resolve *conflicts*. They are caused by non-determinisms which are an inherent characteristic of every qualitative model. Thus the resulting uncertainty can be reduced by means of quantitative and heuristic information as follows:

The process phases which can be described by (1) are coupled with respective places of the Petri net. Marking these places with a token invokes their simulation providing a more precise description of the process dynamics. The computation stops depending on predefined criteria like exceeding limiting times or values. Passing the specific result back to the intermediate layer, this result is transformed into a Boolean expression which influences the further token flow in the Petri net and resolves the conflict.
If the conflict can be solved neither by proceeding the qualitative input variables nor by activating a quantitative model in the lower layer the problem is transfered to the upper layer. An inference engine will be started to look for rules which match the current situation. In this case, as a consequence convenient algorithms are invoked to produce Boolean expressions in order to clear the conflict. Otherwise a dialog with the operator is started, represented by the symbolic input variables in Figure 1. They are interpreted as answers to questions about conflict situations the model can not handle by itself.

3. Example: Modelling of a nuclear reactor

As a technical application consider the model of the starting-up phase of a nuclear power plant [4] shown in Figure 2. The Petri net comprises as the result of the operator's global analysis the qualitative

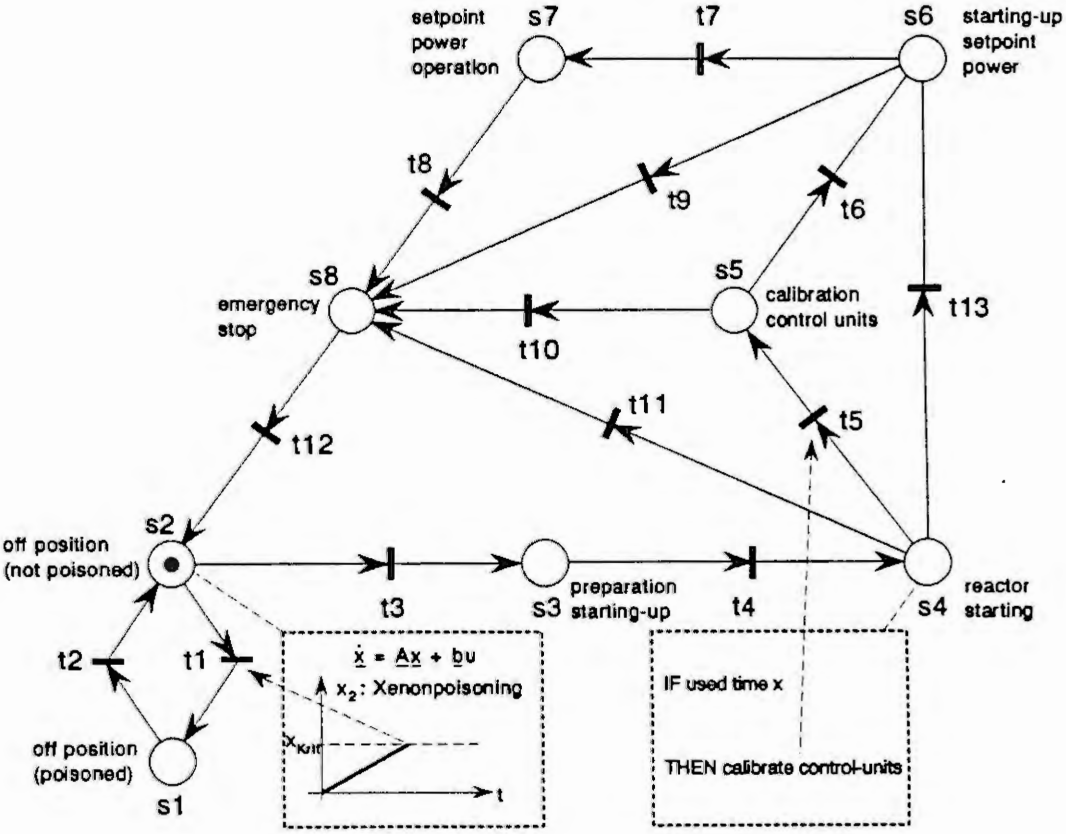


Figure 2: Global phases starting-up a nuclear power plant

states s_1-s_8 and the transitions t_1-t_{13}. As a global description, this Petri net is a useful example for the intermediate qualitative layer of the three-layer-model.
The necessity of the quantitative model layer becomes obvious by consideration of place s_2. The marking of this place, describing the shut-off state of the reactor causes a conflict with respect to the transitions

t_1 and t_3. To retract this conflict, it is necessary to observe the reactor's poisoning state caused by high Xenon concentration. Although it is not possible to get the degree of poisoning by measurement, there exists an approximate state space model like (1) to estimate it. Depending on the current situation, the simulation of the poisoning x_2 exceeds its critical value x_{krit} or not. If $x_2 > x_{krit}$, the computation stops and t_1 fires causing the token to move from s_2 to s_1. Otherwise the computation stops after reaching the steady state of x_2 and the token stays in s_2. Hence the result of the simulation provides the essential information for the operator whether to begin the starting-up or not.

The influence of the heuristics of the upper layer is demonstrated by consideration of place s_4. If it is marked, the reactor is ready for start-up. The conflict among the following transitions t_5, t_{11}, and t_{13} expresses the operator's decision whether to calibrate the control units first or to begin immediately with the start-up. This decision depends on the history i.e. the previous performance of the reactor. So the conflict can be retracted by defining the firing condition for t_5 with the following rule:

```
IF the control-units have been used since time x THEN calibrate them.
```

4. Implementation

The presented three-layer-model has been designed for a current research project that aims at constructing a *model-based consulting system* for a nuclear power plant [4].

Both the Petri nets and the rules as the qualitative parts of the model are embedded in an environment of the *expert system shell* Goldworks II [3]. Its knowledge base is structured into an object-orientated hierarchy of *frames* and their *instances*. So the places and transitions of the Petri nets and general variables are defined as frames, their instances are given by the concrete modelling of the plant representing the current process state.

From the knowledge base, new facts can be derived by the included inference engine by firing *rules*. This leads to the change of the model's qualitative state (i. e. the corresponding transitions fire and produce the new marking).

The quantitative layer of the model is implemented by a separate software module. As described above, the *on-line simulation* is started by the marking of places of the Petri net which are to simulate. Depending on predefined criteria it stops causing a state change of the qualitative model.

The data exchange between the two programs is organized by software interfaces.

5. Conclusions

The proposed three-layer-model provides a powerful means for describing complex systems using as well qualitative as quantitative kinds of knowledge. Hence this model will be the key element of a model-based consulting system to be built.

References

[1] Abel, D.: Petri-Netze für Ingenieure. Berlin Heidelberg: Springer-Verlag, 1990.

[2] König, R.; Quäck, L.: Petri-Netze in der Steuerungstechnik. Berlin: Verlag Technik, 1988.

[3] GoldWorks II for the Sun Workstations, User's Guide. Gold Hill Computers, Inc., 1989.

[4] M. Kluwe, V. Krebs, J. Lunze, H. Richter: Ereignisdiskrete Modellierung kontinuierlicher Prozesse: Konzeption und Erprobung in einem Beratungssystem für den Rossendorfer Forschungsreaktor. Zwischenbericht des DFG-Projektes, Technische Universität Hamburg-Harburg, Arbeitsbereich Regelungstechnik, 1993.

Qualitative Modeling for Control Synthesis and Diagnosis

Klaus Nökel
SIEMENS AG, Corporate Research & Development, Formal Design Methods
81730 München, Germany
e-mail: noekel@zfe.siemens.de

Abstract: When dealing with controlled systems qualitative model-based reasoning should not be confined to the physical principles governing the controlled process but encompass reasoning about the controller. In particular, generation of control software becomes an integral part of systems design. We argue that qualitative, compositional models should be used to automatically synthesize control software. In addition we propose to reuse the same models to make use of information about intended function (purpose) within fault diagnosis.

1 Introduction

The role of qualitative model-based reasoning in systems engineering is generally seen [2], [3] as a general purpose framework for modeling artifacts[1] and answering some kinds of questions about their behavior that may arise in engineering problem solving. Many important design criteria follow directly from this aim:

1. Qualitative representations are used to focus on significant distinctions in quantity spaces. This way predictions do not depend on specific numerical parameters of an artifact which is essential e.g. in early design stages where a tentative design is evalutated before all parameters are fixed..
2. Models are compositional in that aggregates are formed from simpler components, and the aggregate behavior follows from the interaction of the component behaviors.
3. Models separate the description of component type behavior from the structural description of an individual artifact. All component type descriptions are collected in a library which can be used to construct models for arbitrary plants built up from instances of the component types.

Despite the generality of the approach we observe that the vast majority of published work is concerned with fault diagnosis, the rest dealing mainly with design. To date practically all systems use a model for only one particular application task (e.g. diagnosis *or* design). Furthermore researchers have emphasized the representation and reasoning about the physical laws ("first principles") accounting for the behavior of an artifact.

This paper proposes a generalization of this state of affairs by

- introducing a novel application task to be supported: synthesis of control software,
- specifying purpose-dependent constraints on control behavior in a qualitative model, and
- demonstrating how such a model can provide additional information for fault diagnosis.

2 Controlled Systems

The problem with limiting oneself to physical principles is that today only very few processes of industrial relevance are left to the laws of nature alone; most are controlled in some way with software increasingly taking the role of the controller. The overall behavior of a controlled system depends critically on the controller which therefore must not be excluded from consideration. An example: assume that we want to construct a simple elevator system (Figure 1). Working out the electromechanics of cabin, cable, motor, winch etc. is an important, but not the

[1] Throughout this paper we use "artifact", "plant", and "process" as synonyms.

only part of the job. The elevator system will serve its purpose only, if the control software is correctly designed and hooked up to the hardware via sensors and actuators.

In the same manner many potential faults of an elevator system can only be addressed if both process and control are considered. An elevator may get stuck due to electromechanical problems, but also due to a fault in the controller (including sensors and actuators). Furthermore, a robust controller may mask faulty process behavior by taking compensatory actions.

The main consequence of this observation is that design and diagnosis of systems should take into account the control aspect explicitly (if present) and reason about both in the same framework. In particular, construction of the control software should be a part of the design of the controlled system. On a closer look, the most important arguments for using a model-based approach for design (generalization, reuse) also apply to the construction of control software. Therefore we may ask whether it is possible to reduce the development costs not only for hardware, but also for the software part of a system by synthesizing the control software from a model of the control task.

3 Hybrid Systems

It is generally accepted that program synthesis in the general case is hopelessly complex; for this reason any approach to synthesis of control software will have to limit itself with respect to the kinds of models used for specification and with respect to the abstract target machine. Although the process to be controlled is usually described in terms of continuous variables and a set of differential equations, this kind of system descriptions is not well-suited to automated synthesis of control software. Firstly, synthesis requires extensive symbolic manipulation, and secondly, the analytic system description contains more detail than is needed for the synthesis of the control logic.

Just as in qualitative modeling for design or diagnosis, it is useful to abstract from the detailed system description and derive from it a discrete description of the control software in which only those distinctions are kept that have a bearing on the control decisions. This type of system description which combines a continuous process model with a discrete model of the control has been studied extensively in the literature under the name of hybrid systems [4], [5], [7]. It has proved successful mainly for its analytic properties (cf. the work on simulation of discrete-event

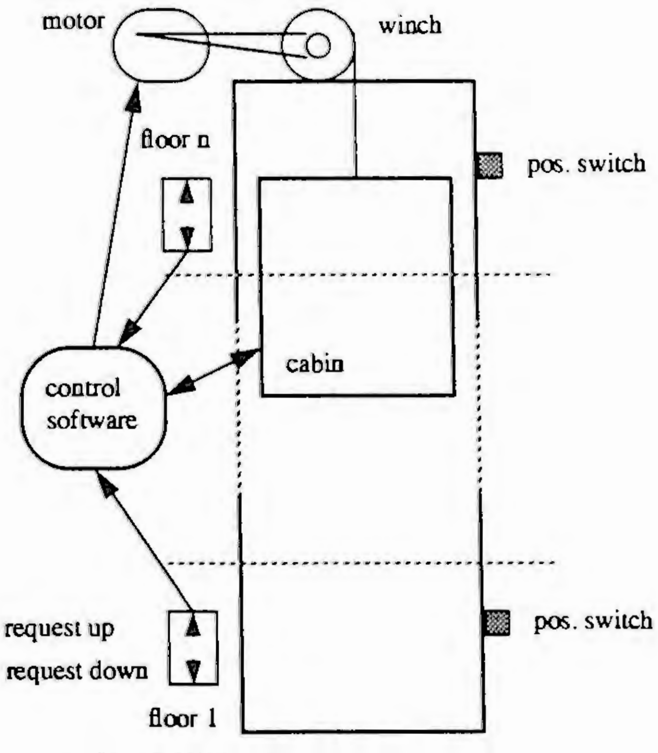


Figure 1 Parts of a simple elevator system

systems [6]). In the work cited the control is frequently assumed to be a finite automaton which is amenable to both informal and formal [8] analysis.

However, for control tasks of industrial size automata tend to either grow very large, if a centralized control is used, or - in a distributed control - the communication channels between individual automata rapidly multiply with the number of automata . In either case manual construction of the finite state machine quickly becomes impossible. To get the benefits of hybrid systems even for large-scale applications, we have to be able to automatically construct the control part from a specification that can be handled more easily by humans.

4 Qualitative Models for Control Synthesis

Our design goals for control synthesis are remarkably similar to those for model-based qualitative reasoning in general:

1. In the application domains that we studied, control software is repeatedly written for similar tasks differing mainly in the compositional structure of the controlled plant. For elevator systems, the numbers of floors and shafts may vary, for railway signalling applications the track layout may be different in each case. We want to factor out the part of the specification that is common to such a family of control tasks; each individual system description merely adds the (much smaller) structure description in the form of a labelled graph.
2. An intuitive representation of plant structure enumerates the component instances and defines how they are interconected. Since control software depends on a particukar plant configuration, we use the same structure description (i.e. in terms of *plant* components) for control synthesis as well.
3. Unlike for diagnosis the component library does not describe the behavior of the components itself; instead it specifies for each component type which constraints the *control* behavior must meet for each instance present in the plant. These constraints are instantiated according to the structure description and from these state transition function and output function of the finite automaton are computed.
4. The component library must be general enough to be used in conjunction with hardware of different makes; it therefore abstracts from many of the numerical parameters and embodies only the qualitative control behavior.

Our method of synthesizing control software from a model is explained in the companion paper [9]. Here we examine the models further.

Consider again the simple elevator system. Modeling starts with the identification of relevant plant component types. Obvious candidates are floors and cabins (or shafts), since individual plant structures will differ in these. Further components are added, if they need to be controlled (e.g. the motor) or if they are the origin of events that influence the control (e.g. the push buttons).

Next we decide how to model the control behavior for each component type. Each type description specifies a set of input symbols, a set of output symbols, a set of control state attributes, and a set of partial constraints on the state transition and output functions, called "transitions". Input symbols correspond to events in the process that require a reaction of the controller. Examples for input symbols are `requestUpPressed` for type `floor` and `floorReached(N)` (N = 1, 2, ...) for type `cabin`. While the former event is by nature discrete, cabin position is originally a continuous variable. Control actions are, however, only required at finitely many positions (the floors). Hence we recognize only those position measurements as input symbols that may trigger a control action. Here qualitative abstraction is essential, because the union of type-specific input symbols will form the (finite) input alphabet of the automaton.

Output symbols correspond to control actions. In our example we view the continuous fine control of the cabin position (e.g. stopping exactly at floor level at a convenient deceleration) as part of the process. Hence, control actions are discrete, such as `stop`, `goUp`, and `goDown` for the `motor` or `openDoor` and `closeDoor` for the `cabin`.

Almost always, the control reaction to a given input depends on previous events. This information is recorded in type-specific state attributes, e.g. `position` and `direction` for the `cabin`. Again qualitative abstraction is used wherever discrete values are sufficient for control decisions. In our example, the lowest possible resolution for position is the domain `{floor1, (floor1;floor2), ..., (floorN-1;floorN), floorN}`. Again, if we want to synthesize a classical Mealy automaton, it is necessary to specify discrete domains for all state attributes, as the cartesian product of the domains will form the (finite) state space of the automaton.

For each component type the control behavior itself is defined by a set of "transitions". Each transition is a triple (*input, condition, output*) where the condition is a propositional formula denoting a relation on two successive control states. If the controller receives *input* and there is a future control state which together with the current control state satisfies *condition*, then one such successor state is selected and *output* is emitted. Transitions should not be confused with rules: first, multiple transitions for the same input are applied simultaneuously and secondly, transitions may be nondeterministic[1]. Example: The `cabin` description may specify the transition

```
(floorReached(Self, N),
 direction(Self) is up /\ moves(Self) is yes /\ requestUp(N) /\
 next (moves(Self) is no /\ door(Self) is open),
 [stop(motor(Self)), openDoor(Self)])
```

which stops the elevator at floor N in case it needs to pick up a person.

Figure 2 sums up the comparison between models used for diagnosis and for control synthesis.

Figure 2 Comparison between models for diagnosis and control synthesis

	component oriented model for diagnosis	**model for control synthesis**
organisation	separate component library (types) from structure description of individual plant (instances)	dito
component library	set of component types with description of physical behavior	set of component types with description of control behavior
state concept	quantities, often with qualitative domains	structured state with state attributes, each having a finite number of possible values
description of behavior	relational model based on quantities	propositional formulae based on state attributes
abstractions	qualitative values	dito
structure description	instances of component types, connected via ports	instances of component types, connections given as graph labelled with instance names

5 Towards a Representation of Purpose in Diagnosis

Despite their structural resemblance a fundamental difference remains between the two kinds of models. While the component library of a plant model describes the physical principles governing the process behavior, our models specify the *intended* behavior of the controlled system. In which sense can our model of a controller be regarded as a statement of purpose? In the strict sense the purpose of the elevator system is to transport people, picking them up and setting them down at different floors. This purpose is only implicit in the controller model. Instead, control models describe the purpose of the controlled system from the perspective of the control actions necessary to force the system behavior to one particular of the physically possible trajectories: one that achieves the purpose.

[1] The synthesis procedure picks one representative from the set of all deterministic automata satisfying all transitions.

What can we gain by reasoning explicitly about intended behavior in fault diagnosis?

1. In a controlled system, faults are not meaningfully characterized as "anything giving rise to observed behavior that contradicts the behavior predicted by the physical plant model". Again the control must be taken into account, too. Most people would call it a fault, if they saw an elevator crash into the bottom of the shaft at full speed, even though this may happen due to a faulty controller and in perfect accordance with the physical laws governing motor, winch, cable and cabin.
2. Tweaking the physical plant model so that unwanted behaviors are not predicted any more is not only conceptually problematic, since by definition physical laws cannot be changed in this way. Reusability of the model would also be seriously impaired: the very same behavior may actually be *intended* in a shaft designed to conduct experiments under micro-gravitation [1]. Thus, faults have to be characterized relative to purpose, purpose is achieved by forcing the system to the intended behavior, and intended behavior must be represented explicitly as a counterpart to the physical plant model. The "faulty" behavior violates the control model for the elevator system which specifies that the cabin should stop under conditions similar to the example in section 4. It may not violate a control model for the free-fall shaft.
3. Diagnostic procedures based on qualitative simulation are often plagued by ambiguities in the envisioned behavior, especially when reasoning about temporally distributed fault symptoms. Through control actions (setting certain variables) the control constantly influences the process, thereby acting as a powerful global filter on the envisioning of the plant behavior. If e.g. for complexity reasons the diagnostic procedure does not reason about time and is applied separately to observations at several time instants, a trace of the control behavior may be used to relate the findings at each time instant. Making use of "observations" within the control is especially attractive, because the "state variables" in the control are simply program variables which can be accessed at (essentially) zero "measurement cost".

Fortunately, it seems quite straightforward to combine control and plant models to address these questions. Similar to the standard formulation of hybrid systems we divide the model *SD* of the controlled system into three parts

$$SD = SD_C \cup SD_P \cup SD_I$$

where SD_C and SD_P are the control and plant models, respectively, and SD_I is the interface mapping between parts of SD_C and SD_P.

The *plant* is modelled by

$$SD_P = (Q_P, q_{P0}, R_P \subseteq Q_P^*)$$

where

- $Q_P = dom(v_{p1}) \times \ldots \times dom(v_{pn}), v_{pi} \in V_P$.
- $V_P = U \cup X \cup Y$ with U input variables, X state variables and Y output variables of the process.
- $q_{P0} \in Q_P$ initial state.

The relation R_P describes the plant behavior as the set of traces of value assignments to the process variables. In practice, it is given implicitly (e.g. as a set of constraints) and the traces are computed from the constraints by a process called envisioning. In qualitative models all variables have discrete domains.

The model of the *controller* describes the control behavior as a state sequence.

$$SD_C = (Q_C, q_{C0}, R_C \subseteq Q_C^*)$$

where

- $Q_C = dom(v_{c1}) \times \ldots \times dom(v_{cm}), v_{ci} \in V_C, dom(v_{ci})$ finite.
- $V_C = I \cup S \cup O$ with I input variables, S state variables and O output variables of the controller.
- $q_{C0} \in Q_C$ initial state.

Again, in practice, R_C is given implicitly (e.g. as a finite automaton synthesized from a qualitative model), and the traces are computed by evaluation of the automaton.

The interface SD_I consists of two maps $CP: O \rightarrow U$ and $PC: Y \rightarrow I$ respectively.[1] *CP* models the influence of the controller on the plant (via actuators), *PC* models feedback from the plant (via sensors).

With these definitions it is possible to extend the notion of "envisionment" to behaviors respecting the combined model *SD*. The extended envisionment contains only those behaviors which are both physically possible and intended. Hence, it is the adequate basis for diagnosis. Other concepts in model-based diagnosis (conflicts, candidates, diagnoses) are readily adapted.

We are currently working on refining this definition, using it to model real applications, and experimenting with implementations.

6 Conclusion

Limiting the use of qualitative model-based reasoning in systems engineering to reasoning about physical laws has serious drawbacks when systems have a non-trivial control. Rather than mixing physical laws and purpose in an ad hoc way both should be represented explicitly. In analogy to physical plant models it is useful and intuitive to describe the control task in a qualitative compositional model. Automated synthesis of the control software from such a model is possible. In addition we propose to reuse the control model for a second application task (diagnosis) by inserting control and plant models into the standard framework for hybrid systems. In this respect our proposal represents a step towards making use of information about intended behavior within diagnosis.

7 Acknowledgments

This paper has benefited from fruitful discussions with Hartmut Freitag, Oskar Dressler, Hans Heller, Peter Bader, Wolfram Büttner, and Peter Struss.

8 References

[1] H.J. Rath: *Experimente unter Mikrogravitation: der Fallturm Bremen*, in: Spektrum der Wissenschaft, 10/1988

[2] D. Weld, J. de Kleer (eds.): *Readings in Qualitative Reasoning about Physical Systems*, Morgan-Kaufman, 1989

[3] B. Faltings, P. Struss (eds.): *Recent Advances in Qualitative Physics*, MIT Press, 1992

[4] J. A. Stiver, P. J. Antsaklis: *State Space Partitioning for Hybrid Control Systems*, in: Proc. American Control Conference 1993

[5] A. Benveniste, P. Le Guernic: *Hybrid Dynamical Systems Theory and the SIGNAL Language*, IEEE Trans. on Automatic Control, vol. 35, no. 5, May 1990

[6] B. P. Zeigler: *DEVS Representation of Dynamical Systems: Event-Based Intelligent Control*, in: Proc. IEEE, vol. 77, no. 1, 1989

[7] R. Alur, T. A. Henzinger, P. Ho: *Automatic Symbolic Verification of Embedded Systems*, extended abstract, AT&T Bell Labs and Cornell University (Dept. of Comp. Science), 1993

[8] T. Filkorn, *A Method for Symbolic Verification of Synchronous Sequential Circuits*, IFIP 10th Int. Symp. on Computer Hardware Description Languages and their Applications (1991), 249-260

[9] P. Bader, H. Heller, K. Nökel: *Automatic Generation of Control Software for Large-Scale Control Systems*, in these proceedings

[1] Two additional mappings *DOM-CP* and *DOM-PC* are needed to convert between the domains of O and U and between Y and I, if these are not per se compatible.

INTEGRATING QUANTITATIVE AND QUALITATIVE MODELS

Erling A. Woods
SINTEF Automatic Control,
N-7034 Trondheim
NORWAY
email: Erling.Woods@regtek.sintef.no

Abstract: The Hybrid Phenomena Theory, HPT, specifies how physical knowledge may be encoded in terms of generic definitions of physical phenomena. From a description of process topology, the HPT implementation derives a model with three components: a state space model, a topological model and a phenomenological model. The two latter components formalize important assumptions underlying the state space model. The complete model ensures that the structure of the state space model is adapted in accordance with changes in operating regime or assumptions.

1. INTRODUCTION

Different types of models reveals different subsets of the characteristic properties of a given system. Quantitative mathematical model formulations will typically produce accurate results when used for simulation or analysis, but such formulations may yield poor results if some of the assumptions underlying the model is violated. Qualitative modeling techniques typically attempt to take account of this by generating all possible solutions sanctioned by the less accurate qualitative model, [2], [3].

This paper describes the Hybrid Phenomena Theory, HPT, [5]. The HPT claims that mathematical state space models based on physical principles may be derived by relating a description of process topology, including descriptions of processing units, material, location and connections, to a set of generic phenomena definitions. Reasoning at the qualitative level may directly affect the corresponding entities at the quantitative level, and computations at the quantitative level may directly affect entities at the qualitative level. The HPT builds on the Qualitative Process Theory, [1]. The HPT has been implemented in the Common Lisp Object System (CLOS).

2. DEFINING PHENOMENA

Knowledge on properties of different types of process equipment and different material substances is encoded as a set of predefined CLOS classes. Knowledge on physical interactions is encoded as definitions of types of *views* and *phenomena*. Examples of such definitions are shown in Fig. 1. In the following, the concepts of algebraic and dynamic influences are introduced first, then follows a discussion on the purpose of the different types of statements in the definitions.

2.1. Influences

Influences express how the value of a given variable is affected by a set of other variables. There are two types, *dynamic* and *algebraic influences*. Dynamic influences specify how the derivative of the influenced variable is affected by a set of influencing variables. Algebraic influences specifies how the value of the influenced variable is affected by a set of influencing variables. The syntax is as follows:

```
(dyn-inf <influenced-variable> <list of influencing variables> <function-spec>)
(alg-inf <influenced-variable> <list of influencing variables> <function-spec>)
```

An example where a variable x_1 is dynamically influenced by the two variables x_2 and x_3 are shown in (1). The amount of influence is here computed as a nonlinear function of x_2 and x_3. If this is the only influence affecting x_1, the derivative of x_1 will be computed from the equation in (2).

$$(dyn-inf\ x_1\ (x_2 x_3)\ (sqrt(x_2 x_3))) \tag{1}$$
$$\dot{x}_1 = \sqrt{x_2 x_3} \tag{2}$$

```
(defview heat-bridge (obj1 obj2)
 (individuals
  (obj1 (is-a object-with-heat-capacity))
  (obj2 (is-a object-with-heat-capacity))
  (heat-connected obj1 obj2))
 (relations
  (define-parameter \alpha
   (:value 1200 :unit "J s-1 K-1 m-2"))
  (define-parameter \kappa
   (:unit "J s-1 K-1"
    :compfunc
     (* \alpha (MIN area(obj1)
                    area(obj2))))))
```

```
(defphenomenon heat-flow (src dst hbr)
 (individuals
  (hbr (instance-of heat-bridge(src dst)))
  (src (is-a object-with-heat-capacity))
  (dst (is-a object-with-heat-capacity)))
 (quantityconditions(> temp(src) temp(dst)))
 (relations
  (define-variable hstr
   (:unit "J s-1"))
  (alg-infl hstr (temp(src) temp(dst))
   (* \kappa(hbr) (- temp(src) temp(dst)))))
 (dynamics
  (dyn-infl temp(dst) (hstr) (/ hstr c(dst)))
  (dyn-infl temp(src) (hstr)
    (/ hstr c(src)))))
```

Figure 1: Definitions of physical interactions.

2.2. Parts of a definition

The **individuals** are listed in the first line of the definition, *src*, *dst* and *hbr* are the individuals in the definition of *heat-flow* in Fig. 1. For an instance of a definition to be created, an object must be assigned to each individual such that the objects satisfy the conditions defined in the individuals field. **Preconditions** describe modeling assumptions, including external actions, e.g. turning on or off the power supply for a hot plate. **Quantityconditions** relate to the state in the system. Together, the individuals conditions, the preconditions and the quantity conditions form both necessary and sufficient conditions for a definition to be applicable.

The **relations** are used to specify all consequences of activity except for dynamic interactions between variables. A statement in the relations field performs one of the following tasks; Define a new variable for the instance; Specify a new object to be created; Establish a logical relation; Define an algebraic influence between variables within the scope of the definition. A variable, parameter or constant is said to be within the scope of a definition if it is defined by that definition, or if it is within the scope of the definition of any object bound to one of the individuals in the definition. In addition, there exists a set of global constants which are inside the scope of all objects.

The **dynamics** field specifies dynamic influences between variables within the scope of the definition.

3. THE HPT MODEL

This paper will use the example system in Fig. 2. The HPT model consists of three components: topological model, phenomenological model and state space model. The topological model includes the entities describing the physical objects and the relationships between these. An initial version of this model is specified in the input description which consists of a number of **Defobject** and **Defrelation** statements. The example includes three defobect statement defining the PAN, WATER and HOTPLATE.

For each combination of objects which satisfy the individual conditions in a definition, an instantiation of that definition is created. To find the objects satisfying a definition, the system reasons about relations. The definition of the **heat-bridge** view requires that the two individuals be **heat-connected**. However, this relation need not be explicitly defined. The system uses a set of rules in a backward chaining manner to establish that the pan and hotplate are heat-connected from the relationship **rests-on pan hotplate** which was explicitly stated. Note that view and phenomena instantiations may be bound to ,individuals in other instantiations. The set of phenomena instantiations constitute the phenomenological model.

Fig. 3 shows all HPT-objects created for the example. Due to spatial constraints, only a subset of the definitions used are given .in Fig. 1. The syntax in Fig. 3 describes an instantiation by the name of the definition, followed by a number uniquely identifying the instantiation and the objects bound to the individuals.

Only *active* instances influence the analysis. To be active, all pre- and quantity- conditions for an instance must be satisfied, and all of the objects bound to individuals in the instance have to be active.

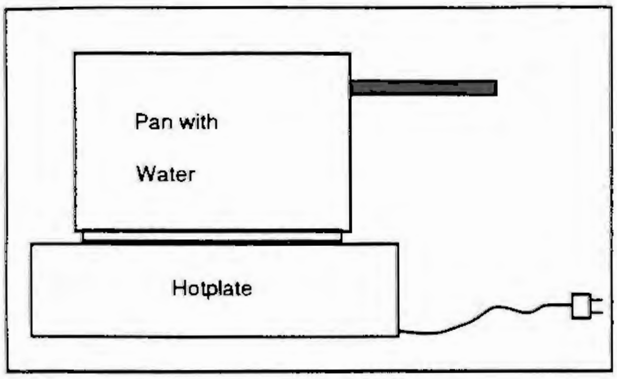


Figure 2: A simple example system.

GLOBAL
PAN
HOTPLATE
WATER
HEAT-BRIDGE-1 (PAN WATER)
HEAT-BRIDGE-2 (WATER PAN)
HEAT-BRIDGE-3 (HOTPLATE PAN)
HEAT-BRIDGE-5 (HOTPLATE CONTAINER-WITH-LIQUID-1)
HEAT-BRIDGE-6 (CONTAINER-WITH-LIQUID-1 HOTPLATE)
HEAT-FLOW-5 (CONTAINER-WITH-LIQUID-1 HOTPLATE HEAT-BRIDGE-6)
HEAT-FLOW-6 (HOTPLATE CONTAINER-WITH-LIQUID-1 HEAT-BRIDGE-5)
BOILING-1 (CONTAINER-WITH-LIQUID-1 HEAT-FLOW-6)
BOIL-V2-1 (WATER HEAT-FLOW-4)
HEAT-BRIDGE-4 (PAN HOTPLATE)
ELECTRIC-HEAT-HOTPLATE-1 (HOTPLATE)
HEAT-FLOW-1 (PAN HOTPLATE HEAT-BRIDGE-4)
HEAT-FLOW-2 (HOTPLATE PAN HEAT-BRIDGE-3)
HEAT-FLOW-3 (WATER PAN HEAT-BRIDGE-2)
HEAT-FLOW-4 (PAN WATER HEAT-BRIDGE-1)
CONTAINER-WITH-LIQUID-1 (PAN WATER)

Figure 3: Objects, views and phenomena instances.

4. REASONING WITH THE HPT

Now consider how changing the assumptions described in the qualitative part of the model will affect the resulting state space model. The view instance CONTAINER-WITH-LIQUID-1 implements an assumption that WATER and PAN will be treated as a single object in the context of heat transfer, the heat flow from the HOTPLATE must thus pass directly to CONTAINER-WITH-LIQUID-1. Assuming that the initial values for the variables have been selected in a manner causing the heat to be generated in the HOTPLATE to flow into the CONTAINER-WITH-LIQUID-1 and the liquid in the container to boil, the HPT program produces the following model.

$$\dot{x}_1 = -\frac{u_2}{h_1} \tag{3}$$

$$\dot{x}_2 = \frac{u_2}{c_1} - \frac{u_2}{c_1} \tag{4}$$

$$\dot{x}_3 = \frac{u_1}{c_2} - \frac{u_2}{c_2} \tag{5}$$

$$u_2 = \kappa_3 \cdot (x_3 - x_2) \tag{6}$$

The interpretation of the quantities are as follows: x_1 - mass of WATER, x_2 - temperature of CONTAINER-WITH-LIQUID-1, x_3 - temperature of HOTPLATE, u_1 - power consumption of HOTPLATE, u_2 - heat flow from src to dst of HEAT-FLOW-6, c_1 - heat capacity of CONTAINER-WITH-LIQUID-1, c_2 - heat-capacity of HOTPLATE, κ_3 - heat transfer per Kelvin of HEAT-BRIDGE-5, h_1 - vaporization-heat of WATER.

One aspect of handling this kind of assumptions involves preventing view and phenomena instances made obsolete by the assumption from becoming active. This is automatically prevented by what is known as the *subsumption mechanism*. Simply stated, the subsumption mechanism compares all instances of the same phenomenon. In the example, it will find that there is one heat flow going from the HOTPLATE to the PAN while at the same time there is a heat flow going from the HOTPLATE to CONTAINER-WITH-LIQUID-1 which is *binding* PAN as one of its individuals. The subsumption mechanism therefore disables the first heat flow since this is considered *less specific* than the second because the destination object of the second heat flow encompasses the destination object of the first. See [4] for a more elaborate discussion on the subsumption mechanism.

The user interface of the system enable the modeler to modify the value of any variable or parameter, and to change any explicit assumption. The assumption that PAN and WATER should be considered as one object in a heat flow context may now be revoked. This results in the following model.

$$\begin{aligned}
\dot{x}_1 &= -\frac{u_5}{h_1} && (7)\\
\dot{x}_3 &= \frac{u_1}{c_2} - \frac{u_4}{c_2} && (8)\\
\dot{x}_4 &= \frac{u_5}{c_5} - \frac{u_5}{c_5} && (9)\\
\dot{x}_5 &= \frac{u_4}{c_6} - \frac{u_5}{c_6} && (10)\\
u_4 &= \kappa_7 \cdot (x_3 - x_5) && (11)\\
u_5 &= \kappa_8 \cdot (x_5 - x_4) && (12)
\end{aligned}$$

The following quantities were not included in the previous model: x_4 - temperature of WATER, x_5 - temperature of PAN, u_4 - heat flow from src to dst of HEAT-FLOW-2, u_5 - heat flow from src to dst of HEAT-FLOW-4, c_5 - heat-capacity of WATER, c_6 - heat-capacity of PAN, κ_7 - heat transfer per Kelvin of HEAT-BRIDGE-3, κ_8 - heat transfer per Kelvin of HEAT-BRIDGE-1. Note that the state variable x_2 representing the temperature of CONTAINER-WITH-LIQUID has been removed and supplanted by the variables x_4 and x_5 representing the temperatures of the WATER and PAN respectively.

5. DISCUSSION AND CONCLUSION

The HPT demonstrates that it is possible to formalize knowledge on physical interactions in a manner allowing automatic derivation of the quantitative model. Further, the framework formalizes important assumptions about the underlying models. A program tracking the dependencies between different assumptions and the structure of the model has been implemented. The HPT extends our capability for modeling physical systems like process plants. This provides an interesting platform for future research. A significant remaining challenge is to formalize a substantial amount of the existing knowledge about physical interactions within the described framework.

6. ACKNOWLEDGMENT

This research was supported by the Royal Norwegian Council for Scientific and Industrial Research under doctoral scholarship grant no ST.10.12.220290. The author would like to thank Professor Jens G. Balchen and others at the Norwegian Institute of Technology, Division of Engineering Cybernetics and SINTEF Automatic Control for their support and inspiration.

7. REFERENCES

[1] Forbus, K.D., A Qualitative Process Theory. Artificial Intelligence, 24, 1984.

[2] Kuipers, B., Qualitative Simulation. Artificial Intelligence 29, 1986.

[3] Struss, P., Problems of Intervall-Based Qualitative Reasoning.
In: Weld, D.S. & de Kleer, J., (Eds.), Readings in Qualitative Reasoning about Physical Systems. Morgan Kaufman, 1990.

[4] Woods, E.A., The Hybrid Phenomena Theory. In: J. Mylopoulos & R.Reiter (Eds.)
Proceedings of the 12'th IJCAI, Morgan Kaufmann Publishers, 1991.

[5] Woods, E.A., The Hybrid Phenomena Theory. Dr. ing. thesis,
ITK-rapport 1993: 72-W, The Norwegian Institute of Technology, 1993.

CONTROL OF VISUAL SENSORS
WITH A
PHYSIOLOGICALLY INSPIRED NEURAL NETWORK

I. Bottemanne, J. M. Barreto, A. Roucoux
Lab of Neurophysiology, Université Catholique de Louvain, Belgium
Av. Hippocrate 54, 1200 Brussels, Belgium (e-mail: jbarreto@nefy.ucl.ac.be)

Abstract. Artificial visual sensors can have their orientation controlled using traditional structures of control systems. However such controllers lack some properties present in control systems of biological sensors, such as foveal or selective spatial attention processes and use of fused information from different sensory modalities. A model featuring a foveal zone and performing fusion of different sensory inputs (proprioceptive, visual, auditory) is presented here. The model uses physiological data as a guide in several aspects such as the network topology and the individual neuronal model. A new mechanism of coordinate transformation based on gain modulation is proposed.

1. INTRODUCTION

This work presents a model of the fusion of sensory information (proprioceptive, visual, auditory), and coordinate transformation in view of the control of orientation of visual sensors in robots. It is well known that traditional structures of control systems can be used to control the orientation of such sensors. However these controllers lack some properties present in biological systems such as selective spatial attentional processes and fusion of inputs from different sensory modalities. The model adopted uses physiological data as a guide in several aspects: a) the use of a neural network as a part of the controller; b) the neural network topology and c) the individual neuron model. It is hoped that this approach could overcome, at least partially, some of the limitations intrinsic to controllers used presently.

To model the mechanisms of fusion of different sensory inputs to control an action (the orientation of a visual sensor, for example), it is useful to divide the problem in two parts:

o As each sensor has its own system of coordinates, it is necessary to operate an appropriate transformation of coordinates to use all information available. For example, a visual sensor uses its own orientation in space as the reference of coordinates; a proprioceptive sensor uses the position of the articulation as the reference of coordinates, which are expressed in angular form.

o Sensory pieces of information sometimes are of different nature. They must be fused in order to generate a dynamic control action to be applied to the plant.

In all artificial neurons of the network, a plausible physiological model was chosen in which some important dynamic aspects of the biological neuron are considered:

- instead of using a recurrent model for the neuron, a continuous time dynamic system is adopted (first order system where the constant can be interpreted as the time constant of membrane transmission),
- the time of transmission of the pulses in the axon and the neuron is neglected (much lower than the membrane time constant),
- the output represents the frequency of discharge of the biological neuron.

These problems are solved in biological systems. In humans, the input information is generally heterogeneous: visual, auditive, proprioceptive, etc... Vision is however the most developed kind of information available. In fact, vision allows not only the identification of the attributes of objects of interest including color, shape and even sometimes the constituent material, but also its relative place in the surrounding space.

The eye is constituted by individual sensors organised in an structure called the retina. The retina is a collection of visual receptors non uniformly distributed in such a way that the resolution of the eye varies from the centre to the periphery. Only a small central part, the fovea has a large number on individual visual receptors and so, it is provided with a very good spatial resolution (retinotopic reference). Therefore the eye must be oriented appropriately to collect pertinent information.

Several different mechanisms are used to orient the eye in an efficient manner. Among them saccadic movements are used to orient the visual sensor in such way that the image of a target is brought onto the fovea very fast. This allows to select an object in the visual field, and thus to transmit the minimum necessary information. In fact, it is well known that only a small fraction of an object detected by the retina is necessary for the global perception of the object. The information transmission between the eye and the central nervous system is a communication with a limited bandwidth optimised by nature during phylogenesis as a compromise among dimension, weight and the information necessary for survival.

Audition is performed by two sensors symmetrically disposed on the head. The detection of the spatial localisation of a sound depends on intensity and phase of the signal received by each ear. Thus the reference of auditory systems is the head (craniotopic reference). The head must also be oriented appropriately to collect adequate information.

In artificial systems the visual captor is often an electronic device such as a high resolution camera. The main advantage of such a device is the high resolution (similar to the fovea) with a uniform array of distributed captors in all the field covered by the camera. However, processing, in real time, a quantity of information of the same order of magnitude as in humans, is, in the present state of technology, not feasible at reasonable cost. It is expected that, in some specific cases, a physiologically inspired solution could be a good way to deal with the problem. In order to do so, the following correspondence is considered:

o In the artificial system, a camera is provided with a selective attention region. This selective region is a small part of the entire sensitive region of the camera similar to the fovea, that can be electronically displaced. A displacement of the selective attenrion region corresponds to an eye movement.

o The camera is attached to a mobile platform with larger inertia. Moving the platform corresponds to a head movement.

o Two microphones are disposed on the platform symmetrically on each side of the camera.

The orientation of the visual captor is however depending not only on visual information but also of auditory information. This allows to increase the region of possible target detection but the complexity of information processing is increased.

As was said before, in the present case where there are information of visual and auditory kinds, we must consider that:

o Each sensor is associated with a system of coordinates. In fact, the position error of the visual information is coded in relarion to a coordinate system where the fovea is the origin (retinotopic referential), but the auditory captors (ear or microphone) is coded in relation to the head (craniotopic referential). To control the position of the visual captor it is necessary to make appropriate transformarion of coordinates, in our case, to retinotopic coordinates.

o Sensory information to be converted in a common frame of reference are of different nature. They must be fused in order to generate a dynamic control action to be applied to the plant. This second point is, in fact, performed automatically when a neural network is used since in both cases, information is represented by a particular neuron that is activated. Difference of precision can be represented by a suitable projection of information in the neuron network input layer. For this reason this second point is not emphasised in this work.

The model developed here is biologically plausible. The functional aspects are related to physiology and its constituent parts are derived from neuroanatomy. It is proposed this model could be used as a paradigm in implementing an artificial system.

2. THE MODEL

The image formed on the retina is transmitted to a particular brain structure: the Superior Colliculus (SC). This structure plays the role of interface between the sensory information and the commands to produce suitable motor actions. Information of several different nature, coded in their own system of coordinates arrive at the Colliculus. The upper part of the SC is essentially visual and the lower part auditive (see Figure 1).

In the intermediate part of the SC, the different kinds of information are integrated and transformed to form a unique premotor system. The deep part of the SC is directly involved in the control of movement. A model study of this deep structure was done by Lefèvre and Galiana (1992) [3,4]. In Figure 1, a version of this model is presented, modified in order to fuse different kinds of information and their different frames of reference.

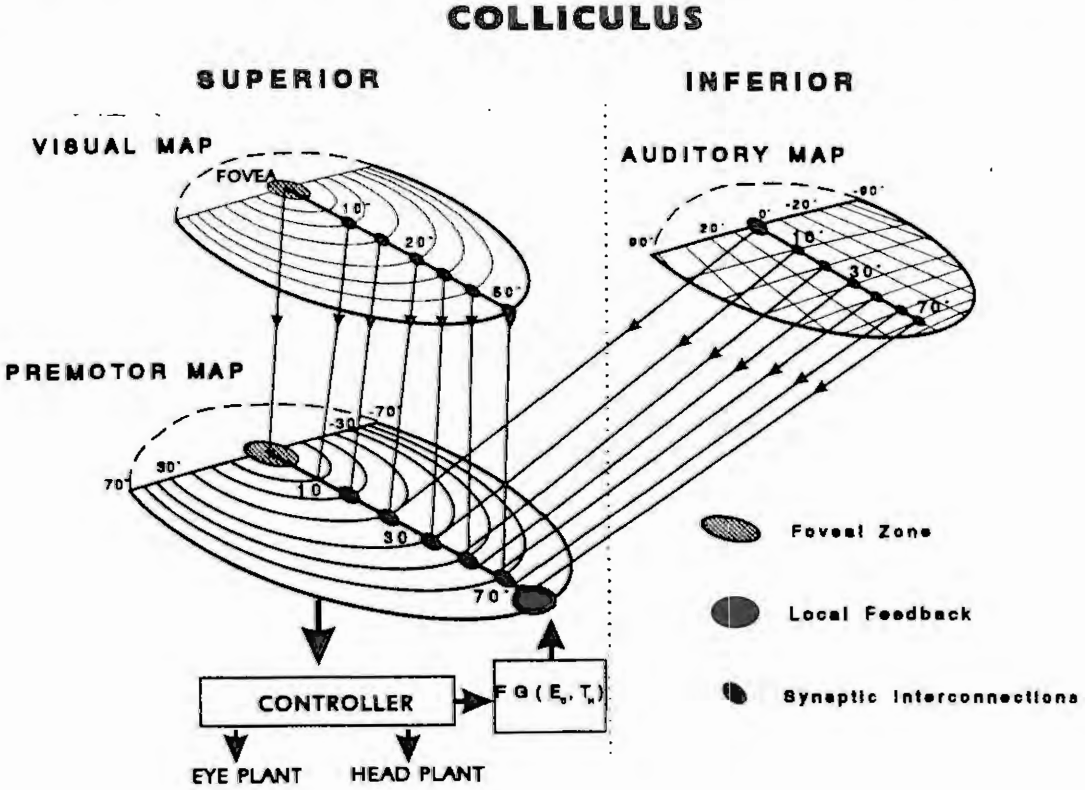


Figure 1. Global structure of the model

The present model performs the movement of the head and eyes. To be physiologically plausible, the model uses the association of a neural network and a non-linear controller, described in more detail by Lefèvre [3].

The neural network is composed of three different neuron layers in which different sorts of information are mapped. The model is composed of two sensory layers where visual and auditory information is mapped and one premotor layer where premotor information is created to be transmitted to the controller of the head and eye. The connectivity between the sensory layers and the premotor layer is illustrated in Figure 1. The output of each neuron in the visual sensory layer is used as input of only one neuron of the premotor layer, preserving position within the layer. The auditory layer, on the contrary, projects preferentially on neurons far from the fovea. A premotor neuron can receive more than one auditory input. Our model contains 50 neurons in each layer. The biological neuron (Figure 2) is modelled by a continuous time first order linear system with a time constant equal to 3ms. Each neuron of the sensory layers is connected to its two neighbours, with symmetrical synaptic weight G and to itself, by an internal feedback with a gain of K (Figure 2). This feedback is used to increase the resultant time constant while keeping the 3ms time constant of the neuron, which has a physiological meaning.

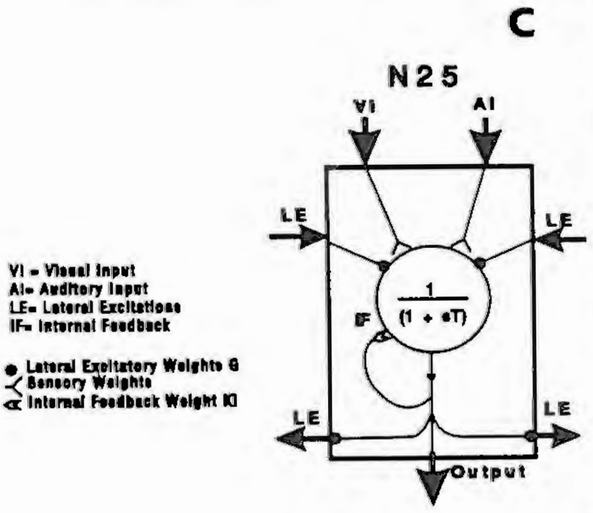


Figure 2. Neuron structure

The output weight of the neurons in the sensory layers is constant (unitary) and the output weight of the premotor layer depends on the position of the neuron in the layer according to the expression:

$$\lambda_i = (N_m - n_i)\,A \qquad (1)$$

where λ_i is the output weight of neuron in the position $n_{i,}$, N_m is the number of neurons in the premotor layer (in our case 50) and A (equal to 0.82) is an adaptation constant in relation to the initial model [3].

The controller follows the paradigm of model control [5]. The plant models used are continuous time linear systems, first order for the eyes, and second order for the head. The parameters of these models are chosen according to the available animal physiological data. The neural controller feeds the inputs to the head and eye plants and to the models of these plants. During the movement there is no visual feedback from the real objects, the models assure the necessary feedback. A visual feedback is indeed not possible due to the too long processing time within the retina. A velocity signal proportional to the gaze (movement of the head plus the eye) is applied at the neuron farthest from the fovea and allows the displacement of the maximum activity from this point to the fovea. This activation displacement is the basic mechanism of a dynamic coordinate conversion. The operator performing this conversion is represented by:

$$FG(T_H,E_O) \qquad (2)$$

and is shown in Figure 1 where FG is the feedback gain, E_O is the initial position of the eye in the orbit, and T_H is the initial value of craniotopic error. This value changes when a neuron of the auditive cluster is activated and it is constant when the input information is of visual nature. For an auditive source of information this operator has its value set at the beginning of the movement and is invariable during it. This value is supposed to depend on two main variables: the initial position of the eye in the orbit (E_O) and the initial value of craniotopic error (T_H) as expressed in Equation 2. The displacement velocity of the maximum activity on the layer varies monotonically with the feedback gain. Indeed, when the value of the feedback gain is higher, the value of the displacement velocity of the maximum activity is also higher. From the previous model study of Lefèvre [4] it is possible to conclude that the saccade amplitude is proportional to the activity duration on the layer. However, in this previous model, the displacement of activity starting from an initial activated neuron had always the same dynamic characteristic. As shown in Figure 3, in the present model, the dynamic characteristics of the wave of activity may be modulated in shape and in duration by Equation 2 for a same activated neuron. A consequence is that, for a same initial activated neuron, if the velocity feedback gain is increased, the saccade amplitude decreases, and if the gain decreases, the saccade amplitude increases. This mechanism and the topology of the network perform the transformation of coordinates.

3. SIMULATIONS

3.1. Propagation of the transversal wave

Figure 3 illustrates the transversal wave propagation in the SC layer and the mechanisms of coordinates transformation. The Figure is composed of 3 parts: I, II, III. Each part is divided in two graphs A and B.

For all the simulations presented in the Figure, an auditory target is presented at 30 degrees to the right. The activity created by the auditory stimulus in the auditory layer is transmitted to the neuron 17 in the premotor layer independently of the initial position of the eye in the orbit. The target is presented 100 ms after the onset of the simulation. Its duration is equal to 100ms and it creates an initial activity of 100 spike/s in the auditory layer.

In part I, II, and III the initial position of the eye in the orbit is respectively 0 degree, 17 degrees right and 10 degrees left. For these values of the initial position, and taking into account the craniotopic error, the values of FG are 3, 200, 0.02 and the results are shown in I, II and III.

The A graph of each part shows the activity on the premotor layer during the saccade. The 3 axes are: cell number, time and cell activity. The initial activated neuron is indicated by an arrow (17th cell), except in part III. There are 50 neurons in the layer and neuron 50 receives the projection of the fovea (indicated by an arrow in the graph). In all simulations, the feedback gain is distributed to the first 8 units. The time axis ranges from 0 to 300ms. The Z axis quantifies, in a relative scale, the neuronal activity in the layer. The B graph shows the saccades generated by the premotor activities seen in A. The axes are time and amplitude of saccade.

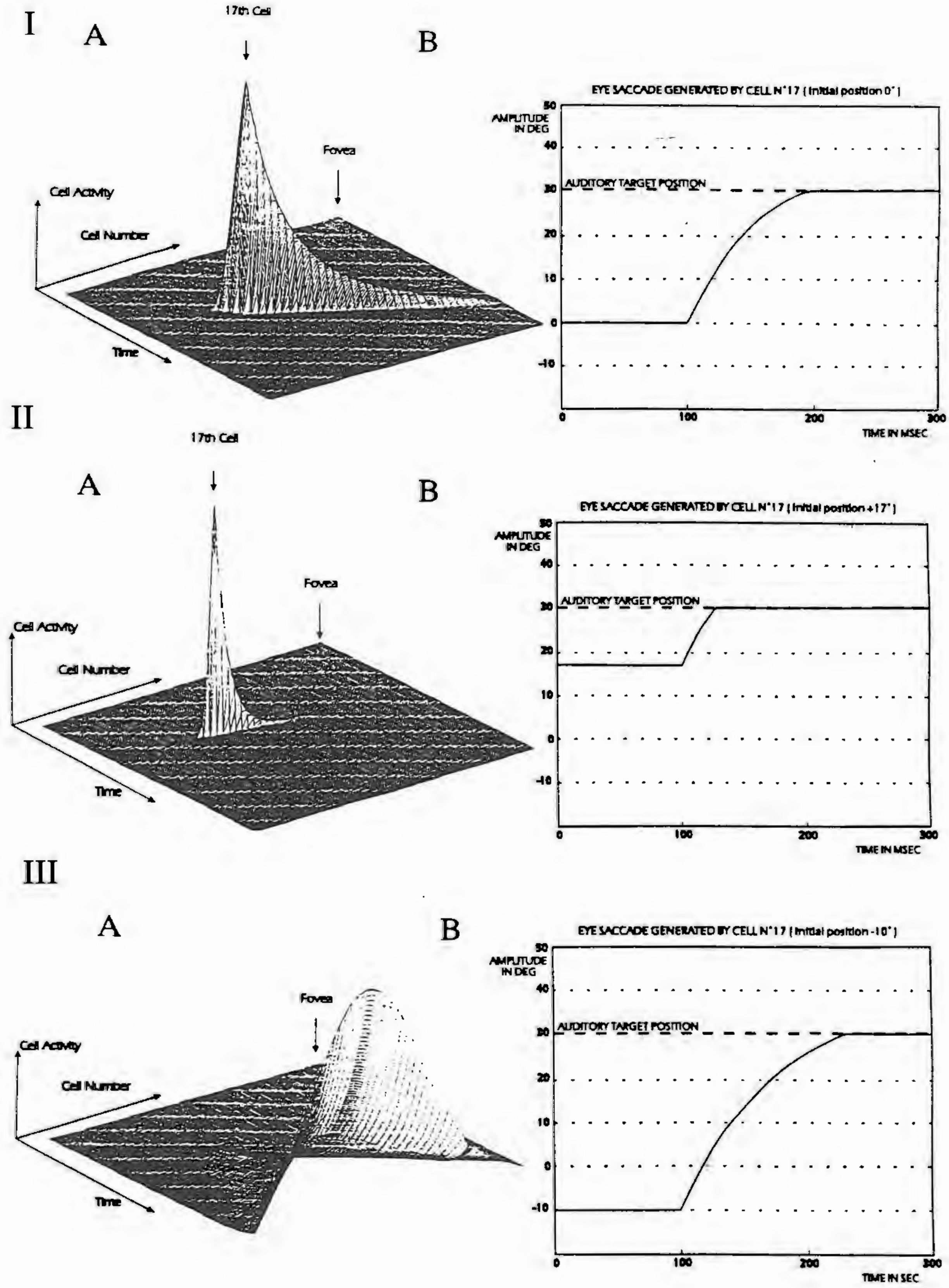


Figure 3: Simulation for three gain values (3, 200, 0.02)

In Figure 3.I, it can be seen that the value of the feedback gain is such that the activity spreads in all neurons from the initial neuron to the projection of the fovea. When the activity arrives at the fovea, the movement is stopped. The amplitude of the saccade equals 30 degrees. In Figure 3.II, the high value of the gain inhibits very quickly the activity present on the layer. Note the shape of wave of activity, very sharp, due to the inhibitory action of the feedback. The resulting saccade amplitude is 13 degrees. It is clear that an increase of the gain produces a decrease of the saccade amplitude (compare with Figure 3.I). By this mechanism the gaze accurately reaches the target. The last part (Figure 3.III) shows a simulation in which the value of the gain is small. In this case, the initial activity carried by neuron 17 is only slightly inhibited by the feedback. But as can be seen in graphic A the maximum of activation is always near the 17th neuron at the end of the movement. The saccade amplitude is equal to 40 degrees. As can be seen the initial starting position of eye in the orbit has been compensated by the decrease of gain. In order words, the craniotopic to retinotopic transformation has been suitably performed.

3.2. Movement towards an auditory target

Figure 4 presents the simulation results of a movement towards an auditory target at 40 degrees to the right. The auditory target activates a neuron in the sensory layer representing the craniotopic position, and as a result, the neuron 11 in the premotor layer is activated. As, in this example, the initial eye position is 20 degrees to the left, the feedback gain that results is very low (0.05 in this case). In this case the resulting saccade is not sufficient to bring the target to the fovea and the system must create a new corrective saccade. A new feedback value is set to attain the goal with a saccade having 20 degrees amplitude. The simulation shows that the movement was decomposed in two saccades. The precision of the movement depends on the estimation of the initial error and on the mechanisms adjusting the correct gain value. These two mechanisms could explain why a movement using an auditory target is generally performed with a precision lower than in the case of a visual target.

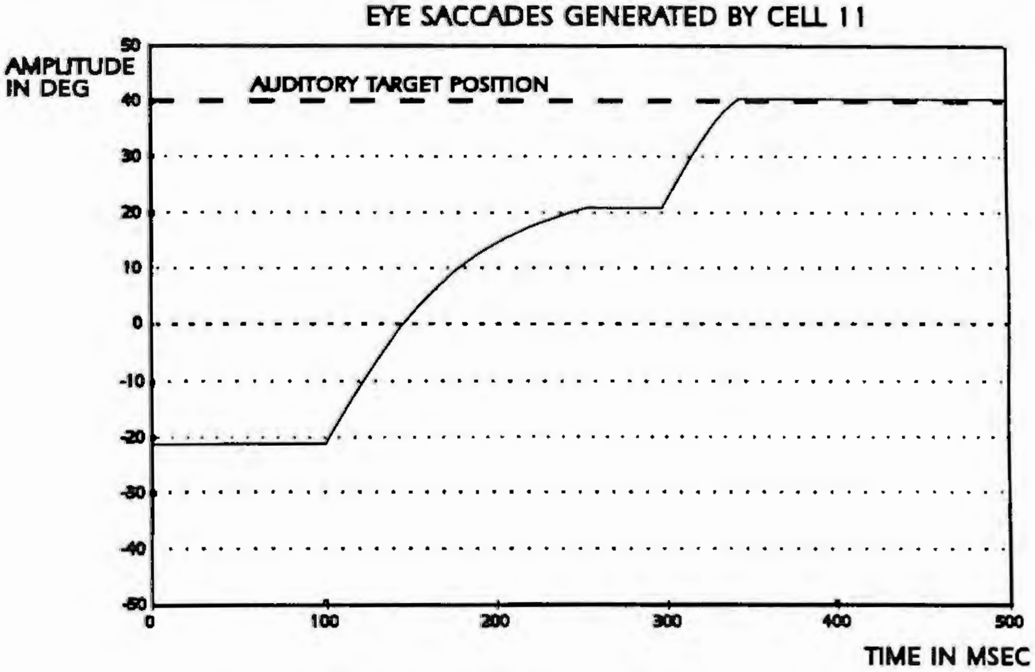


Figure 4. Simulation result of double saccade

3.3. Head Free

Figure 5 presents the simulation results of a saccade made with head free . A target is presented at 35 degrees to the right. For this angle an eye movement only is not sufficient to bring the target onto the fovea and a head movement is necessary. In this simulation, it can be seen that the eye reaches 28 degrees. At this position, as the head has already reached 7 degrees, the gaze (eye + head) is on the target. Then follows a phase when the head continues moving and the eye goes back in the orbit. This phenomenon hides, at least partially, the coordinate transformation mechanism, but the global functioning is perfectly suitable for the main goal of bringing the target image onto the fovea.

This mechanism has a counterpart in the artificial systems presented above. In fact, due to the electronic virtual shift of the fovea the 'eye' (selective attention region), can be oriented very fast. The 'head' (camera + platform) having a high inertia and being oriented mechanically is slower to move.

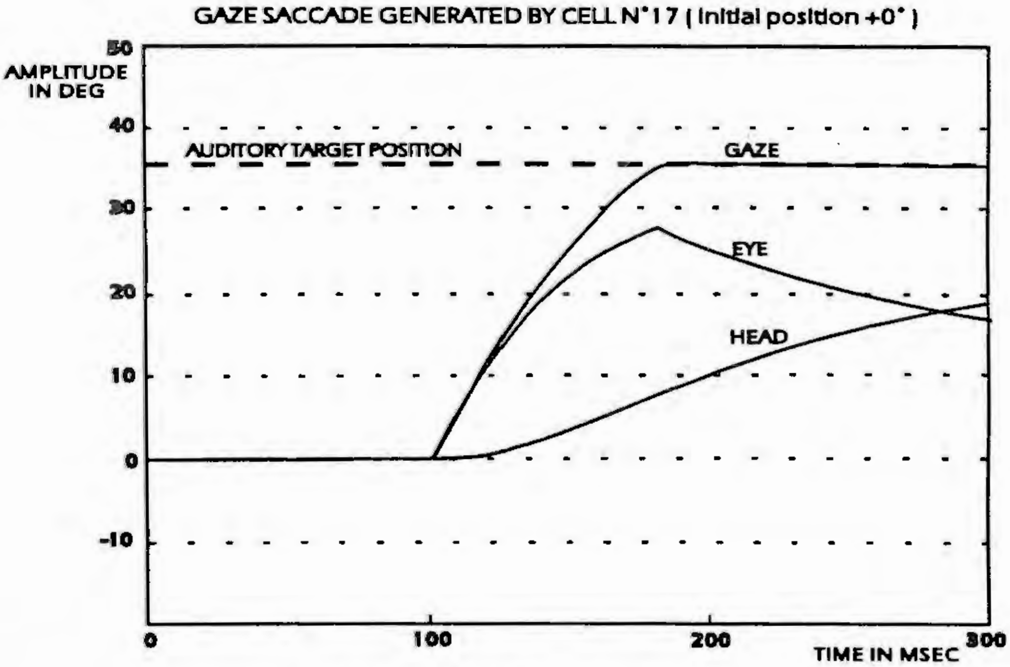


Figure 5. Head free simulation

4. CONCLUSIONS

This work presents a model study of the saccadic mechanism integrating visual and auditory information. The model uses the paradigm of neural networks and a biologically plausible neuron model to perform transformation of coordinates. It is proposed that the mechanisms of gaze control follow an internal model control paradigm as studied in [2,3]. This transformation of coordinates is based on the following facts:

- Existence of a projection from the sensory layers to the premotory layer of the model.
- The possibility of modulation of the feedback gain provided by an internal model that in the present model depends on initial eye position in the orbit and on in initial craniotopic error.

The model explains the occurence of multiple saccades towards an auditory target, performs the fusion of different kinds of sensory information.

5. ACKNOWLEDGEMENTS

We acknowledge the support of the Ministry of Technological Development of the Région Wallonne, Belgium, (FIRST Project n°2363). This work was also supported by European Community ESPRIT project MUCOM.

6. REFERENCES

[1]-Bottemanne, I., Marechal, C., Barreto, J., Roucoux, A., A Neural Network Model of the Superficial Collicular Layers, European Journal of Neuroscience supplement 5 (1992),.280 (Abstract)

[2]-Bottemanne, I., Barreto, J., Lefevre, P., Roucoux, A., Collicular mechanism of multisensory control of gaze: a model, European Journal of Neuroscience supplement 6 (1993), 274 (Abstract).

[3]-Lefèvre, P., Galiana, H.L., Dynamic feedback to the superior colliculus in a neural network model of the gaze control system. Neural Networks, n°5, 871-890, 1992.

[4]-Lefèvre, P., Experimental Study and Modeling of Eye Head Orientation. Doctoral Thesis, Faculty of Applied Sciences, Catholic University of Louvain, 1992.

[5]-Morari, M. , Zafiriou, E., Robust Process Control, Prentice-Hall, 1989.

A NEUROFUZZY CONTROLLER SYNTHESIS USING CMAC APPROACH

George Calcev

Control Dept. of University Polytechnica of Bucharest
Splaiul Independentei 313, Bucharest, Romania

ABSTRACT

This paper presents a self-tuning fuzzy controller using the similarities from a fuzzy controller and a CMAC archi-tecture. A modified backpropagation learning law based on propagation of output error through the plant is used. His properties is studied in software environment.

1. INTRODUCTION

The recent great interest in fuzzy logic has been , in part, due to the extent of application area in which fuzzy control has been used. The complexity of behavior of many practical systems and unavailability of an analytical model motivates technics like fuzzy control and neural control.

The main features of the fuzzy control in complex dynamical systems are: the transparency and local representation of operator knowledge, use of qualitative reasoning imitating the human operator , the possibility of control without an analytical model of the plant, robustness against noise and parameters variance of the plant, the generalization and interpolations properties

The limitation of fuzzy control (FC) are : rigid and generally unadaptive, unsolved problem of a general tuning strategy .

On the other hand the neural network controllers are characterized by : adaptation, learning and self organization, generalization and interpolation, robustness and plasticity , fault and noise tolerance , and the limitation : difficulty representing structured knowledge, inability to handle diverse types of knowledge, black box solution (difficult to explain results), may require excessive training times.

With this observations, is naturally the attempting to combine this two technics in order to obtain a better performance. In this sense we propose to add at the fuzzy rules a weight layer, which make possible on line tuning of controller using output error and signs of Jacobian of the plant.

2. FUZZY CONTROLLER

A fuzzy controller (FC) typically take the form of a set of IF-THEN rules whose antecedents and consequent are fuzzy values with membership functions.

A FC infers a value for its outputs variables from fuzzy values of input variables and from the rules base . The fuzzy rules are expressed as IF error is Ai AND change error is Bj THEN output is Cij where Ai, Bi, Cij are fuzzy numbers described by their membership functions (usually triangular or trapezoidal) . Given an actual situation a subset of rules are triggered and then is calculated the nonfuzzy (crisp) command, by a combination of consequence of this triggered rules ..Usually, as inputs the scaled error (E) and scaled change error (CE) are used, thought relationship such as:

$E= KE * \text{error}(t) \qquad CE= KC * \text{change\_error}(t)$

where error =set point - output and change error = $1/T(\text{error}(t) - \text{error}(t-1))$

KE , KC are scaling factors required by the ranges of the FC inputs , allowing the introduction of proportional and derivative controller modes and T is time sampling interval. The shape of membership functions and the rules are issued from operator experience and are affected by some degree of uncertainties. The final value (crisp value) is obtained by a "center of gravity" method .

The individual contributions are computed by MAX_MIN composition, where t-norm MIN it is used to evaluate the certainty factor of each rule and co-t-norm MAX for union contribution of the rules with the same consequence.

3. SIMILARITIES BETWEEN A FUZZY CONTROLLER AND CMAC

A neural controller performs a specific form of adaptive control, with the controller taking the form of a nonlinear multilayer network and the adaptable parameters being the strengths of the inter-connections between neurons. A special architecture of neural network used in control is CMAC. Cerebellum Model Articulation Controller (CMAC) is an artificial neural network proposed by James Albus [1]. The main advantage of CMAC against other networks is that CMAC is capable of learning nonlinear mapping functions extremely quickly since only a small subset of weights are active at each point in the input space. This local approximation technique also has the ability to train the network in one part of the input space without corrupting what was already been learned in more distant region. His high rate of learning recommends him to real time applications.

A CMAC neural network is a perceptron like associative memory with overlapping receptive fields, and uses two maps: S: X ----> A and P: A ----> Y where X is the input space, A association space and Y output space.

The first, maps the input state space in the "state space detectors " which are AND- gates with several binary inputs and a binary output. This map explains input generalization. The overlaying sensors are arranged in such way that each input variable excites exactly C input sensors, and C is called generalization factor [2].

The second map connects the AND-gates to the output via adjustable weights Wi (Wij for multivariable case) $Y = \sum_i w_i \times S_i(x)$

In order to learn a nonlinear behavior, the weights are modified with a Bernard-Widrow delta rule ,

$$w_i = w_i + \beta \times \frac{(y_d - y) \times S_i(x)}{\sum_j S_j^2(x)}$$

where yd and beta are desired output and training gain.

Recently [3] was proposed for the receptive functions B-splines functions which offers both function and function derivatives learning capability.

The main idea of this paper is that the fuzzy controller is the generalization of a CMAC controller in the following sense:

-The receptive fields of CMAC, B-splines function can be considered the membership functions for a fuzzy controller with constraint that only two receptive fields can be overlaid. C=2 generalization factor.

-The MIN gate from fuzzy controller are generalization of AND gate from CMAC

-The first map from CMAC can be replaced by fuzzy rules from FC

With this remarks it is naturally that a weight layer can be associate with each of the rules. These weights will be adjusted with a learning rule like in CMAC.

The output of this fuzzy weighted controller will be

$$out = \frac{\sum_i \mu_i \times w_i \times u_i}{\sum_j \mu_j}$$

4. HOW IT WORKS

The main advantage of a such structure hybrid in some sense is that the controller has apriori knowledge about how to compute the command (fuzzy numbers, fuzzy rules) embedded in the fuzzy part. The weight

layer is neural weight like and can be adapted for fine tuning of the controller for a specific plant.
The change of weight is a learning process CMAC like, which use delta rule used. In on-line tuning we use the method of specialized learning network training [5].
The principle for on line adjusting of weights is to minimize the square of difference between output and desired output. $Eu = (ud(t)-u(t))^2$

The learning law is gradient type $\Delta w = -\eta \times \frac{\partial Eu}{\partial w}$ Where $\eta > 0$ and ud(t) is optimal command desired.
Unfortunately the desired command is not known on-line and even off line without a very precise analytical model of the plant is difficult to compute .
In [4] authors propose to update the weight of neural network by using the error between set point and measure.

$$Ey = (yd(t)-y(t))^2$$

The convergence condition is that

$$\text{sign}(ud(t) -u(t)) = \text{sign}(yd(t) -y(t))* \frac{\partial y}{\partial u}$$

Where signs from Jacobian are already known (they are embedded in the fuzzy rules) and only their signs are important, or it can be used an evaluation of $\frac{\partial y}{\partial u} = \frac{\Delta y}{\Delta u}$

5. EXPERIMENTAL RESULTS

We have used a FC with seven fuzzy numbers for error and change error and the look-up table of fuzzy rules. The reference was rectangular pulses. The model of the plant used was a nonlinear equation suggested in [6] and defined by

With PD like command we have a good convergence but a steady state error. With PI form the tuning is poor, the system become unstable. The best result was obtained when we have used PD PI form, with no stationary error and quick tuning of controller with an epoch learning approach. The learning expression for weights was:

The behavior of this controller was tested in PD form : u(t)=Ku*out and PI form Δu(t)=Ku*out
The stop condition is the rate of sum square of errors.

6. CONCLUSIONS

The paper propose and present the experiments of a new self tuning fuzzy controller with neural learning technique. This approach provides an elegant way for multivariable controller tuning, with a good adaptive capacity. The use of error for training and control with few information about process made him easily to implement for a large class of plants. In present we analyze the possibility to improve the convergence of parameters, and we test him for different multivariable nonlinear process (robots, bioprocesses).

References

[1] Albus James "Brains, Behavior, and Robotics ", pp.143-186, 1981 Byte Publications Inc.

[2] Burgin George "Using Cerebellar Arithmetic Computers", AI Expert June 1992,pp 32-41

[3] Lane S., Handelman D, Gelfand J. " Theory and Development of
Higher-Order CMAC Neural Networks", IEEE Control Magazine, april 1992, pp 23-30

[4] Khang Shin ,Xianzhong Cui, " Design of an Industrial Process Controller Using Neural Networks" ACC 1992 pp.508-512

[5] Psaltis D, Sideris , Yamamura A."A Multilayered Neural Controller ", IEEE Control Systems, April

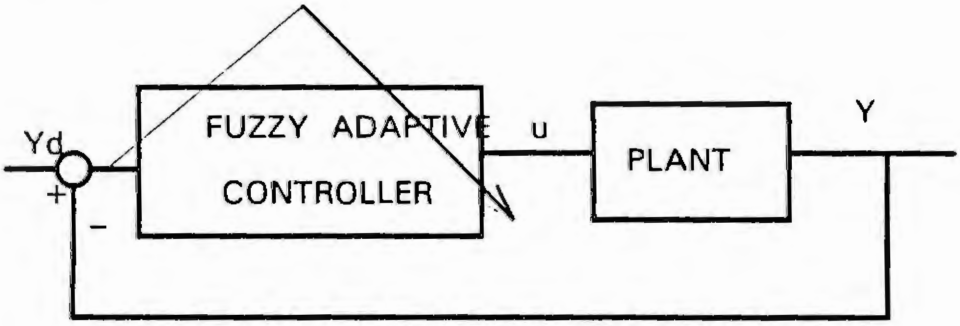
1988, pp.17-20

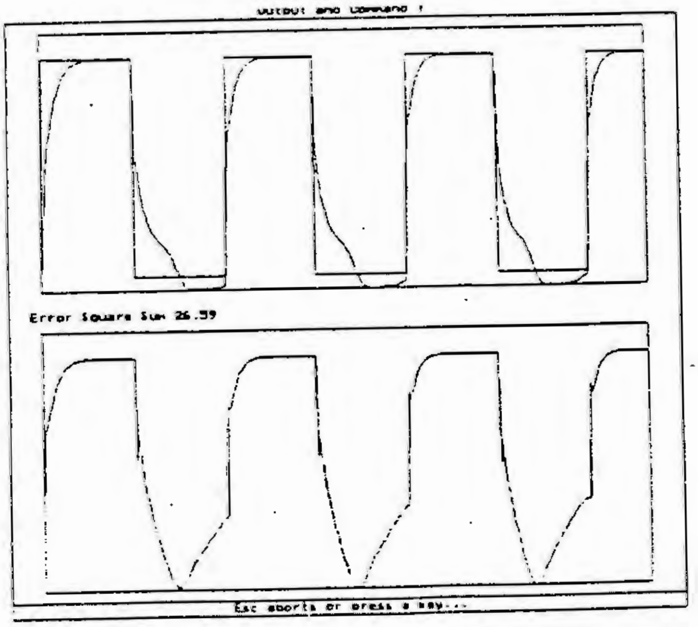
[6] Kumpati S Narendra and Kannan Parthasarathy, "Identification and control of dynamical systems using neural networks", IEEE Tr on Neural Networks , March ,1990,pp4-27

RULES LOOK-UP TABLE

ER CER	NL	NM	NS	ZE	PS	PM	PL
NL	NL	NL	NL	NL	NM	NS	ZE
NM	NL	NL	NL	NM	NS	ZE	PS
NS	NL	NL	NM	NS	ZE	PS	PM
ZE	NL	NM	NS	ZE	PS	PM	PL
PS	NM	NS	ZE	PS	PM	PL	PL
PM	NS	ZE	PS	PM	PL	PL	PL
PL	ZE	PS	PM	PL	PL	PL	PL

NL/PL negative/positive large
NM/PM negative/positive medium
NS/PS negative/positive small
ZE zero ER error
CER change of the error




Output and Command !

Error Square Sum 26.59

Esc aborts or press a key...

QUALITATIVE MODELS FOR SIMULATION OF CONTROL SYSTEMS

R. GOREZ and M. DE NEYER

Université Catholique de Louvain
Centre for Systems Engineering and Applied Mechanics
Bât. EULER, Avenue Georges Lemaître, 4
B-1348 Louvain-la-Neuve, Belgium *
Tel: +32-10-47 23 76 Fax: +32-10-47 21 80 e-mail: gorez@auto.ucl.ac.be

Abstract

A procedure for building qualitative models of control systems components is described. Using a multi-valued universe of discourse, introducing fuzzyness and holding it when processing the variables throughout the model allow accurate fuzzy qualitative simulation. Simulation results for first-order and second-order systems with real or complex poles are presented and the influence on the modelling accuracy of several factors such as the quantity space and the membership functions is investigated.

1. INTRODUCTION

Starting from the pioneering work of Forbus and Kuipers on naive physics and commonsense reasoning [10,12] qualitative modelling and simulation have gained an increasing interest among the control community over the past ten years (see e.g. [1,2,4,5,6,13,15,18]). There are several approaches to qualitative models, depending on the purpose of the model [3] and behavioural properties such as precision, accuracy and uncertainty [14]; one of them is based on the use of fuzzy qualitative transposition of numerical discrete-time models [15,18]. An attractive feature of this approach is that it can be used in model-based control systems [11] as well as for relatively accurate simulation of dynamical systems[16]. The accuracy of such a qualitative simulation, in other words the closeness of the model behaviour to that of the modelled system, depends on several factors, in particular on the number of distinctions supported by the description of the behaviour [17] and on the functions and operators used in fuzzyfication and defuzzyfication processes. The influence of such factors will be shown in the next section considering mainly fuzzy qualitative modelling and simulation of a simple first-order system. This choice is dictated by obvious reasons of simplicity but also by the fact that first-order systems are fundamental components of most control systems and that the analysis of qualitative simulation results of such simple components can give valuable qualitative hints on the ability of using qualitative simulation for more complex systems. Section 3 presents some simulation results for first- and second-order systems. Comments and conclusions are given in the last section.

2. QUALITATIVE MODELLING AND SIMULATION OF A LINEAR SYSTEM

2.1. Quantitative model of a first-order system

A continuous-time single-input-single-output first-order linear system can be described by the following differential equation:

$$\frac{dy}{dt} = a_c \cdot y + b_c \cdot u, \tag{1}$$

where u and y are respectively the input and output variables of the system, t is time, and a_c and b_c are parameters: $a_c = 0$ and $b_c = 1$ for au integrating system, $-a_c = b_c = 1/T > 0$ for a self-regulatory system with time-constant T and unit gain.

*This paper presents research results of the Belgian Program on InterUniversity Poles of Attraction initiated by the Belgian State, Prime Minister's Office, Science Policy Programming. The scientific responsibility rests with its authors

Taking the sampling interval as time-unit the discrete-time numerical model of such a system is:

$$y(k+1) = a \cdot y(k) + b \cdot u(k), \tag{2}$$

where k refers to the sampling time and a and b are parameters: $a = b = 1$ for an accumulator (corresponding to an integrator), $a = exp(1/T) \in (0,1)$ and $b = 1 - a$ for a self-regulatory system with time-constant T and unit gain.

2.2. Building a qualitative model of a first-order system

The obvious conclusion of the previous section is that *the next output of a first-order linear system is a weighted sum of the current input and output values.* Qualitative modelling using sign algebra which can only distinguish between crisp zero, positive and negative values is unable to take into account the relative values of the weighting factors a and b. Besides, even in the simplest case ($a = b = 1$), the accuracy of such a qualitative simulation will be very poor. Then it has been proposed to extend the *quantity space*, in other words to increase the number of qualitative values or distinctions involved in the discretisation of the range of variables, and/or to replace the qualitative discretisation which is actually based on crisp membership functions by a fuzzy discretisation using smooth membership functions [5,11,17]. Extending the quantity space allows the substitution of a more complete decision table for the naive qualitative addition rule (Table 1). Moreover replacing the crisp discretisation by a fuzzy discretisation allows some kind of interpolation between the values given in the decision table. Obviously, attaching membership grades to qualitative values gives more information on the true value of any variable; however this information should be kept and conveyed throughout all the simulation process, even for simple systems such as the accumulator described by Eq.(2) with $a = b = 1$. Such a system indeed involves a regenerative loop feeding the output back to one of the inputs (Fig. 1). In pure qualitative simulation this loop may lead to fast saturation of the system output, in fuzzy qualitative simulation it may result in an *explosion of fuzzyness* as pointed in [7,8]. In fuzzy qualitative simulation the easiest way of avoiding loss of information is to introduce artificial defuzzyfication and fuzzyfication interfaces in the previous loop as shown on Fig. 1, and more generally in any transfer of variable from a node of the model to another one [5,7,8,11].

		$[v]$		
		−	0	+
$[u]$	−	−	−	?
	0	−	0	+
	+	?	+	+

					$[v]$			
		NB	NM	NS	ZE	PS	PM	PB
$[u]$	NB	NB	NB	NB	NB	NM	NS	ZE
	NM	NB	NB	NB	NM	NS	ZE	PS
	NS	NB	NB	NM	NS	ZE	PS	PM
	ZE	NB	NM	NS	ZE	PS	PM	PB
	PS	NM	NS	ZE	PS	PM	PB	PB
	PM	NS	ZE	PS	PM	PB	PB	PB
	PB	ZE	PS	PM	PB	PB	PB	PB

Table 1: Addition rules in sign algebra and in a 7-valued qualitative universe of discourse

Decision Table 1 is strictly valid only in the case $a = b = 1$. In other cases it is possible to change some values in the decision table as it was done in [11]. However this allows only crude approximations of the system response for time-constants of the order of two or three times the sampling interval. A more accurate representation is obtained via the obvious trick of tuning the *interface parameters*, namely the scaling factors used in the conversion of the real world onto the qualitative universe of discourse and vice versa, as it is also done in qualitative control [9]. These interface parameters must be selected in such a way that the range of each input variable is converted in the normalized interval $[-1,+1]$ taking into account the input gains of the system to be modelled. For instance the right-hand side of the model (2) is viewed as the true addition of $[a \cdot y(k)]$ and $[b \cdot u(k)]$, both variables being normalized in such a way that their normal values lie within the interval $[-1,+1]$.

Then the choices that the model builder has still to do are concerned with the *definition of the quantity space* (*number* of qualitative values, *distribution* of these values on the range of the normalized variable: the mapping of the real world into the universe of discourse may be linear or not) and the

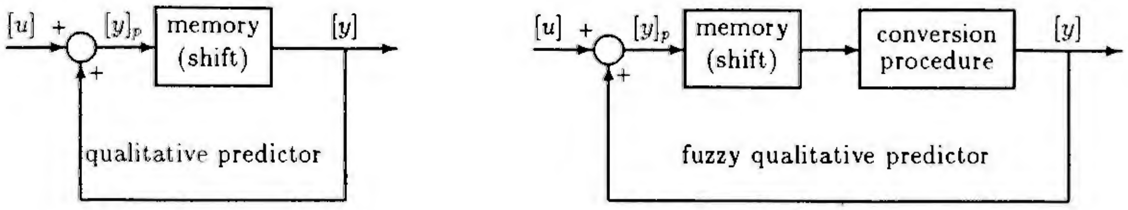


Figure 1: Block-diagrammes of qualitative and fuzzy qualitative predictors.

introduction of fuzzyness in the qualitative model. In the followings n will denote the number of nonzero values on each side of the zero value; for examples, in sign algebra $n = 1$, for the 7-valued discretisation in Table 1 $n = 3$. If the discretisation is regular, in other words if the qualitative values are equally distributed over the interval $[-1, +1]$, the real value attached to the i^{th} positive value is then i/n. The case where the distribution of qualitative values is not regular can also be considered by assigning arbitrary q_i's to the qualitative values.

Introducing fuzzyness in the qualitative model implies the selection of *membership functions*, the choice of a *rule for composition* of the two added variables and the definition of a *defuzzyfication procedure*. In this study one has checked the followings:

membership functions : triangular and bell-shaped functions have been used for regular distributions, triangular membership functions have also been tried for nonregular distributions;

composition rule : *product-* and *minimum-* rules have been used; it means that, for given real inputs u and v, if $\mu_r(u)$ and $\mu_k(v)$ are the membership grades attached to the qualitatives values corresponding respectively to the r^{th} row entry and the k^{th} column entry of the decision table, the weight assigned to the output value $q_{r,k}$ corresponding to the crossing of the r^{th} row and the k^{th} column is $\mu_{r,k} = \mu_r(u) \cdot \mu_k(v)$ for the *product*-rule or $\mu_{r,k} = min\{\mu_r(u), \mu_k(v)\}$ for the *minimum*-rule;

defuzzyfication procedure : the real value assigned to the model output is given by

$$y = \frac{\sum_{r,k} \mu_{r,k} \cdot q_{r,k}}{\sum_{r,k} \mu_{r,k}}.$$

2.3. Qualitative models of high order systems

The basic qualitative block presented in the previous section can be used in serial and/or parallel combinations for building more complex systems. For example a second-order system with real poles can be represented by two cascaded first-order blocks. This is no longer possible for a system with complex poles such as:

$$y(k+2) = a_1 \cdot y(k+1) + a_0 \cdot y(k) + b \cdot u(k), \tag{3}$$

with $a_1^2 + 4a_2 < 0$. However for any values of the parameters a qualitative model of the system described by Equation (3) can be set up by adding all the variables in the right-hand side of (3). As qualitative addition has been defined for two variables only we have to use the associative property of addition:

$$[u] + [v] + [w] = ([u] + [v]) + [w], \tag{4}$$

hence using as many 2-input adders as there are "+" operators in the right-hand side of the original quantitative model. Obviously artificial fuzzyfication-defuzzyfication interfaces must be included in the connections between the various adders. Both approaches will be illustrated by simulation results.

3. SIMULATION RESULTS

Fig. 2 shows the step and free responses of a first-order system and several qualitative models using 3- and 7-valued universes of discourse ($n = 1$) and ($n = 3$), symmetrical triangular or bell-shaped membership functions and *product*- or *minimum*-composition rule. Three magnitudes of the step input or of the initial output value have been checked: 1/3, 2/3 and full range. It is clear that the accuracy of the simulation with sign algebra is very poor except in the neighbourhood of zero. On the contrary with a 7-valued universe of discourse the use of triangular membership functions with the *product*-composition rule or of bell-shaped membership functions with the *minimum*-composition rule gives very accurate simulation results except for values close to 1 that is to say near the boundary of the range of variables. Same conclusions hold for a 5-valued universe of discourse and for various values of the response time of the system. On the other hand use of asymmetrical membership functions has led to very poor simulation.

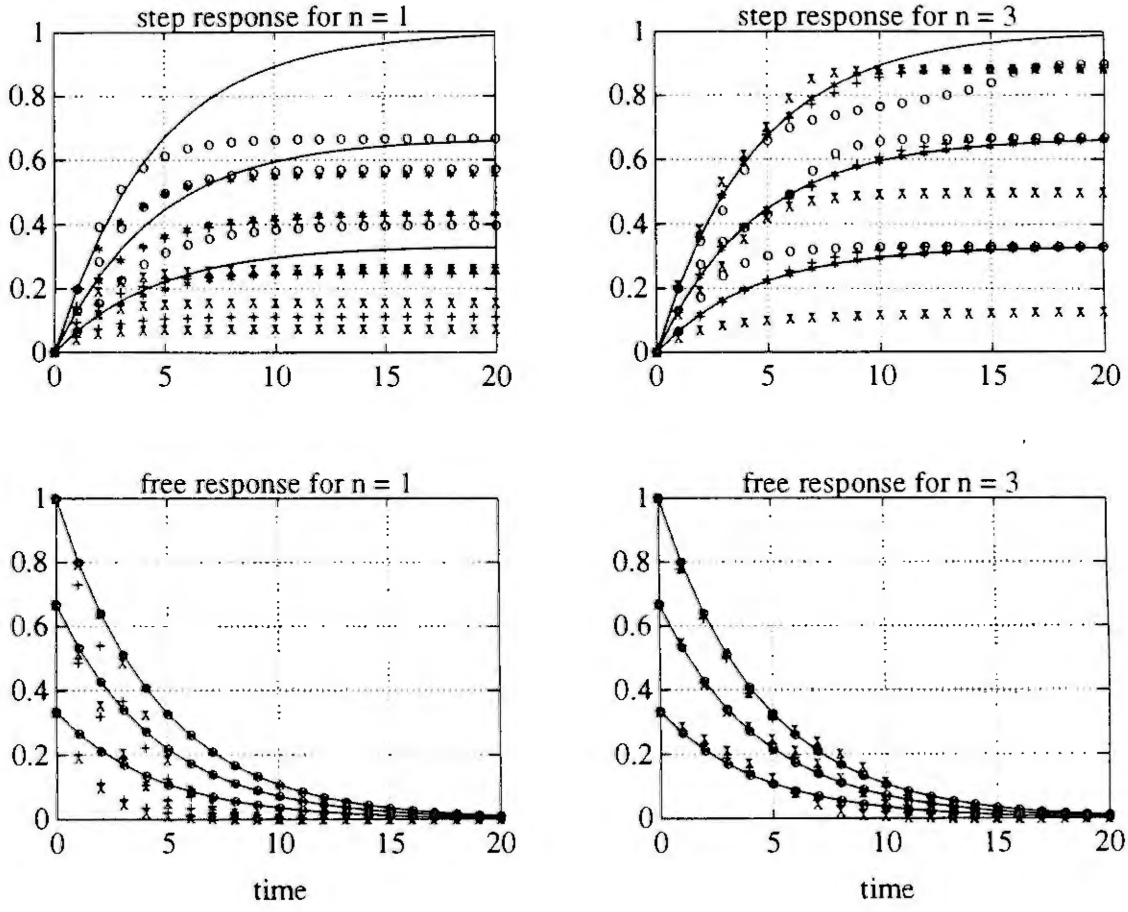


Figure 2: Responses of qualitative models using 3- or 7-valued universe of discourse

— actual system

* triangular membership functions and *product*-composition rule

o triangular membership functions and *minimum*-composition rule

x bell-shaped membership functions and *product*-composition rule

+ bell-shaped membership functions and *minimum*-composition rule

Fig. 3 shows the step response of the qualitative models of two second-order systems using symmetrical triangular membership functions and a 7-valued universe of discourse. The first system is critically damped and is represented by a cascade of two fist-order blocks. The other one is underdamped and is modelled according to Equation (3). Again it can be seen that the accuracy of qualitative simulation is very good over more than 2/3 of the range of variables.

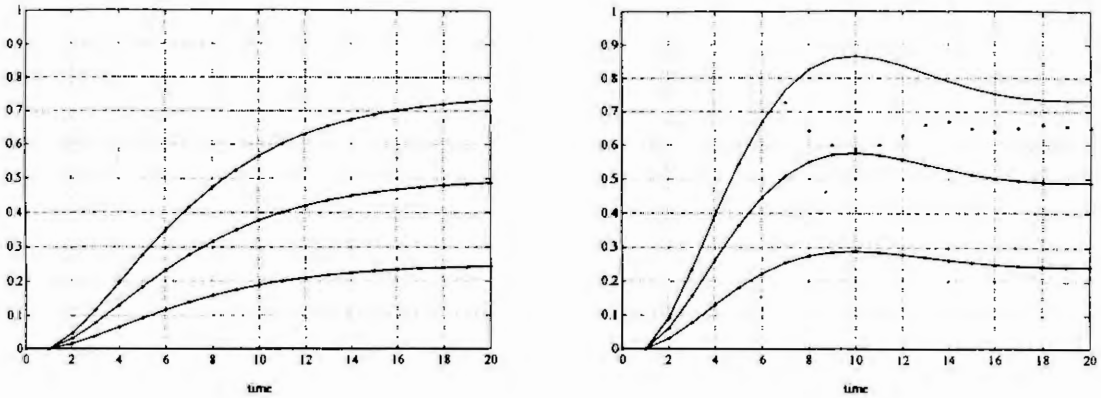


Figure 3: Step response of qualitative models of two second-order systems

4. CONCLUSIONS

A procedure for building qualitative models in view of qualitative simulation of control systems components has been described. The use of a 5- or better a 7-valued universe of discourse and the introduction of fuzzyness allow accurate fuzzy qualitative simulation of systems as it is shown by simulation results for first-order and second-order systems with real or complex poles. The qualitative model of a first-order subsystem is then a simple adder the parameters of the true system being included in the scaling factors of fuzzyfication interfaces. Obviously introducing fuzzyness gives more information on the true values of the processed variables. However this information could be lost or scattered during the transfer of variables throughout the model; this is avoided by means of artificial defuzzyfication-fuzzyfication interfaces which allow the assignment of membership grades to qualitatives values. Concerning the fuzzyfication procedure it appears that the use of qualitative values regularly distributed over the range of the processed variables with symmetrical triangular membership functions and of the product-composition rule gives the best accuracy. Still using bell-shaped membership functions and the minimum-composition rule results also in a reasonable accuracy. Obviously such defuzzyfication-fuzzyfication interfaces allow hybrid simulation mixing qualitative and quantitative models of the various components of the system to be simulated.

Up to now only linear systems have been considered in this study. However this simulation technique could be applied to additive nonlinear systems, either through nonlinear scaling at the input of the fuzzyfication interfaces or through an appropriate irregular distribution of the qualitative values over the range of the processed variables.

REFERENCES

[1] J. Barreto, M. De Neyer, Ph. Lefèvre, and R. Gorez. Qualitative physics versus fuzzy sets theory in modelling and control. In *IECON'91: IEEE/SICE International Conference on Industrial Electronics, Control and Instrumentation*, volume 2, pages 1651–1656, Kobe, Japan, 1991.

[2] K. Bousson, L. Travé-Massuyès, and J. Aguilar-Martin. Causal qualitative representation of dynamical systems. In *ECC'93: European Control Conference*, volume 3, pages 1769–1773, Groningen, The Netherlands, 1993.

[3] B. Bredeweg and C. Shut. Building qualitative models: an introduction. In *ECC'93: European Control Conference*, volume 3, pages 1751–1756, Groningen, The Netherlands, 1993.

[4] F.E. Cellier. *Continuous system modeling*. Springer Verlag, New York, 1991.

[5] F.E. Cellier, A. Nebot, F. Mugica, and A. De Albornoz. Combined qualitative/quantitative simulation models of continuous-time processes using fuzzy inductive reasoning techniques. In *IFAC Symp. on Intelligent Control and Instruments for Control Applications*, pages 589–593, Malaga, Spain, 1992.

[6] W.F. Clocksin and A.J. Morgan. Qualitative control. In *European Conference on Artificial Intelligence*, pages 350–356, 1986.

[7] M. De Neyer, R. Gorez, and J. Barreto. Disturbance rejection based on fuzzy models. In M.G. Sing and L. Travé-Massuyès, editors, *Qualitative Reasoning and Decision Support Systems*, pages 215–220. North-Holland, Amsterdam, The Netherlands, 1991.

[8] M. De Neyer and R. Gorez. Integral action in fuzzy control. In *EUFIT'93:First European Congress on Fuzzy and Intelligent Technologies*, volume 1, pages 156–162, Aachen, Germany, 1993.

[9] T. Fitzpatrick. On qualitative control. In *ECC'93: European Control Conference*, volume 3, pages 1765–1768, Groningen, The Netherlands, 1993.

[10] K.D. Forbus. Qualitative process theory. *Artificial Intelligence*, 24:85–168, 1984.

[11] R. Gorez, M. De Neyer, D. Galardini, and J. Barreto. Model-based control systems: Fuzzy and qualitative realizations. In *IFAC Symp. on Intelligent Components and Instruments for Control Applications*, pages 681–686, Malaga, 1992.

[12] B.J. Kuipers. Commonsense reasoning about causality: deriving behaviour from structure. *Artificial Intelligence*, 24:169–230, 1984.

[13] B.J. Kuipers. Qualitative simulation. *Artificial Intelligence*, 29:289–388, 1986.

[14] R.R. Leitch. Model selection and composition for on-line control. In *ECC'93: European Control Conference*, volume 3, pages 1757–1760, Groningen, The Netherlands, 1993.

[15] J. Lunze. Qualitative modelling and qualitative control of continuous-variable systems. In *ECC'93: European Control Conference*, volume 3, pages 1761–1764, Groningen, The Netherlands, 1993.

[16] Q. Shen and R.R. Leitch. Fuzzy qualitative simulation. *IEEE Transactions on Systems, Man and Cybernetics*, 1993.

[17] Q. Shen and R.R. Leitch. On extending the quantity space in qualitative reasoning. *The International Journal for Artificial Intelligence in Engineering*, 7(3):167–173, 1993.

[18] M. Sugeno and T. Yasukawa. A fuzzy-logic-based approach to qualitative modeling. *IEEE Transactions on Fuzzy Systems*, 1(1):7–31, 1993.

MODELLING OF NEURAL NETWORKS IN VHDL

P.S.SZCZEPANIAK, and D.KACA
Technical University of Łódź
Institute of Computer Science
ul.Sterlinga 16/18, 90-217 Łódź, Poland
Tel.+(42)329757, Fax+(42)368522

Abstract. The paper deals with the implementation of artificial neural networks (ANN) in VHDL - a programming language derived primarily for hardware design and description . The features of VHDL in the context of its use for modelling and simulation of ANN are briefly presented. The modelling approach is explained on some types of neural processing elements.

1. INTRODUCTION

The VHSIC Hardware Description Language (VHDL) is a programming language designed to describe Very High Speed Integrated Circuits [1,2]. As a standard of IEEE it finds a wide interest among the designers of electronic circuits. However it can be successfully used also for other purposes, see e.g.[3,4], particularly if implemented processes run parallelly and processed data are interpretable in terms of directed signals of real, logical or other values.

The purpose of this paper is to present how VHDL can be applied to modelling and simulation of artificial neural networks (ANN). The use of the concepts of this language in the modelling of ANN sometimes makes it necessary to rethink and modify the classic models of neurons and nets.

2. FEATURES OF VHDL

The main features of VHDL particularly useful in the modelling of ANN are:

- the concept of signal and process,
- activation of the simulation process by changes of the values of signals, to which the process is sensitive ("sensitivity list"),
- modular and hierarchical structure of the program,
- quasi-parallel running of computations.

A set of operations, which form a logical whole (i.e. the operations are executed sequentially and enclosed between **process** - **end process** VHDL keywords) is referred to as a "process" and the pathways along which communication between processes takes place over - as "signals". If the complexity of a particular system imposes so, a proper behaviour can be a collection of independent processes running parallely (executed concurrently).

The simulation cycle is two-staged. In the first stage, all signals scheduled for updating, obtain new values. In the second stage (only) the activated processes are executed. At the completion of the simulation cycle, the simulation clock is set to the next simulation time at which a transaction is to occur and the cycle is started again. For the description of the event driven simulation one needs the concept of the signal in VHDL and the definition of an event.

A signal in VHDL is an item which:
- has a type associated with it,
- defines a directed data pathway (i.e. one side of the pathway generates the value and the other side receives that value),
- can activate a computational process if the process is sensitive to the signal considered.

The activation means that the processes contained in the architecture body of the VHDL-module will be executed. The computational process will be activated if at least one of the signals contained in the sensitivity list changes its value in the current simulation cycle. The possible new value assigned to a signal is checked in the next simulation cycle. Self-activation is of course possible, too. When a process is sensitive to a signal, it is sensitive to events on that signal only, and not to general transactions being propagated over its data pathway.

Simulation runs only if there occurs an event on at least one of the signals in the process sensitivity list. However, it is possible to impose some extra conditions on these signals to effect that if a proper change of the signal value takes place and the conditions are not fulfilled the computational process will not be activated.

Example

Let $\mathbf{y} = \mathbf{f}(\mathbf{x}^{\mathbf{a}}_{1},..,\mathbf{x}^{\mathbf{a}}_{\mathbf{m}},\mathbf{x}_{\mathbf{m}+1},..,\mathbf{x}_{\mathbf{n}})$; $\mathbf{y} \in \mathbf{Y}$, $\mathbf{x}_i \in \mathbf{X}_i$ where $i = 1.. n$, (1)

n - number of all signals, m - number of "active signals"

be a given functional relationship between $\mathbf{x}_i$ and **y** where $\mathbf{X}_i$ and **Y** are sets of vectors of signals. Let us think of **f** as a behaviour description of a device and of **y** as a response of the device to input signals $\mathbf{x}_i$. Signals $\mathbf{x}^{\mathbf{a}}_{1},..,\mathbf{x}^{\mathbf{a}}_{\mathbf{m}}$ and $\mathbf{y}^{\mathbf{a}}$ ($\mathbf{y}^{\mathbf{a}}$ is the part of the response y, which can be redirected to the input and cause self-activation) can activate computational process. Others signals (i.e. $\mathbf{x}_i$;i = m+1...n) are used for computations only and are not "active signals". Let us consider the following example ,where only three sets **X** , $\mathbf{X}^{\mathbf{a}}$ and **Y** are present:

```
entity Structure_of_device is
   generic (f1,f2,...fk);
-- fi - declarations for exact or
-- approximate description of f
   port ( x : in    datatypeof_x;
          x^a: in    datatypeof_x^a;
          y : inout  datatypeof_y );
end Structure_of_device;

architecture Behaviour of Structure_of_device is
  ...
begin
  process (x^a,y^a)
  ...
  begin
  -- "implementation of y=f(x,x^a,y)"
  ...
  end process;
end Behaviour;
```

When a system to be modelled is complex and consists of many elements performing different or the same behaviour (e.g. neural networks), it is reasonable to build the whole model from single devices of simpler functionality. These subdevices, named "components" in VHDL can communicate with each other over their ports, which may be an input,an output or both (inout ports). The described feature of the structure is modularity. The construction of hierarchy is a simple task in VHDL.

3. DATA PROCESSING IN NEURAL NETWORKS

Example 1

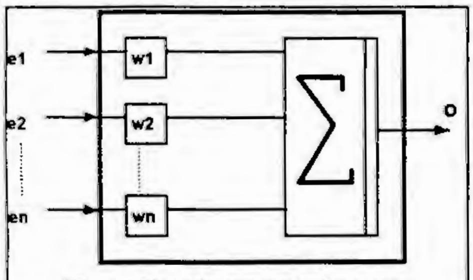


Fig.1. Classic model of neuron

Let us begin with the following model of a single neuron (Fig.1) frequently used in the literature. The output of the neuron is computed by multiplying each input e_i (i=1,2,...,n) by its associated weight wi and then adding all results up. If the sum is greater than or equal to the threshold value of 0, the 1 is applied to the output, otherwise the element generate the 0. The inputs and the output are usually interpreted as directed signals propagating data from elements of one layer to those of the successive layer of the neural net. Any change of data on any input causes the described data processing for a possible new value of the output. The weights are taken to be passive parameters and they can be also declared in this way.

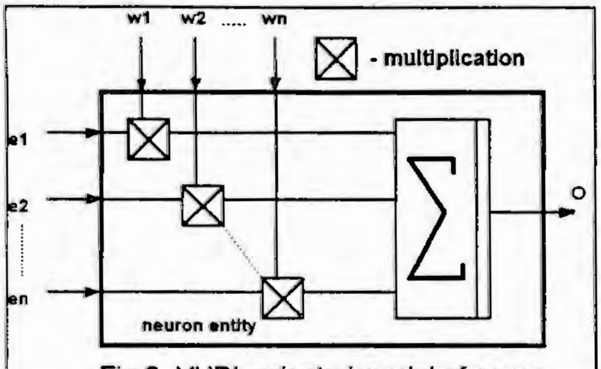


Fig.2. VHDL-oriented model of neuron

It is however advantageous in VHDL to declare all inputs, outputs and weights as signals. With reference to the outline of a program given in Section 2, the input vector **e** can be interpreted as the active vector $\mathbf{x}^a$, the vector of weights **w** - as the inactive vector **x**, and the output o - as the single signal **y**. The vector y^a is not present in this case and the generic declaration can determine the threshold parameters of the element. These declarations make it necessary to change the model of neuron - see Fig.2. The conclusion is: both inputs and weights are signals of different functions; they are discriminable according to whether or not they are active in the process.

Note that only the data propagation is described here. Although the learning requires a special procedure, both the old computational behaviour and declarations stay, and new learning behaviour and new declarations needed can be added easily. The neuron model described here can be implemented in VHDL as follows:

```
entity Neuron is
   generic(NumberOfInputs:natural;
           TresholdValue :natural);
   port  (e          :in    RealArr(1 to NumberOfInputs);
          o          :out real:=0.0;
          w          :in    RealArr(1 to NumberOfInputs);
          );
 end Neuron;
 architecture Classic of Neuron is
    function CalculateSum(Inputs:RealArr;Weights:RealArr) return real is
      begin
      ...                                                  -- definition of function
      end CalculateSum;
    begin                                                  -- description of behaviour
    Computational_process:
    process(e)
    begin
      if CalculateSum(e,w) > TresholdValue  then  o <= 1.0 ;
      else  o <= 0.0;
      end if;
   end process;
   -- and here is the place for Learning_process !
 end Classic;
```

Example 2

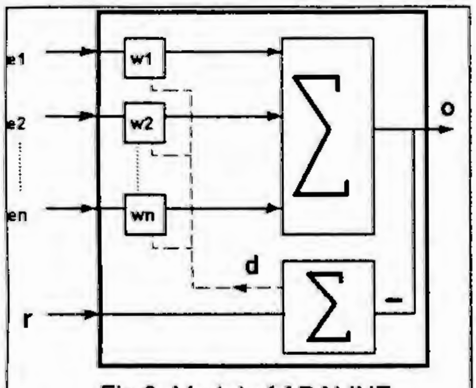


Fig.3. Model of ADALINE

Consider a model of a single neuron named "adaptive linear element" - ADALINE (Fig.3.), which should learn to project the given input vector e into the required output r, while the actual output o differs from r by the value d; see e.g.[5]. The delta method of the form

$$w^{k+1} = w^{k} + \eta de, \qquad k=0,1,\ldots \qquad (2)$$

should be applied to modify the vector of weights, where η is a real number and k denotes the number of learning step.

Assuming that e, w, o and r are defined as signals, the model of ADALINE can be modified in many ways. Figure 4. shows a proposition in which the structure of the neuron presented in Figure 2 is used. Here, however, the weight signals w_i should be placed in the sensitivity list to assure the activation of the neuron for a constant input signals. The training entity generates no events on weight signals if the neuron has been learned satisfactorily. It is of course also possible to integrate the module updating the weights into the neuron entity and thus to make it sensitive to the signal d, or to perform the whole data processing inside the neuron entity - alternatives are a few.

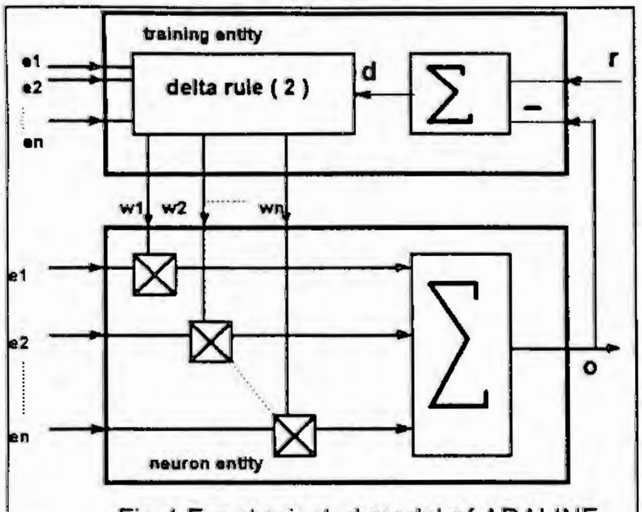


Fig.4.Event oriented model of ADALINE

4. FINAL REMARKS

In the paper the features of VHDL have been presented and it has been shown how the capabilities of VHDL can be used for modelling and simulation of artificial neural networks. Due to the introduction of the concepts of a signal, quasi-parallel data processing, modularity and event driven simulation, VHDL can be successfully used for the considered task.

ACKNOWLEDGEMENT

This work was inspired by the cooperation with the Dept.of Data processing, University of Duisburg, Germany, and partly done with support of the European Communities, Ref.No: ERB-CIPA-CT-92-2058/8411.

REFERENCES

[1] P.J.Ashenden, The VHDL Cookbook. Univ.of Adelaide, S.Australia,1991.

[2] IEEE Standard 1076 VHDL Language Reference Manual. IEEE, New York, 1988.

[3] J.Prang, and P.S.Szczepaniak, Simulation of a controlled preheating process in power stations - a new approach using VHDL. Proceedings of the 11th IASTED Int.Conference: Modelling, Identification and Control, Innsbruck, Austria, 1992, 203-206.

[4] J.Prang, H.-D.Huemmer, and P.S.Szczepaniak, Modelling of power plants using VHDL. 7th Int.Symp. "System-Modelling-Control",Zakopane, Poland, 1993, Vol.2., 111-115.

[5] R.Tadeusiewicz, Neural networks, Akademicka Oficyna Wydawnicza RM, Warszawa, 1993 (in Polish).

ON A MATHEMATICAL MODEL FOR THE HEAT TRANSFER THROUGH A TIM-WALL

A. Handlovicová[*] and R. Van Keer [**]

[*] Institute for Applied Mathematics, Comenius University, 85 102 Bratislava, Slovakia
[**] Department of Mathematical Analysis, University of Ghent, 9000 Gent, Belgium

Abstract. We present a mathematical model for the evaluation of the heat gain through an external TIM-wall of a building, exposed to solar radiation. We state the underlying heat transfer problem in the multicomponent structure under consideration and describe the source terms arising from the solar heating. The much involved problem can be given a nonstandard variational formulation, suitable for numerical approximations.

1. INTRODUCTION

This paper deals with a mathematical model for the transient heat transfer through an external TIM-wall of a building, exposed to solar radiation. Here TIM stands for 'transparent isolation material', recently deviced for building elements for reducing the energy demand for the heating of buildings. In a nutshell, TIM combines the transparency for short wave radiation and a good heat resistance, the latter in contrast to standard glazing. Although intensive experimental research on TIM-building elements is reported on, see e.g. [3], no comprehensive mathematical approach seems to exist, evaluating their thermal performance and rational energy use potential.

To fix the ideas, consider a typical external TIM-wall, with a single isolation sheet, as shown in Fig. 1.

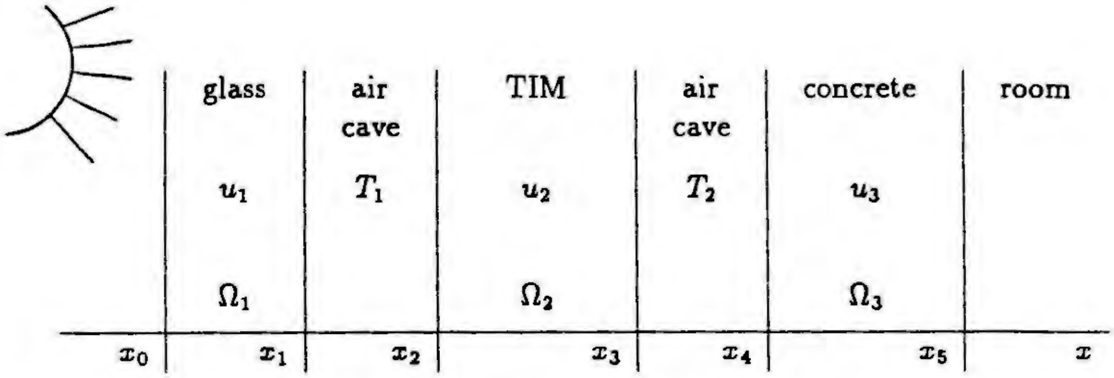


Fig. 1. A cross section of a TIM-wall

A relevant physical quantity is the heat gain, given by the heat flux at the indoor wall surface, $x = x_5$, related to (a) the influence of the time varying solar radiation, incident upon the glass sheet and (partly) passing through the composite medium up to the surface $x = x_4$ of the opaque material (concrete), (b) the thermo-physical properties of the various layers in the multicomponent structure, including the respective convective and radiative transfer coefficients. For the analysis of the thermal behaviour of the TIM-wall, also the temperature raise inside the TIM-material has to be taken into account, with respect to its durability.

The underlying mathematical problem is to determine the temperatures u_i, $1 \leq i \leq 3$, in the glass, the TIM-sheet and the concrete respectively, as well as the air temperatures T_1 and T_2 in the two caves.

For the sake of simplicity, we deal with a one dimensional model, the heat transmission through the wall being in one direction only, orthogonal to the parallel surfaces, which is a standard assumption in heat transfer problems in building physics. Likewise the air in the caves is taken to be homogeneously at the temperatures $T_1(t)$ and $T_2(t)$ respectively. Moreover the caves are not ventilated.

2. THE GOVERNING EQUATIONS

2.1. The differential equations (DEs) and boundary conditions (BCs)

In the slabs Ω_i, $1 \leq i \leq 3$, heat is transferred by conduction. Hence the temperatures u_i, $1 \leq i \leq 3$, obey the DEs

$$c_i \cdot \rho_i \cdot \frac{\partial u_i}{\partial t} = \frac{\partial}{\partial x}(k_i \cdot \frac{\partial u_i}{\partial x}) + f_i(x,t;u_i), \qquad x \in \Omega_i,\ t > 0 \tag{2.1}$$

where c_i, ρ_i and k_i (with respective units J/kgK , kg/m^3 and W/mK) represent the specific heat, the density and the thermal conductivity of the material in Ω_i respectively. In general we may allow for the case $c_i = c_i(x,t;u_i)$, etc. In (2.1) the source term f_i correspond to the solar heating of the wall, $[W/m^3]$. In the opaque material Ω_3 we have $f_3 = 0$ throughout.

At the outdoor wall surface heat is transferred by convection and radiation, i.e.

$$k_1 \cdot \frac{\partial u_1}{\partial x} = h_{co} \cdot (u_1 - u_0) + h_{ro}\,(u_1^4 - F_{SE}\,u_0^4 - F_{SS}\,u_{sky}^4), \qquad x = x_0, t > 0 \tag{2.2}$$

Here h_{co} and h_{ro} are the convective and radiative heat transfer coefficients at $x = x_0$ respectively, $[W/m^2K]$, (which may depend on t and u_1), while $u_0 = u_0(x_0,t)$ is the outdoor temperature and $u_{sky} = u_{sky}(x_0,t) = u_0 - 21$ (or $\alpha \cdot u_0, \alpha = 0.93$) . Moreover $F_{SE} = (1 - cos\gamma)/2$ is the view factor wall- surroundings, γ being the inclination angle, $[^0]$, of the wall with respect to a horizontal plane, and $F_{SS} = 1 - F_{SE}$ is the view factor wall-sky.

At the indoor wall surface we may assume heat to be transferred by convection only, i.e.

$$-\,k_3 \cdot \frac{\partial u_3}{\partial x} = h_{c5}(u_3 - u_{in}), \qquad x = x_5,\ t > 0 \tag{2.3}$$

where h_{c5} is the convective heat transfer coefficient at $x = x_5$ (possibly adjusted to count for linearized radiation) and $u_{in} = u_{in}(x_5,t)$ is the indoor air temperature.

The transfer coefficients h_{co}, h_{ro} and h_{c5} depend on the material properties. Typically, one has $h_{co} = a + bv$, where v is the velocity of the air flow parallel to the plane surface and a and b are constants depending on the type of the surface (when the velocity of the wind exceeds 5.03 m/s , a power law, $h_{co} = c \cdot v^d$, with c and d constants, is more appropriate), see e.g. [5]. Moreover $h_{ro} = e_0 \cdot \sigma$, where σ is the Stefan-Boltzmann constant and e_0 is the emissivity of the surface $x = x_0$.

2.2. The transmission conditions (TCs)

At the surface $x = x_1$ of the glass sheet heat is transferred by convection to the air in the cave and by radiation to the opposite surface of the cavity, i.e.

$$-k_1 \cdot \frac{\partial u_1}{\partial x} = h_{c1} \cdot (u_1 - T_1) + h_{r1} \cdot (u_1^4 - u_2^4(x_2, t)) \ , \ x = x_1, t > 0$$

A similar situation occurs at the surfaces $x = x_2$ and $x = x_3$, explicitly ,

$$k_2 \cdot \frac{\partial u_2}{\partial x} = h_{c2} \cdot (u_2 - T_1) + h_{r2} \cdot (u_2^4 - u_1^4(x_1, t)) \ , \ x = x_2, t > 0$$

$$-k_2 \cdot \frac{\partial u_2}{\partial x} = h_{c3} \cdot (u_2 - T_2) + h_{r3} \cdot (u_2^4 - u_3^4(x_4, t)) \ , \ x = x_3, t > 0$$

In the opaque material we had $f_3 = 0$ throughout. Correspondingly the influence of the solar radiation on this layer is thought to be concentrated at the surface $x = x_4$. This give rise to a supplementary term, denoted by f_{SOL} , in the heat flux expression, viz.

$$k_3 \frac{\partial u_3}{\partial x} = h_{c_4} \cdot (u_3 - T_2) + h_{r4} \cdot (u_3^4 - u_2^4(x_3, t)) + f_{SOL}, \ x = x_4, t > 0 \qquad (2.4)$$

Again the convective and radiative transfer coefficients may vary with time and may depend on the respective temperatures in the slabs, thus $h_{c1} = h_{c1}(t, u_1(x_1, t))$, etc. The term f_{SOL} entering (2.4) will be described in the next section, as it is also the case with the terms f_1 and f_2 , arising in (2.1).

2.3. The heat balance equations in the air caves

In the air caves, which are neither ventilated nor heated, the change of heat, corresponding to a change of the air temperature, is in balance with the heat gained / lost by convection at the two surfaces of the cave. Thus, for the first cavity, when during a time interval $(t, t + \Delta t)$, the temperature change is ΔT_1, we have

$$C_p \cdot V_1 \cdot \Delta T_1 = [h_{c1} \cdot (u_1(x_1, t) - T_1(t)) + h_{c2} \cdot (u_2(x_2, t) - T_1(t))] \cdot \Delta t \cdot S$$

or, in the limit $\Delta t \to 0$,

$$\frac{dT_1}{dt} = \frac{1}{\ell_1 \cdot C_p} \cdot [h_{c1} \cdot (u_1(x_1, t) - T_1(t)) + h_{c2} \cdot (u_2(x_2, t) - T_1(t))], \quad t > 0 \qquad (2.5)$$

where

C_p = the thermal capacity of the air, $[J/Km^3]$
V_1 = the volume of the 1st cave, $[m^3]$
S = the area of the surfaces of the caves, $[m^2]$
ℓ_1 = $x_1 - x_0$ = the length of the cave, $[m]$

Similarly, $T_2(t)$ obeys

$$\frac{dT_2}{dt} = \frac{1}{\ell_2 \cdot C_p} \cdot [h_{c3} \cdot (u_2(x_3,t) - T_2(t)) + h_{c4} \cdot (u_3(x_4,t) - T_2(t))], \; t > 0 \qquad (2.6)$$

where $\ell_2 = x_4 - x_3$.

2.4. Initial conditions

The system (2.1)–(2.6) for the five temperatures $u_1(x,t), \ldots, T_2(t)$ must be completed by the initial data at $t = 0$, corresponding to sunrise, say

$$u_i(x,0) = u_i^{(0)}(x), \; x \in \Omega_i, \; 1 \le i \le 3; \; T_j(0) = T_j^{(0)}, \; j = 1,2. \qquad (2.7.a\text{–}b)$$

3. THE HEAT GENERATION

In (2.1) the source terms f_1 and f_2 represent the internal heat generation in the glass sheet and the TIM-layer respectively, due to the solar radiation. Likewise, the contribution F_{SOL} to the heat flux at $x = x_4$, see (2.4), represents the influence of the solar radiation on the opaque material, which is taken to be concentrated at this surface.

3.1. The source terms f_1 and f_2 (transparent materials)

This subsection is based upon [2], see also [1].
The source term in the transparent material Ω_1 and Ω_2 can be written as

$$f_i = \alpha_{bi} \cdot q_{bi} + \alpha_{di} \cdot q_{di} \, , \quad i = 1,2 \qquad (3.1)$$

where q_{bi} and q_{di} stand for the density of heat flow, $[W/m^2]$, of the direct (beam) radiation and diffuse radiation respectively and where α_{bi} and α_{di} are the corresponding internal absorption coefficients, $[1/m]$. The coefficients α_{bi} are determined experimentally, while α_{di} may be taken to be constant and may be derived from α_{bi} .

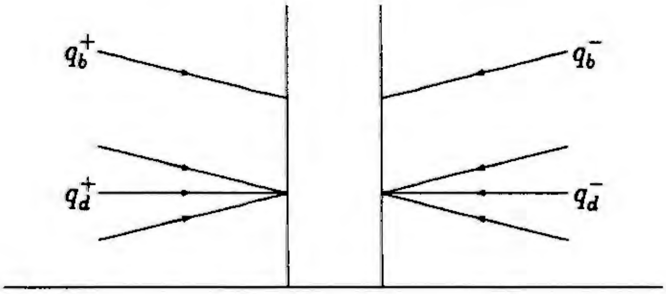


Fig. 2. Direct (beam) and diffuse radiation in a slab.

The solar radiation incident on transparent material is partly reflected at the boundary, partly absorbed and partly transmitted (the internal reflection and scattering may be disregarded in a first model). Therefore, in a one dimensional situation, we have (when dropping the index i)

$$q_b = q_b^+ - q_b^- \quad , \quad q_d = q_d^+ - q_d^- \qquad (3.2.a\text{–}b)$$

where the plus sign indicates the part of the (beam or diffuse) solar radiation going from the left (outdoor) to the right (indoor), while the minus sign indicates solar radiation parts travelling in the opposite direction.

The direct solar radiation passing through Ω_i from left to right is absorbed at a rate α_{bi} per unit distance and per unit density of heat flow. Hence q_{bi}^+ varies with x according to the DE

$$\frac{\partial q_{bi}^+}{\partial x} = -\alpha_{bi} \cdot q_{bi}^+ , \qquad x \in \Omega_i \quad , \quad i = 1, 2 \tag{3.3}$$

Similarly, the direct solar radiation passing through Ω_i from right to left obeys

$$\frac{\partial q_{bi}^-}{\partial x} = \alpha_{bi} \cdot q_{bi}^- , \qquad x \in \Omega_i \quad , \quad i = 1, 2 \tag{3.4}$$

The equations of radiation transfer for the diffuse radiation parts q_{di}^+ and q_{di}^- are the same as (3.3) and (3.4) respectively, with α_{bi} replaced by α_{di} .

To obtain q_{bi}^+ and q_{bi}^- from (3.3) and (3.4) appropriate BCs have to be added.
We denote by $q_{xb} = q_{xb}(t; \beta, \gamma; \lambda)$ the density of the heat flow of the direct radiation incident on the surface $x = x_0$, λ being the wave length and β the angle of incidence. q_{xb} is obtained experimentally.
Denoting the reflection coefficients of the glass and TIM-surfaces by ρ_1 and ρ_2 respectively, the appropriate BCs for q_{b1}^+ at $x = x_0$ and for q_{b2}^+ at $x = x_2$ read

$$q_{b1}^+(x_0) = (1 - \rho_1) \cdot q_{xb} \tag{3.5a}$$

$$q_{b2}^+(x_2) = q_{b1}^+(x_1) \cdot (1 - \rho_1) \cdot (1 - \rho_2). \tag{3.5b}$$

Note that in the last equation the factor $q_{b1}^+(x_1)$ is known from the problem (3.3), (3.5a) for q_{b1}^+, while the factor $(1 - \rho_1)$ counts for the transmission of that beam at $x = x_1$.

The BC for the beam q_{b1}^- passing through Ω_1 from right to left may be written in the form

$$q_{b1}^-(x_1) = -q_{b1}^+(x_1) \cdot \rho_1 - q_{b1}^+(x_1) \cdot (1 - \rho_1).\rho_2 \cdot (1 - \rho_1) \tag{3.6a}$$

where the first term at the right-hand side corresponds to the reflected part of the beam incident from the left on the surface $x = x_1$, while the second term corresponds to the transmitted part of that beam, which is then further reflected at the surface $x = x_2$ (giving rise to the factor ρ_2) and finally is transmitted again at the surface $x = x_1$, now from right to left of course (inducing the last factor $(1 - \rho_1)$).

Similarly, the BC for q_{b2}^- reads

$$q_{b2}^-(x_3) = -q_{b2}^+(x_3) \cdot \rho_2 - q_{b2}^+(x_3) \cdot (1 - \rho_2) \cdot \rho_3 \cdot (1 - \rho_2) \tag{3.6b}$$

where ρ_3 is the reflection coefficient of the surface $x = x_4$ and where $q_{b2}^+(x_3)$ is already known from the problem (3.3), (3.5b) for q_{b2}^+.

In summary, the equations (3.2a)–(3.6) constitute a closed system for q_{bi}, $x \in \Omega_i$, $i = 1, 2$. By a completely analogous system the diffuse radiation densities q_{di}, $x \in \Omega_i$, $i = 1, 2$, may be determined, now leaving from the density of heat flow, $q_{xd} = q_{xd}(t; \gamma; \lambda)$, of the diffuse radiation incident on the surface $x = x_0$. This concludes the representation of the source terms f_1 and f_2, (3.1).

3.2. The term f_{SOL} (opaque material)

By the same arguments as above we have for the contribution of the solar radiation (only from the left to the right) to the heat flux at $x = x_4$:

$$f_{SOL} = [q_{b2}^{+}(x_3) + q_{d2}^{+}(x_3)] \cdot (1 - \rho_2) \cdot (1 - \rho_3) \tag{3.7}$$

the two terms at the right-hand side corresponding to the beam and diffuse radiation incident from the left on the surface $x = x_3$, transmitted at this surface, passing through the air space and being not reflected (i.e. being 'kept') at the surface $x = x_4$ of the opaque material.

4. AN IMPERFECT THERMAL CONTACT PROBLEM. VARIATIONAL FORMULATION.

The problem (2.1)–(2.7) can be reformulated as an imperfect thermal contact problem in a multicomponent domain, by 'eliminating' the cavities. First the explicit expressions of $T_1(t)$ and $T_2(t)$, obtained from the first order initial value problems (2.5)–(2.6), (2.7b), e.g. (for the case that h_{c1} and h_{c2} are constant in time),

$$T_1(t) = T_1^0 \cdot e^{-(1/\ell_1 \cdot C_p)\cdot(h_{c1}+h_{c2})\cdot t} + \frac{1}{\ell_1 \cdot C_p} \cdot \int_0^t [h_{c1} \cdot u_1(x_1, \tau) + h_{c2} \cdot u_2(x_2, \tau)] e^{-(1/\ell_1 \cdot C_p)\cdot(h_{c1}+h_{c2})\cdot(t-\tau)}\, d\tau \tag{4.1}$$

are substituted in the TCs (2.4).

Next, the domain Ω_2 is shifted to the left so as to be adjacent to the domain Ω_1 and similarly Ω_3 is shifted to the left to be adjacent to Ω_2. We end up with a parabolic problem for a scalar unknown function $u(x,t)$ in a multicomponent domain $\Omega = \Omega_1 \cap \Omega_2 \cap \Omega_3$, where $u_i = u|_{\Omega_i}$, $1 \leq i \leq 3$. Both the flux and the temperature jump at the interfaces of the subregions (which now read $x = x_1 = x_2$ and $x = x_3 = x_4$), according to (2.4a–b) and (2.4c–d). The jumps turn out to be expressed in terms of Volterra operators acting on the traces of the unknown function from both sides of the internal boundaries, see (4.1). Parabolic problems of related types are studied in [4] and [6].

The elimination of the cavities, to arrive at an imperfect thermal contact problem, allows us to recast the much involved problem for the temperatures u_i, $1 \leq i \leq 3$, into a transparent, although nonstandard, variational form, which is suitable for numerical approximations. We proceed in two steps.

First we deal with the boundary value problem for u_i in the usual way, i.e. we multiply both sides of the DE (2.1) with a testfunction $v_i \in H^1(\Omega_i)$, $H^1(\Omega_i)$ being the first order Sobolev space on the domain Ω_i, we integrate over Ω_i, we apply integration by parts to reduce the order of differentiation by one and finally we invoke the BCs (2.2)–(2.3) and the TCs (2.4) [Hereby, of course, a change of notation is made, corresponding to the shiftings, mentioned above]. Next, the resulting variational equations for u_i, $1 \leq i \leq 3$, are added. This results in a variational equation for $u = [u_1, u_2, u_3] \in V = H^1(\Omega_1) \times H^1(\Omega_2) \times H^1(\Omega_3)$ of the type

$$(w \cdot \frac{\partial u}{\partial t}, v) + a(u,v) + b(u,v) + g(u, V(u); v) = (f, v) + m(v), \quad \forall v = [v_1, v_2, v_3] \in V. \tag{4.2}$$

Here $(w \cdot \frac{\partial u}{\partial t}, v)$, $a(u,v)$ and (f,v) arise from the parabolic term, the elliptic term and the source term of (2.1) respectively. Moreover $b(u,v)$ and $m(v)$ correspond to the BCs (2.2)–(2.3). Finally, $g(u, V(u); v)$ arises from the TCs (2.4), including Volterra operators acting on u of the type (4.1), which is summarized by the notation $V(u)$.

Leaving from (4.2) numerical approximation methods may be outlined, such as the Rothe finite element method, developed e.g. in [4] and [6] for related parabolic problems.

5. RESULTS

Above we briefly described a mathematical model for evaluating the heat gain through an external TIM-wall of a building, exposed to solar radiation, as well as the temperature raise in the TIM-layer. After recasting the mathematical problem in a variational form, the physical relevant flux, (2.3), and temperature may be obtained by an elegant and reliable numerical method.

6. REFERENCES

[1] Duffic, J.A. and Beckman, W., Solar Energy Thermal Processes. J. Wiley, New York, 1980.

[2] Hens, H. and Verbeeck, G., A note on the solar radiation transfer through transparent insulation material. Technical Report of the Laboratory of Building Physics, Catholic University of Leuven (KUL) (1991).

[3] Fesh, L.F., Proceedings of the 6th International Meeting on Transparent Insulation Technology (June 3–5, 1993, Birmingham, UK). The Franklin Company Consultants Ltd, Birmingham, 1993.

[4] Kačur, J. and Van Keer R., On the numerical solution to semilinear parabolic problems in multicomponent structures with Volterra operators in the transmission conditions and in the boundary conditions. To appear in : Zeitsch. Angew. Math. Mech.

[5] Pratt, A.W., Heat Transmission in Buildings. J. Wiley, Chichester, 1981.

[6] Van Keer R. and Kačur, J., On the numerical solution of nonlinear heat flow through composite media with imperfect contacts. In : Ch. Hirsch et al. (Ed.), Numerical Methods in Engineering '92. Elsevier, Amsterdam, 1992.

HEAT-DYNAMICS MODEL OF A HOME

Vladimír HAMATA, Jaromír KŘEMEN, Miroslav PŘEUČIL, Rudolf PAĎOUK
Civil Engineering Faculty, Czech Technical University Prague
Thákurova 7, CZ-166 29 Prague 6, Czech Republic

Abstract.
A continuous simulation model of a living home approximated by means of lumped parameters, a heating-system model and a model of automatic heat control are introduced. The simulation model helps in designing the desired heat-dynamics of the building (even for improvement or reconstruction purposes) and in investigation of the function of different types of controlled heating systems under given dynamics of the building. The model further enables selection of daily/weekly regimes of heating that yield maximal energy savings and assures thermal comfort of the interior. Results for a real family home are introduced as an illustration.

1. INTRODUCTION

Simulation model of the heat dynamics of a building presented in this paper is described by physical quantities like heat-capacity, heat-conductivity, temperature, power, etc. The user on the other hand thinks and uses "construction" quantities like dimensions of walls, columns, ceilings, floors etc. and their material composition, various types of radiators, ... Transformation of these two types of quantities is provided by an interface not introduced in this paper - the attention is given to he simulation model itself.

2. MATHEMATICAL MODEL

The over-all mathematical model consists of mathematical models of following systems:

- the heating system (i.e. heat-producing system),
- the control system and
- the building object as well.

The mathematical model of the heat-producing system is created by models of the heating agregate (i.e. boiler or heat-accumulating tank etc.). The model of the control system consists of submodels of the boiler- and room-thermostats, mixing valve, hot-water pump, pipe- and radiator systems.

For illustration the heating agregate is in our case represented by a water-accumulation tank with 14 cubic meters volume. It is heated by electric power (Power [W]), which is switched on by the signal PowerOn, if the temperature of water (Ts) is lower than its upper limit (in our case 98 °C) and simultaneously the interval suitable for the night cheaper rate goes on. The power from the heat-capacity Cs (6.0e+7 Ws/°C) of the accumulating tank is withdrawn partly by the imperfect heat-insulation (heat-conductivity G_{SB} between Ts and the temperature of the ground T_B) and partly in form of a utilizable power for heating. The temperature Ts in time t is given by the following equation (T_{S0} is the initial temperature)

$$T_S=T_{SO}+\frac{1}{C_S}.\int_0^t\left[\text{PowerOn.Power}-C_{H_2O}.V.\text{PumpOn}.(T_{SMIX}-T_R)-G_{SB}.(T_S-T_B)\right]d\tau \qquad (1)$$

The mixing valve is described by the relation $T_{SMIX}=\mu.T_S+(1-\mu).T_R$,
where: T_R is the temperature of water which is fed back
μ is the mixing ratio.

The heat-water pump ensures - the signal PumpOn - the movement of heating water with the specific volume heat C_{H_2O} [$Ws/m^3.°C$] by the speed v [m^3/s] in dependence on the function of the room thermostat.

Water is pumped through the distributing pipes, which are represented by the heat-capacity C_P. Heat-conductivity G_{PA} between the temperature T_P in the distributing pipes and temperature T_{AIR} of the interior air represents heat-loss on the conduit. Temperature T_P is given as follows

$$T_P=T_{PO}+\frac{1}{C_P}.\int_0^t\left[C_{H_2O}.V.\text{PumpOn}.(T_{SMIX}-T_P)-G_{PA}.(T_P-T_{AIR})\right]d\tau \qquad (2)$$

The radiator with a heat-capacity C_R is connected to the distributing pipes. The heat-power drawn from it is given by the heat-conductivity G_{RA} between the radiator and air and by the difference of their temperatures as well. The temperature T_R of the water leaving the radiator is given by

$$T_R=T_{RO}+\frac{1}{C_R}.\int_0^t\left[C_{H_2O}.V.\text{PumpOn}.(T_P-T_R)-G_{RA}.(T_R-T_{AIR})\right]d\tau \qquad (3)$$

According to the current purpose of our model the heat dynamics of the building is highly comprimated, i.e. less structured. The cladding is approximated by its heat-capacity divided into two parts (internal and external one) that are mutually connected with the heat-conductivity. The mathematical model of the building consists of the mathematical descriptions of phenomena on heat-capacities and heat-conductivities of the following parts of the building:

C_A [$W.s/^oK$] heat-capacity of the air in the building's interior
C_I heat-capacity of the interior (partition walls and interior equipment)
C_{CI} heat-capacity of the internal part of the cladding
C_{CO} heat-capacity of the external part of the cladding
C_{LOFT} heat-capacity of the internal loft construction
G_{AB} [$W/^oK$] heat-conductivity between temperature of air T_{AIR} and ground T_B
G_{AI} heat-conductivity between T_{AIR} and temperature of interior T_I
G_{ACI} heat-conductivity between T_{AIR} and internal part of coat T_{CI}
G_{WND} heat-conductivity between T_{AIR} and outer temperature T_{OUT}
G_{CLAD} heat-conductivity between T_{CI} on C_{CI} and T_{CO} on C_{CO}
G_{COO} heat-conductivity between T_{CO} on C_{CO} and outer temperature T_{OUT}
G_{AL} heat-conductivity between T_{AIR} on C_A and T_{LOFT} on C_{LOFT}
G_{LO} heat-conductivity between T_{LOFT} on C_{LOFT} and outer temperature T_{OUT}

This model is represented by following equations:

$$T_{AIR}=T_{AIRO}+\frac{1}{C_{A}}\cdot\int_{0}^{t}\Big[G_{RA}.(T_{R}-T_{AIR})+G_{ACI}.(T_{CI}-T_{AIR})+G_{AL}.(T_{LOFT}-T_{AIR})+ +G_{AB}.(T_{B}-T_{AIR})+G_{WND}.(T_{OUT}-T_{AIR})+G_{AI}.(T_{I}-T_{AIR})+G_{PA}.(T_{P}-T_{AIR})\Big]\,d\tau \qquad (4)$$

$$T_{I}=T_{IO}+\frac{1}{C_{I}}\cdot\int_{0}^{t}\Big[G_{AI}.(T_{AIR}-T_{I})\Big]\,d\tau \qquad (5)$$

$$T_{CI}=T_{CIO}+\frac{1}{C_{CI}}\cdot\int_{0}^{t}\Big[G_{ACI}.(T_{AIR}-T_{CI})+G_{CLAD}.(T_{CO}-T_{CI})\Big]\,d\tau \qquad (6)$$

$$T_{CO}=T_{COO}+\frac{1}{C_{CO}}\cdot\int_{0}^{t}\Big[G_{COO}.(T_{OUT}-T_{CO})+G_{CLAD}.(T_{CI}-T_{CO})\Big]\,d\tau \qquad (7)$$

$$T_{LOFT}=T_{LOFTO}+\frac{1}{C_{LOFT}}\cdot\int_{0}^{t}\Big[G_{AL}.(T_{AIR}-T_{LOFT})+G_{LO}.(T_{OUT}-T_{LOFT})\Big]\,d\tau \qquad (8)$$

The model is implemented on standard IBM PC architecture using simulation system PSI/e that has been developed at TU Delft. The results i.e. time dependencies of selected variables are shown at Fig.1. These results were obtained for two-storey brick-built family home with six rooms built in 1930.

3. RESULTS

The simulation model of the introduced structure represents the general view on the heat-dynamics of the building. It was formed by an order of designer and not by the demand of relatively simple approximation of a real building with a heating system as might seem. Richly structured model are under development to study other related problems.

A set of experiments has been made with the presented model - at the beginning to verify it by comparing it to the real object, later to study the heating regimes in buildings with different heat dynamics under different external climatic conditions.

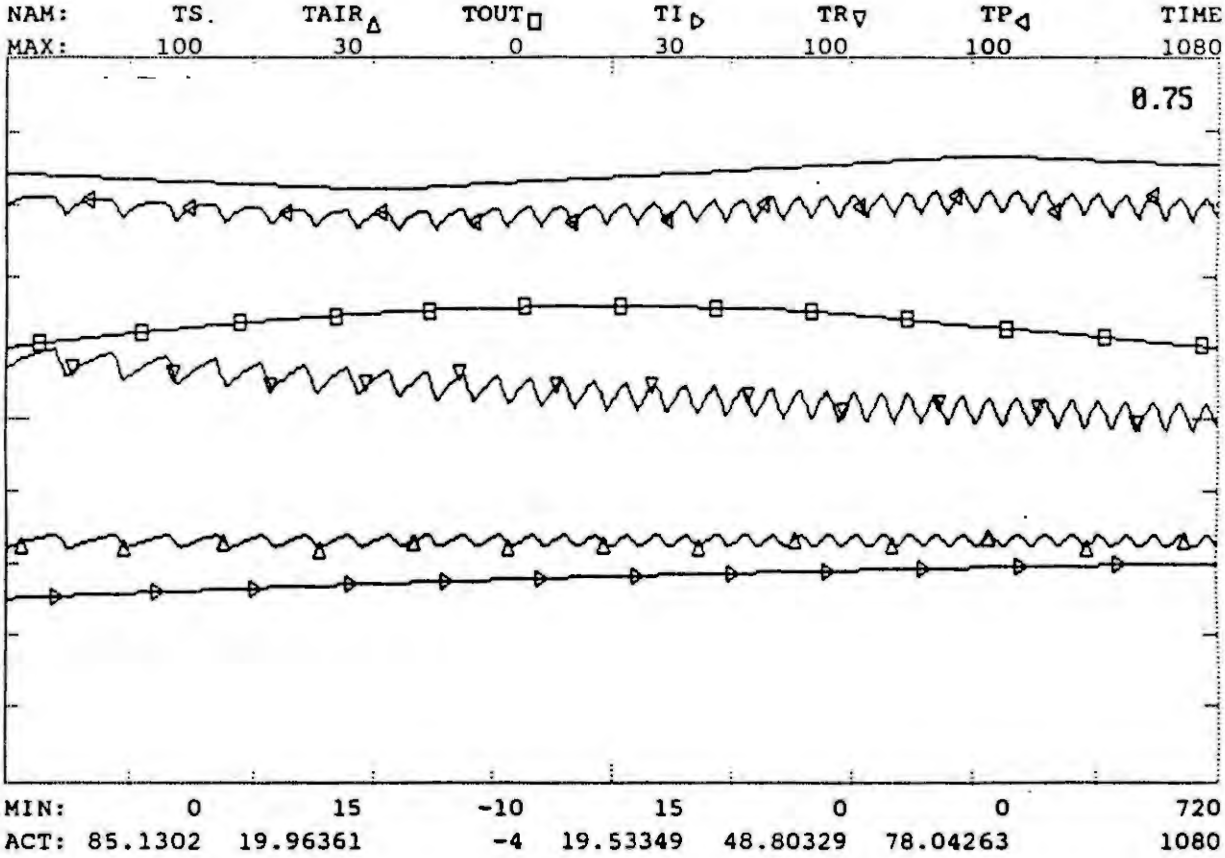
Obtained results show fairly good correspondence of the simulated heat-dynamics phenomena with those observed and measured on the real reference home.

4. REFERENCES

[1] Přeučil et al.: Energy Conservation within Residential and Civic Buildings (part 4). In: Workshop '93 (part C), CTU Prague, 1993.

[2] Hamata, Křemen, Paďouk, Přeučil: Simulation model of the system heat source/controller/heated object as a tool for investigation of thermal dynamical phenomena (in Czech). Research report supported by CTU grant #8003. Prague 1992.

[3] Hamata, Křemen, Paďouk, Přeučil: Simulation of heat dynamics of building for heating-system design (in Czech). Proc.15 Int.Colloq., Ostrava 1993.


NAM: TS. TAIR△ TOUT□ TI▷ TR▽ TP◁ TIME
MAX: 100 30 0 30 100 100 1080
0.75
MIN: 0 15 -10 15 0 0 720
ACT: 85.1302 19.96361 -4 19.53349 48.80329 78.04263 1080

A MATHEMATICAL MODEL OF ISOTROPIC MATERIAL BEHAVIOUR INCLUDING SECOND-ORDER-EFFECTS

HOLM ALTENBACH* and ALEXANDER ZOLOCHEVSKY**

* Institut für Werkstofftechnik und Werkstoffprüfung
Otto-von-Guericke-Universität Magdeburg
Postfach 4120, 39 016 Magdeburg, Deutschland

** Department of Strength and Dynamics of Mashines
Kharkov Technical University
Frunze str. 21, 310002 Kharkov 2, Ukraine

Abstract. Some experimental investigations show significant second-order-effects for isotropic materials in the elastic, creep or failure behaviour. The proposed unified model based on a potential formulation taking into account all three stress tensor invariants. The proposed model containts some material parameters, which can be determed by so-called basic experiments. A good agreement between theoretical and experimental results for multiaxial stress states is obtained.

1. INTRODUCTION

Some isotropic artificial or natural materials (light alloys, cast irons, plastics, ceramics, composites, geomaterials etc.) show in tests several types of non-negligible second-order-effects. These effects are connected with the loading conditions. So we can obtain

- different behaviour in tension and compression,
- different normalized stress-strain-diagramms in tension and torsion,
- dependence upon superposed hydrostatic pressure,
- compressibility,
- *Poynting-Swift*-effect etc.

Examples are shown on Figs. 1 — 4.

2. THEORETICAL BACKGROUND

In the frame of the geometrical linear theory the constitutive equations for elastic and creep behaviour and the criteria of strength (limit state) of this class of isotropic materials can be developed by an unified approach, based on a potential formulation. The potential or the limit surface is given by a funktion F in dependence of some equivalent stress σ_{eq}

$$F = F(\sigma_{eq}). \tag{1}$$

The equivalent stress is a function only of the stress invariants in the case of isotropic materials. In case of classical theories F is a function of the second invariant of the stress deviator (*von Mises*-type theories), and the influence of the first and the third invariant is neglected.

The simplest extension of the *von Mises*-type theories can be given in the following form. The equivalent stress expressions take also into account the first or the third invariant of the stress tensor. In general the function F is more complex. Let us suggest here a dependence on all three invariants.

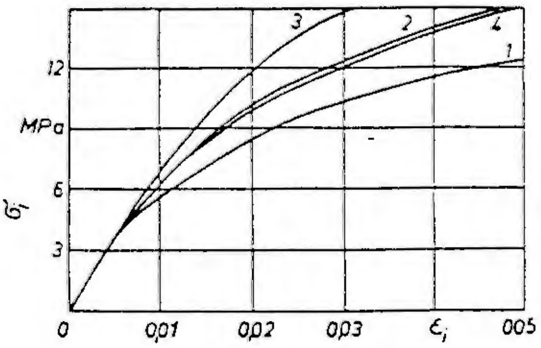


Figure 1: Stress-strain diagramm of Celluloid (after *Nishitani*[2]), $T = 338K$, 1 - tension, 2 - pure torsion, 3 - compression, 4 - uniaxial tension and hydrostatic pressure

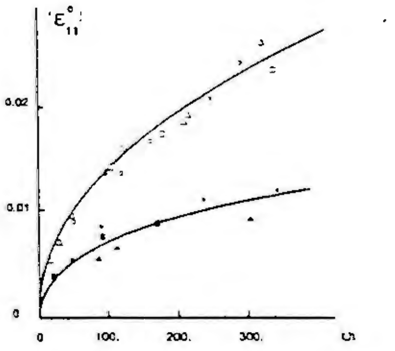


Figure 2: Creep curves of Zircaloy-2 (after *Lucas & Pelloux*[1]), tension - open symbols, compression - full symbols

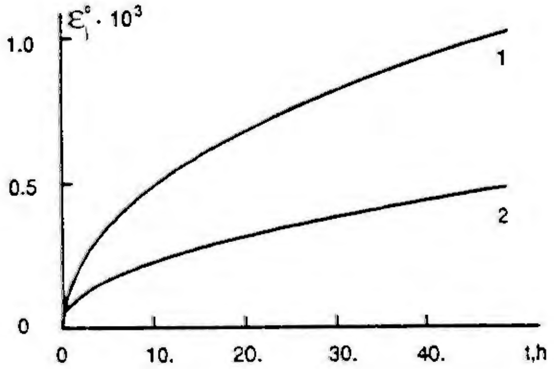


Figure 3: Creep curves of the austenitic steel E1 - 257 (after *Rabotnov*[3]), $\sigma_i = 170 MPa, T = 873K$, 1 - tension, 2 - torsion

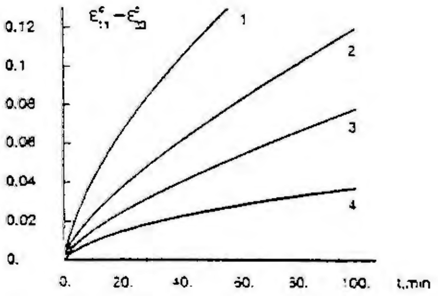


Figure 4: Influence of hydrostatic pressure p (after *Nishitani*[2]), $\sigma_i = 11,5MPa, T = 338K, 1-p=0, 2-p=\sigma_i, 3-p=2\sigma_i, 4-p=3\sigma_i$

There are two different possibilities to define the invariants. The first is connected with the pure mathematical definition of the linear invariant

$$I_1 = \sigma_{ij}\delta_{ij}, \tag{2}$$

the quadratic invariant

$$I_2 = \sigma_{ij}\sigma_{ij} \tag{3}$$

and the cubic invariant

$$I_3 = \sigma_{ij}\sigma_{jk}\sigma_{ik}, \tag{4}$$

where σ_{ij} is the stress tensor, and δ_{ij} is the *Kronnecker*'s delta. Let us suggest some new combinations of these invariants, e.g.

$$\sigma_1 = AI_1, \sigma_2^2 = \frac{1}{2}(BI_2 + CI_1^2), \sigma_3^3 = \frac{1}{3}(DI_3 + LI_1I_2 + MI_1^3). \tag{5}$$

Then we propose the equivalent stress in the following form

$$\sigma_{eq} = \alpha\sigma_1 + \sigma_2 + \beta\sigma_3. \tag{6}$$

Here α and β are numerical coefficients characterizing the "specific weight" of the linear and the cubic invariants, $A, \ldots, M$ are parameters. They should be determined for each material.

Starting from Eq.(6), we assume the existence of a potential in the form

$$F = \frac{1}{2}\sigma_{eq}^2. \tag{7}$$

In the case of the elastic behaviour we get the following constitutive relation

$$\varepsilon_{ij} = \frac{\partial F^{elastic}}{\partial \sigma_{ij}}. \tag{8}$$

For the description of creep we use the associated creep law

$$\dot{\varepsilon}_{ij}^c = \mu \frac{\partial F^c}{\partial \sigma_{ij}}, \tag{9}$$

where μ is a *Lagrange*'s multiplier.

The second possibility is connected with *Novoshilov*'s invariants

$$I_1 = \sigma_{ij}\delta_{ij}, \sigma_i = \sqrt{\frac{3}{2}s_{ij}s_{ij}}, \sin 3\xi = -\frac{27}{2}\frac{det(s_{ij})}{\sigma_i^3}, \tag{10}$$

where $s_{ij} = \sigma_{ij} - I_1\delta_{ij}/3$. In this case we can propose the equivalent stress in the following form

$$\sigma_{eq} = \lambda_1\sigma_i \sin\xi + \lambda_2\sigma_i \cos\xi + \lambda_3\sigma_i + \lambda_4 I_1 + \lambda_5 I_1 \sin\xi + \lambda_6 I_1 \cos\xi. \tag{11}$$

Here λ_i are the material parameters. Then the limit surface can be described by the equation

$$\sigma_{eq} = \sigma_T. \tag{12}$$

3. BASIC EXPERIMENTS

As an example we determine the parameters in Eqs. (5) for creep behaviour. Let us discuss the following basic experiments:

- uniaxial tension

$$\dot{\varepsilon}_{11}^c = L_+\sigma_{11}^n, \dot{\varepsilon}_{22}^c = Q\sigma_{11}^n; \tag{13}$$

- uniaxial compression

$$\dot{\varepsilon}_{11}^c = -L_-|\sigma_{11}|^n; \tag{14}$$

- pure torsion

$$2\dot{\varepsilon}_{12}^c = N\sigma_{12}^n, \dot{\varepsilon}_{11}^c = M\sigma_{12}^n; \tag{15}$$

- hydrostatic pressure

$$\dot{\varepsilon}_{11}^c = \dot{\varepsilon}_{22}^c = \dot{\varepsilon}_{33}^c = -P|\sigma_{11}|^n. \tag{16}$$

Here constants L_+, L_-, Q, N, M, P, n are known from tests.

For determination of the parameters in the strength criterion (11), (12) we can use the following failure tests:

- uniaxial tension

$$\sigma_{11} = \sigma_T; \tag{17}$$

- uniaxial compression

$$\sigma_{11} = -\sigma_C; \tag{18}$$

- pure torsion

$$\sigma_{12} = \tau_T; \tag{19}$$

- thinwalled tube under inner pressure

$$\sigma_{11} = \frac{\sigma_B}{2}, \sigma_{22} = \sigma_B, \sigma_B = \frac{pR}{h}; \tag{20}$$

- uniaxial tension under hydrostatic pressure

$$\sigma_{11} = \frac{F}{A} - q, \sigma_{22} = -q, \sigma_{33} = -q, \sigma_{11} = \frac{2}{3}\sigma^{**}, \sigma_{22} = \sigma_{33} = -\frac{1}{3}\sigma^{**}; \tag{21}$$

- thinwalled tube under inner pressure and tension

$$\sigma_{11} = \frac{F^*}{A^*} + \frac{\sigma_t}{2}, \sigma_{22} = \sigma_t, \sigma_t = \frac{p^* R^*}{h^*}, \sigma_{11} = \sigma_{22} = \sigma^*. \tag{22}$$

4. APPLICATION

Let us consider the failure of grey cast iron with $\sigma_T = 253MPa, \sigma_C = 624MPa, \tau_T = 168MPa, \sigma_B = 222MPa, \sigma^* = 195MPa, \sigma^{**} = 592MPa$. The results of calculations of the equivalent stress by Eq. (11) and the *Huber-von Mises-Hencky*'s criterion are shown in the Table.

σ_{11}	σ_{22}	σ_{33}	σ_{eq} Eq.(11)	$\sigma_{eq}^{vonMises}$
253	0	0	253	253
187	-100	-100	253.6	287
128	-200	-200	261.2	328
64	-300	-300	263.8	364
-15	-400	-400	251.4	385
-80	-500	-500	253	420
222	111	0	211.9	192.2
176	38	-100	223.1	239
136	-32	-200	240	290.9
82	-109	-300	243.7	330.8
10	-159	-400	230	355.1
-50	-275	-500	227.9	389.7

Table 1: Comparision of the new and the clasical strength criteria

ACKNOLEDGEMENT

The research project on modelling of material behaviour including second-order-effects was promoted by a stay of the second author (A.Z.) at Magdeburg with the financial support of the *Alexander von Humboldt*-Stiftung.

REFERENCES

1. Met. Trans., 12A(1981), 1321 - 1331
2. Trans. ASME. J. Pressure Vessel Technol., 100(1978), 271 - 276
3. Rabotnov, Y.N., Creep problems in structural members, Amsterdam, 1969

The Mathematical Model of Shapeformation Processes of Metalpolymer Composite Shells

Lvov G., Odintsova E.

(Kharkov Polytechnical Institute)

Polytechnical Institute, Frunze 21, 310002 Kharkov, Ukraine

Abstract. The paper is devoted to the problem of creation of mathematical model of shapeformation of composit shell. The material of shell is multilayered composite. It consists of metal and plastic layeres .The solution was built on the base of theory of effective moduls. The problem was solved for shadow shells and shells of rotation.

Recently the utilization of laminated metalpolymer composite material for making of shell elements of machinebuilding constructions is getting broader because these materials have many advantages over metals,such as :their weight is less than weight of metals about 15—20 %, they have heightened viscosity of destruction and vibration strength.That is why the interest to study of properties of these materials and creation of methods of calculation of composite construction increases. Durind calculation of behavior of composite material which consists of few components under mechanical loading many difficulties arise. In order to overcome these difficulties in practice real material is substituted for model, analysis of stress—deformation state of it makes possible to obtain exact notion of behavior of composite. The elastic—plastic deformation of intermediate products of simple forms (cylindrical shells, plane sheets and etc.) is effective method of the obtaining the products from composite materials. The investigation of these processes requires of definition of a special problems of shapeformation. For shells which were made from homogeneous materials these problems were definited in [1].

1. In this paper the inverse problem of shapeformation of composite shell which on process of deformation acquires a predetermined form is solved. It is necesssry to determine external influence which will make this shapeformation and components of stress— deformation state of shell. In order to build the model of composite the hypothesises of continuity and homogeneity were introduced. Thus during calculation composite is changed of homogeneous surroundings mechanical behaviour of this surroundings completely models the behavior of real material. The theory of effective moduls is used. The effective moduls are inverage values of stiffness and strength,they account properties of all phases of heterogeneous surroundings and their interaction.

The process of the definition of effective moduls of laminated composite consists of two stages. During the first stage the problem of calculation of effective moduls of plastic layers is solved. The material of plastic is fibres reinforced composite. The physical constants of the plastic are definitited with account properties of fibres and matrix and their volume content.

In order to solve this problem the structural model of composite is introduced,as a rule this is polydispersed or threephase model [2,3]. At the second stage the effective moduls of the packet on the whole are calculated. For the purpose of definited the stress–deformation state of shell we need to know the elastic constants in different directions :$E_{11}, E_{22}, \nu_{12}, \nu_{21}, \mu_{12}$.

These values are expressed through characteristics of layers. Let shell consists of n layers. The coordinate axises α_1, α_2 are situated in the plane of layers, axis α_3 is perpendicular to it. In order to obtain the expression of the moduls of elasticity of composite we considered the tension in α_1. and α_2 directions in turns.

$$E_{ii}=\sum_{m=1}^{n} E_{ii}^{m}\cdot V^{m}, \qquad (i=1,2) \tag{1.1}$$

where $V^m = h^m/H$ – volume portion of m – layer ;h^m, H – thicknesses of m – layer and packet on the whole.

Consider stress state $\sigma_{11} \neq 0,\ \sigma_{22} = 0,\ \varepsilon_{11}^{m} = \varepsilon_{11}, \varepsilon_{22}^{m} = \varepsilon_{22}$ to definite Poisson 's coefficients ν_{12}, ν_{21}

$$\nu_{12} = \frac{a_1 - 1}{b_1}, \nu_{21} = \frac{b_1}{a_1}, \tag{1.2}$$

where

$$a_1 = \frac{1}{E_{11}\cdot H}\cdot\sum_{m=1}^{n}\frac{E_{11}^{m}\cdot h^{m}}{1-\nu_{12}^{m}\cdot\nu_{21}^{m}}, \qquad b_1 = \frac{1}{E_{11}\cdot H}\cdot\sum_{m=1}^{n}\frac{E_{11}^{m}\cdot h^{m}}{1-\nu_{12}^{m}\nu_{21}^{m}}\cdot\nu_{21}^{m}.$$

Consider stress sytate of pure shear to obtain the expression of shear modul μ_{12}

$$\mu_{12} = \frac{1}{H}\cdot\sum_{m=1}^{n}\mu_{12}^{m}\cdot h^{m} \tag{1.3}$$

(1.3)

Like this, we obtained the expressions of effective moduls of laminated composite through elastic constants of separate layers.

2. During study of elastic–plastic properties of composite we used the theory of plasticity of orthotropic material with isotropic strengthening. According to this theory the condition of the beginning of plasticity has view

$$H_0\cdot(\sigma_{11}-\sigma_{22})^2+F_0\cdot(\sigma_{22}-\sigma_{33})^2+G_0\cdot(\sigma_{33}-\sigma_{11})^2+ \\ +2\cdot N_0\cdot\tau_{12}+2\cdot L_0\cdot\tau_{23}+2\cdot M_0\cdot\tau_{13}-1=0, \qquad (2.1)$$

where H_0,F_0,G_0,N_0,L_0,M_0 – initial values of anisotropic parameters.

It is necessary to definite effective values of anisotropic parameters of composite on the whole to calculate the stress state of shell under the conditions of elastic–plastic deformations.

At the first on the base of known diagrams of tension of plastic and metal which were obtained under conditions of tension of examples in axis α_1 and axis α_2 directions, the diagram of tension of composite was constructed and than it was approximated. Consider the efforts which arise in the example after one–axial tension in axis α_1 and axis α_2 directions and pure shear to obtain the values of plastic parameters of composite through constants of layers.

For the solving of elastic–plastic problem the method of additional deformations was used, it allows to reduction elastic–plastic problem to elastic with additional deformations.

3. Resolving system of equations of inverse problem of shapeformation of shell of rotate which was made from cylindrical blank is constructed. In initial state the middle surface of shell is represented with parametrical equation with help of Lagrange coordinate $\vec{R}_0=\vec{R}_0(\alpha_1,\alpha_2)$. At the end of the deformation process the form of shell must satisfy the equation $f(\vec{R})=0$. Geometrical condition of realization of this shapeformation reqires that the vector of displacements $\vec{U}(\alpha_1,\alpha_2)$ of points of middle surface satisfy an equation $f(\vec{R}_0+\vec{U})=0$.

This equation is added with complete system of equations of physical and geometrically nonlinear theory of shells. The forces actors were expressed through displacements and than one of the equations of equilibrium was rearranged an such form

$$a_1(\alpha)u_{,11}+a_2(\alpha)u\ ,1+a_3(\alpha)u+a_4(\alpha)u_{,1}u+ \\ a_5(\alpha)u^2+a_6(\alpha)u^3+L^w(\alpha)-L^p(\alpha)=0 \qquad (3.1)$$

So the resolving equation was obtained, it is the differential equation of second power with varying coefficients.The method of shooting (the modification of Runge–Kutta method for boundary problem)was used to solve this equation. The axial displacements $U(\alpha_1,\alpha_2)$ were obtained. The second equation is solved to obtain the pressure which made this shapeformation.

Literature.

[1]. Бурлаков А.В. ,Львов Г.И. Об одном классе обратных задач урпугопластического формоизменения оболочек. Изв. АН СССР. МТТ. 1980, 5,с.116 – 123.

[2]. Кристенсен Р. Введение в механику композитов. М.,Мир,1982,336.

[3]. Сендецки Дж. Механика композиционных материалов.М., Мир,1978.

[4]. Кармишин А.В.,Лясковец В.А., Мяченков В.И., Фролов А.М. Статика и динамика тонкостенных оболочечных конструкций. М.,Машиностроение,1975.

[5]. Хилл Р. Математическая теория пластичности. М.,ГИТТЛ,1956.

TURBULENT SCALE CHANGE DEPTH BENEATH FREE SURFACE

Winston KHAN
University of Puerto Rico
Mayaguez, Puerto Rico

Abstract. On the approach of turbulent vortices or eddies to an interface or free surface under varying interfacial conditions, it is believed and evidenced by various observers and experiments based on agitated systems, stirred or otherwise, that the scale of turbulence changes at some depth beneath the interface. In this work, we derive an expression for this depth as a function of the Reynolds number and the surface compressional modulus of elasticity which arises due to surfactants.

1. INTRODUCTION

Surface or interfacial phenomena have gained paramount importance in industry and, in particular, the shipping industry, which lends itself to naval interest spanning the last decade. It is in this light, that vortex, jet and eddy interaction with clean and contaminated surfaces are being investigated. The scale change depth is germane to Navy considerations and other industrial enterprises involving these physical situations. The author believes that the interaction of vortex dynamics with interfacial phenomena plays a major and significant role in this change of character corresponding to a characteristic depth beneath the free surface. In most of the past and present literature embodying the various ideas and evidence, the following are prominent.

(a). It appears that the boundary conditions at a free surface must influence this phenomenon either through the dynamics of the approaching eddies which creates an interaction with the interfacial boundary conditions to restrict its entry by a shear stress analysis, or

(b). Through a redistribution of the energy, which in turn translates into a restriction in the eddy turbulence scale.

The above phenomena must occur at a certain depth beneath the free surface, and it is the purpose of this work to determine this depth in terms of turbulent vortices and interfacial properties.

2. METHOD

Surface or interfacial phenomena has gained paramount importance in industry where jets and eddy interactions with clean and contaminated surfaces are being investigated. This particular problem contemplates a theoretical investigation of eddy or turbulent vortex interaction with surface phenomena.

In this light, it would be necessary to dwell for a few moments on the properties of surfactants. It is well known that variations in area in general involve variations in surface tension, such that $\delta\gamma/\gamma_o$ = N dA/A, where A is the area available per molecule and gives rise to the suraface compressional modulus of elasticity, C_s^{-1}, but defined differently as -A $d\pi/dA$. Since an approaching eddy would create dynamic disturbances close to the interface, such as dynamic pressures and stresses, its presence near the free surface would cause interfacial conditions to vary and become active due to dynamic disturbances through its elastic effect.

As the free surface is approached, fluctuations normal to the surface are diminished in comparison to those parallel to the surface, which are accentuated, resulting in an interfacial "turn around" of an approaching eddy. At the interface then the scale of the motion is two-dimensional and parallel to the surface, resulting in a scale change. However, the depth at which this scale change commences would correspond to the depth at which the dynamic properties influence interfacial conditions, which then retaliates through its elastic effect to negate the dynamic pressures exerted by an approaching eddy. We assume that this depth at which the scale change occurs is comparable to the average size of an eddy migrating into the interface from below.

Experimental observation reveal that the deformation created in the normal direction is small in comparison to the horizontal motion of a migrating eddy into the interface. We may interpret this to imply that the impact of an eddy of fluid on the free surface causes the fluid to be redirected parallel to the surface and attributed to a redistribution of energy. Hence we are seeking an energy redistribution solution, hence a clue to the model involved, that is, an energy approach.

The model is that of a body of fluid in the form of an eddy or turbulent vortex which impinges on the interface under different interfacial conditions: (i) clean, and
(ii) contaminated with insoluble film.

If λ is the size of an average eddy, v is the velocity, γ the surface tension, and C_s^{-1} the surface compressional modulus of elasticity, then energy considerations show that

$$\frac{1}{2}\rho V^2\lambda^3 = (\lambda + C_s^{-1})\lambda^2$$

Hence, $\lambda = (\gamma + C_s^{-1})/v^2 = f(Re).g(C_s^{-1}) = (\gamma + C_s^{-1})/Re^2$.

We interpret the average size of an approaching eddy to be comparable to the depth at which the dynamics of an eddy interacts with the surface conditions.

3. CONCLUSIONS

The depth λ characterizes a zone of damped turbulence near the free surface Eddies outside this depth are completely unaffected by the conditions of the free surface.

The capillary forces must oppose the dynamic thrust which in turn is affected by the contamination through C_s^{-1} and incorporated through energy considerations.

4. FIGURES

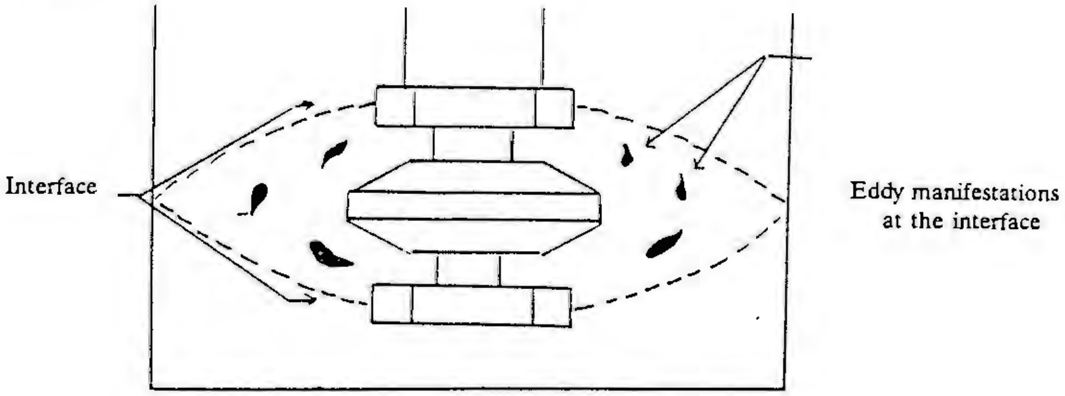


Fig. 1. Physical model.

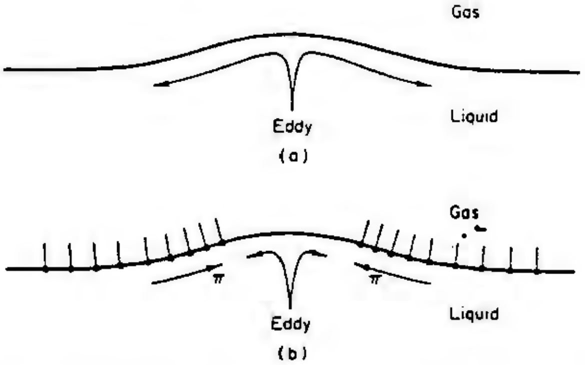


Fig. 2. Approach of eddy to (a) clean surface and to (b) surface with film of surface active agent present.

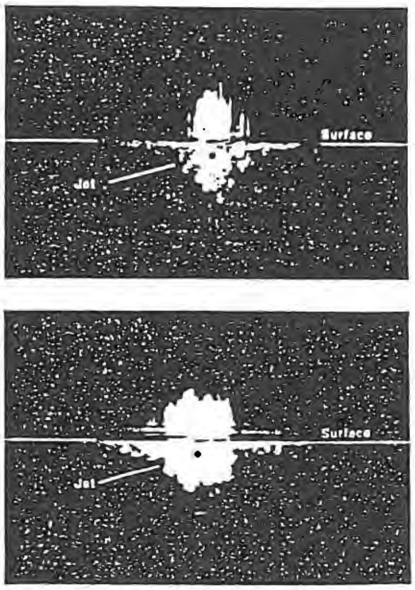


Fig. 3. Laser-induced fluorescence (LIF) photograph of a jet discharging beneath a clean free surface.

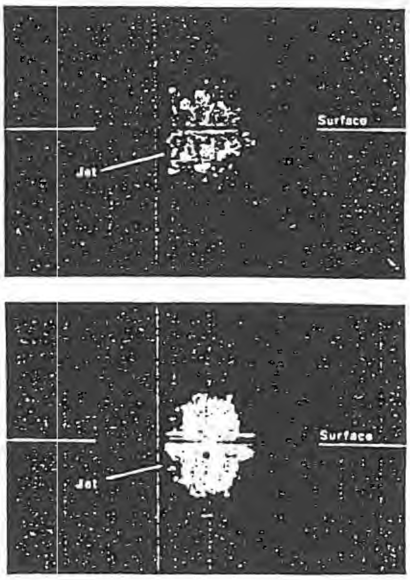


Fig. 4. LIF photograph of a jet discharging beneath a contaminated free surface

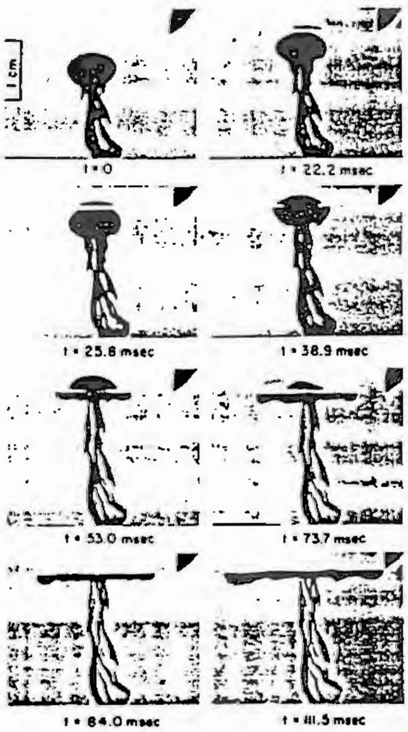


Fig. 5. The approach of a single pulse of water containing permanganate to a free surface

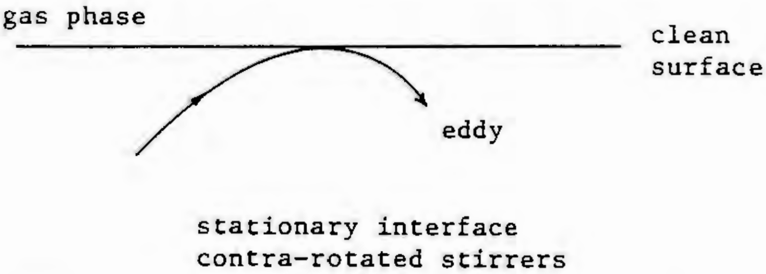


Fig. 6. Surface renewal phenomenon; eddy brings fresh liquid into the interface

5. ACKNOWLEDGEMENTS

The author wishes to gratefully acknowledge the research support provided by the Office of Naval Research, Washington, D.C., U.S.A.

6. REFERENCES

[1] Anthony, D.G., On the Interaction of a Submerged Turbulent Jet With a Clean or Contaminated Free Surface. Physics of Fluids A, Vol. 3, No. 2, February 1991.

[2] Walker, D.J., Anthony, D.G., Chen, C.Y. and Willmarth, W-W, Interaction of a Turbulent Jet with a Free Surface, Office of Naval Research Workshop, March 1992.

[3] Davies, J.T.,Turbulence Phenomena. Academic Press, 1972.

[4] Khan, W., Jet Interaction with a Free Surface, In Turbulence Phenomena (J.T. Davies), Academic Press, 1992.

[5] Levich, J.G. Physicochemical Hydrodynamics. Prentice Hall, 1962.

Variable Surface Tension Effects on Drift Velocity of Water Waves

Kewal K. Puri
University of Maine
Orono, Maine 04469-5752
U.S.A.

Abstract. In this paper, we consider the effect of the presence of a monomolecular film on the free surface of an infinite horizontal extant. The asymptotic expressions for the Eulerian velocity field and the mass transport velocity are obtained for the motion induced by a surface wave in the regions of the free surface, the bottom boundary layers as also in the interior region outside of the two boundary layers. The bottom layer is assumed to be fully turbulent and the model developed by Trowbridge and Madsen (1984) is used. Also the wave damping coefficient is calculated.

It is shown that for progressive waves the wave damping is $0(\sqrt{v})$ where v is the kinematic viscosity. This term vanishes with the vanishing film parameters. Also, it is established that the drift velocity just outside the bottom boundary layer changes direction for small values of (the wave number × the water depth). The results for the drift in the presence and in the absence of the surface film are compared.

Reference:

Trowbridge, John and Madsen, Ole. Turbulent wave boundary layer I & II. J. Geophysical Research 89, CS (1984).

A Multivariable Controller Reduction Method via Frequency Response Approximation

Ralf Müller and Sebastian Engell
Lehrstuhl für Anlagensteuerungstechnik
Fachbereich Chemietechnik
Universität Dortmund
D-44221 Dortmund, Germany
e-mail: ast@astaire.chemietechnik.uni-dortmund.de

Abstract We present a novel technique for reducing linear multivariable controllers by approximating the frequency response of the high-order controller. For a 3×3 controller of 9th order the method is compared to frequency-weighted balancing and truncating as proposed by Enns where we use the output weighting and a two-sided weighting which is the same as applied in our frequency response approximation.

1. Introduction

The problem of controller reduction is distinct from that of plant model reduction because the closed-loop behaviour rather than the open-loop characteristics should be preserved. To achieve this, any controller reduction procedure ought to take the plant model into account what leads to frequency-weighted approximation techniques. The weightings should be known a priori so they can be calculated automatically from the frequency responses of the nominal controller and the plant.

We present a method for approximating a multivariable controller frequency response matrix by a transfer matrix of specified structure and orders. While in [1] we discussed the controller design strategy, in this paper we emphasize the order reduction method, assuming a high-order controller was obtained by whatever method seemed appropriate.

In the third section, we summarize the concept of frequency-weighted balancing and truncating introduced by Enns. In addition to his input weighting resp. output weighting we introduce a two-sided weighting which is similar to that used in our approximation technique.

The application of the frequency response approximation method and the method of Enns demonstrates that a multivariable controller of 9th order can be reduced to 4th order using both approaches if similar weightings are applied.

2. Controller reduction by frequency response approximation

We consider a unity feedback system with r reference inputs and r regulated outputs. The plant $\underline{P}$ and the controller $\underline{C}$ are r×p resp. p×r transfer matrices, with p≥r, and the reference-to-output frequency response is

$$\underline{T}(j\omega) = \left(\underline{I} + \underline{P}(j\omega)\underline{C}(j\omega)\right)^{-1}\underline{P}(j\omega)\underline{C}(j\omega) \quad . \tag{2.1}$$

To calculate the sensitivity of the closed-loop frequency response to deviations of the compensator frequency response from its nominal value $\underline{C}_0(j\omega)$, we write the approximated controller as

$$\underline{C}(j\omega) = \underline{C}_0(j\omega) + \Delta\underline{C}(j\omega) \quad . \tag{2.2}$$

The aim of the approximation is to minimize the deviation of the closed-loop system frequency response matrix from its nominal value which depends nonlinearly on $\Delta\underline{C}(j\omega)$

$$\Delta\underline{T}(j\omega) = \left(\underline{I} + \underline{P}(j\omega)\left(\underline{C}_0(j\omega) + \Delta\underline{C}(j\omega)\right)\right)^{-1}\underline{P}(j\omega)\left(\underline{C}_0(j\omega) + \Delta\underline{C}(j\omega)\right) - \underline{T}_0(j\omega) \quad . \tag{2.3}$$

Using the nominal resp. actual sensitivity functions

$$\underline{S}_0(j\omega) = \left(\underline{I} + \underline{P}(j\omega)\underline{C}_0(j\omega)\right)^{-1} \text{ and } \quad \underline{S}(j\omega) = \left(\underline{I} + \underline{P}(j\omega)\underline{C}(j\omega)\right)^{-1} \quad , \tag{2.4}$$

$\Delta\underline{T}(j\omega)$ can be brought to the simple form

$$\Delta\underline{T}(j\omega) = \underline{S}_0(j\omega)\underline{P}(j\omega)\Delta\underline{C}(j\omega)\underline{S}(j\omega) \quad . \tag{2.5}$$

For a good approximation, $\underline{S}(j\omega)$ can be replaced by $\underline{S}_0(j\omega)$ what yields a linear relation of $\Delta\underline{T}(j\omega)$ and $\Delta\underline{C}(j\omega)$

$$\Delta\underline{T}(j\omega) \approx \underline{S}_0(j\omega)\underline{P}(j\omega)\Delta\underline{C}(j\omega)\underline{S}_0(j\omega) \quad . \tag{2.6}$$

If the desired closed-loop system is decoupled, $\underline{S}_0(j\omega)$ is diagonal and each element of $\Delta\underline{C}(j\omega)$ affects only the corresponding column of $\Delta\underline{T}(j\omega)$

$$\Delta t_{kj}(j\omega) \approx s_{0k}(j\omega)s_{0j}(j\omega)\sum_{i=1}^{p}\left(p_{ki}(j\omega)\Delta c_{ij}(j\omega)\right) \quad . \tag{2.7}$$

Thus the controller columns can be approximated independently. To solve the nonlinear problem of optimizing numerator and denominator polynomials simultaneously, the iterative method of Sanathanan and Koerner[2] is used prescribing a common denominator in each controller column (for more details see [1,3]). As the orders of the numerator polynomials can be different and each coefficient can be set to zero, arbitrary structures as e.g. blockdiagonal controllers can be optimized.

If the nominal controller frequency response is approximated by a relatively simple transfer matrix the achieved sensitivity function differs significantly from $\underline{S}_0(j\omega)$. Most detrimental is the fact that the off-diagonal elements do not vanish and thus the error in one controller column influences not only the same column of $\underline{T}(j\omega)$. Hence the solution of the column-by-column optimization is far from optimal and can be improved significantly by a nonlinear minimization of the overall quadratic error

$$J = \sum_{k=1}^{r}\sum_{i=1}^{r}\sum_{\ell=1}^{N}\left|\frac{\Delta t_{ik}(j\omega_\ell)}{\omega_\ell}\right|^2 \quad , \tag{2.8}$$

where N is the number of frequency points. $\Delta\underline{T}$ is divided by ω to approximate the step response rather than the impulse response. We use Powell's unconstrained gradient-free minimization algorithm [4], starting from the solution of the independent optimization of the controller columns. Attempts to solve (2.8) directly by global optimization require enormous computation times due to the huge number of local minima and the results were not as good as from the two-step approach. Furthermore the column-by-column approximation is an effective and computationally cheap indicator if the overall minimization is feasible with the chosen orders.

In order to avoid unstable approximations of stable controllers, the denominators $d_k(s)$ of order n_k are parametrized as

$$d_k(s) = \begin{cases} \displaystyle\prod_{i=1}^{n_k/2}\left((s/\omega_i)^2 + 2\zeta_i/\omega_i + 1\right) & n_k \text{ even}, \\ \displaystyle(s/\omega_0 + 1)\prod_{i=1}^{(n_k-1)/2}\left((s/\omega_i)^2 + 2\zeta_i/\omega_i + 1\right) & n_k \text{ odd}. \end{cases} \qquad \omega_i, \zeta_i > 0 \quad . \tag{2.9}$$

Stability of the closed-loop system is not ensured in general but depends on the approximation error.

3. Balanced Order Reduction

3.1 The Balancing Technique

Let us consider an n^{th} order, linear time-invariant system $\underline{G}(s)$ with a minimal realization

$$\underline{G}(s) = \underline{C}(s\underline{I} - \underline{A})^{-1}\underline{B} + \underline{D} \quad . \tag{3.1}$$

If the system is asymptotically stable, the controllability Gramian $\underline{P}$ and the observability Gramian $\underline{Q}$ exist and are given by

$$\underline{P} = \int_0^\infty e^{\underline{A}t}\underline{B}\underline{B}^T e^{\underline{A}^T t}dt \,, \quad \underline{Q} = \int_0^\infty e^{\underline{A}^T t}\underline{C}^T\underline{C}e^{\underline{A}t}dt \quad . \tag{3.2}$$

The Gramians can be calculated from the Lyapunov equations

$$\underline{A}\underline{P} + \underline{P}\underline{A}^T + \underline{B}\underline{B}^T = 0 \,, \quad \underline{A}^T\underline{Q} + \underline{Q}\underline{A} + \underline{C}^T\underline{C} = 0 \quad . \tag{3.3}$$

A realization $(\underline{A},\underline{B},\underline{C},\underline{D})$ of $\underline{G}(s)$ is said to be (internally) balanced if

$$\underline{P} = \underline{Q} = \underline{\Sigma} = \mathrm{diag}\{\sigma_i \ldots \sigma_n\} \quad \text{with} \quad \sigma_i = \sqrt{\lambda_i\{\underline{P}\underline{Q}\}} \quad . \tag{3.4}$$

It is assumed that the state variables bave been permuted so that $\sigma_i \geq \sigma_{i+1}, i = 1,\ldots,n-1$ where σ_i is the i^{th} Hankel singular value (HSV) that measures the input-output importance of the balanced state x_i. The eigenvalues of $\underline{P}$ and $\underline{Q}$ change under equivalence transformations but the HSVs are invariants of the input-output behaviour of the system.

Let the balanced realization be partitioned as

$$\begin{pmatrix}\dot{\underline{x}}_1\\ \dot{\underline{x}}_2\end{pmatrix}=\begin{pmatrix}\underline{A}_{11} & \underline{A}_{12}\\ \underline{A}_{21} & \underline{A}_{22}\end{pmatrix}\begin{pmatrix}\underline{x}_1\\ \underline{x}_2\end{pmatrix}+\begin{pmatrix}\underline{B}_1\\ \underline{B}_2\end{pmatrix}\underline{u},\quad \underline{y}=\begin{pmatrix}\underline{C}_1 & \underline{C}_2\end{pmatrix}\begin{pmatrix}\underline{x}_1\\ \underline{x}_2\end{pmatrix},\quad \underline{\Sigma}=\begin{pmatrix}\underline{\Sigma}_1 & \underline{0}\\ \underline{0} & \underline{\Sigma}_2\end{pmatrix},\tag{3.5}$$

where $\underline{A}_{11}$ and Σ_1 are r×r matrices. Moore [5] suggested that the subsystem ($\underline{A}_{11}$,$\underline{B}_1$,$\underline{C}_1$) should be a good approximation of the balanced system if $\sigma_r >> \sigma_{r+1}$. This direct truncation (DT) has the properties:

- The subsystems ($\underline{A}_{ii}$,$\underline{B}_i$,$\underline{C}_i$), i =1,2 are internally balanced with Gramians Σ_i [6].
- The subsystems ($\underline{A}_{ii}$,$\underline{B}_i$,$\underline{C}_i$), i =1,2 are asymptotically stable if $\sigma_r > \sigma_{r+1}$ [6].
- There is an upper bound for the approximation error (see e.g. [7]) :

$$\left\|\underline{C}(j\omega\underline{I}-\underline{A})^{-1}\underline{B}-\underline{C}_1(j\omega\underline{I}-\underline{A}_1)^{-1}\underline{B}_1\right\|_\infty \le 2\sum_{i=r+1}^{n}\sigma_i \quad .\tag{3.6}$$

The reduction error of the direct truncation method generally tends to vanish at high frequencies. There is a DC-gain mismatch between the high-order and the reduced-order model. If the order is reduced by 1, the error achieves the upper bound $2\sigma_n$ at ω=0 [7].

In controller reduction one would like to retain as much as possible of the low and medium frequency properties of the high-order transfer matrix. This can be achieved by applying the singular perturbation technique to the balanced realization instead of directly truncating it. The singular perturbation approximation (SPA) can be envisioned as making the eliminated states infinitely fast. Setting $\dot{\underline{x}}_2$ = 0 yields:

$$\begin{aligned}\underline{A}_r &= \underline{A}_{11}-\underline{A}_{12}\underline{A}_{22}^{-1}\underline{A}_{21} & \underline{C}_r &= \underline{C}_1-\underline{C}_2\underline{A}_{22}^{-1}\underline{A}_{21}\\ \underline{B}_r &= \underline{B}_1-\underline{A}_{12}\underline{A}_{22}^{-1}\underline{B}_2 & \underline{D}_r &= \underline{D}-\underline{C}_2\underline{A}_{22}^{-1}\underline{B}_2 \quad .\end{aligned}\tag{3.7}$$

In [8] it was shown that the error bound (3.6) for the direct truncation method is also valid for the SPA. But opposite to the direct truncation there is no DC-gain mismatch and the error is shifted to higher frequencies.

3.2 Frequency-Weighted Balancing

Enns introduced frequency weighting into the balanced controller reduction to preserve properties of the closed-loop system instead of doing open-loop reduction. The basic idea is to change the controllability or observability Gramian to reflect the influence of the plant.

From stability robustness in the presence of plant uncertainties, the following conditions for closed-loop stability (assuming that the nominal controller stabilizes the plant) can be derived [7]. The closed-loop system with the reduced-order controller will remain stable if

$$\left\|\underline{W}_o(j\omega)\Delta\underline{C}(j\omega)\underline{W}_i(j\omega)\right\|_\infty < 1 \quad .\tag{3.8}$$

with the weightings

$$\begin{aligned}&\underline{W}_o(j\omega)=\underline{S}_o(j\omega)\underline{P}(j\omega)\,,\quad \underline{W}_i(j\omega)=\underline{I} \quad \text{"output weighting" or}\\ &\underline{W}_o(j\omega)=\underline{I}\,,\quad \underline{W}_i(j\omega)=\underline{S}_o(j\omega)\underline{P}(j\omega) \quad \text{"input weighting"}\end{aligned}\tag{3.9}$$

depending whether the "uncertainty" due to the controller reduction is represented at the input or the output of the controller. The weightings have the poles of the closed-loop system and hence are stable. Note that only in the scalar case the two weightings will give the same weighted controller.

To calculate the observability Gramian in the output weighted case, $\underline{A}$ and $\underline{C}$ in (3.3) are replaced by the corresponding matrices of the controller cascaded with $\underline{W}_o(s)$. For input weighting, the controllability Gramian is determined with $\underline{A}$ and $\underline{B}$ taken from the series connection of $\underline{W}_i(s)$ and $\underline{C}(s)$. Balancing the controller with this modified Gramians reflects the importance of the states to the closed-loop system; the controller can be thought of as frequency-weighted.

It is possible to use both weightings in (3.8) simultaneously. If the closed-loop approximation error is to be minimized, the weightings

$$\underline{W}_o(s)=\underline{S}_0(s)\underline{P}(s)\,,\quad \underline{W}_i(s)=\underline{S}_0(s) \quad .\tag{3.10}$$

result from (2.6). Compared to the output weighted case which uses the same $\underline{W}_o(s)$, the weighting on low frequencies where the loop gains are high is reduced. This two-sided weighting will be referred to as performance weighting in the sequel.

4. Example

To compare the two methods, we reduce a 3×3 controller of 9th order for the vertical dynamics of an aircraft which has been considered in several publications. The plant model and the nominal controller are taken from [9] and are given in the appendix.

Using frequency response approximation, the controller can be reduced to order 5 without significant differences in the step responses. Reduction to order 4 with orders 1/1/2 in columns 1-3 respectively and one integral term in each column yields

$$\underline{C}_1(s)=\begin{pmatrix} \dfrac{-9.79s+1.47}{s} & \dfrac{0.243s+2.63}{s} & \dfrac{-3.86s^2+17.37s+24.61}{0.0003s^2+s} \\ \dfrac{-1.33s+0.665}{s} & \dfrac{7.76s+0.9}{s} & \dfrac{-0.32s^2+2.9s+7.27}{0.0003s^2+s} \\ \dfrac{-17.1s+2.2}{s} & \dfrac{-0.556s+5.95}{s} & \dfrac{-20.8s^2+0.375s+33.4}{0.0003s^2+s} \end{pmatrix}.$$

Approximation of a multivariable PI-controller yields a stabilizing controller with very poor performance. Up to order 6 no blockdiagonal controller with good performance could be obtained. Figure 1 compares $\underline{C}_I(s)$ with the nominal controller.

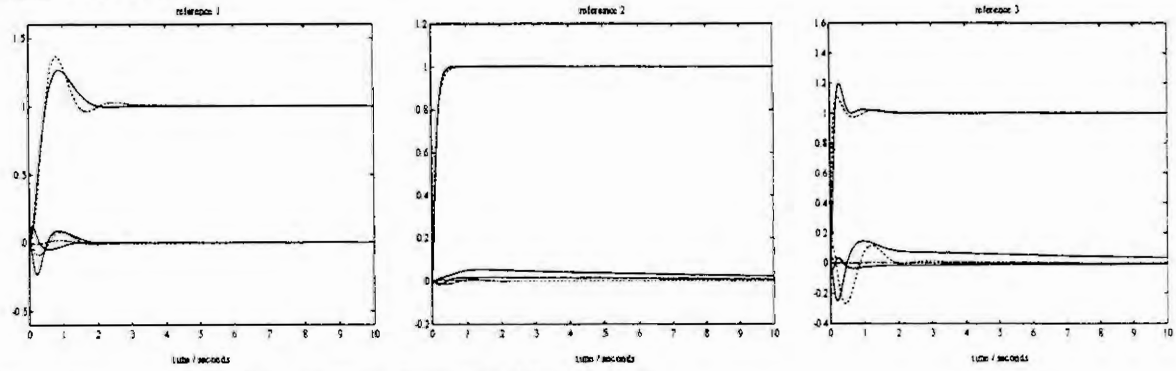


Figure 1: Step responses with the nominal controller (solid) and $\underline{C}_1(s)$ (dashed)

To balance the nominal controller the poles in the origin must be extracted. In modal canonical form (m.c.f.) the original controller is partitioned and after balancing and reduction the stable part is (again in m.c.f.) put together with the unstable part.

From the unweighted HSVs of the stable part of the nominal controller σ_i=26.6/24.1/21/19.1/3.6/0.01 it can be seen that the reduction to order 4(+3 integrators) is not difficult. But already the reduction to order 3(+3) gives an intolerable loss in performance especially for steps on reference 3.

The output weighted HSVs of the controller are σ_i=0.926/0.893/0.248/0.131/0.081/0.001. As expected, reduction to order 2(+3) does not cause significant changes in the step responses. The reduction to order 1(+3) gives the following controller (the unstable part can be taken from the high-order controller)

$$\underline{C}_2:\quad A=-11.16,\quad \underline{B}=\begin{bmatrix}3.28 & -0.006 & 3.15\end{bmatrix},\quad \underline{C}=\begin{bmatrix}-11.94\\ -2.71\\ 102.9\end{bmatrix},\quad \underline{D}=\begin{bmatrix}-3.11 & -0.698 & 10.9\\ 0.695 & 7.07 & 0.994\\ -38.86 & -0.0574 & -36.48\end{bmatrix}.$$

Performance weighting gives the HSVs σ_i=0.843/0.799/0.249/0.126/0.080/0.007 which are similar to that of the output weighting. The following controller results from reduction to order 1(+3):

$$\underline{C}_3:\quad A=-13.07,\quad \underline{B}=\begin{bmatrix}2.947 & -0.00631 & 3.666\end{bmatrix},\quad \underline{C}=\begin{bmatrix}-27.97\\ -3.63\\ 113.2\end{bmatrix},\quad \underline{D}=\begin{bmatrix}-0.315 & -0.769 & 15.38\\ 0.716 & 7.07 & 1.247\\ -34.14 & -0.0638 & -39.2\end{bmatrix}.$$

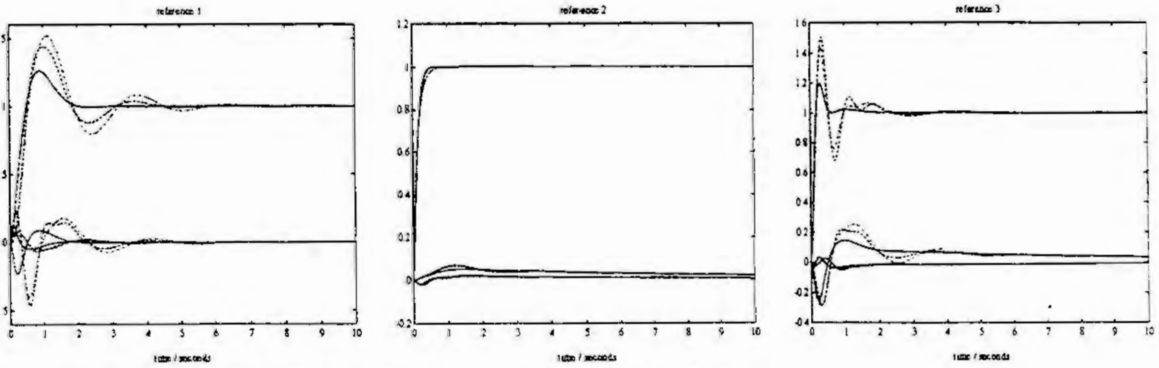


Figure 2: Step responses with the nominal controller (solid), $\underline{C}_2$ (dashed) and $\underline{C}_3$ (dashdot)

The step responses in Figure 2 show no great differences between the controllers obtained by output weighting and the two-sided weigbting. The behaviour of the controller optimized by frequency response approximation is closer to the nominal behaviour. Our opinion is that this is due to the difference between the approximated and the nominal sensitivity function which can only be taken into account in the nonlinear optimization..

5. Conclusions

It was demonstrated that order reduction of a 3×3 controller from order 9 to order 4 using the presented frequency response approximation and the frequency-weighted balancing introduced by Enns give qualitatively similar results. The importance of weightings which are known a priori and can be calculated automatically was shown. Advantages of the frequency response approximation technique are the possiblity to approximate controllers with specified structure and the direct applicability for plants with deadtimes. It may also be conjectured that the quality of the approximation will be better for the same controller structure in most cases because of the minimization of a quadratic cost functional. The truncation of frequency-weighted balanced realizations is computationally cheaper. The weighted Hankel singular values are good indicators for the achievable order reduction.

References

[1] S.Engell, R.Müller, Multivariable Controller Design by Frequency Response Approximation, Proc. of the 2nd European Control Conference, 1993, 1715-1720

[2] C.K.Sanathanan, J.Koerner, Transfer function synthesis as a ratio of two complex polynomials, IEEE TAC-8, 1963, 56-58

[3] S.Engell, Compensator design by frequency-weighted approximation, Proc. IEE International Conference Control, 1988, 253-258

[4] Press W.H., B.P. Flannery, S.A. Teukolsky, W.T. Vetterling, Numerical Recipes in Fortran, Cambridge University Press, 1988

[5] Moore, B.C., Principal component analysis in linear systems: Controllability, observability and model reduction, IEEE TAC-26, 1982, 17-32

[6] Pernebo, L. and L.M. Silverman, Model reduction via balanced state space representations, IEEE TAC-27, 1982, 382-387

[7] Enns, D.F., Model reduction for control systems design, Ph.D. Thesis, Stanford University, USA, 1984

[8] Liu, Y. and B.D.O. Anderson, Singular perturbation approximation of balanced systems, Proc. of the 28th CDC, Tampa, Florida, 1989, 1355-1360

[9] McFarlane, D.C. and K. Glover, Robust controller design using normalized coprime factor plant descriptions, Springer, 1990

Appendix

The state-space description of the plant is :

$$\underline{A} = \begin{bmatrix} 0 & 0 & 1.1320 & 0 & -1 \\ 0 & -0.0538 & -0.1712 & 0 & 0.0705 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0.0485 & 0 & -0.8556 & -1.0130 \\ 0 & -0.2909 & 0 & 1.0532 & -0.6859 \end{bmatrix}, \quad \underline{B} = \begin{bmatrix} 0 & 0 & 0 \\ -0.1200 & 1 & 0 \\ 0 & 0 & 0 \\ 4.4190 & 0 & -1.6650 \\ 1.5750 & 0 & -0.0732 \end{bmatrix}, \quad \underline{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}, \quad \underline{D} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} .$$

The nominal controller (transformed to modal canonical form) is taken from [9] :

$$\underline{A} = \text{diag}\left\{ -213.3496 \quad -94.1743 \quad -8.2673 \quad \begin{bmatrix} -28.8261 & 0.5725 \\ -0.5725 & -28.8261 \end{bmatrix} \quad -0.1 \quad 0 \quad 0 \quad 0 \right\}$$

$$\underline{B}^T = \begin{bmatrix} -3033.6 & 6755 & -463.76 & -263.59 & -100.15 & -0.0337 & 3.496 & 0 & 0 \\ -1.5378 & 247.21 & 0.0823 & -112.43 & 127.55 & 0.0011 & 0 & 1.5414 & 0 \\ -12076 & -2313.7 & -141.87 & 757.1 & 2065.1 & 0.0165 & 0 & 0 & 3.1804 \end{bmatrix}$$

$$\underline{C} = \begin{bmatrix} -0.6594 & -0.736 & -0.2566 & -0.8279 & -0.4132 & 0.0337 & -0.1823 & -0.021 & 0.2479 \\ -0.0479 & 0.0199 & -0.0232 & -1.2914 & 0.4691 & 0.0017 & -0.0025 & 0.4624 & 0.0076 \\ 0.8309 & -0.4839 & -0.7953 & 0.0388 & 0.1823 & 0.0738 & -0.2295 & 0.0040 & -0.1968 \end{bmatrix}, \quad \underline{D} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} .$$

ON MODAL TECHNIQUES FOR MODEL REDUCTION

A. VARGA

DLR - Oberpfaffenhofen
Institute for Robotics and System Dynamics
P.O.B. 1116, D-82230 Wessling, Germany

Abstract. The general applicability of the modal approach for model reduction is restricted by the lack of guaranteed bounds for approximation errors and of a satisfactory modal dominance analysis procedure. Functional and computational enhancements of this approach are proposed. Functional enhancements arise by combining the modal techniques with other methods and by using improved dominance analysis techniques. The computational enhancements are the results of employing numerically reliable algorithms for both dominance analysis as well as for model reduction.

1. INTRODUCTION

The modal approach to model reduction proposed initially by Davison [1] was later extended with new variants by several authors: Marschall [2], Chidambara [3], Fossard [4], Litz [5] and others. The importance of the modal approach as a useful model reduction technique resides in its applicability to reduce high order systems as those arising for example from modelling of large mechanical structures or of large power systems. The method can handle models with lightly damped modes and even unstable systems. In case of very large order systems, the modal technique is one of the very few applicable methods.

Several limitations of the modal approach raise problems for a general use of this approach. In the first place, the lack of a generally applicable modal dominance analysis method prevents the use of this method in many cases as for example when the original system has multiple poles. The existing methods fail sometimes even to detect exact structural non-minimality, that is, poles which are uncontrollable or unobservable. Another weekness of this approach is the lack of a guaranteed bound for the approximation error which has as consequence the frequent need to experiment on a trial and error basis with different approximations.

In this paper we shortly survey the main existent modal reduction approaches and some of available techniques for dominance analysis. Then we discuss possible enhancements of the modal reduction approach. These enhancements consist in: 1) combining the modal techniques with other approaches; 2) using a new, more powerful method for modal dominance analysis; and 3) using numerical techniques with guaranteed numerical reliability. The proposed new approach is well suited for robust software implementation.

2. MODAL REDUCTION TECHNIQUES

Consider the n-th order original state-space model $G := (A, B, C, D)$ with the $p \times m$ *transfer-function matrix* (TFM) $G(\lambda) = C(\lambda I - A)^{-1}B + D$, and let $G_r := (A_r, B_r, C_r, D_r)$ be an r-th order approximation of the original model ($r < n$), with the TFM $G_r = C_r(\lambda I - A_r)^{-1}B_r + D_r$. The modal approach to model reduction can be interpreted as performing a similarity transformation Z yielding

$$\left[\begin{array}{c|c} Z^{-1}AZ & Z^{-1}B \\ \hline CZ & D \end{array}\right] := \left[\begin{array}{cc|c} A_1 & 0 & B_1 \\ 0 & A_2 & B_2 \\ \hline C_1 & C_2 & D \end{array}\right], \qquad (1)$$

where A_1 and A_2 contains the r *dominant* and respectively, the $n-r$ *non-dominant* eigenvalues (modes) of A, and then defining the reduced model on the basis of this partitioned representation. The above partition of system matrices is equivalent with the additive decomposition $G = G_1 + G_2$, where $G_1 := (A_1, B_1, C_1, D)$ and $G_2 := (A_2, B_2, C_2, 0)$ are the dominant and non-dominant subsystems, respectively.

For our discussion of different modal approaches for model reduction we assume that the original system is already additively decomposed. Furthermore we assume that the system is asymptotically stable. This assumption is only a technical one, because for an unstable system the modal approach can be performed on its stable projection.

We consider three basic approaches:

Method 1. Define $G_r := (A_1, B_1, C_1, D)$. This is basically the modal approximation proposed in [1]. The approximation error $\Delta = G - G_r$ tends to zero at high frequencies. However, the DC-gains mismatch of the original and reduced models could be large.

Method 2. Define $G_r := (A_1, B_1, C_1, D + G_2(\gamma))$, where $\gamma = 0$ for a continuous-time system and $\gamma = 1$ for a discrete-time system. Note that $G(\gamma) = G_r(\gamma)$, and thus this approximation preserves the DC-gain of the original system, but the approximation error at high frequencies could be large. The methods of [2, 3, 4], which compensate the steady-state errors can be viewed as particular cases of Method 2.

Method 3. Define $G_r := (A_1, B_1, C_1 + C_2E, D)$, where E is to be determined such that $G(\gamma) = G_r(\gamma)$. This approximation automatically ensures small errors at high frequencies. From the equality of DC-gains follows that E should satisfy

$$C_2(\gamma I - A_2)^{-1}B_2 = C_2E(\gamma I - A_1)^{-1}B_1.$$

This is a system of pm linear equations with $(n - r)r$ unknowns and a solution (with possibly minimal norm) generically exists provided $pm \leq (n - r)r$, a condition fulfilled in most applications. The method of [5] results if we impose the stronger condition

$$(\gamma I - A_2)^{-1}B_2 = E(\gamma I - A_1)^{-1}B_1,$$

which usually leads to an E with higher norm. The generical solvability condition in this case is $m \leq r$, which in most applications is also fulfilled. The additional freedom arising from the non-unicity of E can be used to optimally tune the free parameters of E to minimize for instance the output error norm.

One difficulty in using the modal approach is the lack of and *a priori* computable bound for the resulting approximation error $\Delta = G - G_r$. The actual error can be computed only after that a choice has been made, and thus the model reduction can be done only on a *trial and error* basis. In contrast, methods based on balancing, as for example the *balance & truncate* (B&T) method [6, 7], provide *a priori* information (the Hankel-singular values) which can be used to select the appropriate order for an acceptable approximation error.

It is possible to combine the modal approach with other techniques. For example, if the system is already decomposed as in (1), then the reduction can be performed separately on G_1 and G_2. Let $G_r = G_{1r} + G_{2r}$ be the resulting reduced model, where G_{1r} and G_{2r} are the resulting reduced subsystems computed say with the B&T method. If for the separate reduction of terms we have that $\|G_i - G_{ir}\| \leq \varepsilon_i$ for $i = 1, 2$, then $\|G - G_r\| \leq \varepsilon_1 + \varepsilon_2$. Thus, by reducing individually the terms, we can also control the resulting global error by choosing appropriate orders for the reduced subsystems. The technique can be readily extended to additive decompositions with more than two terms (see the next section) and many variations of it are possible by employing alternative model reduction methods.

The real advantage of such combinations is more evident when we have to reduce very large order models, as those which typically result from finite-element analysis of large mechanical structures. Because the large orders of such models, the modal approach is frequently the only method which can be used for order reduction. This reduction is often only a preliminary reduction which makes tractable further reductions with the help of more powerful methods.

3. MODAL DOMINANCE ANALYSIS

The main limitation of the modal approach to model reduction is the lack of a reliable, general purpose method for modal dominance analysis. The existence of such a method is highly questionable because for any of existing methods counterexamples can be easily constructed showing their failures in producing useful dominance information. An counterexample to the method of Litz [5] is given in [8], where a 12-th order system with distinct and equally dominant poles is presented for which a good 4-th order approximation can be computed. We can see this as a basic limitation of the modal approach which can permanently occur, because often the identified dominant parts have still too large orders and thus further reductions should have recourse to alternative techniques.

In this section we discuss the limitations of existing dominance analysis techniques and we propose an alternative approach to overcome them. The new technique allows an easy handling of systems with multiple poles or of systems which are exactly or nearby non-minimal.

Consider the system $G = (A, B, C, D)$ with the state matrix A in a *block-diagonal form* (BDF)

$$A = \mathrm{diag}(A_1, \ldots, A_k) \tag{2}$$

and the matrices B and C partitioned accordingly

$$B = [\,B_1^T, \ldots, B_k^T\,]^T, \qquad C = [\,C_1, \ldots, C_k\,]. \tag{3}$$

This partition of system matrices is equivalent with the additive decomposition $G = D + \sum_{i=1}^{k} G_i$, where $G_i(\lambda) = C_i(\lambda I - A_i)^{-1}B_i$, for $i = 1, \ldots, k$. We use this decomposition to present an unifying treatment of modal dominance analysis methods.

The earlier modal reduction methods [1, 2, 3, 4] concerns exclusively with continuous-time systems and always assume that A is *diagonalizable*, and thus all blocks in (2) are 1×1. An eigenvalue λ_i is called *dominant* (or *slow*) if it is situated not too far from the imaginary axis and *non-dominant* (or *fast*) otherwise. The fast modes lying far from the imaginary axis are always neglected, even if they have a substantial contribution to the system dynamics.

A more satisfactory approach was proposed by Litz [9]. As dominance index for an eigenvalue λ_i he used the quantity

$$R_i = \|D_1 G_i(0) D_2\|, \tag{4}$$

where D_1 and D_2 are diagonal output and input scaling matrices, respectively, and $\|F\| := \sum_{i,j} |f_{ij}|$ or $\|F\| := \max_{i,j} |f_{ij}|$. Those eigenvalues having the largest dominance indices are called *dominant* and are retained in the reduced model. In order to evidence week dynamic interactions, Litz also introduced a somewhat heuristically defined frequency-weighted dominance index. The choice of matrices D_1 and D_2 should reflect the relative importance of different output and input variables. A possible choice for the diagonal elements of these matrices is to take them as the reciprocal of the absolute maximum values of the the corresponding output and input variables. Note that dominance indices equivalent with (4) can be defined by using any norm for TFMs as for instance the 2-, ∞- or Hankel-norm. Each TFM $G_i(\lambda)$ being of the form $C_iB_i/(\lambda - \lambda_i)$, the evaluation of these norms can be done by using easily computable explicit formulas: $\|G_i\|_\infty = \Gamma_i$, $\|G_i\|_2 = \sqrt{|\lambda_i|/2}\Gamma_i$, $\|G_i\|_H = \Gamma_i/2$, where $\Gamma_i = \|G_i(0)\|_2$.

The main limitation of using such dominance indices is the requirement for A to be diagonalizable. Even if A is diagonalizable, all discussed dominance indices are not appropriate for detecting exact or nearby structural non-minimality, as evidenced by the following simple example $A = \mathrm{diag}(-1, -1, -10)$, $B = [\,1\;1\;1\,]^T$, $C = [\,1\;-1\;1\,]$. Apparently the *slow* eigenvalues $\lambda_1 = \lambda_2 = -1$ should be kept in the reduced model and the *fast* eigenvalue $\lambda_3 = -10$ should be removed. The dominance indices $R_1 = R_2 = 1$, $R_3 = 0.1$ computed with (4) support this decision. However, it is easy to observe that an *exact* minimal realization of this system is $A = -10$, $B = 1$, $C = 1$.

The possible enhancements of the modal dominance analysis are directed towards handling the cases of multiple eigenvalues, or of exact or nearby non-minimality. We assume that in the BDF (2), any of two diagonal blocks have no common eigenvalues. Let n_i be the order of the i-th block and let $\sigma_j^{(i)}$, $j = 1, \ldots n_i$ the decreasingly ordered *Hankel singular values* (HSV) of the subsystem $G_i = (A_i, B_i, C_i)$ (the square-roots of the eigenvalues of the product of the corresponding gramians). The eigenvalues of a diagonal block A_i for which $\sigma_{n_i}^{(i)} > \varepsilon$, are called *dominant*, where ε is a given tolerance on the HSV. If $\sigma_1^{(i)} \le \varepsilon$ then the eigenvalues of A_i are called *non-dominant*. If $\sigma_j^{(i)} > \varepsilon$ for $j = 1, \ldots r_i$, then r_i of the eigenvalues are dominant and $n_i - r_i$ are non-dominant. To uncontrollable and/or unobservable eigenvalues correspond null singular values. Thus, by setting $\varepsilon = 0$, the dominant eigenvalues are those which are both controllable and observable. The non-dominant part of a subsystem $G_i = (A_i, B_i, C_i)$ can be removed by applying one of several powerful model reduction methods, as for instance the *balancing-free square-root* variant of B&T method [10].

The following straightforward procedure can be used to compute reduced order models by combining the modal approach with a suitable model reduction method capable to handle non-minimal systems:

1. Reduce the system (A, B, C, D) to the additively decomposed form (2)-(3), where $\lambda(A_i) \cap \lambda(A_j) \neq \phi$ for $i \neq j$.

2. For $i = 1, \ldots, k$ determine r_i, the number of dominant eigenvalues of block A_i.

3. For each n_i-th order subsystem $G_i = (A_i, B_i, C_i)$ compute its r_i-th order dominant part $G_{ir} = (A_{ir}, B_{ir}, C_{ir}, D_{ir})$ by using a suitable model reduction algorithm.
4. Construct $G_r = (A_r, B_r, C_r, D_r)$, where $A_r = \text{diag}(A_{1r}, \ldots, A_{kr})$, $B_r = [\, B_{1r}^T \ \ldots \ B_{kr}^T \,]^T$, $C_r = [\, C_{1r} \ \ldots \ C_{kr} \,]$, $D_r = D + \sum_{i=1}^{k} D_{ir}$.

This procedure can be easily implemented to determine a reduced system of a *specified* order or a reduced system G_r satisfying $\|G - G_r\| \leq \varepsilon_a$, where ε_a is a given absolute error tolerance. In the latter case, the orders r_i of reduced subsystems G_{ir}, $i = 1, \ldots, k$ can be usually determined automatically. For instance when using the B&T method we can choose r_i such that for a given ε_a we have

$$\|G - G_r\|_\infty \leq 2 \sum_{i=1}^{k} \sum_{j=r_i+1}^{n_i} \sigma_j^{(i)} \leq \varepsilon_a,$$

where we used the expressions of bounds derived in [7] for the B&T method. Note however that the actual error is generally greater (sometimes even much greater) than that resulting from the application of the B&T method directly to the whole system. Various other aims (DC-gain matching, phase preserving) can be accommodated by using alternative techniques (see [11] for a survey of model reduction methods). It is easy to see that when A has distinct eigenvalues, then the above procedure can be so devised to be equivalent with any of mentioned modal methods.

4. NUMERICAL ASPECTS

The model reduction procedure of previous section can be implemented by using exclusively numerically reliable algorithms. For the computation of the BDF at step 1 the algorithm of [12] can be used followed possibly by the reordering and enlarging of diagonal blocks. Note however that in many cases (finite-element models, non-minimal TFM realizations) A is already block-diagonal. In such cases only the reordering of blocks is necessary in order to include nearby eigenvalues in the same blocks.

For the dominance analysis at step 2 the HSV can be computed very accurately by using the square-root algorithm of [13]. The same algorithm is applicable to both continuous- and discrete-time systems. The only difference consists in solving continuous- or discrete-time Lyapunov equations to compute the corresponding gramians. The term *square-root* designates a class of new model reduction methods with enhanced accuracy in which the computation of reduced models is based exclusively on square-root information as for instance the Cholesky factors of the gramians. The computation of Cholesky factors can be done by solving directly for these factors the corresponding Lyapunov equations by using the algorithms proposed in [14].

The reduction at step 3 can be done by using any of the recently developed model reduction algorithm with enhanced accuracy (the so-called *square-root* or *balancing-free square-root* methods) (see the references in the companion paper [11]). All these methods are appropriate to handle exact or nearby non-minimality and thus can be also used very effectively as minimal realization procedures. At both steps 2 and 3 additional computational efficiency arises by exploiting the particular quasi-upper triangular form of diagonal matrices A_i which results usually from the reduction to BDF.

Because of usually low dimensions of subsystems G_i, the involved computational effort is mainly due to the reduction to the BDF and thus is about $15n^3$ operations. If the procedure is properly implemented, all computations can be done practically with minimum additional storage (at most n^2 locations if the reduction to BDF is necessary).

5. CONCLUSIONS

A model reduction procedure based on an enhanced modal dominance analysis technique has been proposed. The proposed procedure fulfills the basic requirements (generality, numerical reliability, enhanced accuracy) for a satisfactory numerical algorithm and thus can serve as basis for robust software implementation. The new procedure extends the range of applicability of the modal approach to the reduction of arbitrary continuous- or discrete-time systems. In the same time, it can be seen as enlarging also the applicability of many powerful model reduction methods to very large order systems.

9. REFERENCES

[1] E. J. Davison. A method for simplifying linear dynamic systems. *IEEE Trans. Autom. Contr.*, 11:93–101, 1966.

[2] S. A. Marschall. An approximate method for reducing the order of a linear system. *Contr. Eng.*, 10:642–648, 1966.

[3] M. R. Chidambara. Further remarks on simplifying linear dynamic systems. *IEEE Trans. Autom. Contr.*, 12:213–214, 1967.

[4] A. Fossard. On a method for simplifying linear dynamic systems. *IEEE Trans. Autom. Contr.*, 15:261–262, 1970.

[5] L. Litz. Ordnugsreduktion linearer Zustandsraummodelle durch Beibehaltung der dominanten Eigenbewegungen. *Regelungstechnik*, 27:80–86, 1979.

[6] B. C. Moore. Principal component analysis in linear system: controllability, observability and model reduction. *IEEE Trans. Autom. Contr.*, 26:17–32, 1981.

[7] K. Glover. All optimal Hankel-norm approximations of linear multivariable systems and their L^{∞}-error bounds. *Int. J. Control*, 39:1115–1193, 1974.

[8] H. Kiendl and K. Post. Invariante Ordnungsreduktion mittels transparenter Parametrierung. *Regelungstechnik*, 36:92–101, 1988.

[9] L. Litz. Praktische Ergebnisse mit einem neuen modalen Verfahren zur Ordnungsreduktion. *Regelungstechnik*, 27:273–280, 1979.

[10] A. Varga. Efficient minimal realization procedure based on balancing. In A. El Moudni, P. Borne, and S. G. Tzafestas, Eds., *Prepr. of IMACS Symp. on Modelling and Control of Technological Systems*, vol. 2, pp. 42–47, 1991.

[11] A. Varga. Numerical methods and software tools for model reduction. In *Proc. of 1st MATHMOD Conf., Viena*, 1994.

[12] C. Bavely and G. W. Stewart. An algorithm for computing reducing subspaces by block diagonalization. *SIAM J. Numer. Anal.*, 16:359–367, 1979.

[13] M. S. Tombs and I. Postlethwaite. Truncated balanced realization of a stable non-minimal state-space system. *Int. J. Control*, 46:1319–1330, 1987.

[14] S. J. Hammarling. Numerical solution of the stable, non-negative definite Lyapunov equation. *IMA J. Numer. Anal.*, 2:303–323, 1982.

ORDER REDUCTION FOR CONTROL APPLICATIONS

Peter Hippe
Institut für Regelungstechnik
Universität Erlangen-Nürnberg
Cauerstraße 7, D-91058 Erlangen

Abstract. When comparing the quality of different reduced order models, the purpose of the order reduction should be stated clearly. Here the problem of compensator order reduction is adressed. Starting from a full order compensator, the reduction process is carried out in view of the desired closed loop properties. The methods used are the frequency weighted balancing technique of Enns, and the so-called LQG balancing. Using three nontrivial examples from literature, the two cited methods are compared with unweighted reduction results.

1. INTRODUCTION

Using modern control system synthesis, the resulting compensators usually have the order of the plant. In view of industrial applications, however, compensators of small orders are often desirable. Therefore, order reduction plays an important role in control literature [11].

With a few exceptions, order reduction has been considered as an isolated problem. Given a system of nth order, find a reduced order model of order $n_r<n$, such that the input output behaviour of the system and of the reduced order model coincide as closely as possible. Probably the most widely accepted method for obtaining such models is the balancing technique [4]. If the reduced order model is needed for simulation studies, that model out of two will be preferred, whose time responses match that of the the original system best.

If, however, reduced order models are used for compensator design, or if one looks for a reduced order model of a given compensator, the above criterion does not necessarily apply. Out of two reduced order compensator models, the one giving a better approximation of the full order compensator's step response could yield a closed loop response inferior to that achieved with the other. Therefore, attempts have been made to include the actual goal in the reduction process, namely the closed loop response resulting with the reduced order compensator.

One of the first methods was the so-called frequency weighted balancing, developed by Enns in his Stanford dissertation [3]. The compensator balancing is modified such, that not the compensator's frequency response, but that of the closed loop is the approximation goal.

Another method is the LQG balancing of Jonckheere and Silverman [8]. Starting from an LQG design for the compensator, the balancing is carried out for the two cost functions (quadratic control and estimation error covariance). A different approach to reduced order compensator design was presented in [9]. Starting from a compensator for a reduced order plant model, the (reduced order) compensator is updated to make up for the modelling errors.

In this contribution we assume that a full order compensator has been designed to meet the requirements for the closed loop dynamics, and that this compensator is then reduced in order. Three methods are compared, namely compensator balancing with subsequent truncation, frequency weighted balancing, and LQG balancing. After a short description of the methods, they are applied to three examples taken from literature. It turns out, that in nearly all cases the frequency weighted balancing method of Enns gives the best results.

2. BALANCING

The balancing method is well known. The state equations of a system are transformed such, that the controllability and the observability grammians are both diagonal and equal, with the diagonal entries arranged in descending order. By truncation of the last κ states, a reduced order model with $n_r = n-\kappa$ is obtained. A reduced order model which is of better quality in the frequency range of interest results from keeping the steady state properties of the discarded states within the model [7]. This is automatically included in the *modred*-function of MATLAB. Models obtained from this procedure will be called Trunbal models in the sequel.

3. FREQUENCY WEIGHTED BALANCING

We assume that a compensator of order n has been designed to meet the design specifications for the considered closed loop. The idea of Enns can be described as follows: find a balanced representation of the compensator such, that by truncation of its last κ states, the error frequency response between the closed loop representations with full order and with reduced order compensator becomes small. In the SISO case, this can be carried out by balancing the frequency weighted compensator transfer function $F_C(s)=N_C(s)/D_C(s)$, where the weighting function is the closed loop transfer function for input disturbances

$$W_o(s) = \frac{N(s)D_C(s)}{D(s)D_C(s) + N(s)N_C(s)} \tag{1}$$

Postmultiplying the compensator $F_C(s)$ by this weighting function yields

$$F_C(s)W_o(s) = \frac{N(s)N_C(s)}{D(s)D_C(s) + N(s)N_C(s)} \tag{2}$$

so that the reference transfer function of the classical one degree of freedom structure is the approximation goal. The algorithm for the frequency weighted balancing can either be found in [3] or in [7]. Closed loop stability is guaranteed also with the reduced order compensator $F_C^r(s)$ as long as

$$E_\infty = \| W_o(s)[F_C(s) - F_C^r(s)] \|_\infty < 1 \tag{3}$$

holds. Again, it proves to be beneficial to retain the statics of the discarded states. The resulting models shall be characterized as Freqbal models.

4. LQG BALANCING

Prerequisite for the LQG balancing technique is an LQG compensator design [8]. The system equations are then transformed such, that the solutions P and $\bar{P}$ of the two dual Riccati equations

$$PA + A^TP - PBR^{-1}B^TP + Q = 0 \quad \text{und} \quad A\bar{P} + \bar{P}A^T - \bar{P}C^T\bar{R}^{-1}C\bar{P} + \bar{Q} = 0 \tag{4}$$

are diagonal and equal, with the diagonal entries arranged in descending order. Discarding the last κ states of the (full order) compensator

$$\begin{aligned} \dot{z} &= T(A - DC - BK)T^{-1}z + TDy \\ u &= KT^{-1}z \end{aligned} \tag{5}$$

a reduced order compensator is obtained. An algorithm for LQG balancing is available [5]. The reason for discarding states with minor contributions to the LQG cost functions is not too obvious. However, compared to the isolated order reduction either for the plant or for the compensator alone, it takes the closed loop situation well into account. Different from the other two methods, it works without modifications also for unstable compensators. The models resulting from LQG balancing (also here, retaining the statics gives a considerable improvement of model quality) will be denoted by LQGbal.

5. EXAMPLES

Tables 1 to 3 contain the complete data for the examples. Denoting the numerator and denominator polynomials by $N(s) = a_ms^m + ... + a_0$ and $D(s) = s^n + b_{n-1}s^{n-1} + ... + b_0$ the Tables give the coefficients a_i and b_i in descending order.

5.1 The four disc system

This experimental system was developed in Stanford to investigate flexible structures. It consists of four discs connected by a flexible wire, with a motor for applying torques to the third disc, and a sensor measuring the angular displacement of the first disc. The system has two poles at s=0, its vibratory modes have only 2% damping, and it is non-minimum phase [3]. Fig. 1 shows in solid lines the reference step responses for a

System	N:	6.4432e-3	2.3196e-3	7.1252e-2	1.00020
8th order		.104550	.99551		
	D:	0.1610	6.0040	0.58215	9.98350
		0.40727	3.9820	0	0
Compens.	N:	1.05942	.210907	6.36779	.861650
8th order		10.6033	.849316	4.23928	.180
	D:	3.60685	12.4957	25.6244	42.2586
		49.9108	43.8075	27.0835	12.2954
Trunbal	N:	.544827	.807423	1.25372	.197770
3rd order	D:	13.4208	1.89438	13.5092	
Freqbal	N:	.241883	-.0038002	.163794	.007002
3rd order	D:	.564788	.974336	.478296	
LQGbal	N:	-1.388e-4	.118817	.369896	.0527747
3rd order	D:	2.51008	3.40513	3.60496	

Table 1. Data for Example 1

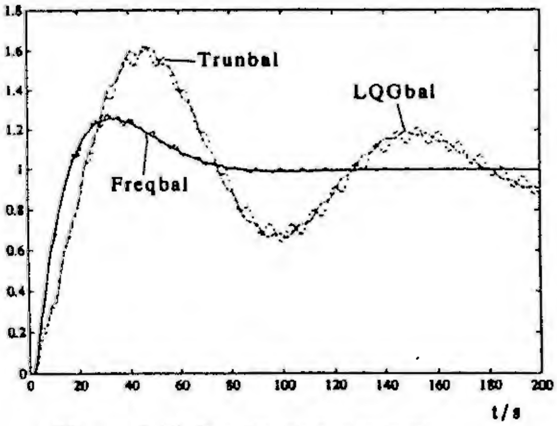


Figure 1. Reference step responses

System	N:	.089506	.857467	-.326526	-1.36961
5th order		3.74806			
	D:	7	16	8	-17
		-15			
Compens.	N:	1.89618e4	1.51784e5	4.55952e5	6.08726e5
5th order		2.85596e5			
	D:	142.0841	1.032977	-8812.93	1.49508e4
		6.13609e4			
Trunbal	N:	-.0781726	1.89979e4	1.12149e5	2.30824e5
4th order		1.42782e5			
	D:	140.393	-297.998	-8.0571e3	3.06769e4
Freqbal	N:	-.837790	1.92414e4	1.12932e5	2.32607e5
4th order		1.43691e5			
	D:	142.4830	-311.4323	-8079.198	3.08723e4
LQGbal	N:	-8.263e-4	1.87743e4	9.41966e4	1.83472e5
4th order		1.08751e5			
	D:	169.1331	-805.5315	-4721.264	23365.437

Table 2. Data for Example 2

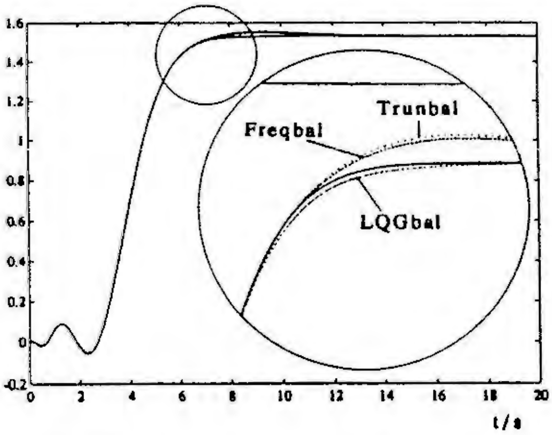


Figure 2. Input disturbance step responses

System	N:	-1e-4	-1.6e-6	-2.78264e-5	-1.87056e-6
8th order		-1.94543e-4	6.51688e-7	5.90592e-5	
	D:	.53	3.05276	1.37533	1.838526
		.5232089	.342179	.0282333	.0144229
Compens.	N:	-1214.957	-1390.831	-3644.978	-3449.347
8th Order		-2077.973	-1106.165	-292.3311	-104.3821
	D:	3.099396	7.712953	12.78172	15.94241
		14.55674	9.512509	3.856581	.6836875
Trunbal	N:	10.27270	-1305.092	182.6832	-3194.357
5th order		492.3531	-670.4497		
	D:	2.138934	4.472006	6.627740	4.615417
		4.391349			
Freqbal	N:	-262.6990	-298.8781	-593.2736	-717.9377
5th order		-31.91177	-143.7487		
	D:	.5452315	3.493795	1.596430	2.840391
		.9415331			
LQGbal	N:	-.0325069	-1213.005	-718.1121	-2867.837
6th order		-1654.844	-289.1383	-427.0284	
	D:	2.596341	5.829594	9.020316	8.834577
		7.133880	2.796974		

Table 3. Data for Example 3

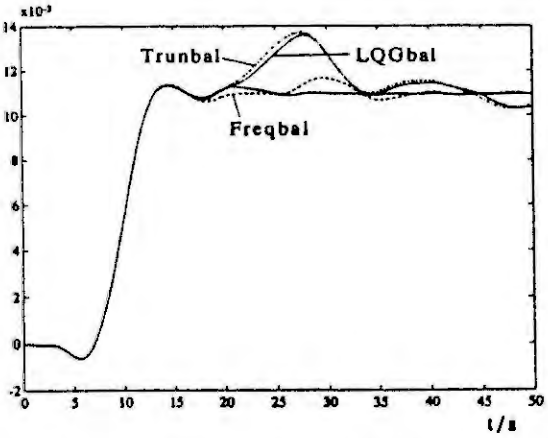


Figure 3. Input disturbance step responses

compensator designed in [10], (case q_2=100). Reducing the compensator from 8th order to third order, Fig.1 shows the results for the reduced order models (Trunbal dotted, Freqbal broken and LQGbal dash-dotted lines). The Freqbal results stay closest to the nominal one, while the Trunbal and LQGbal models give considerable deviations.

5.1 An unstable system

Next consider the LQG control of an unstable system, previously investigated in [6]. The weightings were $Q = 100c^Tc,\ R = 1,\ \bar{Q} = I,\ \bar{R} = 0.001$. Fig. 2 shows in solid lines the input disturbance step response with the compensator of (full) order 5. This compensator has two unstable modes and no stabilizing model with n_r<4 could be found. Fig. 2 shows the responses with 4th order models (Trunbal dotted, Freqbal broken, and LQGbal dash-dotted). Here, LQG balancing seems to work especially well.

5.3 Flexible missile

The third system, a structure block of a flexible missile, was already investigated in [2]. Again an LQG compensator was designed with the weightings $Q = 10^6c^Tc,\ R = 1,\ \bar{Q} = 10^6bb^T,\ \bar{R} = 1$. Fig. 3 shows in solid lines the input disturbance step response with the nominal compensator of order 7. The reduced order Trunbal and Freqbal models of 5th order give the responses in dash-dotted and broken lines, respectively. The dotted line results with an LQGbal model of order 6. Closed loop stability is achieved for n_r>1 with Freqbal, for n_r>4 with Trunbal, and for n_r>5 with LQGbal models.

6. CONCLUSIONS

Using nontrivial examples, different methods for compensator order reduction were investigated. The results demonstrate that better results are obtained, when closed loop properties are considered in the reduction process. Only for example two, where the compensator contained two unstable poles, the LQGbal models gave better results than the isolated Trunbal procedure. The best results can be obtained with the frequency weighted balancing of Enns. It should be noted, that the frequency weighted balancing presented in [1] is different from the one considered here, as the restrictions imposed in [1] do not allow weighting functions of the form (1).

7. REFERENCES

[1] Al-Saggaf, U.M., and Franklin, G.F., Model reduction via balanced realizations: An extension and frequency weighting techniques. IEEE Trans. Aut. Control AC-33 (1988), 687-692.

[2] Chen, T.C., Chang, C.Y., and Han, K.W., Model reduction using the stability-equation method and the continued-fraction method. Int. J. Control 32 (1980), 81-94.

[3] Enns, D.F., Model reduction for control system design. Dissertation Stanford, 1984.

[4] Glover, K., All optimal Hankel-norm approximations of linear multivariable systems and their L_∞-error bounds. Int. J. Control 39 (1984), 1115-1193.

[5] Guth, R., Toolbox zum Entwurf strukturbeschränkter Zustandsregler mittels balancierter Realisierungen. GMUG Rundbrief No. 5, Dezember 1992.

[6] Hippe, P., Bemerkungen zur Ordnungsreduktion instabiler Systeme. Automatisierungstechnik 39 (1991), 135-139.

[7] Hippe, P., Frequenzgewichtete Ordnungsreduktion zum Reglerentwurf. Automatisierungstechnik 40 (1992), 447-453.

[8] Jonckheere, E.A., and Silverman, L.M., A new set of invariants for linear systems - Application to reduced compensator design. IEEE Trans. Aut. Control AC-28 (1983), 953-964.

[9] Kreisselmeier, G., and Mevenkamp, M., A note on reduced-order controller synthesis. IEEE Trans. Aut. Control AC-33 (1988), 878-880.

[10] Liu, Y., and Anderson, B.D.O., Controller reduction via stable factorization and balancing. Int. J. Control 44 (1986), 507-531.

[11] Troch, I., Müller, P.C., and Fasol, K.-H., Modellreduktion für Simulation und Reglerentwurf. Automatisierungstechnik 40 (1992), 45-53, 93-99, 132-141.

Order Reduction in the Control Design Configuration

P.M.R. Wortelboer
Philips Research Laboratories
Prof. Holstlaan 4, 5656 AA Eindhoven, The Netherlands
Tel.: +31 40 742917, E-mail: wortel@prl.philips.nl

O.H. Bosgra
Mechanical Engineering Systems and Control Group
Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands

Abstract. This paper focusses on order reduction within a general controller design configuration. Theory and usage of a simple extension of frequency weighted balanced reduction are explained. The main features are the direct link with controller design methods and the ease of adjusting frequency weighting functions to further limit the performance degradation. The procedure is illustrated on a CD-player tracking problem starting from a 120^{th}-order model. Both model reduction and controller reduction are performed.

1 Introduction

The increasing popularity of model-based controller design in industrial applications has revealed that LQG-, H_∞-, and μ-controller design codes, to name a few, are only intermediate steps in synthesizing controlled systems with satisfactory performance. One of the hardest problems is to find an appropriate linear model of the system to be controlled and the operating conditions under which a good description is required. In the earliest stage there is little ground to neglect dynamic phenomena, obliging one to model in great detail and this usually leads to high-order models. There are two reasons to reduce high-order models prior to controller design. The first is to speed up the design process and to make it more reliable. The second reason is the controller complexity that roughly equals the complexity of the controller design model. Low complexity is to be preferred from a reliability and cost point of view. In general it will not be possible to achieve satisfactory performance with a low-order controller without any type of iteration. Trial and error is indispensable to obtain a good understanding of the essential system dynamics and the impact of the performance wishes on the controller complexity. In the light of this it is important to have easy manageable reduction schemes.

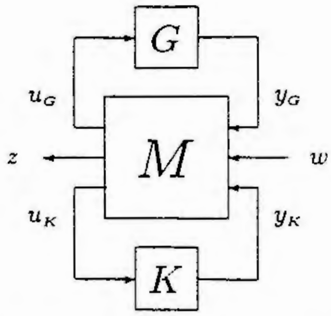
This paper discusses a collection of extensions of the famous balance and truncate procedure to achieve fast and accurate order reduction within the controller design configuration. Iterations on the modelling method or the controller design method are not discussed. We concentrate on continuous time linear systems with performance specifications in the frequency domain.

In literature many results have been reported on order reduction of linear continuous time systems. Balanced reduction [12] is one of the strongest. The extensions to order reduction within controlled systems owe much to Enns [5, 6], who developed frequency weighted balanced reduction, that can be used for closed-loop relevant reduction. It does not apply to unstable systems however.

Graph or fractional balanced reduction removed this restriction and besides can find reduced-orders for which the closed-loop system is guaranteed to be stable [11, 3]. LQG-balanced reduction [10] is equivalent, [13] discusses the H_∞-case. An overview of closed-loop relevant reduction methods is given in [1]. A drawback of these methods is that they are computationally demanding and that they cannot be easily modified to fit into a controller design configuration with changing frequency weightings.

The method that we use here is basically an extension of frequency weighted balanced reduction that can be applied directly to either the system model or the controller within the weighted closed-loop configuration [4], and only requires the closed-loop system to be stable. It is supplemented with scalar interval based frequency weighting functions that can be used to *i)* model frequency contents of any input signal or *ii)* express the importance of approximating output signals in certain frequency ranges [7, 14, 15].

In order to be able to do *system* and *controller* reduction in any stage of design, the system and controller have to be isolated from the controller design configuration. The general interconnection structure proposed is given alongside: $z = \mathcal{I}(M, G, K)w$. This interconnection structure was first used to define the robust controller design problem [2, 16]. The performance is the attenuation that is achieved between signals w and z. G is the system model, and K is the controller. M will be referred to as the connector system or connector and incorporates any selection of weighting functions. This structure is closely related to the configuration that is used for calculating optimal controllers.



The fruitful use of the method heavily depends on an adequate implementation. The result of order reduction should be visualized in relevant performance terms. In the current MATLAB4 implementation these are attenuation measures (H_2, H_∞) and frequency-magnitude plots. These plots are the clue to making scalar frequency functions (mouse driven) that can improve the approximation locally (on the frequency axis).

The organisation of this paper is as follows. First the theory of reduction is reviewed, then we present reduction results needed in designing a tracking controller for a CD-player mechanism.

2 Reduction within a general controller-design configuration

Continuous-time finite-dimensional time-invariant linear systems can be written in state-space as [1]

$$G = [A \quad {}^{B} \quad {}_{C} \quad D] \qquad \text{for} \qquad G: \begin{array}{rcl} \dot{x} &=& Ax + Bu \\ y &=& Cx + Du \end{array}$$

For clarity we will write G_n for a realization of n^{th}-order. n is the full-order, r is the reduced-order and $t = n - r$. Using $\tilde{x} = T^{-1}x$ other realizations can be defined:

$$\tilde{G} = [T^{-1}AT \quad {}^{T^{-1}B} \quad {}_{CT} \quad D] \qquad \text{for} \qquad \tilde{G}: \begin{array}{rcl} \dot{\tilde{x}} &=& T^{-1}AT\tilde{x} + T^{-1}Bu \\ y &=& CT\tilde{x} + Du \end{array}$$

Note that for $T \neq I$, $G \neq \tilde{G}$, but their transfer matrices are equal: $G(s) = \tilde{G}(s) = C(sI{-}A)^{-1}B + D$.

Balanced reduction is a special case of truncation of a system realization. Let $\Gamma_r = \begin{bmatrix} I_r \\ O \end{bmatrix}$ with $O \in \mathbb{R}^{t \times r}$ a zero matrix, then the truncation of realization G_n to order r is:

$$G_r = \mathcal{R}_{[\Gamma_r^T, \Gamma_r]}(G_n) \overset{\text{def}}{=} [\Gamma_r^T A \Gamma_r \quad {}^{\Gamma_r^T B} \quad {}_{C\Gamma_r} \quad D] = [A_{(1:r,1:r)} \quad {}^{B_{(1:r,:)}} \quad {}_{C_{(:,1:r)}} \quad D] \overset{\text{def}}{=} \mathcal{R}_{r-n}(G_n),$$

In the same format a state transformation can be written as

$$\tilde{G}_n = \mathcal{R}_{[T^{-H}, T]}(G_n)$$

Let $\breve{G}_n = \mathcal{R}_{[\breve{T}^{-H}, \breve{T}]}(G_n)$ be a balanced realization, and $\breve{R}_r = \breve{T}_{(:,1:r)}$, $\breve{L}_r = [\breve{T}^{-H}]_{(:,1:r)}$, then

$$\breve{G}_r = \mathcal{R}_{r-n}(\breve{G}_n) = \mathcal{R}_{[\breve{L}_r, \breve{R}_r]}(G_n) \overset{\text{def}}{=} \text{bal}\mathcal{R}_{r-n}(G_n)$$

defines balanced reduction.

The balanced states are ordered according to the Hankel Singular Values (HSVs) which makes balanced reduction unique for appropriate r. A balancing transformation $\breve{T} = \breve{R}_n$ ($\breve{T}^{-H} = \breve{L}_n$) satisfies

$$\breve{L}_n^H P_n \breve{L}_n = \breve{R}_n^H Q_n \breve{R}_n = \text{diag}(\sigma_n)$$

with $\sigma_n = \sqrt{\lambda(P_n Q_n)}$ the HSVs, $P_n = \mathcal{P}(G_n)$ the controllability Gramian and $Q_n = \mathcal{Q}(G_n)$ the observability Gramian. The transformation is exclusively based on P_n and Q_n:

$$[\breve{L}_n, \breve{R}_n] = \mathcal{T}(P_n, Q_n)$$

To define balanced reduction within a general interconnection structure we need to analyse the realizations of the interconnected system $\mathcal{I}(M, G, K)$. From linear fractional transformation theory [16] we know that

$$\mathcal{I}(M, G, K) = \mathcal{F}_l(\mathcal{F}_u(M, G), K) = \mathcal{F}_u(\mathcal{F}_l(M, K), G)$$

[1] $[A \quad {}^{B} \quad {}_{C} \quad D]$ is a *system realization*, not a collection of constant matrices $[A \quad B \quad C \quad D]$.

The key point for our scheme is that we make a realization of $\mathcal{I}(M,G,K)$ with a state vector that is built from G-states, M-states, and K-states in that precise order: $x_{_I}^T = \left[\begin{array}{ccc} x_{_G}^T & x_{_M}^T & x_{_K}^T \end{array}\right]$. Balanced reduction of G within $\mathcal{I}$ follows the standard balanced reduction procedure with the difference that instead of P_n, Q_n parts of the Gramians of $\mathcal{I}(M,G_n,K)$ are used. The scheme for G (or $[K]$) reduction then is:

$$\begin{aligned} P_{_G} &= [\mathcal{P}(\mathcal{I}(M,G_n,K))]_{(1:n,1:n)} \overset{\text{def}}{=} \mathcal{P}(\mathcal{I}(M,\underline{G_n},K)) \\ Q_{_G} &= [\mathcal{Q}(\mathcal{I}(M,G_n,K))]_{(1:n,1:n)} \overset{\text{def}}{=} \mathcal{Q}(\mathcal{I}(M,\underline{G_n},K)) \\ [\tilde{L}_n, \tilde{R}_n] &= \mathcal{T}(P_{_G}, Q_{_G}) \\ \tilde{G}_r &= \mathcal{R}_{[\tilde{L}_r,\tilde{R}_r]}(G_n) \overset{\text{def}}{=} \text{bal}\mathcal{R}_{r-n}(\mathcal{I}(M,\underline{G_n},K)) \end{aligned}$$

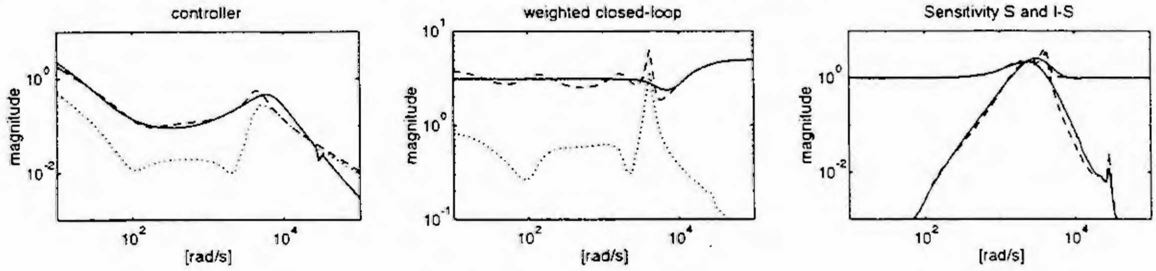
$$\left[\begin{aligned} P_{_K} &= \mathcal{P}(\mathcal{I}(M,G,\underline{K_n})) \\ Q_{_K} &= \mathcal{Q}(\mathcal{I}(M,G,\underline{K_n})) \\ \tilde{K}_r &= \text{bal}\mathcal{R}_{r-n}(\mathcal{I}(M,G,\underline{K_n})) \end{aligned}\right]$$

The full procedure allows interval-based frequency weighting of w and z [14, 15]. Let $\psi(\omega)$ be a positive symmetric frequency function which does not need to be continuous; it may even contain pulses. A direct solution exists for the $\psi_w(\omega)$-weighted controllability of system $\mathcal{I}$, and the $\psi_z(\omega)$-weighted observability of $\mathcal{I}$. We simply write $\mathcal{P}(\mathcal{I}\cdot\psi_w)$ and $\mathcal{Q}(\psi_z\cdot\mathcal{I})$. Parts of these frequency weighted Gramians are used to transform and truncate the system and/or controller: $\tilde{G}_r = \text{bal}\mathcal{R}_{r-n}(\psi_z \cdot \mathcal{I}(M,\underline{G_n},K)\cdot\psi_w)$.

3 Application to a CD-player Mechanism

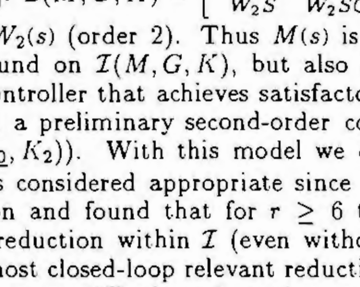
For a CD-player mechanism, the elastodynamic behaviour has been modelled by means of the finite-element method resulting in a model of order 120. It is assumed that this model describes the system well for a large number of operating conditions. The low inherent damping of such systems requires cautious order reduction. Open-loop reduction is not appropriate.

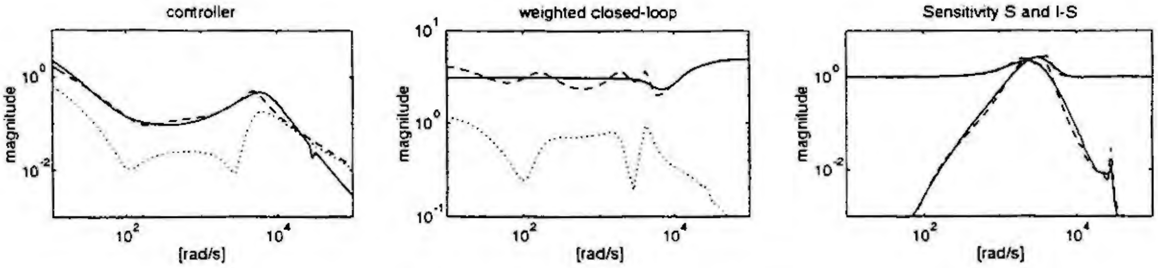
Only the tracking control is considered which means that $K(s)$ is siso. The configuration used in H_∞ controller design is a slight simplification of the one proposed in [8, 9]. $\mathcal{I}(M,G,K) = \left[\begin{array}{cc} W_3KS & W_3S \\ W_2S & W_2SG \end{array}\right]$ with sensitivity $S = [I - GK]^{-1}$ and weights $W_3(s)$ (order 3) and $W_2(s)$ (order 2). Thus $M(s)$ is of order 5. The performance specification is not merely an upper bound on $\mathcal{I}(M,G,K)$, but also includes sensitivity considerations and controller order. The simplest controller that achieves satisfactory performance is sought. The first step is model reduction. We used a preliminary second-order controller (PID-type) K_2 to reduce G_{120}: $G_{32} = \text{bal}\mathcal{R}_{32-120}(\mathcal{I}(M,\underline{G_{120}},K_2))$. With this model we designed K_{37} such that $\|\mathcal{I}(M,G_{32},K_{37})\|_\infty \leq 5$. Model reduction was considered appropriate since we achieved $\|\mathcal{I}(M,G_{120},K_{37})\|_\infty \leq 5$. Next we did controller reduction and found that for $r \geq 6$ the performance degradation was very small. Thus balanced controller reduction within $\mathcal{I}$ (even without additional $\psi(\omega)$) can quickly generate controllers of moderate order; most closed-loop relevant reduction techniques will give similar results. For $r < 6$ the reduction is much more difficult and requires some iteration with ψ. First we show that $K_4 = \text{bal}\mathcal{R}_{4-37}(\mathcal{I}(M,G_{32},\underline{K_{37}}))$ performs clearly worse than K_{37}:



— (with) K_{37}, - - (with) K_4, ··· difference

In order to diminish the performance degradation near 4000 rad/s, we tried to emphasize this frequency in the reduction procedure by adding a frequency pulse. The figure on the right shows the crucial peak of the (old) weighted closed-loop configuration together with the final pulse definition for the new reduction. $\hat{\psi}(\omega)$ was found after a few iterations. The 4^{th}-order controller $\hat{K}_4 = \text{bal}\mathcal{R}_{4-37}(\hat{\psi}\cdot\mathcal{I}(M,G_{32},\underline{K_{37}})\cdot\hat{\psi})$ performs almost as good as K_{37} (see next page):





— (with) K_{37}, - - (with) $\check{K}_4$, ··· difference

Balanced reduction ($\check{K}_4 = \text{bal}\mathcal{R}_{4-37}(K_{37})$) only achieved $\|\mathcal{I}(M, G_{32}, \check{K}_4)\|_\infty = 15.77$, a factor 3 worse. $\check{K}_4$ also performed satisfactorily on G_{120}.

In many more examples closed-loop balanced reduction with frequency shaping of the inputs and outputs was used to find a good trade-off between the order of the controller and the performance level. It should be stressed that the proposed iterative computer-aided procedure is only one way of achieving a satisfactory order-performance compromise.

4 Conclusion

Simple extensions of frequency weighted balanced reduction are very suitable for application of order reduction within controlled systems. Model and controller reduction can be used in conjunction with model-based control design, since the same configuration is used. Flexible computer implementation using MATLAB4 facilitates the interactive design of systems with relatively low-order controllers.

References

[1] Anderson, B.D.O., and Y. Liu, (1989). "Controller reduction: concepts and approaches," *IEEE Trans. Automat. Contr.*, vol.34, 802–812.

[2] Balas, G.J., J.C. Doyle, K. Glover, and A.K. Packard (1991). "μ-Analysis and Synthesis Toolbox", MUSYN Inc., Minneapolis.

[3] Bongers, P.M.M., and O.H. Bosgra, (1991). "Controller reduction with closed-loop stability margins," *Selected Topics in Identification, Modelling and Control, vol.3*, Delft Univ. Press, 35–41.

[4] Ceton, C., P. Wortelboer, and O. Bosgra, (1993). "Frequency weighted closed-loop balanced reduction," *Proc. 2nd European Control Conference,*, Groningen June 26- July 1, 1993, 697–701.

[5] Enns, D.F. (1984). "Model reduction with balanced realization: an error bound and a frequency weighted generalization," *Proc. 23rd IEEE Conf. Dec. & Contr.*, 127–132.

[6] Enns, D.F. (1984). *Model reduction for control system design*, Ph.D. Thesis, Dept. Aeronautics and Astronautics, Stanford University, Stanford, CA, USA.

[7] Gawronski, W., and J-N Juang, (1990). "Model Reduction for Flexible Structures," *Control and Dynamic Systems*, vol.36, 143–222.

[8] Groos, P.J.M. van, (1993). "Robust control of a Compact Disc Player." *Philips Technical Note TN 143/93.*

[9] Groos, P.J.M. van, M. Steinbuch and O.H. Bosgra, (1993). "Multivariable control of a compact disc player using μ synthesis", *Proc. 2nd European Control Conference*, Groningen June 26- July 1, 1993, pp. 981-985.

[10] Jonckheere, E.A., and L.M. Silverman (1983). "A new set of invariants for linear systems-application to reduced order compensator design," *IEEE Trans. Automat. Contr.*, vol.AC-28, 953–964.

[11] Meyer, D.G. (1988). "A fractional approach to model reduction", *Proc. American Control Conf.*, 1041–1047.

[12] Moore, B.C. (1981). "Principal component analysis in linear systems: controllability, observability, and model reduction," *IEEE Trans. Automat. Contr.*, AC-26, 17–32.

[13] Mustafa, D. (1989). "$\mathcal{H}_\infty$-Characteristic values," *Proc. 28th IEEE Conf. Dec. & Contr.* , 1483–1487.

[14] Wortelboer, P.M.R., and O.H. Bosgra, (1992). "Generalized frequency weighted balanced reduction," *Selected Topics in Identification, Modelling and Control, vol.5,* Delft Univ. Press, 29–36.

[15] Wortelboer, P.M.R., and O.H. Bosgra, (1992). "Generalized frequency weighted balanced reduction," *Proc. 31st IEEE Conf. Dec. & Contr.* , 2848–2849.

[16] Zhou, K., J. Doyle, and K. Glover, (1993). *Robust and Optimal Control, preprint*

CONTROLLER DESIGN FOR A HYDRO POWER PLANT BASED ON LINEARIZED ORDER REDUCED MODELS

Karl Heinz Fasol
University of Bochum, Dept. of Mechanical Engineering
D-44780 Bochum, Germany

Abstract. In the course of a simulation study, the high order nonlinear simulation model of a hydro power plant was elaborated by physical analysis. This model was linearized in a number of operating points and approximated by 5th order transfer functions to be used for stability analysis and controller design. It is the objective of the paper to report on the application of model approximation and order reduction to a real engineering problem.

1. INTRODUCTION

In recent years, several simulation studies for hydro power plants were carried out by the author. Reasons for such studies are to predict the dynamic behaviour of the system, to design the control algorithms, and to investigate problems appearing in already operating plants. Nowadays, such studies gaine significance when renewing old plants. In this connection, application of model simplification and order reduction may be important as shown in this contribution.

Recently, a simulation study was carried out for the three highhead turbine-generator sets of a plant commissioned and put into operation almost 40 years ago. Ever since, there were serious stability problems at speed controlled no-load operation before synchronizing with the grid. This was due to the fact that, regarding the state of both control engineering and controller technology in the fifties, one could not do better at the time the plant was designed. Persistently, the speed control loops performed limit cycles around set point thus causing rather high pressure amplitudes endangering the penstock. It was never possible to start all machines simultaneously but some "tricks" had to be applied to start and synchronize one unit after the other. Sometimes, this procedure took nearly 20 minutes from standstill to full load of the three machines.

In 1993, the old mechanical governors were replaced by digital controllers and the author was asked to design improved control algorithms for start-up and all other modes of operation. A similar task was already described in [1]. Among other commitments, the above mentioned 20 minutes should be reduced to 4 minutes.

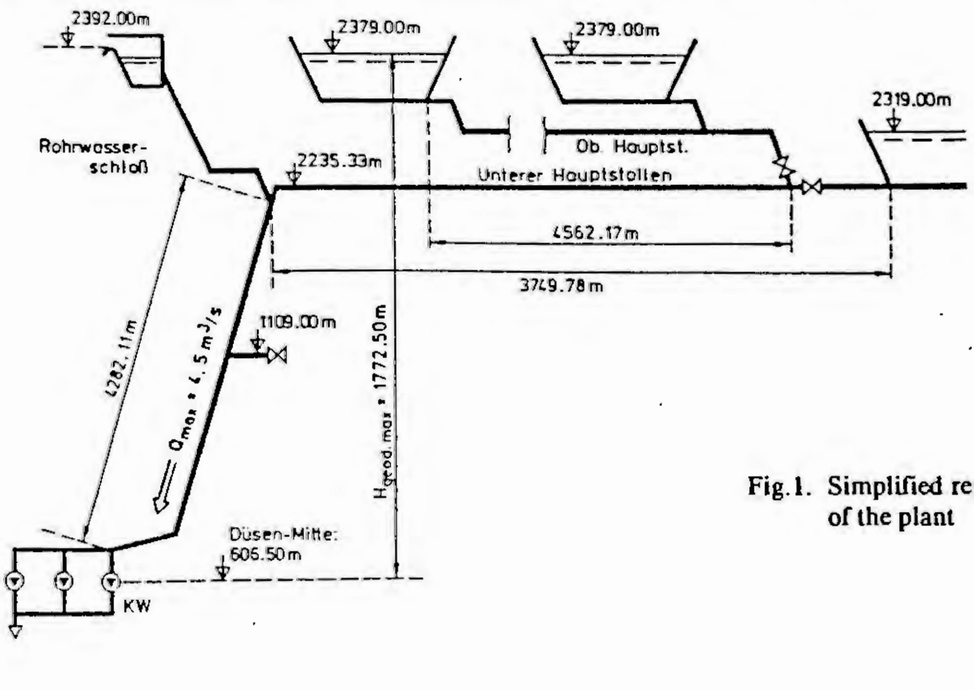


Fig.1. Simplified representation of the plant

In this paper, little extent will be dedicated to controller design but modelbuilding, applied order reduction, and model verification will be outlined with some more emphasis.

2. MODELBUILDING

Figure 1 is an extremely simplified representation of the plant. Note that the lenghts of both tunnels and the penstock are about 4 km each. The pressure head of the Pelton-type turbines is still the highest in the world.
A detailed mathematical model of the installation was developed by means of theoretical analysis. Models of all hydraulic subsystems are essentially based on the instationary equations of both continuity and motion. These relations are the basis for all methods to simulate unsteady flow through closed conduits [4,5]. The equations were discretized in the axial coordinate and used to model the tunnel sections, the surge tank, and the penstock. The model of the surge tank had to consider the complex profile of the tank. Since the transient performance of a hydraulic turbine is very fast in relation to that of the conduit, it was sufficient to model the turbine by its static performance characteristic. In the case of both start-up (no load) and isolated network operation , the relation between turbine power output and running speed resulted from the balance of inertias. For interconnected grid operation, however, it is common practice to assume constant rotating speed. Modelling the hydraulic system and the turbines was based on detailed plans of all installations, 35 years old results of acceptance tests, on-site measurements of friction losses, performance graphs, etc., and on appropriate experience. With some initial simplifying assumptions, the model consisted of 52 equations in its first stage. Further model simplification was based on physical considerations and supported by simulation. Finally, the **simulation model** consisted of 12 nonlinear and 13 linear first-order differential equations, 7 algebraic connections, and 5 nonlinear functions.

This model was simulated by means of the block oriented simulation tool FSIMUL which is described in [3]. This package is written in C and offers about 120 different block operations, unlimited number of calculation levels, display of block diagrams, facility to create user defined macros (sometimes called super blocks), as well as a comfortable user's guidance. Programming the above indicated highly nonlinear simulation model needed 327 block operations of FSIMUL; 309 of those were assembled to 20 defined macros. Simulation was carried out on a PC with a 486 processor. The relation between simulation time and real time was 1.8 for the above 25th order simulation model.

3. LINEARIZATION AND ORDER REDUCTION

Linearization and order reduction was carried out simultaneously by means of the multi purpose software tool REDU_RP described in [2]. This package applies the method of cost function vector optinuization to model approximation and controller design. The software offers 24 cost functions both in time domain and frequency domain as well as several optimization algorithms. The reduced model used for root loci calculations and controller design was developed as follows.

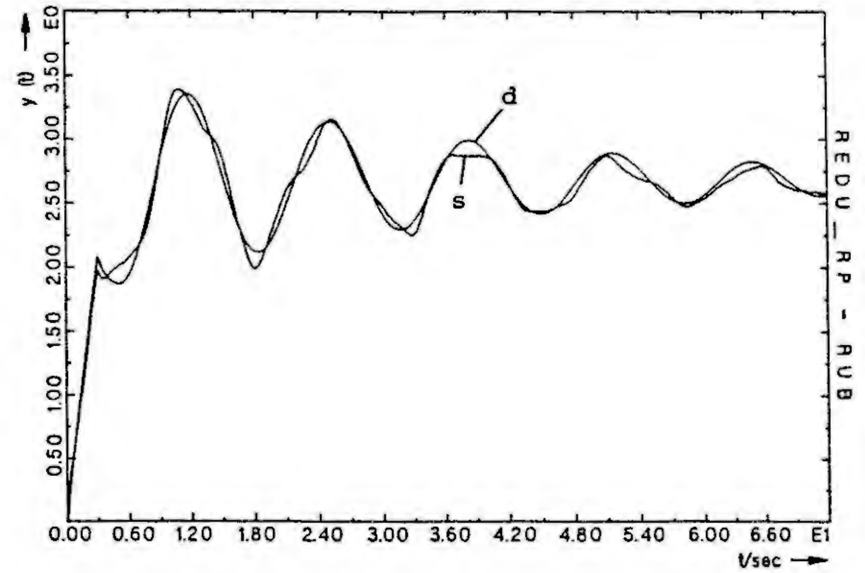


Fig.2. Example for model approximation. **s** simulation model, **d** design model. (y = 0 according to 23.9 MW)

The time response of the hydraulic system simulation model to a rampwise opening of the turbine's nozzle was simulated with FSIMUL and intermediately stored in a file. After playback into REDU_RP, this ramp response was approximated by the answer of a transfer function model with 4 zeros and 5 poles. Based on cost-function-vector optimization, as described in [2], the error between both responses was minimized. This procedure was executed for 10 operating points of grid supply and 6 operating points of no-load performance. This way, the hydraulic power acting on the turbine was represented by a 5th order transfer function with operating-point dependent parameters. This order reduced model was called **design model**. Figure 2 is an example: The turbine nozzle was opened from 80% to 100%. In the figure, the resulting oscillation has the period of 13.2 sec, corresponding with the imaginary parts $\pm j0.476\ \mathrm{sec}^{-1}$ of the ill-damped pair of eigenvalues (see Fig. 4).

4. MODEL VERIFICATION

To prove the reliability of both the 25th order simulation model and the 5th order design model, various comparisons between observations or measurements in the plant (before replacement of the old governors) and simulation results were made. Two examples are discussed in the following passages.

4.1 Simulation of the observed instability

This was one of the experiments to verify the 25th order simulation model. Figure 3, as example, demonstrates the no-load operation when each tuibine is controlled by the old governor (see eq.(1) in the following chapter). The two strip charts display the pressure head acting on the turbines. The upper one is a section of a recording from the plant; the other chart is the corresponding simulation result. First, two machines run in parallel. The speed control loop of the third machine is closed after 100 sec. The pressure amplitudes received in simulation fully agree with the inadmissible amplitudes observed in the plant. It should be mentioned that, contrary to simulation, the pressure signal recorded in the plant is filtered by sensor, transducer, and recorder.

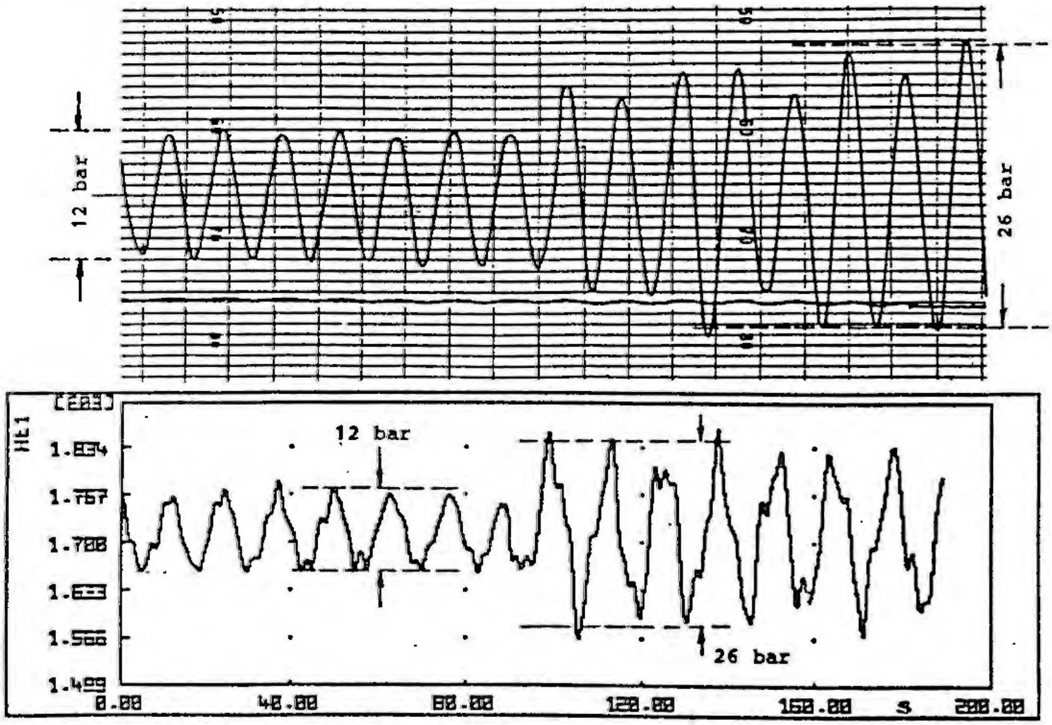


Fig. 3. Model verification: Penstock pressure head at no-load operation of two and three turbines in parallel.
Above: Recorded in the plant. Below: Result of simulation

4.2 Stability analysis with root loci

As explained in the introduction, the speed control loop was unstable at no-load operation all the time since commissioning. This unfavourable behaviour could be easily explained and demonstrated by calculating the respective root loci. This was one of the possibilities to prove the reliability of the design model.

The speed controller used until 1993 had the transfer function (speed deviation to turbine nozzle opening).

$$G_c(s) = \frac{6.25(s+0.20)}{s^2+6.39s+0.05}\ . \qquad (1)$$

The connection between turbine nozzle opening (at the respective no-load operating point) and hydraulic power is represented by

$$G_T(s) = \frac{573.3(s^4 + 1.98s^3 + 6.91s^2 + 0.33s + 1.58)}{s^5 + 17.5s^4 + 43.9s^3 + 113s^2 + 10.7s + 25.8}\,; \qquad (2)$$

(zeros: + 0.011 ±j0.485, -1.00 ± j2.39;
poles: - 0.002 ±j0.485, -1.20 ± j2.41, -15.1).

The non-minimum phase property is significant of high-head water supply systems.

Considering inertia and friction losses, the relation between turbine power and running speed of the turbine-generator set at no-load condition is described simply by

$$G_s(s) = \frac{6.211}{s + 0.0186}. \qquad (3)$$

Thus, the open loop transfer function has 5 zeros and 8 poles. Figure 4 shows the section of the root locus relevant for stability analysis. As not otherwise expected, the closed loop has a pair of eigenvalues at the imaginary axis with ± j0.476 sec^{-1} well corresponding with the first eigenfrequency measured in the plant (see Fig.3). Thus, the validity of the 5th order reduced model was sufficiently demonstrated.

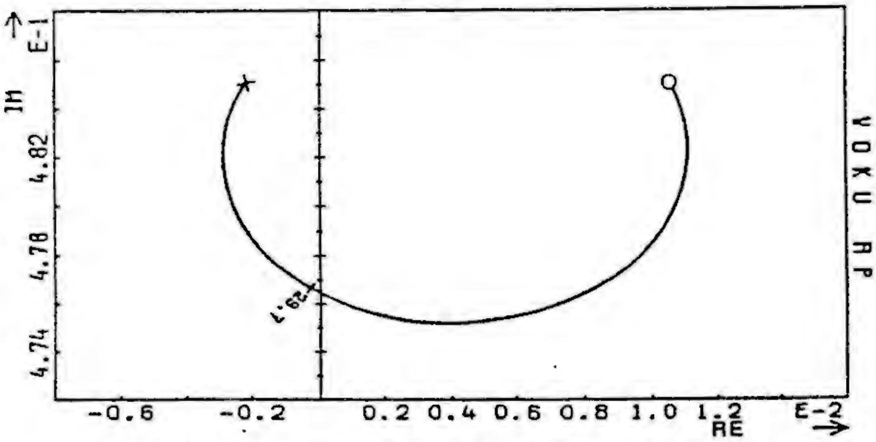


Fig.4. Branch of root locus responsible for the unstable no-load operation with the controller to be replaced

5. CONTROLLER DESIGN

Control algorithms were designed for speed control at both start-up (no-load) and isolated network conditions as well as power control at interconnected grid operation. With respect to the problematic eigenvalue distribution of the hydraulic system, excitation of penstock pressure had to be minimized. This necessity requests filtering of frequencies near and higher than the first eigenfrequency of the hydraulic system.

For speed control, the problem was solved by design of a rather complex cascade structured controller combined with a switching circuit. The deflector of the Pelton turbine is controlled essentially in a fast inner loop by a 2nd order controller with lead behaviour. The nozzle is controlled in the slow outer loop by a 3rd order low-pass filter with operating-point dependent gain. The power-output controller is a 2nd order lead-lag system, again with operating-point dependent gain. The order reduced design model was used and cost-function vector optimization by means of REDU_RP was applied to design these various control algorithms. Eventually, before programming the new digital controllers, many simulation runs were carried out with the high order simulation model. The results were satisfactory.

6. RESULTS

In September and November 1993, the new digital controllers were installed and the algorithms indicated in the previous chapter were implemented. Extensive on-site tests were carried out to prove the effectiveness of the designed algorithms. Figure 5 shows start-up of one turbine-generator set. Steady state speed is reached without overshoot after 70 sec. To demonstrate no-load stability, the speed setpoint is changed stepwise. Figure 6 shows the performance of one machine at interconnected grid operation. The power setpoint is changed rampwise. There is almost no overshoot of actual power output. Due to the effectiveness of the designed power control, the pressure head at the turbine remains nearly constant. Figure 7 shows the start-up procedure and no-load performance of three sets as recorded in the plant. To again demonstrate the no-load stability, the speed setpoint of one machine is changed stepwise whereas the other setpoints remain constant. Note that the actual value of running speed is sufficiently steady and the highest pressure amplitude is not more than 1.6 bar; it was 13 bar before our inverstigations. Thus, the comparison of the pressure amplitudes in Figures 3 and 7 is convincing and these results are very satisfactory. Finally, it should be mentioned that the time from standstill to full load of all sets could be reduced to 4 min as requested.

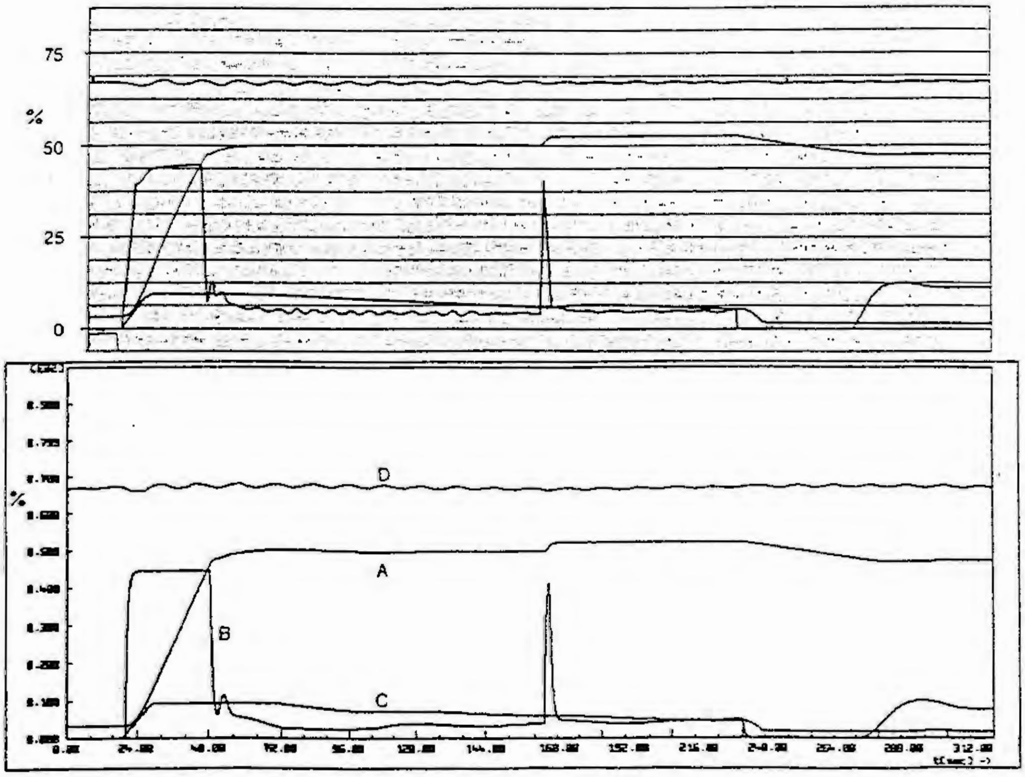


Fig.5. Start-up and speed control of one machine at no-load operation with the designed control algorithms. Above: Recorded in the plant. Below: Result of simulation. A: Running speed. B: Deflector position. C: Nozzle opening. D: Pressure at turbine inlet. Speed setpoint: 750 - 787.5 - 712.5 min^{-1} .

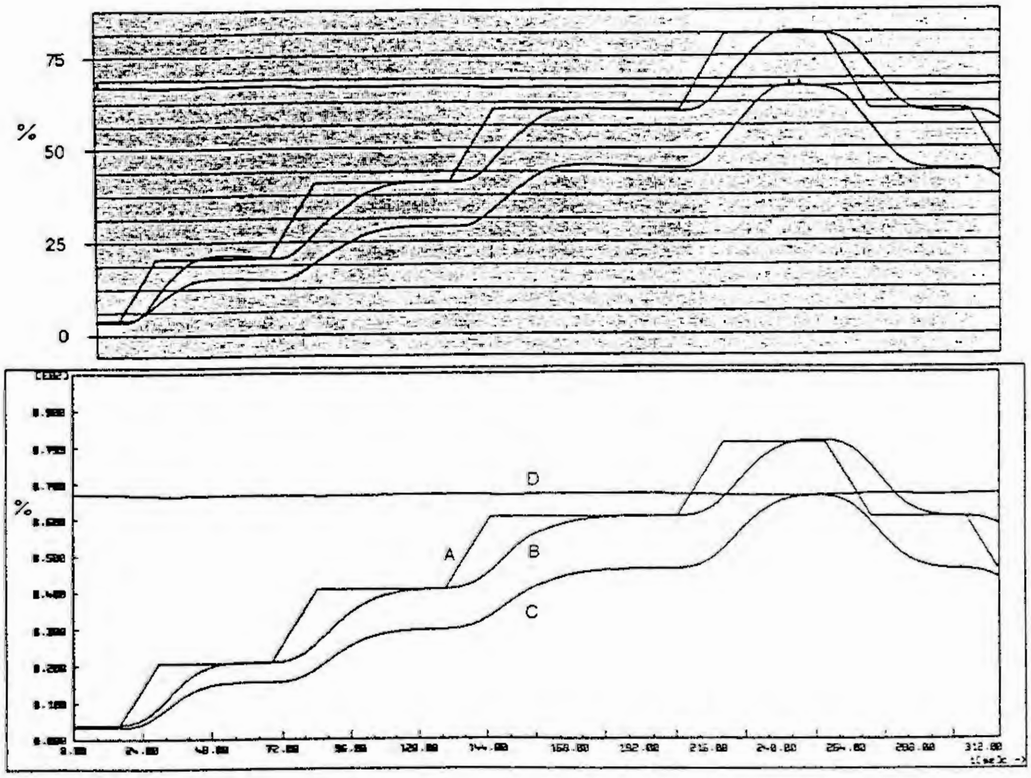


Fig.6. Power control at interconnected grid operation. Above: Recorded in the plant. Below: Result of simulation. A: Power setpoint. B: Actual power output. C: Nozzle opening. D: Pressure at turbine inlet. Changes of power setpoint: 2 - 5 - 10 - 20 - 15 MW.

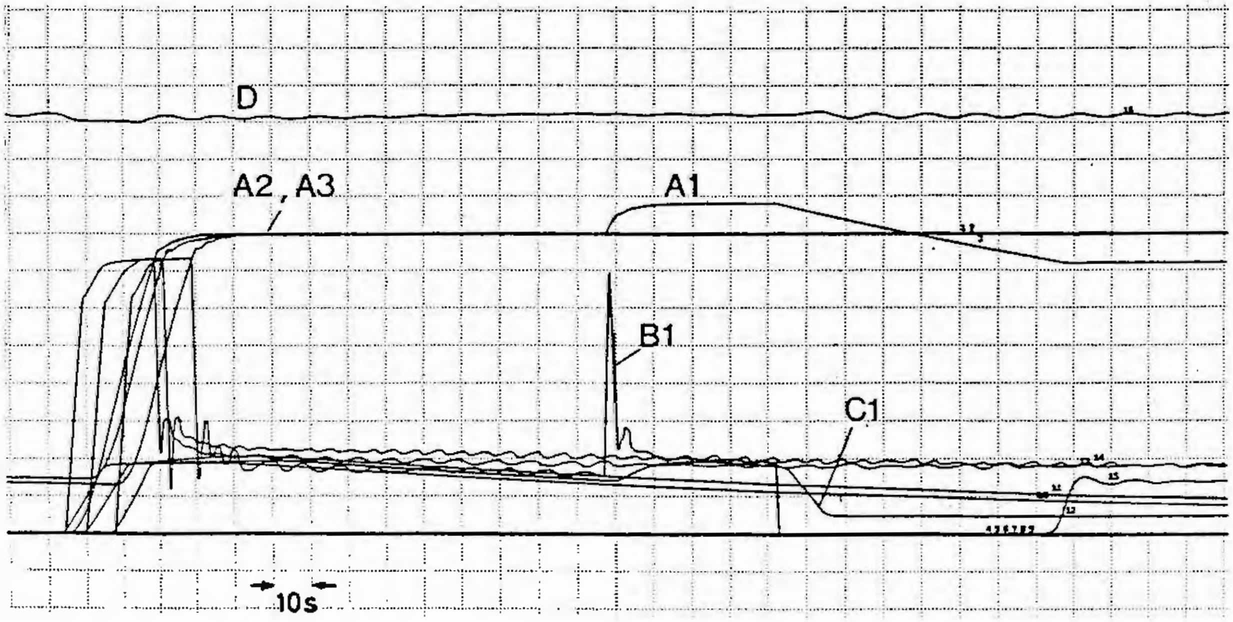


Fig.7. Start-up procedure and no-load performance of three turbine-generator sets. Recorded in the plant. A1,A2,A3 Running speeds of turbines 1,2,3. B1 Deflector position, C1 Nozzle opening of turbine 1. D Pressure at turbine inlet. Changes of speed setpoint of turbine 1: 750 - 825 - 675 min^{-1}.

7. CONCLUSION

Simulation as well as the application of effective controller design software proved to be powerful tools. Modelbuilding, model approximation and order reduction respectively were the bases to successfully solve the described real world problem of control engineering.

8. ACKNOWLEDGEMENT

I am very grateful to Dr.-Ing. H.G.Gehre for his effort in programming and executing all simulation runs as well as for many fruitful discussions.

9. REFERENCES

[1] Fasol, K.H. and Pohl, G.M., Simulation, Controller Design and Field Tests for a Hydropower Plant - A Case Study. Automatica, 26 (1990), 475-484.

[2] Fasol, K.H. and Gehre, H.G., Order Reduction, Model Approximation, and Controller Design. A survey on some known methods and recommendation of a new approach. Syst. Anal. Model.Simul. 8 (1991), 485-505.

[3] Gebhardt, B., Die blockorientierte Simulationssprache FSIMUL. In: Fasol, K.H. and Diekmann, K. (Ed.), Simulation in der Regelungstechnik. Springer-Verlag, Berlin, Heidelberg, etc., 1990, 249-268.

[4] Raabe, J., Hydro Power. VDI-Verlag, Düsseldorf, 1985.

[5] Wylie, E.B. and Streeter, V.L., Fluid Transients. McGraw-Hill, New York, 1978.

Comparison of some order reduction methods by application on high order models of technical processes

Dipl.-Ing. Carsten Jörns, Prof. Dr.-Ing. Lothar Litz
Universität Kaiserslautern, Lehrstuhl für Automatisierungstechnik
Postfach 3049, D - 67653 Kaiserslautern

Abstract. Modelling of large and complex technical processes often results in very high order models (e.g. 50 ... 100). In the case of automatic control or simulation design, problems may arise due to the fact that either methods do not work or they become very time consuming. For this reason, an engineer may be supported by order reduction methods. In this paper practical aspects for selecting an applicable order reduction method are investigated.

1. INTRODUCTION

With the need for using an order reduction method a selection process is started. Several order reduction methods have been presented by various authors with reference to the mathematical criterion used (for example [1] with 143 references). Their general properties have also been described according to the application on special problems.

Consider the "normal application engineer", who is to select one method for a special problem. He / she is no expert, nor is there time to get all the information concerning the methods. The selection process is determined by a large variety of "practical" criteria instead:

- What is the reason for the approximation (control design / simulation)?
- What kind of quality criterion / weighting is important?
- Which methods are available in present tools?
- Is the method transparent / are there transparent parameters to adjust the reduction?
- Is it really appropriate to handle linear MIMO-models of high order (if necessary)?
- Is the tool easy to handle with respect to interactive dialogue, etc. ?

In this paper a guideline is given how to procede during the search for an applicable order reduction method concerning the points mentioned above.

2. GENERAL PROBLEMS

2.1. The plant model

Order reduction problems arise if the modelling of a technical process results in high order models. This might occur due to a very detailled modelling or the complexity of the plant. For both cases order reduction is applied to improve the handling of the system with respect to automatic control design or simulation. Both targets result in different demands for order reduction methods.

Any consideration of simplifying the model is to start with the general properties of the plant and its mathematical model. By analyzing these charactersitics, a first clue for the selection of an order reduction method type is given. Characteristical properties like stability/instability and aperiodic/ oscillatory behaviour may have an influence on the choice of the weighting criterion or might even permit the general use of a special order reduction method. This will be shown in the examples.

Two high order models of stable technical processes were available for a critical examination: a metal strip processing line [2] with a linear modell of 41st order and a linearized model of a residential home with floor heating of 15 rooms being of 74th order. In both cases the purpose for the reduction was a model order just high enough to include all states with physical meaning. The systems are very different in their dynamic behaviour and their characteristics are related to a variety of plants.

2.2. Hardware / software

After the plant model is given, it has to be transformed in a linear state-space-representation, so that order reduction methods may be used. Therefore, a computer tool is helpful due to the large number of states. These tools are more or less well known by its users and consist of both hard- and software. It is obvious that the capabilities supported by both components have a major influence on how often tools are used for solving a certain kind of problem. A demand is, that the order reduction methods to be used in the described situation should be available in such a tool.

Two very different tools have been investigated for this study on two different hardware platforms:

- The package PILAR by the University of Karlsruhe on an IBM-compatible PC under MS DOS[5].
- The $\mathrm{MATRIX_X}$-package by Integrated Systems Inc. on an IBM RISC-workstation [6].

2.3. Handling of available methods

With a given plant model and hard-/software-packages there are still degrees of freedom left for the final choice of the order reduction method. In most cases it depends on how often a software package is used, until it gets useful. I.e. a software product that is very complex is - in most cases - hard to handle. Its advantage in solving problems gets bigger the more often it is applied to a certain kind of problem. Since order reduction is not often used by the application engineer, the effort for a single reduction run is a decisive factor. This effort is determined by the general user support by the tool like mouse support, interactive dialogue and the special properties of the order reduction module.

To include very different kinds of tools, the two abovementioned ones have been chosen on very different platforms. An interactive, MS-DOS based module shell like PILAR can hardly be compared with a command-line program like MATRIXx on a RISC-machine. But a decision is not only determined by hardware features. Especially in the case of very high order models it is important to ask for a comfortable system editor, i.e. changing of special matrix-elements, viewing and changing of submatrices etc. On the other hand the use of the order reduction module should not result in a permanent look-up in the user's manual. There should be either interactive dialogue or good online-help with respect to the demanded parameters or upcoming errors.

3. CHOICE OF THE METHOD

After a brief description of the general factors and the technical components is given in the earlier sections, the criteria are applied to the examplary problems. In this section it is pointed out, what a selection process for an order reduction method might be like in practice.

3.1. Choice by weighting criterion

In the case of the metal strip processing line, the system behaviour is oscillatory, but stable. Therefore order reduction methods with frequency weighted criteria and modal order reduction methods are of major interest. The Safonov/Chiang-method [3] in MATRIXx is of the first type, where the additive error reduction focuses on errors of the form $\|G(j\omega) - G_r(j\omega)\|_\infty$ with G the originally given transfer function, or model, and G_r is the reduced one. An example for the second type is the method by Litz [2], where the dominant modes are determined by dominance measures and kept in the reduced system together with an optimal approximation of the nondominant ones. For the second plant, the characteristic is aperiodic. In addition to the first example, errors in the step responses could be taken into account here, like the method by Eitelberg [4] that is minimizing the equation-error.

3.2. Available methods

According to 2.2. present tools were inspected for applicable methods. The two abovementioned tools included different modules for model order reduction. The package PILAR included one module for the Litz- and Eitelberg- method respectively. In MATRIXx very similar methods were found based on the additional error and multiplicative error methods. But as one of the demands was the transparency of the modules, the module REDSCHUR has been selected, because there is one input parameter, the order of the reduced system. Another advantage of this method is the possibility to keep the physical meaning of the states by not balancing the system, only determining the output-matrix $\underline{C}$ of the original. The reduced systems of all applied methods may be compared therefore.

Both tools included a matrix/system editor with different properties. For the typical application case, the engineer would probably first select the tool he/she knows best and then take a look at the methods that are part of the tool. This might be the wrong decision, especially if there is no tool with properties demanded in one of the other claims. It is more important to find a compromise between all the criteria.

3.3. Finding the compromise

Suppose there's still a choice. In MATRIXx the methods are very similar with respect to the reduction properties. It is a question of time and work worth spending for a highly sophisticated adjusting of the several module parameters. In the mostly common case the easiest method will be chosen if the user is no expert. Therefore the REDSCHUR-module is first choice because of only few precognition for the first reduction is necessary. If a more sophisticated knowledge is supported by the reduction-module user, this choice might have been wrong. All of the modules used in MATRIXx are based on either balanced truncation or multiplicative error-minimization method and therefore are mathematically demanding (including singular value decomposition, spectral factorisation etc.) with respect to the derivation. Access to the basic steps is not easy for a normal user. Even the user's manual is only a guideline for a long search through other references. The online-help also is no help by itself, it may be concerned as a reminder of what kind of parameters have to be inserted in the module call instead, no other information is supplied in there.

PILAR includes two methods in the modules RMG (Eitelberg) and RMO (Litz). Both methods are easy to handle in their basic version and may be customized for highly sophisticated application by an expert. The online-help is very close to the user's manual, describing the basic algorithm and being a guideline throughout the module. This facilitates the first use of the modules and yields results of high transparency. Every step during the reduction is briefly described. But there are disadvantages as well. PILAR is about ten years old, based on a DOS-Shell. Every step of a further order reduction run has to be inserted on and on again. A system editor is supported, but simulation facilities are not quite flexible. There is an option for data interchange with MATLAB (incorporating similar modules like MATRIXx), but it is no substitute to what is demanded here.

4. REDUCTION RUNS

Reduction runs have been carried out with the abovementioned examples and tools to compare the results. According to the first sections, the MATRIXx-package supported a very flexible and fast system editor. Although PILAR's editor is good as well, it is not as flexible as its competitor. Especially for very high order systems, this is a very important factor. Flexibility is also much better in MATRIXx regarding the simulation facilities.

Suppose the plant model is given in each of the tools. The reduction process is more important now. In PILAR, each module is requesting the input parameters. Errors are avoided, because each input value is checked before process is continuing. This takes a little while, but modules are used without major problems. To forget one module while progressing is the only mistake that could happen during the preparation for the RMO-module. Normally, this shouldn't occur as the users's manual as well as the online-help point out every single step including the mathematical derivation. A major problem is the MS DOS-based memory management of PILAR. So the Litz-method could not be applied to the 74th order model because of memory problems. In the other cases the use of PILAR was easy, and the behaviour of the reduced systems was according to the method's properties. Both methods resulted in stable and stationary exact systems with nonminimum phase behaviour for the aperiodic system due to a large reduction horizon.

MATRIXx's reduction module is command line based. To start one module, only one command line is to be typed and results are the left hand side of a function call. Therefore, many kind of mistakes are possible using this modules. The easiest module has been chosen and applied to the examples to avoid problems caused by the wrong use of parameters. All methods are minimizing a frequency error resulting in a non stationary exact model for both examples. For the reduced simulation model of the domestic heating system, a special system output matrix $\underline{C}$ had to be used, to approximate special states. To adjust the other modules by using parameters, this is only possible by reading many chapters in the user's manual, presuming a large mathematical background. The great advantage is the fast acting module, but with only little transparency.

The following table briefly summarizes the results of the operation tests with the selected methods and tools. The most important criteria for the choice are shown. The results for the methods are according to their application on the examples.

	properties of the method			properties of the tool				
method / tool	transpa-rency	stationary exactness	transient	simulation facilities	system editor	(Online-) Help	effort for first run	effort f. next run
Eitelberg / PILAR	+	++	-	O	+	+	+	-
Litz / PILAR	+	++	++	O	+	+	O	-
Safonov-Chiang / MATRIXx	-	-	+	++	++	-	-	+

5. RESULTS

Summarizing the experience of this study, a short list of claims for a user-oriented model order reduction tool results:

- Several order reduction methods with **different** characteristics should be implemented in one tool. (During the studies, the Litz-method was implemented in MATRIXx by the authors for a comparison of the reduction properties with respect to the 74th order model)
- The tool is to take care of the user's demands. It is to include a good system editor, simulation module and many more useful modules for the application engineer
- No highly sophisticated method can be applied to a certain problem by a non-expert. The normal application case is time-limited, so that only the basic algorithms will be applied to the problems. A support for the user is necessary to raise the transparency of the reduction results, i.e. show the way the reduction method is proceeding.
- Easy model order reduction application tools will help spreading the use of order reduction to solve problems. This can only be achieved by yielding the first claims.

6. REFERENCES

[1] Troch, I., Müller, P.C., Fasol, K.-H., Model reduction for simulation and controler design. In: at - Automatisierungstechnik 40(1992), No. 2 - 4, 45-53, 93-99, 132-141

[2] Litz, L., Order reduction of linear state-space models via optimal approximation of the nondominant modes, In: Large Scale Systems 2(1981), 171-184

[3] Safonov, M. G., Chiang, R. Y., Model reduction for robust control: a Schur relative-error method. In: Proc. Amer. Cont. Conf., 1988, 1685-1690.

[4] Eitelberg, E., Modellreduktion durch Minimieren des Gleichungsfehlers. In: Regelungstechnik 10(1978), 320-322.

[5] Integrated Systems Inc., MATRIXx Model Reduction Module. Integrated Systems Inc.; Santa Clara, Ca., 1991.

[6] Föllinger, O, Benutzerhandbuch zum Programmsystem PILAR (Programmodule zur Interaktiven Lösung von Aufgabenstellungen der Regelungstechnik). Universität Karlsruhe, 1990.

SOFTWARE FOR THE COMPARISON AND APPLICATION OF MODEL REDUCTION TECHNIQUES

D P Atherton and D Xue
School of Engineering
University of Sussex
Falmer, Brighton
East Sussex, BN1 9QT, UK

Abstract: The paper describes a menu driven MATLAB program for model reduction. The software contains several model reduction algorithms including one based on an optimisation technique which can be used for original systems or reduced order models with time delay. Some examples of applications of the software are given in the paper.

1. INTRODUCTION

Model reduction techniques have attracted much attention in the last two decades and good reviews can be found in Decoster and van Cauwenberghe (1), Mahmoud and Singh (2), and Bultheel and van Barel (3). Model reduction finds many applications in control engineering, some examples of which are the reduction of a plant transfer function in order that a control design technique can be used which requires a lower order plant model and the reduction of a high order controller transfer function, for example obtained using the H infinity design approach, to a simpler lower order model for practical implementation. Many analytical techniques have been presented in the literature and without embedding them in software it is very difficult to compare their advantages and disadvantages. A software package is described in this paper which has been developed to perform model reduction by several methods and to compare the results. It has almost become a 'de facto' standard to compare the quality of model reduction by comparing only the step responses of the original system model and the reduced order model. In the software, however, one can compare the response of the original system and the reduced order model to a variety of input signals and, also, probably more importantly, for feedback control system design, can compare their frequency responses.

In addition to the classical methods of model reduction, the software also includes a routine for model reduction by optimisation where the parameters of the reduced order model of a selected order are optimised so that the difference between the output from the original system model and the reduced order model can be minimised according to a variety of criteria for a user specified input. An advantage of the optimisation approach is that it can allow for either the original model, or the reduced order model, or both, to incorporate a time delay. The analysis when a time delay exists is done using a Padé approximation for the time delay with an order which may be specified by the user. The best reduced order model for any application is that reduced model which matches as closely as possibly the input/output behaviour of the original system under the actual operating conditions. The optimisation approach enables one to examine this facet much more easily. In the next section the software, which has been written in MATLAB, is described and this is followed by a section giving some examples of its application. Finally a brief conclusion is presented.

2. DESCRIPTION OF THE SOFTWARE

The software is menu-driven so that it can be used by a person who has no previous experience with MATLAB. The first requirement is to enter the transfer function of the original system model and this is done by responding to the first menu which is:

Main menu for model reduction

```
1)    Enter G(s) in transfer function form
2).   Enter G(s) in block diagram form
3)    Load G(s) in a data file
4)    Keep G(s) unchanged
0)    Quit
      Select a menu number  ⇒
```

If 1) is selected for transfer function entry, then the user is asked to enter a string written as a ratio of polyonomials in s, for example 0.2s2+s+3/(s+1)6. The power of the polynomial appears after the s and the slash divides the numerator and denominator. If the user wishes to incorporate a time delay then this is requested after the string has been entered. Once a transfer function has been entered the main menu for model reduction is again displayed with the additions:

5) display the original system
6) perform model reduction
7) comparative analysis of models.

If 6) is selected, then the available methods for model reduction are listed as indicated in the following menu:

Available Reduction Methods

1) Continued fraction expansion method
2) Ordinary Padé approximant method
3) Routh based reduction method
4) Dominant modes method
5) Balanced realisation method
6) FF - Padé approximation method
7) Optimal reduction method
8) Optimal Routh approximant method
0) Go back to the main menu
Select a menu number ⇒

After a number is selected, then some additional information may be requested in order to perform the model reduction. For instance, if 3) is selected, then the user is asked to enter the expected reduction order and, once this is done, the numerator and denominator of the reduced model are given. All reduced models are then stored, so that at a later time the user, when using 7) in the **main menu for model reduction**, namely, comparative analysis of models, a list is given of the available models. This may include models of different orders obtained by the same technique as well as models derived by different methods. If the optimal reduction method 7) is selected then the user is asked to enter the required order of the numerator, the required order of the denominator and then a list is given of available weighting functions, which are used in the minimisation criterion for the error between the output of the original system and the reduced model, to choose from. The available waiting functions are:

1) $F(t) = 1$: ISE criterion
2) $F(t) = t^{2N}$
3) $F(t) = \exp(-2at)$
4) $F(t) = \{1 - \exp(-at)\}^2$
5) A user defined prefilter $G_p(s)$

The input, *r(t)*, to the model and the reduced model is defined according to the parameters a_1, a_2, a_3 in the formula $r(t) = a_1 u(t) + a_2 m(t)$ where $u(t)$ is a unit step and $m(t) = e^{-t/a_3}$. The objective function, *J*, of the error criterion is $J = \min_\theta \left[\int_o^\infty F(t) e^2(t,\theta) dt \right]$ where $e(t,\theta)$ is the error between the output from the original system model and the reduced order model as shown in Fig. 1, and θ is the parameter vector of the reduced order model. The initial guess for θ may be provided by the user or if not the program will use those of a Padé approximation. The integral is evaluated in the s domain which is possible for all the weighting functions given [4].

Frequency responses and time responses to various inputs of the reduced models as well as the original system, and errors between these responses, can be obtained by responding appropriately to the menu options provided.

3. SOME EXAMPLE APPLICATIONS

Two examples are given to illustrate usage of the program and the facilities provided.

Example 1 The original system is assumed to have a transfer function $G(s) = 1/(s+1)^6$. Four reduced order models are obtained, namely third order continued fraction, Routh, balanced realisation and a single time constant plus time delay obtained using the ISE criterion for a step input. As is usually the case the balanced realisation is not a strictly proper transfer function being of order three over three, whereas the continued fraction and Routh are of order two over three. Figs 2 and 3 show respectively the step responses and frequency responses on a Nyquist plot for the original model and the reduced order ones. It can be seen that on both figures the 'best' reduced models are the continued fraction and balance realisation ones. The errors, which are shown in Fig. 4 for the step responses of these two reduced models, are only significant for small values of time and correspondingly at high frequencies on the frequency response plots.

Example 2 The original system is assumed to have the transfer function $G(s) = 10(s^2+s+1)/(s+1)^5$. The unit step responses of this original model and its second order balanced realisation reduced model, which has $G_r(s) = (0.0080s^2 - 0.2090s + 2.1102)/(s^2 + 0.8201s + 0.2110)$ are difficult to separate on a graph so the error in the balanced realisation step response is shown in Fig. 5. The error which has a maximum value of approximately 0.5% of the maximum response at time 4.3 seconds does not appear too significant. PID controllers were designed using the same method for the original system (controller 1) and for the reduced model (controller 2) and then their performance checked on control of the original plant. The resulting closed loop step responses are shown in Fig. 6 from which it can be seen that the PID controller designed using the balanced realisation reduced model performs relatively poorly on the original plant. The reason for this poor performance is due to the discrepancy between the frequency responses around the 180° phase point which is clearly seen in Fig. 7 which shows the Nyquist frequency response loci in this region. The critical gain and critical frequency, which were used to design the PID controllers as is done practically in the autotuning method, have values of 1.959 and 2.300 for the original model and 4.244 and 2.884 for the reduced model.

4. CONCLUSIONS

In this paper a menu driven MATLAB program has been described which includes several model reduction methods including one based on optimisation which can be used for original models with time delays and for any original model produce a reduced order one with or without a time delay. The software is easy to use and has many features to assess the accuracy of any reduced order model produced. Two application examples of the software have been given.

5. REFERENCES

[1] Decoster M and van Cauwenberghe A R, *A Comparative Study of Different Reduction Methods* Journal A, 17, pp 68-74, 125-134, 1976

[2] Mahmoud M S and Singh M G, *Large Scale System Modelling*, Pergamon Press, 1981

[3] Bultheel A and van Barel M, *Padé Techniques for Model Reduction in Linear System Theory: A Survey*, Journal of Computational and Applied Mathematics, 14, pp 401-438, 1986

[4] Xue D and Atherton D P, *An Optimal Model Reduction Method for Linear Systems*, ACC'91, Boston, USA, pp 2128-2129, June 1991

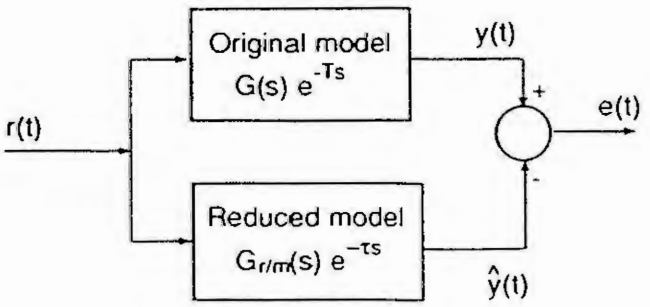


Fig. 1 Block diagram for optimal model reduction

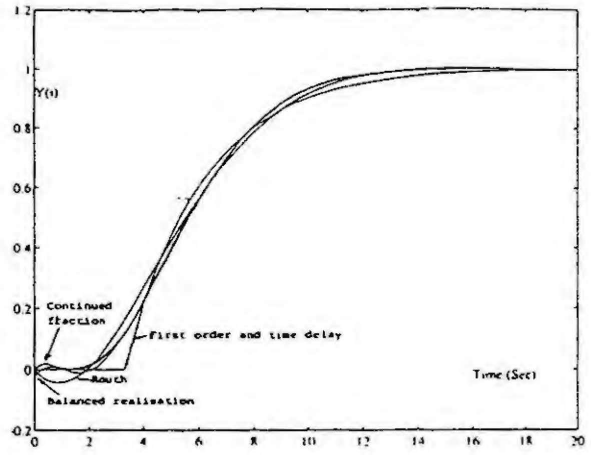


Fig. 2 Step response comparisons

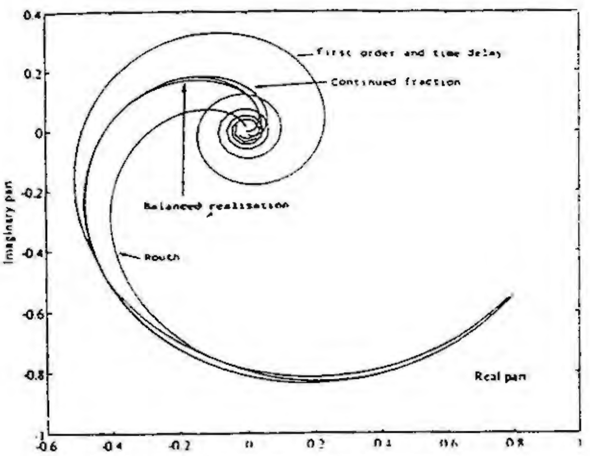


Fig. 3 Frequency response comparisons

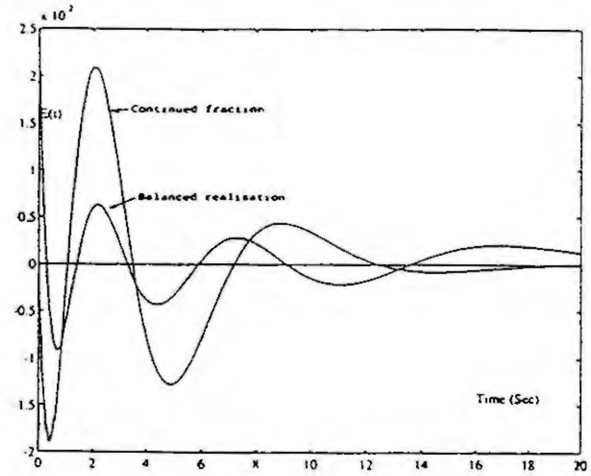


Fig. 4 Step response errors

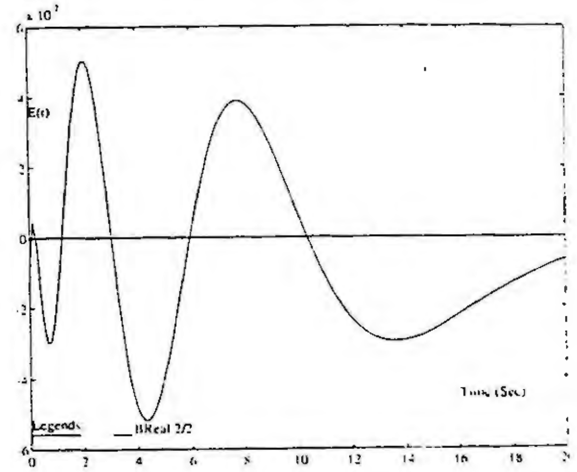


Fig. 5 Step response error

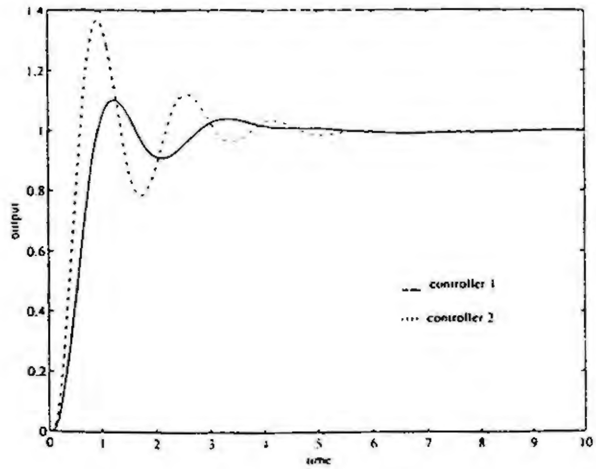


Fig. 6 Closed loop responses

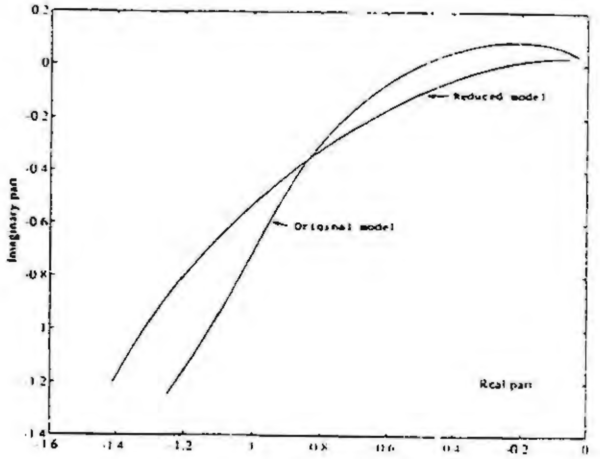


Fig. 7 Frequency response plots

NUMERICAL METHODS AND SOFTWARE TOOLS FOR MODEL REDUCTION

A. VARGA

DLR - Oberpfaffenhofen
Institute for Robotics and System Dynamics
P.O.B. 1116, D-82230 Wessling, Germany

Abstract. An overview of numerically reliable algorithms for model reduction is presented. The covered topics are the reduction of stable and unstable linear systems as well as the computational aspects of frequency weighted model reduction. The presentation of available software tools focuses on a recently developed Fortran library RASP-MODRED implementing a new generation of numerically reliable algorithms for model reduction.

1. INTRODUCTION

Model reduction is of fundamental importance in many modeling and control applications. The basic reduction algorithms discussed in this paper belong to the class of methods based on or related to balancing techniques [1, 2, 3, 4] and are primarily intended for the reduction of linear, stable, continuous- or discrete-time systems. All methods rely on guaranteed error bounds and have particular features which recommend them for use in specific applications. The basic methods combined with coprime factorization or spectral decomposition techniques can be used to reduce unstable systems [5] or to perform *frequency-weighted model reduction* (FWMR) [6, 7].

The surveyed algorithms represent the latest developments of various procedures for solving computational problems appearing in the context of model reduction. Most algorithms possess desirable attributes as generality, numerical reliability, enhanced accuracy, and thus are completely satisfactory to serve as bases for robust software implementations. Such implementations are available in a recently developed Fortran 77 library for model reduction called RASP-MODRED [8]. The implementations of routines are based on the new linear algebra standard package LAPACK [9]. It is worth mentioning that the implemented algorithms are generally superior to those implemented in the model reduction tools of commercial packages [10, 11, 12].

2. MODEL REDUCTION ALGORITHMS

Consider the n-th order original state-space model $G := (A, B, C, D)$ with the *transfer-function matrix* (TFM) $G(\lambda) = C(\lambda I - A)^{-1}B + D$, and let $G_r := (A_r, B_r, C_r, D_r)$ be an r-th order approximation of the original model ($r < n$), with the TFM $G_r = C_r(\lambda I - A_r)^{-1}B_r + D_r$. A large class of model reduction methods can be interpreted as performing a similarity transformation Z yielding

$$\left[\begin{array}{c|c} Z^{-1}AZ & Z^{-1}B \\ \hline CZ & D \end{array}\right] := \left[\begin{array}{cc|c} A_{11} & A_{12} & B_1 \\ A_{21} & A_{22} & B_2 \\ \hline C_1 & C_2 & D \end{array}\right],$$

and then defining the reduced model (A_r, B_r, C_r, D_r) as the leading diagonal system (A_{11}, B_1, C_1, D). When writing $Z := [T\ U]$ and $Z^{-1} := [L^T\ V^T]^T$, then $\Pi = TL$ is a projector on T along L and $LT = I_r$. Thus the reduced system is $(A_r, B_r, C_r, D_r) = (LAT,\ LB,\ CT,\ D)$. Partitioned forms as above can be used to construct a so-called *singular perturbation approximation* (SPA). The matrices of the reduced model in this case are given by

$$\begin{aligned} A_r &= A_{11} + A_{12}(\gamma I - A_{22})^{-1}A_{21}, \\ B_r &= B_1 + A_{12}(\gamma I - A_{22})^{-1}B_2, \\ C_r &= C_1 + C_2(\gamma I - A_{22})^{-1}A_{21}, \\ D_r &= D + C_2(\gamma I - A_{22})^{-1}B_2. \end{aligned}$$

where $\gamma = 0$ for a continuous-time system and $\gamma = 1$ for a discrete-time system. Note that SPAs preserve the DC-gains of stable original systems.

Specific requirements for model reduction algorithms are formulated and discussed in [13]. Such requirements are: (1) applicability of methods regardless the original system is minimal or not; (2) emphasis on enhancing the numerical accuracy of computations; (3) relying on numerically reliable procedures.

The first requirement can be fulfilled by computing L and T directly, without determining Z or Z^{-1}. In particular, if the original system is not minimal, then L and T can be chosen to compute an *exact* minimal realization of the original system [14].

The emphasis on improving the accuracy of computations led to so-called algorithms with *enhanced accuracy*. In many model reduction methods, the matrices L and T are determined from two positive semi-definite matrices P and Q, called generically *gramians*. The gramians can be always determined in Cholesky factorized forms $P = S^TS$ and $Q = R^TR$, where S and R are upper-triangular matrices. The computation of L and T can be done by computing the *singular value decomposition* (SVD)

$$SR^T = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \mathrm{diag}(\Sigma_1, \Sigma_2) \begin{bmatrix} V_1 & V_2 \end{bmatrix}^T$$

where

$$\Sigma_1 = \mathrm{diag}(\sigma_1, \ldots, \sigma_r), \quad \Sigma_2 = \mathrm{diag}(\sigma_{r+1}, \ldots, \sigma_n),$$

and $\sigma_1 \geq \ldots \geq \sigma_r > \sigma_{r+1} \geq \ldots \geq \sigma_n \geq 0$.

The so-called *square-root* (**SR**) methods determine L and T as [15]

$$L = \Sigma_1^{-1/2} V_1^T R, \qquad T = S^T U_1 \Sigma_1^{-1/2}.$$

If r is the order of a minimal realization of G then the gramians corresponding to the resulting realization are diagonal and equal. In this case the minimal realization is called *balanced*. The **SR** approach is usually very accurate for well-equilibrated systems. However if the original system is highly unbalanced, potential accuracy losses can be induced in the reduced model if either L or T is ill-conditioned.

In order to avoid ill-conditioned projections, a *balancing-free* (**BF**) approach has been proposed in [16] in which always well-conditioned matrices L and T can be determined. These matrices are computed from orthogonal matrices whose columns span orthogonal bases for the right and left eigenspaces of the product PQ corresponding to the first r largest eigenvalues $\sigma_1^2, \ldots, \sigma_r^2$. Because of the need to compute explicitly P and Q as well as their product, this approach is usually less accurate for moderately ill-balanced systems than the **SR** approach.

A *balancing-free square-root* (**BFSR**) algorithm which combines the advantages of the **BF** and **SR** approaches has been introduced in [14]. L and T are determined as

$$L = (Y^TX)^{-1}Y^T, \qquad T = X,$$

where X and Y are $n \times r$ matrices with orthogonal columns computed from the QR decompositions $S^TU_1 = XW$ and $R^TV_1 = YZ$, while W and Z are non-singular upper-triangular matrices. The accuracy of the **BFSR** algorithm is usually better than either of **SR** or **BF** approaches.

The SPA formulas can be used directly on a balanced minimal order realization of the original system computed with the **SR** method. A **BFSR** method to compute SPAs has been proposed in [17]. The matrices L and T are computed such that the system (LAT, LB, CT, D) is minimal and the product of corresponding gramians has a block-diagonal structure which allows the application of the SPA formulas.

Provided the Cholesky factors R and S are known, the computation of matrices L and T can be done by using exclusively numerically stable algorithms. Even the computation of the necessary SVD can be done without forming the product SR^T. Thus the effectiveness of the **SR** or **BFSR** techniques depends entirely on the accuracy of the computed Cholesky factors of the gramians. In the following sections we discuss the computation of these factors for several concrete model reduction techniques.

3. ALGORITHMS FOR STABLE SYSTEMS

In the *balance & truncate* (B&T) method [1] P and Q are the controllability and observability gramians satisfying a pair of continuous- or discrete-time Lyapunov equations

$$AP + PA^T + BB^T = 0, \quad A^TQ + QA + C^TC = 0;$$

$$APA^T + BB^T = P, \quad A^TQA + C^TC = Q.$$

These equations can be solved directly for the Cholesky factors of the gramians by using numerically reliable algorithms proposed in [18]. The **BFSR** version of the B&T method is described in [14]. Its **SR** version [15] can be used to compute balanced minimal representations. Such representations are also useful for computing reduced order models by using the SPA formulas [2] or the *Hankel-norm approximation* (HNA) method [4]. A **BFSR** version of the SPAs method is described in [17]. Note that the B&T, SPA and HNA methods belong to the family of absolute error methods which try to minimize $\|\Delta_a\|_\infty$, where Δ_a is the absolute error $\Delta_a = G - G_r$.

The *balanced stochastic truncation* (BST) method [3] is a relative error method which tries to minimize $\|\Delta_r\|_\infty$, where Δ_r is the relative error defined implicitly by $G_r = (I - \Delta_r)G$. In the BST method the gramian Q satisfies a Riccati equation, while the gramian P still satisfies a Lyapunov equation. Although the determination with high accuracy of the Cholesky factor of Q is computationally involved, it is however necessary to guarantee the effectiveness of the **BFSR** approach. Iterative refinement techniques are described for this purpose in [13].

Both the SR and SRBF versions of the B&T, SPA and BST algorithms are implemented in the RASP-MODRED library. The implementation of the HNA method uses the **SR** version of the B&T method to compute a balanced minimal realization of the original system. All implemented routines are applicable to both continuous- and discrete-time systems. It is worth mentioning that implementations provided in commercial software [10, 11, 12] are only for continuous-time systems.

4. REDUCTION OF UNSTABLE SYSTEMS

The reduction of unstable systems can be performed by using the methods for stable systems in conjunction with two imbedding techniques. The first approach consists in reducing only the stable projection of G and then including the unstable projection unmodified in the resulting reduced model. The second approach is based on computing a stable *rational coprime factorization* (RCF) of G say in the form $G = M^{-1}N$, where M, N are stable and proper rational TFMs, and then to reduce the stable system $[\,N\ M\,]$. From the resulting reduced model $[\,N_r\ M_r\,]$ we obtain $G_r = M_r^{-1}N_r$.

The coprime factorization approach used in conjunction with the B&T or BST methods fits in the general projection formulation introduced in Section 2. The gramians necessary to compute the projection are the gramians of the system $[\,N\ M\,]$. The computed matrices L and T by using either the **SR** or **BFSR** methods can be directly applied to the matrices of the original system. The main computational problem is how to compute the RCF to allow a smooth and efficient imbedding which prevents computational overheads. Two factorization algorithms proposed recently compute particular RCFs which fulfill these aims: the RCF with prescribed stability degree [19] and the RCF with inner denominator [20]. Both are based on a numerically reliable Schur technique for pole assignment. The use of other RCFs is presently under consideration.

RASP-MODRED provides all necessary tools to perform the reduction of unstable system. Routines are provided to compute left/right RCFs with prescribed stability degree or with inner denominators, to compute additive spectral decompositions, or to perform the back transformations. A modular implementation allows arbitrary combinations between various factorization and model reduction methods.

5. ALGORITHMS FOR FWMR

The FWMR methods try to minimize a weighted error of the form $\|W_1(G - G_r)W_2\|_\infty$, where W_1 and W_2 are suitable weighting TFMs. Many controller reduction problems can be formulated as FWMR problems [21]. Two basic approaches can be used to solve such problems. The approach proposed in [7] can be easily imbedded in the general formulation of Section 2. Provided G and the weights W_1 and W_2 are all stable TFMs, then P and Q are the frequency-weighted controllability and observability gramians of GW_2 and W_1G, respectively (for details see [21]). Unfortunately no proof of stability of the two-sided weighted approximation exists unless either $W_1 = I$ or $W_2 = I$.

In the second approach we assume that G is stable and W_1, W_2 are invertible, having only unstable poles and zeros. The technique proposed in [6] to solve the FWMR problem computes first G_1 the n-th order stable projection of $W_1 G W_2$ and then computes the r-th order approximation G_{1r} of G_1 by using one of methods for stable systems. Finally G_r results as the r-th order stable projection of $W_1^{-1} G_{1r} W_2^{-1}$.

RASP-MODRED provides all necessary tools to perform FWMR. Special routines based on algorithms proposed in [22] are provided to compute efficiently the stable projections for the second approach.

6. THE RASP-MODRED LIBRARY

RASP-MODRED is one of the first numerical libraries developed by using the new linear algebra package LAPACK [9]. The library provides a rich set of computational facilities for model reduction. Besides the already mentioned functions, routines to evaluate Hankel- and L^2-norms of TFMs, to perform bilinear transformations, to compute systems couplings, are also available. Many lower level computational routines can have a special importance for other applications areas. In its present state of development the library consists of 77 routines and is continuously extended. Routines for alternative FWMR methods, for computing normalized RCF, or for evaluation of L_∞-norm are presently under development.

The implementation of the library has been done in accordance with the newly established RASP/SLICOT mutual compatibility concept [23]. Thus the implemented routines belong simultaneously to both RASP [24] and SLICOT [25] libraries. This software sharing strategy is meant to save future efforts in developing both libraries.

7. CONCLUSIONS

We presented an up to date overview of numerically reliable algorithms and associated software tools for model reduction. The algorithmic richness and the complexity of the model reduction problems require efficient and robust software implementations which can exploit efficiently all structural aspects of the underlying computational problems. This is possible only in high level languages such as Fortran. In contrast, implementations in MATLAB, although much more compact than the corresponding Fortran codes, are generally less efficient with respect to both operation count and memory usage. Moreover, many MATLAB implementations, done unfortunately by people with insufficient numerical expertise, are unsatisfactory with respect to requirements as generality, numerical reliability, accuracy.

8. ACKNOWLEDGMENTS

Many of presented algorithms as well as the development of the RASP-MODRED library are the results of a two-years stay of the author at the Ruhr-University Bochum as an Alexander von Humboldt research fellow under the direction of Prof. K.H. Fasol. The author also acknowledges the support of Prof. G. Grübel (DLR-Oberpfaffenhofen). Without his support the development of the RASP-MODRED library would have been not possible.

9. REFERENCES

[1] B. C. Moore. Principal component analysis in linear system: controllability, observability and model reduction. *IEEE Trans. Autom. Contr.*, 26:17–32, 1981.

[2] Y. Liu and B. D. O. Anderson. Singular perturbation approximation of balanced systems. *Int. J. Control*, 50:1379–1405, 1989.

[3] U. B. Desai and D. Pal. A transformation approach to stochastic model reduction. *IEEE Trans. Autom. Contr.*, 29:1097–1100, 1984.

[4] K. Glover. All optimal Hankel-norm approximations of linear multivariable systems and their L^∞-error bounds. *Int. J. Control*, 39:1115–1193, 1974.

[5] Y. Liu and B. D. O. Anderson. Controller reduction via stable factorization and balancing. *Int. J. Control*, 44:507–531, 1986.

[6] G. A. Latham and B. D. O. Anderson. Frequency-weighted optimal Hankel norm approximation of stable transfer functions. *Syst. Contr. Lett.*, 5:229–236, 1985.

[7] D. Enns. *Model Reduction for Control Systems Design.* PhD thesis, Dept. Aeronaut. Astronaut., Stanford Univ., Stanford, CA, 1984.

[8] A. Varga. *RASP Model Order Reduction Programs.* University of Bochum and DLR-Oberpfaffenhofen, TR R88-92, August 1992.

[9] E. Anderson *et al. LAPACK User's Guide.* SIAM, Philadelphia, 1992.

[10] R. Y. Chiang and M. G. Safonov. *Robust Control Toolbox 2.0.* The MathWorks Inc., 1992.

[11] G. Balas, J. Doyle, K. Glover, A. Packard, and R. Smith. *μ-Analysis and Synthesis Toolbox 1.0.* The MathWorks Inc., 1991.

[12] B.D.O Anderson and B. James. *MATRIX$_X$ Model Reduction Module.* Integrated Systems Inc., Santa Clara, CA, 1991.

[13] A. Varga and K. H. Fasol. A new square-root balancing-free stochastic truncation model reduction algorithm. *Prepr. 12th IFAC World Congress, Sydney,* vol. 7, pp. 153–156, 1993.

[14] A. Varga. Efficient minimal realization procedure based on balancing. In A. El Moudni, P. Borne, and S. G. Tzafestas, editors, *Prepr. of IMACS Symp. on Modelling and Control of Technological Systems,* vol. 2, pp. 42–47, 1991.

[15] M. S. Tombs and I. Postlethwaite. Truncated balanced realization of a stable non-minimal state-space system. *Int. J. Control,* 46:1319–1330, 1987.

[16] M. G. Safonov and R. Y. Chiang. A Schur method for balanced-truncation model reduction. *IEEE Trans. Autom. Contr.,* 34:729–733, 1989.

[17] A. Varga. Balancing-free square-root algorithm for computing singular perturbation approximations. *Proc. 30th IEEE CDC, Brighton, UK,* pp. 1062–1065, 1991.

[18] S. J. Hammarling. Numerical solution of the stable, non-negative definite lyapunov equation. *IMA J. Numer. Anal.,* 2:303–323, 1982.

[19] A. Varga. Coprime factors model reduction based on accuracy enhancing techniques. *Syst. Anal. Model. Sim.,* 11:303–311, 1993.

[20] A. Varga. A Schur method for computing coprime factorizations with inner denominators and applications in model reduction. *Proc. 1993 ACC, San Francisco, CA,* pp. 2130–2131, 1993.

[21] B. D. O. Anderson and Y. Liu. Controller reduction: concepts and approaches. *IEEE Trans. Autom. Contr.,* 34:802–812, 1989.

[22] A. Varga. Explicit formulas for an efficient implementation of the frequency-weighted model reduction approach. *Proc. 1993 ECC, Groningen, NL,* pp. 693–696, 1993.

[23] G. Grübel, A. Varga, A. van den Boom, and A. J. Geurts. Towards a coordinated development of numerical CACSD software: the RASP/SLICOT compatibility concept. *Prepr. CACSD'94, Tucson,* 1993.

[24] G. Grübel and H.-D. Joos. RASP and RSYST - two complementary program libraries for concurrent control engineering. In *Prepr. 5th IFAC/IMACS Symp. CADCS'91, Swansea, UK,* pp 101–106. Pergamon Press, 1991.

[25] A. van den Boom *et al.* SLICOT, a subroutine library in control and systems theory. *Prepr. 5th IFAC/IMACS Symp. CADCS'91, Swansea, UK,* pp. 89–94. Pergamon Press, 1991.

A PRACTICAL EXERCISE IN SIMULATION MODEL VALIDATION

D.J. Murray-Smith and Mingrui Gong
Department of Electronics and Electrical Engineering
University of Glasgow
Glasgow G12 8QQ
Scotland

Abstract. The validation of continuous system simulation models is an important topic which often receives insufficient attention in courses on system modelling. The paper describes a practical exercise, using a laboratory-scale system involving two inter-connected tanks of liquid, which can serve to introduce students to validation concepts in a practical fashion.

1. INTRODUCTION

The validation of mathematical models and of associated computer simulation programs is a very important aspect of the process of developing usable simulation models. This fact is recognised in the introductory chapters of many textbooks on mathematical modelling and continuous system simulation. However, this recognition of validation as a central part of the modelling process seldom extends to a more detailed treatment of specific methods of validation in later sections of these texts. Students are, unfortunately, often left with only a vague indication of the steps to be taken in trying to establish the adequacy, or otherwise, of a model which they are developing or intend to use in a given application.

Terminology presents immediate difficulties in discussing questions of model credibility and validation methodology but useful guidelines have been provided by the Technical Committee on Model Credibility of the Society for Computer Simulation [2]. Some aspects of these guidelines are particularly important, especially the recommendation that a strong distinction should be drawn between 'verification' and 'validation'. 'Verification' may be defined as the task of proving that a computer-based description is consistent with the underlying mathematical model to a specified accuracy level, while 'validation' is concerned with questions of the adequacy of the mathematical model. In order to avoid confusion it has been proposed that the terminology suggested by the SCS Technical Committee be extended by using the adjectives 'internal' and 'external', respectively, when discussing verification and validation [1]. The practical exercise described here involves both internal verification of a simulation program and external validation of the underlying mathematical model and should help to emphasise the differences between these activities, both of which are of great practical importance.

The system providing the basis of this exercise involves bench-top equipment of a type used widely for laboratory experiments in the teaching of elementary concepts in automatic control. It involves two coupled water tanks and provides a basis for the implementation of liquid level control systems using either continuous or digital control techniques. Suitable equipment of this type is available for the educational market from a number of different sources. This type of system is inherently nonlinear but is relatively easy to model by the application of simple physical laws and principles familiar to most students of the physical sciences and engineering. Students have no difficulty in understanding how the real system works and can thus focus their attention on issues of model credibility.

2. THE TWO-TANK SYSTEM

Figure 1 shows a schematic diagram of the type of system used in the validation exercise. Coupling between the tanks is provided by a number of holes of various diameters near the base of the partition and the extent of the coupling may be adjusted through the insertion of plugs into one or more of these holes. The system is equipped with a drain tap, under manual control, and the flow rate from one of the tanks can be adjusted through this. The other tank has an inflow provided by a variable speed pump which is electrically driven. Both tanks are equipped with sensors which detect the level of liquid and provide a proportional electrical output voltage.

The mathematical modelling of a hydraulic tank with volume flow rate in of Q_{vi} and outflow Q_{v1} is based upon the fact that the rate of change of volume of liquid in the tank must be equal to the difference between the flow rate in and the flow rate out. In mathematical terms this means that, for the first tank which is of uniform cross-sectional area A_1,

$$A_1 \frac{dH_1}{dt} = Q_{vi} - Q_{v1} \qquad (1)$$

where H_1 is the height of liquid in tank 1, Q_{vi} is the input volume flow rate and Q_{v1} is the volume flow rate from tank 1 to tank 2. Similarly for tank 2 we can write

$$A_2 \frac{dH_2}{dt} = Q_{v1} - Q_{vo} \qquad (2)$$

where H_2 is the height of liquid in tank 2 and Q_{vo} is the flow rate of liquid out of tank 2. Treating the holes connecting the two tanks and the drain tap as simple orifices allows the flow rates to be related to the liquid heights by the following two equations

$$Q_{v1} = C_{d_1} a_1 \sqrt{2g(H_1 - H_2)} \qquad (3)$$

and

$$Q_{vo} = C_{d_2} a_2 \sqrt{2g(H_2 - H_3)} \qquad (4)$$

where a_1 is the cross sectional area of the orifice between the two tanks, H_2 is the cross-sectional area of the orifice representing the drain tap, H_3 is the height of the drain tap above the base of the tank and g is the gravitational constant. The discharge coefficients C_{d1} and C_{d2} are constants. Continuous monitoring of the liquid levels and of the input flow rate is possible.

It is a straightforward task to write a simulation program for the nonlinear model of the coupled tank system. Nominal parameter values corresponding to a real laboratory-scale coupled tank system [3] are as follows: $A_1=A_2=9.7*10^{-3}m^2$; $a_1=3.956*10^{-5}m^2$; $a_2=3.85*10^{-5}m^2$; $C_{d1}=0.75$; $C_{d2}=0.5$; $H_3=0.03m$; and $g=9.81ms^{-2}$.

Simple simulation experiments on this simulation model could involve investigation of

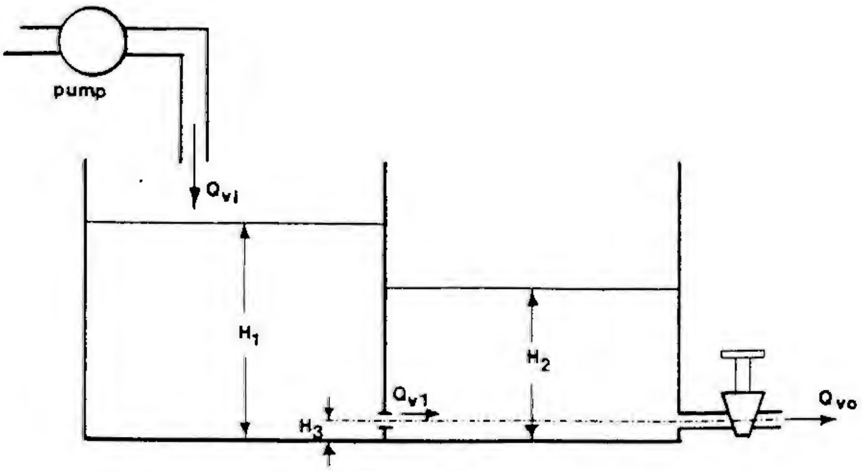


Figure 1. Schematic diagram of the two-tank system

the changes of depth H_1 and H_2 versus time for a number of different initial values for the levels and different inflow rates. It is important to consider whether results obtainable from such simulation experiments are likely to be meaningful when compared with the behaviour of the real system. Is the given mathematical model adequate and does the simulation program represent the model to a sufficient degree of accuracy? In order to answer these questions in a satisfactory way students must consider carefully how the simulation is to be used, how the computer program can be verified and how the model itself can be validated.

2.1 Internal verification of the simulation program

The first stage of verification is concerned with checking that the structure of the simulation program is consistent with the mathematical model. This involves working from the statements in the simulation program to ensure that when translated back to the form of differential equations they are the same as those of the original model. Checks must also be made of the parameter values used in the program or in the parameter input file to ensure that they correspond exactly to the parameter set of the model itself.

The second stage of verification is concerned with numerical accuracy. In the case of fixed step integration methods comparisons can be made of results obtained with a number of different sizes of integration step length and with different integration techniques. This provides the user with some understanding of the sensitivity of results to the step length and of the overall suitability of the numerical methods chosen. In the case of variable step integration algorithms tests can be carried out to compare results with different settings of the relative and absolute error limits and with different values of the minimum integration step to be allowed. Comparisons can also be made using a number of different values of the communication interval, which determines the time between output samples, to ensure that interesting events in the simulation model responses are not being hidden from the user simply because of an inappropriate choice of this parameter.

2.2 External validation of the simulation model

There is no single approach to the checking of a mathematical model which can provide a basis for definitive statements about the overall validity of that model. Statements about model validity must be made in the context of an intended application. For this modelling exercise the computer simulation is to be used subsequently as a basis for the design of an automatic control system which will ensure that a given level is maintained in one of the tanks. There is particular interest therefore in the accuracy of the model in predicting steady state conditions and in predicting the form of small transients about any given steady operating point.

Agreement between steady-state measurements and steady-state model predictions is generally quite good for most parts of the operating range,but typical results with the nominal parameter set suggest immediately that the model is not perfect. Dynamic tests can also show significant differences between the simulation model predictions and the behaviour of the real system. Differences between the steady-state liquid levels in the simulation model and in the real system, for a given value of input flow rate, are found to vary slightly with H_1 and H_2 due to the limitations of Equations 3 and 4 in describing the relationships between output flow and the liquid level in each tank. These equations apply to an ideal simple orifice and the actual physical effects at the tank outlets are unlikely to correspond exactly this empirical model.

Such findings provide an opportunity for students to attempt to improve the model. This may involve obtaining better estimates for the discharge coefficients C_{d1} and C_{d2} from further steady-state tests or dynamic tests on the real system. Parameter estimation and system identification methods are, of course, of general importance in the external validation of dynamic models and this provides an opportunity to introduce students to the idea of using such tools in a validation context. Although it is not necessary to use any advanced concepts of system identification in this application, certain important principles can be illustrated, such as the need to carry out tests to estimate model parameters separately from the tests used to generate system response data against which the predictive properties of the resulting simulation model are to be compared.

3. DISCUSSION

This exercise provides a practical and open-ended introduction to the concepts of internal verification and external validation of simulation models using a relatively simple nonlinear system. The simulation model is easily implemented and the relatively simple nature of the system and the variables which are accessible for measurement in the real hardware make this an interesting but straightforward system for the application of external validation methods.

4. ACKNOWLEDGEMENT

Mingrui Gong acknowledges support from the U.K. Science and Engineering Research Council. Experimental work carried out as part of her current programme of research has contributed to the development of this teaching exercise on model validation.

5. REFERENCES

[1] Murray-Smith, D.J., A review of methods for the validation of continuous system simulation models. In: Nock, K. (Ed.), Proceedings 1990 UKSC Conference on Computer Simulation, UKSC, Burgess Hill, 1990.

[2] SCS Technical Committee on Model Credibility, Terminology for model credibility, Simulation, 32, 103-104, 1979.

[3] Wellstead, P.E., Coupled Tanks Apparatus: Manual, TecQuipment Ltd, 1981.

MODEL REDUCTION, ZEROS AND GAIN

Alan Johnson
Kramers Laboratory
Delft University of Technology
Prins Bernhardlaan 6
2628 BW Delft, The Netherlands

Abstract. The basic modal reduction technique is modified to allow the gain and the zeros of the reduced-order model to be arbitrarily chosen. Use is made of the ease with which both the gain and the zeros can be changed in the controllability canonical form. An example is provided to illustrate the method.

1. NOTATION

Lower case letters indicate vectors, upper case letters matrices. Scalars are denoted by lower case Greek letters. The following symbols are used:

$\lambda_i(A)$ is the i-th eigenvalue, or pole of the matrix A
$\phi(\mathbf{M},s)$ is the characteristic polynomial of the model $\mathbf{M}$
$\psi(\mathbf{M},s)$ is the zero polynomial of the model $\mathbf{M}$
$|A|$ is the determinant of A
I is the unity matrix.

2. INTRODUCTION

There are two approaches to reducing the order of a state-space model which are commonly used in control engineering, the modal approach [1] and the balanced realization approach [2]. The modal approach decomposes the model into 'dominant' and 'non-dominant' eigenvalues, where only the dominant eigenvalues are retained. It has tended to neglect the questions of how dominant zeros are to be included in the reduced model and whether the gain of the model is invariant under the reduction technique. More precisely, suppose the following continuous-time state space LTI model $\mathbf{M}(A,b,c)$ is given:

$$\dot{x}(t) = Ax(t) + bu(t) \tag{1.1}$$

$$x(0) = x_0 \tag{1.2}$$

$$y(t) = cx(t) \tag{1.3}$$

where $A \in \mathbb{R}^{nxn}$, $b \in \mathbb{R}^{nx1}$, $c \in \mathbb{R}^{1xn}$. The gain of $\mathbf{M}(A,b,c)$ is given by $k(\mathbf{M})$, its characteristic polynomial is $\phi(\mathbf{M},s)$, and its zero polynomial $\psi(\mathbf{M},s)$. Assume $\mathbf{M}(A,b,c)$ has $p \leq n$ dominant poles (the roots of the dominant characteristic polynomial $\phi_p(s)=0$) and $m<p$ dominant zeros (the roots of the dominant zero polynomial $\psi_m(s)=0$). In this paper we present a method for finding a reduced-order state space LTI model $\mathbf{M}_R(A_R,b_R,c_R)$ where $A_R \in \mathbb{R}^{pxp}, b_R \in \mathbb{R}^{px1}, c_R \in \mathbb{R}^{1xp}$ such that $k(\mathbf{M}_R)=k(\mathbf{M})$, $\phi(\mathbf{M}_R,s)=\phi_p(s)$ and $\psi(\mathbf{M}_R,s)=\psi_m(s)$.

The basis of all modal approaches, Davison's technique, is shown in Fig.1, steps 1 to 3. The model $\mathbf{M}(A,b,c)$ is transformed to the Jordan form $\mathbf{M}_D(H^{-1}AH,H^{-1}b,Hc)$ using the non-singular matrix $H = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix}$, so that

$H^{-1}AH = \begin{bmatrix} \Lambda_1 & 0 \\ 0 & \Lambda_2 \end{bmatrix}$, $H^{-1}b = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix}$ and $Hc=[f_1 \;\; f_2]$. Note that $k(\mathbb{M})$, $\psi(\mathbb{M},s)$ and $\phi(\mathbb{M},s)$ are invariant under a similarity transformation. The dominant subsystem $\mathbb{M}_s(\Lambda_1,g_1,f_1)$, with $|sI-\Lambda_1|=\phi_p(s)$, is split off in step 2, and in the third step the reduced model $\mathbb{M}_R(H_{11}\Lambda_{11}^{-1},H_{11}g_1,f_1H_{11}^{-1})$ is created by a similarity transformation using the nonsingular matrix H_{11}. The zero polynomial $\psi(\mathbb{M}_R,s)$ will not, in general, be $\psi_m(s)$, as the following example shows.

Example 1

$$\mathbb{M}(A,b,c)=\mathbb{M}(\begin{bmatrix} -10.00 & -1.60 & 0.77 & 2.50 \\ 3.50 & -0.88 & -2.10 & 0.20 \\ 1.10 & -0.40 & -3.00 & -0.90 \\ 0 & 0 & 0 & -2.67 \end{bmatrix}, \begin{bmatrix} 1.0 \\ 0 \\ 0.4 \\ 0 \end{bmatrix}, [2.6 \;\; 3.1 \;\; 0.9 \;\; 3.8])$$

has eigenvalues = -9.4195, -0.9510, -3.5095, -2.6700, gain = 0.5206 and zeros = -0.7499, -7.3722, -2.6700. Using the modal approach the fastest mode is discarded, leaving

$$\mathbb{M}_R(A_R,b_R,c_R)=\mathbb{M}_R(\begin{bmatrix} -1.5260 & -1.8915 & 1.5433 \\ -0.6030 & -2.9345 & -0.4778 \\ 0 & 0 & -2.6700 \end{bmatrix}, \begin{bmatrix} 0.4833 \\ 0.6105 \\ 0 \end{bmatrix}, [2.6201 \;\; 1.0549 \;\; 4.7979])$$

which has eigenvalues = -0.9510, -3.5095, -2.6700, gain= 0.4091 and zeros = -0.7148, -2.67.

□

To remedy this shortcoming of the modal approach a modification to the method is proposed as shown in Fig.1, steps 4-9. The modification consists of a number of transformations leading to the controllable canonical form, where the gain and zero polynomial can be easily manipulated.

3. THE CONTROLLABLE CANONICAL FORM

Recall that if (A,b) is controllable, $\mathbb{M}(A,b,c)$ described by (1.1)-(1.3) can be transformed by the nonsingular matrix T into the state space LTI model $\mathbb{M}_c(A_c,b_c,c_c)$ in controllable canonical (or phase variable) form where:

$$A_c=T^{-1}AT=\begin{bmatrix} 0 & 1 & . & 0 & 0 \\ 0 & 0 & . & 0 & 0 \\ . & . & . & . & . \\ 0 & 0 & . & 0 & 1 \\ -\alpha_n & -\alpha_{n-1} & . & -\alpha_2 & -\alpha_1 \end{bmatrix}, b_c=kT^{-1}b=\begin{bmatrix} 0 \\ 0 \\ . \\ 0 \\ k \end{bmatrix}, c_c=k^{-1}cT=[c_{c1}c_{c2}c_{c3}...c_{cn}].$$

The transformation is accomplished by using the matrix T:

$$T=[b \;\; Ab \;\; ... \;\; A^{n-1}b]\begin{bmatrix} \alpha_{n-1} & \alpha_{n-2} & \cdots & \alpha_2 & \alpha_1 & 1 \\ \alpha_{n-2} & \alpha_{n-3} & \cdots & \alpha_1 & 1 & 0 \\ \alpha_{n-3} & \alpha_{n-4} & \cdots & 1 & 0 & 0 \\ . & . & . & . & . & . \\ 1 & 0 & \cdots & 0 & 0 & 0 \end{bmatrix}$$

Note that this is not in the form usually given, but has the advantage that all the poles are in A_c, all the zeros in c_c and the gain in b_c. More precisely,
the *characteristic polynomial* of $\mathbf{M}_c(A_c,b_c,c_c)$, and hence of $\mathbf{M}(A,b,c)$, is given by:

$$|sI-A| = |sI-A_c| = s^n + \alpha_1 s^{n-1} + \alpha_2 s^{n-2} \ldots + \alpha_{n-1}s + \alpha_n, \tag{3.1}$$

the *zero polynomial* of $\mathbf{M}_c(A_c,b_c,c_c)$, and hence of $\mathbf{M}(A,b,c)$ is given by:

$$|sI-A|c(sI-A)^{-1}b = |sI-A_c|c_c(sI-A_c^{-1})b_c = k[c_{c1} + sc_{c2} + \ldots + s^{n-1}c_{cn}] \tag{3.2}$$

and the *gain* of $\mathbf{M}_c(A_c,b_c,c_c)$, and hence of $\mathbf{M}(A,b,c)$ is given by (2.2):

$$-cA^{-1}b = -c_cA_c^{-1}b_c = -k^{-1}cT\begin{bmatrix} 0 & 0 & 0 & \ldots & 0 & -\alpha_n^{-1} \\ 1 & 0 & 0 & \ldots & 0 & 0 \\ 0 & 1 & 0 & \ldots & 0 & 0 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & \ldots & 1 & 0 \end{bmatrix}\begin{bmatrix} 0 \\ 0 \\ 0 \\ .. \\ k \end{bmatrix} = k \tag{3.3}$$

4. THE METHOD

The steps in the method are described briefly with reference to Fig.1. Notice that any uncontrollable modes are preserved, since these will have been introduced by the mechanistic modelling technique which produced $\mathbf{M}(A,b,c)$.

Step 4. The uncontrollable modes are split off, so $(\mathcal{A}_{11},\mathscr{b}_1)$ is controllable
Step 5. Since $(\mathcal{A}_{11},\mathscr{b}_1)$ is controllable, a T exists to form $\mathbf{M}_c(A_c,b_c,c_c)$.
Step 6. Set $k[c_{c1} + sc_{c2} + \ldots + s^{n-1}c_{cn}] = \psi_m(s)$ and $b_c^* = k$.
Step 7. Transform back using T, so that $\psi_7(s) = \psi_m(s)$ and $k_7 = k$.
Step 8. Append the uncontrollable modes. $\psi_8(s) = |sI-A_{22}|\psi_m(s)$ and $k_8 = k$.
Step 9. As in Davison's technique. $\psi_9(s) = |sI-A_{22}|\psi_m(s)$ and $k_9 = k$.

The choice of $\psi(s)$ and to some extent k in step 6 presents a degree of freedom which could be usefully exploited. For example, it might be possible to choose $\psi(s)$ so that the states of the reduced model resemble those of the original model in some well-defined way.

Example 2
With the same $\mathbf{M}(A,b,c)$ as in Example 1, we wish to find a reduced model with the same eigenvalues as before, but with gain = 0.5206 and a zero at -4.0. Using the proposed method, we find that:

$$\mathbf{M}_R(A_R,b_R,c_R) = \mathbf{M}_R\left(\begin{bmatrix} -1.7621 & -1.9728 & 1.3503 \\ -0.7185 & -2.6983 & -0.6456 \\ 0 & 0 & -2.6700 \end{bmatrix}, \begin{bmatrix} 1.8782 \\ 1.9696 \\ 0 \end{bmatrix}, [-1.5107 \;\; 1.6612 \;\; -4.5122]\right)$$

The gain of the reduced model is 0.5206, and the zeros -4.0, and -2.67.

□

5. CONCLUSIONS

A method has been presented which ensures that a reduced model has desired poles, zeros and gain. This could be necessary when process variables are decoupled and zeros are lost. It would also be useful to extend the method to the multivariable case.

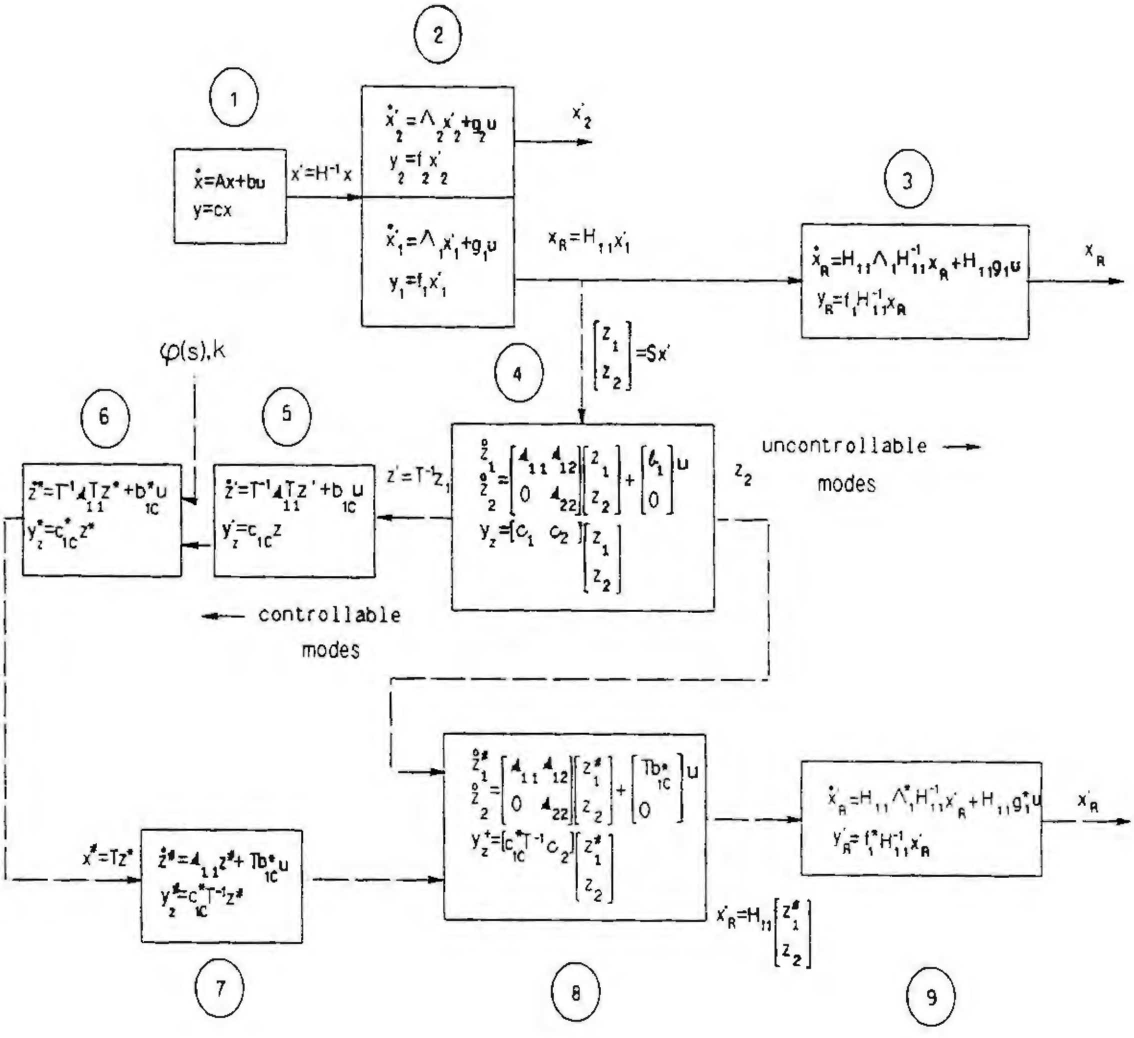

1
2
3
4
5
6
7
8
9
$\dot{x}=Ax+bu$
$y=cx$
$x'=H^{-1}x$
$\dot{x}'_2=\Lambda_2x'_2+g_2u$
$y_2=f_2x'_2$
$\dot{x}'_1=\Lambda_1x'_1+g_1u$
$y_1=f_1x'_1$
x'_2
$x_R=H_{11}x'_1$
$\dot{x}_R=H_{11}\Lambda_1H_{11}^{-1}x_R+H_{11}g_1u$
$y_R=f_1H_{11}^{-1}x_R$
x_R
$\begin{bmatrix}z_1\\z_2\end{bmatrix}=Sx'$
$\varphi(s),k$
uncontrollable modes
z_2
$z'=T^{-1}z_1$
controllable modes
$\dot{z}^*=T^{-1}A_{11}Tz^*+b^*_{1c}u$
$y^*_z=c^*_{1c}z^*$
$\dot{z}'=T^{-1}A_{11}Tz'+b_{1c}u$
$y'_z=c_{1c}z$
$x^*=Tz^*$
$\dot{z}^*=A_{11}z^*+Tb^*_{1c}u$
$y^*_z=c^*_{1c}T^{-1}z^*$
$x'_R=H_{11}\begin{bmatrix}z^*_1\\z_2\end{bmatrix}$
$\dot{x}'_R=H_{11}\Lambda^*_1H_{11}^{-1}x'_R+H_{11}g^*_1u$
$y'_R=f^*_1H_{11}^{-1}x'_R$
x'_R

Fig.1

6. REFERENCES

[1] Bonvin, D. and D.A.Mellichamp., A unified derivation and critical review of modal approaches to model reduction. Int.J.Control, 35 (1982), 5, 829-848.

[2] Moore, B.C., Principal component analysis in linear systems: controllability, observability and model reduction. IEEE Trans. Aut. Control, AC-26 (1981), 1, 17-32.

Order Reduction and Determination of Dominant State Variables of Nonlinear Systems

Dr.-Ing. Boris Lohmann , Höhrenbergstr. 10 , D - 78476 Allensbach

By the method presented here, nonlinear system models can be simplified by reducing the number of state equations. The dominant state variables (which the reduced system approximates) can be chosen according to technical reasons or with the help of a dominance analysis. All computations are based on proven algorithms and most of them are free of iterations.

1 Introduction

The simulation, analysis and controller-design of technical systems are frequently complicated by the complexity of the corresponding nonlinear system models. These tasks can be simplified by reducing the order of the system. Consider the current state space representation of a nonlinear time-invariant system

$$\dot{x}(t) = f(x,u) \tag{1}$$

with the state vector x of dimension n and the input vector u of dimension p. This can be rewritten in the form

$$\dot{x}(t) = Ax(t) + Bu(t) + F\,g(x,u) \tag{2}$$

without loss of generality. The vector $g(x,u)$ exclusively comprises the *nonlinear* summands of the elements of $f(x,u)$, that is, every such term appears only once and is free from any possible constant factors. Starting from (2) the task of order reduction is now to find a system

$$\dot{\tilde{x}}(t) = \tilde{A}\tilde{x}(t) + \tilde{B}u(t) + \tilde{F}\,g(W\tilde{x},u) \tag{3}$$

of lower order $\tilde{n}$ by which the behaviour of the original (2) can be imitated or, more precisely, by which at least the *most important* state variables can be approximated. The most important or so called *dominant* state variables are oftenly identified easily. These not only include the variables of importance in primary systems, such as measurement and feedback variables, but also include internal variables which are important for the dynamic properties. Once identified, all dominant state variables can be combined into a vector x_{do} and are related to the original vector x by

$$x_{do} = Rx \tag{4}$$

where every row of the matrix R contains a single "one" and otherwise zeros.

The special case of a *linear system* occurs, when the vector $f(x,u)$ in (1) consists exclusively of liner combinations of the elements of x and thereby the term $Fg(x,u)$ in (2) disappears. Many different kinds of methods for the order reduction of such systems are already known [1,4,11]. For nonlinear systems however they are still few[1]. Worth mentioning is an approach from *U. Pallaske* [8,12] which is based on an *orthogonal projection* of the state vector of the system. Furthermore *Singular Perturbation* methods [5,13] can be applied to nonlinear systems.

[1] However there are some very helpful methods which aim primarily not a reduction of the system order but a simplification of the right hand side of the state equation (1) [2,7].

2 The Order Reduction Method

The matrices $\tilde{A}, \tilde{B}, \tilde{F}, W$ in (3) must subsequently be determined so that $\tilde{x}(t)$ approximates the time curves of the dominant state variables. These must be chosen by the designer and combined in x_{do} according to (4). Assuming an ideal order reduction, the original system and the reduced system (3) react to every stimulus with the same paths $x_{do}(t) = \tilde{x}(t)$, and $\dot{x}_{do}(t) = \dot{\tilde{x}}(t)$ is also valid. By substitution eq.(3) becomes

$$\dot{x}_{do}(t) = \tilde{A}x_{do}(t) + \tilde{B}u(t) + \tilde{F}g(Wx_{do}, u) . \tag{5}$$

In practice this ideal cannot be reached. Following an approach to order reduction of *linear* systems by *E. Eitelberg* [3] it is sufficient if eq.(5) is *almost* fulfilled and the *equation error*

$$d_1(t) = \dot{x}_{do}(t) - \tilde{A}x_{do}(t) - \tilde{B}u(t) - \tilde{F}g(Wx_{do}, u) \tag{6}$$

is kept small. It will be shown that from this requirement the matrices $\tilde{A}, \tilde{B}, \tilde{F}$ can be determined in a straightforward manner (see also [9,10]). The requirement for matrix W on the other hand is to achieve its direct task as well as possible, namely to reconstruct the complete state vector x from x_{do} as it is used as an argument for g. This means keeping the equation error

$$d_2(t) = x(t) - Wx_{do}(t) \tag{7}$$

as small as possible. For which functions of time should this task be solved? Functions which can be produced by the *simulation* of the original system are certainly suitable. In order to cover the largest areas of the state space as possible, it will be necessary to carry out not one but several simulations with various initial values and input functions. This results in vectors $x(t_{0i}), x(t_{1i}), \ldots, x(t_{ei})$, $i = 1, \ldots, r$ for each of the r simulations. The error equation (7) can therefore be viewed at the discrete points in time $t_{01}, \ldots, t_{e1}, \ldots, \ldots, t_{0r}, \ldots, t_{er}$. In order to keep d_2 small, the weighted sum of the squares of the euclidean norm of $d_2(t)$ is minimized at the specified discrete points in time:

$$J_2 = q^2(t_{01})|d_2(t_{01})|^2 + \ldots + q^2(t_{er})|d_2(t_{er})|^2 \overset{!}{=} \min . \tag{8}$$

The positive real weighting factors $q(t_{01}), \ldots, q(t_{er})$ can be prescribed by the designer and can be used for a time weighting within a single simulation or for a weighting of complete simulations against each other. By arranging the error vectors d_2 side by side in matrix D_2, and the weighting factors q in the diagonal matrix Q, the quality gauge J_2 can be rewritten in a matrix formulation:

$$J_2 = trace\{D_2 Q Q^T D_2^T\} . \tag{9}$$

In order to facilitate the substitution of D_2 also the vectors $x(t_i)$ and $x_{do}(t_i)$ are combined into matrices X and X_{do}, leading to $D_2 = X - WX_{do}$. The optimal matrix W is then computed by evaluating the necessary condition $\partial J_2 / \partial W = 0$ for a minimum of J_2 with the final result

$$W = XQQ^T X_{do}^T \left(X_{do} QQ^T X_{do}^T\right)^{-1} . \tag{10}$$

In order to solve the second task, the analogous quality gauge

$$J_1 = q^2(t_{01})|d_1(t_{01})|^2 + \ldots + q^2(t_{er})|d_1(t_{er})|^2 = trace\{D_1 QQ^T D_1^T\} \tag{11}$$

is considered. Again the vectors $d_1(t_i)$ are combined into matrix D_1. By combining in addition the vectors $\dot{x}_{do}(t_i)$ (computed from eq.(2)) into matrix $\dot{X}_{do}$, the vectors $u(t_i)$ into U and the vectors $g(Wx_{do}(t_i), u(t_i))$ of non-linerities into matrix Γ, matrix D_1 can be expressed by

$$D_1 = \dot{X}_{do} - \tilde{A}X_{do} - \tilde{B}U - \tilde{F}\Gamma = \dot{X}_{do} - \left[\tilde{A},\tilde{B},\tilde{F}\right]\begin{bmatrix} X_{do} \\ U \\ \Gamma \end{bmatrix} = \dot{X}_{do} - EM, \tag{12}$$

with the abbreviations $E = \left[\tilde{A},\tilde{B},\tilde{F}\right]$ and $M^T = \left[X_{do}^T, U^T, \Gamma^T\right]$. From that, the optimal matrix E_{opt} is resolved by evaluating the necessary condition for a minimum of J_1 :

$$E_{opt} = \left[\tilde{A},\tilde{B},\tilde{F}\right] = \dot{X}_{do}QQ^T M^T \left(MQQ^T M^T\right)^{-1}. \tag{13}$$

The reduced system (3) is now completely determined. In addition to the approximation $\tilde{x}(t)$ of $x_{do}(t)$ which is provided by the reduced system, the *complete* state vector x of the original system (including the *non-dominant* states) can be approximated by the relation $\hat{x}(t) = W\tilde{x}(t)$.

Secondary conditions of the form $L = EH$ can be fulfilled (when minimizing J_1) by choosing

$$\begin{aligned} E_{opt} = {} & \dot{X}_{do}QQ^T M^T \left(MQQ^T M^T\right)^{-1} + \left(L - \dot{X}_{do}QQ^T M^T (MQQ^T M^T)^{-1} H\right) \cdot \\ & \cdot \left(H^T (MQQ^T M^T)^{-1}\right)^{-1} H^T (MQQ^T M^T)^{-1} \end{aligned} \tag{14}$$

instead of eq.(13). If *equilibrium sets* of the original system are arranged in H according to

$$H = \begin{bmatrix} x_{do,stat,1} & \dots & x_{do,stat,m} \\ u_{stat,1} & \dots & u_{stat,m} \\ g(Wx_{do,stat,1}, u_{stat,1}) & \dots & g(Wx_{do,stot,m}, u_{stat,m}) \end{bmatrix} \tag{15}$$

and one choses $L = 0$, then eq.(14) guarantees that the equation error vector d_2 will disappear at the equilibrium points. Thus also the reduced sytem possesses these equilibrium sets. If the equilibrium sets are identical to some steady state values of the original system corresponding to some system stimulii of interest, then in general the reduced system too will tend towards the same steady state values when stimulated by the same input functions. This formalism for the first time allows influencing the *steady state behaviour* of nonlinear order-reduced systems. It can easily be extended to general input functions.

3 Determination of Dominant State Variables

When applying the steps described above to technical systems, difficulties may occur in choosing dominant state variables. Hence, the question arises if some *linear combinations* of state variables would not be more suitable as the dominant variables. This can be phrased more generally: can a *linear transformation* of the state vector be found which allows the computation of expressive *dominance numbers*, just as the *modal transformation* [1] or the transformation to *balanced representation* [11] do for *linear* systems? The determination of such a transformation is in fact possible based on an analysis of the state trajectories, as was used in the form of controllability matrices to provide the starting point to *B.C. Moore's* dominance analysis for *linear* systems [11]. First, the task is defined: A state transformation

$$\underset{(n,1)}{z(t)} = \underset{(n,n)}{V} \; \underset{(n,1)}{x(t)} \tag{16}$$

is required, with the property, that the time-curves for all n state variables $x_1,\dots,x_n$ of the original system can be *approximated* from the first $\tilde{n}$ components $z_1,\dots.,z_{\tilde{n}}$ of z which are combined in the vector

$$\underset{(\tilde{n},1)}{\boldsymbol{z}_{top}(t)} = \underset{(\tilde{n},n)}{\boldsymbol{V}_{top}}\, \underset{(n,1)}{\boldsymbol{x}(t)} \,, \tag{17}$$

by the approximating equation

$$\hat{\boldsymbol{x}}(t) = \boldsymbol{W}\, \boldsymbol{z}_{top}(t) \ . \tag{18}$$

This approximation has to be optimal in the least squares sense

$$J = \sum_{t_i=t_0}^{t_e} (\boldsymbol{x}(t_i) - \hat{\boldsymbol{x}}(t_i))^T \boldsymbol{S}^T \boldsymbol{S} (\boldsymbol{x}(t_i) - \hat{\boldsymbol{x}}(t_i)) = trace\left\{\boldsymbol{S}\left(\boldsymbol{X} - \hat{\boldsymbol{X}}\right)\left(\boldsymbol{X} - \hat{\boldsymbol{X}}\right)^T \boldsymbol{S}^T\right\} \overset{!}{=} \min \,. \tag{19}$$

The components of $\boldsymbol{z}_{top}$ are then dominant compared to the other components of $\boldsymbol{z}$. In (19) $\boldsymbol{S}$ denotes a regular diagonal matrix of positiv real weighting factors $s_1,\ldots,s_n$, by which the designer can influence the quality of approximation of the single state variables. The vectors $\boldsymbol{x}(t_i)$ are known from the simulations.

The task can be solved via a *singular value decomposition* [6] of the matrix $\Psi = \boldsymbol{S}\,\boldsymbol{X}$. This produces the orthogonal matrices $\boldsymbol{T}$ and $\boldsymbol{P}$ and the diagonal matrix Σ of the real non-negative *singular values* of Ψ (sorted in non-increasing order) with the property

$$\Psi = \boldsymbol{T}\, \Sigma\, \boldsymbol{P}^T \ . \tag{20}$$

By choosing

$$\boldsymbol{V} = \boldsymbol{T}^T \boldsymbol{S} \ , \qquad \boldsymbol{V}_{top} = \boldsymbol{T}_L^T \boldsymbol{S} \ , \qquad \boldsymbol{W} = \boldsymbol{S}^{-1} \boldsymbol{T}_L \tag{21}$$

where the matrix $\boldsymbol{T}_L$ comprises the first $\tilde{n}$ columns of $\boldsymbol{T}$, the quality gauge J is minimized [6]. Furthermore the singular values $\sigma_1,\ldots,\sigma_n$ on the diagonal of Σ indicate the *inaccuracy* which arises due to the approximation (18), i.e. the minimum value of J is given by

$$\varepsilon(\tilde{n}) = J_{opt} = \sigma_{\tilde{n}+1}^2 + \ldots + \sigma_n^2 = \sum_{i=\tilde{n}+1}^{n} \sigma_i^2 \ . \tag{22}$$

Hence the order reduction is done as follows: a singular value decomposition of the matrix $\Psi = \boldsymbol{S}\,\boldsymbol{X}$ is carried out, where the diagonal elements of $\boldsymbol{S}$ are initially chosen to be equal to the reciprocals of the maximum deviations of the state variable curves from their mean or steady-state values. The order $\tilde{n}$ of the reduced system is chosen so that the singular values $\sigma_{\tilde{n}+1},\ldots,\sigma_n$ are small compared to the others. In the transformed system model

$$\dot{\boldsymbol{z}}(t) = \boldsymbol{V}\boldsymbol{A}\boldsymbol{V}^{-1}\boldsymbol{z}(t) + \boldsymbol{V}\boldsymbol{B}\,\boldsymbol{u}(t) + \boldsymbol{V}\boldsymbol{F}\, \boldsymbol{g}(\boldsymbol{V}^{-1}\boldsymbol{z},\boldsymbol{u}) \tag{23}$$

the *first* $\tilde{n}$ state variables are in fact the dominant ones. The singular values σ_i can be considered *dominance size values*, as σ_i^2 is precisely that value with which J increases (i.e. the approximation (18) disimproves) when the state variable z_i is defined as *not-dominant*. As a result the transformed system (23) can be reduced according to section 2. From the time curves $\tilde{\boldsymbol{z}}(t)$ of the resulting reduced system

$$\dot{\tilde{\boldsymbol{z}}}(t) = \tilde{\boldsymbol{A}}\,\tilde{\boldsymbol{z}}(t) + \tilde{\boldsymbol{B}}\boldsymbol{u}(t) + \tilde{\boldsymbol{F}}\,\boldsymbol{g}(\boldsymbol{W}\tilde{\boldsymbol{z}},\boldsymbol{u}) \tag{24}$$

approximations of the original curves are achieved by $\hat{\tilde{\boldsymbol{x}}}(t) = \boldsymbol{W}\,\tilde{\boldsymbol{z}}(t)$.

4 Outlook

A simple and effective approach to order reduction of nonlinear system has been presented here. In [10] a nonlinear *vehicle suspension model* of order 10 has been reduced successfully to orders seven and five, shortening the simulation time by a factor of 10 and 12 respectively. Other satisfactory reductions of nonlinear systems have been carried out, both, with and without dominance analysis and have reduced system complexity and simulation time. However there remain numerous open questions concerning the as yet young field of non-linear model simplification:

- Firstly comes the question of which special measures can be used to improve the behaviour of *output variables* $y_i = c_i(\boldsymbol{x}, \boldsymbol{u})$.
- It would certainly be desirable to be able do without simulations of the original system, either fully or in part.
- The result of the reduction in all currently known methods is judged on the basis of a subjective evaluation of the resulting time curves. A systematic method would be helpful in this and would also assist in the investigation of *stability behaviour*.
- As in the theory of linear systems, the relationships between the order reduction and the terms of *minimal realisation*, of *controllability* and of *observability* are of interest.

These issues provide interesting fields of study for the future, not least because in practice the simplification of non-linear system models is more urgently required than for linear models.

References

[1] Bonvin, D. and Mellichamps, D.A., A Unified Derivation and Critical Review of Modal Approaches to Model Reduction. Int. Journal Control 35 (1982), 829-848.

[2] Desrochers, A.A. and Al-Jaar,R.Y., A Method for High Order Linear System Reduction and Nonlinear System Simplification. Automatica 21 (1985) 93-100.

[3] Eitelberg, E., Model Reduction by Minimizing the Weighted Equation Error. Int. Journal Control 34 (1981) 1113-1123.

[4] Glover, K., All Optimal Hankel-Norm Approximation of Linear Multivariable Systems and their L_∞-Error Bounds. Int. Journal Control 39 (1984) 1115-1193.

[5] Kokotovic, P.V., O'Malley, R.E. and Sannuti,P., Singular Perturbations and Order Reduction in Control Theory - An Overview. Automatica 12 (1976) 123-132.

[6] Lawson, C.L. and Hanson, R.J., Solving Least Squares Problems. Prentice-Hall, 1974.

[7] Lin, C.S. and Chang, P.R., Automatic Dynamics Simplification for Robot Manipulators. Proc. IEEE Conf. Decision and Control, Las Vegas 1984,.752-759.

[8] Löffler, H.P. and Marquardt, W., Order Reduction of Nonlinear Differential-Algebraic Process Models. J. Proc. Control 1 (1991) 32-40.

[9] Lohmann, B., Order-Reduction Method for Nonlinear Dynamical Systems. Electronics Letters 28 (1992) 658-659.

[10] Lohmann, B., Order Reduction of Nonlinear Systems and Application to a Hydropneumatic Vehicle suspension. Submitted to IEEE Trans. Control Systems Technology 1993, Special Issue on Automotive Control Systems, to appear 1994, copies available from the author.

[11] Moore, B.C., Principal Component Analysis in Linear Systems: Controllability, Observability and Model Reduction. IEEE Trans. Autom. Control 26 (1981) 17-32.

[12] Pallaske, U., Ein Verfahren zur Ordnungsreduktion mathematischer Prozessmodelle. Chem. Ing. Tech. 7 (1987) 59, MS 1617/87.

[13] Saksena, V.R., O'Reilly, J. and Kokotovic, P.V., Singular Perturbations and Time-Scale Methods in Control Theory: Survey 1976-1983. Automatica 20 (1984) 273-293.

Numerical Approximation of Slowly Varying Disturbances in Nonlinear Systems

Gerta Zimmer, Florian Hiss
Technische Universität Berlin
email: {gerta,hiss}@math.tu-berlin.de

Abstract Observing the state of a nonlinear system by means of solving an optimization problem is shown to be well-posed. By making use of this result, disturbances which are slowly time varying are modelled as constant parameters and approximated by piecewise constant functions.

1. Introduction

Consider a nonlinear dynamical system described in the state space form by

$$\dot{x} = f(x) \qquad y = c^T x \tag{1}$$

where $f \in C^2(\mathbb{R}^n, \mathbb{R}^n)$, $c \in \mathbb{R}^n/\{0\}$. Let

$$\Phi(\cdot; t_0, \cdot) : \mathbb{R}^n \to C^1(\mathbb{R}, \mathbb{R}^n) \quad , \quad x_0 \mapsto \Phi(\cdot, t_0, x_0) \tag{2}$$

denote the flow of (1). I.e. for all $x_0 \in \mathbb{R}^n, t_0 \in \mathbb{R}$, $\Phi(\cdot; t_0, x_0)$ is the uniquely determined solution of $\dot{x} = f(x)$, $x(t_0) = x_0$.

With $T > 0$ let us now restrict our attention to the interval $I := [0, T]$. Then clearly (1) defines a mapping

$$\mathcal{F} \ : \ \mathbb{R}^n \to C^1(I, \mathbb{R}) \quad , \quad x \mapsto c^T \Phi(\cdot; 0, x)_{|_I} =: \mathcal{F}(x) \tag{3}$$

If (1) is the correct description of the dynamical system under consideration, the observation of the system output during a time interval of lenght T would yield a function $y \in \mathcal{F}(\mathbb{R}^n)$. Taking the inverse $\mathcal{F}^{-1}(y)$ and plugging it into the flow Φ one would immediately recover the complete state function of the system. But, as a matter of fact, it is highly unlikely that the description (1) is anything but an approximation of the real system dynamics, given by

$$\dot{x} = \tilde{f}(x) \quad , \quad \tilde{y} = \tilde{c}^T x \tag{4}$$

with $\tilde{f} \approx f$ and $\tilde{c} \approx c$, but both $\tilde{f}$ and $\tilde{c}$ are not exactly known.

Therefore we cannot expect the real system output $\tilde{y}$ to be in $\mathcal{F}(\mathbb{R}^n)$. If we additionally allow for disturbances in the measurement, the measured output is assumably not even in $C^1(I, \mathbb{R})$.

Given the exact initial state x_0, we know that y and $\tilde{y}$ are close if $||f - \tilde{f}||$, and $||c - \tilde{c}||$, respectively, are "sufficiently" small. So if $\mathcal{F}^{-1}$ could be continuously extended to a neighborhood of $\mathcal{O}$, $\mathcal{F}^{-1}(\tilde{y})$ would be an approximation of the true state x_0 at $t = 0$, and $\Phi(\cdot; 0, \mathcal{F}^{-1}(\tilde{y}))$ might be fairly well suited to describe the true state function in a neighborhood of $t = 0$.

Now first we are going to derive conditions under which $\mathcal{F}^{-1}$ exists and can be continuously extended to a neighborhood of $\mathcal{F}(\mathbb{R}^n)$. Afterwards we point out how to use the output function of the real system to continuously update the approximation of the state function.

2. Preliminaries

In order to successfully recover the state function of system (1) by using the output on I we have to demand observability of (1) in the sense that $\mathcal{F}$ is injective. I.e. different initial states generate different output functions.

Definition 1 Let $\mathcal{R} \subset \mathbb{R}^n$. (1) is called *observable on* $\mathcal{R} \times I$ iff $\mathcal{F}_{|_\mathcal{R}}$ is injective.

We shall require in the following that $\mathcal{R} \subset \mathbb{R}^n$ is compact and (1) is observable on $\mathcal{R} \times I$. As $C^1(I, \mathbb{R}) \subset L_2(I)$, we may conceive $\mathcal{F}$ as a mapping into the Banachspace $L_2(I)$. To abbreviate let $Y := L_2(I)$ and $\mathcal{O} := \mathcal{F}(\mathcal{R})$, and let $< \cdot, \cdot >_Y$ and $|| \cdot ||_Y$ denote the inner product and the norm in Y, respectively. Since f is twice continuously differentiable, it follows with [2], p.99, that

Lemma 1 $\mathcal{F} : \mathcal{R} \to Y$ is twice continuously differentiable.

For the following outline we have to consider the derivative map of $\mathcal{F}$, $\mathcal{F}' : \mathcal{R} \times \mathbb{R}^n \to Y$, $(x, z) \mapsto \mathcal{F}'(x)z$ given by $[\mathcal{F}'(x)z](t) := c^T \frac{\partial}{\partial x} z$. As $\frac{\partial \Phi(\cdot;0,x)}{\partial x} =: \overset{\Phi(t;0,x)}{\Phi_{f_x}(\cdot;0,x)}$ is the flow of the linearized system

$$\dot{z} = f_x(\Phi(t;0,x))z \quad , \quad v = c^T x \tag{5}$$

the association between the derivative map and the linearised system is obvious. Thus for all $x \in \mathcal{R}$, $z \in \mathbb{R}^n$ we have

$$[\mathcal{F}'(x)z](t) := c^T \Phi_{f_x}(t;0,x)z \tag{6}$$

Additionally we have to assume that $\mathcal{F}'(x)$ has full rank for all $x \in \mathcal{R}$. This is equivalent to assume that the linearized system (5) ist observable for all $x \in \mathcal{R}$. Using (6) we deduce immediately

Lemma 2 $\int_0^T \Phi_{f_x}^T(t;0,x)cc^T\Phi_{f_x}(t;0,x)\,dt$ is positive definite *iff* (5) is observable on I.

The proof is analogous to the proof of Theorem 1 in [6].

3. Extension of $\mathcal{F}^{-1}$

As $\mathcal{F}$ is injective, there exists a unique invers $\mathcal{F}^{-1} : \mathcal{O} \to \mathbb{R}^n$. We are now going to extend $\mathcal{F}^{-1}$ on Y. For this reason we define

$$\mathcal{N} : \mathbb{R}^n \times Y \to \mathbb{R} \quad , \quad (x, y) \mapsto \frac{1}{2}\|y - \mathcal{F}(x)\|_Y^2 \tag{7}$$

Then it follows from [5] that the Hessian of $\mathcal{N}(x, y)$, $\mathcal{N}_{xx}(x, y)$, is positive definite for any $(x, y) \in \mathcal{R} \times Y$ where $y = \mathcal{F}(x)$. We shall further see that for every (x_0, y_0) with $y_0 = \mathcal{F}(x_0)$, there exist ρ and ε, such that $\mathcal{N}_{xx}(\cdot, y)$ is strictly convex in $B_\rho(x_0)$ for all $y \in B_\varepsilon(y_0)$. Here $B_\rho(x_0) := \{x \in \mathbb{R}^n, \|x - x_0\| \leq \rho\}$ and $B_\varepsilon(y_0) := \{y \in Y, \|y - y_0\| \leq \varepsilon\}$.

Lemma 3 For all $x_0 \in \mathcal{R}$, there exist ρ, $\varepsilon > 0$ such that $\mathcal{N}_{xx}(x, y)$ is positive definite in $(B_\rho(x_0) \cap \mathcal{R}) \times B_\varepsilon(\mathcal{F}(x_0))$.

The proof is omitted for reasons of space.

Lemma 3 gives a tool to derive sufficient conditions for the existence of a unique minimizer of $\mathcal{N}(\cdot, y)$.

Lemma 4 Let $\mathcal{R}$ be convex. For every $y_0 \in \mathcal{O}$ with $x_0 := \mathcal{F}^{-1}(y_0) \in \mathcal{R}$ there exist ρ, $\varepsilon_y > 0$ such that $\mathcal{N}(\cdot, y)$ has a unique minimizer in $B_\rho(x_0) \cap \mathcal{R}$ for all $y \in B_{\varepsilon_y}(y_0)$.

As $\mathcal{R}$ and $B_\rho(x_0)$ are convex, $\mathcal{R} \cap B_\rho(x_0)$ is convex, too. According to Lemma 3, $\mathcal{N}(\cdot;y)$ is convex on $\mathcal{R} \cap B_\rho(x_0)$. And since $\mathcal{R} \cap B_\rho(x_0)$ is convex and compact, there exists a unique minimizer $x \in \mathcal{R} \cap B_\rho(x_0)$ of $\mathcal{N}(\cdot, y)$. □

If $\mathcal{R}$ is unfortunately not convex, the statement of the above Lemma is still valid for any $\mathcal{F}^{-1}(y_0) \in \mathring{\mathcal{R}}$.

Corollary 1 For every $y_0 \in \mathcal{O}$ with $x_0 := \mathcal{F}^{-1}(y_0) \in \mathring{\mathcal{R}}$ there exist ρ, $\varepsilon_y > 0$ such that $\mathcal{N}(\cdot, y)$ has a unique minimizer in $B_\rho(x_0)$ for all $y \in B_{\varepsilon_y}(y_0)$.

Using [4], p.195, the result of Lemma 4 can be extended.

Lemma 5 For all $y \in Y$ there exists $x_0 \in \mathcal{R}$ so that for all $x \in \mathcal{R}$:

$$\|y - \mathcal{F}(x)\|_Y \geq \|y - \mathcal{F}(x_0)\|_Y \quad \text{i.e.} \quad \mathcal{N}(x, y) \geq \mathcal{N}(x_0, y) \tag{8}$$

Lemma 5 enables us to extend $\mathcal{F}^{-1}$ on Y by defining a "pseudoinverse" for $\mathcal{F}$ according to the following definition:

$$\mathcal{F}^- : Y \to \mathcal{R} \quad , \quad y \mapsto \arg\min_{x \in \mathcal{R}} \mathcal{N}(x, y) \tag{9}$$

For any $y \in Y$ we may regard $\mathcal{F}(\mathcal{F}^-(y))$ as a generalised projection of y on $\mathcal{O}$. And because of (9) we see immediately that for any $y_0 \in \mathcal{O}$, $\|\mathcal{F}(\mathcal{F}^-(y)) - y\| \leq \|y_0 - y\|$. Using the triangle inequality we thus get $\|\mathcal{F}(\mathcal{F}^-(y)) - y_0\| \leq 2\,\|y_0 - y\|$.

$\mathcal{F}^-$ may not be unique on Y. But with Lemma 4 it is obviously locally unique near $\mathcal{O}$ if $\mathcal{R}$ is convex.

Corollary 2 Let $\mathcal{R}$ be convex. For all $y_0 \in \mathcal{O}$ exists $\varepsilon = \varepsilon(y_0)$ such that $\mathcal{F}^-_{|B_\varepsilon(y_0)}$ is unique.

As we pointed out in the introduction, we are mainly interested in an inverse of $\mathcal{F}$ in a neighborhood of $\mathcal{O}$. And although $\mathcal{F}^-$ might not be unique in the entire space Y, it is not only unique but also continuously differentiable in a neighborhood of almost every y_0 of $\mathcal{O}$.

Theorem 1 Let $y_0 \in \mathcal{O}$ with $\mathcal{F}^{-1}(y_0) \in \overset{\circ}{\mathcal{R}}$. Then there is $\gamma = \gamma(y_0) > 0$ such that $\mathcal{F}^-_{|B_\gamma(y_0)}$ is continuously differentiable.

Proof: Let ε_y and ρ according to Lemma 4. Then $\mathcal{F}^-_{|B_\gamma(y_0)}$ is unique for any $\gamma \in (0, \varepsilon_y)$. So $\mathcal{F}^-$ may locally be described by

$$\mathcal{F}^- : B_\gamma(y_0) \rightarrow \mathbb{R}^n \quad , \qquad y \mapsto \arg \min_{x \in B_\rho(x_0)} \mathcal{N}(x, y)$$

Because $\mathcal{N}(\cdot, y)$ is strictly convex in $B_\gamma(y_0)$, this is equivalent to

$$\mathcal{F}^-(y) \quad = \quad \arg \min_{x \in B_\rho(x_0)} \Big(\mathcal{N}_x(x, y) = 0\Big)$$

Hence $\mathcal{F}^-(y)$ is implicitely defined by $\mathcal{N}_x(\mathcal{F}^-(y), y) = 0$. And as $\mathcal{N}_x(x_0, y_0) = 0$ and $\mathcal{N}_{xx}(x_0, y_0)$ is positive definite, $\mathcal{N}_x(\cdot, \cdot)$ meets the requirements of the implicit function theorem. Thus there exists $\gamma \in (0, \varepsilon_y)$ such that $\mathcal{F}^-_{|B_\gamma(y_0)}$ is continuously differentiable. Its derivative map in y is given by

$$D\mathcal{F}^-(y) : Y \rightarrow \mathbb{R}^n \quad , \qquad v \mapsto [D\mathcal{F}^-(y)]v$$

with

$$[D\mathcal{F}^-(y)]v \quad := \quad \Big(\mathcal{N}_{xx}(\mathcal{F}^-(y), y)\Big)^{-1} \int_0^T \Phi^T_{f_x}(t; 0, \mathcal{F}^-(y))\, v(t)\, dt$$

$D\mathcal{F}^-(\cdot)$ is obviously continuous with respect to y. □

So the problem of observing the system (1) by using a slightly disturbed output ist well posed.

4. Design of an Observer

Recall our intention to design an observer for (4) which simultanously approximates some unknown parameters or slowly varying disturbances. We account for unknown parameters as well as time varying disturbances by augmenting the state vector in (1) by some "constants". As we just saw, the observation problem is well posed. So we feel justified to use (1) to design an observer for (4). The restriction is that the linearization of the resulting system has to be observable.

Before proceeding we have to introduce some additional notation.

Let $\tilde{\mathcal{R}} \subset \mathbb{R}^n$ be compact and invariant with respect to the differential equation (4a). Let $\tilde{\Phi}(\cdot; 0, \cdot)$ denote the flow of (4), $\tilde{\mathcal{F}}$ the corresponding map on Y, and $\tilde{\mathcal{O}} := \tilde{\mathcal{F}}(\tilde{\mathcal{R}})$ the set of possible outputs of (4).

Additionally, let L denote a global Lipschitz constant for f on $\mathcal{R}$ and let $\tilde{\mathcal{R}} \subset \mathcal{R}$. The difference of $\tilde{f}$ and f on $\tilde{\mathcal{R}}$ shall be bounded by δ_f and the difference of $\tilde{c}$ and c on $\tilde{\mathcal{R}}$ shall be bounded by δ_c. Then it is well know that for any $x_0 \in \mathcal{R}$ and $\tilde{x}_0 \in \tilde{\mathcal{R}}$ the difference of $\Phi(t; 0, x_0)$ and $\tilde{\Phi}(t; 0, \tilde{x}_0)$ is bounded by

$$||\Phi(t; 0, x_0) - \tilde{\Phi}(t; 0, \tilde{x}_0)|| \quad \leq \quad e^{L\,t}(\delta_f + ||x_0 - \tilde{x}_0||) \tag{10}$$

Analogously, the difference in the output functions are bounded and in particular

$$||\mathcal{F}(x_0) - \tilde{\mathcal{F}}(\tilde{x}_0)||_Y \quad \leq \quad \delta_c||\Phi(T; 0, x_0)|| + (||c|| + \delta_c)e^{L\,T}(\delta_f + ||x_0 - \tilde{x}_0||) \tag{11}$$

Note that, except for the invariance property which will be needed for technical reasons, no particular information about $\tilde{f}$ is specified.

Using (11) we see that $\mathcal{F}(x_0)$ and $\tilde{\mathcal{F}}(\tilde{x}_0)$ are close in Y if x_0 and $\tilde{x}_0$ are close in $\mathcal{R}$ and with Theorem 1 the other way is true, too, if the model (1) is good enough that for any $\tilde{x}_0$ of $\tilde{\mathcal{R}}$, $\tilde{\mathcal{F}}(\tilde{x}_0) \in U_\varepsilon(\mathcal{O})$. Then there exists a unique best approximation $x_0 \in \mathcal{R}$ with

$$\mathcal{N}(x_0, \tilde{\mathcal{F}}(\tilde{x}_0)) \quad < \quad \mathcal{N}(x, \tilde{\mathcal{F}}(\tilde{x}_0))$$

for all $x \in \mathcal{R}/\{x_0\}$. Additionally, x_0 depends continuously on $\tilde{\mathcal{F}}(\tilde{x}_0)$. Hence we may use $\mathcal{N}(x, \tilde{\mathcal{N}}(\tilde{x}_0))$ to measure the quality of the approximation x. And instead of directly trying to recover $\tilde{x}$ continuously, we shall try to approximate $\tilde{x}(\cdot)$ at discrete points $\tilde{x}(T\,k) =: \tilde{x}_k$ by using the hybrid form observer which was proposed in [5].

We are going to show that this observer is suited to approximate the state function of (4) in the sense that it generates a sequence $\{x_k\}_{k \in \mathbb{N}}$ for which $x_k \approx \tilde{x}_k$. This sequence can then be extended to a piecewise continuous function [6].

The Newton step observer will be designed by using the model (1). Hence the rules read:

- Given x_k as an approximation of $\tilde{x}_k := \tilde{\Phi}(kT; 0, \tilde{x}_0)$.
- Improve the approximation by a Newton-Step:

$$\bar{x}_k \quad := \quad x_k - \mathcal{N}_{xx}^{-1}(x_k, \tilde{\mathcal{F}}(\tilde{x}_k))\mathcal{N}_x(x_k, \tilde{\mathcal{F}}(\tilde{x}_k)) \tag{12}$$

- Procede to the time $T(k + \mathrm{I})$ by setting

$$x_{k+1} \quad := \quad \Phi(T; 0, \bar{x}_k) \tag{13}$$

We are going to give conditions which assure that the Newton-Step observer is applicable. Further we will show convergence of the observations in the sense that for any $\varepsilon > 0$

$$\|x_k - \tilde{x}_k\| \quad \leq \quad \varepsilon \qquad\qquad (\ k \text{ large enough }\ )$$

provided that the model is sufficiently good and $\|x_0 - \tilde{x}_0\|$ is small enough.

5. Proof of Convergence

First we will give conditions which assure that, given x_0 and $\tilde{\mathcal{F}}(\tilde{x}_0)$, a Newton Step can be performed. Additionally we will estimate the difference between the improved approximation and the real system state relative to the difference between the old approximation and the system state.

Lemma 6 Let $x_0 \in \mathcal{R}$ and $\tilde{x}_0 \in \tilde{\mathcal{R}}$ and κ, $\varepsilon > 0$. Then there are $\Delta_0 = \Delta_0(\kappa), \delta_f = \delta_f(\varepsilon), \delta_c = \delta_c(\varepsilon) > 0$ such that $\mathcal{N}_{xx}(x_0, \tilde{\mathcal{F}}(\tilde{x}_0))$ and $\mathcal{N}_{xx}(x_0, \mathcal{F}(\tilde{x}_0))$ are invertible *and* $\|x_0 - \mathcal{N}_{xx}^{-1}(x_0, \tilde{\mathcal{F}}(\tilde{x}_0))\mathcal{N}_x(x_0, \tilde{\mathcal{F}}(\tilde{x}_0)) - \tilde{x}_0\| \leq \kappa\|x_0 - \tilde{x}_0\| + \varepsilon$. if $\|x_0 - \tilde{x}_0\| < \Delta_0$, $\|c - \tilde{c}\| < \delta_c$ and $\max\limits_{x \in \tilde{\mathcal{R}}} \|f(x) - \tilde{f}(x)\| < \delta_f$.

The proof is purely technical and therefore omitted.

Using Lemma 6 and equation (10), we are able to estimate the difference between $\tilde{x}_{k+1}$ and x_{k+1} with respect to the difference between $\tilde{x}_k$ and x_k.

Lemma 7 Let $\kappa, \varepsilon > 0$ and $\Delta_0 = \Delta_0(\kappa), \delta_f = \delta_f(\varepsilon), \delta_c = \delta_c(\varepsilon) > 0$ according to Lemma 6. Further let $x_0 \in \mathcal{R}$ and $\tilde{x}_0 \in \tilde{\mathcal{R}}$ with $\|x_k - \tilde{x}_k\| \leq \Delta_0$. Then

$$\|x_{k+1} - \tilde{x}_{k+1}\| \quad \leq \quad e^{LT}\kappa\|x_k - \tilde{x}_k\| + e^{LT}(\delta_f + \varepsilon)$$

if $\|c - \tilde{c}\| < \delta_c$ and $\max\limits_{x \in \tilde{\mathcal{R}}} \|f(x) - \tilde{f}(x)\| < \delta_f$.

This Lemma leads directly to the final theorem.

Theorem 2 Let $\kappa > 0$ with $\kappa e^{LT} < 1$, $\Delta_0 = \Delta_0(\kappa)$ according to Lemma 6, $x_0 \in \mathcal{R}$ and $\tilde{x}_0 \in \tilde{\mathcal{R}}$ with $\|x_0 - \tilde{x}_0\| \leq \Delta_0$. Let $\varepsilon > 0$ so that $\frac{2e^{LT}\varepsilon}{1 - \kappa e^{LT}} < \Delta_0$, $\delta_f := min\{\delta_f(\varepsilon), \varepsilon\}, \delta_c = \delta_c(\varepsilon)$. Then for all $k \in \mathbb{N}$

$$1)\ \|x_k - \tilde{x}_k\| \ \leq\ \Delta_0 \ \text{ and } \ 2)\ \|x_k - \tilde{x}_k\| \ \leq\ (e^{LT}\kappa)^k\ \|x_0 - \tilde{x}_0\| + \frac{e^{LT}(\varepsilon + \delta_f)}{1 - e^{LT}\kappa}$$

if $\|c - \tilde{c}\| < \delta_c$ and $\max\limits_{x \in \tilde{\mathcal{R}}} \|f(x) - \tilde{f}(x)\| < \delta_f$.

Proof: Using the assumptions, the first part is easily seen via complete induction.

The first part already assured us that the observation process is applicable. Again using complete induction, we see that

$$\|x_k - \tilde{x}_k\| \ \leq\ (e^{LT}\kappa)^k\ \|x_0 - \tilde{x}_0\| + e^{LT}(\varepsilon + \delta_f)\sum_{i=0}^{k-1}(e^{LT}\kappa)^i \ \leq\ (e^{LT}\kappa)^k\ \|x_0 - \tilde{x}_0\| + \frac{e^{LT}(\varepsilon + \delta_f)}{1 - e^{LT}\kappa}$$

□

6. An Example

The efficiency of the method is illustrated for the system

$$\begin{aligned}
\dot{C}_A &= \frac{\dot{V}}{V_R}(C_{A0}C_A) - k_1(\vartheta)C_A - k_3(\vartheta)C_A^2 \\
\dot{C}_B &= \frac{\dot{V}}{V_R}C_B + k_1(\vartheta)C_A - k_2(\vartheta)C_B \\
\dot{\vartheta} &= \frac{\dot{V}}{V_R}(\vartheta_0 - \vartheta) + \frac{k_W A_R}{\rho C_P V_R}(\vartheta_K - \vartheta) - \frac{\mathrm{I}}{\rho C_P}\left(k_1(\vartheta)C_A\Delta H_{R_{AB}} + k_2(\vartheta)C_B\Delta H_{R_{BC}} + k_4(\vartheta)C_A^2\Delta H_{R_{AD}}\right) \\
\dot{\vartheta}_K &= \frac{1}{m_K C_{P_K}}\left(\dot{Q}_K + k_W A_R(\vartheta - \vartheta_K)\right) \\
y &= (C_B, \vartheta)
\end{aligned}$$

due to Klatt and Engell [1], which describes a continuous stirred tank reactor with consecutive and side reaction. $k_i(\vartheta)$ are temperature dependant reaction rate constants. All parameters are more or less uncertain and the concentration of the inflow C_{A0} varies between 4 and 6 mol/l. The relatively largest uncertainty is in $\Delta H_{R_{AB}}$. So this parameter as well as the widely unknown inflow were accounted for by two additional equations.

$$\Delta \dot{H}_{R_{AB}} = 0 \quad \text{and} \quad \dot{C}_{A0} = 0$$

The initial values and initial guess, respectively, were set to

$$\begin{pmatrix} C_A \\ C_B \\ \vartheta \\ \vartheta_K \\ \Delta H_{R_{AB}} \\ C_{A0} \end{pmatrix} = \begin{pmatrix} 1.23 \\ 0.8 \\ 134 \\ 134 \\ 4.2 \\ 5.5 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} C_A \\ C_B \\ \vartheta \\ \vartheta_K \\ \Delta H_{R_{AB}} \\ C_{A0} \end{pmatrix}_{guess} = \begin{pmatrix} 1.5 \\ 0.8 \\ 134 \\ 130 \\ 5.2 \\ 5.0 \end{pmatrix}$$

Figure 1 depicts the total error of the state observation when the disturbance and the parameter were not (a) and were observed (b), Figure 2 depicts the observation and the true values of $\Delta H_{R_{AB}}$ (a) and C_{A0} (b).

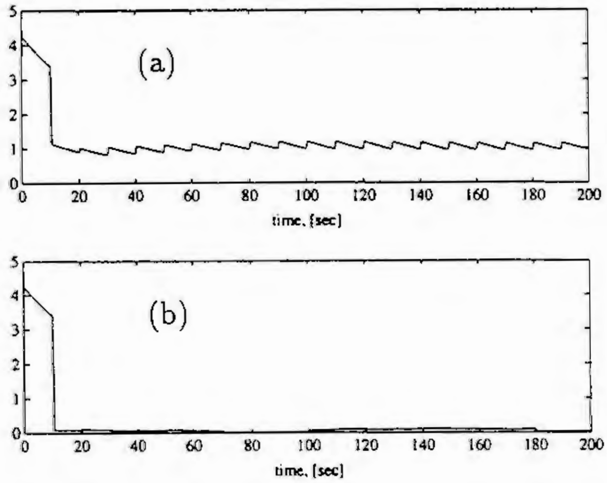


Figure 1

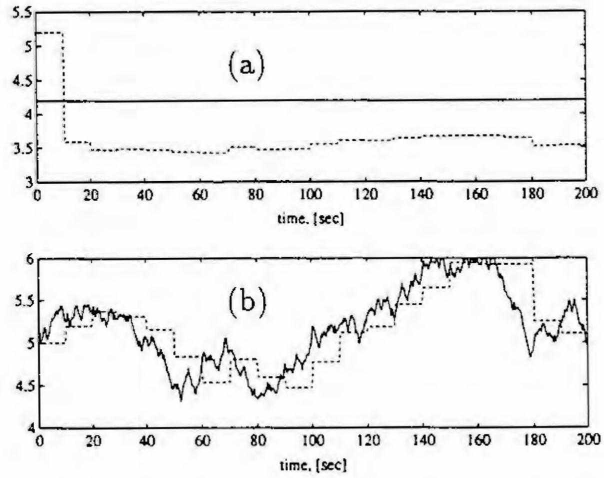


Figure 2:
true values: (———), observed values: (- - -)

7. Conclusions

In this paper a numerical method to observe slowly varying disturbances in nonlinear systems was introduced. The disturbance is modelled as a constant and augments the state vector. The restriction is that the resulting system has to have a twice continuously differentiable state function and an observable linearization. If the resulting system is sufficiently close to the original system, a disturbance can be approximated by a piecewise constant function.

The proposed method is very flexible and applicable for a large class of nonlinear systems. An application result was given by the model of a continuous stirred tank reactor with consecutive and side reaction.

8. References

[1] Klatt, K.-U., and Engell, S., 1993, VDI-Bericht Nr. 1026, 101 - 108.

[2] Hartman, Ph., 1964, Ordinary Differential Equations, (New York: John Wiley & Sons).

[2] Kosmol, P., 1989, Methoden zur numerischen Behandlung nichtlinearer Gleichungen und Optimierungsaufgaben, (Stuttgart: Teubner).

[4] Prenter, P. M., 1975, Splines and Variational Methods, (New York: Wiley and Sons).

[5] Zimmer, G., 1991, Preprint Nr. 304, Fachbereich Mathematik der TU Berlin.

[6] Zimmer, G., 1993, Proc. ECC 93, 391-396.

A Shrink-Fit-Model for Dynamical Load Transmission

J.Braun, F.Pfeiffer

Lehrstuhl B für Mechanik, Technische Universitat München,
80290 München, Germany

Abstract. Shrink fits are often used as machine elements in drive trains. Their influence on the system dynamics have not been taken into account so far. Based on the mechanism of load transfer in shrink fits a mechanical model for a shrink fit is developed taking into account the dynamical behaviour. The reduction of the model to a nonlinear force element is shown.

1 INTRODUCTION

The increasing demands for accurate modelling of vibration and noise in drive trains as well as the necessity for determining dynamical loads on machine elements require precise descriptions of all relevant physical effects. Fig. 1 shows a steering mechanism of a Diesel engine as it was investigated in [5]. The mechanical model used in [5] describes the dynamical behaviour of the steering mechanism very well if the gear stiffness values, which are calculated usually by the approach of Ziegler [6] or by DIN 3990 [2], are reduced by a factor of approximately six. This fact is shown in Fig. 2 for the example of the rotational vibration of the camshaft gear wheel.

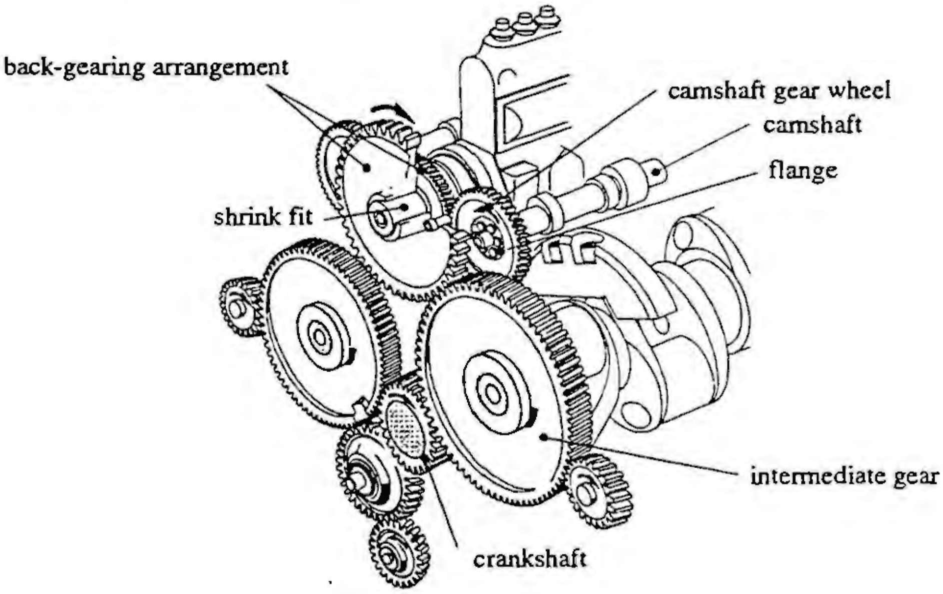


Figure 1: Steering mechanism of a Diesel engine

The frequency spectrum calculated with reduced gear stiffness values is in good agreement with the measurement whereas the result received by using Zieglers stiffness values deviates distinctly.

In [5] the shrink fit as well as the flange (Fig. 1) are assumed to be rigid. We now assume that the dynamical behaviour of these machine elements explains the stiffness reduction, which is necessary to bring the simulation results in good agreement with measurements. Therefore, the dynamical behaviour of shrink fits and flanges has to be studied. In this paper only shrink fits are investigated.

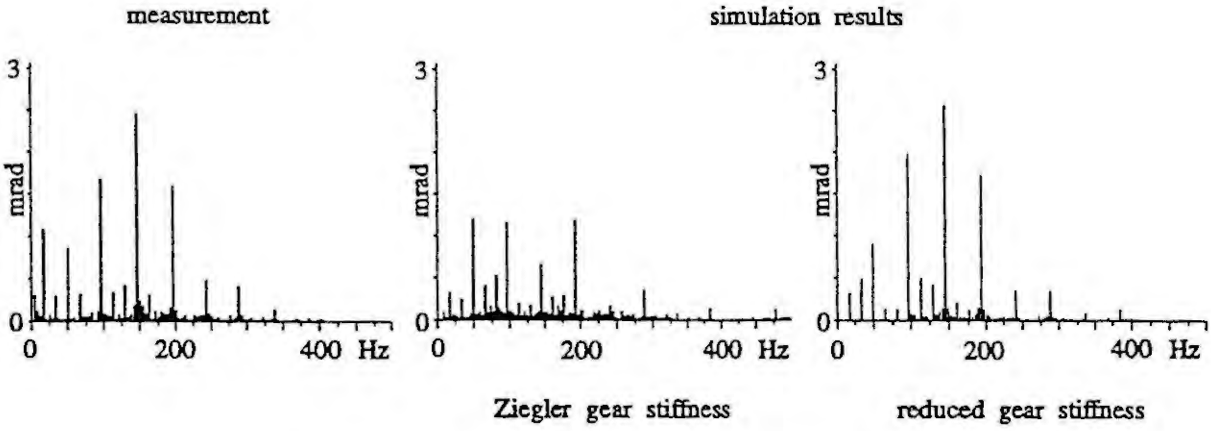


Figure 2: Frequency spectrum of the rotational vibration of the camshaft gear wheel

2 MODEL OF A SHRINK FIT

The model has to reproduce the mechanism of torque transfer as it is described in [4]. For small loads the transmission of torques in shrink fits is linear elastic. If the load increases local slip occurs at the beginning of the joint surface where the shaft goes into the hub. The torque at the transition from no slip to first slip is called the elastic limit torque M_E [4], the corresponding deformation φ_E results from the elastic deformation of the shaft and the hub. For a further increase of the load the slip zone grows to a load dependent depth. The rest of the joint surface sticks. If the load reaches the slide torque M_S, the whole joint surface slips and the shrink fit has failed. The model representing these three ranges of torque transfer is shown in Fig 3.

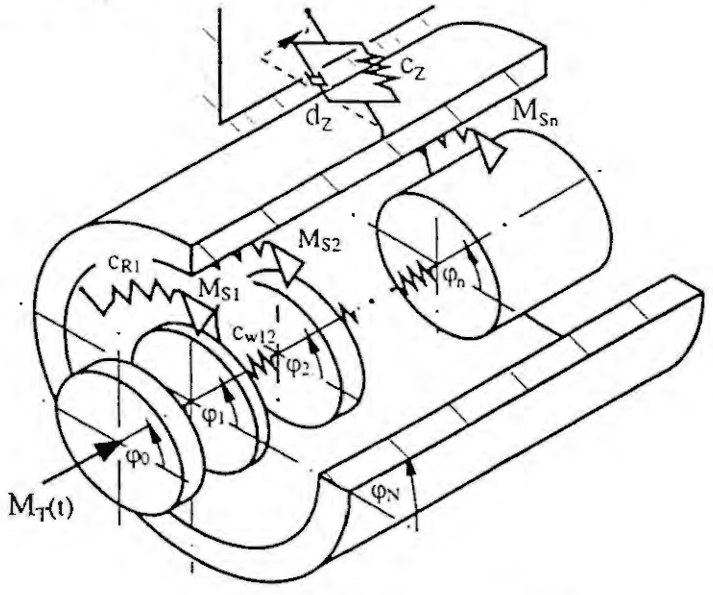


Figure 3: Dynamical model of a shrink fit

The shaft is divided in shaft disks connected by linear elastic springs. The hub is modelled as a rigid body. Both bodies are connected by a joint-friction element [3]. This force element consists of a friction element in series with an elastic spring. The slide torque of every force element is determined from the average shrink pressure at the corresponding shaft disk. The stiffness of the force element repesents the elasticity of the hub and is determined by the elastic limit load and the corresponding relative angle. The hub is connected to the enviroment by a spring-damper element representing the gear stiffness and damping. The equations of motion of this dynamical system have the well known form

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{D}\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} = \mathbf{h}(\mathbf{q}, \text{history}, t).$$

The vector of the generalized coordinates q contains the torsion angles of the shaft disks φ_i and the hub φ_N. The diagonal mass matrix M is constant and positive definite. The damping matrix D includes only the viscous damping coefficient d_Z of the gear tooth system whereas the stiffness matrix K contains the stiffness of the gear tooth system c_Z as well as the stiffness of the springs between the shaft disks c_{wij}. The vector h on the right hand side contains the nonlinear friction element momentums and the external load $M_T(t)$.

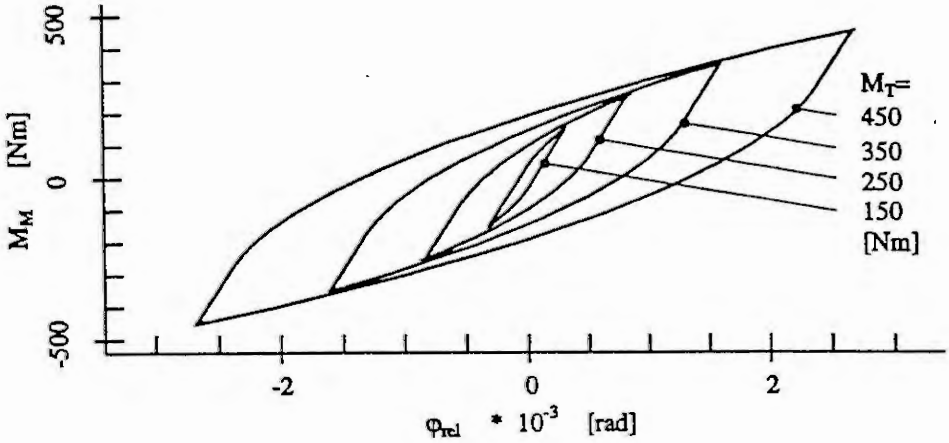


Figure 4: Hysteresis loops for different load amplitudes

The solution of the equations of motion has to be done by numerical integration due to the nonlinearity of the friction elements. A Runge-Kutta method of second and third order with step size control together with an algorithm to determine the transition points between the sticking and the sliding state of the friction elements is used. Fig. 4 shows the the transfered torque M_M, which is the sum of the transfered torques in the friction elements, over the relative angle $\varphi_{rel} = \varphi_N - \varphi_0$. The hysteresis loops increase with increasing excitation torque amplitudes M_T. A view at one hysteresis loop shows that for decreasing load the complete model sticks so that a linear part in the hysteresis loop occurs. If the maximum load decreases by two times the elastic limit torque, the slope of the hyseresis loop starts to decrease. The reason for this behaviour is that the first shaft disk starts to slip. For a further decreasing torque the second shaft disk starts slipping and so on.

3 FORCE TRANSFER ELEMENT

It is obvious that the eigenfrequencies of a sticking shrink fit are very high in comparison with the frequency range of the transfered torques ($0-400Hz$). Therefore, mass effects are negligible so a reduction of the model (Fig. 3) to a force transfer element is possible.

The element connects the rigid shaft and hub by a nonlinear, continuous torque - relative angle relation. It can be interpreted as a linear spring in series with a nonlinear spring with decreasing stiffness with its deflection. The characteristic for the degressive stiffness is determined as follows. If the load exceeds the elastic limit torque M_E, the stiffness of the model starts to decrease continuously. For an external load of the magnitude of the slide torque M_S, the deflection of the element must be the maximum relative angle φ_S [1]. These requirements supply three conditions to determine a square root function. The function has to go through the points (M_E, φ_E) and (M_S, φ_S). The gradient of the function at the transition form sticking to sliding is M_E divided by φ_E. If the applied load M_M decreases, a sticking phase occurs. After an unloading of twice the elastic limit load M_E, sliding in opposite direction begins. This transition point is the starting point for a new square root function. Therefore, if a transition from sticking to sliding in the opposite direction as before occurs a new square root function has to be determined.

In Fig. 6 the simulated hysteresis loops for the reduced shrink fit modell are shown. The four constant force transfer element parameter $(M_E, \varphi_E, M_S, \varphi_S)$ are choosen equal to the values used for calculating the results of Fig. 4. The comparison shows no significant difference. This means that the reduced model is a very good approximation for the shrink fit model (Fig. 3).

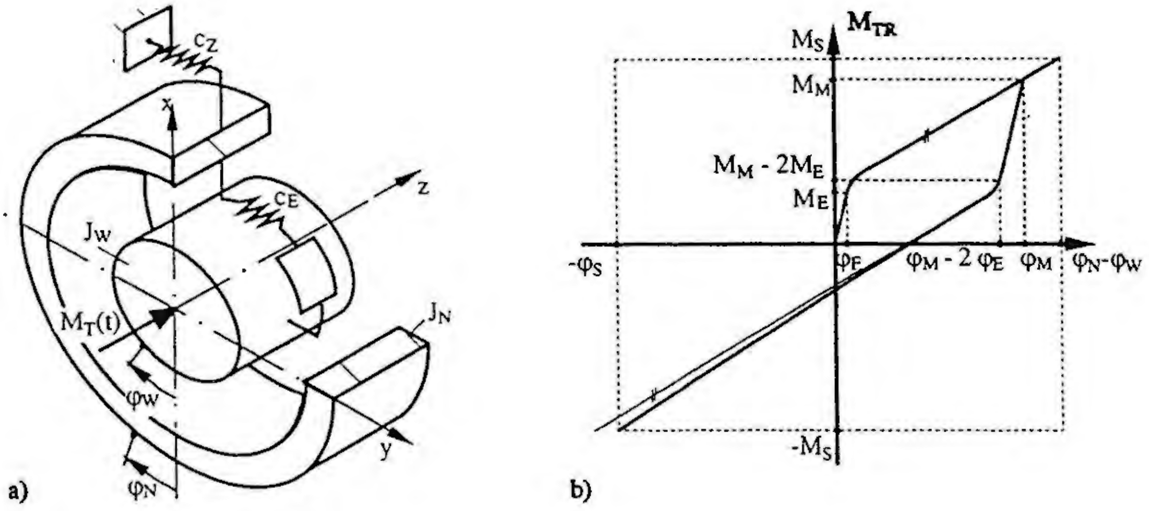


Figure 5: a) Reduced shrink fit model
b) Torque - relative angle relation

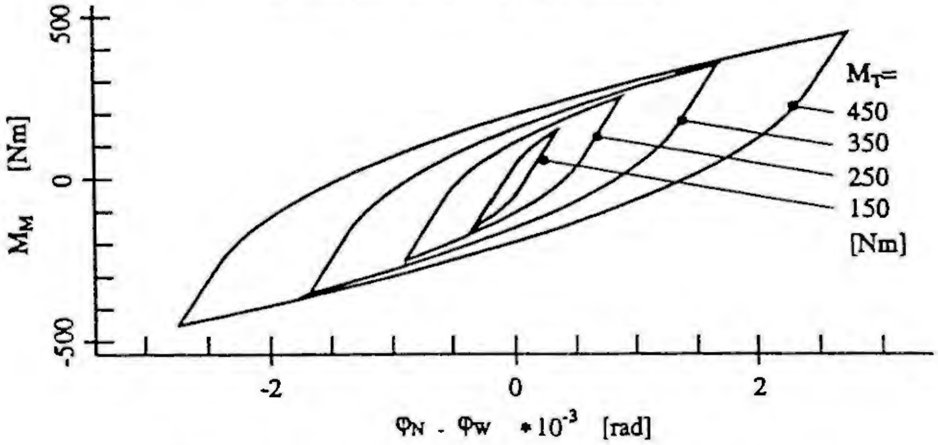


Figure 6: Simulation results of the reduced shrink fit model

4 RESULTS

The mechanism of torque transfer in a shrink fit is discussed and a mechanical model representing the behaviour of shrink fits is derived. A reduced shrink fit model approximates the detailed shrink fit model very well. The model parameters are independent of the applied load and can be determined approximately by simple calculations. The verification of the presented model is the next task.

REFERENCES

[1] Braun, J. Pfeiffer, F., *Dynamical Behaviour of Shrink Fits*. To appear in: R. Bogacz, K.Popp (Ed.), Dynamical Problems in Mechanical Systems, Proceedings of the 3rd Polish-German Workshop, July 26-31, 1993, Wierzba.

[2] DIN 3990, *Tragfähigkeitsberechnung von Stirnrädern*, Berlin, 1987.

[3] Iwan, W.D., *A Distributed-Element Model for Hysteresis and Its Steady-State Dynamic Response*. Journal of Applied Mechanics, 1966, 893-900.

[4] Müller, H.W., *Drehmoment-Übertragung in Preßverbindungen*. Konstruktion, Band 14, Heft 2, 1992.

[5] Prestl, W., *Zahnhämmern in Rädertrieben von Dieselmotoren*. VDI Fortschritts-Berichte, Reihe 11, Nr. 145, VDI-Verlag, Düsseldorf, 1991.

[6] Ziegler, H.,*Verzahnungssteifigkeit und Lastverteilung schrägverzahnter Stirnräder*. Dissertation, RWTH Aachen, Mai 1971.

INDEPENDENT JOINT CONTROL: ESTIMATION AND COMPENSATION OF COUPLING AND FRICTION EFFECTS IN ROBOT POSITION CONTROL

R. HU, P.C. MÜLLER

Safety Control Engineering, University of Wuppertal
Gauß Str. 20, D-42097 Wuppertal, Germany

Abstract. The method of independent joint control has been widely used in the position control of industrial robots. In order to improve the control performance of this type of controllers, the concept of nonlinearity-estimation and -compensation is introduced. With this extended method comparable results can be obtained as with the method of exact linearization. Especially by the treatment of unmodeled or inaccurate effects, e.g. friction, load variation or parameter inaccuracy, the presented control concept shows great advantages.

1. INTRODUCTION

Today's industrial robots are almost exclusively equiped with independent joint controllers for the position control, although the robot dynamics are highly nonlinear e.g. due to the position-dependent inertia moments and the coupling effects through the Coriolis moments. Therefore, for the improvement of the control performance different control concepts to decouple and compensate the nonlinear dynamics have been developed in the robotics research since years [1]. Such concepts can be subdivided into two groups. On the one hand multivariable controllers based on the multibody models of robot dynamics are designed, e.g. with the help of the method of exact linearization through state feedback. On the other hand the structure of independent joint control is kept and the nonlinear effects are compensated through feedforwarding the 'computed torques' obtained from desired trajectories or through the feedback of joint torques which are measured at the driven sides of each robot axis ('joint torque control') . Alternative to these the method of nonlinearity-estimation and -compensation [2] can be used, in which the disturbance torques are estimated with independent joint observers and compensated through a corresponding feedback.

Of the two model-based methods the method of exact linearization is methodically more precise than the computed torque method, although it is more involving due to the on-line state feedback than the latter, in which the required feedforwarding torques can be computed off-line. In both cases, however, a complete knowledge of models is assumed. Parameter inaccuracies and incompletely known friction effects lead herewith to a need for a robust control design. These problems disappear however with the method of joint torque control or with the method introduced here. While the method of joint torque control needs additional measurement devices, the control method introduced in this paper works with the usual measurements and the necessary information for the compensation is obtained by observers.

This observer-based control concept was first developed by Müller and Ackermann [3] for the compensation of friction effects in the position control of robots. In the meantime it was also used in [4-6]. A general and fundamental description of this method can be found in [2]. Based on the results of [7], in which this method was systematically used for the development of a decentralized robust position controller of robots on the basis of rigid body dynamic models, we will investigate in this paper a more complicated case, in which the joint elasticity and motor dynamics are also considered. Like [7] we use simple simulation examples to compare our method with the method of exact linearization.

2. DYNAMIC MODEL OF ROBOTS

An elastic joint robot together with its motor dynamics can be modeled according to [8, 9] as:

$$\begin{aligned} \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{K}(\mathbf{q}-\mathbf{p}) &= 0, & \text{(1a)} \\ \mathbf{J}\ddot{\mathbf{p}} - \mathbf{K}(\mathbf{q}-\mathbf{p}) &= \mathbf{m}, & \text{(1b)} \\ \mathbf{T}\dot{\mathbf{m}} + \mathbf{m} &= \mathbf{G}\mathbf{u}, & \text{(1c)} \end{aligned}$$

in which $\mathbf{q}$ is the vector of joint coordinates, $\mathbf{M}(\mathbf{q})$ the positive definite mass matrix, $\mathbf{h}(\mathbf{q},\dot{\mathbf{q}})$ the Coriolis and centripetal as well as the gravitational forces. The vector $\mathbf{p}$ defines the motor angels relative to the gear ratios, $\mathbf{J}$ and $\mathbf{K}$ are the diagonal matrices which stand for the effective moments of inertia of the motors and the stiffnesses between motor and robot arm respectively. $\mathbf{m}$ are the drive torques of the motors, $\mathbf{T}$ the time constants and $\mathbf{G}$ the torque gains. $\mathbf{u}$ is the vector of input voltages of the motors.

If the effects of friction are also considered, then they are included in terms of their corresponding torques in the description of $\mathbf{h}(\mathbf{q},\dot{\mathbf{q}})$.

3. CONTOLLER DESIGN

3.1. Exact Linearization

According to [9] controllers can be designed with the method of exact linearization based on (1). Using the joint coordinates $\mathbf{q}$ as output variables, a system decoupling can be explicitely given as

$$\mathbf{q}^{(5)} = \mathbf{a}_6(\mathbf{q},\dot{\mathbf{q}},\mathbf{p},\dot{\mathbf{p}},\mathbf{m}) + \mathbf{M}^{-1}(\mathbf{q})\mathbf{K}\mathbf{J}^{-1}\mathbf{T}^{-1}\mathbf{G}\mathbf{u}, \tag{2}$$

where the function $\mathbf{a}_6$ is obtained from differentiations of (1). Chosing the drive voltages $\mathbf{u}$ as

$$\mathbf{u} = \mathbf{G}^{-1}\mathbf{T}\mathbf{J}\mathbf{K}^{-1}\mathbf{M}(\mathbf{q})(\mathbf{v} - \mathbf{a}_6), \tag{3}$$

a decoupled linear system

$$\mathbf{q}^{(5)} = \mathbf{v} \tag{4}$$

has been obtained. Here $\mathbf{v}$ is the new input vector. If e.g. a desired trajectory $\mathbf{r}(t) = \mathbf{q}_d(t)$ is to be realized, it can thus be reached through the feedback

$$\mathbf{v} = \mathbf{q}_d^{(5)} + \mathbf{k}_{q^{(4)}}(\mathbf{q}_d^{(4)} - \mathbf{q}^{(4)}) + \mathbf{k}_{q^{(3)}}(\mathbf{q}_d^{(3)} - \mathbf{q}^{(3)}) + \mathbf{k}_{\ddot{q}}(\ddot{\mathbf{q}}_d - \ddot{\mathbf{q}}) + \mathbf{k}_{\dot{q}}(\dot{\mathbf{q}}_d - \dot{\mathbf{q}}) + \mathbf{k}_q(\mathbf{q}_d - \mathbf{q}), \tag{5}$$

in which $\mathbf{k}_{q^{(4)}}$, $\mathbf{k}_{q^{(3)}}$, $\mathbf{k}_{\ddot{q}}$, $\mathbf{k}_{\dot{q}}$ and $\mathbf{k}_q$ are diagonal matrices with positive diagonal elements which determine the dynamics of each joint controllers.

This procedure is demonstrated using the following simple simulation example.

Example 1: A one-axis robot is considered, the arm of which carries out a pendulum motion. According to (1), its dynamic model is described by

$$\begin{aligned} M_0\ddot{q} + m_0 gL\sin q + K(q-p) &= 0, & \text{(6a)} \\ J\ddot{p} - K(q-p) &= m, & \text{(6b)} \\ T\dot{m} + m &= Gu. & \text{(6c)} \end{aligned}$$

Here $M_0 = 1\,Nms^2$ and $J = 1\,Nms^2$ are the moments of inertia of the arm and the motor respectively, $m_0 = 1\,kg$ is the mass of the arm and $L = 1\,m$ the distance between the joint axis and the mass center of the arm. The gravitational acceleration is taken as $g = 9.81\,ms^{-2}$. Additionally, we have the joint elasticity $K = 10^2\,Nm$ and the time constant $T = 10^{-2}\,s$ as well as the torque gain of the motor $G = 0.3\,NmV^{-1}$. As desired trajectory for the robot simulation a motion with

$$q_d(t) = \begin{cases} 0 & 0 \le t < 0.25 \\ (t-0.25)^2 & 0.25 \le t < 0.75 \\ 0.25 + (t-0.75) & 0.75 \le t < 1.25 \\ 0.75 + (t-1.25) - (t-1.25)^2 & 1.25 \le t < 1.75 \\ 1 & 1.75 \le t < 2 \end{cases} \tag{7}$$

is assumed, see also Fig. 1a.

According to (3-5) the controller is chosen as

$$u = \frac{TM_0J}{GK}(v - a_6) \tag{8a}$$

with

$$v = q_d^{(5)} + k_{q^{(4)}}(q_d^{(4)} - q^{(4)}) + k_{q^{(3)}}(q_d^{(3)} - q^{(3)}) + k_{\ddot{q}}(\ddot{q}_d - \ddot{q}) + k_{\dot{q}}(\dot{q}_d - \dot{q}) + k_q(q_d - q) \tag{8b}$$

and

$$\begin{aligned} a_6 \;=\; & \frac{m_0gL}{M_0}\sin q\,(-3\frac{m_0gL}{M_0}\sin q\,\dot{q}) + \frac{m_0gL}{M_0}\cos q\,\dot{q}(\dot{q}^2 + \frac{m_0gL}{M_0}\cos q + \frac{K}{M_0}) \\ & + \frac{K}{M_0}(q - p)(-3\frac{m_0gL}{M_0}\sin q\,\dot{q}) + \frac{K}{M_0}(\dot{q} - \dot{p})(\frac{K}{M_0} + \frac{K}{J} + \frac{m_0gL}{M_0}\cos q\,) \\ & + (-\frac{K}{TM_0J}m). \end{aligned} \tag{8c}$$

Chosing the eigenvalues of the control system as $\lambda_i = -15\,s^{-1}, i = 1, 2, \cdots, 5$, with an initial condition $q(0) = 0$, $\dot{q}(0) = 0$, $p(0) = 0$, $\dot{p}(0) = 0$ and $m(0) = 0$ a control error $q_d(t) - q(t)$ is obtained according to Fig. 1b. The control error comes essentially from the numerical differentiations within (8).

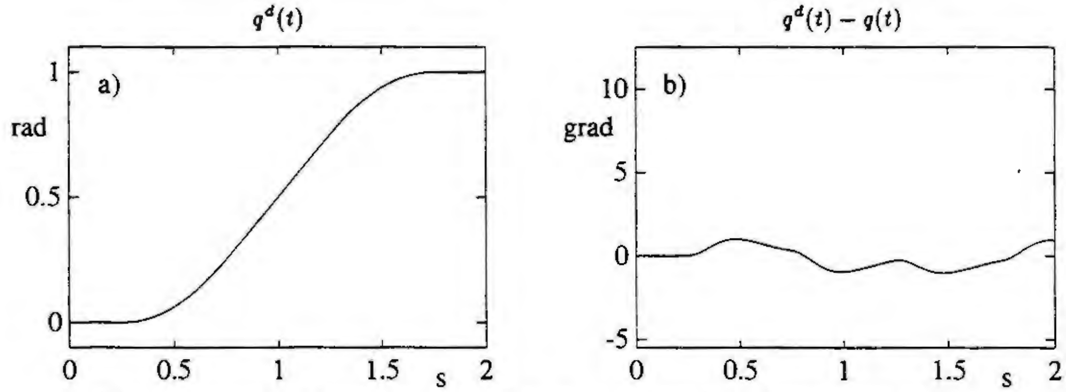


Fig. 1. Control performance of the robot with the method of exact linearization:
a) Desired trajectory, b) Control error

3.2. Nonlinearity-Estimation and -Compensation

3.2.1. State space description

Starting-point for the improved design of joint controllers is the coupled system (1), which is now written down relative to the single axes. The mass matrix is divided into a constant diagonal matrix of the mean values of the moments of inertia and a remaining position-dependent part,

$$\mathbf{M}(\mathbf{q}) = \mathbf{M}_0 + \Delta\mathbf{M}(\mathbf{q}). \tag{9}$$

The mean values are chosen with regard to a typical work space of the robot or along a desired trajectory. Summerizing further for each axis i all nonlinear terms in n_i,

$$n_i = \sum_j \Delta M_{ij}(\mathbf{q})\ddot{q}_j + h_i(\mathbf{q}, \dot{\mathbf{q}}), \tag{10}$$

equation (1) can be seperately considered for the single axis:

$$\begin{aligned} M_{0i}\ddot{q}_i + n_i + K_i(q_i - p_i) &= 0, && (11a) \\ J_i\ddot{p}_i - K_i(q_i - p_i) &= m_i, && (11b) \\ T_i\dot{m}_i + m_i &= G_iu_i. && (11c) \end{aligned}$$

This one-axis model can then be described correspondingly in state space. Leaving out the index i for the sake of brevity, the desciption

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{N}n + \mathbf{B}u, \quad (12a)$$

$$y = \mathbf{C}\mathbf{x} \quad (12b)$$

is obtained with the state vector $\mathbf{x} = [\, q \quad \dot{q} \quad p \quad \dot{p} \quad m\,]^T$ and the matrices

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ -\frac{K}{M_0} & 0 & \frac{K}{M_0} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ \frac{K}{J} & 0 & -\frac{K}{J} & 0 & \frac{1}{T} \\ 0 & 0 & 0 & 0 & -\frac{1}{T} \end{bmatrix}, \quad \mathbf{N} = \begin{bmatrix} 0 \\ -\frac{1}{M_0} \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \frac{G}{T} \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}^T. \quad (12c)$$

Here the measurement of the joint coordinate q is assumed.

The objective of the position control of robots is the tracking of joint coordinate $q(t)$ along a desired trajectory $r(t) = q_d(t)$ which is determined by path planning. The control error is

$$z = \mathbf{F}\mathbf{x} + Rr \quad (12d)$$

where

$$\mathbf{F} = [\, -1 \quad 0 \quad 0 \quad 0 \quad 0\,], \qquad R = 1. \quad (12e)$$

3.2.2. Tracking control and nonlinearity-compensation

The design of each joint controller is based on the model (12). The nonlinearities and coupling effects, which are contained in n, will be compensated with the method of nonlinearity-estimation and -compensation [2]. The tracking control is reached through a feedforward, which is methodically determined using the method of disturbance rejection control [10]. Altogether an asymptotically stable control with

$$z(t) \to 0 \qquad \text{for} \qquad t \to \infty \quad (13)$$

is the desired design objective.

The time signal n of the nonlinearities and couplings is approximated appropriately by time functions, which are themselves solutions of an adequately selected linear dynamic system

$$n(t) \approx \mathbf{H}_1\mathbf{v}_1(t), \qquad \dot{\mathbf{v}}_1(t) = \mathbf{V}_1\mathbf{v}_1(t). \quad (14a)$$

It was shown in [2] that this approximation can be best carried out with step functions, so that an integrator model is chosen in (14a):

$$\mathbf{H}_1 = 1, \qquad \mathbf{V}_1 = 0. \quad (14b)$$

The desired trajectory $r(t) = q_d(t)$ is assumed to be known. For the feedforwarding however, the derived variables $\dot{q}_d(t)$, $\ddot{q}_d(t)$, $q_d^{(3)}(t)$, $q_d^{(4)}(t)$ and $q_d^{(5)}(t)$ are needed, which although are theoretically available too, but do not exactly correspond to the derivations of the actually requested trajectory due to possible disturbance influences. Therefore, they are estimated by an observer which is constructed like (14a) as well, see [10]:

$$r(t) \approx \mathbf{H}_2\mathbf{v}_2(t), \qquad \dot{\mathbf{v}}_2(t) = \mathbf{V}_2\mathbf{v}_2(t). \quad (15a)$$

The approximation is based on step and ramp functions, so that

$$\mathbf{H}_2 = [\, 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0\,], \qquad \mathbf{V}_2 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (15b)$$

is selected.

The design of observers for the estimation of the signal n and $\dot{r}, \ddot{r}, r^{(3)}, r^{(4)}, r^{(5)}$ as well as $\dot{q}$, p and $\dot{p}$ is based on a linear system, which is obtained by inserting (14, 15) in (12):

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\mathbf{v}}_1 \\ \dot{\mathbf{v}}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{N}\mathbf{H}_1 & 0 \\ 0 & \mathbf{V}_1 & 0 \\ 0 & 0 & \mathbf{V}_2 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{v}_1 \\ \mathbf{v}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ 0 \\ 0 \end{bmatrix} u, \tag{16a}$$

$$\begin{bmatrix} y \\ r \end{bmatrix} = \begin{bmatrix} \mathbf{C} & 0 & 0 \\ 0 & 0 & \mathbf{H}_2 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{v}_1 \\ \mathbf{v}_2 \end{bmatrix}, \quad . \tag{16b}$$

$$z = \begin{bmatrix} \mathbf{F} & 0 & R\mathbf{H}_2 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{v}_1 \\ \mathbf{v}_2 \end{bmatrix}. \tag{16c}$$

This extended system with the given system matrices is completely observable. Therefore, an observer can be designed according to one of the usual methods. Here a quadratic optimal identity observer is determined using [11]. It is shown at the same time that the observer can be seperated into two parts for $n, \dot{q}, p, \dot{p}$ and $\dot{r}, \ddot{r}, r^{(3)}, r^{(4)}, r^{(5)}$:

$$\begin{bmatrix} \dot{\hat{\mathbf{x}}} \\ \dot{\hat{\mathbf{v}}}_1 \end{bmatrix} = \begin{bmatrix} \mathbf{A} - \mathbf{L}_x\mathbf{C} & \mathbf{N}\mathbf{H}_1 \\ -\mathbf{L}_{v1}\mathbf{C} & \mathbf{V}_1 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}} \\ \hat{\mathbf{v}}_1 \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ 0 \end{bmatrix} u + \begin{bmatrix} \mathbf{L}_x \\ \mathbf{L}_{v_1} \end{bmatrix} y, \tag{17}$$

$$\dot{\hat{\mathbf{v}}}_2 = (\mathbf{V}_2 - \mathbf{L}_{v2}\mathbf{H}_2)\hat{\mathbf{v}}_2 + \mathbf{L}_{v2}r. \tag{18}$$

The desired estimated values are obtained from (17, 18), e.g. $\hat{n} = \mathbf{H}_1\hat{\mathbf{v}}_1$.

With the estimated variables $\hat{\mathbf{x}}$, $\hat{\mathbf{v}}_1$ and $\hat{\mathbf{v}}_2$ a feedback

$$u = -\mathbf{K}_x\hat{\mathbf{x}} - \mathbf{K}_{v1}\hat{\mathbf{v}}_1 - \mathbf{K}_{v2}\hat{\mathbf{v}}_2 \tag{19}$$

is constructed. The gain matrix $\mathbf{K}_x$ of the state feedback can be set with standard methods like pole assignment (the complete controllability of the matrices $(\mathbf{A}, \mathbf{B})$ is fulfilled in the present case). The gain matrix $\mathbf{K}_{v1}$ for the compensation of the nonlinearities and coupling effects and the gain matrix $\mathbf{K}_{v2}$ for the feedforwarding of the desired trajetory are determined from the equations

$$(\mathbf{A} - \mathbf{B}\mathbf{K}_x)\mathbf{X}_1 - \mathbf{X}_1\mathbf{V}_1 - \mathbf{B}\mathbf{K}_{v1} = -\mathbf{N}\mathbf{H}_1, \tag{20a}$$

$$\mathbf{F}\mathbf{X}_1 = 0, \tag{20b}$$

$$(\mathbf{A} - \mathbf{B}\mathbf{K}_x)\mathbf{X}_2 - \mathbf{X}_2\mathbf{V}_2 - \mathbf{B}\mathbf{K}_{v2} = 0, \tag{21a}$$

$$\mathbf{F}\mathbf{X}_2 = -R\mathbf{H}_2. \tag{21b}$$

Here the problem is also seperated into two parts. $\mathbf{X}_1$ and $\mathbf{X}_2$ are secondary matrices, which characterize the stationary behaviour of $\mathbf{x}(t)$ depending on $\mathbf{v}_1(t)$ and $\mathbf{v}_2(t)$. However, the solutions of $\mathbf{K}_{v1}$, $\mathbf{K}_{v2}$ are of primary interest. They result in

$$\mathbf{K}_{v1} = -\frac{1}{K}K_{x3} - (\frac{1}{G} + K_{x5}), \qquad \mathbf{K}_{v2} = \begin{bmatrix} -K_{x1} - K_{x3} \\ -K_{x2} - K_{x4} \\ -\frac{M_0}{K}K_{x3} - (\frac{1}{G} + K_{x5})(M_0 + J) \\ -\frac{M_0}{K}K_{x4} - \frac{T}{G}(M_0 + J) \\ -(\frac{1}{G} + K_{x5})\frac{M_0 J}{K} \\ -\frac{TM_0 J}{GK} \end{bmatrix}^T . \tag{22}$$

With that, the controller is finally obtained.

3.2.3. Robustness of the controller

The robustness of the controller (19) can easily be verified. If the parameters in the system description (1) are inaccurate, e.g. due to the varied loads, or unknown friction torques appear, the real system behaviour is then described with a modified model

$$\begin{aligned} \mathbf{M}'(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}'(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{K}(\mathbf{q}-\mathbf{p}) &= \mathbf{0}, & (23a)\\ \mathbf{J}\ddot{\mathbf{p}} - \mathbf{K}(\mathbf{q}-\mathbf{p}) &= \mathbf{m}, & (23b)\\ \mathbf{T}\dot{\mathbf{m}} + \mathbf{m} &= \mathbf{G}\mathbf{u}. & (23c) \end{aligned}$$

Whereas the controller (3-5) is still based on the nominal model (1) so that the mismatch problem with its possible sensitivity is present, the controller (19) can completely react upon the modified system behaviour. Let

$$\mathbf{M}'(\mathbf{q}) = \mathbf{M}_0 + \Delta\mathbf{M}'(\mathbf{q}), \tag{24}$$

then one has only to replace the term n_i in the joint axis description (11) by

$$n_i' = \sum_j \Delta M_{ij}'(\mathbf{q})\ddot{q}_j + h_i'(\mathbf{q},\dot{\mathbf{q}}). \tag{25}$$

As the design of the controller (19) is based on the fact that n_i - and thus n_i' - are to be interpreted as unknown variables, and hence to be estimated by the observer (17), the controller fulfils its task also by varied model data. The controller (19) is structurally robust against parameter inaccuracies and unmodeled effects.

4. SIMULATION EXAMPLES

Example 1 is again used for the simulations here, in which a one-axis robot, the motion of which is described by (6), is considered. The control behaviour of this system based on an ideal knowledge of the model and the controller (8) was shown in Fig. 1. Now the controller (19) has to be tested. The two observers (17, 18) are designed with the method described in [11], which guarantees a minimal quadratic estimation error. The gain matrix $\mathbf{K}_x$ in (19) is set according to the method of pole assignment for the poles $\lambda_i = -15\ s^{-1}, i = 1, 2, \cdots, 5$. Fig. 2 shows the simulation results. The ideal nonlinear term $n(t)$ and its estimation value $\hat{n}(t)$ are displayed in Fig. 2a. In Fig. 2b the control error $q_d(t) - q(t)$ is seen. It is recognized that the estimation follows the nonlinearity quantitatively very well, but with a certain phase delay and because of this the control error is also bigger than the result in Fig. 1.

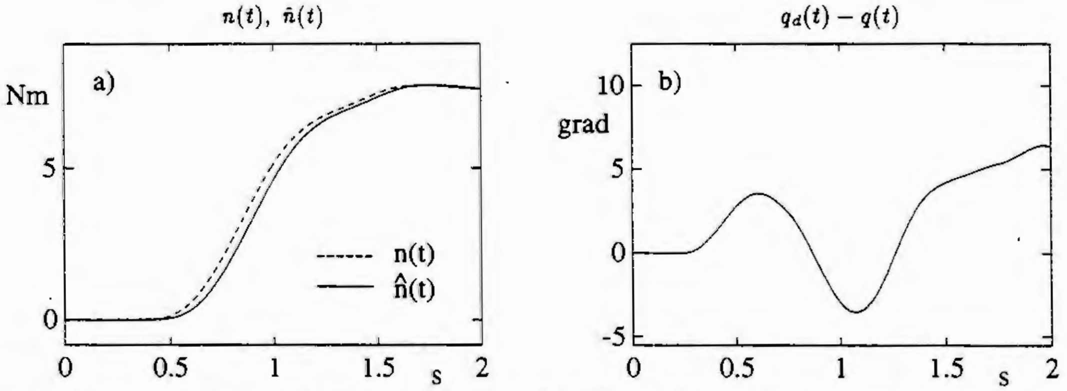


Fig. 2. Control performance of the example robot (6) with nonlinearity-estimation and -compensation: a) Nonlinearity-estimation, b) Control error

The advantages of the controller (19) are recognized, only when the real robot problem with inaccurate parameters and/or with neglected effects is regarded. Considering e.g. the influnce of friction effects of the form

$$F(\dot{q}) = \begin{cases} F_0\,\mathrm{sgn}(\dot{q}) + f\dot{q} & \dot{q} \neq 0 \\ -m_0 g L \sin q - K(q-p) & \dot{q} = 0 \end{cases} \tag{26}$$

with $F_0 = 2\ Nm$ und $f = 2\ Nms$, the behaviour of the robot arm is described by

$$\begin{aligned} M_0\ddot{q} + m_0 gL \sin q + F(\dot{q}) + K(q-p) &= 0, & \text{(27a)}\\ J\ddot{p} - K(q-p) &= m, & \text{(27b)}\\ T\dot{m} + m &= Gu & \text{(27c)} \end{aligned}$$

instead of (6). The controller (8) and (19) remain however unchanged. The control errors obtained from simulations are presented in Fig. 3, where a) is for the controller (8) and b) for the controller (19). It is seen that the second controller with nonlinearity-estimation and -compensation is more satisfying.

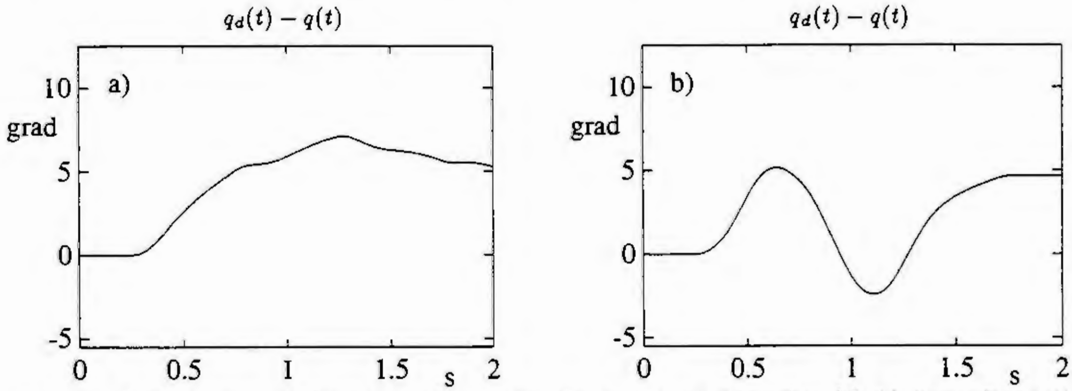


Fig. 3. Control errors of the robot with friction effects (27): a) Controller (8), b) Controller (19)

If load variation and parameter inaccuracies appear in additon to friction effects, e.g. with $M_0' = 1,5\ Nms^2$ and $m_0' = 1,5\ kg$, see also (23), so that $\Delta M_0' = 0,5$ and $\Delta m_0' = 0,5\ kg$, then the results in Fig. 4 are obtained with the two different controllers, which again illustrate the advantages of the proposed controller (19).

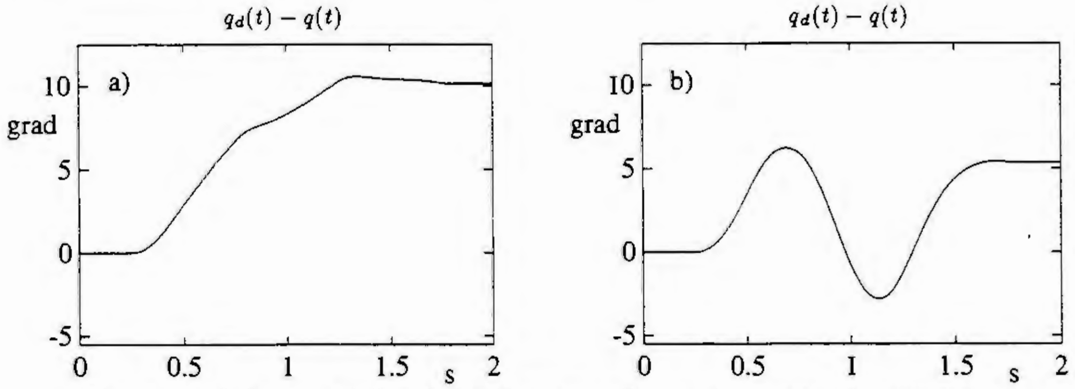


Fig. 4. Control errors of the robot with friction effects and parameter inaccurrancies:
a) Controller (8), b) Controller (19)

5. CONCLUDING REMARKS

The method of nonlinearity-estimation and -compensation has been proved to be a suitable approach for the design of robust position controllers for robots. In addition to its robustness, its decentral structure offers additional advantages, where the concept of independent joint control can further be used. This method has also demonstrated its good properties in practical applications [2, 12]. The requirements on the modeling of the robot dynamics are very low.

A number of questions remain to be answered however. For example, the setting of the gain matrices for both controller and observer should be discussed in consideration of measurement noises and possible saturations of input devices. The use of reduced-order observers instead of identity observer may occasionally bring some advantages. Further investigations will be referred to simulations of more complicated robot systems, e.g. robots with multi degrees of freedom. Additionally, the use of controllers based on simplified rigid body models for the control of elastic joint robots will be analysed.

REFERENCES

[1] Kuntze, H.-B, Regelungsalgorithmen für rechnergesteuerte Industrieroboter. Regelungstechnik 32 (1984), 215-226.

[2] Müller, P.C., Schätzung und Kompensation von Nichtlinearitäten. VDI-Bericht 1026, Nichtlineare Regelung - Methoden, Werkzeuge, Anwendungen, VDI-Verlag, Düsseldorf 1993, 199-208.

[3] Müller, P.C, Ackermann, J., Nichtlineare Regelung von elastischen Robotern. VDI-Berichte 598, Steuerung und Regelung von Robotern, VDI-Verlag, Düsseldorf 1986, 321-333.

[4] Schäfer, U., Brandenburg, G., Position Control for Elastic Pointing and Tracking Systems with Gear Play and Coulomb Friction and Application to Robots. Proc. 1991 IFAC Symposium on Robot Control, Wien 1991, 183-191.

[5] Nakao, M., Ohnishi, K., Miyachi, K., A Robust Decentralized Joint Control Based on Interference Estimation. Proc. 1987 IEEE Int. Conf. on Robotics and Automation, 326-331.

[6] Gorez, R., Galardini, D., Robot Control with Disturbance Observers. Proc. 5th Int. Conf. on Advanced Robotics, Pisa 1991.

[7] Hu, R., Müller, P.C., Robuste dezentrale Regelung von Robotern. VDI-Berichte 1094, Intelligente Steuerung und Regelung von Robotern, VDI-Verlag, Düsseldorf 1993.

[8] Spong, M.W., Vidyasagar, M., Robot Dynamics and Control. John Wiley & Sons, New York 1989.

[9] Müller, P.C., Modellvereinfachung nichtlinearer Systeme. VDI-Berichte 925, VDI-Verlag, Düsseldorf 1992, 161-188.

[10] Müller, P.C., Lückel, J., Zur Theorie der Störgrößenaufschaltung in linearen Mehrgrößensystemen. Regelungstechnik 25 (1977), 54-59.

[11] Müller, P.C., Truckenbrodt, A., Entwurf eines optimalen Beobachters. Regelungstechnik 25 (1977), 381-387.

[12] Neumann, R., Moritz, W., Observer-Based Decentralized Robot Joint Control. Proc. 2nd German-Polish Workshop on Dynamical Problems in Mechanical Systems, Paderborn 1991; reprinted in: Müller, P.C.(ed.), Bergisches Seminar für Robotik, Bergische Universität-GH Wuppertal 1991, 25-35.

Obtaining LFT descriptions of state-space models with parametric uncertainties

Paul Lambrechts
Mechanical Engineering Systems and Control Group
Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands
Jan Terlouw
Dutch National Aerospace Laboratory (NLR)
Anthony Fokkerweg 2, 1059 CM, Amsterdam, The Netherlands
Samir Bennani
Fac. of Aerospace Engineering, Section Stability and Control,
Delft University of Technology, Kluyverweg 1, 2629 HS Delft, The Netherlands
Maarten Steinbuch
Philips Research Laboratories
P.O.Box 80.000, 5600 JA Eindhoven, The Netherlands

Abstract. In this paper a general approach for modelling structured real-valued parametric perturbations is presented. It is based on a decomposition of perturbations into linear fractional transformations (LFTs), and is applicable to rational multi-dimensional polynomial perturbations of entries in state-space models. Model reduction is used to reduce the size of the uncertainty structure.

1 Introduction

In both robustness analysis and robust control system design the concept of the structured singular value μ as introduced by Doyle is of great importance [3]. It allows a high degree of detail in modelling the conditions under which the considered control system should operate satisfactorily, both in the sense of stability and performance. The calculation of μ for such models then results in a single number acting as an accurate measure in indicating whether the behaviour of the controlled system is satisfactory or not (see for instance [1, 5, 10, 11]).

Recently developed methods for calculating close upper and lower bounds for the most general cases ([6, 14]) now motivate the effort of modelling uncertainties in great detail. The main issue of this paper is to offer a complete procedure for setting up the general structure for the calculation of μ when uncertainties like real-valued parameter variations in state-space models and variations in operational conditions occur. This procedure has been implemented as a PC Matlab toolbox, offering easy to use interactive tools and a direct interface with the μ-Tools toolbox for μ analysis and design [15].

First, we will define Linear Fractional Transformations (LFTs), the structured singular value μ and some relevant uncertainty sets. Section 3 will then present a procedure for parametric uncertainty modelling based on a state-space model in which uncertain entries may be given as rational polynomial functions of a set of parameters. It will appear that this procedure can be interpreted as the solution to a multi-dimensional realization problem, also known as ND-realization problem [2]; Some concluding remarks follow in section 4.

2 Preliminaries

This section will review some of the properties of Linear Fractional Transformations (LFTs) and the structured singular value μ along the lines of Doyle et al. [4]. We will give a definition of upper and lower LFTs and discuss the LFT concept as a framework for uncertainty modelling. Within this framework we will give a definition of μ and some relevant uncertainty sets.

We will consider matrices with entries that are fractions of polynomials in a complex-valued variable s; the space of all such real rational functions will be denoted as $\mathrm{R}(s)$, $M \in \mathrm{R}(s)^{p\times q}$ will denote that M is a $p \times q$ matrix with entries in $\mathrm{R}(s)$. Suppose this matrix M is partitioned as:

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \quad \in \mathrm{R}(s)^{(p_1+p_2)\times(q_1+q_2)} \tag{1}$$

and let $\Delta_u \in \mathrm{R}(s)^{q_1 \times p_1}$ and $\Delta_l \in \mathrm{R}(s)^{q_2 \times p_2}$ be arbitrary. We will then define the *upper* and *lower* LFTs as operators on Δ_u and Δ_l respectively:

$$\begin{aligned} \mathcal{F}_u(M, \Delta_u) &:= M_{22} + M_{21}(I - \Delta_u M_{11})^{-1}\Delta_u M_{12} \\ \mathcal{F}_l(M, \Delta_l) &:= M_{11} + M_{12}(I - \Delta_l M_{22})^{-1}\Delta_l M_{21} \end{aligned} \tag{2}$$

Either LFT will be called *well defined* if the concerning inverse exists: $\det(I - \Delta_u M_{11}) \neq 0$ and $\det(I - \Delta_l M_{22}) \neq 0$. The matrix M is sometimes referred to as the coefficient matrix of the LFT. LFTs can be seen as operations resulting from feedback structures as given in fig.1; eq.2 then defines a closed loop transfer functions from w_M to z_M in both cases.

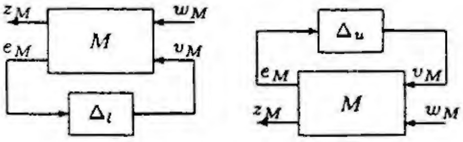


Fig. 1: Upper and lower LFT as feedback structure

An important reason for using the concept of LFTs in linear systems theory is that linear interconnections of LFTs can be rewritten as one single LFT. This implies that LFTs can be used to separately model specific details of the system under consideration after which a complete system description can be obtained by working out all connections.

Another main advantage of the LFT concept is that it provides a framework for uncertainty modelling. The coefficient matrix can be seen as the part of a linear model that is assumed to be correct: the nominal model then results as an LFT on $\Delta = 0$. By taking $\Delta \in \mathbf{\Delta}$ with $\mathbf{\Delta} \subset \mathrm{R}(s)^{q \times p}$ a given subspace, it is then possible to specify a set of linear models rather than a single one.

A measure for determining whether all models within this set are stable was introduced by Doyle [3] as the structured singular value or μ and is based on a block-diagonal structure of $\mathbf{\Delta}$:

$$\mathbf{\Delta} = \{\mathrm{diag}(\delta_1 I_{k_1}, \ldots, \delta_r I_{k_r}, \Delta_1, \ldots, \Delta_f) : \ \delta_i \in \mathrm{R}(s), \Delta_i \in \mathrm{R}(s)^{k_{r+i} \times k_{r+i}}\} \tag{3}$$

in which $\delta_i I_{k_i}, i = 1 \ldots r$ denote *repeated scalar* blocks and $\Delta_i, i = 1 \ldots f$ denote *full* blocks. As $\mathbf{\Delta} \subset \mathrm{R}(s)^{q \times p}$ we can consider the ∞-norm as a measure for the magnitude of an element of $\mathbf{\Delta}$ or one of its sub-blocks, with the ∞-norm of a matrix M defined as: $\|M\|_\infty := \sup_\omega \bar{\sigma}(M(j\omega))$ with $\bar{\sigma}$ denoting the largest singular value. We can then define the structured singular value as follows:

Definition 2.1 *Given a block-diagonal structure as in eq.3 and a compatible matrix* $M \in \mathrm{R}(s)^{p \times q}$. $\mu_{\mathbf{\Delta}}(M(j\omega))$ *is then defined as:* $\mu_{\mathbf{\Delta}}(M(j\omega)) := \{\min(\bar{\sigma}(\Delta(j\omega)) : \Delta \in \mathbf{\Delta}, \det(I - \Delta(j\omega)M(j\omega)) = 0)\}^{-1}$ *unless no* $\Delta \in \mathbf{\Delta}$ *makes* $I - \Delta(j\omega)M(j\omega)$ *singular in which case* $\mu_{\mathbf{\Delta}}(M(j\omega)) := 0$.

Now if we consider a matrix M as given in eq.1 as the coefficient matrix of an upper LFT, then this definition implies that $\mu_{\mathbf{\Delta}}(M_{11}(j\omega))$ determines the 'smallest' $\Delta \in \mathbf{\Delta}$ for which the LFT is no longer well defined at ω. If $\mathcal{F}_u(M, 0)$ and all $\Delta \in \mathbf{\Delta}$ are stable transfer function matrices, $\sup_\omega \mu_{\mathbf{\Delta}}(M_{11}(j\omega))$ determines the smallest $\Delta \in \mathbf{\Delta}$ for which $\mathcal{F}_u(M, \Delta)$ is unstable.

We thus have the possibility to test the properties of a *set* of systems by constructing an appropriate LFT with Δ representing some bounded perturbations. For an overview of such tests in the general case of eq.3 we refer to Doyle et al. [4]. Furthermore, we will not go into detail on computational issues with respect to μ but simply refer to recent developments as reported in [14] and available in [15].

We will concentrate on a restricted set of Δs that directly results from real valued parameter variations in state-space models as considered in section 3:

$$\mathbf{\Delta}_{rr} := \{\mathrm{diag}(\delta_1 I_{k_1}, \ldots, \delta_r I_{k_r}) : \delta_i \in \mathrm{R}\} \tag{4}$$

This implies that Δ is *square* and *diagonal* and consist only of *real-valued repeated scalar blocks.*

3 Parametric uncertainty modelling

In this paragraph we will consider the problem of state-space models with parametric uncertainty occurring as real rational multi-dimensional polynomials. This generalizes earlier results in parametric uncertainty modelling as given by [8, 12, 13]. An example can be found in [7].

Consider a vector $p = (p_1, ..., p_r) \in \mathrm{R}^r$ containing r bounded scalar parameters. Let the model of the perturbed system be given as a state-space realization in which the entries of the matrices depend on the parameter vector p:

$$\begin{aligned} \dot{x} &= A(p)x + B(p)u \\ y &= C(p)x + D(p)u, \end{aligned} \tag{5}$$

Now we would like to construct an LFT as follows:

$$\mathcal{F}_u(M,\Delta) = \begin{pmatrix} A(p) & B(p) \\ C(p) & D(p) \end{pmatrix} = M_{22} + M_{21}(I - \Delta M_{11})^{-1}\Delta M_{12} \tag{6}$$

with $\Delta \in \mathbf{\Delta}_{rr}$ (eq.4) and the matrices $M_{22}, M_{21}, M_{11}, M_{12}$ independent of Δ. Note that the state-space description is a function of $(p_1, ..., p_r)$, usually denoting physical parameters, while Δ is a function of $(\delta_1, ..., \delta_r)$, with δ_i a normalized version of p_i.
If we consider only the non-trivial case that $\delta_i \neq 0, \;\; i = 1 \dots r$ we can then define $\rho_i := 1/\delta_i$ and rewrite eq.6 as:

$$\mathcal{F}_u(M,\Delta) = M_{22} + M_{21}\left\{\mathrm{diag}(\rho_1 I_{k_1}, \dots, \rho_r I_{k_r}) - M_{11}\right\}^{-1} M_{12} \tag{7}$$

Note that we have transformed the problem of finding an LFT representation of eq.5 to an ND-realization problem [2].
Using a constructive algorithm we are now able to do the following statement.

> *A solution to the problem of transforming a state-space model with parametric uncertainty to an LFT exists if the entries of the state-space matrices are bounded and can be given as real rational polynomials in the parameters.*

Real rational varying entries in a state-space model can be described as LFTs individually. Based on the properties of the interconnection of LFTs, treated in section 2, these individual LFTs can be collected in one LFT afterwards. Minimality of the obtained LFT can not be guaranteed since it is not straightforward to generalize the 1D concepts of controllability and observability to ND-systems [9].
The procedure consists of eight steps:

1. **Scaling the varying parameters**
Lower and upper bound vectors for the parameter vector p can be determined, denoted respectively as $\underline{p}$ and $\bar{p}$: Now define $p_o := (\underline{p} + \bar{p})/2$, $s := (\bar{p} - \underline{p})/2$, $\delta = (\delta_1 \dots \delta_r)$, $\delta_i \in [-1, +1]$, such that $p_i = p_{oi} + s_i\delta_i$ for $i = 1 \cdots r$. Substitution of this result in eq.5 then gives scaled polynomial expressions for all varying numerators and denominators.

2. **Individual varying terms as LFTs**
The varying parts of a numerator or denominator consist of a number of terms that can be written as separate LFTs acting on the δ_i.

3. **Numerators of varying entries**
Using the fact that two parallel LFTs form again an LFT, the addition of all terms in each numerator can again be written as an LFT.

4. **Denominators of varying entries**
To obtain an LFT of the *inverse* of a polynomial, set up an LFT for the denominator as was done for the numerators in the previous step, subtract 1 and put the result in the feedback path. The obvious fact that the entries of the nominal model must be bounded guarantees well definedness of this feedback structure.

5. **Combining numerators and denominators of individual entries**
Cascade connection of the LFTs of each numerator-denominator pair found in the previous steps leads to a single LFT for each entry.

6. **Combining all varying entries**
LFTs for the A, B, C and D matrices can be set up separately and can be rewritten as one single LFT with $\Delta = \mathrm{diag}(\Delta_A, \Delta_B, \Delta_C, \Delta_D)$.

7. **Transformation to the real-repeated blockstructure**
Δ can be rearranged into the real-valued repeated scalar block structure of eq.4 by interchanging rows and columns of the LFT.

8. **Reducing the dimension of Δ**
The resulting LFT can be set in state-space form in which the uncertainty inputs can be appended to u and the uncertainty outputs can be appended to y. Any uncontrollable and/or unobservable parts of this state-space model can be removed using a standard reduction technique, thus reducing the dimensions of the 'block of integrators'. The same procedure can be used to reduce the size of any of the real-repeated blocks in Δ.
Rewrite the LFT by considering x as an uncertainty input and $\dot{x}$ as an uncertainty output; this

implies that the block of integrators is appended to Δ. Next, separate the uncertainty block $\delta_1 I$ from Δ and consider its uncertainty inputs as 'pseudo-states' and its uncertainty outputs as 'pseudo-derivatives'. Removing the parts that are uncontrollable and/or unobservable when considering all other inputs and outputs will then reduce the size of $\delta_1 I$. This procedure can be repeated for all other real-repeated uncertainty blocks.

With these steps we now have an LFT description which is equivalent to the state-space system of eq.5. As mentioned before, these steps have been implemented within the environment of PC Matlab, such that the entire procedure can be performed interactively.

4 Conclusions

The development of methods for analysis and design based on the structured singular value μ causes an increasing demand for the construction of accurate uncertainty models in the form of LFTs. Usually the knowledge concerning uncertainty in mathematical models of physical systems is available in terms of parameter variations. In state-space models this often appears as variations of entries, that can be approximated accurately by means of ratios of multi-dimensional polynomials in independent variables, having a physical interpretation.

In this paper an algorithm is presented, which is used to transform a state-space model with this type of parametric uncertainty to an LFT description with a real-repeated perturbation matrix. Although the dimension of this perturbation matrix may initially be very high, a reduction procedure is proposed that usually decreases it significantly. However, this procedure does not guarantee minimality of the resulting structure. The procedure has been implemented in PC MatLab, such that uncertainty models can be set up in an interactive user-friendly manner.

5 Acknowledgements

The authors wish to thank Paula Rocha from Delft University of Technology fac. of Mathematics for her help in gaining a better understanding of developments in the area of 2D and ND system representations. This research was sponsored by Philips Research Laboratories in Eindhoven, Delft University of Technology and National Aerospace Laboratory (NLR) in Amsterdam.

References

[1] Balas G.J., Packard A., Doyle J.C., "Theory and applications of robust multivariable control", in *H_∞ and μ Short Course, Musyn inc., Delft, june 25-28,* 1990.

[2] Bose N.K., *Applied multidimensional system theory.* Van Nostrand Reinhold Co., N.Y., 1982.

[3] Doyle J.C., "Analysis of feedback systems with structured uncertainties.", *IEE Proc.. Part D, vol.129, no.6,* pp.242-250, 1982.

[4] Doyle J.C., Packard A., Zhou K., "Review of LFTs, LMIs and μ", in *Proc. IEEE Conf. on Decision and Control,* pp.1227-1232, 1991.

[5] Doyle J.C., Lenz K., Packard A., "Design examples using μ-synthesis: space shuttle lateral axis FCS during reentry", in *Proc. IEEE Conf. on Decision and Control,* pp.2218-2223, 1986.

[6] Fan M.K.H., Tits A.L., "Characterization and efficient computation of the structured singular value.", *IEEE Trans. on Automatic Control, vol.AC-31, no.8,* pp.734-743, 1986.

[7] Lambrechts P.F., Terlouw J.C., Bennani S., Steinbuch M., "Parametric uncertainty modeling using LFTs", in *Proc. American Control Conf.,* pp.267-272, 1993.

[8] Morton B.G., McAfoos R.M., "A μ-test for robustness analysis of a real-parameter variation problem.", in *Proc. American Control Conf.,* pp.135-138, 1985.

[9] Roesser R.E., "A discrete state-space model for linear image processing", *IEEE Trans. on Automatic Control vol AC-20, no.1,* pp.1-10, 1975.

[10] Skogestad S., Morari M., Doyle J.C., "Robust control of ill-conditioned plants: high-purity distillation.", *IEEE Trans. on Automatic Control, vol.AC-33, no.12,* pp.1092-1105, 1988.

[11] Stein G., Doyle J.C., "Beyond singular values and loop shapes", *J. of Guidance, vol.14, no.1,* pp.5-16, 1991.

[12] Steinbuch M., Terlouw J.C., Bosgra O.H., Smit S.G., "Uncertainty modelling and structured singular value computation applied to an electromechanical system", *IEE Proc., Part D, vol.139, no.3,* pp.301-307, 1992.

[13] Steinbuch M., Terlouw J.C., Bosgra O.H., "Robustness analysis for real and complex perturbations applied to an electro-mechanical system", in *Proc. American Control Conf.,* pp.556-561, 1991.

[14] Young P.M., Newlin M.P., Doyle J.C., "μ analysis with real parametric uncertainty", in *Proc. IEEE Conf. on Decision and Control,* pp.1251-1256, 1991.

[15] μ-analysis and synthesis toolbox, MUSYN inc. and The MathWorks, Inc. 1991.

H_2 Performance for Unstructured Uncertainties

Maarten Steinbuch
Philips Research Laboratories
Prof. Holstlaan 4, 5656 AA Eindhoven, The Netherlands, email: steinbuc@prl.philips.nl
Okko H. Bosgra
Mechanical Engineering Systems and Control Group
Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands
email: bosgra@tudw03.tudelft.nl

Abstract. This paper considers a mixed H_2/H_∞ optimal control problem. An explicit parametrization is given of H_∞ norm bounded uncertainties, as lossless bounded real functions. An unconstrained optimization problem is formulated for the H_2-performance worst-case H_∞ norm-bounded uncertainty. The theory is applied to a Compact Disc robust control problem.

1 Robust performance H_2/H_∞ optimal control

The design of robust control systems has attracted a lot of attention during the last decade, resulting in the development of H_∞ control theory [3]. Although this approach is very attractive to ensure stability of an uncertain closed-loop system, it is not always useful for performance requirements. In particular, in many applications noise disturbances act on the system, and uncertainties lead to significant variations of the systems dynamics. This has led to the design problem of stating performance and robustness objectives in the H_2 and H_∞ framework simultaneously [2, 4, 6, 7, 8, 9, 10, 11, 12]. In [2, 4, 6, 7, 8, 9, 12] the mixed H_2/H_∞ optimal control problem has been stated in terms of minimizing the H_2 norm of a system with a constraint on the H_∞ norm of a related transfer function. In contrast to these results, in this paper a robust performance mixed H_2/H_∞ optimal control problem is considered including a parametrization of the worst case H_∞ norm bounded uncertainty relating the signals w_2 and z_2 [10], see Fig. 1.a.

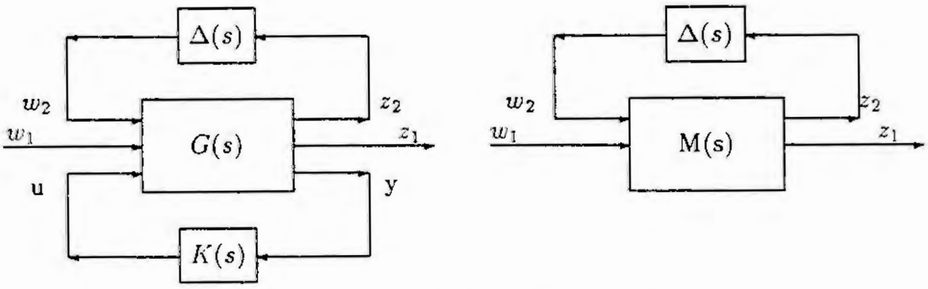


Fig. 1: H_2/H_∞ design problem with uncertainty.
a: Δ, K problem. b: Δ problem.

The robust performance mixed H_2/H_∞ control problem is to minimize the H_2 norm of the transfer function from w_1 to z_1 using the feedback $K(s)$, while maximizing the H_2 norm of the same transfer function over the allowable uncertainties, i.e. $\min_{K(s)} \max_{\|\Delta\|_\infty \leq \gamma} \| T_{w_1 \to z_1}(K, \Delta) \|_2$.
In this paper we will not consider the control design problem, but instead concentrate on the calculation of the worst-case unstructured uncertainty. The theory will be applied to a Compact Disc control problem.

2 Parametrization of H_∞ Norm Bounded Transfer Functions

In [10] we developed a parametrization for strictly proper H_∞ norm-bounded uncertainty models. Based on numerical experiences with this parametrization, the assumption we make in this paper is that the worst-case perturbation will be lossless bounded real. The argument is that in maximizing the H_2 norm all, possibly infinite, dynamics of the perturbation will yield a perturbation on its bound at all frequencies and for all its singular values. To develop a suitable parametrization, let us first consider lossless positive real and lossless bounded real transfer functions [1].

Definition 2.1 *The real rational function* $\Gamma(s)$, $s \in \mathbb{C}$, *is lossless positive real if* $\Gamma(s) + \Gamma^T(-s) = 0$.

Definition 2.2 *The real rational function* $\Delta(s)$, $s \in \mathbb{C}$, *is lossless bounded real if* $\Delta^T(-s)\Delta(s) = I$.

Lemma 2.3 *Let* $\Delta(s) = (I - \Gamma(s))(I + \Gamma(s))^{-1}$, *then* $\Delta(s)$ *is lossless bounded real iff* $\Gamma(s)$ *is lossless positive real.*

Proof: $\Delta^T(-s)\Delta(s) = (I+\Gamma^T(-s))^{-1}(I-\Gamma^T(-s))(I-\Gamma(s))(I+\Gamma(s))^{-1} = (I+\Gamma^T(-s))^{-1}(I+\Gamma^T(-s)\Gamma(s))(I+\Gamma(s))^{-1} = (I + \Gamma^T(-s))^{-1}(I + \Gamma^T(-s))(I + \Gamma(s))(I + \Gamma(s))^{-1} = I$, and conversely. □

Lemma 2.4 *Let* $\Gamma(s) = \bar{H}(sI - \bar{F})^{-1}\bar{G} + \bar{J}$, *with* $\bar{F} + \bar{F}^T = 0$, $\bar{G} = \bar{H}^T$ *and* $\bar{J} + \bar{J}^T = 0$, *with* $\bar{F} \in \mathbb{R}^{n\times n}$ *and* $\bar{J} \in \mathbb{R}^{m\times m}$, *and with* $\bar{H}$ *and* $\bar{G}$ *of compatible dimensions. Then the real matrices* $\bar{F}$, $\bar{H}$ *and* $\bar{J}$ *parametrizes all lossless positive real transfer functions* Γ *with state dimension* n.

Proof: follows by direct application of Definition 2.1. □

Lemma 2.5 *Let* $\Delta(s) = (I - \Gamma(s))(I + \Gamma(s))^{-1}$ *with* $\Gamma(s) = \bar{H}(sI - \bar{F})^{-1}\bar{G} + \bar{J}$, *with* $\bar{F}$, $\bar{H}$ *and* $\bar{J}$ *as defined in the previous lemma. Then a state space realization for* $\Delta(s)$ *is given by:*

$$\left[\begin{array}{c|c} F & G \\ \hline H & J \end{array}\right] = \left[\begin{array}{c|c} \bar{F} - \bar{H}^T(I + \bar{J})^{-1}\bar{H} & -\sqrt{2}\bar{H}^T(I + \bar{J})^{-1} \\ \hline \sqrt{2}(I + \bar{J})^{-1}\bar{H} & (I - \bar{J})(I + \bar{J})^{-1} \end{array}\right] \tag{1}$$

And this is a parametrization for all stable lossless bounded real $\Delta(s)$.

Proof: follows by some matrix manipulations and is omitted here. □

Since matrices $\bar{F}$ and $\bar{J}$ are defined as skew-symmetric (see Lemma 2.4), we further reduce the number of free variables and end this section with the main result.

Theorem 2.6 *Define the matrices* θ *and* ϕ *as upper triangular real matrices, with zero on their diagonal, and with appropriate dimensions, such that* $\bar{F} = \theta - \theta^T$, *and* $\bar{J} = \phi - \phi^T$, *then the triple* $(\theta, \phi, \bar{H})$ *parametrizes all stable lossless bounded real transfer functions* $\Delta(s) = H(sI - F)^{-1}G + J$, *with* H,F,G *and* J *defined by (1).*

3 Worst case perturbations

Consider Fig. 1.b. The noise disturbances w_1 have unit noise intensity, and the uncertainty $\Delta(s)$ represents unstructured H_∞ norm bounded perturbations of the nominal closed-loop system ($M(s)$ includes $G(s)$ and $K(s)$). In contrast to [11], causality of the perturbation is assumed as an implicit and necessary ingredient to pose the true problem. As performance indicator we use the variance of the signal z_1, i.e. the H_2 norm of the closed-loop transfer function from w_1 to z_1. Let $M(s)$ have the state space realization

$$\begin{aligned} \dot{x} &= Ax + B_1 w_1 + B_2 w_2 \\ z_1 &= C_1 x + \qquad D_{12} w_2 \\ z_2 &= C_2 x + D_{21} w_1 \end{aligned} \tag{2}$$

and let $\Delta(s) = H(sI - F)^{-1}G + J$ be defined as in Theorem 2.6. Then the closed-loop system is:

$$\begin{aligned} \begin{pmatrix} \dot{x} \\ \dot{p} \end{pmatrix} &= \begin{pmatrix} A + B_2 J C_2 & B_2 H \\ G C_2 & F \end{pmatrix} \begin{pmatrix} x \\ p \end{pmatrix} + \begin{pmatrix} B_1 + B_2 J D_{21} \\ G D_{21} \end{pmatrix} w_1 \\ z_1 &= \begin{pmatrix} C_1 + D_{12} J C_2 & D_{12} H \end{pmatrix} \begin{pmatrix} x \\ p \end{pmatrix} \end{aligned} \tag{3}$$

denoted in the sequel as $[A], [B], [C]$. Notice that $D_{12} J D_{21} = 0$ for $\| T_{w_1 \to z_1}(\Delta) \|_2 < \infty$.
Let the system (3) be stable and consider the constrained optimization problem

$$\max_{\|\Delta\|_\infty = 1} \| T_{w_1 \to z_1}(\Delta) \|_2 = tr[C]^T[C]S \tag{4}$$

with the state-variance matrix $S = S^T$ the solution to:

$$[A]S + S[A]^T + [B][B]^T = 0 \tag{5}$$

This optimization problem can be reformulated as an **unconstrained** optimization problem:

$$\max_{(\theta,\phi,\bar{H})} tr[C]^T[C]S \tag{6}$$

with S the solution to (5), and with (F, G, H, J) defined by (1), where $\bar{F} = \theta - \theta^T$, $\bar{J} = \phi - \phi^T$.

4 Example: Compact Disc player robust control problem

In Fig.2 a schematic view of a Compact Disc mechanism is shown. The mechanism is composed of a turn-table DC-motor, and a balanced radial arm for track-following. An optical element is mounted at the end of the radial arm. A diode located in this element generates a laser beam that passes through a series of optical lenses to give a spot on the information layer of the disc. An objective lens, suspended by two parallel leaf springs, can be actuated vertically for focussing.

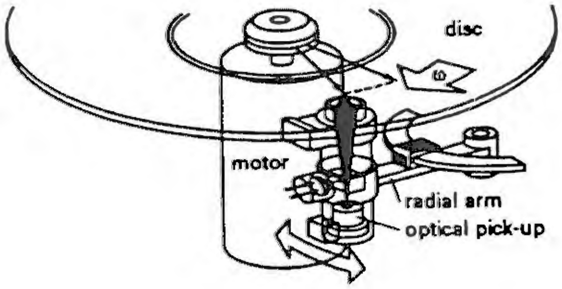


Fig. 2: Schematic view of a rotating arm Compact Disc mechanism.

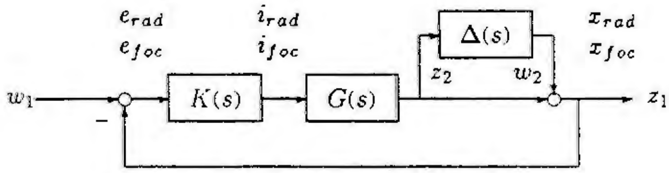


Fig. 3: Configuration of the control loops.

In Fig.3 a block-diagram of the control loop is shown. The difference between the radial (x_{rad}) and vertical (x_{foc}) spot position and the reference track w_1 is detected by an optical pick-up which generates a radial error signal (e_{rad}) and a focus error signal (e_{foc}). A controller $K(s)$ feeds the system with the currents i_{rad} and i_{foc}.

In the numerical experiments a model (G) of order 21 is used, the controller (K) is a 16th order controller designed using μ synthesis [5]. The uncertainty Δ is 20 %. The noise disturbances acting on the multivariable control loop are due to imperfect shape of the tracks and enter the loop at the reference point w_1. The transfer functions between the variables of interest are:

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} (I+GK)^{-1}GK & (I+GK)^{-1} \\ K(I+GK)^{-1} & -K(I+GK)^{-1} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \tag{7}$$

Our interest is in how the uncertainty $\Delta(s)$ connecting w_2 with z_2 can disturb the H_2 performance from w_1 to z_1. By using a general purpose optimization program the results are obtained. The inputs w_1 and w_2 have been scaled such that the nominal performance equals 1 (i.e. $\| (I+GK)^{-1}GK \|_2 = 1$) and such that $\| \Delta(s) \|_\infty = 1$ corresponds to 20 % model uncertainty. For various choices of the dynamic order of the perturbation $\Delta(s)$ results are generated and presented in the table below.

Table 1: Worst case H_2 performance for CD player

Order of Δ	no Δ	0	1	2	9	14
$\max_\Delta \| T_{w_1 \to z_1}(\Delta) \|_2$	1	3.95	5.63	5.79	5.79	5.79

The variance of the performance variables z_1 increases with a factor 5.79 for $\Delta(s)$ with two states or more. In the following figure the perturbed transfer functions are shown.

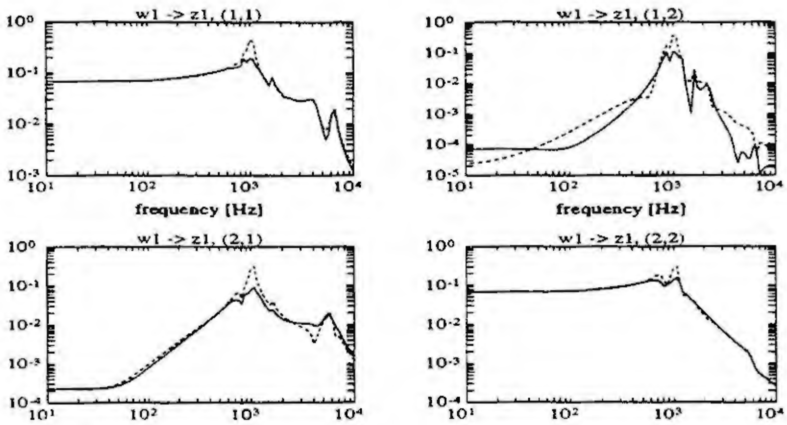


Fig. 4: Nominal (–) and worst case perturbed (- -) transfer function from w_1 to z_1

5 Conclusions

A new formulation of the robust performance H_2/H_∞ optimal control problem has been proposed in this paper, and an explicit parametrization for a worst-case norm-bounded uncertainty has been used, yielding an unconstrained optimization problem. A Compact Disc system with an unstructured uncertainty has been discussed. Using a numerical algorithm we have shown that it is possible to calculate worst case uncertainties. This allows as a next step to redesign and robustify the controller to counteract with the worst case perturbation.

References

[1] Anderson, B.D.O. and S. Vongpanitlerd, '*Network Analysis and Synthesis*', Prentice Hall, Englewood Cliffs, 1973.

[2] Bernstein, D.S. and W.M. Haddad, 'LQG control with an H_∞ performance bound: a Riccati equation approach', *IEEE Trans. on Aut. Control*, AC-34, pp. 293-305, 1989

[3] Doyle, J.C., K. Glover, P. Khargonekar and B.A. Francis, 'State-space solutions to standard H_2 and H_∞ control problems', *IEEE Trans. on Aut. Control*, AC-34, pp. 831-847, 1989.

[4] Doyle, J.C., K. Zhou and B. Bodenheimer, 'Optimal control with mixed H_2 and H_∞ performance objectives', *Proc. 1989 American Control Conference*, 2065-2070.

[5] Groos, P.J.M. van, M. Steinbuch and O.H. Bosgra, "Multivariable control of a compact disc player using μ synthesis", *Proc. 2nd European Control Conference*, Groningen June 26- July 1, 1993, pp. 981-985.

[6] Mustafa, D, 'Relations between maximum entropy/H_∞ control and combined H_∞/LQG control', *Systems and Control Letters*, 12 , pp.193-203, 1989

[7] Rotea, M.A. and P.P. Khargonekar, 'H_2- Optimal control with an H_∞-constraint, the state-feedback case', *IFAC Automatica*, vol. 27 (1991), pp.307-316.

[8] Scherer, C.W., 'Multiobjective H_2/H_∞ control', preprint 1993.

[9] Steinbuch, M. and O.H. Bosgra 'Necessary Conditions for Static and Fixed Order Dynamic Mixed H_2/H_∞ Optimal Control', *Proc. 1991 American Control Conference*, pp.1137-1143.

[10] Steinbuch, M. and O.H. Bosgra 'Robust performance in H_2/H_∞ optimal control', *Proc. IEEE CDC 1991*, pp. 539-550.

[11] Stoorvogel, A.A., 'The robust H_2 control problem: a worst case design', *IEEE Trans. on Aut. Control*, AC-38, pp. 1358-1370, 1993

[12] Yeh, H.H., S.S. Banda and B.C. Chang, ' Necessary and sufficient conditions for mixed H_2 and H_∞ optimal control', *IEEE Trans. on Aut. Control*, AC-37, pp. 355-358, 1992.

SOLVING TOTAL AND MODEL LEAST SQUARES ITERATIVELY AND RECURSIVELY FOR ROBUST IDENTIFICATION

Alexander WEINMANN,
Technical University Vienna, Austria,
A-1040 Gusshausstrasse 27

Abstract
An iterative and a recursive algorithm solving the Total Least Squares Problem is presented. In view of process simulation and identification problems for continuous-time and discrete-time systems, a new method called Model Least Square is developed: The assumption is stated that the entire set of coefficients associated with the derivatives in the differential equation is deteriorated whereas the instantaneous values of the input and output variables are unperturbed. The methods of Total Least Squares, Model Least Squares and ordinary Least Squares are portrayed by unified algorithms.

1 Introduction

For system modelling processes, the Total Least Squares Method is an interesting alternative versus Ordinary Least Squares [1-4]. In various publications it is argued that it is not worth wile to apply the Total Least Square method, regarding the fact that the results in the optimal parameter $\mathbf{p}$ calculated by the Total Least Squares and the ordinary Least Squares method agree up to second-order terms and that even column scaling of the measurement matrix $\mathbf{M}$ does not influence the error $(\mathbf{E}, \boldsymbol{\varepsilon})$ up to terms of second order.

In spite of this, a new method "Model Least Squares Approach" combined with Total Least Squares guarantees short computation time, good convergence facilities and a great region of attraction and eleviates the application of the Total Least Squares. Especially for highly noisy data, improving the whole model $\mathbf{M}$ by an error matrix $\mathbf{E}$ facilitates the simulation.

Moreover, the method of Model Least Square is defined as a specialized case of Total Least Square and investigated in detail.

In this paper Total Least Square (TLS), Model Least Square (MLS) and ordinary Least Square (LS) are considered in detail. In order to avoid tedious symbols, no indices are used in connection with $\mathbf{E}$, $\boldsymbol{\varepsilon}$, $\mathbf{p}$ although there are different results in $\mathbf{E}$, $\boldsymbol{\varepsilon}$, $\mathbf{p}$ for TLS, MLS and LS; only by the headlines adequate distinctions are given.

2 Total Least Square. Direct Iterative Solution

Consider a list of measurements $\mathbf{y}$ to be approximated by the model $\mathbf{Mp}$ such that the error $\boldsymbol{\varepsilon}$ is minimized via Frobenius norm. If the process identification of linear difference equations is considered, $\mathbf{y}$ and $\mathbf{M}$ contain input and output variables available from time instants. In order to obtain coefficient unity associated with $\mathbf{y}$, a "process of normalization" is applied to the remaining coefficients, i.e., the parameter vector $\mathbf{p}$. Moreover, the model matrix $\mathbf{M}$ should be improved by an additional matrix $\mathbf{E}$. Minimizing $||\mathbf{E} \vdots \boldsymbol{\varepsilon}||_F$ leads to the Total Least Squares approach [5]. The original measurement vector $\mathbf{y}$ is partitioned into the combined model $(\mathbf{M}+\mathbf{E})\mathbf{p}$ and into the error $\boldsymbol{\varepsilon}$

$$(\mathbf{M}+\mathbf{E})\mathbf{p}+\varepsilon=\mathbf{y}\,. \tag{1}$$

Using the weighting matrices $\mathbf{V}$ and $\mathbf{W}$, the resulting matrix to be minimized by its Frobenius norm is $(\mathbf{VE} \vdots \mathbf{W}\varepsilon)$. Hence,

$$\|(\mathbf{VE} \vdots \mathbf{W}\varepsilon)\|_F=\mathrm{tr}\Big\{(\mathbf{VE} \vdots \mathbf{W}\varepsilon)^T(\mathbf{VE} \vdots \mathbf{W}\varepsilon)\Big\} \ \rightarrow \min_{\mathbf{p},\mathbf{E}}\ . \tag{2}$$

The dimensions are

$$\mathbf{y},\ \varepsilon,\ \boldsymbol{\lambda}\in\mathcal{R}^{n_m}; \qquad \mathbf{p}\in\mathcal{R}^{n_p}; \qquad \mathbf{E},\ \mathbf{M}\in\mathcal{R}^{n_m\times n_p}; \qquad \mathbf{V},\ \mathbf{W}\in\mathcal{R}^{n_m\times n_m}\ . \tag{3}$$

Evaluating yields

$$\mathrm{tr}\Big\{[\mathbf{VE} \vdots \mathbf{W}(\mathbf{y}-\mathbf{Ep}-\mathbf{Mp})]^T[\mathbf{VE} \vdots \mathbf{W}(\mathbf{y}-\mathbf{Ep}-\mathbf{Mp})]\Big\}= \tag{4}$$

$$=\mathrm{tr}\Big\{\begin{pmatrix}\mathbf{E}^T\mathbf{V}^T\\ \dots\ \dots\ \dots\\ (\mathbf{y}^T-\mathbf{p}^T\mathbf{E}^T-\mathbf{p}^T\mathbf{M}^T)\mathbf{W}^T\end{pmatrix}\Big(\mathbf{VE} \vdots \mathbf{W}(\mathbf{y}-\mathbf{Ep}-\mathbf{Mp})\Big)\Big\} \ \rightarrow \min_{\mathbf{p},\mathbf{E}} \tag{5}$$

$$\mathrm{tr}\Big\{\begin{pmatrix}\mathbf{E}^T\mathbf{V}^T\mathbf{VE} & \vdots & \mathbf{E}^T\mathbf{V}^T\mathbf{W}(\mathbf{y}-\mathbf{Ep}-\mathbf{Mp})\\ \dots\ \dots\ \dots\ \dots & & \dots\ \dots\ \dots\ \dots\ \dots\ \dots\\ (\mathbf{y}^T-\mathbf{p}^T\mathbf{E}^T-\mathbf{p}^T\mathbf{M}^T)\mathbf{W}^T\mathbf{VE} & \vdots & (\mathbf{y}^T-\mathbf{p}^T\mathbf{E}^T-\mathbf{p}^T\mathbf{M}^T)\mathbf{W}^T\mathbf{W}(\mathbf{y}-\mathbf{Ep}-\mathbf{Mp})\end{pmatrix}\Big\} \ \rightarrow \min_{\mathbf{p},\mathbf{E}} \tag{6}$$

$$\mathrm{tr}\Big\{\mathbf{E}^T\mathbf{V}^T\mathbf{VE}+(\mathbf{y}^T-\mathbf{p}^T\mathbf{E}^T-\mathbf{p}^T\mathbf{M}^T)\mathbf{W}^T\mathbf{W}(\mathbf{y}-\mathbf{Ep}-\mathbf{Mp})\Big\} \ \rightarrow \min_{\mathbf{p},\mathbf{E}} \tag{7}$$

$$\begin{aligned}\mathrm{tr}\Big\{\ \mathbf{E}^T\mathbf{V}^T\mathbf{VE} \ &+\ \mathbf{y}^T\mathbf{W}^T\mathbf{Wy}-\mathbf{p}^T\mathbf{E}^T\mathbf{W}^T\mathbf{Wy}-\mathbf{p}^T\mathbf{M}^T\mathbf{W}^T\mathbf{Wy}-\mathbf{y}^T\mathbf{W}^T\mathbf{WEp}+\\ &+\ \mathbf{p}^T\mathbf{E}^T\mathbf{W}^T\mathbf{WEp}+\mathbf{p}^T\mathbf{M}^T\mathbf{W}^T\mathbf{WEp}-\mathbf{y}^T\mathbf{W}^T\mathbf{WMp}+\\ &+\ \mathbf{p}^T\mathbf{E}^T\mathbf{W}^T\mathbf{WMp}+\mathbf{p}^T\mathbf{M}^T\mathbf{W}^T\mathbf{WMp}\ \Big\}\rightarrow \min_{\mathbf{p},\mathbf{E}}\ .\end{aligned} \tag{8}$$

Deriving with respect to the matrix $\mathbf{E}$ by applying matrix derivative calculus,

$$\frac{\partial}{\partial\mathbf{E}}\mathrm{tr}\mathbf{E}^T\mathbf{AEB}=\mathbf{A}^T\mathbf{EB}^T+\mathbf{AEB},\quad \frac{\partial}{\partial\mathbf{E}}\mathrm{tr}\mathbf{E}^T\mathbf{AE}=(\mathbf{A}^T+\mathbf{A})\mathbf{E},\quad \frac{\partial}{\partial\mathbf{E}}\mathrm{tr}\mathbf{EB}=\mathbf{B}^T,\quad \frac{\partial}{\partial\mathbf{E}}\mathrm{tr}\mathbf{E}^T\mathbf{B}=\mathbf{B}\ , \tag{9}$$

yields

$$\mathbf{V}^T\mathbf{VE}-\mathbf{W}^T\mathbf{Wyp}^T+\mathbf{W}^T\mathbf{WEpp}^T+\mathbf{W}^T\mathbf{WMpp}^T=0 \tag{10}$$

$$(\mathbf{W}^T\mathbf{W})^{-1}\mathbf{V}^T\mathbf{VE}-\mathbf{yp}^T+\mathbf{Epp}^T+\mathbf{Mpp}^T=0\ . \tag{11}$$

The solution for $\mathbf{V}=\mathbf{W}$ is given by

$$\hat{\mathbf{E}}=(\mathbf{y}-\mathbf{M}\hat{\mathbf{p}})\hat{\mathbf{p}}^T(\mathbf{I}+\hat{\mathbf{p}}\hat{\mathbf{p}}^T)^{-1} \quad \text{for any } \mathbf{W}\ . \tag{12}$$

For general $\mathbf{V}$, using the Kronecker product and the relations $\mathrm{col}\,\mathbf{ABC}=(\mathbf{C}^T\otimes\mathbf{A})\mathrm{col}\,\mathbf{B}$ and $\mathrm{col}\,\mathbf{yp}^T=\mathbf{p}\otimes\mathbf{y}$,

$$\mathrm{col}\,\hat{\mathbf{E}}=\{\mathbf{I}\otimes[(\mathbf{W}^T\mathbf{W})^{-1}\mathbf{V}^T\mathbf{V}]+(\hat{\mathbf{p}}\hat{\mathbf{p}}^T)\otimes\mathbf{I}\}^{-1}[\hat{\mathbf{p}}\otimes(\mathbf{y}-\mathbf{M}\hat{\mathbf{p}})] \tag{13}$$

is obtained.

If $\mathbf{VE}$ in Eq.(2) is replaced by $\mathbf{VEU}$, $\mathbf{U}\in\mathcal{R}^{n_m\times n_m}$, then in Eq.(13) the former matrix $\mathbf{I}$ has to be substituted by $\mathbf{UU}^T$.

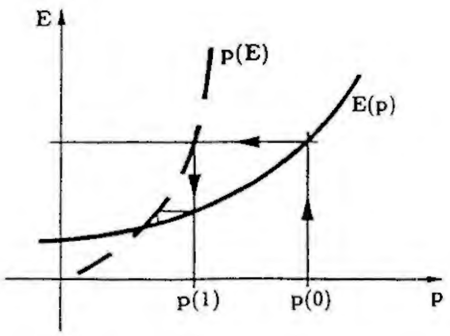


Figure 1: Brief sketch concerning the algorithm Ω

Deriving with respect to p yields, calculations omitted,

$$\hat{\mathbf{p}} = [(\mathbf{M}+\hat{\mathbf{E}})^T\mathbf{W}^T\mathbf{W}(\mathbf{M}+\hat{\mathbf{E}})]^{-1}(\mathbf{M}+\hat{\mathbf{E}})^T\mathbf{W}^T\mathbf{W}\mathbf{y} \ . \tag{14}$$

This results is independent of **V**.

Omitting Eq.(12) or (13) and setting $\mathbf{E} = 0$, then Eq.(14) represents the well-known ordinary least-squares estimation method

$$\hat{\mathbf{p}} = [\mathbf{M}^T\mathbf{W}^T\mathbf{W}\mathbf{M}]^{-1}\mathbf{M}^T\mathbf{W}^T\mathbf{W}\mathbf{y} \ . \tag{15}$$

3 Ω-Algorithm

Consider **E** and **p** as obtained in Eq.(12) and (14). Starting with an initial p(0), from Eq.(14) a preliminary result **E** is achieved, then inserting into Eq.(12) or (13) produces p(1) an so on, see Fig. 1. This algorithm is referred to as Ω. The result can be written as

$$(\hat{\mathbf{p}}, \hat{\mathbf{E}}) = \Omega_{TLS}[\mathbf{p}, \mathbf{E}; \mathbf{V}; \mathbf{p}(0)] \ . \tag{16}$$

4 Iterative Model Least Square

If the measurement vector **y** is considered free of noise, one has $\epsilon \equiv 0$. Then the model error **E** has to compensate for all the inaccuracies in the system equation

$$(\mathbf{M}+\mathbf{E})\mathbf{p} = \mathbf{y} \ . \tag{17}$$

Taking the weighting matrix **W** into account as before, the scalar function to be minimized is

$$\mathrm{tr}\,\{\mathbf{E}^T\mathbf{V}^T\mathbf{V}\mathbf{E}\} + \lambda^T[(\mathbf{M}+\mathbf{E})\mathbf{p}-\mathbf{y}] \rightarrow \min_{\mathbf{p},\mathbf{E}} \tag{18}$$

where λ is a vector Lagrange multiplier. Equating the derivative with respect to **E** to zero and combining with Eq.(17) yields

$$\frac{\partial}{\partial \mathbf{E}}\mathrm{tr}\{\mathbf{E}^T\mathbf{V}^T\mathbf{V}\mathbf{E}\} + \lambda^T[(\mathbf{M}+\mathbf{E})\mathbf{p}-\mathbf{y}] = 0 \ . \tag{19}$$

By Eq.(20)[1],

$$2\,\mathbf{V}^T\mathbf{V}\mathbf{E} + \lambda\mathbf{p}^T = 0 \quad | \times \mathbf{p} \tag{21}$$

[1] For **A** and **a** given, the problem of evaluating **b** from the overdetermined equation $\mathbf{A} \doteq \mathbf{a}\mathbf{b}^T$ in order to minimize $\|\mathbf{X}\|_F$ yields

$$\mathbf{A}+\mathbf{X} = \mathbf{a}\mathbf{b}^T; \quad \|\mathbf{X}\|_F \rightarrow \min_{\mathbf{b}} \quad \leadsto \quad \mathbf{b} = \mathbf{A}^T\mathbf{a}/(\mathbf{a}^T\mathbf{a}) \ . \tag{20}$$

	$\hat{\mathbf{p}}$ and $\hat{\mathbf{E}}$	Residual
LS	$\hat{\mathbf{p}} = [\mathbf{M}^T\mathbf{W}^T\mathbf{W}\mathbf{M}]^{-1}\mathbf{M}^T\mathbf{W}^T\mathbf{W}\mathbf{y}$ $\hat{\mathbf{E}} \equiv \mathbf{0}$	$\|\mathbf{W}(\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})\|_F$
TLS $\mathbf{V} = \mathbf{W}$	$\hat{\mathbf{p}} = [(\mathbf{M} + \hat{\mathbf{E}})^T\mathbf{W}^T\mathbf{W}(\mathbf{M} + \hat{\mathbf{E}})]^{-1}(\mathbf{M} + \hat{\mathbf{E}})^T\mathbf{W}^T\mathbf{W}\mathbf{y}$ $\hat{\mathbf{E}} = (\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})\hat{\mathbf{p}}^T(\mathbf{I} + \hat{\mathbf{p}}\hat{\mathbf{p}}^T)^{-1}$	$\|\mathbf{W}\hat{\mathbf{E}} \vdots \mathbf{W}[\mathbf{y} - (\mathbf{M} + \hat{\mathbf{E}})\hat{\mathbf{p}}]\|_F$
TLS $\mathbf{V} \neq \mathbf{W}$	$\hat{\mathbf{p}} = [(\mathbf{M} + \hat{\mathbf{E}})^T\mathbf{W}^T\mathbf{W}(\mathbf{M} + \hat{\mathbf{E}})]^{-1}(\mathbf{M} + \hat{\mathbf{E}})^T\mathbf{W}^T\mathbf{W}\mathbf{y}$ $\text{col}\ \hat{\mathbf{E}} = \{\mathbf{I} \otimes [(\mathbf{W}^T\mathbf{W})^{-1}\mathbf{V}^T\mathbf{V}] + (\hat{\mathbf{p}}\hat{\mathbf{p}}^T) \otimes \mathbf{I}\}^{-1}[\hat{\mathbf{p}} \otimes (\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})]$	$\|\mathbf{V}\hat{\mathbf{E}} \vdots \mathbf{W}[\mathbf{y} - (\mathbf{M} + \hat{\mathbf{E}})\hat{\mathbf{p}}]\|_F$
MLS	$\hat{\mathbf{p}} = [(\mathbf{M} + \hat{\mathbf{E}})^T\mathbf{W}^T\mathbf{W}\mathbf{M}]^{-1}(\mathbf{M} + \hat{\mathbf{E}})^T\mathbf{W}^T\mathbf{W}\mathbf{y}$ $\hat{\mathbf{E}} = (\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})\hat{\mathbf{p}}^T(\hat{\mathbf{p}}^T\hat{\mathbf{p}})^{-1}$	$\|\mathbf{V}\hat{\mathbf{E}}\|_F$

Table 1: Comparing LS, TLS and MLS

$$2\,\mathbf{V}^T\mathbf{V}\mathbf{E}\mathbf{p} + \boldsymbol{\lambda}\mathbf{p}^T\mathbf{p} = 0 \tag{22}$$

$$\boldsymbol{\lambda} = -\frac{2\,\mathbf{V}^T\mathbf{V}\mathbf{E}\mathbf{p}}{\mathbf{p}^T\mathbf{p}} = -\frac{2\,\mathbf{V}^T\mathbf{V}(\mathbf{y} - \mathbf{M}\mathbf{p})}{\mathbf{p}^T\mathbf{p}} \tag{23}$$

$$\hat{\mathbf{E}} = -0.5\,(\mathbf{V}^T\mathbf{V})^{-1}\boldsymbol{\lambda}\mathbf{p}^T = \frac{\mathbf{y}\mathbf{p}^T - \mathbf{M}\mathbf{p}\mathbf{p}^T}{\mathbf{p}^T\mathbf{p}}\ . \tag{24}$$

Hence, for any $\mathbf{p}$ a solution exists. This result seems to be derived in a trivial way since the Eq.(24) immediately could result from Eq.(17). However, the additional calculation of the optimum versus $\mathbf{p}$ is not trivial and $\boldsymbol{\lambda}$ is needed for this optimum as well. Eq. (24) coincides with the result of Eq.(10) or (13) specialized for $\mathbf{W} \to \infty$.

If, additionally, the minimum solution should be a minimum versus $\mathbf{p}$ as well, the derivative with respect to $\mathbf{p}$ must vanish

$$\frac{\partial\boldsymbol{\lambda}^T(\mathbf{M} + \mathbf{E})\mathbf{p}}{\partial\mathbf{p}} = 0 \quad \leadsto \quad (\mathbf{M} + \mathbf{E})^T\boldsymbol{\lambda} = 0\ . \tag{25}$$

Using Eq.(23), the result is

$$(\mathbf{M} + \mathbf{E})^T(\frac{-2\,\mathbf{V}^T\mathbf{V}(\mathbf{y} - \mathbf{M}\mathbf{p})}{\mathbf{p}^T\mathbf{p}}) = 0\ . \tag{26}$$

From Eq.(26), omitting the scalar $-0.5\,\mathbf{p}^T\mathbf{p}$, yields

$$(\mathbf{M}^T + \mathbf{E}^T)\mathbf{V}^T\mathbf{V}\mathbf{y} - (\mathbf{M}^T + \mathbf{E}^T)\mathbf{V}^T\mathbf{V}\mathbf{M}\mathbf{p} = 0 \tag{27}$$

and

$$\hat{\mathbf{p}} \;=\; [(\mathbf{M} + \hat{\mathbf{E}})^T\mathbf{V}^T\mathbf{V}\mathbf{M}]^{-1}(\mathbf{M} + \hat{\mathbf{E}})^T\mathbf{V}^T\mathbf{V}\mathbf{y} \tag{28}$$

$$\hat{\mathbf{E}} \;=\; \frac{1}{\hat{\mathbf{p}}^T\hat{\mathbf{p}}}\,(\mathbf{y}\hat{\mathbf{p}}^T - \mathbf{M}\hat{\mathbf{p}}\hat{\mathbf{p}}^T) = (\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})\hat{\mathbf{p}}^T(\hat{\mathbf{p}}^T\hat{\mathbf{p}})^{-1}\ . \tag{29}$$

An iteration for solving Eqs.(28) and (29) is defined as

$$(\hat{\mathbf{p}}, \hat{\mathbf{E}}) = \Omega_{MLS}[\mathbf{p}, \mathbf{E}; \mathbf{V}; \mathbf{p}(0)]\ . \tag{30}$$

5 Comparing Ordinary Least Square, Total Least Square and Model Least Square

In Table 1, a comparison between the results of ordinary Least Square, Total Least Square and Model Least Square is given. The residual is the minimum performance.

With respect to the rank lowering fact [6], Total Least Squares methods have poor convergence facilities. In spite of TLS, MLS shows excellent convergence and yields a result very close to the TLS result. If the MLS result is used as the initial value for TLS, a well-balanced over-all performance is obtained.

6 Residual

6.1 Total Least Square

Since

$$\mathbf{E} = (\mathbf{y} - \mathbf{M}\mathbf{p})\mathbf{p}^T(1 + \mathbf{p}^T\mathbf{p})^{-1} = \boldsymbol{\varepsilon}\mathbf{p}^T \qquad \text{and} \qquad \boldsymbol{\varepsilon} = (\mathbf{y} - \mathbf{M}\mathbf{p})(1 + \mathbf{p}^T\mathbf{p})^{-1} \ , \tag{31}$$

$$\begin{aligned} \|\hat{\mathbf{E}} \vdots \hat{\boldsymbol{\varepsilon}}\|_F &= \|\hat{\boldsymbol{\varepsilon}}\hat{\mathbf{p}}^T \vdots \hat{\boldsymbol{\varepsilon}}\|_F = \|\hat{\boldsymbol{\varepsilon}}\begin{pmatrix}\hat{\mathbf{p}}\\1\end{pmatrix}^T\|_F = \mathrm{tr}\{\begin{pmatrix}\hat{\mathbf{p}}\\1\end{pmatrix}\hat{\boldsymbol{\varepsilon}}^T\hat{\boldsymbol{\varepsilon}}\begin{pmatrix}\hat{\mathbf{p}}\\1\end{pmatrix}^T\} = \hat{\boldsymbol{\varepsilon}}^T\hat{\boldsymbol{\varepsilon}}(1 + \hat{\mathbf{p}}^T\hat{\mathbf{p}}) & (32)\\ &= (\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})^T(\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})(1 + \hat{\mathbf{p}}^T\hat{\mathbf{p}})^{-1} \ . & (33) \end{aligned}$$

6.2 Model Least Square

From Eq.(29),

$$\|\hat{\mathbf{E}}\|_F = \mathrm{tr}\{\mathbf{p}(\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})^T(\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})\hat{\mathbf{p}}^T\}(\hat{\mathbf{p}}^T\hat{\mathbf{p}})^{-2} = (\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})^T(\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})(\hat{\mathbf{p}}^T\hat{\mathbf{p}})^{-1} \ . \tag{34}$$

6.3 Ordinary Least Square

For the purpose of comparison, the well-known ordinary least squares result is repeated

$$\|\hat{\boldsymbol{\varepsilon}}\|_F = \|\mathbf{y} - \mathbf{M}\hat{\mathbf{p}}\|_F = (\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})^T(\mathbf{y} - \mathbf{M}\hat{\mathbf{p}}) \ . \tag{35}$$

7 Recursive Algorithm for V = I and W = I

Consider the case that $\mathbf{M}$ and $\mathbf{y}$ are augmented by an additional row $\mathbf{m}^T$ and scalar y_Δ. Then, the error matrix $\mathbf{E}$ increases by the row $\mathbf{e}^T$, and $\mathbf{p}$ changes to $\mathbf{p} + \Delta\mathbf{p}$. Caused by the additional information $\mathbf{m}^T$ and y_Δ, the error matrix $\mathbf{E}$ may change to $\mathbf{E} + \Delta\mathbf{E}$.

Comparison of the algorithms for iterative TLS and MLS shows close similarities. Define $\mathbf{L}, \mathbf{R}, \mathbf{l}$ and $\mathbf{r}$ as given in the table

V = I, W = I	L	R	l	r
TLS	M + E	M + E	m + e	m + e
MLS	M + E	M	m + e	m

where $\mathbf{l}^T$ and $\mathbf{r}^T$ are referred to as the last row of $\mathbf{L}$ and $\mathbf{R}$ when proceeding from the k-th to the $(k+1)$-th step of recursion, respectively. Now, the derivation of the well-known recursive algorithm for ordinary least-squares estimation can be utilized [7,8]. Hence,

$$\begin{aligned} \hat{\mathbf{p}}_k &= (\mathbf{L}_k^T\mathbf{R}_k)^{-1}\mathbf{L}_k^T\mathbf{y}_k & (36)\\ \hat{\mathbf{p}}_{k+1} &= (\mathbf{L}_{k+1}^T\mathbf{R}_{k+1})^{-1}\mathbf{L}_{k+1}^T\mathbf{y}_{k+1} \ . & (37) \end{aligned}$$

Defining

$$\mathbf{y}_{k+1} \triangleq \begin{pmatrix}\mathbf{y}_k\\ y_{\Delta,k+1}\end{pmatrix}, \qquad \mathbf{L}_{k+1} \triangleq \begin{pmatrix}\mathbf{L}_k\\ \mathbf{l}_{k+1}^T\end{pmatrix}, \qquad \mathbf{R}_{k+1} \triangleq \begin{pmatrix}\mathbf{R}_k\\ \mathbf{r}_{k+1}^T\end{pmatrix} \ , \tag{38}$$

it results

$$\mathbf{L}_{k+1}^T\mathbf{y}_{k+1} = \mathbf{L}_k^T\mathbf{y}_k + \mathbf{l}_{k+1}y_{\Delta,k+1} \tag{39}$$

$$\mathbf{L}_{k+1}^T\mathbf{R}_{k+1} = \mathbf{L}_k^T\mathbf{R}_k + \mathbf{l}_{k+1}\mathbf{r}_{k+1}^T \ . \tag{40}$$

Adapting the algorithm yields

$$\begin{aligned} \boldsymbol{\gamma}_k &= \frac{\mathbf{P}_k\mathbf{l}_{k+1}}{1 + \mathbf{r}_{k+1}^T\mathbf{P}_k\mathbf{l}_{k+1}} & (41)\\ \mathbf{P}_{k+1} &= \mathbf{P}_k - \boldsymbol{\gamma}_k\mathbf{r}_{k+1}^T\mathbf{P}_k & (42)\\ \hat{\mathbf{p}}_{k+1} &= \hat{\mathbf{p}}_k + \boldsymbol{\gamma}_k[y_{\Delta,k+1} - \mathbf{r}_{k+1}^T\hat{\mathbf{p}}_k] & (43) \end{aligned}$$

7.1 Total Least Square

From Eq.(12),

$$\mathbf{E}_{k+1} = \begin{pmatrix} \mathbf{E}_k \\ \mathbf{e}_{k+1}^T \end{pmatrix} = [\begin{pmatrix} \mathbf{y}_k \\ y_{\Delta,k+1} \end{pmatrix} \mathbf{p}_{k+1}^T - \begin{pmatrix} \mathbf{M} \\ \mathbf{m}_{k+1}^T \end{pmatrix} \mathbf{p}_{k+1}\mathbf{p}_{k+1}^T](\mathbf{I} + \mathbf{p}_{k+1}\mathbf{p}_{k+1}^T)^{-1} \ . \tag{44}$$

From the last row of the equation above, by transposition one finds Eq.(46). At the k-th step the following calculations have to be executed until an adequate stopping condition is satisfied

$$\begin{array}{lll} & \mathbf{p}_{k+1} & := \mathbf{p}_k \\ \P & \mathbf{e}_{k+1} & = (\mathbf{I} + \mathbf{p}_{k+1}\mathbf{p}_{k+1}^T)^{-1}(y_{\Delta,k+1}\mathbf{p}_{k+1} - \mathbf{p}_{k+1}\mathbf{p}_{k+1}^T\mathbf{m}_{k+1}) \\ & \mathbf{l}_{k+1} & = \mathbf{m}_{k+1} + \mathbf{e}_{k+1} \quad \text{and} \quad \mathbf{r}_{k+1} = \mathbf{m}_{k+1} + \mathbf{e}_{k+1} \\ & \text{Eqs.} & \text{(41) through (43), go to } \P \ . \end{array} \begin{array}{r} (45) \\ (46) \\ (47) \\ (48) \end{array}$$

The result of the iterative procedure per step k yields $\hat{\mathbf{e}}_{k+1}$ and $\hat{\mathbf{p}}_{k+1}$ based on the last results $\mathbf{m}_{k+1}$ and $y_{\Delta,k+1}$.

7.2 Model Least Square

From Eq.(29),

$$\mathbf{E}_{k+1}\mathbf{p}_{k+1}^T\mathbf{p}_{k+1} = \begin{pmatrix} \mathbf{E}_k \\ \mathbf{e}_{k+1}^T \end{pmatrix} \mathbf{p}_{k+1}^T\mathbf{p}_{k+1} = \begin{pmatrix} \mathbf{y}_k \\ y_{\Delta,k+1} \end{pmatrix} \mathbf{p}_{k+1}^T - \begin{pmatrix} \mathbf{M}_k \\ \mathbf{m}_{k+1}^T \end{pmatrix} \mathbf{p}_{k+1}\mathbf{p}_{k+1}^T \ ; \tag{49}$$

the last row above is given by the transposed Eq.(51). At the k-th step the following procedure has to be calculated

$$\begin{array}{lll} & \mathbf{p}_{k+1} & := \mathbf{p}_k \\ \P & \mathbf{e}_{k+1} & = \dfrac{y_{\Delta,k+1}\mathbf{p}_{k+1} - \mathbf{p}_{k+1}\mathbf{p}_{k+1}^T\mathbf{m}_{k+1}}{\mathbf{p}_{k+1}^T\mathbf{p}_{k+1}} \\ & \mathbf{l}_{k+1} & = \mathbf{m}_{k+1} + \mathbf{e}_{k+1} \quad \text{and} \quad \mathbf{r}_{k+1} = \mathbf{m}_{k+1} \\ & \text{Eqs.} & \text{(41) through (43), go to } \P \ . \end{array} \begin{array}{r} (50) \\ (51) \\ (52) \\ (53) \end{array}$$

The result of the iteration is denoted $\hat{\mathbf{e}}_{k+1}$ and $\hat{\mathbf{p}}_{k+1}$.

8 Recursive Total Least Square Algorithm for Diagonal V and W

For general but diagonal matrices **V** and **W**, the results of the preceding section can be modified and computed recursively as well. Adapting the table yields

$\mathbf{V} \neq \mathbf{I},\ \mathbf{W} \neq \mathbf{I}$	L	R	l	r
TLS	$\mathbf{W}(\mathbf{M}+\mathbf{E})$	$\mathbf{W}(\mathbf{M}+\mathbf{E})$	$\mathbf{W}(\mathbf{m}+\mathbf{e})$	$\mathbf{W}(\mathbf{m}+\mathbf{e})$
MLS	$\mathbf{V}(\mathbf{M}+\mathbf{E})$	$\mathbf{VM}$	$\mathbf{V}(\mathbf{m}+\mathbf{e})$	$\mathbf{Vm}$

.

From the preceding section, except the changes in the table above and in the following derivation, the results can be taken over. Starting from Eq.(10) ,

$$\begin{pmatrix} \mathbf{V}^T\mathbf{V} & 0 \\ 0^T & v_{k+1} \end{pmatrix} \begin{pmatrix} \mathbf{E}_k \\ \mathbf{e}_{k+1}^T \end{pmatrix} - \begin{pmatrix} \mathbf{W}^T\mathbf{W} & 0 \\ 0^T & w_{k+1} \end{pmatrix} \left[\begin{pmatrix} \mathbf{y}_k \\ y_{\Delta,k+1} \end{pmatrix} + \begin{pmatrix} \mathbf{E}_k \\ \mathbf{e}_{k+1}^T \end{pmatrix} \mathbf{p}_{k+1} + \begin{pmatrix} \mathbf{M}_k \\ \mathbf{m}_{k+1}^T \end{pmatrix} \mathbf{p}_{k+1} \right] \mathbf{p}_{k+1}^T = 0 \tag{54}$$

$$\mathbf{e}_{k+1} = w_{k+1}(y_{\Delta,k+1} - \mathbf{m}_{k+1}^T\mathbf{p}_{k+1})(v_{k+1}\mathbf{I} + w_{k+1}\mathbf{p}_{k+1}\mathbf{p}_{k+1}^T)^{-1}\mathbf{p}_{k+1} \ . \tag{55}$$

9 Model Least Square Subject to Conditions

Consider the problem of Eq.(2) subject to the condition $\mathbf{Bp} = \mathbf{a}$ where $\mathbf{B} \in \mathcal{R}^{n_B \times n_p}$ and $n_B \le n_p$. Augmenting Eq.(2) by the expression $\boldsymbol{\mu}^T(\mathbf{Bp} - \mathbf{a})$ with the vector Lagrange multiplier $\boldsymbol{\mu}$,

$$\mathrm{tr}\{\mathbf{E}^T\mathbf{V}^T\mathbf{V}\mathbf{E}\} + \boldsymbol{\lambda}^T[(\mathbf{M}+\mathbf{E})\mathbf{p} - \mathbf{y}] + \boldsymbol{\mu}^T[\mathbf{Bp} - \mathbf{a}] \rightarrow \min_{\mathbf{E},\mathbf{p}} \quad . \tag{56}$$

By deriving with respect to $\mathbf{E}$, the Eqs.(23) and (24) are achieved, as before. The result is the same as Eq.(23). Deriving with respect to $\mathbf{p}$ yields

$$[\boldsymbol{\lambda}^T(\mathbf{M}+\mathbf{E})]^T + \mathbf{B}^T\boldsymbol{\mu} = 0 \ . \tag{57}$$

Combining with Eq.(23) and substituting into $\mathbf{a} = \mathbf{Bp}$ yields

$$\boldsymbol{\mu} = \{0.5\,\mathbf{B}[(\mathbf{M}+\mathbf{E})^T\mathbf{V}^T\mathbf{V}\mathbf{M}]^{-1}\mathbf{B}^T\mathbf{p}^T\mathbf{p}\}^{-1} \times \{\mathbf{B}[(\mathbf{M}+\mathbf{E})^T\mathbf{V}^T\mathbf{V}\mathbf{M}]^{-1}(\mathbf{M}+\mathbf{E})^T\mathbf{V}^T\mathbf{V}\mathbf{y} - \mathbf{a}\} \tag{58}$$

and, finally,

$$\hat{\mathbf{p}} = [(\mathbf{M}+\hat{\mathbf{E}})^T\mathbf{V}^T\mathbf{V}\mathbf{M}]^{-1}(\mathbf{M}+\hat{\mathbf{E}})^T\mathbf{V}^T\mathbf{V}\mathbf{y} + \mathbf{p}_a(\hat{\mathbf{E}}) \tag{59}$$

where

$$\mathbf{p}_a(\hat{\mathbf{E}}) = [(\mathbf{M}+\hat{\mathbf{E}})^T\mathbf{V}^T\mathbf{V}\mathbf{M}]^{-1}\mathbf{B}^T\{\mathbf{B}[(\mathbf{M}+\hat{\mathbf{E}})^T\mathbf{W}^T\mathbf{W}\mathbf{M}]^{-1}\mathbf{B}^T\}^{-1} \times \tag{60}$$

$$\times\ \{\mathbf{a} - \mathbf{B}[(\mathbf{M}+\hat{\mathbf{E}})^T\mathbf{V}^T\mathbf{V}\mathbf{M}]^{-1}(\mathbf{M}+\hat{\mathbf{E}})^T\mathbf{V}^T\mathbf{V}\mathbf{y}\} \ . \tag{61}$$

Additionally, Eq.(29) is in use.

10 Distinct System Uncertainty $\Delta\mathbf{p}$

Distinct system uncertainties $\Delta\mathbf{p}$ are taken into consideration [9], now, in addition to the measurement deterioration as discussed in the preceding sections. From Eq.(12), parameters $\bar{\mathbf{p}}$ in the vicinity of $\hat{\mathbf{p}}$ are considered. They cause a bigger error matrix $\bar{\mathbf{E}}$

$$\bar{\mathbf{E}} = (\mathbf{y} - \mathbf{M}\bar{\mathbf{p}})\bar{\mathbf{p}}^T/(1 + \bar{\mathbf{p}}^T\bar{\mathbf{p}}) \ . \tag{62}$$

The optimal $\hat{\mathbf{p}}$ is the Total Least Squares or the Model Least Squares optimum. The question arises which parameter tolerance $\Delta\mathbf{p}$ from $\bar{\mathbf{p}} = \hat{\mathbf{p}} + \Delta\mathbf{p}$ has to be taken into consideration when $\bar{\mathbf{E}}$ deviates from the optimal $\hat{\mathbf{E}}$ by $\Delta\mathbf{E}$ where $\bar{\mathbf{E}} = \hat{\mathbf{E}} + \Delta\mathbf{E}$.

Combining Eq.(62) and $\bar{\mathbf{p}} = \hat{\mathbf{p}} + \Delta\mathbf{p}$ and applying $1/(1+N) \doteq 1 - N$ for $N \ll 1$, one has

$$\begin{aligned}
\bar{\mathbf{E}} &= [\mathbf{y} - \mathbf{M}\hat{\mathbf{p}} - \mathbf{M}\Delta\mathbf{p}](\hat{\mathbf{p}} + \Delta\mathbf{p})^T/[1 + (\hat{\mathbf{p}} + \Delta\mathbf{p})^T(\hat{\mathbf{p}} + \Delta\mathbf{p})] && (63)\\
\bar{\mathbf{E}} &= \frac{(\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})\hat{\mathbf{p}}^T - \mathbf{M}\ \Delta\mathbf{p}\ \hat{\mathbf{p}}^T + (\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})\Delta\mathbf{p}^T - \mathbf{0}}{1 + \hat{\mathbf{p}}^T\hat{\mathbf{p}} + 2\Delta\mathbf{p}^T\hat{\mathbf{p}} + 0} && (64)\\
\bar{\mathbf{E}} &= \frac{[(\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})\hat{\mathbf{p}}^T - \mathbf{M}\ \Delta\mathbf{p}\ \hat{\mathbf{p}}^T + (\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})\Delta\mathbf{p}^T]}{1 + \hat{\mathbf{p}}^T\hat{\mathbf{p}}}(1 - \frac{2\Delta\mathbf{p}^T\hat{\mathbf{p}}}{1 + \hat{\mathbf{p}}^T\hat{\mathbf{p}}}) && (65)\\
\bar{\mathbf{E}} &= \underbrace{\frac{(\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})\hat{\mathbf{p}}^T}{1 + \hat{\mathbf{p}}^T\hat{\mathbf{p}}}}_{\hat{\mathbf{E}}} + \underbrace{\frac{-\mathbf{M}\ \Delta\mathbf{p}\ \hat{\mathbf{p}}^T + (\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})\Delta\mathbf{p}^T}{1 + \hat{\mathbf{p}}^T\hat{\mathbf{p}}} - \frac{2(\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})\hat{\mathbf{p}}^T\Delta\mathbf{p}^T\hat{\mathbf{p}}}{(1 + \hat{\mathbf{p}}^T\hat{\mathbf{p}})^2}}_{\Delta\mathbf{E}} && (66)
\end{aligned}$$

$$\Delta\mathbf{E}\underbrace{(1 + \hat{\mathbf{p}}^T\hat{\mathbf{p}})^2}_{\hat{=}\alpha^2} = [-\mathbf{M}\ \Delta\mathbf{p}\ \hat{\mathbf{p}}^T + (\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})\Delta\mathbf{p}^T]\underbrace{(1 + \hat{\mathbf{p}}^T\hat{\mathbf{p}})}_{\hat{=}\alpha} - \underbrace{2(\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})\hat{\mathbf{p}}^T}_{\hat{=}\mathbf{B}}\underbrace{\hat{\mathbf{p}}^T}_{\hat{=}\mathbf{a}^T}\Delta\mathbf{p} \ . \tag{67}$$

Since $\mathbf{col}(\mathbf{Ba}^T\mathbf{c}) = (\mathbf{col\ B})\mathbf{a}^T\mathbf{c} = [\mathrm{diag}\{a_i\mathbf{col\ B}\}]\mathbf{c}$, one finds

$$\alpha^2\mathrm{col}\Delta\mathbf{E} = -\alpha(\hat{\mathbf{p}} \otimes \mathbf{M})\Delta\mathbf{p} + \alpha\Big[\mathbf{I}_{n_p} \otimes (\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})\Big]\Delta\mathbf{p} - 2\Big[\mathbf{diag}\{p_i\mathbf{col}[(\mathbf{y} - \mathbf{M}\hat{\mathbf{p}})\hat{\mathbf{p}}^T]\}\Big]\Delta\mathbf{p} \tag{68}$$

$$\begin{aligned}\Delta\mathbf{p} &= \alpha^2\Big(\underbrace{-\alpha[\hat{\mathbf{p}}\otimes\mathbf{M}-\mathbf{I}_{n_p}\otimes(\mathbf{y}-\mathbf{M}\hat{\mathbf{p}})]-2\,\mathbf{diag}\{p_i\mathbf{col}[(\mathbf{y}-\mathbf{M}\hat{\mathbf{p}})\hat{\mathbf{p}}^T]\}}_{\hat{=}\boldsymbol{\Omega}}\Big)^{-1}\mathbf{col}\Delta\mathbf{E}\\ \Delta\mathbf{p}^\star &= \Delta\mathbf{p}=\alpha^2\boldsymbol{\Omega}^{-1}\mathbf{col}\Delta\mathbf{E}\,. \qquad (69)\end{aligned}$$

The parameter tolerance $\Delta\mathbf{p}$ provides minimum Frobenius norm distance to a predetermined $\Delta\mathbf{E}$, as outlined in Eq.(20). Bigger $\Delta\mathbf{p}$ would yield worse approximation.

The Frobenius norm of $\Delta\mathbf{p}^\star$ can be bounded by

$$||\Delta\mathbf{p}^\star||_F \leq (1+\hat{\mathbf{p}}^T\hat{\mathbf{p}})^2\;||\Delta\mathbf{E}||_F/\sigma_{\min}[\Omega]\,. \qquad (70)$$

If measurements are available for a set of distinct system uncertainties less than $\Delta\mathbf{E}$, then Eq.(69) provides the minimum norm parameter deviation. The expected $\Delta\mathbf{p}$ will exceed $\Delta\mathbf{p}^\star$.

11 Conclusion

In view of the weak convergence facilities of the Total Least Squares problem and with respect to empirical data, the approach of "Model Least Squares" is suggested, considering the system matrix M deteriorated and the measured vector **y** free of noise. This corresponds to the field of process simulation and identification where only the present measurement is free of noise and every delayed information is contaminated with errors. A set of two equations in the error matrix E and the parameter **p** has to be solved alternatively, providing a generalized treatment of the Total, the Model and the ordinary Least Squares. The Model Least Squares Approach shows a wide range of attraction and excellent convergence facilities.

By means of weighting matrices **V** and **U**, the problems of contamination with noise can be confined to practically every submatrix or some or all rows or columns subject to measurement noise.

The approach is extended to a recursive algorithm and to the case of distinct process uncertainty. The latter case proves useful for robust modelling and identification.

A formula is presented relating the minimum parameter deviation $\Delta\mathbf{p}$ to the upper bound $\Delta\mathbf{E}$ of measured distinct system uncertainty.

References

[1] *Björk, Å.,* Least Squares Methods. In: Ciarlet, P.G., and Lions, J.L., (Eds.), Handbook of Numerical Analysis, Vol. 1. North-Holland, Amsterdam, 1990, pp. 465-647

[2] *Van Huffel, S., and Vandewalle, J.,* Analysis and properties of the generalized total least squares problem **AX** ≈ **B** when some or all columns in **A** are subject to error, *SIAM J. Matrix Anal. Appl.* **10** (1989), pp. 294-315

[3] *Lawson, C.L., and Hanson, R.J.,* Solving Least Squares Problems. Prentice Hall, Englewood Cliffs, 1974

[4] *Mendel, J.M.,* Discrete Techniques of Parameter Estimation. Dekker, New York , 1973

[5] *Golub, G.H., and Van Loan, C.F.,* An analysis of the total least squares problem, *SIAM J. Numer. Anal.* **17** (1980), pp. 883-893

[6] *Demmel, W.J.,* The smallest perturbation of a submatrix which lowers the rank and constrained total least squares problems, *SIAM J. Numer. Anal.* **24** (1987), pp.199-206

[7] *Bard, Y.,* Nonlinear Parameter Estimation. Academic Press, New York, 1974

[8] *Papageorgiou, M.,* Optimierung. Oldenbourg, München Wien, 1991

[9] *Weinmann, A.,* Uncertain Models and Robust Control. Springer, Wien New York, 1991

TARGET ESTIMATION TECHNIQUES

V. BOULET*, E. DRUON*, E. DUFLOS*,
P. BORNE**, D. WILLAEYS*** and P. VANHEEGHE*

*I.S.E.N.
Département Signaux et Systèmes,
41 Bld VAUBAN
59046 LILLE
FRANCE

**Ecole Centrale LILLE
Cité Scientifique - BP 48
59651 VILLENEUVE D'ASCQ
FRANCE

*** Laboratoire d'Automatique
Industrielle et Humaine, U.V.H.C.
Le Mont Houy
59326 VALENCIENNES
FRANCE

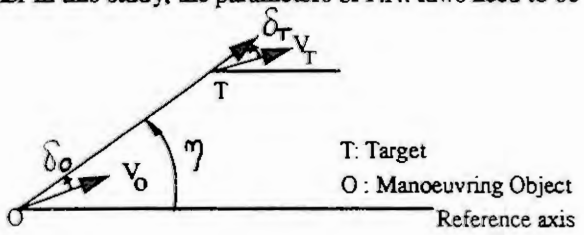
Abstract: This paper concerns the study of three parameters processed by a fuzzy algorithm in order to estimate the real target in a multitarget environment.

I Introduction

The problem is to estimate, as soon as possible, the target of maneuvering objects directed towards different possible targets with different techniques using fuzzy logic. This method, widely used in estimation and decision applications, seems interesting to investigate to solve this problem[1] [2].

In this paper, maneuvering objects are considered controlled by Proportional Navigation laws (P.N. laws) [3]. Each maneuvering object is assumed to be directed towards a pre-defined non-maneuvering target. The velocities of both targets and maneuvering objects are assumed constant. Target velocities are supposed to be very low compared to maneuvering object velocities.

Before going further in this study, the parameters of P.N. laws need to be briefly defined.



δ_O is the angle between $\vec{V}_O$ and line of sight OT
δ_T is the angle between $\vec{V}_T$ and line of sight OT
r is the distance OT
η is the angle between the Reference axis and line of sight OT

figure 1 : Notations

The kinematic equations of maneuvering object O, controlled by a P.N. law with coefficient A, and directed towards target T are:

$$\dot{r} = V_T \cos\delta_T - V_O \cos\delta_O \qquad (1)$$

$$r \cdot \dot{\eta} = -V_T \sin\delta_T + V_O \sin\delta_O \qquad (2)$$

$$d\delta_0 = (1-A)\, d\delta_T \qquad (3)$$

It seems important to notice that these equations are derived for a target and for a maneuvering object directed towards that target.

II Contribution of the estimation of the P.N. coefficient to the target estimation problem

The trajectory of an object controlled by a P.N. law is directly determined by the value of the P.N. coefficient according to equation (3). This coefficient belongs to a specific bounded interval. The real target is characterized by a constant P.N. coefficient during all the length of the trajectory when the study is done from the point towards which the maneuvering object is directed. What is now the evolution of this coefficient if the study is done from an other point?

For more convenience the P.N. coefficient, considered as a function of time t along the trajectory studied from the target point T_i, will be called $A_i(t)$. The curve of the trajectory being the same whatever the point from where it is derived, it can be deduced that if T_1 belongs to the space generated by the tangents to the trajectory, it exists a time t_0 for which $A_1(t_0)$ becomes infinite. From the integration of the equations (1) (2) (3) under the hypothesis that V_T is very small compared to V_O, it comes that $A_1(t)$ decreases towards zero as the object is approaching towards the target. A non target point is then characterized by a decrease towards zero of its P.N. coefficient $A_1(t)$ as the object is approaching towards the target and/or by a divergence towards infinity at a particular point. The estimation versus time of the P.N. coefficient is consequently a possible way to estimate the real target within different possible targets. Using an extended Kalman filter to estimate the theoretical P.N. coefficient and the kinematic parameters of the object [4], simulations have been performed and confirm these results.

III Optimal control approach

An optimal guidance law that minimizes terminal miss distance can be derived using linear optimal control theory[3][5]. By making several simplifying assumptions (zero acceleration target, maneuvering object completely controllable with perfect response and small line of sight angles) the optimal guidance law can be reduced to P.N. law with a coefficient A equal to three.

It is assumed that sensors are sending values of r and η every Δt. Normal acceleration Γ_i and optimal control vector $\hat{u}_i$ (optimal normal acceleration), seen by each possible target T_i can be derived using Kalman filtered parameters $\dot{r}$, $\dot{\eta}$ and A. The idea is to estimate, for each possible target, the optimal control vector $\hat{u}_i$ and c_i: the difference between Γ_i and $\hat{u}_i$. Using the property that the normal acceleration of the real target tends towards zero and that this convergence is faster than for the other possible targets, the complementary evolutions of $\hat{u}_i$ and c_i can be used to estimate as soon as possible the most possible targets.

IV Conclusion: Fuzzy algorithm

For the optimal control approach, in order to separate the different targets, fuzzy sets are built to take into account the fast convergence of $\hat{u}_i$ and c_i towards zero for the real target. Membership functions are built so that the limits for some fuzzy sets are calculated dynamically and tend to emphasize the importance of the area around zero.

For the estimation of the P.N. coefficient, fuzzy sets emphasize the fact that the P.N. coefficient has to be constant for the real target.

Different sets of rules based on decision tables taking into account the particular behavior of the considered parameters are merged using max-min methods.

The defuzzification process is obtained by a centroid defuzzification technique.

The results obtained so far with simplified decision tables show some interesting possibilities of such an approach. Once tuned, the fuzzy algorithm estimates the correct target rapidly with only a short transition time.

REFERENCES

[1] TERANO T., ASAI K., SUGENO M. «Fuzzy Systems Theory and its Applications», Academic Press, 1992

[2] BOUCON B. - Actes du Colloque Intelligence Artificielle et Sciences Humaines, Lyon, 1991

[3] CARPENTIER R. «Guidage des avions et des missiles aérodynamiques» - Cours ENSAE

[4] DELLERY B. «Modélisation et caractérisation de la trajectoire d'un mobile manoeuvrant dans le cadre du problème de la poursuite. Application aux problèmes de l'estimation et de l'extrapolation d'état» - Thèse de doctorat - Université d'AIX-MARSEILLE - 1983

[5] RIGGS «Linear optimal guidance for short range air-to-air missiles» -Proceedings of NAECON-Vol 2-(5/79)

OBSERVERS FOR SINGLE INPUT GENERALIZED STATE SPACE SYSTEMS

By D.LEFEBVRE and F.ROTELLA.
Laboratoire d'Automatique et d'Informatique Industrielle, URA CNRS D1440
Ecole Centrale de Lille, BP 48,
59651 Villeneuve d'Ascq, France.
email: rotella%idnges,dnet@decip.citilille.fr

Abstract The aim of this paper is the design of observers for single input generalized state space systems in continuous time. A dynamic precompensator is used to normalize the system. Full and reduced order observers are described.

1. INTRODUCTION

Let us consider the process which can be described by a generalized state space model in continuous time:

$$\begin{array}{l} E\dot{x} = Ax + Bu, \\ y = Cx, \end{array} \tag{1}$$

where $x \in I\!R^n, y \in I\!R^q$, and u is a scalar. E is a square matrix of rank $r < n$. An unique solution $x(t)$ exists for all $u(t)$, if Δ, tbe determinant of $(sE - A)$, is not identically zero. We suppose that the previous condition is true in the following, and we define d as the degree of Δ. The study of generalized state space systems was developped by many autbors [1, 2, 3, 4, 5, 8], who mention that a lot of difficulties arise in control problems, because of dynamic impulsive motions. A way of bypassing these discontinuities in the response of the system is to separate tbe singular part from the regular one using systems equivalences, to study each part in the appropriate way. The result of this decomposition is called [2] the standard form of (1):

$$\begin{array}{ll} \dot{x_1} = A_1x_1 + B_1u, & (a) \\ N\dot{x_2} = x_2 + B_2u, & (b) \\ y = C_1x_1 + C_2x_2, & \end{array} \tag{2}$$

where A_1 is a $d * d$ matrix, and N a nilpotent matrix of index ν:

$$N = \mathrm{diag}\{N_i = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 1 & 0 \\ 0 & \dots & \dots & 0 & 1 \\ 0 & \dots & \dots & \dots & 0 \end{pmatrix}\}.$$

The obtention of the standard form was developped by Kailath, Verghese, Lewis, Ozcaldiran, Rotella, and Lefebvre [1, 6, 7, 8, 9]. This form is useful to solve a lot of control problems, but some difficulties remain particularly with the observer design, when the knowledge of the singular part of the state is required. Concretely, the construction of a singular observer of (1) is not possible. Normal observers are proposed by many authors. Methods using the singular value decomposition of the matrix E were developped by El Tohami *et al.*, Verhaegen and Van Dooren, Fahmy and O'Reilly, Shields [13, 14, 15, 16]. Another method using the Moore-Penrose inverse of the matrix C was proposed by Shafai and Caroll. [17]. A common drawback of the previous methods is that specific assumptions are needed in all cases.

We introduce another approach based on the addition of a dynamic precompensator that normalize the system. We call augmented form the resulting regular representation, and we build an observer of this form. The first section deals with the description of a dynamic precompensator, and of the resulting augmented form. In the second section we apply this result to build full and reduced order observers of x_1. In the third section we define full and reduced order observers of (1).

2. PRECOMPENSATOR DESIGN

Let us consider the generalized system (1) under its standard form. The equation (2b) allows us to write x_2 as a linear combination of the successive derivatives of u [12]:

$$x_2 = -\sum_{i=0}^{\nu-1} N^i B_2 u^{(i)}. \tag{3}$$

Defining v, a new input, as the $(\nu-1)^{\text{th}}$ derivative of u, and the vector ϵ as $(\, u \quad \dots \quad u^{(\nu-2)}\,)^T$, a realization of the precompensator

$$u = \frac{1}{s^{\nu-1}} v, \tag{4}$$

is given by:

$$\begin{aligned} \dot\epsilon &= A_p\epsilon + B_p v, \\ u &= C_p\epsilon, \end{aligned} \tag{5}$$

with:

$$A_p = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ \vdots & & & \ddots & 1 \\ 0 & \dots & \dots & \dots & 0 \end{pmatrix}, \quad B_p = \begin{pmatrix} 0 \\ \vdots \\ \vdots \\ 0 \\ 1 \end{pmatrix}, \quad C_p = (\, 1 \quad 0 \quad \dots \quad \dots \quad 0\,).$$

Considering now the augmented state vector z as $(\, x_1^T \quad \epsilon^T\,)^T$, and $\bar B_1$ as $(\, B_1 \quad 0 \quad \dots \quad 0\,)$, we can define the following regular state space representation, called the augmented form, of dimension $\bar n = n_1 + \nu - 1$:

$$\begin{aligned} \dot z &= \bar A z + \bar B v, \\ y &= \bar C z + \bar D v, \end{aligned} \tag{6}$$

with:

$$\bar A = \begin{pmatrix} A_1 & \bar B_1 \\ 0 & A_p \end{pmatrix} \in I\!R^{\bar n * \bar n}, \quad \bar B = \begin{pmatrix} 0 \\ B_p \end{pmatrix} \in I\!R^{\bar n}, \quad \bar D = -C_2 N^{\nu-1} B_2 \in I\!R,$$

$$\bar C = (\, C_1 \quad -C_2 B_2 \quad \dots \quad \dots \quad -C_2 N^{\nu-2} B_2\,) \in I\!R^{1 * \bar n}.$$

3. OBSERVER DESIGN FOR x_1

Let us first consider an observer of the augmented form. Because this representation is regular, we can apply usual methods to build an observer of z. If the pair $(\bar A, \bar C)$ is observable, an observer of the augmented form:

$$\begin{aligned} \dot{\hat z} &= \bar A\hat z + \bar B v + \bar K(\bar C z - \bar C\hat z), \\ &= (\bar A - \bar K\bar C)\hat z + (\bar B - \bar K\bar D)v + \bar K y, \end{aligned}$$

could be obtained very easily by the Bass and Gura method [11], using a gain matrix $\bar K$ such that the eigenvalues of $\bar A - \bar K\bar C$ are negative.

But the observation of the complete state is not required to control the system (6). In fact the components $u, \dot u, \dots, u^{\nu-1}$, and thus the vector ϵ are directly accessible in the augmented form through the precompensator. Thus, we only need the observation of the state vector x_1. A full order observer of x_1 exists, provided the pair (A_1, C_1) is observable [4] (*i.e.* the system (2) is R-observable). It is given by:

$$\begin{aligned} \dot{\hat x}_1 &= A_1\hat x_1 + B_1 u + K_1(y_1 - C_1\hat x_1), \\ &= (A_1 - K_1C_1)\hat x_1 + B_1 u + K_1 y_1, \\ &= (A_1 - K_1C_1)\hat x_1 + B_1 u + K_1(y - y_2). \end{aligned}$$

The measurement of y_1 is not accessible but can be obtained from the difference between y and $y_2 = C_2x_2$. Using the relation (3), informations are taken from the output of the successive integrators of the precompensator to build the observer:

$$\begin{aligned}\dot{\hat{x}}_1 &= (A_1 - K_1C_1)\hat{x}_1 + (B_1 + K_1C_2B_2)u + K_1C_2 \textstyle\sum_{i=1}^{\nu-1} N^i B_2 u^{(i)} + K_1 y,\\ &= (A_1 - K_1C_1)\hat{x}_1 + M_1 (\,\epsilon^T \quad v\,)^T + K_1 y,\end{aligned}$$

where:

$$M_1 = (\, B_1 + K_1B_2C_2 \quad K_1C_2NB_2 \quad \dots \quad K_1C_2N^{\nu-1}B_2 \,) \in I\!R^{1*\nu}$$

Let us denote $PCOMP$ the precompensator, composed of a series of $\nu - 1$ integrators, we have the following structure for a full order observer:

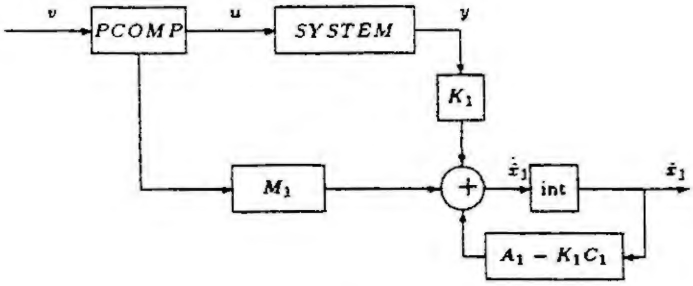


Fig. 1: Observer of x_1

It is obvious that, provided the matrix C_1 is of full row rank q, a reduced $d - q$ order observer of x_1 can be obtained [11], but for the sake of brevity, it won't be detailed here.

4. OBSERVER FOR THE GENERALIZED SYSTEM

The final point is the determination of an observer of the vector $x = (\, x_1^T \quad x_2^T \,)^T$. Such an observer exists, provided the generalized state space system is R-observable. The observation of x_1 is obtained in the same way as previously, and the vector x_2 is obtained with a matrix L_1 composed of combinations of the components of ϵ. L_1 collects informations from the different stages of the precompensator:

$$L_1 = (\, -B_2 \quad -NB_2 \quad \dots \quad -N^{\nu-1}B_2 \,) \in I\!R^{1*\nu}$$

such that $x_2 = L_1 (\,\epsilon^T \quad v^T\,)^T$.

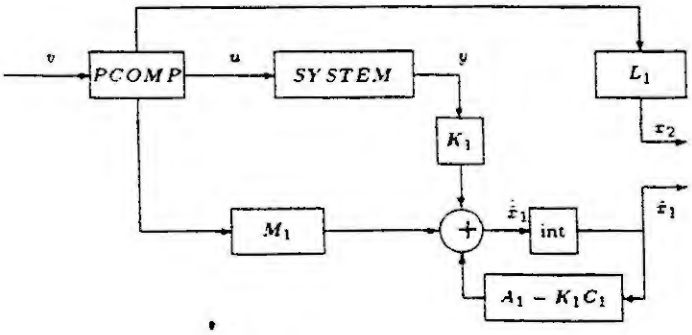


Fig. 2: Observer of x

Provided the matrix C_1 is of full row rank q, a reduced $n - q$ order observer of x can also be obtained.

5. CONCLUSION

In this paper, we have proposed an observer for single input generalized systems. The standard form is required, and an additional dynamic precompensator composed of a series of integrators is used to normalize the system. Informations are taken on each component of this precompensator to estimate the vector x_2. The observation of the vector x_1 is obtained using full or reduced order observers.

The extension to multi inputs systems is now in development. Because the dimension of the augmented state z increases with the number of inputs, a minimal dynamic precompensator is required [12]. The previous results will be adapted with such a precompensator.

References

[1] G.C. VERGHESE, B.C. LEVY, T. KAILATH , *A generalized state-space for singular systems*, IEEE, *Trans. Auto. Cont.*, Vol. AC-26. no. 4, pp 811-830, August 1981.

[2] F.R. GANTMACHER, *Theory of matrices*, New York, Chelsea, 1959.

[3] H.H. ROSENBROCK, *Structural properties of linear dynamical systems*, Int. J. Cont. Vol. 20, no. 2, pp 191-202, 1974.

[4] L. DAI, *Singular control systems*, Lecture notes in control and information sciences, Springer-Verlag, pp 4-22, 1989.

[5] F. ROTELLA, *Singular systems*, Concise encyclopedia of modelling and simulation, Pergamon Press, pp 435-442, 1992.

[6] F.L. LEWIS, K. OZCALDIRAN, *The relative eigenstructure problem and descriptor systems*, SIAM National Meeting, June 1983.

[7] G.C. VERGHESE, T. KAILATH, *Eigenvector chains for finite and infinite zeros of rational matrices*, 18th IEEE Conf. Decision and Control, December 1979.

[8] F.L. LEWIS, *A survey of linear singular systems*, Circuits Systems Signal Process, Vol. 5, no. 1, 1986.

[9] F. ROTELLA, D LEFEBVRE, *Generalized state space systems: Obtention of the standard form*, submitted March 1993.

[10] T. KAILATH, *Linear systems*, Englewood Cliffs, Prentice-Hall, 1980.

[11] P. BORNE, G. DAUPHIN-TANGUY, J.P. RICHARD, F. ROTELLA, I. ZAMBETTAKIS, *Commande et optimisation des processus*, Technip, pp 37-43, 1990.

[12] D. LEFEBVRE, F. ROTELLA, An augmented form for generalized state space systems IEEE/SMC, vol. 4, pp.615-620, Le Touquet, France,1993.

[13] M. EL TOHAMI, V. LOVASS-NAGY, R. MUKUDAN , *On the design of observers for generalized state space systems using singular value decomposition*, Int. J. Cont., Vol. 38, no. 3, pp 673-683, 1983.

[14] B. VERHAEGEN, P. VAN DOOREN, *Observers for singular systems*, Syst. and Cont. Letters, Vol. 8, no. 29, 1986.

[15] M.M. FAHMY, J. O'REILLY, *Observers for descriptor systems*, Int. J. Cont., Vol. 49, no. 6, pp 2013-2028, 1989.

[16] D.N. SHIELDS, *Observers for descriptor systems*, Int. J. Cont., Vol. 55, no. 1, pp 249-256, 1992.

[17] B. SHAFAI, R.L. CAROLL, *Design of a minimal order observer for singular systems*, Int. J. Cont., Vol. 45, no. 3, pp 1075-1081, 1987.

[18] ARMANTANO, , Syst. and Cont. Letters, Vol. 4, no. , 1984.

[19] C.W. YANG, H.L. TAN, *Observer design for singular systems with unknown inputs*, Int. J. Cont., Vol. 49, no. 6, pp 1937-1946, 1989.

Mathematical Modeling and Computational Principles for the Analysis and Simulation of Long–Distance Energy Systems

KURT SCHLACHER, ANDREAS KUGI
Johannes Kepler University Linz
Institute for Automatic Control and Electrical Drives
Altenbergerstraße 69
A-4040 Linz, Auhof

Abstract. Methods for the modeling and simulation of long–distance energy systems are considered. Due to Kirchhoff's laws a special type of non linear equation is crucial for the steady state and the transient analysis. A reliable algorithm to solve these equations based on Newton's method is presented. The uniqueness of the solution and the convergence of the method are proved by the stability theory of Liapunov. To apply this method, the calculation of some derivatives is necessary. An approach for C++ to do this in an automatic way without rewriting a program is presented.

1. INTRODUCTION

Long–distance energy systems are growing fast because of ecological improvements . The size, the complexity and the nonlinearity of the hydraulic equations command the modeling and simulation of these systems. Since a computer must be used to integrate the differential equations, it is natural to use it for setting up the equations, too.

The propagation of temperature and pressure are the most important phenomena in these systems. One builds the network for the first effect but a big part of the control equipment is needed to handle the second one. The wave speed of a water hammer is about $1000[ms^{-1}]$. The delay due to the finite propagation velocity is not negligible any more in energy systems for whole cities. Simulation based on mathematical modeling and reliable algorithms offer a good way in understanding the phenomena of those systems.

2. HYDRAULIC NETWORKS

A hydraulic network is formed by connecting together terminals like pumps, valves, pipes, etc.. The connection points are called *nodes* and the terminals connecting two nodes are called *branches*. This type of network obeys the two laws of Kirchhoff. The *current law* says that the total mass flow to a node is zero. The density of water may be assumed to be constant in the interesting range of temperature and pressure. Therefore one can replace the mass flow by the volume flow. The *voltage law* says that the difference of the pressure of two nodes connected by a branch is equal to the difference pressure of this branch. Now the most important branches are presented.

2.1. Pipes

The one–dimensional transient flow in pipes can be described by a pair of quasi linear hyperbolic partial differential equations of first order

$$\frac{\partial}{\partial x}h(x,t)+\frac{1}{gA}\frac{\partial}{\partial t}q(x,t)+\frac{f}{2gDA^2}\left|q(x,t)\right|q(x,t)=0 \quad \text{and} \quad \frac{\partial}{\partial t}h(x,t)+\frac{a^2}{gA}\frac{\partial}{\partial x}q(x,t)=0\ .$$

$h(x,t)[m]$ denotes the pressure head, $q(x,t)[m^3 s^{-1}]$ the volume flow, A the area of the pipe, D the pipe diameter, g the gravitational acceleration, f the friction factor and a the equivalent wave speed [7]. The boundary conditions of a pipe of length l at a time t are of the form

$$h(l,t) = -\frac{a}{gA}q(l,t) + R_1 \ , \quad h(0,t) = -\frac{a}{gA}q'(0,t) + R_2 \quad \text{and} \quad q'(0,t) = -q(0,t)$$

with R_1 and R_2 depending only on values for $\tau < t$. These conditions describe a 1-port which consists of a linear port and a controlled pressure source in series. The minus sign for q' is used to have the same flow reference direction on both end points. The boundary conditions of a pipe can be embedded in the theory of 1-port networks [4].

2.2. Pumps

In long-distance energy systems only rotary pumps are used. As a good approximation the transient behavior of a pump is described by the steady state characteristic curves of the total head increase $dh = dh(q,\omega)$ and the shaft torque $m(q,\omega)$ with ω as the angular velocity. The torque equation takes the form

$$\theta\frac{d}{dt}\omega = m_d(\omega) - m(q,\omega)$$

with θ as the moment of inertia and $m_d(\omega)$ the torque of the driving motor [7]. If necessary the dynamics of the driving motor is taken into consideration, too.

2.3. Valves

The difference pressure of the valve is given by

$$dh = -r\frac{|q|q}{\alpha^2} \quad \text{and} \quad \frac{d}{dt}\alpha = f(q,\alpha,u) \quad \text{with} \quad 0 \le \alpha \le 1$$

with r as the resistance factor, α as the grade of opening and u as the control input. The topology of the hydraulic network changes at each time, when a switching component like a valve is opening or closing [7].

2.4. Steady state and transient analysis

A large system of nonlinear equations has to be solved in the case of steady state analysis. The transient behavior is described by systems of partial and ordinary differential equations with nonlinear equations as restrictions. The latter ones are a consequence of Kirchhoff's circuit laws. Investigations about the computing time have shown, that the solution of this system of nonlinear equations takes much more time than the integration of the differential equations. These circuit equations are described by the theory of 1-port networks most efficiently. They are of the same type for the steady state and the transient analysis, therefore only the first one is treated. It is very important to use efficient algorithms to solve this problem in order to obtain a short computing time. A fast algorithm will be presented after the uniqueness of the solution of the hydraulic network equations is shown.

3. FACTS OF NETWORKS

Only the steady state equations are considered according to the above-mentioned observations. The circuit is formed by connecting together special terminals. A *graph* $G = \{N, B\}$ is related to the network. N is the finite set of *nodes* of cardinality n and B is the finite set of *branches*. A branch has exactly two end points which must be nodes. The number of branches is b. The flow q^i and the difference pressure dh_i are assigned to the branch i. The pressure h_j is related to a node j. Now the laws of Kirchhoff can be expressed as

$$h_j d_l^j + h_k d_l^k = dh_l \qquad \text{and} \qquad \sum\nolimits_j d_j^i q^j = 0$$

with

$$d_l^i = \begin{cases} 1 & \text{if node } i \text{ and branch } l \text{ are connected, the direction of the flow to } i \text{ is positive} \\ -1 & \text{if node } i \text{ and branch } l \text{ are connected, the direction of the flow off } i \text{ is positive} \\ 0 & \text{otherwise}. \end{cases}$$

The *flow* q of a network is a point $q = (q^1, \ldots, q^b) \in \mathbf{R}^b$. q is said to be admissible if q satisfies Kirchhoff's current law. A *difference pressure* dh is a point $dh = (dh_1, \ldots, dh_b) \in (\mathbf{R}^b)^*$. dh is said to be admissible if dh satisfies Kirchhoff's voltage law. Let ρ be the natural bilinear map $\rho : (\mathbf{R}^b)^* \times \mathbf{R}^b \to \mathbf{R}$

$$\rho(dh, q) = \sum_j dh_j q^j$$

then Tellegen's theorem can be expressed in the following form [5].

Theorem 1: *For all admissible q and dh is*

$$\rho(dh, q) = 0 \ .$$

The linear map $D : \mathbf{R}^b \to \mathbf{R}^n$

$$x^i = \sum_j d_j^i q^j$$

is well defined. For a basis $\{a_i\}$ of Ker D there exists a linear map $A : \mathbf{R}^c \to \text{Ker } D$

$$q^i = \sum_j a_j^i x^j$$

which is bijective for some c. Tellegen's theorem leads now to the following results [2].

Theorem 2: *A flow q is admissible iff*

$$Dq = 0 \qquad or \qquad q = Ax \ . \tag{1}$$

A difference pressure dh is admissible iff

$$dh \in \text{Im } D^* \qquad or \qquad A^* dh = 0 \ . \tag{2}$$

It is a well known fact that one can always find a basis of Ker D that x^j is the flow q^i of a branch i.

4. A POTENTIAL FUNCTION

Only 1-ports are considered as branches for this type of network. Then there is a branch equation of the form

$$\text{a)} \quad dh_i = f_i(q^i) \ , \quad \text{b)} \quad dh_i = \text{const.} \quad \text{or} \quad \text{c)} \quad q^i = \text{const.}$$

for each branch i. For the sake of simplicity only equations of type a) are considered, the extensive case is presented in [4]. The short form of the branch equations is

$$dh = f(q) \ . \tag{3}$$

A flow q is said to be a solution of the network equations if q and $dh(q)$ are admissible (equations (1), (2)) and satisfy the equation above or

$$q = Ax \qquad \text{and} \qquad A^* f(q) = 0 \tag{4}$$

hold. Based on Tellegen's theorem we define the function

$$v(q) = -\int_0^q \rho(f(x), dx) = -\sum_i \int_0^{q^i} f_i(x) dx \ .$$

For an admissible flow the derivative of v with respect to x^i is given by

$$\frac{\partial}{\partial x^i} v(Ax) = \frac{\partial}{\partial q} v(Ax) \frac{\partial}{\partial x^i}(Ax)(x) = -\sum_j f_j(\sum_l a_l^j x^l) a_i^j = -\rho(f(Ax), a_i) \ .$$

Each solution of the network satisfies the equation

$$\frac{\partial}{\partial x} v(Ax) = 0 \ .$$

Now it is possible to give the main result of this paper.

Theorem 3: *If the branch equations f_i (3) of the network satisfy the conditions $f_i \in C^1(-\infty,\infty)$ and*

$$\frac{d}{dx}f_i(x) \leq \varepsilon < 0 \ ,$$

then the solution q of the network equations (4) is unique.

To proof this theorem first the functions

$$g_i(q^i) = -\int_0^{q^i} f_i(Ax)dx^i$$

are considered. It can be easily shown that g_i is *strictly convex* [3] and that the condition

$$\lim_{|q^i|\to\infty} g_i(q^i) = \infty$$

is satisfied. $v(q)$ is strictly convex, since $v(q)$ is the sum of strictly convex functions and the condition

$$\lim_{\|q\|\to\infty} v(q) = \infty$$

is satisfied, too. Because of Ker $A = \{0\}$ the restriction of $v(q)$ to $v(Ax)$ is strictly convex and

$$\lim_{\|x\|\to\infty} v(Ax) = \infty$$

holds. It is a well known fact of the theory of convex functions that $v(Ax)$ has a global minimum and x is the unique solution of

$$\frac{\partial}{\partial x}v(Ax) = 0 \ .$$

This completes the proof. It must be emphasized that the conditions of theorem 3 are no restriction for hydraulic networks.

5. A MODIFIED NEWTON METHOD

Due to the theory of gradient systems and the present considerations a simple algorithm to solve the network equations is given by

$$\frac{d}{dt}x^i = \rho(f(Ax), a_i) \ .$$

In general a faster algorithm can be constructed using Newton's method [3]. This algorithm is applicable since $v(Ax) \in C^2$ and v has a unique minimum. Let J_v be the Jacobian and H_v the Hessian of v. Then the sequence

$$x(i+1) = x(i) - \alpha\,(x(i))\,H_v^{-1}\,(x(i))\,J_v^T\,(x(i)) \tag{5}$$

with the tuning function $\alpha\,(x(i))$ is well defined. The Hessian of $v(Ax)$ is positive definite since $v(Ax)$ is strictly convex. In this special case the Hessian H_v is given by

$$H_v(Ax) = A^*(-J_f)A$$

with J_f as Jacobian of f of equation (3). Expanding v into a Taylor series the equation

$$v\,(x(i+1)) = v\,(x(i)) + J_v\,(x(i))\,(x(i+1) - x(i)) + O(\|x(i+1) - x(i)\|^2)$$

follows for sufficiently small $\|x(i+1) - x(i)\|$. With the help of

$$v\,(x(i+1)) - v\,(x(i)) = -\alpha\,(x(i))\,J_v\,(x(i))\,H_v^{-1}\,(x(i))\,J_v^T\,(x(i)) + O(\alpha^2\,(x(i)))$$

one can ensure that the inequality

$$v\,(x(i+1)) - v\,(x(i)) < 0$$

holds for sufficiently small $\alpha\,(x(i)) > 0$. If the process of minimizing is considered as a (time) discrete process the theory of Liapunov gives an easy proof of the next theorem [6].

Theorem 4: *The Newton series (5) converges for each initial value to the unique solution of the network equations (4) for a sufficiently small choice of the tuning function α. If $v \in C^3$ the order of convergence is at least two provided that the initial value is sufficiently close to the solution.*

This theorem ensures not only the convergence it offers a way to choose α by checking the difference $v\,(x(i+1)) - v\,(x(i))$, too.

6. AN ALGORITHM FOR CALCULATING DERIVATIVES

The set of the network equations is too large to be set up by hand. It changes also during a simulation according to the action of switching terminals. It can be shown that the graph searching algorithm *"depth first"* offers an efficient way setting up the equations in an automatic way [4]. Although this algorithm can be used to calculate all required derivatives for the Newton procedure, too, a more general method is presented. This approach needs a modern programming language like C++ which allows overloading of functions of binary operations and dynamic memory management [1].

First a new type of variable $x = (x_v, (x_i))$ is defined as a pair of a float variable x_v and an array of float variables (x_i) of dimension one. Next the binary operations

$$x + y = (x_v + y_v, (x_i + y_i)) \ , \qquad x \cdot y = (x_v y_v, (y_v x_i + x_v y_i))$$

and the rule

$$f(x) = \left(f(x_v), (\frac{d}{dx}f(x_v)x_i)\right)$$

for the value of the real function f are introduced. Let x be a finite Taylor series of the form $x = x(z) + \sum_i \frac{\partial}{\partial z_i}x(z)dz_i$. The value of the variable x at z is defined as $x = (x(z), \frac{\partial}{\partial z_i}x(z))$. It is easy to verify that these operations implement the rules to handle this special kind of series. If these operations are used in a program all required derivatives are calculated in an automatic way. The advantage of this approach is that only a new type of variable and overloading of some operations is required. It is not necessary to write a new program.

7. RESULTS

The above introduced methods are the basis of the programming system FLUIDIX for the simulation of long-distance energy systems. This package consists of a graphic editor, a data base of mathematical models, an interactive simulator and a graphic post processor. The user only draws the plan of the circuit assisted by mouse and menus. Parameters of the models are added by the user or by the data base. The system derives the equations for the steady state or transient analysis automatically. During the simulation the user can trigger events with the mouse or keyboard. The results of a simulation are documented by the graphic post processor.

The programming system FLUIDIX has been used for the analysis of long-distance energy systems of cities with more than 250.000 inhabitants. Therefore one can say that the methods presented in this paper are applicable for real world problems.

REFERENCES

[1] Gorlen K.E., Orlow S.M., Plexico P.S.: Data Abstraction and Object-Oriented Programming in C++, John Wiley. 1991.

[2] Hirsch M., Smale S.: Differential Equations, Dynamical Systems and Linear Algebra, Academic Press, 1974.

[3] Luenberger D.G.: Linear and Nonlinear Programming, Addison Wesley, 1989.

[4] Schlacher K.: Systemanalytische Methoden für hydraulische Netze, Habilitationschrift an der TU-Graz, 1990.

[5] Penfield P., Spence Jr. R, Diunker S.: Tellegen's Theorem and Electrical Networks, MIT Press, 1970.

[6] Vidyasagar M.:Nonlinear Systems Analysis, Prentice Hall, 1993.

[7] Wylie E.B., Streeter V.L., Suo L.: Fluid Transients in Systems, Prentice Hall 1993.

Modelling Non–Precise Observations and Measurements

Reinhard Viertl

Technische Universität Wien

Real observations and measurement data are often not precise real numbers but contain different uncertainties. Besides errors and statistical variation a single measurement is often not a real number but more or less fuzzy.

This kind of uncertainty is called *imprecision* and can be modelled by so called *fuzzy numbers* which are sprecial fuzzy subsets of the real line.

Using this kind of data in stochastic models it is necessary to generalize statistical inference methods to *non-precise data*. This is possible and in the contribution generalizations of classical statistical procedures as well as Bayesian methods will be given. Moreover examples of non-precise data and their *characterizing functions* will be given.

Reference

R. Viertl: On Statistical Inference Based on Non-precise Data. In H. Bandemer (Ed.): *Modelling Uncertain Data*, Akademie Verlag, Berlin, 1993.

A MARKOV CHAINS SINGULAR PERTURBATION RESOLUTION

D. RACOCEANU, A. EL MOUDNI, M. FERNEY and S. ZERHOUNI

Laboratoire de Mécanique et Productique , Ecole Nationale d'Ingénieurs de Belfort
8, Bd. Anatole France, B.P. 525, 90016 Belfort (France)

Abstract. This paper contributes to the development of the perturbational decomposition method in the case of homogeneous markovian systems, corresponding to a discret time and discret state space. Our work concerns the adaptation of Phillips singular perturbation method to the category of ergodic Markov chains presenting the two-weighting-scale property, the equivalent of the two-time-scale property in the case of state systems.

1. INTRODUCTION

Many systems, in reason of the multiple dependency parameters, can only be studied with a random pattern. It is the case for example in failure detection, waiting phenomena, equipment wear and so on. When the pattern has a great dimension, it seems necessary to reduce it, in order to simplify its resolution.

2. DEFINITION OF THE STUDIED ELEMENTS [2]

2.1. Type of studied systems

The system that we are studying is the first order finite Markov chain. Its *fundamental formula* is :

$$\mathbf{P}(k+1) = \mathbf{P}(k) \cdot \boldsymbol{\tau} \qquad (2.1)$$

$\mathbf{P}(k)$ is the vector of absolute probabilities at the moment k, with the component $P_i(k)$ $(i=1,...,r)$, the probability that the system will be in the e_i state at the moment k, and $\boldsymbol{\tau}$ is the transition matrix with components p_{ij} $(i,j=1,...,r)$ the transition probabilities between states.

If these transition probabilities are independent of the considered moment k, the chain is called *homogeneous in time*. The transition probabilitys are called in this case *stationary*.

• ***Remark*** *:* Observing that the fundamental formula of the first order finite Markov chain has a similar form to the state equation of autonomous discrete systems, we had the idea to use for its simplification one classical method employed usually for the state equation resolution. It is the singular perturbation method.

2.2. The direct method for ergodic chains limit resolution

A chain is called *ergodic* if its limit distribution P (∞) exists, it is unique, and do not depends on the initial distribution P(0). The fundamental formula takes than the following form (I is the identity matrix r x r) :

$$P(\infty) = P(\infty)\boldsymbol{\tau} \quad \text{or} \quad P(\infty)(I - \boldsymbol{\tau}) = 0 \qquad (2.2)$$

If we note $I - \boldsymbol{\tau} = \mathcal{D}$ ($\mathcal{D}$ is called *dynamic matrix*), the (2.4) equation can be write :

$$P(\infty) \cdot \mathcal{D} = 0 \qquad (2.3)$$

The (2.5) system is a r equation system with r unknowns. The dynamic matrix is a singular matrix, so in order to solve this system (solutions will be $P_1(\infty)$, ... , $P_r(\infty)$), we add a further relation,

$$\begin{cases} [P_1(\infty) \;\; P_2(\infty) \;\; \dots \;\; P_r(\infty)] \cdot \mathcal{D} = 0 \\ P_1(\infty) + P_2(\infty) + \dots + P_r(\infty) = 1 \end{cases} \qquad (2.4)$$

• ***Remark*** *:* The presented method is simple and efficient in the case of a system with reasonable dimension (little r). In the case of a great dimension, the resolution using only this method can ask many computer aids, and despite this aids, divergences can appear when the system is bad conditioned. In order to eliminate this difficulty, we propose a method of simplification.

3. MARKOV CHAINS SINGULAR PERTURBATION MODELING

• ***Remark*** *:* the term of "two-weighting-scale", that we utilise for designate the two-time-scale property in the case of Markov chains, comes from the fact that the slow part states will have a preponderant influence in the future evolution of the system, whereas the fast part states, even if they will always be present in the systems evolution, they will do this with a small frequency and for a small period of time. So the meaning is not the same that in automatic. We call then the slow part of a stochastic system - *strong* , and the fast one - *weak*.

3.1. Two-weighting-scale definition

A stochastic system characterized by the (2.1) equation has the *two-weighting-scale* property if it can be decomposed in two unconnected subsystems :

$$\left[\mathbf{P}_s(k+1)\ \mathbf{P}_w(k+1)\right] = \left[\mathbf{P}_s(k)\ \mathbf{P}_w(k)\right] \cdot \begin{pmatrix} \mathcal{T}_s & 0 \\ 0 & \mathcal{T}_w \end{pmatrix} \qquad (3.1)$$

with $\mathbf{P}(k) = [\mathbf{P}_s(k)\ \ \mathbf{P}_w(k)]$ the decomposition of the initial vector with $\mathbf{P}_s \in \mathbb{R}^{r_1}$ and $\mathbf{P}_w \in \mathbb{R}^{r_2}$, $r_1 + r_2 = r$, such as the eigenvalues λ of the matrices $\mathcal{T}_s$ and $\mathcal{T}_r$ satisfie :

$$|\lambda_{\min}(\mathcal{T}_s)| \gg |\lambda_{\max}(\mathcal{T}_w)| \qquad (3.2)$$

The matrix $\mathcal{T}_s$ (respectively $\mathcal{T}_w$) regroupes the great module eigenvalues (respectively the little one) of the initial stochastic matrix $\mathcal{T}$. This leads to the separation of the initial system states in two parts, corresponding to the strong part and the weak one.

3.2. Application of the Phillips singular perturbations to the Markov chains

3.2.1. Dynamics bring to the fore

If the two-weighting-scale property exists, the fundamental equation (2.1) can be write :

$$\left[\mathcal{P}(k+1)\ \ \mathcal{R}(k+1)\right] = \left[\mathcal{P}(k)\ \ \mathcal{R}(k)\right] \cdot \begin{pmatrix} \mathcal{T}_{11} & \mathcal{T}_{12} \\ \mathcal{T}_{21} & \mathcal{T}_{22} \end{pmatrix} \qquad (3.3)$$

where $\mathbf{P} = [\mathcal{P}\ \ \mathcal{R}]$ is the decomposition of the initial vector in its strong respectively weak parts r_1 = dimension of the strong part and r_2 = dimension of the weak one ($r = r_1 + r_2$). The matrices dimensions are so : $\mathcal{T}_{11}$ $(r_1 \times r_1)$, $\mathcal{T}_{12}$ $(r_1 \times r_2)$, $\mathcal{T}_{21}$ $(r_2 \times r_1)$, $\mathcal{T}_{22}$ $(r_2 \times r_2)$.

3.2.2. The adapted singular perturbed form

The Phillips singular perturbed modeling [5] adapted to the fundamental equation gives :

$$\left[\mathcal{P}(k+1)\ \ \mathcal{R}(k+1)\right] = \left[\mathcal{P}(k)\ \ \mathcal{R}(k)\right] \cdot \begin{pmatrix} \mathcal{T}_{11} & \mu^{j}\,\mathcal{T}^*_{12} \\ \mu^{1-j}\,\mathcal{T}^*_{21} & \mu\,\mathcal{T}^*_{22} \end{pmatrix} \qquad (3.4)$$

with : $\mathcal{T}^*_{12} = \mathcal{T}_{12} / \mu^j$, $\mathcal{T}^*_{21} = \mathcal{T}_{21} / \mu^{1-j}$, and $\mathcal{T}^*_{22} = \mathcal{T}_{22} / \mu$, $\mu \in]\,0,1]$, $j \in [0,1]$.

3.2.3. Dynamics decoupling

In the case of the stochastical systems, the j parameter takes the maximal value ($j = 1$), because of the disproportion of the $\mathcal{T}_{12}$ and $\mathcal{T}_{22}$ submatrices raporting to $\mathcal{T}_{11}$ and $\mathcal{T}_{21}$ ones. The *decoupled form of the fundamental equation* will be:

$$\left[\mathcal{P}_s(k+1)\ \ \mathcal{R}_w(k+1)\right] = \left[\mathcal{P}_s(k)\ \ \mathcal{R}_w(k)\right] \cdot \begin{pmatrix} \mathcal{T}_s & 0 \\ 0 & \mathcal{T}_w \end{pmatrix} \qquad (3.5)$$

$$\mathcal{T}_s = \mathcal{T}_{11} + \mathcal{T}_{11}^{-1}\mathcal{T}_{12}\mathcal{T}_{21}\ , \qquad \mathcal{T}_w = \mathcal{T}_{22} - \mathcal{T}_{21}\mathcal{T}_{11}^{-1}\mathcal{T}_{12} \qquad (3.6)$$

with the initial conditions :

$$\mathcal{P}_s(0) = \mathcal{P}(0) \quad \text{and} \quad \mathcal{R}_w(0) = \mathcal{R}(0)$$

The approach of the strong and respectively weak parts is :

$$\mathcal{P}(k) \cong \mathcal{P}_s(k)\ , \quad \mathcal{R}(k) \cong \mathcal{R}_w(k) \qquad (3.7)$$

The resolution of the stochastic system (2.1) is reduced also to the resolution of two subsystems strong and weak, which keep the characteristics of the initial one. On top of that, the strong subsystem is almost stochastic.

3.3. The complete resolution method

Our work concerns the adaptation of the singular perturbation method to the category of ergodic Markov chains presenting the two-time-scale property. For Markov chains, the two-time-scale property will be a property of *two-weighting-scale* of the states in the system evolution. The eigenvectors of the associated stochastical matrix will have the meaning of *system evolution directions*.

In sight of adapting the method of singular perturbation to the Markov chains, we firstly remark that the fundamental formula of a first order chain corresponds to the autonomous form of a discrete system. The methodology of resolution of the Markov chains by the method of singular perturbation assumes firstly the detection of the irreductible classes, and secondly, the decomposition of each final ergodic class presenting the two-weighting-scale property.

Let 2.1 be the form of the fundamental ecuation of such an ergodic subchain.

3.3.1. **Dynamics decoupling** (paragraph 3.2) :

$$\left[\mathcal{P}_s(k+1)\ \ \mathcal{R}_w(k+1)\right]=\left[\mathcal{P}_s(k)\ \ \mathcal{R}_w(k)\right]\cdot\begin{pmatrix}\mathcal{T}_s & 0\\ 0 & \mathcal{T}_w\end{pmatrix} \tag{3.8}$$

3.3.2. **Strong part resolution** :

The $\mathcal{T}_s$ matrix is practically a stochastic matrix, so we obtain a Markov chain corresponding only to the strong part of the initial one. This allowed us to solve this subsystem like a markovian subchain. We apply on it the direct method of resolution (paragraph 2.2). The resolving system is :

$$\begin{cases}\mathcal{P}(\infty)\cdot\mathcal{D}' = 0\\ \mathrm{P}_1(\infty)+\mathrm{P}_2(\infty)+\mathrm{P}_3(\infty)+\ldots+\mathrm{P}_{r_1}(\infty)=1\end{cases} \tag{3.9}$$

with : $\mathcal{D}' = \mathrm{I} - \mathcal{T}_s$ dynamical matrix of the strong part (singular matrix) , I an identity matrix $r_1 \times r_1$,

$$\mathcal{P}(k)=\left[\mathrm{P}_1(k)\ \ \mathrm{P}_2(k)\ \ \ldots\ \ \mathrm{P}_{r_1}(k)\right]$$

The solutions of the system are :

$$\mathrm{P}_1(\infty)=s_1\,,\ \mathrm{P}_2(\infty)=s_2\,,\ \ldots\,,\ \mathrm{P}_{r_1}(\infty)=s_{r_1}; \tag{3.10}$$

3.3.3. **Reinjection in the initial system** :

Reinjecting the strong part solutions in the initial system equations correspondint to the weak part :

$$\mathcal{R}(\infty) = \mathcal{P}(\infty)\cdot\mathcal{T}_{12} + \mathcal{R}(\infty)\cdot\mathcal{T}_{22}, \tag{3.11}$$

we will obtain :

$$\mathcal{R}(\infty)(\mathrm{I}-\mathcal{T}_{22}) = \mathcal{P}(\infty)\cdot\mathcal{T}_{12} \tag{3.12}$$

with :

$$\mathcal{P}(\infty)=\left[\mathrm{P}_1(\infty)\ \ \mathrm{P}_2(\infty)\ \ \ldots\ \ \mathrm{P}_{r_1}(\infty)\right]=\left[s_1\ \ s_2\ \ \ldots\ \ s_{r_1}\right] \quad \text{already calculate,}$$

$$\mathcal{R}(\infty)=\left[\mathrm{P}_{r_1+1}(\infty)\ \ \mathrm{P}_{r_1+2}(\infty)\ \ \ldots\ \ \mathrm{P}_r(\infty)\right] \quad \text{to be calculate.}$$

3.3.4. **Weak part resolution** :

We will note $\mathcal{D}'' = \mathrm{I} - \mathcal{T}_{22}$ the dynamic matrix (nonsingular) of the weak part (I identity matrix $r_2 \times r_2$). The (3.12) system becomes :

$$\mathcal{R}(\infty)\mathcal{D}'' = \mathcal{P}(\infty)\cdot\mathcal{T}_{12} \tag{3.13}$$

We obtain the r_2 equations system with r_2 unknowns :

$$\left[\mathrm{P}_{r_1+1}(\infty)\ \ \mathrm{P}_{r_1+2}(\infty)\ \ \ldots\ \ \mathrm{P}_r(\infty)\right]=\left[s_1\ \ s_2\ \ \ldots\ \ s_{r_1}\right]\mathcal{T}_{12}\,\mathcal{D}'^{-1} \tag{3.14}$$

The solutions are :

$$\mathrm{P}_{r_1+1}(\infty)=w_1,\ \mathrm{P}_{r_1+2}(\infty)=w_2,\ \ldots,\ \mathrm{P}_r(\infty)=w_{r_2} \tag{3.15}$$

The whole system approaching solutions are so :

$$s_1,\ s_2,\ \ldots,\ s_{r1} \text{ and } w_1,\ w_2,\ \ldots,\ w_{r2} \qquad (3.16)$$

For demonstrating the applicability of our method, we give in following a suggestive example.

4. EXAMPLE

Let $\mathcal{T}$ be the associate stochastic matrix of a five states Markov irreductible ergodic chain :

$$\mathcal{T} = \begin{pmatrix} 0{,}99 & 0{,}005 & 0{,}003 & 0 & 0{,}002 \\ 0{,}005 & 0{,}98 & 0{,}015 & 0 & 0 \\ 0 & 0{,}02 & 0{,}97 & 0 & 0{,}01 \\ 0{,}88 & 0 & 0 & 0{,}005 & 0{,}115 \\ 0 & 0 & 0{,}997 & 0{,}001 & 0{,}002 \end{pmatrix}$$

a) The study of the two-weighting-scale :

The eigenvalues of the stochastic matrix are :

$$\lambda_1 = 1\ ;\ \lambda_2 = 0{,}98\ ;\ \lambda_3 = 0{,}96\ ; \quad \lambda_4 = -\,0{,}014 \text{ and } \lambda_5 = 0{,}010.$$

These eigenvalues can be separate in two groupes, the perturbational coefficient well be then : $\mu = |\lambda_4| / |\lambda_3| = 0{,}01 << 1$, so the two-weighting-scale property exists. The strong part contains 3 states and the weak one, 2 states.

b) Dynamics decoupling :

$$\left[\mathcal{P}(k+1) \quad \mathcal{R}(k+1)\right] = \left[\mathcal{P}(k) \quad \mathcal{R}(k)\right] \begin{pmatrix} 0{,}99 & 0{,}005 & 0{,}005 & 0 & 0 \\ 0{,}005 & 0{,}98 & 0{,}015 & 0 & 0 \\ 0 & 0{,}019 & 0{,}98 & 0 & 0 \\ 0 & 0 & 0 & 0{,}005 & 0{,}1121 \\ 0 & 0 & 0 & 0{,}001 & -0{,}0082 \end{pmatrix}$$

with : $\mathcal{T}_s = \begin{pmatrix} 0{,}99 & 0{,}005 & 0{,}005 \\ 0{,}005 & 0{,}98 & 0{,}015 \\ 0 & 0{,}019 & 0{,}98 \end{pmatrix}$ and $\mathcal{T}_w = \begin{pmatrix} 0{,}005 & 0{,}1121 \\ 0{,}001 & -0{,}0082 \end{pmatrix}$

c) Resolution at the limit :

The whole system probability distribution approach is :

$$P(\infty) = \left[\mathcal{P}(\infty)\ \mathcal{R}(\infty)\right] = \left[P_1(\infty)\ \ P_2(\infty)\ \ P_3(\infty)\ \ P_4(\infty)\ \ P_5(\infty)\right]$$

$$P(\infty) = \left[0{,}2100 \quad 0{,}4191 \quad 0{,}3709 \quad 0{,}0000041 \quad 0{,}0041\right]$$

So the real distribution : $P_{\text{real}}(\infty) = \left[0{,}2100 \quad 0{,}4192 \quad 0{,}3667 \quad 0{,}0000041 \quad 0{,}00409\right]$ is verry well approached.

The average precision of the approach is about 0,64% .

5. CONCLUSIONS

Our method gives satisfying approaches of the real solutions, obtained by application of the direct method to the initial matrix $\mathcal{T}$. It presentes the advantage of a work simplicity (systems with little dimensions), and of a identification of the strong and weak parts of the Markov chain, so of the real corresponding system.

Usually, for a system designer, the weak part of the Markov chain must contain all the states (events) of the system, wich are to avoid in its evolution. In this condition and from this point of view, the system will be well designed.

6. REFERENCES

[1] Boukhlal, R., Etude comparative en discret, de différentes modélisations sous formes singulièrement perturbées. Application à la commande optimale, Thèse 3-th cycle en automatique, Casablanca, 1991.

[2] Chretienne, P., Faure, R., Processus stochastiques leurs graphes, leurs usages, Gauthier-Villars, Paris, vol. 2, 1974.

[3] El Moudni, A., Contribution à la modélisation et à l'analyse des systèmes discrets à échelle de temps multiples. Application à la commande optimale, Thèse ès sciences physiques, Lille, 1985.

[4] Norman, T., J., Bailey, The elements of Stochastic Processes , Wiley, New York, 1970.

[5] Phillips, R., G., Reduced order modelling and control of two-time scale discrete systems, Int. J. Contr., nr 31 (1980), pp. 765.

A STOCHASTIC APPROACH TO MODELLING DYNAMICAL SYSTEMS IN ECOLOGY

Reinhart Funke

Fraunhofer-Institute for Information and Data Processing,
Branch Lab for Process Optimisation
Kurstr. 33, D-10117 Berlin, Germany

Abstract. A stochastic approach to modelling dynamical systems by stochastic differential equations (SDEs) in *Itos* sense is shown. The modelling idea is the following: greater systems are being built using basic deterministic or stochastic modules where the structure design principle of *Peschel* and *Mende* [8] is used. "Elementary" stochastic processes such as *Ornstein-Uhlenbeck* processes or processes with hyper-gamma (compound *Poisson*) distribution just as small systems generating some desired output can be used as basic modules.

1. INTRODUCTION

Stochastic processes as solutions of SDEs can be mapped into a simulation model in several ways. Calculating the n-dimensional distribution density is theoretically always the best but very expensive, sometimes even practically impossible. Other ways are simulating pathwise solutions of SDEs using stochastic approximation schemes or calculating (deterministic) scenaria for the system on the basis of calculated dynamical confidence intervals by conventional uumerical approximation methods. A comparison of these last to ways shows both methods answer different questions. Stochastic approximation shows the more probable behaviour of the system under stochastic but "normal" conditions. The behaviour of the system under "extremal" conditions (the occurence of relatively improbable sample paths of the input processes) can be discovered by this method only after a very high number of simulated paths. The second method is more suitable for these purposes. Two examples of aquatic ecosystems illustrate this.

2. THE STRUCTURE DESIGN PRINCIPLE FOR NONLINEAR PROCESSES

The action of the structure design principle shall be shown with an simple example. Let a dynamical system of dimension n with values in $\mathbf{R}_n^+$ be given in form of n time series or, for simplicity, analytically in $\mathbf{R}_1^+$. All steps provided analytically in this case can be carried out numerically so that an analytically given one-dimensional function is not limiting the generality but the idea can be explained more clearly. Given, e.g., the function

$$x(t) = C\left(\frac{1}{2} + \frac{1}{\pi}\operatorname{arctg}(\lambda t + \mu)\right)$$

find an autonomous differential equation system having the given function x(t) as one of its components. Using differential operators $F_1 = \frac{d\ln}{dt}$ and $F_k x = \frac{1}{1-k}\frac{dx^{1-k}}{dt}$, with k integer, $k \geq 0$, $k \neq 1$, we find the following deterministic differential system (1) suitable for stochastic modelling using SDEs:

$$\text{(1)}\qquad \begin{aligned} \dot{x}_1 &= -\frac{2\pi\lambda}{C}, & x_1(0) &= -\frac{2\pi\mu}{C} \\ \dot{x}_2 &= x_1\, x_2^2, & x_2(0) &= \frac{C\lambda}{\pi\left(1+\mu^2\right)} \\ \dot{x}_3 &= x_2\,, & x_3(0) &= C\left(\frac{1}{2}+\frac{1}{\pi}\operatorname{arctg}\mu\right) \end{aligned}$$

with $x_3(t) = x(t)$ the given function. Changing λ and the initial conditions into inconsistent ones (such initial values for which real C, λ and μ satisfying the right-hand equations of system (1) do not exist) one gets structurally new solutions of the system (1). The system (2), e.g.,

$$\text{(2)}\qquad \begin{aligned} \dot{x}_1 &= 0, & x_1(0) &= a \neq 0 \\ \dot{x}_2 &= x_1\, x_2^2, & x_2(0) &= b \neq 0 \\ \dot{x}_3 &= x_2\,, & x_3(0) &= c, \end{aligned}$$

yields the following solution (with bounded definition domain incase ab > 0):

$$\begin{aligned} x_1(t) &= a = \text{const.}, \\ x_2(t) &= \frac{b}{1-abt}, \\ x_3(t) &= -\frac{1}{a}\ln(1-abt)+c. \end{aligned}$$

So the differential system (1) represents a greater class of functions then the (analytically) given function x(t) depending on some parameters does. Consequently, differential systems such as (1) divide the process resulting in the output of some real system into simpler steps each described by a simpler differential equation what allows a greater variability of modelling the underlying real system.

3. THE STOCHASTIC MODELLING METHOD

We use SDEs in *Itos* sense for our modelling purposes of stochastic real systems. We want to use such *Markov* processes as basic modules which transition probability is known, because there exist unique maps of dynamical confidence intervals for monotonous transformations of the state variables, but not for the moments. There is only a small class of solutions of one-dimensional SDEs which distribution is known: *Wiener* process, forming the basic process in the underlying stochastic calculus, *Ornstein-Uhlenbeck* process, linear processes with lognormal distribution, *Feller* process and more general processes with Gamma or Hyper-Gamma distribution.
Let $\{\Omega, \mathfrak{F}, \mathbf{P}\}$ be a complete probability space, $\mathfrak{F}_t \subseteq \mathfrak{F}$ a family of increasing σ-algebras, and w(t) a Standard Wiener process, adopted to the increasing family of σ-algebras $\mathfrak{F}_t$. The following SDEs define the named above processes, where the coefficients α, α_i, β may be time dependent.

3.1. Ornstein-Uhlenbeck process

The solution $\xi(t)$ of the following SDE (3) with coefficient functions α and $\beta > 0$

$$\text{(3)}\qquad d\,\xi(t) = -\,\alpha(t)\,\xi(t)\,dt + \beta(t)\,dw(t),\quad \xi(0) = x_0,$$

is called Ornstein-Uhlenbeck process. Its transition probability is normal Gaussian with expectation

$$\mathbf{E}\,\xi(t) = x_0\, e^{\int_0^t -\alpha(u)du} \quad \text{and dispersion} \quad \mathbf{D}\,\xi(t) = \int_0^t \beta^2(s) e^{-2\int_s^t \alpha(u)du}\, ds.$$

3.2. Linear processes with lognormal distribution

The solution of the SDE (4)

(4) $$d\,\xi(t) = \alpha(t)\,\xi(t)\,dt + \beta(t)\,\xi(t)\,dw(t),\ \ \xi(0) = x_0,\ x_0 > 0,$$

can be written in the form

$$\xi(t) = x_0 \exp\left\{\int_0^t\left(\alpha(s) - \frac{\beta^2(s)}{2}\right)ds + \int_0^t \beta(s)dw(s)\right\}.$$

Its transition probability is lognormal. Expectation and dispersion are given by

$$\mathbf{E}\,\xi(t) = x_0\,e^{\int_0^t \alpha(u)du} \quad\text{and}\quad \mathbf{D}\,\xi(t) = x_0^2 e^{\int_0^t 2\alpha(u)du}\left[e^{\int_0^t \beta^2(u)du} - 1\right].$$

3.3. Processes with (Hyper-) Gamma distribution

Consider the SDE (5)

(5) $$d\,\xi(t) = [(1+\mu)\,\alpha_0(t) + \alpha_1(t)\,\xi(t)]\,dt + \sqrt{2\alpha_0(t)\xi(t)}\ dw(t),\ \ \xi(0) = x_0,\ x_0 \geq 0.$$

If there are positiv for all t > 0 solutions of the initial value problem (IVP)

$$f'(t) = \alpha_0(t) + \alpha_1(t)\,f(t), \qquad f(0) = 0,$$
$$g'(t) = \alpha_1(t)\,g(t), \qquad g(0) = 1,$$

then the process $\xi(t)$ possesses Hyper-Gamma distribution (or compound Poisson distribution with imbedded exponential distribution) with characteristic function

$$\varphi(t,u) = \mathbf{E}\,e^{iu\xi(t)} = \left(\frac{1}{1-iuf(t)}\right)^{1+\mu}\exp\left\{\frac{iux_0 g(t)}{1-iuf(t)}\right\}.$$

Expectation and dispersion are defined by $\mathbf{E}\,\xi(t) = (1+\mu)\,f(t) + x_0\,g(t)$ and $\mathbf{D}\,\xi(t) = (1+\mu)\,f^2(t) + 2\,x_0\,f(t)\,g(t)$. If $\alpha_1(t) < 0$, then g(t) and therefore the dependence on x_0 vanishes asymptotically for t going to infinity meaning that $\xi(t)$ becomes asymptotically Gamma distributed.

Consider now the SDE (6)

(6) $$d\,\xi(t) = [(1+\mu)\,\alpha_0(t) + \alpha_1(t)\,\xi(t) + \alpha_2(t)\,\xi^2(t)]\,dt + \sqrt{2\left[\alpha_0(t)\xi(t) - \alpha_2(t)f(t)\xi^2(t)\right]}\ dw(t),\ \ \xi(0) = 0,$$

with $\alpha_0 \geq 0$ and $\alpha_2 \leq 0$. If the function f(t) is positiv for all t > 0 and solves the IVP

$$f'(t) = \alpha_0(t) + \alpha_1(t)\,f(t) + (2+\mu)\,\alpha_2(t)\,f^2(t)\,, \quad f(0) = 0,$$

then the process $\xi(t)$ possesses Gamma distribution with the characteristic function

$$\varphi(t,u) = \mathbf{E}\,e^{iu\xi(t)} = \left(\frac{1}{1-iuf(t)}\right)^{1+\mu},$$

expectation $\mathbf{E}\,\xi(t) = (1+\mu)\,f(t)$ and dispersion $\mathbf{D}\,\xi(t) = (1+\mu)\,f^2(t)$. In both cases μ is a parameter of stochasticity, where μ going to infinity means deterministic motion and $\mu = 0$ means maximal stochastic disturbances under which the process $\xi(t)$ remains regular in $\mathbf{R}_1^+$.

3.4. Stochastic modelling of system (1)

There are many possibilities to use these processes for stochastic modelling of system (1), e.g.: In order to model x_1 stochastically it can be substituted by the contrary of the solution of equ. (5) with α_0 = const > 0 and $\alpha_1 \equiv 0$, or the Ornstein-Uhlenbeck process to be shifted by the constant $-\dfrac{2\pi\lambda}{C}$ can be used instead of this constant itself (the dimension of the system is then 4); in order to model x_2 it can be replaced by the process $\eta(t) = \frac{1}{\xi(t)}$ with $\xi(t)$ solving equ. (5) under the assumptions $\alpha_0(t) = \frac{1}{1-\mu} x_1(t) > 0$, $\mu > 1$, and $\alpha_1 \equiv 0$; the third differential equation of system (1) can be replaced by the following SDE (7)

$$(7) \qquad d\,\xi_3(t) = x_2(t)\,dt + \sqrt{\frac{2}{1+\tilde{\mu}} x_2(t)\xi_3(t)}\;dw(t), \qquad \xi_3(0) = C\left(\frac{1}{2} + \frac{1}{\pi}\operatorname{arctg}\mu\right)$$

where here the parameter of stochasticity is $\tilde{\mu}$.

4. APPLICATIONS

Stochastic environmental influences like global radiation, water temperature and water flow rate can appropriately be modeled using stochastic processes which solve equ. (3) or (5).

4.1. Global radiation and water temperature

Let $\alpha_0(t) = a_{11} + a_{12}\cos\omega t + a_{13}\sin\omega t$ with $\omega = 2\pi/T$ and $T = 365.25$ [days] be a positiv periodical function and $\alpha_1(t) = \alpha_1 = \text{const} < 0$. Then the global radiation $\xi(t)$ [kJ/cm²] solves equ.(5) with $\mu \approx 2.5$. The water temperature $\Theta(t)$ can then be modeled by the following stochastically driven deterministic differential equation (8):

$$(8) \qquad d\,\Theta(t) = [a_{21}\,\xi(t) + a_{22}\,\Theta(t) + a_{23}]\,dt, \quad \Theta(0) = \Theta_0.$$

4.2. Water flow rate

Let $\alpha_3(t) = a_{31} + a_{32}\cos\omega t + a_{33}\sin\omega t$ (with the same ω and T as in 4.1.) be again a positiv periodical function, and α, β be positiv constants. Then for stochastical modelling the water flow rate $\Phi(t)$ [m³/s] of a surface water current a shifted Ornstein-Uhlenbeck process (equ. (3)) in the form (9) is suitable under certain circumstances:

$$(9) \qquad d\,\Phi(t) = -\,\alpha\,[\,\Phi(t) - \alpha_3(t)\,]\,dt + \beta\,dw(t), \; \Phi(0) = \Phi_0.$$

It has to be said that the water flow is existencially depending on the precipitation which, strictly speaking, have firstly to be forecasted in order to model suitably the water flow of a river or a lake, what is much more difficult. Only under "normal" conditions, i.e. more or less regular precipitation depending only on the season, it is possible to reflect this real process by the proposed SDE (9).

4.3. Water quality models

The given above equations have been used as stochastic driving forces in two water quality models one of them describing the nitrogen, phosphorus and silicon dynamics together with chlorophyta, cyanophyta, diatomea, and detritus in a lake (see [3],[5]), and the other describing the nitrogen cycle in the Elbe river (see [4],[5],[6]).

5. REFERENCES

[1] Braun, P.: Das Eutrophierungsmodell ERNA, Dissertation A, TU Dresden, 1984.

[2] Funke, R.: A Nonlinear Diffusion with Hyper - Gamma Distribution. SAA 7(1989)1, 19-33.

[3] Funke, R.: Vorhersage der Wasserqualität am Zu- und Abfluß von Seen. Inf.-Fachb. Vol. 275, 230-236.

[4] Funke, R.: Prediction of Water Quality of the Elbe River on the Basis of Stochastic Processes. SAMS **8**(1991)6, 443-447.

[5] Funke, R.: Stochastische Modellierung Dynamischer Systeme in der Ökologie, Dissertation, Universität Osnabrück, 1992.

[6] Funke, R.: Markovian Diffusions as Stochastic Driving Forces in Hydroecology, Colloquy on Statist. & Math. Modelling in the fields of Agriculture, Food Technology & Environment, Berlin, 1993, Oct., 11.-15.

[7] Gard, T.C.: Introduction to Stochastic Differential Equations. Marcel Dekker, New York, 1988.

[8] Peschel, M., W. Mende: The Predator-Prey Model: Do We Live in a Volterra World ? Akademie Verlag Berlin and Springer Verlag Vienna, 1986.

[9] Rudolph, P., Sommerfeld, V.: Parameterestimation in Water Quality Models. SAMS **8**(1991)6, 421-427.

Optimal Trajectory Planning for Robots under the Consideration of Stochastic Parameters and Disturbances
- Computation of an Efficient Open-Loop Strategy -

K. Marti and S. Qu
Institute of Mathematics and Computer Science
Federal Armed Forces University Munich
85577 Neubiberg, Germany

Abstract. Efficient control strategies of robots should cause only low on-line correction expenses. Hence, the mostly available statistical and a priori informations about the random parameters and disturbances of the underlying mechanical system and its environment should be considered already for off-line programming of robots. Measuring the violations of the basic mechanical conditions by means of expected penalty costs, a stochastic optimization problem is obtained for the computation of an optimal open-loop control. The stochastic optimization problem can be solved - after discretization - by parameter optimization.

1. Introduction

In this paper we discuss the trajectory planning problem for a given collision-free geometric path in the work space [3], $x = x_e(s),\ 0 \le s \le s_e$, where s is a path parameter. From the kinematic equation

$$\mathbf{T}(\mathbf{q}, \mathbf{p}) = \mathbf{x}, \tag{1}$$

where $\mathbf{x}$ is a vector describing the position und orientation of the end-effector of the robot, $\mathbf{q}$ is a vector denoting the configuration coordinates of the robot, and $\mathbf{p}$ is a vector of model parameters, the configuration variable $\mathbf{q}$ can be also described as a function of s, i.e. $\mathbf{q} = \mathbf{q}(s, \mathbf{p}), 0 \le s \le s_e$.

A robot is driven by the torques and forces generated by the motors of the robot. The torques and forces have to be calculated such that the end-effector of the robot run through the given path in work space. For this we need the dynamic equation

$$\sum_{j=1}^{n} J_{ij}\ddot{q}_j + \sum_{j,k=1}^{n} D_{ijk}\dot{q}_j\dot{q}_k + G_i = \tau_i,\ i = 1, 2, ...n, \tag{2}$$

where J_{ij} D_{ijk} and G_i are the elements of the inertia matrix, the coefficients of the centrifugal and Coriolis forces and the gravity forces, respectively, which are functions of $\mathbf{p}$ and $\mathbf{q}$, and τ_i are the torques and forces generated by the motors.

Describing the path parameter s as a function of time t, $s = s(t)$, equation (2) yields

$$a_i(s, \mathbf{q}, \mathbf{q}', \mathbf{p})v' + b_i(s, \mathbf{q}, \mathbf{q}', \mathbf{q}'', \mathbf{p})v + c_i(s, \mathbf{q}, \mathbf{p}) = \tau_i,\ \ i = 1, 2, ...n, \tag{3}$$

where $v = \dot{s}^2$,

$$a_i(s, \mathbf{q}, \mathbf{q}', \mathbf{p}) = \frac{1}{2}\sum_{j=1}^{n} J_{ij}(\mathbf{q}, \mathbf{p})q_j{}', \quad i = 1, 2, ...n, \tag{4}$$

$$b_i(s, \mathbf{q}, \mathbf{q}', \mathbf{q}'', \mathbf{p}) = \sum_{j=1}^{n} J_{ij}(\mathbf{q}, \mathbf{p})q_j'' + \sum_{j,k=1}^{n} D_{ijk}(\mathbf{q}, \mathbf{p})q_j'q_k', \quad i = 1, 2, ...n, \tag{5}$$

$$c_i(s, \mathbf{q}, \mathbf{p}) = G_i(\mathbf{q}, \mathbf{p}), \quad i = 1, 2, ...n, \tag{6}$$

and ′ means the derivative with respect to s.

The problem of optimal trajectory planning can be then defined mathematically as follows:

$$\underset{v, q}{min} \int_0^{s_e} f_0(s, q, q', q'', v, v')ds \tag{7}$$

$$\text{subject to} \qquad v(0) = 0 \quad , \quad v(s_e) = 0$$

$$\tau_{min,i} \leq a_i v' + b_i v + c_i \leq \tau_{max,i}, \quad i = 1, 2, ...n \tag{8}$$

$$0 \leq v \leq v_{max}(s, \mathbf{p}) \tag{9}$$

$$\mathbf{T}(\mathbf{q}(\mathbf{s}), \mathbf{p}) = \mathbf{x}(\mathbf{s}). \tag{10}$$

This problem was solved in [2]. Especially for the time optimal problem there is a so-called **Phase-Plan-Method** [2]. Having got v and $\mathbf{q}$, we can calculate τ_i as follows

$$\tau_i = \tau_i(v(s), q(s), p) = a_i v' + b_i v + c_i, \quad i = 1, 2, ...n. \tag{11}$$

2. Trajectory planning under uncertainty

The coeficients a_i, b_i, c_i in (8) and v_{max} in (9) depend on the model parameter $\mathbf{p}$. The solution $\mathbf{q} = \mathbf{q}(\mathbf{s}, \mathbf{p})$ of the inverse kinematic problem (10) is also dependent on $\mathbf{p}$. Due to the very often existing uncertainty at model parameters $\mathbf{p}$, we must reformulate these constraints. In [1] some substitute problems were proposed, so that the stochastic variation of the parameter $\mathbf{p}$ is taken into account **within** the optimal trajctory planning process. In this paper a new substitute problem is presented.

Let $\bar{\mathbf{p}}$ be the expectation of $\mathbf{p}$. For any given $\bar{v} = \bar{v}(s)$, $\bar{\mathbf{q}} = \bar{\mathbf{q}}(s)$ and setting $\mathbf{p} := \bar{\mathbf{p}}$ in (11), we get $\bar{\tau}_i(s) := \tau_i(\bar{v}, \bar{\mathbf{q}}, \bar{\mathbf{p}})$. From $\dot{s}^2 = \bar{v}(s)$ we have $s = \bar{s}(t)$ or $t = \bar{t}(s)$. Let $\mathbf{q} = \mathbf{q}(t|\mathbf{p}, \bar{\tau}_i)$ be the solution of

$$\sum_{j=1}^{n} J_{ij}\ddot{q}_j + \sum_{j,k=1}^{n} D_{ijk}\dot{q}_j\dot{q}_k + G_i = \bar{\tau}_i(\bar{s}(t)), \; i = 1, 2, ...n, \tag{12}$$

with initial values $\mathbf{q}(0) = \bar{\mathbf{q}}(0)$ and $\dot{\mathbf{q}}(0) = 0$. For $\mathbf{p} = \bar{\mathbf{p}}$ we have $\mathbf{q}(t) = \bar{\mathbf{q}}(\bar{s}(t))$. If $\mathbf{p} \neq \bar{\mathbf{p}}$, then $\mathbf{q} = \mathbf{q}(t)$ deviates from the path $\mathbf{q} = \bar{\mathbf{q}}(\bar{s}(t))$ in the configuration space. Differentiating (12) with respect to $\mathbf{p}$, we find

$$\sum_{j=1}^{n}(A_{ij}^{(1)}\ddot{q}_j{}^{(k)} + B_{ij}^{(1)}\dot{q}_j{}^{(k)} + C_{ij}^{(1)}q_j^{(k)}) + D_i^{(k)} = 0, \tag{13}$$

where

$$q_j^{(k)} = \frac{\partial^k q_j}{\partial p^k}, \quad \dot{q}_j^{(k)} = \frac{d}{dt}\frac{\partial^k q_j}{\partial p^k}, \quad \ddot{q}_j^{(k)} = \frac{d^2}{dt^2}\frac{\partial^k q_j}{\partial p^k}, \tag{14}$$

$$A_{ij}^{(1)} = J_{ij}, \; B_{ij}^{(1)} = \sum_{k=1}^{n}(D_{ijk} + D_{ikj})\dot{q}_k, \; C_{ij}^{(1)} = \sum_{k=1}^{n}(\frac{\partial J_{ik}}{\partial q_j}\ddot{q}_k + \sum_{\eta=1}^{n}\frac{\partial D_{i\eta k}}{\partial q_j}\dot{q}_\eta)\dot{q}_k + \frac{\partial G_i}{\partial q_j} \tag{15}$$

and $D_i^{(k)}$ are certain functions of $A_i^{(1)}$, $B_i^{(1)}$, $C_i^{(1)}$ and $\mathbf{q}$, $\mathbf{q}^{(j)}$, $j = 1,2,...k-1$. Solving (13) with $\mathbf{p} := \bar{\mathbf{p}}$ and the initial values $\mathbf{q}^{(k)}(0) = 0$, $\dot{\mathbf{q}}^{(k)}(0) = 0$, we can determine the derivatives $\frac{\partial^k \mathbf{q}}{\partial p^k}(t, \bar{\mathbf{p}})$ for all $k = 1,2,3,...K$.

Considering again the trajectory planning problem, the violations of the constraints (8) - (10) are evaluated numerically, as in [1], with the help of some chosen penalty functions. Because parameter $\mathbf{p}$ in (8) -(10) is stochastic, we consider the expected penalty costs. It is demanded then, that some prescribed upper bounds $\delta_{u,i}$ $i = 1,2,...n$, $\delta_{u,0}$ and δ_T should not be exceeded. This yields the following substitute problem

$$\min_{\bar{v},\ \bar{\mathbf{q}}} \int_0^{s_e} f_0(s, \bar{v}, \bar{\mathbf{q}})ds \tag{16}$$

subject to $\bar{v}(0) = 0 \quad , \quad \bar{v}(s_e) = 0$

$$E_{\mathbf{p}}u_i(\bar{\tau}_i(\bar{v}, \bar{\mathbf{q}}, \mathbf{p}), \tau_{min,i}, \tau_{max,i}) \leq \delta_{u,i}, \; i = 1,2,...n \tag{17}$$

$$E_{\mathbf{p}}u_0(\bar{v}, v_{max}(s, \mathbf{p}) \leq \delta_{u0} \tag{18}$$

$$E_{\mathbf{p}}u_T(\mathbf{T}(\mathbf{q}, \mathbf{p}) - \mathbf{x_e}) \leq \delta_{\mathbf{T}}, \tag{19}$$

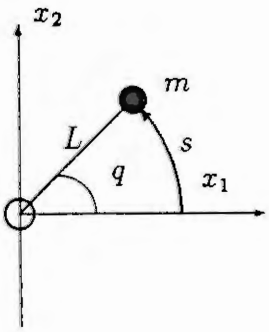
where $\mathbf{q}$ in (19) is the solution of (12) with initial values $\mathbf{q}(0) = \bar{\mathbf{q}}(0)$ and $\dot{\mathbf{q}}(0) = 0$. In the problem (16)-(19) we have to calculate $E_p u_i$, $E_p u_0$ and $E_p u_T$. The numerical calculation of the expected values is usually very difficult because u_i, u_0 and u_T can be complicated functions of $\mathbf{p}$. By means of Taylor expansion of the penalty functions the expectations can be determined approximatively. Since $\mathbf{q}$ in (19) depends on $\mathbf{p}$ too, the above derivatives $\bar{\mathbf{q}}^{(k)} = \frac{\partial^k \mathbf{q}}{\partial p^k}(t, \bar{\mathbf{p}})$ are needed to approximate $E_p u_T$.

If $\bar{v}$ and $\bar{\mathbf{q}}$ are discretized, the substitute problem (16)-(19) is reduced to a nonlinear finite-dimensional parameter optimization problem, which can be solved e.g. by mathematical programming methods.

3. Numerical example

In order to demonstrate the above ideas, we consider a rotary massless arm with a point payload in its hand. The kinematic and dynamic equations for this case are given by (20) and (21), where m is the mass of the point payload and L is the length of the arm.

Suppose that $L \equiv 1$ and m is identically distributed in $[1-\Delta m,\ 1+\Delta m]$, $0 \leq \Delta m < 1$. The constraints about the velocities are neglected, while u_i and u_T are now quadratic functions. The robot is demanded to move the payload m from $q_0 = 0$ to $q_e = 1$. Hence, we chose $\bar{q} = \beta_q s$ and $\bar{v} = \beta_v^2 s(1-s)$ with $\beta_q > 0$ and $\beta_v > 0$.



$$q = \frac{s}{L} \tag{20}$$

$$mL^2\ddot{q} = \tau \tag{21}$$

From these equations and (17) and (19) we obtain

$$\frac{3+\Delta m^2}{12}\beta_q^2\beta_v^4 \leq \delta_u, \tag{22}$$

$$\frac{\beta_q^2}{1-\Delta m^2} - \frac{1}{\Delta m} ln \frac{1+\Delta m}{1-\Delta m}\beta_q + 1 \leq \delta_T. \tag{23}$$

Suppose that $\delta_u := (1-\Delta m)^2$ and $\delta_T := 1 - \frac{1-\Delta m^2}{4\Delta m^2} ln^2 \frac{1+\Delta m}{1-\Delta m}$. For the time optimal problem β_v must be as large as possible. Hence,

$$\beta_v^* = \sqrt[4]{\frac{12(1-\Delta m)^2}{(3+\Delta m^2)\beta_q^{*2}}} \tag{24}$$

and

$$\beta_q^* = \frac{1-\Delta m^2}{2\Delta m} ln \frac{1+\Delta m}{1-\Delta m}. \tag{25}$$

The optimal performing time is

$$t_e = \sqrt[4]{\frac{(3+\Delta m^2)\beta_q^{*2}}{12(1-\Delta m)^2}}\pi. \tag{26}$$

In [1] we have chosen $\beta_q = 1$. For $\Delta m \neq 0$ the mean deviation Eu_T of the new substitute problem is smaller than that of the substitute problem in [1].

References

[1] Marti, K., Qu, S.: *Optimal Trajectory Planning for Robot Considering Stochastic Parameters and Disturbances*, Workshop on Stochastic Optimization, Federal Armed Forces University Munich, June 1993

[2] Pfeiffer, F., Johanni, R.: *A Concept for Manipulator Trajectory Planning*, IEEE J.Robot. Automat. RA-3(3), p.115-123, 1987

[3] Singh, S.K., Leu, M.C.: *Manipulator Motion Planning in the Presence of Obstacles and Dynamic Constraints*, Int.J.Robot.Res. 2(10), p.171-187, 1991

ON STOCHASTIC MODELLING OF THE FLUCTUATION OF AIR-POLLUTANT CONCENTRATION

Mikhail Ya. Postan
Odessa Marine Engineering Institute
34,Mechnikov Str.,270029,Odessa,Ukraine

Abstract. *Paper discusses a stochastic model describing emission of particles with random intensity into bounded atmospheric domain D and removal of particles from D away by transport and gravity scavenging. A velocity of transport and scavenging parameters depend on an alternating renewal process. A stationary probabilistic distribution of concentration of particles is investigated.*

A simplest stochastic models for the concentration of pollutants in air or water environment were considered in the works [3-5]. In these models the process of concentration's change was described by an stochastic ordinary first-order differential equation. However, in the mentioned articles the emission intensity is assumed to be a constant and, besides, the removal process did not depends on the transport of particles.

Analysed below is a more general and approximate to reality model in which both the emission intensity of particles into atmosphere and the velocity of their removal beyond of the boundaries of domain $\boldsymbol{D}$ are random variables .

If $\boldsymbol{c(t)}$ denotes the concentration of particles at moment $\boldsymbol{t}$ in domain $\boldsymbol{D}$, then the mathematical model for the variation of $\boldsymbol{c(t)}$ is expressed as

$$\frac{d}{dt}\, c(t) = Q_{Y(t)} - P_{Y(t)} - a_{Y(t)}\, c(t) + \qquad (1)$$

$$+ (P_0 - Q_0)\; I\,(Y(t)=0, c(t)=0), \qquad \text{a. e.}$$

where: $Y(t)$ is an alternating renewal process (or semi-Markov process with two states: 0 and 1); Q_k is emission intensity from a source in the centre of domain D when $Y(t)= k, k =0,1$ (kg/m^{3}h); P_k is the reduction of concentration velocity as a result of the transport of particles blown away from domain D when $Y(t)= k, k=0,1$(kg/m h); $a_k \geqslant 0$ is a coefficient which accounts for the gravity scavenging when $Y(t)= k, k =0,1$(1/h); $I(A) = 1$ if event A occurs, $I(A) = 0$ otherwise.

Note that the equation (1) may also describe the processes of fluctuation of an inventory level in warehouse or an amount of information in the buffer storage [2].

The particular case of the model (1) when $Q_0 = Q_1, P_0 = P_1=0,$ $a_0 = 0,\; a_1 > 0$ was considered in the paper by Grandell [5]. We assume that the following inequalities are held:

$$0 \leqslant Q_0 < P_0 \;,\; Q_1 > P_1 \geqslant 0$$

i.e. $c(t)$ is decreasing when $Y(t) = 0$ and $c(t)$ is increasing when $Y(t) = 1$. The process of random walking $c(t)$ is defined on the closed interval $[0, L]$ where $L =(Q_1 - P_1)/ a_1$, i.e. 0 and L are the detaining boundaries (if $a_1 > 0$).

For investigation of the type (1) model the semi-Markov processes of replenishment and consumption of inventory are very convenient [6].

Let us denote $\{t_n\}, n\geqslant 1,$ the sequence of the moments of time when process $Y(t)$ is changing and let's

$$\Phi_k(x) = \lim_{n\to\infty} Pr\,\{\, c(t_n + 0) \leqslant x,\; Y(t_n + 0) = k\},$$
$$F_k(x) = \lim_{t\to\infty} Pr\,\{\, c(t) \leqslant x,\; Y(t) = k\},$$
$$k =0,1;\; 0 \leqslant x \leqslant L$$

(if the limits' existence is supposed).

$$W_0(t) = \int_{0+}^{t} B_1(t-u)\, dW_1(u) + W_1(0)B_1(t),$$
$$W_1(t) = \int_{t}^{\infty} [1 - B_0(h(u) - h(t))]dW_0(u) + W_0(t), \quad t \geqslant 0, \qquad (2)$$

where $W_k(t) = \Phi_k(L(1 - \exp(-a_1 t)))$, $k=0,1$; $B_k(t)$ - distribution function of sojourn-time of process $Y(t)$ in the state k; $h(t) = (1/a_0)\ \ln [1 + (La_0/P_0)\ (1 - \exp(-a_1 t))]$.

The functions $F_k(x)$ are expressed through the functions $\Phi_k(t)$ by the formulae

$$F_0(L(1- e^{-a_1 t})) = 2b_0/(b_0 + b_1)\{W_0(t) + \int_{t}^{\infty} \{1 - B_0(h(u) - h(t))]dW_0(t)\},$$
$$F_1(L(1 - e^{-a_1 t})) = 2/(b_0 + b_1)\{W_1(0)\int_{0}^{t} (1 - B_1(u))du + \int_{0}^{t}\int_{0}^{t-u} (1 - B_1(y))dydW_1(u)\}, \quad t \geqslant 0, \qquad (3)$$

where
$$b_K = \int_{0}^{\infty} t\ dB_K(t) < \infty, \quad k = 0,1.$$

If $B_0(t) = 1 - \exp(-\lambda t), t \geqslant 0$, then the given below Volterra integral equation of the 2nd kind for determination of $W_1(t)$ follows from (2)

$$[H_0 + La_0(1 - \exp(-a_1 t))]W_1'(t) =$$
$$= \lambda H_1 e^{-a_1 t} \int_{0}^{t} [1 - B_1(t - x)]W_1'(x)dx + \qquad (4)$$
$$+ \lambda H_1 W_1(0)\ e^{-a_1 t}\ [1 - B_1(t)], \quad t \geqslant 0,$$

where $H_0 = P_0 - Q_0$, $H_1 = Q_1 - P_1$.

For applications it is sufficient to know the mean and the variation of concentration. For calculation of $E\, c^i = \lim_{t\to\infty} E\, c^i(t)$, i = 1,2, the following formulae were obtained

$$E\,c = L\,\{\,1 - (a_1/\sigma_1)[(1 + \sigma_2)/(1 + \lambda\, b_1) - F_0(0)] - $$
$$- 2\, a_1 W_1^*(a_1)/(\,1 + \lambda\, b_1)[(\lambda/a_1)(1 - b_1(a_1)) + b_1(a_1)]\},$$

(5)

$$E\,c^2 = 2L\,(Ec) - L^2\,\{\,1 - (2a_1/\sigma_1)[(1 + \sigma_2)/(1 + \lambda\, b_1) - $$
$$- F_0(o)] - 4a_1 W_1^*(a_1)/(1 + \lambda\, b_1)[(\lambda\,/2a_1)(1 - b_1(2a_1)) + $$
$$+ b_1(2a_1)]\},$$

where $\sigma_1 = \lambda\, H_1/H_0;\quad \sigma_2 = a_0 H_1/\,a_1 H_0;$

$$W_1^*(s) = \int_0^\infty exp(-st)\; W_1(t)\;dt;\; b_1(s) = \int_0^\infty exp\,(-st)\;dB_1(t)\;,$$
$$Re\;s \geq 0\;;\; b_1 = -\,b_1'(0).$$

The stationary probability that there are no pollutants inside domain D is equal to

$$F_0(0) = 2(1 + \lambda\, b_1)^{-1}\;(1 + \sigma_2)^{-\lambda/a_0}\;[1/2 + \qquad (6)$$
$$+ a_1 \sum_{n=1}^{\infty} b_1(na_1) W_1^*(na_1)\sigma_2^n/(1 + \sigma_2)^n\; \prod_{i=1}^{n}(i - 1 + \lambda\,/a_0)/i\,].$$

Particularly, if $B_1(t) = 1 - exp(-\mu\, t), t \geq 0$ (the Markov model) formula (5) is reduced to

$$E\;c = L\;\{\;1 - (a_1/\;\sigma_1)[\mu(\;1 + \sigma_2)/(\lambda + \mu\;) - F_0(0)] - $$
$$- 2a_1 W_1^*(a_1)/(\,\mu + \;a_1)\}.$$

As it follows from (4),

$$sW_1^*(s) = W_1(0)\;\{\;1 + \sigma_1 \int_0^\infty exp\;[-(s + \lambda + \mu))t + $$
$$+ (\lambda\,/a_0 - 1)\;ln\;(1 + \sigma_2(1 - e^{-a_1 t}))]dt\;\}\;.$$

The constant $W_1(0)$ may be found from condition

$$\lim_{s \to 0+}\; s\,W_1^*(s) = 1/2.$$

For another particular case when $a_0 = 0,\ a_1 > 0$ the solution of equation (4) is found by the method developed in [1]:

$$sW_1^*(s) = W_1(0)\ [\ 1 + \sum_{n=0}^{\infty} \sigma_1^{n+1} \prod_{i=1}^{n+1} (1 - b_1(s + ia_1))\ /(s + ia_1)],$$

$$Re\ s \geqslant 0.$$

As $a_0 \rightarrow 0$ from (6) we get

$$F_0(0) = 2(1 + \lambda b_1)^{-1} \quad \exp\,(-\ \sigma_1/a_1)\ [\ 1/2 + a_1 \sum_{n=1}^{\infty} (\ \sigma_1/a_1)^n \quad b_1(na_1)\ W_1^*(na_1)/\ (n-1)!\ ].$$

In a general case the solution of equation (4) may be expressed in an explicit form too (in the Laplace-transform terms) and is given by

$$sW_1^*(s) = W_1(0)\ [\ 1 + \sum_{n=0}^{\infty} \Lambda^{n+1} \sum_{i_1=1}^{\infty} \dots \sum_{i_{n+1}=1}^{\infty} \sigma^{(i_1 + \dots + i_{n+1}) - n - 1} \quad x$$

$$x \prod_{k=1}^{n+1} \ (1 - b_1(s + a\cdot(i_1 + \dots + i_k))/\ (s + a_1(i_1 + \dots + i_k))],$$

where $\Lambda = \sigma_1/(1+\sigma_2)$, $\sigma = \sigma_2/\ (1+\sigma_2)$.

The above results have been used for evoluation of the air- pollutant concentration when transshipping some loose bulk cargoes at several Black Sea ports and for development of the recommendations for reducing the intensities Q_K , k= 0,1.

REFERENCES

[1] Postan,M.Ya. Investigation of One-Channel Queueing Systems with Bounded Waiting Time and Priorities. VINITI, No 4122-81, Moscow,1981 (In Russian).

[2] Prabhu,N.U.,Stochastic Storage Processes (Queues, Insurance Risk and Dams). Springer-Verlag, New York-Heidelberg- Berlin, 1980.

[3] Baker,M.B.,Harrison,H.,Vinelli,J. and Ericksson,K.B., Simple Stochastic Models for the Sources and Sinks of Two Aerosol Types. Tellus,31 (1979),39-51.

[4] Rodhe,H. and Grandell,J., On the Removal Time of Aerosol Particles from the Atmosphere by Precipitation Scavenging. Tellus, 24 (1972),442-454.

[5] Grandell,J., Mathematical Models for the Variation of Air-Pollutant Concentrations. Adv. Appl.Prob., V.14, No 2 (1982),240-256.

[6] Postan,M.Ya., Semi-Markov Processes of Replenishment and Consumption of Inventory. Preprint 89-59, V.M. Glushkov Institute of Cybernetics of Ukrainian Academy of Sciences, Kiev,1989 (In Russian).

ON THE UNIFIED SCHEMES OF TRANSSHIPMENT POINTS' WORK MODELLING

Evgeniy N. Voevudskiy,Mikhail Ya.Postan,
Nikolay P. Didorchuk
Odessa Marine Engineering Institute
34, Mechnikov Str.,270029,Odessa,Ukraine

ABSTRACT. Paper discusses some unified schemes for modelling of transport flow's interaction at transshipments. These schemes are based on some new classes of multidimentional stochastic processes including continuous components of evolutional type.

On the bases of analysis and systematization of plenty of real transshipment points (ports, coaling bases, transport junctions) a classifications of stochastic models of these systems has been worked out in the terms of queueing and inventory theories. Corresponding models received the generalized name of Stochastic Queueing Systems Interacting Transport Flows (QSITF), as the main distinctive feature of transshipments points' functioning is availability of adjacent types of transport interaction (for instance, marine with river-going or railroad) [1] .

There were worked out and investigated mathematical models of some most widely spread types of QSITF with non-homogeneous cargo, particularly of transport-warehouse systems

(TWS), in which interaction of transport units' flow with continuous type of transport usually occurs, as well as of the systems with interaction between loaded and unloaded transport units. Dependence of densities of transport flows and rates of productivity of loading and unloading devices (mechanisms) on current state of the system has been considered (i.e. availability of "feedback" is taken into consideration) [2] . For multichannel TWS with homogeneous cargo a simple recurrent algorithm for calculation of stationary joint distribution of amount of cargo available at warehouses and transport units queue's length has been worked out. For the remaining models there is being discussed the problem of the analytical investigation.

Apparatus of generalized line-class of Markov processes is used for modelling of QSITF. These stochastic processes are multi-dimentional ones and contain discrete components (for description of fluctuaion of transport queue's length), as well as continuous components (for description of evolution of quantity of non-completed by a certain moment of time loading and unloading operations of some transport units, cargo's quantity at warehouses). Conditions of ergodicity of these processes have been found out as well as their physical meaning as measures of reserve of loading-unloading fronts' through-put.

Consider,for instance, one- channel TWS in which there arrive m independent Poisson flows of transport units with cargo for the aim of their unloading. Each transport unit of i-th flow is loaded with i-tn kind of cargo only and its capacity is a random variable with distribution function $G_i(x)$, $i = 1,2,\dots,m$. Let us introduce the following notations:

$\nu(t)$ - whole number of transport units with cargo in the TWS at moment t ;

$\varkappa(t)$ - a random component describing kind of cargo at the transport unit which occupies the channel of unloading at moment t (if $\nu(t) > 0$), $\varkappa(t) = 1,2,\dots,m$;

$\zeta(t)$ - a quantity of cargo which is yet not unloaded from a transport unit which occupies the channel at moment t (in case $\nu(t) = 0$ the components $\varkappa(t)$ and $\zeta(t)$ are not being defined);

$\xi_i(t)$, $i = 1,2,\dots,m$, - a quantity of i-th kind of cargo at warehouse at moment t;

$w_i(l)$, $i = 1,2,\dots,m$; $l = 1,2,\dots$, - rate of i-th kind of cargo unloaded from a transport unit into a warehouse when $\varkappa(t) = i$, $\nu(t) = l$;

$u_i(l)$, $i = 1,2,\dots,m$; $l = 1,2,\dots$, -a velocity of i-th kind of cargo removal from warehouse when $\varkappa(t) = i$, $\nu(t)=l$.

The unloaded cargo from a transport unit comes into the warehouse immediately. The service discipline at the channel of unloading is FIFO.

For TWS described we introduce multi-dimentional Markov process:

$$\sigma(t) = (\nu(t), \varkappa(t), \zeta(t), \xi_1(t), \dots, \xi_m(t)).$$

For determination of stationary distribution functions

$$F_0(y_1,\dots,y_m) = \lim_{t\to\infty} \Pr\{\nu(t)=0, \xi_1(t) \leqslant y_1, \dots, \xi_m(t) \leqslant y_m\}$$

$$F_i(l,x,y_1,\dots,y_m) = \lim_{t\to\infty} \Pr\{\nu(t)=l;\ \varkappa(t) = i,\ \zeta(t) \leqslant x,\ \xi_1(t) \leqslant y_1,\dots,\ \xi_m(t) \leqslant y_m\} ,$$

$x, y_1, \dots, y_m \geqslant 0$; $l \geqslant 1$, $i = 1, 2, \dots, m$, a set of differential equations in partial derivatives and corresponding boundary conditions are deduced. A recurrent algorithm for solution of this boundery-value problem is proposed.

The conditions of ergodicity of process $\sigma(t)$ are (if $K^+ = \{(l,i):\ w_i(l) > u_i(l)\} \neq \emptyset$)

$$\sum_{l=1}^{\infty} F_i(l)\left[w_i(l) - u_i(l)\right] < F_0\, u_i(0), i=1,2,\dots,m,$$

where $F_i(l) = F_i(l;\ \underbrace{\infty, \infty, \dots, \infty}_{m})$,

$$F_0 = F_0(\underbrace{\infty, \dots, \infty}_{m}).$$

The values

$$F_0 u_i(0) - \sum_{l=1}^{\infty} F_i(l)\left[w_i(l) - u_i(l)\right]$$

may be considered as the measures of reserve of unloading fronts' through-put.

The results obtained can be used for optimization of design and management of transshipment points' activity. Some of them were applied for technological projecting of ports' terminals of some ports at the Black Sea coast.

REFERENCES

[1] Voevudskiy,E.N. and Postan,M.Ya.,On Stochastic Models of Transport Flows' Interaction at Transshipments. Kibernetika i Sistemny Analiz, 1 (1993),101-112 (in Russian).

[2] Postan,M.Ya. and Didorchuk,N.P.,Stochastic Model of the One-Channel Transport-Warehouse System with Feedback. In: A.A.Bakaev (Ed.),Problemy Vnedreniya Informatzionnyh Tehnologiy na Transporte. V.M.Glushkov Institute of Cybernetics of Acad. Sc. of Ukraine, Kiev (1992), 33-37 (in Russian).

Alternative Models of LH Pulse Generation

by David Brown, Jonathan Foweraker, Allan Herbison & Robert Marrs
AFRC Babraham Institute, Babraham Hall, Cambridge, CB2 4AT, England.

Abstract. Brown et al [1] describe a model of a luteinising hormone (LH) pulse generator in which LH-releasing hormone (LHRH) and gamma-aminobutyric acid (GABA) cells form an excitable (Fitzhugh-Nagumo type) system, stimulated by stochastic input from adrenergic cells. Such a model exhibits behaviour which correlates well with the observed behaviour of the real neural system: pulsatile output of LHRH and therefore LH, with varying degrees of temporal correlation between LHRH pulses and adrenergic pulses, depending on other model parameters; and a non-monotonic relationship between adrenergic pulse frequency and LHRH pulse frequency. In this paper, we compare this behaviour with that of an alternative model in which the LHRH and GABA cells form an oscillator.

1. Introduction

The secretion of luteinising hormone (LH) from the anterior pituitary into the blood stream is an essential component of the mechanism underlying ovulation. In ovariectomised animals, the steroid feedback from the ovaries to the brain is removed, and the neural control of LH secretion can be observed in a simpler environment. Such animals produce regular high-frequency pulses of LH which correlate well with secretion of LH-releasing hormone (LHRH) from nerve terminals in the median eminence. The LHRH cell bodies are scattered throughout the hypothalamus and little is known about their electrophysiological properties and interactions with other neuronal types. A mathematical model of the system is therefore likely to be helpful when synthesising the very large amount of indirect functional evidence into a coherent scientific hypothesis. Brown et al [1] present a simple non-linear dynamical model in which LHRH and gamma-aminobutyric acid (GABA) form an excitable system of Fitzhugh-Nagumo type (outlined briefly in section 2), pulses being triggered by adrenergic input from other areas of the brain. This model has many properties which characterize the LH system.

Although noradrenaline is believed to play in a role in LHRH pulsatile activity, several authors [2][3] have found that pulsatile activity can occur in the absence of adrenergic input. This evidence conflicts with other experimental findings: e.g. in the monkey, where the extracellular noradrenaline concentrations fluctuate in parallel with LHRH release from the median eminence, suggesting some connection between adrenergic input and LHRH pulse output, at least in this species. Nevertheless, we felt that it would be of value to examine the hypothesis that LHRH and GABA cells form an oscillator without adrenergic input; and to assess how such a system would behave when subject to adrenergic perturbations. Such a modelling exercise might suggest experiments which could help to answer the question: does the LHRH/GABA neural network constitute an excitable system (H_E - the hypothesis of excitability) or an oscillating system (H_O) or a hybrid system? In section 2, we briefly outline a general model of which the two main competing models, H_E, H_O are special cases. Section 3 describes the results of a simulation exercise, assessing pulse output frequency in relation to adrenergic stimulation frequency, and temporal correlation between stimuli and output pulses. These findings are briefly discussed in section 4.

2. A general model

The model components are v, the electrical activity of the LHRH cells which are assumed to be synchronised, and g, the electrical activity of those GABA cells which have reciprocal connections with the LHRH cells. The LHRH cells form a bistable element when isolated from GABA cells and adrenergic input, and the simplest form in which this can be modelled is using a cubic in v; also high GABA activity depresses LHRH activity. GABA activity declines exponentially in the absence of other influences, but is subject to linear input from v. Both are subject to the influence, which could be excitatory or inhibitory, of the tonic level of adrenergic activity, a. The simplest model consistent with these requirements is

$$\frac{dv}{dt} = s_0[-v(v-c)(v-1)-k_1g+k_2a]$$

$$\frac{dg}{dt} = b(v)[k_3a+k_4v-k_5g].$$

where $k_1, \ldots, k_5$ are the synaptic strengths of the various inputs on v and g. $k_1, k_4, k_5 \geq 0$, but k_2, k_3 can take any sign. $b(v)$ is a velocity-scaling function which in the present paper is taken to be a constant, b. By writing $w = g - k_3a/k_5$, and $I_{net} = a(k_2 - k_1k_3/k_5)$ the equations become those of the Fitzhugh-Nagumo model [4][5], which can be further simplified using scale-independent variables to give the second equations below:

$$\frac{dv}{dt} = s_0[-v(v-c)(v-1)-k_1w+I_{net}] \quad \rightarrow \quad \frac{dv}{dt} = \gamma[-v(v-\alpha)(v-1)-w+I_{net}]$$

$$\frac{dw}{dt} = b[k_4v-k_5w] \quad \rightarrow \quad \frac{dw}{dt} = v-\beta w$$

I_{net} is a quantity which reflects the influence of the tonic adrenergic input as interpreted by the LHRH/GABA network. It also acts in the same way as an applied current in the Fitzhugh-Nagumo model, and is thus termed a *network current*. In the present case, to allow for the fact that the system might form an oscillator in the absence of adrenergic input we include a further term, I_0, so that I_{net} is not zero when a=0, giving $I_{net} = I_0 + a(k_2 - k_1k_3/k_5)$. The adrenergic input fluctuates and the effects of these fluctuations are hypothesised to be discrete positive perturbations of v, of size Δv, arriving at times which follow a negative exponential distribution with displaced origin. Depending on the ratio of the displacement to the mean of the exponential, such a point stimulation process could vary from a (temporally) completely random Poisson process to a completely regular pattern of stimulation. There is of course a relationship between a and the distributions of Δv and Δt, of a form $a = q\mathrm{E}[\Delta v/\Delta t]$, where q is a constant. If f_a is the mean frequency of stimulation, and we assume that Δv is also constant, then $a = q\Delta v\mathrm{E}[1/\Delta t] = q\Delta v f_a$. In this initial analysis, we also assume for greater simplicity that the stimulations of v arrive at constant intervals, $\Delta t = 1/f_a$. Then I_{net} can be written $I_{net} = I_0 + (k_2 - k_1k_3/k_5)q\Delta v f_a = I_0 + d f_a$.

The advantages of this formulation are twofold. First, it presents us with the two basic forms of dynamics within a single general model: provided $I_{net} < I_1$, the equations represent an excitable system; for $I_1 < I_{net} < I_2$, the equations represent an oscillating system without stimulatory input. The thresholds I_1, I_2 are functions of the other parameters of the system. Secondly, it enables us to consider separately the effects of changes in the tonic level of adrenergic input (as reflected in the network current), changes in the behaviour due to the I_0 component of I_{net}, and finally changes in the patterning and rate of the fluctuations of adrenergic input.

The definition of what constitutes an LHRH pulse is rather arbitrary. The following definition is used in this paper: a pulse occurs at time $t = t_p$ if $v(t_p) = \tau$, and $v(t) < \tau$ for $t_p - \varepsilon_1 < t < t_p$ and $v(t) > \tau$ for $t_p < t < t_p + \varepsilon_2$ for $\varepsilon_1 > 0.1$ and $\varepsilon_2 > 0$. Put simply, τ is the threshold which must be crossed for a pulse to occur, and there must be a period of lower values of v for a time at least 0.1 before such a crossing. The minimum value of ε_1 of 0.1 and the value of $\tau = 0.6$ used were found to give intuitively acceptable results for the present simulations.

3. Simulation experiments

The issues addressed in these experiments are as follows. (1) What is the relationship between pulse output frequency (of v = LHRH) and stimulus frequency? Is a non-monotonic relationship, which would be consistent with experimental findings, possible? (2) What degree of synchronisation is there between stimuli and output pulses?

The simulations were carried out using a Fortran 77 program calling NAG numerical integration and random number generator routines and UNIRAS graphical routines, on a DECstation operating under UNIX or a Microvax 4000 under VMS. In the results presented, where not otherwise stated, we ignored an initial transient period of 10 time units (equivalent to about 10 cycles of the oscillatory version for these parameters) after starting at $(v, w) = (0, 0)$, and quote statistics of behaviour for the succeeding 10 time units. In most, but not all, cases, this allowed the eventual attractor to be reached. The parameter values given in the legend to Figure 1 were used because, with these values, the excitable system variant of the model (i.e. with $I_{net} = 0$) is known to exhibit behaviour which is similar to that of the real system.

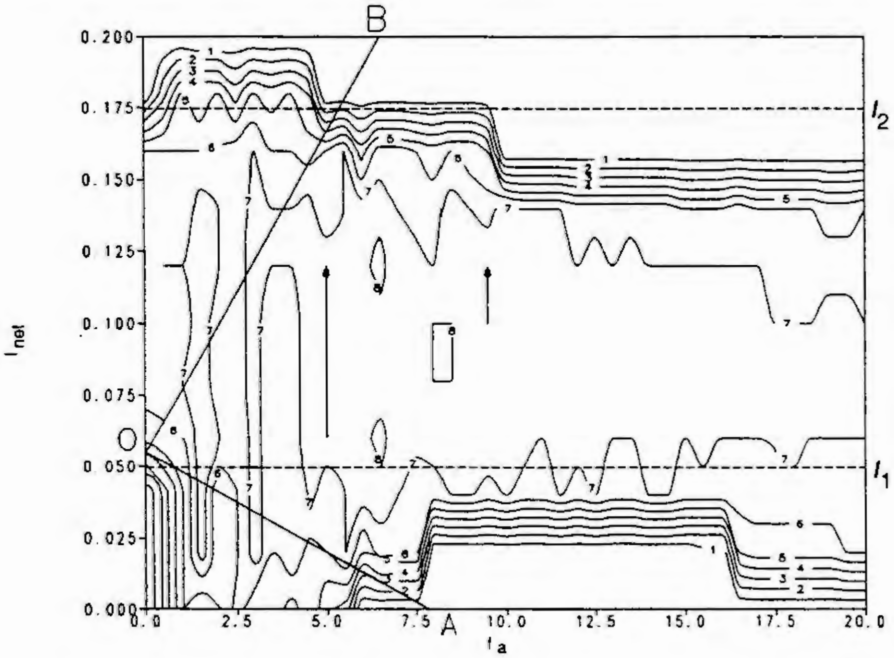


Figure 1. A contour diagram of LHRH pulse frequency (f_p) plotted against I_{net} and f_a. f_p (in pulses/unit time) is obtained by multiplying the contour codes by 0.2. The parameter values used were $\alpha = 0.20, \beta = 2.5, \gamma = 200, \Delta v = 0.33$. Also plotted as dotted lines are the thresholds on I_{net} for the unstimulated system to form an oscillator, $I_1 = 0.0495, I_2 = 0.1745$. Two solid lines (OA,OB) trace the changes in LHRH pulse frequency as f_a increases from a point O within this range (at which $I_0 = 0.055$) with slopes against f_a of d=-0.007 and 0.03. The figure is based on values of I_{net} spanning the range 0 to 0.20 at 0.02 intervals, and of f_a ranging from 0 to 20 in steps of 0.5.

(1) Relationship of LHRH pulse frequency to adrenergic stimulus frequency

$I_{net} = I_0 + df_a$ is a straight line on the contour diagram in Figure 1, intercepting the I_{net} axis at $I_{net} = I_0$ and with slope d. Possible relationships between LHRH pulse frequency (f_p) and f_a, as f_a increases from zero, can be traced by following straight line paths across this contour diagram, and assessing the intersection of vertical planes through them with the surface. An example is indicated by the path OA in Figure 1: first of all f_p rises slightly, but then quickly falls to zero. There are no straight line paths starting from O (or from any point on the I_{net} axis in the oscillator range, *i.e.* $I_1 < I_{net} < I_2$) which can be drawn across the figure in which substantial departures from a monotonic falling relationship as f_a increases occur. For all values of d, f_p rises slightly or stays the same as f_a increases; provided d is great enough in absolute magnitude, it then might fall substantially at high levels of f_a. The only cases in which f_p rises initially and substantially as f_a increases correspond to paths starting from the lower left corner (i.e. $I_0 < I_1$). All paths such that $I_0 < 0.03$ and with $d \leq 0$ show a substantial rise followed by a fall to zero, and a possible further rise. Thus non-monotonic relationships only occur when the unstimulated system is excitable as opposed to oscillatory. All the oscillators plotted on the other hand seem very resistant to change in their natural, unstimulated frequency, except possibly at high levels of I_{net} and f_a, where stimulation keeps the oscillator above the pulse threshold of $\tau = 0.6$.

(2) Synchronisation between adrenergic stimuli and LHRH pulses

The synchronisation between adrenergic stimuli and LHRH pulses is less good when considering the oscillator form of the LHRH/GABA network (the LG-oscillator). If H_E holds, then the only cases in which an LHRH pulse can occur are when triggered by an adrenergic stimulus; not all such stimuli result in an LHRH pulse, but they are necessary for pulses to occur. Thus we never get situations with many LHRH pulses per adrenergic stimulus; when f_a is sufficiently low, the adrenergic stimuli are equal in number to, and coincident with, the LHRH pulses. If f_a is sufficiently high, then n>1 stimuli can result in just one LHRH pulse, so in this case synchronisation between stimuli and pulses is limited to coincidences of stimuli and LHRH pulse every nth stimulus.

If H_O holds, then LHRH pulses occur without simultaneous adrenergic stimuli; and this will almost always be the case when the natural (i.e. undisturbed) frequency of oscillation of the LG-oscillator (f_n) is much greater than f_a. When there is approximate comparability between f_a and f_n, quite frequently the regular stimulation entrains the LG-oscillator, and possibly slightly modifies its pulse frequency. The simulation results already available indicate that when there is a substantial discrepancy between f_a and f_n, there is frequently a many stimuli-one LHRH pulse, or a one stimulus-many LHRH pulse entrainment; only in this rudimentary sense does synchronisation occur. The simulations results indicate in general a resistance to more than slight modification of the natural frequency of the oscillator, and hence there is very limited scope for synchronisation.

4. Discussion

The present simulation results have been confirmed for some other values of Δv, and qualitatively similar results have been obtained with other values of γ, and with other temporally regular patterns of stimulation. If the pattern of stimulation approaches a pure Poisson process, the non-monotonic pulse/stimulus frequency relationship breaks down, indicating that substantial regularity is necessary. For much larger, possibly non-physiological, values of Δv (0.5-0.7), a non-monotonic relationship between f_p and f_a does occur with the oscillator form of the model. Some other form of oscillator might also be at work in the findings detailed in [2] and [3], although the details of the present form are based on many experimental results. If a different form of oscillator operates, it is possible that it will be more amenable to modification by adrenergic input. A further explanation of the findings that LHRH activity can be pulsatile in the absence of adrenergic input is that the neural networks when deprived of adrenergic input are modified. That the network spontaneously bursts in these very different circumstances is no guide to the normal behaviour of the network. The findings of the present, admittedly limited, simulation study tend to reinforce this interpretation.

References

[1] Brown, D., Herbison, A.E., Leng, G. & Marrs, R.W. A mathematical model for the neural control of luteinizing hormone secretion. *J. Ag. Sci. Camb.*, **119** (1992), 137-141.

[2] Kokoris, G.J., Lam, N.Y., Ferin, M., Silvermans, A. & Gibson, M.J. Transplanted gonadotrophin-releasing hormone neurons promote pulsatile luteinizing hormone secretion in congenitally hypogonadal (hpg) male mice. *Neuroendocrinology*, **48** (1988), 45-52.

[3] Leonhardt, S., Jarry, H., Falkenstein, G., Palmer, J., & Wuttke, W. LH release in ovariectomized rats is maintained without noradrenergic neurotransmission in the preoptic/anterior hypothalamic area: extreme functional plasticity of the GnRH pulse generator. *Brain Research*, **562** (1991), 105-110.

[4] Fitzhugh, R. Impulses and physiological states in theoretical models of nerve membranes. *Biophys. J.* **1** (1961), 445-466.

[5] Nagumo, J.S., Arimoto, S. & Yoshizawa, S. An active pulse transmission line simulating nerve axon. *Proc. IRE.*, **50** (1962), 2061-2071.

MATHEMATICAL MODEL OF PACEMAKER ACTIVITY IN BURSTING NEURONS OF SNAIL

Nikolai I. Kononenko

Laboratory of Neurobiology, A.A.Bogomoletz Institute of Physiology,
Bogomoletz str., 4, 252601 GSP, Kiev-24, Ukraine

Abstract. A mathematical model of pacemaker activity in bursting neurons of snail including minimal model of membrane potential oscillations, spike-generating mechanism, inward Ca current, intracellular Ca ions, $[Ca^{2+}]_{in}$, and their buffering, $[Ca^{2+}]_{in}$-inhibited Ca conductance has been developed. The model presented demonstrates adaptation of bursting activity to both polarizing current and changing of stationary Na or K conductances, hysteresis properties.

1. INTRODUCTION

Some identified molluscan neurons generate slow membrane potential (MP) oscillations which trigger bursts of action potential (AP) during depolarizing phase. At present there is no difficulty in generation of bursting electrical discharge in computer simulation based upon experimentally observed membrane and cytoplasmic processes in bursting neurons [3,4,8]. The most difficult task of such modeling is to estimate correctly which the ionic channels and intracellular processes are necessary and sufficient to evoke electrical bursting activity in intact cells and which ones only accompany and modulate bursting activity generation. Earlier a minimal mathematical version of the slow-wave MP oscillations in bursting neurons of snail *Helix pomatia* based on experimental data has been formulated [6]. Main of these data are as follows: a) the presence of a negative resistance region (NRR) on the stationary current-voltage relation (CVR) of a bursting neuron, b) activation of an outward time-dependent current upon hyperpolarization of the bursting neuron membrane, c) the existence of persistent bursting activity under blocking of inward calcium current. Subsequent analysis has shown that minimal version describes many experimental phenomena including an increase of input resistance of bursting neuron during an interburst interval, dependence of oscillation period on polarizing current, induced hyperpolarization, contingent stimulation, bursting activity modulation, etc.[7]. However, some principal facts obtained in experiments with intact bursting neurons could not be simulated in the framework of the minimal model. There are two main facts: firstly, "adaptation" of the frequency of slow-wave oscillations to constant polarizing current [2]; secondly, appearance of slow inward current with the time constant of about 20 s upon clamping the MP of bursting neuron at different phases of wave development [9]. It seems likely that both these events have common origin which is connected with spike generation in intact bursting neuron. It has been hypothesized that average firing frequency produces stationary outward current which is balanced by inward leakage current in the absence of external influence [7]. It is easy to see that a change in average frequency evoked by membrane polarization has to produce adaptation of the slow-wave oscillations. In accordance with Adams and Levitan [1], this apparent outward current is due to a decrease in resting inward Ca current as a results of $[Ca^{2+}]_{in}$ -induced inactivation of stationary Ca conductance. The model developed in this study simulates the features described above.

2. RESULTS

Our model has the following form: $-C_m \dfrac{dV}{dt} = I_K + I_{Na} + I_{Na(V)} + I_B + I_{Na(TTX)} + I_{K(TEA)} + I_{Ca} + I_{Ca\text{-}Ca}$

This model includes: a) the minimal version of slow wave generation [6], b) spike-generating mechanism consisting of both TTX-sensitive sodium and TEA-sensitive potassium conductances, c) potential-activated Ca conductance, d) intracellular Ca ions and their buffering, e) $[Ca^{2+}]_{in}$-inhibited stationary potential-dependent Ca conductance. The corresponding equations and parameters can be found in Appendix.

2.1. Electrical bursting activity

The predicted dependencies of both V(t) and $[Ca^{2+}]_{in}$ are given in Fig.1. It should be noted that the studied model is rather stable and permits considerable changing of its parameters. One can see from Fig.1. that jumps of $[Ca^{2+}]_{in}$ are exactly coordinated in time with spike generation that is in accordance with data obtained on intact bursting cells [5].

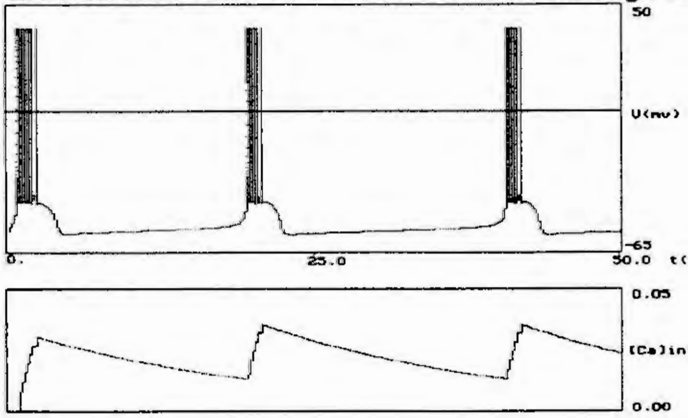


Fig.1. Time course of both electrical bursting activity (upper record) and corresponding intracellular calcium concentration (lower record).

2.2. Adaptation properties of the model

In the studied model of bursting activity application of constant hyperpolarizing current produced at first an increase of the interburst interval with its subsequent partial restoration. As analysis showed, partial restoration of the interburst interval is due to the decrease of $[Ca^{2+}]_{in}$ average level and corresponding increase of $[Ca^{2+}]_{in}$-inhibited stationary potential-dependent calcium conductance. This elevation of calcium conductance produced a compensatory inward current which depolarized the neuronal membrane. It is the so called "adaptation" of electrical bursting activity to polarizing current. On the contrary, the increase of stationary sodium conductance evoking membrane depolarization and decrease of the interburst interval produced increase of average frequency of AP generation and corresponding increase of $[Ca^{2+}]_{in}$ background. This elevation of $[Ca^{2+}]_{in}$ inhibited stationary Ca conductance, $g_{Ca\text{-}Ca}$, and produced restoration of interburst interval, respectively (Fig.2).

2.3. Hysteresis of current-voltage relations

The hysteresis of CVR of the bursting neuron membrane is the well-known phenomenon correlating with oscillatory membrane activity [4]. In this case the CVRs of intact bursting cell obtained with depolarizing and hyperpolarizing ramps were different; with depolarizing ramp, the membrane showed a NRR while with hyperpolarizing ramp the negative resistance was strongly diminished and considerable inflection was to be seen. The present model demonstrates hysteresis properties similar to those observed in intact bursting neurons (Fig.3) after inhibition of potential-activated sodium current participating in AP generation. This corresponds to external application of tetrodotoxin which suppresses APs in unclamped region thus allowing more adequate registration of the hysteresis cur-ve in intact cell [4].

Thus, despite the limitations inherent in the model studied, it predicts many features of electrical discharge in the bursting cells. The present model satisfactorily describes both the MP behavior of bursting neuron under current clamp conditions and the behavior of membrane current under voltage clamp conditions when MP was stopped at different phases of wave development or clamped at different levels and then shifted to testing potential. Finally, model experiments show that intracellular calcium ions taken together with $[Ca^{2+}]_{in}$-inhibited resting calcium conductance play a key role in adaptation of electrical activity of bursting neuron to polarizing current in particular and to alteration of model parameters in general.

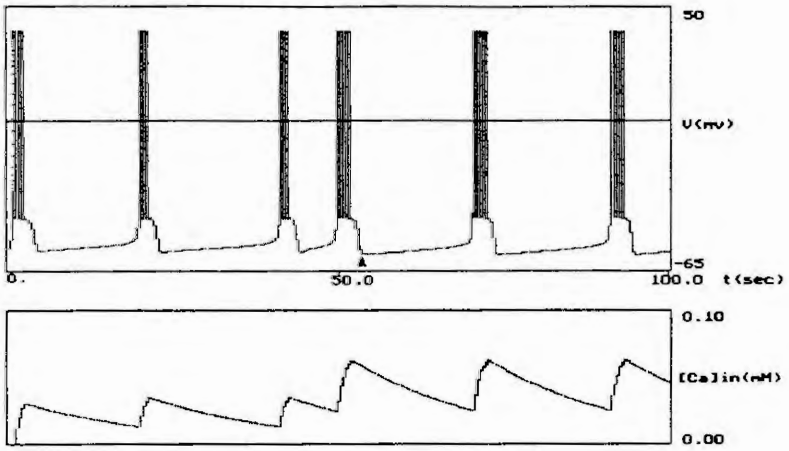


Fig.2. Effect of changing of stationary potential-dependent Na conductance, $g^{(V)}_{Na}$, on both the interburst interval (upper record) and $[Ca^{2+}]_{in}$ (lower record). The triangle shows the moment when g^*_{Na} was changed from initial 0.15 to 0.23 1/MΩ.

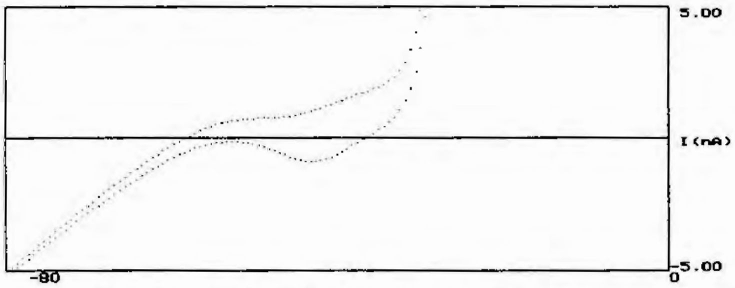


Fig.3. Current-voltage relation obtained with slow ramp in model neuron membrane. $g^*_{Na(TTX)}$ is taken by zero. Holding potential was ramped from -80 to 0 mV and backward.

Appendix.

a) minimal version of slow wave generation [6]:

$$-C_m\frac{dV}{dt} = g_K(V - V_K) + g_{Na}(V - V_{Na}) + g^{(V)}_{Na}\frac{1}{1 + \exp\left[k_{Na}(V - V^*_{Na})\right]}(V - V_{Na}) + g_B m_B h_B(V - V_B);$$

$$\frac{dm}{dt} = \frac{m_\infty - m}{\tau_m}; \qquad m_\infty = \frac{1}{1 + \exp[k_m(V - V^*_m)]};$$

$$\frac{dh}{dt} = \frac{h_\infty - h}{\tau_h}; \qquad h_\infty = \frac{1}{1 + \exp[k_h(V - V^*_h)]};$$

Here and below, C is the membrane capacity, V denotes membrane potential (mV); m_B and h_B are the V-dependent, the Hodgkin-Huxley-like, activation and inactivation variables for g_B conductance.

b) spike-generating mechanism: $I_{Na(TTX)} = g^*_{Na(TTX)}m^3h(V - V_{Na})$; $\quad I_{K(TEA)} = g^*_{K(TEA)}n^4(V - V_K)$;

$$\frac{dm}{dt} = \frac{m_\infty - m}{\tau_m}; \quad m_\infty = \frac{1}{1 + \exp[k_m(V - V^*_m)]}; \quad \frac{dh}{dt} = \frac{h_\infty - h}{\tau_h}; \quad h_\infty = \frac{1}{1 + \exp[k_h(V - V^*_h)]};$$

$$\frac{dn}{dt}=\frac{n_\infty - n}{\tau_n};\;\; n_\infty=\frac{1}{1+\exp[k_n(V-V_n^*)]};$$

c) inward potential- and time-dependent Ca current:

$$I_{Ca}=g_{Ca}^*m^2(V-V_{Ca});\;\; \frac{dm_{Ca}}{dt}=\frac{m_{\infty Ca}-m_{Ca}}{\tau_{m(Ca)}};\;\; m_{\infty Ca}=\frac{1}{1+\exp[k_{m(Ca)}(V-V_{m(ca)}^*)]};$$

The dependencies of τ_m, τ_h, τ_n and $\tau_{m(Ca)}$ on membrane potential were ignored.

d) Time evolution of $[Ca^{2+}]_{in}$ in the model is presented by equation: $\frac{d[Ca]}{dt}=-\frac{I_{Ca}}{2F\upsilon}-k_s[Ca];$

Here, [Ca] is $[Ca^{2+}]_{in}$; F is the Faraday number; υ is the volume of the cell taken $4/3\pi R^3$; k_S is the rate constant of intracellular Ca-uptake by a buffer systems.

e)stationary potential-dependent $[Ca^{2+}]_{in}$-inhibited Ca current:

$$I_{Ca\text{-}Ca}=g_{Ca\text{-}Ca}^*\frac{1}{1+\exp[k_{Ca\text{-}Ca}(V-V_{Ca\text{-}Ca}^*)]}\cdot\frac{1}{1+\exp[k_\beta([Ca]-\beta)]}(V-V_{Ca});$$

Parameters for Eqs.

1) Minimal model of slow-wave generation:

g_K= 0.25 1/MΩ; g_{Na}= 0.02 1/MΩ; $g_{Na}(V)$ = 0.1 1/MΩ; g_B= 0.1 1/MΩ; C_m = 0.02 μF.
V_K= -70 mV; V_{Na} = +40 mV; V_B = -58 mV; k_{Na} = -0.2 1/mV; V^*_{Na}= -45 mV;
k_m= 0.4 1/mV; V^*_m = -34 mV; τ_m = 0.05 s; k_h = -0.55 1/mV; V^*_h = -43 mV; τ_h = 1,5 s;

2) Spike-generating mechanism: $g^*_{Na(TTX)}$ =400 1/MΩ; $g^*_{K(TEA)}$ =10 1/MΩ;
k_m=-0.4 1/mV; V^*_m=-31 mV; τ_m=0.0005 s; k_h= 0.25 1/mV; V^*_h=-45 mV; τ_h=0.01 s;
k_n =-0.18 1/mV; V^*_n =-25 mV; τ_n=0.015 s;

3) Potential-activated calcium inward current:
g^*_{Ca}=1.5 1/MΩ; V_{Ca}=150 mV; $k_{m(Ca)}$=-0.2 1/mV; $V^*_{m(Ca)}$=0 mV; $\tau_{m(Ca)}$=0.01 s;

4) Intracellular calcium concentration: R= 100 μ ; k_S=0.05 1/s;

5) $[Ca^{2+}]_{in}$-inhibited calcium current:
$g^*_{Ca\text{-}Ca}$=0.02 1/MΩ; $k_{Ca\text{-}Ca}$=-0.06 1/mV; $V^*_{Ca\text{-}Ca}$=-45 mV; k_β=30 1/mM; β=0.02 mM;

3. REFERENCES

[1] Adams, W.B., Levitan, I.B., Voltage and Ion Dependences of the Slow Currents Which Mediate Bursting in *Aplysia* Neurone R15. J. of Physiology, 360 (1985), 69-93.

[2] Arvanitaki, A., Chalazonitis, N., Electrical Properties and Temporal Organization in Oscillatory Neurons (*Aplysia*). In: J. Salanki (Ed.), Neurobiology of Invertebrates. Akademiai Kiado, Budapest, 1967.

[3] Canavier, C.C., Clark, J.W., Byrne, J.H., Simulation of the Bursting Activity of Neuron R15 in *Aplysia*: Role of Ionic Currents, Calcium Balance, and Modulatory Transmitters. J.of Neurophysiology, 66 (1991), 2107-2124.

[4] Gola, M., Electrical Properties of Bursting Pacemaker Neuron. In: J. Salanki (Ed.), Neurobiology of Invertebrates. Akademiai Kiado, Budapest, 1976.

[5] Gorman, A.L.F., Hermann, A., Thomas, M.V., Intracellular Calcium and the Control of Neuronal Pacemaker Activity. Federation Proceedings, 40 (1981), 2233-2239.

[6] Kononenko, N.I., Mechanisms of Membrane Potential Oscillation in Bursting Neurons on the Snail, *Helix Pomatia*. Comparative Biochemistry and Physiology, 105A (1993), X-X.

[7] Kononenko, N.I., Dissection of a Model for Membrane Potential Oscillations in Bursting Neuron of Snail, *Helix Pomatia*. To appear in: Comparative Biochemistry and Physiology, (1993).

[8] Rinzel, J., Lee, Y.S., Dissection of a Model for Neuronal Parabolic Bursting. J. of Mathematical Biology, 25 (1987), 653-675.

[9] Wilson, W.A., Wachtel, H., Negative Resistance Characteristic Essential for the Maintenance of Slow Oscillations in Bursting Neurons. Science, 186 (1974), 932-934.

DYNAMIC BEHAVIOUR AND SELF-ORGANIZATION OF NEURAL NETWORKS INFLUENCED BY NOISE INPUT

V. CHINAROV

Scientific Research Center "Vidhuk"
Vladimirskaya Str. 61-b, Kiev 252033, Ukraine

Abstract. The dynamic behaviour of the neural network with only the excitatory population of neurons being driven by external noise and the processes of self-organization in networks is investigated. The algorithm for the optimal trajectory in the phase plane of a system connected distinct basins of attraction is proposed.

1. INTRODUCTION

Information generated and processed by neural systems is expanded in space and time. To study its temporal organization it is reasonable to consider the influence of the dendritic tree topology on the network activity dynamics. The average number of the inter-synaptic junctions needed for processing information is approximately 10 [11]. It is clear that due to such a high connectivity of neural nets, one may treat them as dynamic systems characterized by the ignorance of initial conditions and possessing the stochastic nature of generation of impulse activity.

In contrary to gradient methods realizing associative memory organization [8] and using the feedback mechanism for synaptic alterations [6], relatively low synaptic fluctuations (without any feedback) may provide the function of mechanisms of successive recall of the stored information [12]. The important characteristics of dynamic systems with coexisting attractors are probabilities of transitions between them. These attractors may be associated with memories stored by the neural network [7, 11, 12].

Several papers have appeared in recent years dealing with the problem of the influence of noise on the dynamics of neural system [1, 2, 3, 11]. It was shown that transitions between attractors in neural networks with excitatory and inhibitory interactions (both fixed point and limit cycle types of attractors) driven by stochastic inputs may change substantially the level of self-organization of the network [2, 3]. In some cases low intensity noise may result in considerable transformations of the phase plane of the system and even in the appearance of stochastic oscillations.

The noise may also stabilize the processes of pattern recognition in the presence of feedback describing the alteration of membrane potential of terminal's fibers along which the excitatory influences on the network are entering. Such a network effectively changes its dynamic characteristics in such a way that the phase portrait of a system become more stable [2].

In the present paper the dynamic behaviour of the neural network on the basis of the Wilson-Cowan model [13] with only the excitatory population of neurons being driven by external noise are analyzed. The problem is actual for many neurophysiological systems, in particular, for olfactory bulb of mammals [1, 6] and system controlling movement activity [2, 10].

2. THE MODEL

Let us consider the neural network with its dynamic state to be described in the limiting case of two homogeneous populations of neurons (i=1,2) by a point on the phase plane (X_1 ,X_2) of a system

$$dX_i / dt = -X_i + (k_i - X_i) \cdot S\left(\sum_{k=1} \alpha_{ik} X_k + f_i(t) + \xi_i(t)\right), \tag{1}$$

where the excitatory (X_1) and inhibitory (X_2) activities are the proportions of excitatory and inhibitory cells, respectively, firing per unit time, the factors k_i-X_i are equivalent to the shunting excitation (i=1) and inhibition (i=2) terms in the neural membrane equations (constants k_i determine the interval values of automatic gain control [7], $f_i(t)$ are the inputs to the excitatory and inhibitory populations, and the coefficients α_{ik} describe the average strength of excitatory (α_{ik}>0) and inhibitory (α_{ik}<0) synapses as well as average inhibitory influence upon the excitatory population [13].

The sigmoid function $S(u)$ in (1) represents the neuron response to the inputs from the adjacent neurons and has a form

$$S(u) = 1/[1+\exp(-B(u-A))], \qquad (2)$$

where the parameters A and B describe, respectively, the average threshold and its dispersion over the whole population of neurons. The tenns $\xi_i(t)$ (i=1,2) in (1) are the noise inputs to the network. Their nature will be discussed below.

We are interested here in such a case when values of the parameters α_{ik}, k_i, $f_i(t)+\xi_i(t)$ are such that system (1) has two stationary stable states. These states may be either focuses (nodes), or limits cycles. The stable state may be characterized by the domain of attraction on the phase plane (with the boundary in a form of unstable limit cycle), or may possess the separatrix passing through the saddle point (unstable steady state) and dividing different domains of attraction.

Due to the large-scale fluctuations of the membrane potentials in (1) there may occur, besides the relaxation to steady states relatively large rare fluctuations causing the transitions between them [4]. We will assume that characteristic probabilities W of such transitions are much less than the reciprocal relaxation times τ to the steady state. In result of transitions between steady states some stationary distribution over the system states arises and dependence of statistical characteristics of a system on its parameters becomes single-valued (e.g., average summary activity of neural network $<X_1 + X_2>$ uniquely depends on the input f_1 to excitatory net). We will have than the following inequality when the network is to be used as a memory unit

$$\tau_r \le \tau_{SW} \sim (1/f_1)|df_i/dt|, \quad \tau_O < W^{-1}, \qquad (3)$$

where τ_{SW} is switching time and τ_O is a characteristic time between switchings.

The size of minimal characteristic scale of the switching time is τ_r ($\tau_{SW} \ge \tau_r$) for arbitrarily fast changing of control parameter f_1. The times W^{-1} define "the safety limit" for the storing of information by neural network. Thus, for the reliable performance of the switching element in the working range of parameters the following condition must be fulfilled

$$W << \tau_r^{-1}. \qquad (4)$$

Practically this is a condition on the relative weakness of noise. It leads to the situation when "activation energy" (this definition is of conditional nature because in the non-equilibrium system transitions are not described by the standard Arrhenius law) of the escape from the steady state is much greater than the noise intensity and exponential factor determines, to logarithmic accuracy, the nature of dependence of transition probabilities W on the system parameters. In this case it has sharp exponential dependence on the control parameter f_1. However, in the vicinity of bifurcation points, when condition (4) is not fulfilled, "the safety limit" for the storing of information sharply decreases as working range of control parameter becomes nearer to the bifurcation point. On the other hand, the closer the system approaches to the bifurcation point the smaller the characteristic work of the switching and energy expenditure in memory units.

The existence of above-mentioned alternative specifications of parameters for performance of optimal ratio between reliability and memory storage capacity of the network, makes the problem of determination of W as a function of system parameters (in particular, of the control parameter $f(t)$) to be of great significance.

3. FLUCTUATIONAL TRANSITIONS BETWEEN ATTRACTORS OF THE NETWORK

The problem of influence of the dendritic tree topology on the network activity dynamics we consider in the framework of an approach taking into account the fluctuation of the membrane potential averaged over the small volume of neurons. Fluctuations of averaged non-equilibrium potential are calculated in following approximation. Characteristic dendritic scale of the "dressed" neuron which define the damping length of fluctuations (diffusion length), is much less than the mean inter-neuron distance and therefore the fluctuations of the membrane potential for each neuron are independent.

Let $\xi_i(t)$ be Gaussian stationary random process. The probability density functional for its realization has a form [5]

$$P[\xi(t)] = \exp\left[-\sum_{i,k}\iint dt dt' F_{ik}(t-t')\xi_i(t)\xi_k(t')\right], \tag{5}$$

where function F is defined by relations

$$\Phi_{ik}(w) = \left(F^{-1}(w)\right)_{ik}, \quad F(w) = \int_{-\infty}^{+\infty} dt \exp(iwt) F_{ik}(t), \tag{6}$$

where $\Phi_{ik}(w)$ is a spectral representation of the correlation function of random process $\xi_i(t)$

$$\Phi_{ik}(\tau) = \langle \xi_i(t) \cdot \xi_k(t-\tau) \rangle. \tag{7}$$

For the times $W^{-1} >> t >> \tau_r$ quasistationary distribution $\rho(X_1, X_2)$, arising in the phase space has a form

$$\rho(X_1, X_2) = W(\bar{X}, X_a), \qquad (\bar{X} = (X_1, X_2)), \tag{8}$$

where

$$W(\bar{X}, X_a) = \frac{\int\limits_{\bar{X}(-\infty)=\bar{X}_a} D\xi_1(t) D\xi_2(t) \delta(\bar{X}(t) - \bar{X}) P[\xi(t)]}{D\xi_1(t) D\xi_2(t) P[\xi(t)]}. \tag{9}$$

The integral in (9) is over those fluctuations which lead the system (1) to the point X at a moment t. Boundary conditions are such that for $t \to -\infty$, $\bar{X}(-\infty) = \bar{X}_q$ (point on the initially filled attractor).

In the case of low intensity noise the probabilities of different noise realizations strongly differ among themselves and main contribution into the continual integral gives the small vicinity of the noise optimal trajectory to which certain trajectory of the dynamic system (1) corresponds [5]. The equation for this trajectory may be obtained using variational principle for $P[\xi(t)]$

$$\delta \sum_{i,k} \int dt \xi_i(t) F_{ik}[-id/dt] \xi_k(t) = 0. \tag{10}$$

Solving (1) with respect to $\xi_i(t)$ we will obtain from (10)

$$\delta \sum_{i,k} \left[S^{-1}\left(\frac{\dot{X}_i + X_i}{k_i - X_i}\right) - \sum_{k=1}^{2} \alpha_{ik} X_k - f_i \right]\left[S^{-1}\left(\frac{\dot{X}_k + X_k}{k_k - X_k}\right) - \sum_{j=1}^{2} \alpha_{kj} X_j - f_k \right] = 0. \tag{11}$$

This equation in common with appropriately choosen boundary conditions defines the optimal trajectory $X^{opt}(t)$ in the phase plane (X_1, X_2) which starts up on the initial attractor and goes on to the saddle point vicinity when the limit of domains of attraction is the separatrix passing through the saddle point. For the case of the white noise the transition probability is determined to logarithmic accuracy by the flux over the saddle point vicinity in the phase plane and has the following form

$$W_{ij} = const \cdot \exp\left[-\frac{1}{2D}\int dt \sum_{i,k} \xi_i^{opt}(t) \xi_k^{opt}(t)\right], \tag{12}$$

where

$$\xi_i^{opt}(t) = S^{-1}\left[\frac{dX_i^{opt}/dt + X_i^{opt}}{k_i - X_i^{opt}}\right] - \sum_{k=1}^{2} \alpha_{ik} X_k^{opt} - f_i(t). \tag{13}$$

In order to calculate the fluctuation of the membrane potential we may consider the simple case when the dendritic tree configuration enclosing neurons on the scale of order of branching length l_b (distance between bifurcations) has a spherical symmetry C_{N_O} where N_O is characteristic constant of a given dendritic greed. This symmetry decreases with increasing of its radius and is proportional on the scale of a diffusion length l_d to the ratio $l_d/<l_b>$. Fluctuations of synaptic activity on the surface of such configuration determine than in a self-consistent manner the fluctuation of the membrane potentials (i.e. fluctuations of dynamic variables X_1, X_2). Their statistical properties may be calculated knowing the nature of damping of input inside the enclosing dendritic configuration and the intensity of fluctuations on the synapses placed at the distances of order l_d, where they are statistically independent. The ratio $N = l_d/<l_b>$ is a parameter one must use for the averaging of membrane potential

$$\langle V_k \rangle = \left\langle \sum_j \alpha_{kj} X_j + f_k \right\rangle = (1/N) \sum_i V_i^k \tag{14}$$

4. RESULTS

The algorithm for searching of the optimal trajectory in the phase space of the system which crosses the saddle point and connects two distinct basins of attraction is proposed here. The optimal trajectory starts near the focus or the node at some point (its coordinates are determined by the searching procedure based on the minimum action principle), passes near the saddle point with given precision and immediately gets into the vicinity of the second focus or node. The transition probability is determined by the flux over the saddle point vicinity. Fluctuations of dynamic variables X_1, X_2 are determined by fluctuations of synaptic activity of dendritic tree enclosing neurons and its topology.

5. REFERENCES

[1] Ahn, S.M., Freeman, W.J., Neural Dynamics under Noise in the Olfactory System. Biol. Cybernetics, 17 (1975), 165-168.

[2] Chinarov, V.A., Degtyarenko, A.M., Self-Organization of Neural Network Controlling Movement Activity. In: Proc. Intern. Symp. Neural Networks and Neural Computing. Prague, 1990, 68-71.

[3] Chinarov, V.A., Degtyarenko, A.M., Feldman, M.A., Dynamic Behaviour of Neural Networks Driven by Colored Noise. In: A.V. Holden, V.I. Kryukov (Eds.) Neurocomputers and Attention. Vol. II: Connectionism and Neurocomputers. University Press, Manchester, 1991.

[4] Chinarov, V.A., Dykman, M.I., Smelanski, V.N., Dissipative Corrections to Escape Probabilities of Thermally - Nonequilibrium Systems. Phys. Rev. E47 (1993), 2448-2461.

[5] Feynman, R.P., Hibbs, A.R., Quantum Mechanics and Path Integrals. McGrow-Hill, New-York, 1965.

[6] Freeman, W.J., EEG Analysis Gives Model of Neuronal Template Matching Mechanism for Sensory Search with Olfactory. Bulb. Biol. Cybernetics 35 (1979), 221-234.

[7] Grossberg, S., Nonlinear Neural Networks: Principles, Mechanisms, and Architectures. Neural Network. 1 (1988), 17-61.

[8] Kohonen, T., Self-Organization and Associative Memory. Springer, Berlin, 1984.

[9] Li, Z., Hopfield J.J., Modeling the Olfactory Bulb and its Neural Oscillatory Processing. Biol. Cybernetics 61 (1989), 379-392.

[10] Matsuoka, K., Mechanisms of Frequency and Pattern Control in the Neural Rhythm Generators. Biol. Cybernetics 56 (1987), 345-353.

[11] Szentagothai, J., Arbib, M.A., Conceptual Models of Neural Organization. MIT Press, Cambridge, Mass., 1975.

[12] Tsuda, I., Koerner, E., Shimizu, H., Memory Dynamics in Asynchronous Neural Networks. Prog. Theor. Phys., 78 (1987), 51-71.

[13] Wilson, H.R., Cowan, J.D., Excitatory and Inhibitory Interactions in Localized Populations of Model Neurons. Biophys. J. 12 (1972), 1-24.

MODELLING OF THE EXCITATION AND THE PROPAGATION OF NERVE IMPULSES BY NATURAL AND ARTIFICIAL STIMULATIONS

FRANK RATTAY
Technical University Vienna
A-1040 Vienna, Austria

ABSTRACT

The functional properties of a neuron can be simulated by electrical circuits. This technique is of use for natural as well as for electrical stimulation. In particular, a new model is proposed which shows the influence of the myelinated parts of the neuron.

INTRODUCTION

In contrast to other cells neurons are remarkable because of their shapes showing a very complicated fine structure and because of the electrical properties of the membrane which separates the ionic components of the intracellular and the extracellular fluids. In the resting state the neuron's inside potential is constant with a value of about $75mV$ compared to that of the outside. In general, however, this membrane voltage essentially depends on location.

As an example, we assume that a neuron generates a spike train consisting of 250 spikes/s. These action potentials are due to synaptic activities in the dendritic region and at the cell body. An action potential with a duration of 1 ms will propagate with a velocity of 50m/s (=50mm/ms) along the nerve fibre (axon) which may have a diameter of $10\mu m$ and a length of 50cm. Thus, 5cm intervals of excited membrane regions which are separated by 15cm regions with resting state voltages are traveling along the nerve fibre from the cell body towards the branching part of the nerve fibre (output region). Of course, membrane voltage is also a function of space within the dendritic area and in the different branches in the terminal region.

The main elements of a neuron are the soma (cell body), the dendrites and the axon. Dendrites and soma are usually covered with synapses from other neurons (input zone), whereas the task of the axon is the transport of neural information via action potentials into distant regions. The mammalian axons with diameters from $1 - 20\mu m$ are usually myelinated, i.e. only in the nodes of 'Ranvier', which have a length of about $1\mu m$, the membrane is active. The internode is covered with many layers of membranes of Schwann cells which are tightly wrapped around the axon. The internode has a length of about 100 times the fibre diameter. The insulating properties of the internode allow a $20\mu m$ mammalian fibre to obtain a higher velocity of propagation as it is reached in the unmyelinated giant axons of a squid with diameters up to $1mm$.

The purpose of this paper is to discuss several models which finally can be used to predict the input-output relations of a single neuron.

MEMBRANE MODELS

A patch of the membrane of a neuron can be described by an electrical circuit consisting of a voltage source, which accounts for the resting voltage, a capacitance (1-$3\mu F/cm^2$) and a resistance. The conductance of the active membrane changes within a large range depending on the open/closed status of the ionic channels. The membrane models which are mostly used are the Hodgkin-Huxley model for

Box1 | HODGKIN-HUXLEY MODEL

$$\dot{V} = [-g_{Na} m^3 h (V - V_{Na}) - g_K n^4 (V - V_K) - g_L (V - V_L) + i_{st}]/c \quad \text{(HH-1)}$$

$$\dot{m} = [-(\alpha_m + \beta_m) \cdot m + \alpha_m] \cdot k \quad \text{(HH-2)}$$

$$\dot{n} = [-(\alpha_n + \beta_n) \cdot n + \alpha_n] \cdot k \quad \text{(HH-3)}$$

$$\dot{h} = [-(\alpha_h + \beta_h) \cdot h + \alpha_h] \cdot k \quad \text{(HH-4)}$$

with the coefficient k for temperature T (in $°C$)

$$k = 3^{0.1T - 0.63} \quad \text{(HH-5)}$$

and

$\alpha_m = \frac{2.5 - 0.1V}{exp(2.5 - 0.1V) - 1}$, $\beta_m = 4 \cdot exp\left(-\frac{V}{18}\right)$,

$\alpha_n = \frac{1 - 0.1V}{10.(exp(1 - 0.1V) - 1)}$, $\beta_n = 0.125 \cdot exp\left(-\frac{V}{80}\right)$,

$\alpha_h = 0.07 \cdot exp\left(-\frac{V}{20}\right)$, $\beta_h = \frac{1}{exp(3 - 0.1V) + 1}$,

$V_{rest} = -70[mV]$, $V_{Na} = 115[mV]$,

$V_K = -12[mV]$, $V_L = 10.6[mV]$,

$g_{Na} = 120[k\Omega^{-1} cm^{-2}]$, $g_K = 36[k\Omega^{-1} cm^{-2}]$,

$g_L = 0.3[k\Omega^{-1} cm^{-2}]$, $c = 1[\mu F/cm^2]$,

$m(0) = 0.05$, $n(0) = 0.32$,

$h(0) = 0.6$

Box 2 | CRRSS MODEL

SWEENEY *et al.* (1987) transformed the original data of CHIU, RITCHIE, ROGERT & STAGG (1979) from experimental temperature of $T = 14°C$ to $T = 37°C$. We name the following equations after the investigators the CRRSS model.

$$\dot{V} = [-g_{Na} m^2 h (V - V_{Na}) - g_L (V - V_L) + i_{st}]/c \quad \text{(CRRSS-1)}$$

$$\dot{m} = -(\alpha_m + \beta_m) \cdot m + \alpha_m \quad \text{(CRRSS-2)}$$

$$\dot{h} = -(\alpha_h + \beta_h) \cdot h + \alpha_h \quad \text{(CRRSS-3)}$$

with

$\alpha_m = \frac{97 + 0.363V}{1 + exp\left(\frac{31 - V}{5.3}\right)}$, $\beta_m = \frac{\alpha_m}{exp\left(\frac{V - 23.8}{4.17}\right)}$,

$\alpha_h = \frac{\beta_h}{exp\left(\frac{V - 5.5}{5}\right)}$, $\beta_h = \frac{15.6}{1 + exp\left(\frac{24 - V}{10}\right)}$,

$V_{rest} = -80[mV]$, $V_{Na} = 115[mV]$,

$V_L = 0.01[mV]$,

$g_{Na} = 1445[k\Omega^{-1} cm^{-2}]$, $g_L = 128[k\Omega^{-1} cm^{-2}]$,

$c = 2.5[\mu F/cm^2]$,

$m(0) = 0.003$, $h(0) = 0.75$

Comparison of equations for membrane dynamics for unmyelinated and myelinated parts of a neuron. V denotes membrane voltage; V_{Na}, V_K, V_L voltages across the membrane, caused by different (sodium, potassium, unspecific) ionic concentrations inside and outside of the neuron; g_{Na}, g_K, g_L maximum conductance of sodium, potassium and leakage per cm^2 of membrane; m, n, h probabilities for ionic membrane gating processes; α, β opening and closing rates for ionic channels.

Note the high conductances and the missed potassium currents in the myelinated mammalian node as described by the CRRSS model.

unmyelinated (parts of) neurons (Box 1) and the CRRSS model for the nodal membrane of myelinated mammalian nerve fibres (Box 2). Further discussions about membrane models can be found, e.g. in Ref. 9-11.

LUMPED CIRCUITS TO SIMULATE A NEURON

In order to simulate neural reactions a neuron should be parted into segments and every segment is represented by an electrical circuit. Fig. 1 shows a simplified neuron which only consists of two dendrites, a cell body and a piece of the nerve fibre. We have approximated all the structures by cylinders. The following calculations are done with the reduced voltage V with $V = V_{membrane} - V_{rest}$.

With the spatial neural model described below we can simulate both the influence of an artificial electric field (which causes an extracellular potential V_e) as well as natural situations ($V_e = 0$).

For simplicity we assume that segmentation length Δx is constant, every segment is approximated by a cylinder of diameter d_n and the center of the n-th segment represents the average inside potential $V_{i,n}$. The outside potential (caused by an applied electrical field) is $V_{e,n}$, and the voltage across the membrane is

$$V_n = V_{i,n} - V_{e,n} \quad (1)$$

The currents of the n-th segment are caused by capacity of membrane (C_n), membrane conductance (G_n)

and the axonal resistances to the neighboured segments (e.g. $R_n/2 + R_{n+1}/2$):

$$C_n \cdot \frac{dV_n}{dt} + G_n \cdot V_n + \frac{V_{i,n} - V_{i,n-1}}{R_n/2 + R_{n-1}/2} + \frac{V_{i,n} - V_{i,n+1}}{R_n/2 + R_{n+1}/2} = 0 \tag{2}$$

In order to see the influence of an extracellular electrical field we use (1) and obtain an ordinary differential equation for every segment:

$$C_n \cdot \frac{dV_n}{dt} = -G_n \cdot V_n - \left(\frac{V_n - V_{n-1}}{R_n/2 + R_{n-1}/2} + \frac{V_n - V_{n+1}}{R_n/2 + R_{n+1}/2} \right) - \left[\frac{V_{e,n} - V_{e,n-1}}{R_n/2 + R_{n-1}/2} + \frac{V_{e,n} - V_{e,n+1}}{R_n/2 + R_{n+1}/2} \right] \tag{3}$$

Note: In the case of V_e=0 (natural situation) the expression $[\cdots]$ in eqn. (3) vanishes. Knowing the specific capacity c and conductances g of the membrane as well as the resistivity of axoplasm ρ_i we can approximate the following variables which depend on the number N of layers of myelin which are wrapped around the cylinders with diameter d_n and length Δx:

$$C_n = \frac{\Delta x \cdot d_n \cdot \pi}{N} \cdot c \tag{4}$$

$$G_n = \frac{\Delta x \cdot d_n \cdot \pi}{N} \cdot g \tag{5}$$

$$R_n = \frac{\Delta x \cdot 4}{d^2 \pi} \cdot \rho_i \tag{6}$$

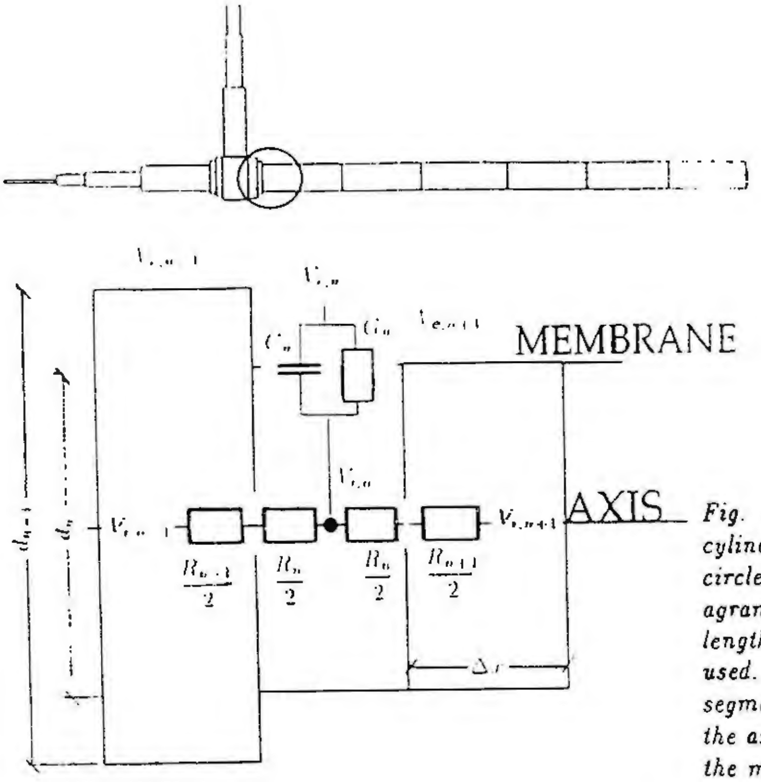


Fig. 1. A simple neuron is represented by cylinders with different diameters (top). The circle marks the place where the bottom diagram is situated. Segments with the same length Δx, but with different diameters are used. The electrical components of the n-th segment are the intracellular resistances (along the axis), and the capacity and conductance of the membrane.

Equations (3)-(6) are suited to simulate the dendrite and the cell body of a neuron. The situation in the nerve fibre (axon) is more complicated: The nodes of Ranvier only have a length L of $1\mu m$, and they are very short compared to the internode which is up to $2mm$ long. Even for qualitative results it is important to use a good model for the ionic currents, because they become the dominant driving force in the excitation process when threshold is reached. Secondly, even in the passive case (small distortion of the resting potential) the electric properties of the node are of importance, because it is not insulated by the many fatty layers of myelin. When the axon is segmented (e.g. 10 elements from node to node), the segments which include a node have to be calculated with Eqns. (4a) and (5a).

$$C_n = \frac{\Delta x \cdot d_n \cdot \pi}{N} \cdot c + L d_n \pi c \tag{4a}$$

$$G_n = \frac{\Delta x \cdot d_n \cdot \pi}{N} \cdot g + i_i L d\pi \qquad (5a)$$

For every segment the ionic current density i_i in (5a) can be calculated with the help of a set of differential equations as it was first described by Hodgkin and Huxley [5]. For the simulation of the mammalian node the CRRSS model of Box 2 [1, 14] or one of the models described in [9] or [11] should be used.

For subthreshold analysis we can approximate the membrane conductance of the node by constant g.

Of special interest is the case of a fibre with constant diameter. In this case we obtain Eqn. (7) from Eqns. (3) - (6), for segments with active membranes Eqn. (7a) has to be used.

$$c \cdot \frac{dV_n}{dt} = -g_n V_n + \frac{d \cdot N}{4(\Delta x)^2 \rho_i}(V_{n-1} - 2V_n + V_{n+1}) + \frac{d \cdot N}{4(\Delta x)^2 \rho_i}[V_{e,n-1} - 2V_{e,n} + V_{e,n+1}] \qquad (7)$$

$$c\left(1 + \frac{LN}{\Delta x}\right)\frac{dV_n}{dt} = -g_n V_n - i_i \frac{LN}{\Delta x} + \frac{dN}{4(\Delta x)^2 \rho_i}(V_{n-1} - 2V_n + V_{n+1}) + \frac{dN}{4(\Delta x)^2 \rho_i}[V_{e,n-1} - 2V_{e,n} + V_{e,n+1}] \qquad (7a)$$

The influence of the extracellularly applied electrical field on the n-th segment of a nerve or muscle fibre is represented by that term in Eqns. (7) and (7a) which contains V_e. It is called the activating function f and it is proportional to the second difference quotient of V_e, which is a function of the fibre's length coordinate:

$$f = \frac{dN}{4\rho_i} \frac{V_{e,n-1} - 2V_{e,n} + V_{e,n+1}}{(\Delta x)^2} \qquad (8)$$

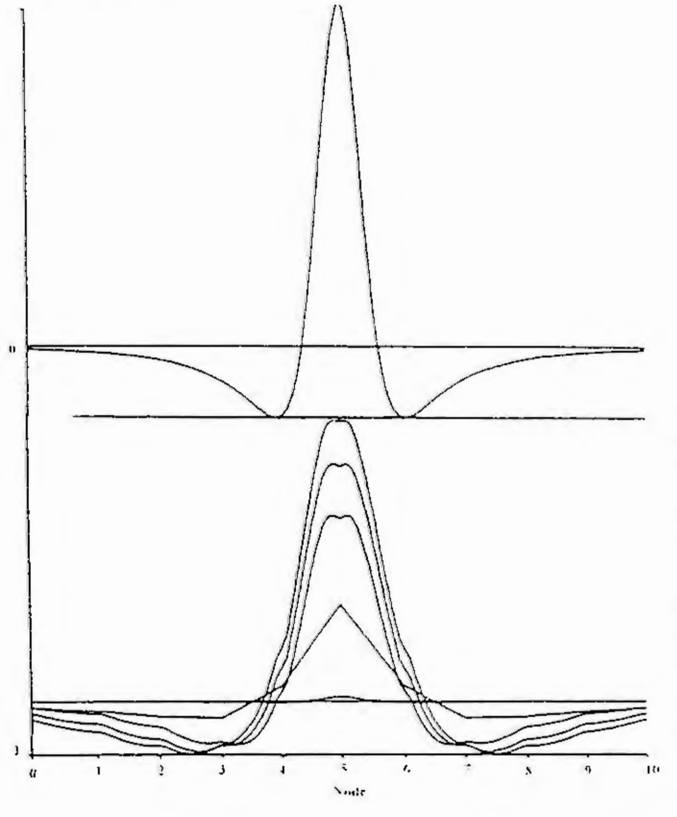


Fig. 2. Influence of myelin. Top panel: The excitation of an unmyelinated fibre at the end of a subthreshold negative 100μs current pulse is a picture of the activating function. Comparison of the computed subthreshold reaction of a 11 node axon with an unmyelinated fibre of same diameter shows that myelinated axons are much more excitable. The curve of the top panel is also included as the smallest one in the lower panel. The excitability increases considerably for myelinated fibres, even if it is assumed that the internode is a perfect insulator (second smallest result). Further increase is seen when N grows up (N=100, 200, 300).

In order to obtain an action potential it is necessary to reach threshold voltage in a segment with an active membrane. This is possible for segments where f is positive. The upper panel of Fig. 2 shows the effect of the activating function for a nonmyelinated fibre, computed with fine segmentation. A point electrode above the central part of the fibre produces an electric field by a negative current pulse. A positive electrode current would cause a result with inverse polarity, i.e. the fibre would be excited at two positions, however, this excitation is much weaker, which gives an explanation why cathodic stimulation is easier.

DISCUSSION

Of course, it is possible to simulate parts of the neuron (e.g. the nerve fibre) with even more accurate models, see e.g. [3]. However, it seems that deviations from experimental findings rather come from neglecting the electrical properties of myelin. With our preliminary results regarding subthreshold reactions we have obtained thresholds which only have about half of the value expected from the McNeal model, which was generally used. It must be expected that many computed results in the field of electrical stimulation have to be recalculated.

There exist only few data considering the ionic channel dynamics in different neural regions [6,7,12]. Especially in the input area there is a lack of information how to model the membrane behaviour. Nevertheless, many authors have obtained results with membrane models of the Hodgkin Huxley type. Different software is used to simulate single neurons and neural nets, see e.g. [4,13]. Some of the authors, e.g. [2], reduce the reaction of a single neuron to a combination of a simplified threshold process with a refractory function, and they do not look at the detailed membrane reactions because of the many elements involved.

REFERENCES

[1] S.Y. Chiu, J.M. Ritchie, R.B. Rogart and D. Stagg D. A quantitative description of membrane currents in rabbit myelinated nerve. J. Physiol. 292, 149-166, 1979

[2] W. Gerstner and J.L. van Hemmen. Associative memory in a network of 'spiking' neurons. Network 3, 139-164, 1992

[3] J.A. Halter and J.W. Clark. A distributed parameter model of the myelinated nerve fiber. J. Theor. Biol. 148, 345-382, 1991

[4] M. Hines. A program for simulation of nerve equations with branching connections. Int. J. Biomed. Comput. 24, 55-68, 1989

[5] A.L. Hodgkin and A.F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. J. Physiol. 117, 500-544, 1952

[6] C. Koch and I. Segev (Eds.) *Methods in neural modeling.* MIT Press Cambridge, MA 1989

[7] R.J. MacGregor. *Neural and brain modeling.* Academic Press. San Diego, CA 1987

[8] D.R. McNeal. Analysis of a model for excitation of myelinated nerve. IEEE Trans. Biomed. Eng. BME-23, 329-337, 1976

[9] F. Rattay. Simulation of artificial neural reactions produced with electric fields. (in press) Simulation Practice and Theory 1, 1993

[10] F. Rattay. Analysis of models for extracellular fiber stimulation. IEEE Trans. Biomed. Eng. BME-36, 676-682, 1989

[11] F. Rattay. *Electrical nerve stimulation.* Springer. Wien. 1990

[12] G.M. Shepherd. (Ed.) *The synaptic organization of the brain.* Oxford University Press. 1990

[13] R.G. Smith. NeuronC: a computational language for investigating functional architecture of neural circuits. J. neuroscience methods. 43, 83-108, 1992

[14] J.D. Sweeney, J.T. Mortimer and D. Durand. Modeling of mammalian myelinated nerve for functional neuromuscular electrostimulation. IEEE 9-th ann. conf. Eng. Med. Biol. Soc. Boston. 1577-1578, 1987

METRIC SPACES: A VERY VERSATILE FORMALIZATION FRAME FOR BIOLOGICAL DATA

Alessandro Giuliani (*) and Romualdo Benigni ($)

(*) Institute for Research on Senescence Sigma-Tau, Pomezia, Roma (ITALY)
($) Istituto Superiore di Sanita', Roma, (ITALY)

The standard way of representing experimental data is to write them on a matrix, which contains the experimental units on the rows and the measured variables on the columns. In some cases it is convenient to shift from this direct formalization to an intrinsic geometry representation of the data set. This is made by computing the distances between units in the multivariate space. The original unit/variable matrix is replaced by a symmetrical unit/unit matrix.

The distance concept was introduced in applied mathematics by the French mathemathician M. Frechet in the early 1900's, and derives from the intuitive concept of spatial distance. In a mathematical sense, a distance between two points A and B (d(AB)) has to respect the following constraints:

1) d(AB) is nonnegative and is zero only when A and B are coincident; 2) d(AB) is equal to d(BA); 3) d(AB) is never higher than the sum of d(AC) and d(CB).

This definition applies to the classical Euclidian distance, as well as to other types of distances like Minkowski, Manhattan, Lagrange, Pearson, Hamming etc... (5).

BIOLOGY AND DISTANCES

In biology, there are many situations in which the use of distance spaces is convenient: simultaneous analysis of variables at different level of definition, qualitative data, lack of reliable functional models... With the transformation into distance matrices, qualitative as well as mixed data can be expressed in terms of quantitative relationships, which can be treated with any mathematical algorithm suited for interval scales.

As an example, Benigni and Giuliani (1) have studied the relationships between mutagenicity short term assays. A number of assays were tested on a common sample of chemicals, and the responses were expressed as positive and negative. These data were transformed into Hamming distances between assays, and their relationships were studied by multivariate analysis. This permitted the selection of classes of assays most suitable for the assessment of the genotoxic risk to man.

Another example was provided by Sneath (9), which demonstrated that the distance spaces have the property of overcoming the problem of missing data. In his work he showed how it is possible to obtain a correct classification of a set of microorganisms by using their distances from a few "reference microorganisms". The problem of dealing with missing data is a problem of generalization almost identical to that faced by neural networks.

It should be also noted that some biological data are directly generated in the form of distances : for example, the genetic distances used by population biologists (6), the similarities between the DNA sequences (7), the results of some psychological tests (8), the variables used in conformational analysis of macromolecules (4).

EXPLORING A DISTANCE SPACE

Therefore it is useful to explore more in depth the common features of metric space representations. With this aim in mind, we performed a number of simulations in which data sets with different structures, defined in a two-dimensional frame, were transformed into distance spaces, and these spaces were analyzed by Principal Component Analysis (PCA) (2).

First, a two dimensional uniform distribution of points (N=100) was generated by random extraction of two variables X and Y. Since X and Y were uncorrelated, PCA produced two PCs, each explaining about 50% of total variability. On the other hand, the PCA of the correspondent Euclidian distance space produced four PCs with eigenvalues > 1. PC1 and PC2 were correlated with X and Y, whereas PC3 and PC4 (explaining together 20% of variance) represented new information with respect to the original variables. These two minor components represented the distance-from-center (PC3) and the distance-from-corners (PC4) features of the points. This result was confirmed by different simulations using subsets of the distance variables.

Small perturbations of the uniform distribution did not substantially affect the distance PC pattern. A remarkable perturbation has been obtained by adding to the uniform distribution further units, which were randomly sorted with the constraint of being localized in a limited area around the center of the data field. The PCA of X and Y extrinsic space again gave two PCs with almost equal explained variance. On the contrary, the PCA of the distance space reflected the strong departure from the previous distribution. Four PC's with eigenvalues > 1 were obtained with spatial features similar to those of the previous simulation. However, the relative rank of explained variance was changed: PC1 showed central simmetry (as the former PC3), thus reflecting the heavy density of points in the centre of the data space.

Other kinds of perturbations of the uniform distribution were reflected by different patterns of explained variability of the same set of four PCs.

DISTANCES AND NONLINEAR DISCRIMINATION

The PCs of the distance space act as nonlinear descriptors of the data space and can be usefully exploited for pattern recognition purposes. We analyzed the so-called "asymmetric structure": an "asymmetric structure" is a cluster of objects that belong to one class, embedded in a cloud of other objects belonging to another class. This case is typically not linearly separable. In fact, a linear discriminant analysis performed on the original X and Y variables failed to separate the two classes. On the contrary, the discriminant analysis, when applied to the PCs of the distance space, was able to separate the two groups.

The separation was made possible by the gain of information produced by the two additional (with respect to the dimensionality of the original space) components. These components provided a sort of holistic description of the entire data set, in terms of relationships among units.

DISTANCES AND NEURAL NETWORKS

The above pattern recognition problem was studied with a Back-Propagation (BP) neural network in order to investigate the relationships between the metric spaces approach and the neural networks. The BP, with three hidden units, found a solution to the "asymmetric structure", as expected. In order to compare the solutions given by BP and by the PCs of distance space, we analyzed the space of the activation values of the hidden units (see also 3). Each point of the data field was defined by three variables (hidden units activation values); this three dimensional space was analyzed with PCA. The analysis highlighted a major component (PC1), which explained 86% of the total variance. It should be noted that the PC1 scores discriminated perfectly the two classes from each other. A complete discrimination was also obtained when the PCs of the distance space were used as inputs of a linear discriminant analysis. Furthermore the PCs scores were highly correlated with the activation values of the hidden units. The PCs of the distance space were singled out in a totally unsupervised manner, whereas the BP procedure was by definition supervised; thus, the observed correlation could not trivially depend on the pattern recognition task but had to rely on some more fundamental basis.

CONCLUSIONS

In conclusion, the computation of distances between objects generates an intrinsic geometry, which, besides containing all the information explicitly present in the original space, also provides new information, which is based on the symmetry properties of the distance function. It should be stressed that the computation of distances and the derivation of PCs is an unsupervised process. The treatment of the data precedes any pattern recognition step: this allows the emerging structure to not be influenced by the specific problem, but to be a detailed description of the data field with new variables. Since these are "natural" views not driven by the problem to be solved, they are also important for the exploration of the scientific aspect of the problem, and not only for obtaining a formally best discrimination.

REFERENCES

(1) Benigni R. and A. Giuliani, A new insight into chemical mutagenesis by multivariate data analysis. J. Theor. Biol., 121 (1986) 477-486.

(2) Benigni R. and A. Giuliani, Analysis of distance matrices for studying data structures and separating classes. submitted.

(3) De Saint Laumer Y.J., M. Chastrette, and J. Devillers, Multilayer neural networks applied to structure-activity relationships. in J. Devillers and W. Karcher (Eds) Applied Multivariate analysis in SAR and environmental studies, Kluwer, Dordrecht, (1991) pp. 479-521.

(4) Karpen E.M., Tobias D.J., and Brooks C.L., Statistical clustering techniques for the analysis of a long molecular dynamics trajectory: analysis of 2.2-NS-trajectories of YPGDN. Biochemistry, 32 (1993) 412-420.

(5) Lee R.C.T., Clustering analysis and its applications. in J.T.Tou (Ed) Advances in information science. Plenum Press, New York, (1981) 168-292.

(6) Nei M., Analysis of Gene Diversity in Subdivided Populations. Proc. Natl. Acad. Sci. USA, 70 (1973) 3321-3323.

(7) Prince J.P., F. Loaiza-Figueroa, and S.D. Tanksley, Restriction fragment length polymorphism and genetic distance among Mexican accessions of Capsicum. Genome, 35 (1992) 726-732.

(8) Shepard R.W., Multidimensional scaling, tree fitting and clustering. Science, 210 (1980) 390-398.

(9) Sneath P.H.A., Distortions of taxonomic structure from incomplete data on a restricted set of reference strains. J. Gen. Microbiol., 129 (1983) 1045-1073.

AUTOMATION OF MODELLING OF MULTIENZYME SYSTEMS USING DATABANKS ON ENZYME AND METABOLIC PATHWAYS (DBEMP).

I.I. Goryanin, K.I. Serdyuk.
Institute of theoretical and experimental biophysics
Russian Acad. Sci. Poustchino Moscow region, 142292 Russia.

Abstract. The algorithm and original software allowing automate process of modelling systems of enzyme reaction were developed. Using examples with analytical solution were shown that developed method is correct and error is too few. Unlike published earlier the computer modelling of almost ideal relaxation oscillator were derived using new method and results obtained in dimensioned form.

1. INTRODUCTION

Todate factographical data banks store a lot of information about enzyme reactions and metabolic pathways. One of them Data Bank on Enzyme and Metabolic Pathways (DBEMP) [2] contains more than 6000 records with physical-chemical and kinetics properties of enzymes: stoichiometric and regulatory matrices, rate constants and etc. Quality and quantity analyses of multienzyme systems and automation of modelling metabolic pathways using information from DBEMP is one of actual task. The original DBsolve software has been created for this purposes.

2. ALGORITHMS

For analyzing stored data in DBEMP data bank, describing automatically enzyme reaction, finding steady state solutions, sensitivity and dynamic analysis the DBsolve program has been developed. It process data described as stoichiometric or regulatory matrices. The stoichiometric matrix contains information about each elementary reactions step characterized by rate constant. Reactions rates of each step will then be described in terms of mass actions kinetics. Alternatively the rate of each equation may described as net-flux equations or as any other biophysical low.

Steps:

1. Write down dependent reactions or steps, balance equations of each ligand or metabolite [1]. Set up conservation equations. Determine rate low.
2. Find the dependence of steady-state metabolic concentrations or steady state flux upon system parameters, sensitivity analysis using BEATLE algorithm. BEATLE algorithm is project for finding steady state solutions of ordinary differential equations (ODE's) system parameters. It uses an effective prediction-correction procedure for convergence to steady points on curve under consideration.
3. Find the time dependence of metabolite concentration using DEQAN algorithm. DEQAN algorithm is designed for solving initial value problem for sets of stiff ODE's. It deals with a

numerical method similar to the Kalahan method with variable number of Newton iterations.

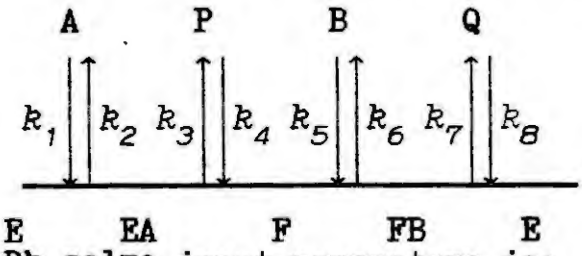
3. EXAMPLES OF IMPLEMENTATION.

3.1 Mechanism bi-bi ping-pong

First example is reaction catalyzed by enzyme transketolase (EC2.2.1.1.)[6]. Rate law of this reaction has an analytical solution [5].

Xu5P+Rib5P ↔ GraP+Sed7P

Mechanism of transketolase action is **bi-bi ping-pong:**



(Fig.1)

Db solve input parameters is:

1.stoichiometric matrix (automatically created and stored in **DPEMP**):

	A	B	P	Q	E	EA	F	FB
1	-1	0	0	0	-1	+1	0	0
2	+1	0	0	0	+1	-1	0	0
3	0	0	+1	0	0	-1	+1	0
4	0	0	-1	0	0	+1	-1	0
5	0	-1	0	0	0	0	-1	+1
6	0	+1	0	0	0	0	-1	+1
7	0	0	0	+1	+1	0	0	-1
8	0	0	0	-1	-1	0	0	+1

2.Default values of elementary steps rate constant k_i and initial concentrations of metabolites are obtained using **DBEMP** records other case values are defined by user.

Numerical experiment shows that time dependencies of metabolite concentrations obtained using DBsolve well fit to analytical formula [5].

$$\frac{d[P]}{dt}=\frac{(N_1[A][B]-N_2[P][Q])[E_t]}{(D_1[A]+D_2[B]+D_3[P]+D_4[Q]+D_5[A][B]+D_6[P][Q]+D_7[B][Q]+D_8[A][P])}$$

where

$N_1= k_1k_3k_5k_7$ $\qquad$ $N_2= k_2k_4k_6k_8$

$D_1 = k_1k_3(k_6+k_7)$ $\qquad$ $D_2=k_5k_7(k_2+k_3)$

$D_3 = k_2k_4(k_6+k_7)$ $\qquad$ $D_4=k_6k_8(k_2+k_3)$

$D_5 = k_1k_5(k_3+k_7)$ $\qquad$ $D_6=k_4k_8(k_2+k_6)$

$D_7 = k_5k_8(k_2+k_3)$ $\qquad$ $D_8=k_1k_4(k_6+k_7)$.

3.2 Model of relaxation oscillator based on covalent enzyme modificator.

Advantages of DBsolve may be demonstrated on enzyme reaction with more complex mechanism [3]. This reaction is a generator relaxation oscillations. For deriving analytical period and amplitude of oscillations one have to analyze system with 17 intermediates and 30 elementary steps . The DBsolve program allows to formalize this process and to obtain result in dimensioned form. The result of comparison between analytical and numerical obtained data are shown in Table 1.

Table 1.

External parameter value	Numerical solution		Analytical formulas		Deviation %	
	A $\times 10^{-2}$	T_0	A $\times 10^{-2}$	T_0	A	T
1.1	1.90	1.635	2.01	1.689	5.41	3.4
1.2	1.75	1.505	1.85	1.548	5.41	2.0
1.3	1.62	1.396	1.68	1.429	3.62	2.4
1.4	1.51	1.302	1.58	1.327	3.82	1.4
1.5	1.41	1.222	1.48	1.238	4.71	1.0
1.6	1.33	1.152	1.38	1.161	3.62	0.4
1.7	1.25	1.091	1.30	1.093	3.79	0.4
1.8	1.19	1.037	1.24	1.032	4.03	0.6
1.9	1.13	0.989	1.17	0.978	3.42	1.3

Where A - amplitude of oscillations, T- period.

4. RESULTS

The mathematic methods using in DBsolve are similar to methods used in METAMOD [4]. However DBsolve handles aspect such as dynamic behavior metabolite and stability of steady state. The essential advantage of DBsolve is processing of stiff ODE's, which allow to investigate systems with complex non-linearities, and wide range of parameters values. It should be noted that there are some problems with numerical accuracy of solution complicated metabolic processes (entire glycolytic system, Krebs circle). It is authors opinion that complex systems solution have to divide on iterations steps. On every iteration step systems are distributed over the rate levels. Thus problem transformed to analysis of algebraic-differential equation.

5. REFERENCES

1. Besdenegnyh, A.A., Engineering methods of deriving rate law equations and calculating of kinetic constants. Himija, Leningrad, 1973
2. Selkov E.E., Goryanin I.I.,Kaimachnicov N.P., Shevelev E.L., Yunus I.A. Studia biophysica, 129, (1989), 155-165.
3. Goryanin, I.I., Selkov, E.E, Serdyuk, K.I.,.Molecular Biology, 26 (1992), 404-417.
4. Hofmeyer, J.H.S., van der Merwe, K.J., Cabios, 2,.(1986), 243-249.
5. Indge,K. I., Childs, R.E., Biochem. J., 155, (1976), 567-570.
6 Mcintyre, L.M., Torburn,D.R., Bubb, W. A. Eur.J.Biochem.,180 (1989), 399-420.

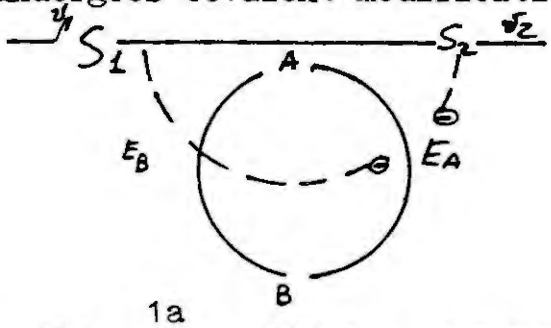
MATHEMATICAL MODELS OF AN ALMOST IDEAL BIOCHEMICAL OSCILLATORS BASED ON COVALENT ENZYME MODIFICATION

K.I. SERDYUK, I.I. GORYANIN, E.E. SELKOV.
Institute of theoretical and experimental biophysics
Russian Acad. Sci. Poustchino Moscow region, 142292 Russia.

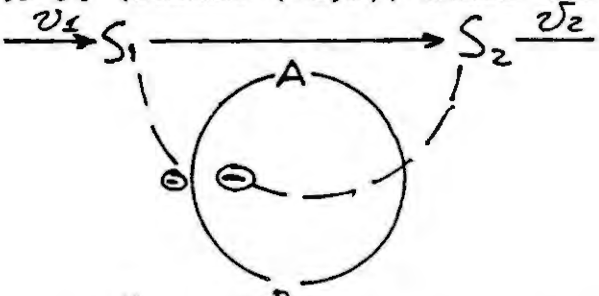
Abstract. A mathematical models for the generation of relaxation autooscillations in an open reaction $\longrightarrow S_1 \longrightarrow S_2 \longrightarrow$ was derived and analyzed. The asymptotical formulas describing the oscillation amplitude and period of the quasi stationary reaction $S_1 \longrightarrow S_2$ and taking into account all the ligand-enzyme complexes E, E_A и E_B where derived. A numerical analysis of models gives a good agreement with analytically obtained values.

1. INTRODUCTION

The examined reaction $\longrightarrow S_1 \longrightarrow S_2 \longrightarrow$ is open and enzyme E(A,B) undergoes covalent modification [1,3-5] (scheme (1a,b)) under the action of modifying enzymes E_A and E_B. As a result, the catalitically active A-form is transformed by the enzyme E_A to the catalitically inactive B-form and the B-form is reactivated to the A-form by the enzyme E_B. It was assumed that the substrate S_1 and the product S_2 inhibit competitively inactivating enzyme E_A in case (1a). In other case (1b) $S_1$and S_2competitively inhibit activating enzyme E_B. Case (1b) look through more detailed in this work as so case (1a) were investigated earlier [6,8].



1a



1b

2. KINETIC MODELS

In kinetic models were assumed that: 1.The enzyme E(A,B) have two different binding center: catalytical, which bind with substrate S_1 and allosteric, which modify by modifying enzyme E_A and E_B what lead to cycle variation of two form $A \rightleftarrows B$; 2. Events in catalytical center of A and B-form is independent from binding with enzyme E_A и E_B but A and B-form have a differ affinity for S_1 /nd differ catalitically effectivity; 3.B-form catalitically inactive; 4.Substrate S_1 and product S_2 of main reaction $S_1 \rightarrow S_2$ are allosteric inhibitors of enzyme-modifier E_A and competitive to one another and not competitive to molecules A и S_1A, which bind with catalytical center of E_A, in case (1a). In case (1b) $S_1$and S_2 competitively inhibit modifying enzyme- E_B and are not competitive to molecules B and S_1B which bind with E_B. 5.Systems are open relative to S_1 and S_2 and are closed relative for enzyme forms E(A,B), E_A, E_B. The rate of exchange between S_1, S_2 and environment is: $v_1 = V_1 - k_1S_1, v_2 = -V_2 + k_2S_2,$ where v_1- rate of influx S_1, V_1- rate of influx when S_1= 0, v_2 - rate of outflow S_2, V_2 - rate of influx S_2, k_1 и k_2 - exchange rate constants. The accepted designation are following: E_0, E_{A0} и E_{B0} - total concentration of E(A,B), E_A and E_B. K_m, K'_m - Michaelis constant for S_1 of A and B-form. K_A- Michaelis constant for E_A; K_{i1}, K_{i2}, K_B - inhibition constant of enzyme E_B and Michaelis constant of E_B for B'. w и w' - productions rate of S_2 by A and B-form, r - relatives maximal rate for E_B, $k_{\pm1}$, $k'_{\pm1}$, k_{+2}, k'_{+2}, $a_{\pm1}$, $b_{\pm1}$, a_{+2}, b_{+2}-rate constant of elementary step. t- dimensioned time.

3. MATHEMATICAL MODELS

For mathematical description is convenient to use variables and parameters:

$$\sigma_1 = \frac{S_1}{K_{i1}},\ \sigma_2 = \frac{S_2}{K_{i2}},\quad \alpha = \frac{A'}{E_0},\ \beta = \frac{B'}{E_0},\quad \nu = \omega + \omega',\ \omega = \frac{w}{k_{+2}E_0},$$

$$\omega' = \frac{w'}{k_{+2}E_0},\ \tau = \frac{k_{+2}E_0}{K_{i2}}t,\ \nu_{1m} = \frac{V_1}{k_{+2}\ E_0},\ \nu_{2m} = \frac{V_2}{k_{+2}\ E_0},\ æ_1 = \frac{k_1K_{i1}}{k_{+2}E_0},\ \varepsilon_2 = \frac{K_{i2}}{K_{i1}},$$

$$æ_2 = \frac{k_2\ K_{i2}}{k_{+2}E_0},\quad æ_m = \frac{K_m}{K_{i1}},\ æ'_m = \frac{K'_m}{K_{i1}},\ æ_A = \frac{K_A}{E_0},\ æ_B = \frac{K_B}{E_0},\ \varepsilon = \frac{k'_{+2}}{k_{+2}}.$$

System (1b) in undimensioned time τ may be described by the system of material balance equations:

$$\left.\begin{aligned} \varepsilon_2\frac{d\sigma_1}{d\tau} &= \nu_{1m} - æ_1\sigma_1 - \nu, \\ \frac{d\sigma_2}{d\tau} &= \nu - æ_2\sigma_1 + \nu_{2m}, \end{aligned}\right\}\quad \begin{aligned} &\text{where } \nu = \omega + \omega', \qquad (2) \\ &\omega = \frac{\sigma_1}{æ_m + \sigma_1}\left(\alpha + \varepsilon_A\frac{\alpha}{æ_A + \alpha}\right), \end{aligned}$$

$$\alpha + \beta + \varepsilon_A\frac{\alpha}{æ_A + \alpha} + \varepsilon_B\frac{\beta}{æ_B + \beta} = 1,\ \omega' = \varepsilon\frac{\sigma_1}{æ'_m + \sigma_1}\left(\beta + \varepsilon_B\frac{\beta}{æ_B + \beta}\right),$$

$$\frac{\alpha}{\ae_m + \alpha} - r\,\frac{\beta}{(\ae_A + \alpha)(1 + \sigma_1 + \sigma_2)} = 0.$$

Where designated: σ_1, σ_2, α и β - undimensioned concentration of S_1, S_2, A' и B'; ν - undimensioned rate of transformation $S_1 \rightarrow S_2$, catalyzed by enzyme E(A,B); ω и ω' - analogous for A and B separately; ν_{1m} и ν_{2m} - influx maximal rate for S_1 и S_2; $\ae_1$ и $\ae_2$ - exchange rate constant S_1 и S_2; $\ae_m$, $\ae'_m$, $\ae_A$, $\ae_B$ - undimensioned Michaelis constant for A, B, E_A, E_B; ε - relative activity of A-form; ε_A and ε_B - relative concentration of enzyme-modificators E_A and E_B, ε_2- relative constant of inhibition E_B by S_2, $\varepsilon_A = E_{A0}/E_0$, $\varepsilon_B = E_{B0}/E_0$. If take in to account that relative concentration of enzyme E_A и E_B is too few (ε_A, $\varepsilon_B \rightarrow 0$), formula (2) will transform to:

$$\left.\begin{aligned} \varepsilon_2 \frac{d\sigma_1}{dt} &= \nu_{1m} - \ae_1\sigma_1 - \nu, \\ \frac{d\sigma_2}{dt} &= \nu - \ae_2\sigma_2 + \nu_{2m}, \end{aligned}\right\} \quad \frac{\alpha}{\ae_m + \alpha} - r\,\frac{1 - \alpha}{(\ae_A + \alpha)(1 + \sigma_1 + \sigma_2)} = 0. \qquad (3)$$

where $$\nu = \frac{\sigma_1}{\ae_m + \sigma_1}\,\alpha + \varepsilon\,\frac{\sigma_1}{\ae'_m + \sigma_1}\,(1-\alpha),$$

4. ANALYTIC FORMULAS AND NUMERICAL SOLUTION

Systems (1a,b) can generate almost ideal relaxation oscillations (Fig 1a) of variable σ_1 , rectangular oscillation of σ_2 in case(1a) and inversely in case (1b) (Fig 1b). A necessary conditions for existence of oscillation is smallness of parameters $\ae_m$, $\ae_A$, $\ae_B$, ε, ε_2 and concentration must be $S_1S_2 >> E_0 >> E_{A0}E_{B0}$. Model (3) come to degeneration model of first order with rupturing right part if $\varepsilon_2 \longrightarrow 0$.

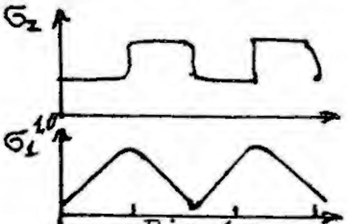


Fig 1a

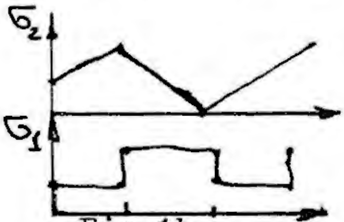


Fig 1b

$$\frac{d\sigma_2}{d\tau} = \nu_{2m} - \ae_2\sigma_2 - \tilde{\nu}(\sigma_2) \qquad (4)$$

In which $\tilde{\nu}(\sigma_2)$ - quasi-stationary rate of reaction $S_1 \rightarrow S_2$. Relaxation oscillation in model (4) , as so as in initial model (3), appear because of hysteresis of function $\tilde{\nu}(\sigma_2)$. Such oscillation on phase plane of variables $(\sigma_2, \tilde{\nu})$ fit to rupturing limit cycle C shown on Fig 2a (for 1a) and Fig 2b for 1b. Using

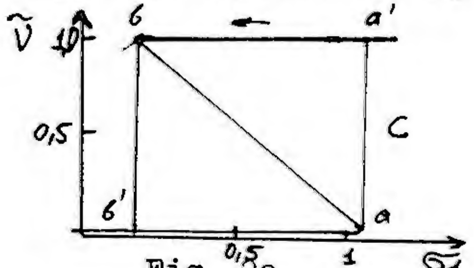


Fig 2a

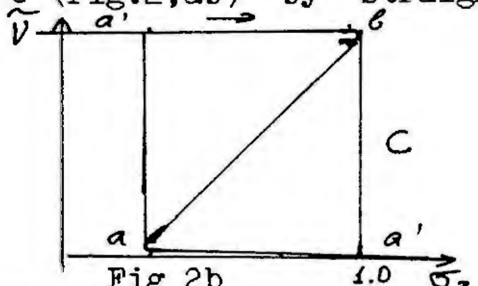


Fig 2b

derived earlier [8] point of minimum and maximum we approximate district of slow moving on limit cycle *C* (Fig.2,ab) by straight lines *ab'* и *ba'*, come to $\ae_A$, $\ae_B$, $\ae_m \longrightarrow 0$ we obtain expression for the period of oscillation:

$$\tau_{0a} = \frac{1}{\ae_2(1 + \varepsilon - \nu_{1m})(\nu_{1m} - \varepsilon)}, \quad \tau_{0b} = \frac{1}{\ae_1(1 + \varepsilon - \nu_{1m})(\nu_{1m} - \varepsilon)} . \quad (5)$$

corresponding for system 1a and 1b. For estimation of error, which we obtain using asymptotical formulas for determination of period in model (3) the control numerical integration were derived using original **DBsolve** software [2]. Both case (1a) and (1b) error not exceed some percents.

5. RESULTS

The small error between analytically and numerical obtained results demonstrate high precision of asymptotical formulas that allow to change a difficult operation of numerical integration on calculation using analytical formulas. Derived asymptotical formulas may be used for great number of equivalents reaction. For example reaction describing system with multyloops regulation as so carbon metabolism [7]. And so asymptotical formulas may be used in theoretical analysis of mechanism dynamic organization of polyenzyme systems.

6. REFERENCES

1. Chock P.B., Ree S.G., Stadtman E.R. Annu. Rev. Biochem, 49, (1980), 813-819.
2. Goryanin I.I, Automation of modelling of multienzyme sistems using databanks on enzyme and metabolic pathways (DBEMP). This volume.
3. Hers H.G., Biochem. Soc Trans. 12, (1984), 729.
4. Holzer H., Duntze W. Annu. Rew. Biochem., 40, (1971), 345-374.
5. Pilkis S.J., Chrisman T., Burgress B., McGrane M., Colosia A., Pilkis Jo, Thomas H.C., and M. Raafat El- Magharabi. Adv. Enzyme Regul. 21, (1983), 147.
6. Selkov E.E., Goryanin I.I., Molecular Biology, 26, (1992), 1550 - 1562.
7. Selkov E.E., Goryanin I.I.,Kaimachnicov N.P., Shevelev E.L., Yunus I.A. Studia biophysica, 129, (1989), 155-165.
8. Selkov E.E., Goryanin, I.I., Serdyuk, K.I.,. 26 (1992), 404-417.

MODELING AND COMPUTER SIMULATION OF HISTAMINE PHARMACOKINETICS

A.Mrhar[1],R.Karba[2],T.Irman-Florjanc[3],S.Primožič[1],F.Erjavec[3]

[1]Faculty of Natural Sciences and Technology,Dept.of Pharmacy
[2]Faculty of Electrical and Computer Engineering
[3]Medical Faculty, Dept. of Pharmacology

Abstract. Histamine plays an important role in physiologic and many pathophysiologic conditions. One of the main products of histamine metabolism is tele-methylhistamine and the second important pathway of histamine biotransformation is oxidative deamination. On the basis of plasma concentrations, obtained after intravenous doses of histamine and methylhistamine to the rats in two separate groups of experiments, a four-compartment model was developed by the aid of anolog-hybrid simulation. The results showed that kinetics of tele-methylhistamine and histamine are comparable as well as kinetics of exogenous and endogenous tele-methylhistamine. It was also found out that demethylation of tele-methylhistamine and/or displacement of histamine from mast cells occurs only in the case of very high plasma levels of methylhistamine, whereas the rate of hystamine metabolic pathways other than methylation are increased after injection of histamine. The results confirmed that processes of histamine metabolism are adaptable.

1. INTRODUCTION

It is well known that histamine (Hi) plays an important role in physiologic and many pathophysiologic conditions. It has been also established that one of the main products of Hi metabolism is tele-methylhistamine (M), which distributes as rapidly as Hi itself and undergoes further metabolic reactions. The second important pathway of Hi biotransformation is oxidative deamination (4). Several studies indicated that alternative pathways could provide a basis for an adaptable pharmacokinetic system.The aim of the present study is to find out 1) whether the kinetics of Hi is comparable to the kinetics of M, 2) whether the kinetics of elimination of exogenous M is comparable to the elimination kinetics of endogenous M and 3) whether increased plasma levels of M exert an influence on plasma levels of Hi.

2. METHODS

Since the knowledge of pharmacokinetic properties of Hi and M is insufficient we carried out an in vivo study on rats to obtain basic pharmacokinetic parameters by method of stripping (6) and to develop compartmental model by computer simulation (5). Rats received an intravenous dose of Hi and M in two separate groups of experiments. Blood samples from carotid artery were collected at specified intervals of time. Plasma Hi and M were assayed by HPLC method (2).

3. RESULTS AND DISCUSSION

The results obtained by the method of stripping showed that the two-compartment model describes plasma profiles of both amines satisfactorily. Two exponential terms were identified from Hi profile and three from M profile after injection of Hi and vice-versa after injection of M.

The method failed at the moment of coupling the results obtained in two separate experiments. By the aid of analog hybrid simulation (hybrid computer EAI 2000) a four-compartment model (Figure 1) was developed by linking two-compartment model of Hi with two-compartment model of M. The procedure with an adaptive model was used for verification of the model structure and identification of its parameters (3).

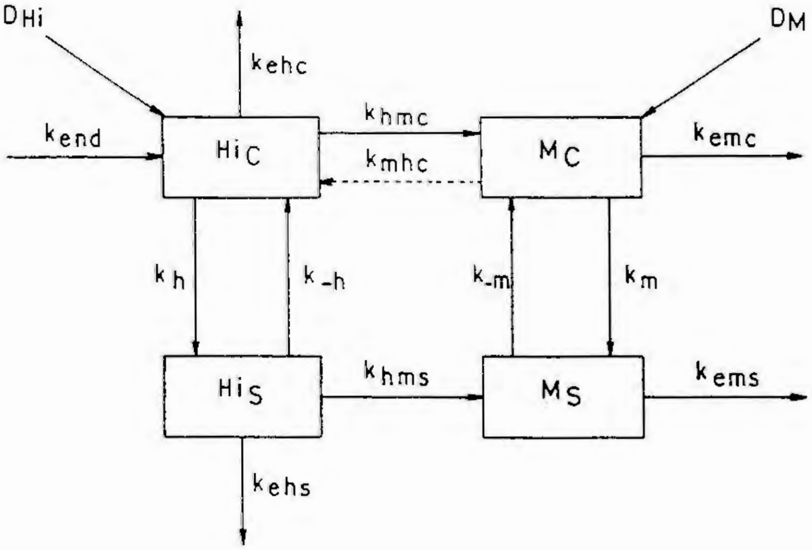


Figure 1: Four-compartment model describing pharmacokinetics/metabolism of Hi and M. Subscripts C and S refer to central and peripheral compartment, k_i - first order rate constant where subscript i means: hmc/mhc - methylation/demethylation in C, hms - methylation in S, h and -h distribution of Hi, m and -m - distribution of M, emc and ems - further metabolic reactions of M in C and S, ehc and ehs - other metabolic reactions of Hi in C and S, end - formation of endogenous Hi, D_{Hi} and D_M - intravenous dose of Hi and M, respectively.

The results of identification procedure on hybrid computer are given in Figure 2. Identified parameters resulted from the curve-fitting procedure are listed in Table 1.

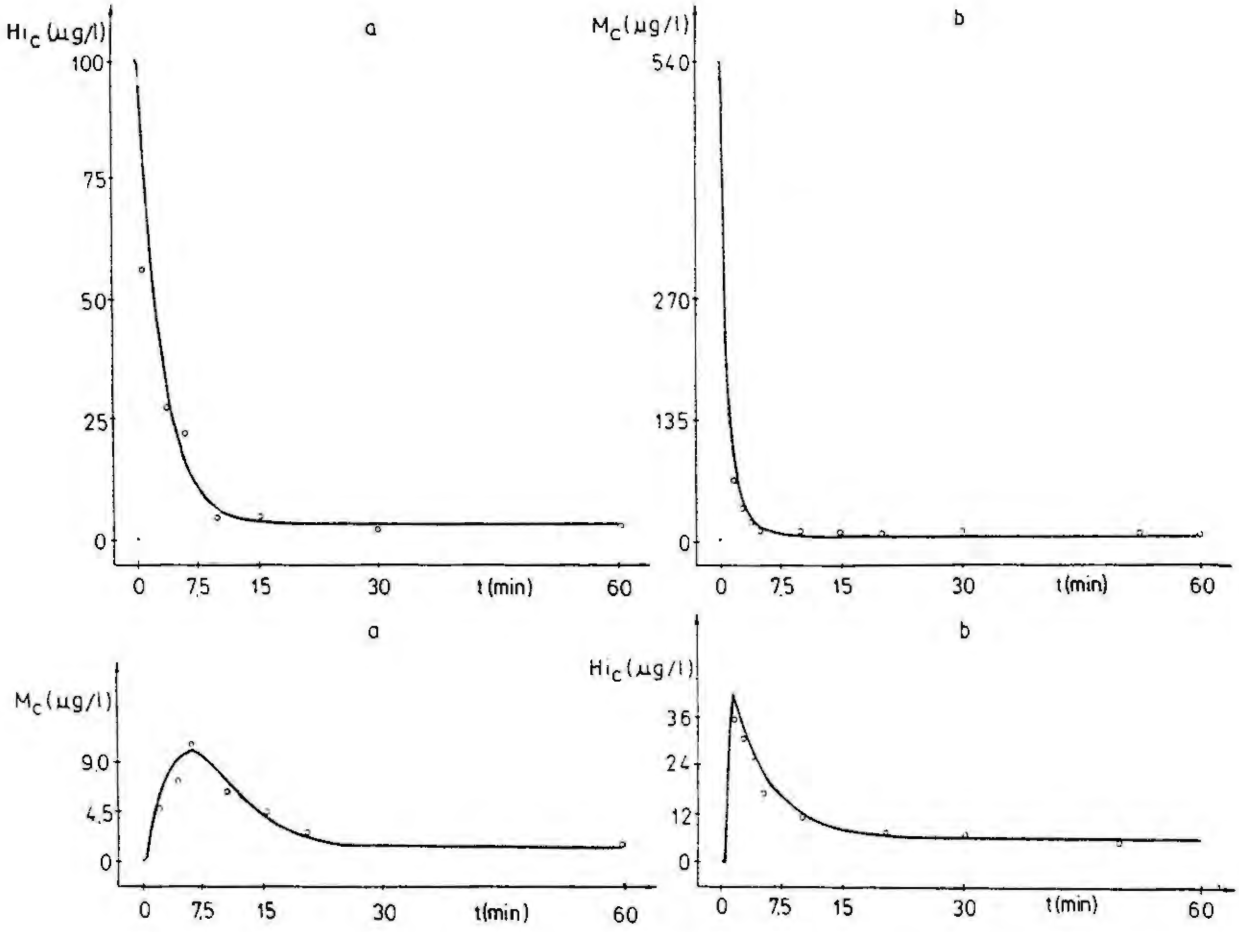


Figure 2: Levels of Hi and M in central compartment.
Curve -model response, points - experimental data
a - after injection of Hi;b - after injection of M

Table 1: Identified pharmacokinetic parameters of Hi

	Injected amine	
Parameter (min^{-1})	Histamine	Methylhistamine
k_{end}	0,01	0,0093
k_{ehc}	0,3	0,134
k_{hmc}	0,013	0,028
k_{ehs}	0,1	0,1
k_h	0,05	0,05
k_{-h}	0,08	0,08
k_{ems}	0,15	0,15
k_{emc}	0,32	0,014
k_m	0,65	0,65
k_{-m}	1,00	1,00
k_{hms}	0,22	0,22
k_{mhc}	-	0,78

Excellent fitting in both central compartments was obtained independently of injected amine. Moreover, majority of model parameters kept their values indicating the effectiveness of the approach for studying histamine pharmacokinetic pattern. It is necessary to point out that three model parameters posessed different values regarding the injected amine. Our results indicate that the kinetics of M is very similar to the kinetics of Hi. The same can be considered also for the kinetics of exogenous and endogenous M. On the basis of obtained results we additionaly found out that demethylation of M and/or displacement of Hi by M (1) from mast cells occurs only in the case of very high plasma levels of M whereas the rate of Hi metabolic pathways other than methylation are increased after injection of Hi. These findings confirm the hypothesis that mechanisms of Hi metabolism are actualy adaptable. It is felt that obtained results will contribute substantialy to the knowledge of the fate of Hi in the body and will elucidate the convenience of Hi and/or M determination in plasma in clinical practice.

4. REFERENCES

(1) Erjavec, F., Tele-methylhistamine as histamine releasing agent in the rat peritoneal mast cells, IUPHAR 9th Congress of Pharmacology, London, Abstracts, (1984), 1068 P.

(2) Irman Florjanc, T., Erjavec, F., Kinetics of histamine and tele-methylhistamine elimination from rat plasma. Agents Actions, 38 (1993), 76-78.

(3) Mrhar, A., Karba, R., Drinovec, J., Primožič, S., Varl, J., Bren, A. F., Kozjek, F., Computer simulation of ciprofloxacin pharmacokinetics in patients on CAPD, Int. J. Artif. Organs, 13 (1990), 169-175.

(4) Schayer, R.W., Cooper, J. A. D., Metabolism C^{14}-histamine in man. J. Appl. Physiol., 9 (1956), 481.

(5) Wartak, J., Clinical Pharmacokinetics, Praeger Publishers, New York, 1983.

(6) Welling, P. G., Pharmacokinetics (Processes and Mathematics), American Chemical Society, Washington (DC), 1986.

MATHEMATICAL MODELLING OF TUMOR GROWTH IN MICE FOLLOWING ELECTROTHERAPY AND BLEOMYCIN TREATMENT

Damijan MIKLAVČIČ, Tomaž JARM and Rihard KARBA
University of Ljubljana, Faculty of Electrical and Computer Engineering
Tržaška 25, 61000 Ljubljana, SLOVENIA

Abstract
A mathematical model was developed to describe subcutaneous growth of tumors (fibrosarcoma SA-1) in A/J mice under different therapeutical conditions. The Gompertz model, which turned out to be the most suitable model of undisturbed tumor growth in our case, was modified in a way that allowed the introduction of electrotherapy by direct electric current and bleomycin influences on tumor growth.

1. INTRODUCTION

Subcutaneous solid tumors in mice were treated by means of locally applied electrotherapy (ET) by direct current and intravenously administered bleomycin. In the study reported in detail previously [4] subcutaneously grown solid tumors were treated by either single shot i) intravenous injection of 250 μg of bleomycin into the tail vein; ii) electrotherapy of one hour duration, direct current 0.6 mA; or iii) combination of both, after the tumors reached 30-40 mm^3. In order to asses the effectiveness of single and combined treatments, tumor volumes were determined by measuring three tumor mean diameters (a, b and c) and by calculating tumor volume according to formula $V=\pi abc/6$. The results were compared to tumor growth in control group where electrodes were placed subcutaneously near the tumors and physiological saline was injected in the tail vein exactly as in previously described experimental groups. Tumor growth data thus obtained are presented in Figure 1. The electrotherapy as single treatment significantly delayed tumor growth in comparison to control group, whereas bleomycin therapy had only moderate effect on tumor growth. The tumor growth delay was calculated as a difference of mean tumor doubling times in therapy and control experimental groups. Tumor

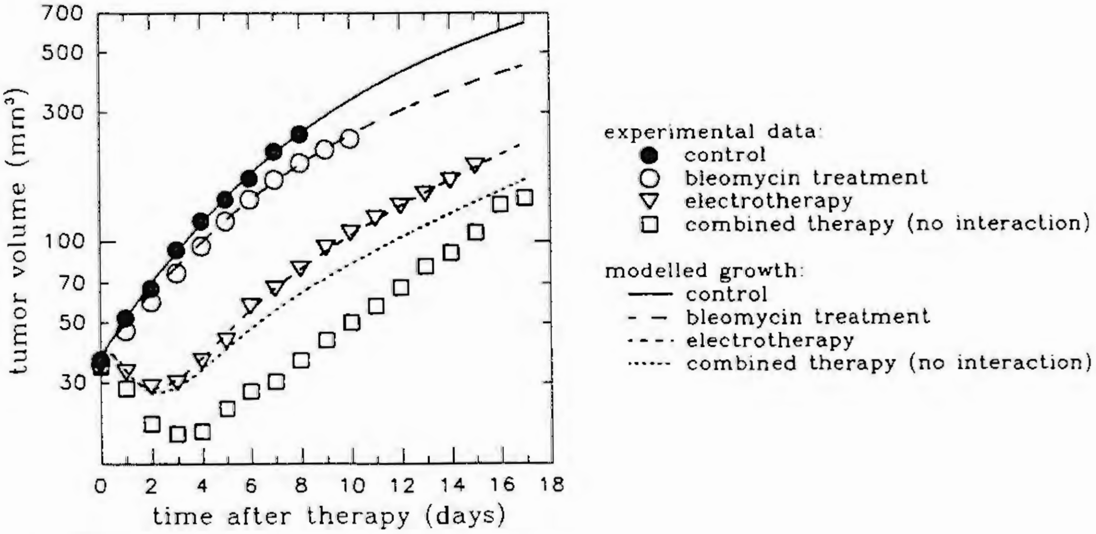


Figure 1: Experimental data and modelled growth curves

doubling time was determined for each individual tumor as the time needed to double the initial tumor volume. The tumor growth delay was 6.7±1.2 days (ν=15) (AM±STD (degrees of freedom)) in electrotherapy group and 0.5±0.2 days (ν=12) in group subjected to bleomycin treatment. When both treatments were combined the tumor growth delay observed was 10.8±1.9 days (ν=16), i.e. the effects were more than additive.

The objective of the study was to model tumor growth after different treatments performed, and to extent the pharmacokinetic model of bleomycin to the pharmacodynamic model, i.e. to asses tumor growth retardation based on the bleomycin pharmacokinetic model. Coupled model of both treatments should produce an additive effect of both single treatments performed.

2. DEVELOPMENT OF THE MODEL

The development of a model required four major steps. First the most convenient model for undisturbed tumor growth was sought, which served as a basis for further work. In next steps this model was modified by introducing the parts which reflected the influences of both therapies.

2.1. Undisturbed (control) tumor growth

Four widely used growth equations were put under investigation [5,7]. These were exponential, Gompertz, Bertalanffy and Verhulst (logistic) equation (equations 1,2,3 and 4). In all of them V represented tumor size (volume) and V_0 was its initial size at the beginning of observation. The values of the parameters in the equations were determined upon fitting them to actual growth data.

$$\frac{dV}{dt} = \lambda V, \quad V(0) = V_0 \qquad (1) \qquad\qquad \frac{dV}{dt} = V\left(\alpha_{V_0} - \beta \ln \frac{V}{V_0}\right), \quad V(0) = V_0 \qquad (2)$$

$$\frac{dV}{dt} = \eta V^{\frac{2}{3}} - \mu V, \quad V(0) = V_0 \qquad (3) \qquad\qquad \frac{dV}{dt} = aV - bV^2, \quad V(0) = V_0 \qquad (4)$$

In the comparative study we fitted them to the average tumor growth data of the control group and also to the growth data of individual tumors in this group. Several numerical criteria were used to estimate model suitability, the most important being goodness of fit and predictability of the model [6]. The goodness-of-fit parameter r^2 was calculated according to equation 5 [2] where x_i and $\hat{x}_i$ were the i-th measured and estimated size of the tumor respectively. The fit was proclaimed to be satisfactory when $r^2>0.98$ (higher r^2 meaning better fit). The predictability was assessed by fitting a model to first m out of n measured experimental points and by calculating the prediction error for the rest of n-m points.

The exponential equation did not meet our requirements at all. The rest of the models gave similar results with the Gompertz model being moderately more suitable. In a few individual cases when either the Bertalanffy or the logistic equation gave the best fit or prediction, the Gompertz one was always very close to them. When fitted to some of the individual tumor growth data the optimisation produced degenerated versions of the Bertalanffy or the logistic model since their parameters tended to zero or even negative values. Reaching global minimum of error function during optimisation of these two models also caused much more trouble than optimisation of the Gompertz model which was thus chosen for further modelling. The result of fitting of the Gompertz model to the control group data is shown in Figure 1.

$$1 - r^2 = \frac{\sum_{i=1}^{n} (\hat{x}_i - x_i)^2}{\sum_{i=1}^{n} x_i^2 - \frac{1}{n}\left(\sum_{i=1}^{n} x_i\right)^2} \qquad (5)$$

2.2. Bleomycin pharmacokinetics

Bleomycin concentration in the blood plasma decreases biexponentially after intravenous bolus injection. This can be described by a linear non physiological two-compartment model [1]. The central compartment represents the blood and all with bleomycin well perfused tissues. The peripheral compartment represents the rest of the body.

Since our goal for the future was to find the level of interaction between electrotherapy and therapy with bleomycin, we split the peripheral compartment into two parts, the first for the tumor (T) and the second for the rest of the periphery (P), as shown in Figure 2. Bleomycin is eliminated from the body mostly by renal elimination which is modelled by secretion from the central (C) compartment [1]. Figure 3 shows quantity/time curves of bleomycin in central and tumor compartments.

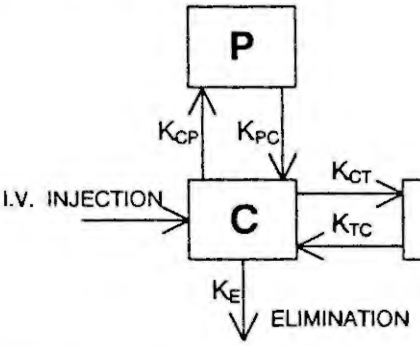


Figure 2: Three-compartment model

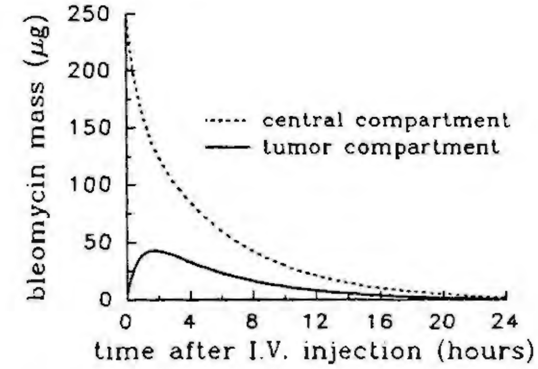


Figure 3: Quantity of bleomycin in central and tumor compartment

2.3. Introducing bleomycin treatment

To exert their cytotoxic properties the bleomycin molecules have to enter the tumor cells, where they damage cellular DNA [3]. It seemed highly inappropriate to introduce the concentration of bleomycin directly into the growth model since there had to be some delay between bleomycin entering the cells and the changing of tumor growth rate. Therefore a new quantity was introduced into our model, namely the influence of bleomycin therapy ($BT(t)$), which was related to the bleomycin concentration through the first order delay. Equation 6 describes tumor growth after treatment with bleomycin. Its optimised form gives the fit to the experimental data of the group of mice treated with bleomycin as shown in Figure 1.

$$\frac{dV}{dt} = V\left(\alpha_{V_0} - \beta \ln \frac{V}{V_0} - \gamma \, BT(t) \right), \quad V(0) = V_0 \tag{6}$$

2.4. Introducing electrotherapy

The influences of electric current were introduced into the Gompertz model in a similar way as those of bleomycin. Again it was necessary to use delay between the current itself and its visual effect, that is altered rate of tumor growth. The modelling showed however, that at least two components had to be added to the original form of the Gompertz equation in order to obtain a satisfactory fit to the experimental data of the third group of mice. These two components ($ET_1(t)$ and $ET_2(t)$) had the third and the second order delay respectively with reference to electric current with $ET_1(t)$ and $ET_2(t)$ having pronounced influence in the late and in the early stage of observation. This suggests at least two important mechanisms involved in antitumor effectiveness of weak DC electric current. Equation 7 presents the modified

Gompertz model of tumor growth after electrotherapy with η_1 and η_2 being constants. The result of the optimised model is shown in figure 1.

$$\frac{dV}{dt} = V\left(\alpha_{V_0} - \beta \ln \frac{V}{V_0} - \eta_1 ET_1(t) - \eta_2 ET_2(t)\right), \quad V(0) = V_0 \tag{7}$$

2.5. Modelling combined treatment

By joining the modified models of tumor growth after single therapies a model of tumor growth after combined therapy was obtained based on assumption that there was no interaction between the therapies. Figure 1 clearly shows that this additive coupling of both therapies in a model does not fit to the experimental data of the group of mice submitted to both therapies where tumor retardation was far more pronounced than suggested by our model.

3. RESULTS

The Gompertz model was used for easiness of introduction of therapeutic effects and convergence in optimisation experiments. The data on tumor growth in control group was used to determine parameters of Gompertz model V_0, α and β. For both single treatments, electrotherapy and bleomycin, extended Gompertz equation was used. In the case of electrotherapy a two component effect with different time constants and second order delay of action had to be introduced in order to obtain satisfactory fit. The effect of bleomycin treatment on tumor growth was obtained by introducing the influential parameter which transferred the bleomycin concentration in tumor tissue obtained from pharmacokinetic model to the effect on tumor growth. The prediction of additive effect of both treatments joined in the model proved to underestimate the experimentally gained combined treatment effectiveness. It is the objective of our future work to evolve the model of interaction of both treatments, electrotherapy and treatment by bleomycin.

4. ACKNOWLEDGEMENT

This research was supported by the Ministry of Science and Technology of the Republic of Slovenia. We are indebted to Gregor Serša, Ph.D. from Institute of Oncology, Ljubljana, who made the animal experimental data available to us.

5. REFERENCES

[1] Gibaldi M., Perrier D., Multicompartment models. Marcel Dekker, New York, 1975.

[2] Ingram D., Bloch R., editors, Mathematical methods in medicine. Part 1, Statistical and analytical techniques. John Wiley & Sons, New York, 1984.

[3] Nippon Kayaku Co., Ltd., Antitumor antibiotic bleomycin. Nippon Kayaku Co:, Tokyo, 1985.

[4] Serša G., Novaković S., Miklavčič D., Potentiation of bleomycin antitumor effectiveness by electrotherapy. Cancer Letters, 69 (1993), 81-84.

[5] Steel G.G., Growth kinetics of tumours. Clarendon press, Oxford, 1977.

[6] Vaidya V.G., Alexandro Jr. F.J., Evaluation of some mathematical models for tumor growth. International Journal of Biomedical Computing, 13 (1982), 19-35.

[7] Wheldon T.E., Mathematical models in cancer research. Adam Hilger, Bristol, 1988.

MATHEMATICAL MODELLING OF THE HUMAN HIP JOINT AND ITS TOTAL REPLACEMENT.

Stehlík J., J. Nedoma, M. Bartoš

In this contribution the problem of mechanical processes, taking place during static loading in the contact area of the acetabulum and the head of the joint and their artificial replacements, will be discussed. During functional loading and during static loading at the contact acetabulum-head of the joint and artificial acetabulum-head of the prosthesis deformation of the contact area occurs and mutual movements of both parts of the joint are evoked. The submitted problem makes it, possible, for instance to investigate how during static loading of the pelvis the loading is transmitted via the acetabulum or its artificial replacement to the head of the joint or head of the prosthesis, resp., and how the contact surface between the acetabulum and the head of the joint is deformed, as well as the state of stresses in the investigated area of the locomotor system. The aim of the present paper is to present a new biomechanical model of human joints, which better describes the function of the loading joints as well as the stress-strain situation on the contact boundaries between the head of the joint and the acetabulum.

Let us denote by $G=G^{a}\cup G^{f}$ the 2D region, with a Lipschitz boundary ∂G, occupied by a part of the loading human skeleton (see Fig.1), where G^{a} denote the pelvis with the acetabulum and G^{f} denote the femur. Let the boundary ∂G consists of three parts $\Gamma_{\tau},\Gamma_{u},\Gamma_{c}$ such that $\partial G=\bar{\Gamma}_{\tau}\cup\bar{\Gamma}_{u}\cup\bar{\Gamma}_{c}$. The equilibrium equations in the differential form for every subdomain G^{ι}, $\iota=a,f$, are $\tau^{\iota}_{ij,j}+F^{\iota}_{i}=0$, $i,j=1,2$, $\iota=a,f$.

The stress-strain and strain-displacement relations are given by $\tau^{\iota}_{ij}=2\mu^{\iota}e_{ij}(u^{\iota})+\lambda^{\iota}e_{kk}(u^{\iota})\delta_{ij}$, $e_{ij}(u^{\iota})=\frac{1}{2}(u^{\iota}_{i,j}+u^{\iota}_{j,i})$, $i,j=1,2$, $\iota=a,f$, where $\lambda^{\iota},\mu^{\iota}$ represent the Lamé coefficients, $e_{ij}(u^{\iota})$ represents the small strain tensor. Since processes in the bone tissues and materials of the total hip prosthesis (THP) are assumed to be elastic and reverse, then the Lamé coefficients are taken in the isothermic state and satisfy the usual Lipschitz and symmetry conditions.

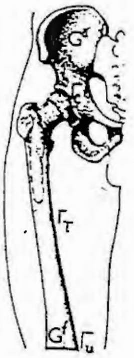


Fig.1.

Muscular tissue causes tendinous (muscular) forces in the insertions which are transmitted by the insertions to osseous tissue. In addition to these forces, there are other loading forces due to the weight of the subject which are transmitted to the skeleton. Based on knowledge of the physiological distribution of insertions on bony tissue and skeletal sites across which the loading forces are transmitted due to man's body weight of the direction and magnitude of these two types of acting forces we shall find the condition of loading of the part of the skeleton (we denote it by Γ_τ) in the form $\tau_{ij}n_j=P_i$.

Let us assume (see Fig.1) that the portion of the studied skeleton is, at a certain boundary Γ_u, fixed. Thus $u_i=0$.

So far, nobody has asked the question what conditions apply in the statically loaded stationary joint. So far the loading of the acetabulum and head of the femur, as well as that of their prostheses were calculated separately, loaded by a resultant of forces acting in a vertical direction which e.g. in case of the hip joint passes through the centre of the head and through the centre of the loaded area of the acetabulum. When the loaded area is inclined from the horizontal plane, the biomechanical equilibrium is impaired, and depending on the inclination migration of the head in a lateral or median direction occurs. The position is, however, more complicated, in particular as regards the action of forces and transmission of loading forces from the acetabulum to the head of the joint, i.e. the action of forces in the contact boundary between the acetabulum and head of the joint. These conditions are called the contact conditions. We shall assume that the friction on the contact boundary between the acetabulum and the head of femur can be neglected.

Let us denote by Γ_c the common contact boundary between both joint components before deformation. Let $n=(n_i)$, $t=(t_i)$ be the outward unit normal and unit tangential vector, respectively, to the contact boundary Γ_c. Let us denote by $\tau_i=\tau_{ij}n_j$ the stress vector, by $\tau_n=\tau_{ij}n_in_j$, $\tau_t=\tau-\tau_n n$, $u_n=u_in_i$, $u_t=u-u_n n$ the normal and tangential components of stress and displacement vectors. Let u^f, u^a denote displacement vectors in the femur and the acetabulum. During the deformation of the joint the opposite contact points are generally transfered in a different way, but in such a way that the joint component G^a cannot penetrate into the joint component G^f, as both joint components are to be elastic. Then

$u_n^f(x)-u_n^a(x)\le 0$.

The principle of action and reaction for the contact forces give $\tau_n^f(x)=-\tau_n^a(x)\equiv\tau_n(x)$, $\tau_t^f(x)=-\tau_t^a(x)\equiv\tau_t(x)$ and since the normal components of contact forces cannot be positive, i.e. cannot be tensile, then $\tau_n^f(x)=-\tau_n^a(x)\equiv\tau_n(x)\le 0$.

During the deformation of both joint components they are in a contact or they are not in a contact. If they are not in a contact, then $u_n^f-u_n^a<0$ and the contact forces are equal to zero, i.e. $\tau_n^f(x)=-\tau_n^a(x)=0$. If the joint components are in a contact, i.e. $u_n^f-u_n^a=0$, then there exist non zero contact forces $\tau_n^f=-\tau_n^a\equiv\tau_n<0$. These conditions can be written in the following form $(u_n^f(x)-u_n^a(x))\tau_n(x)=0$.

For the numerical solution the variational formulation of the problem and the finite element method (FEM) will be used. Let the given domain G, which is occupied by the human skeleton, be approximated by the domain G_h with the polygonal boundary ∂G_h and let be triangulated. We introduce the FEM approximation of the space of virtual displacements and the set of admissible displacements by $V_h=\{v|v_i\in C(\bar{G}),v_{i|T_h}\in P_1,i=1,2,v=0 \text{ on } \Gamma_u \text{ for all } \underline{T}_h\in\mathcal{T}_h\}$ and $K_h=\{v|v\in V_h,v_n^f-v_n^a\le 0 \text{ on } \Gamma_c\}$. The problem leads us to find a minimum of the potential energy over the set of admissible displacements K_h: $\min_{v\in K_h} L(v)$, $L(v)=\frac{1}{2}\int_G\tau_{ij}(v)e_{ij}(v)dx-\int_G f_iv_idx-\int_{\Gamma_\tau}P_iv_ids$. From the theory it is known that this problem is equivalent to the problem of finding the minimum of the quadratic functional with linear constrains i.e. $f(w)=\frac{1}{2}w^TB_hw-d^Tw$, $Cw\le 0$. In our case f(w) is the functional $L(v_h)$, B_h is the positive semi-definite N×N stiff matrix generated by $\frac{1}{2}\int_G\tau_{ij}(v)e_{ij}(v)dx$, d is the vector generated by the body and surface forces, C is the M×N matrix of constrains, generated by $u_n^f-u_n^a\le 0$ on Γ_c in the definition of the admissible set of displacements K_h. Here N represents a double number of points of triangulation in which the displacement is not prescribed and M is the number of points of triangulation on the contact boundary Γ_c.

In the present study our investigations of the human joint is based on the simple model, which assumes the acetabulum as being rigid and the hip joint as being elastic and having the very simple form of a wedge. Therefore in the acetabulum the displacement vector u and the stress-strain tensor being equal to zero, so that we investigate the stress-strain field in the head of the hip joint only. Figs 2a,b show the main stresses in the hip joint in the case when the hip joint is located by the relatively small loaded in the first case and secondly, when the hip joint is sizably overburdened. The main stresses, given at Fig.2a, indicates that in the whole joint and the femur compressive stresses are occurred only (we denote them by ---><---). The maximum of compressive stresses is in the place where the hip joint is maximally loaded (Fig.2b). Its places where the main stresses have great values, these parts of the bone are characterized by a strengthened bone structure. If the hip joint is overburdened, the compressive stresses occur in places where the joint is overburdened. On both sides of the acting overburden force the tensile stresses are observed (see Fig.2b, which are denoted by <---->). This fact is also observed in the orthopaedic practice. Zones with tensile stresses are characterized by a microcracking and in the case of critical overburden by the fracture zones. The Fig.3 represents the osteotomy model of the hip joint.

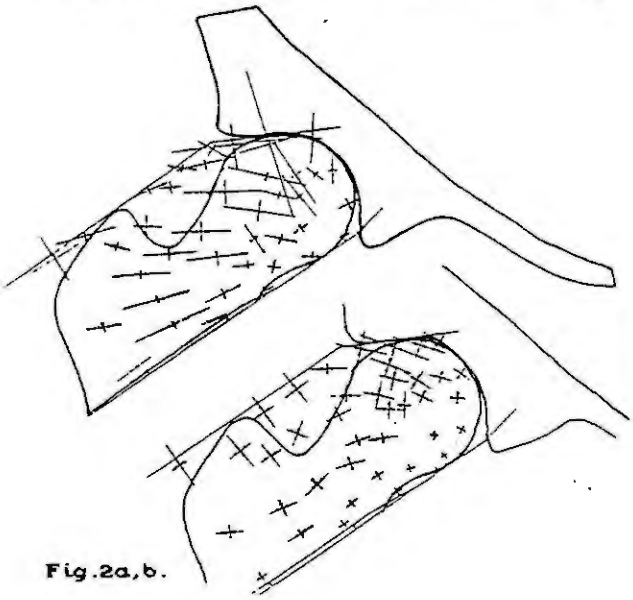


Fig.2a,b.

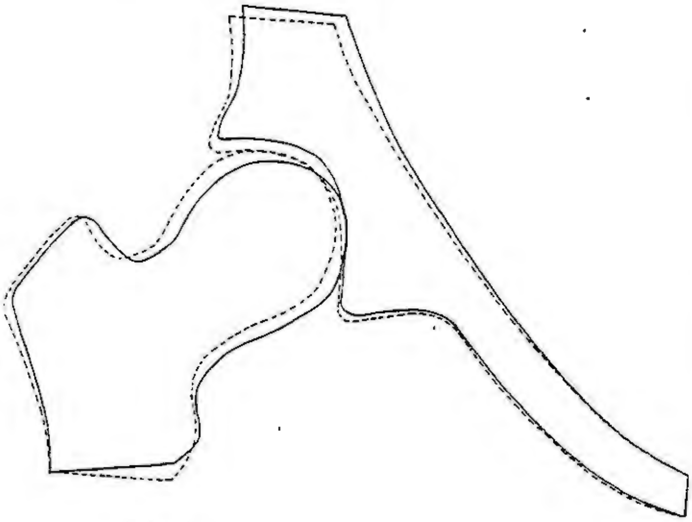


Fig.3.

References.

1. Nedoma J., J.Stehlík (1989): Mathematical simulation of function of great human joints and optimal design of their substitutes. Part I-II. Research Reports V-406, V-407, Inst. of Computer Sciences Czech.Acad.Sci, Prague 1989 (in Czech).

2. Nedoma J.(1993). Mathematical modelling in biomechanics:bone- and vascular-implant systems. ICS AS CR, Prague 1993, 220pp.

MODELING OF HUMAN LOCOMOTION AND ITS APPLICATION IN CONSTRACTION OF PROSTHESES

Victor E. BERBYUK

Department of controlled systems optimization of Institute
of Applied Problems of Mechanics and Mathematics Ukrainian
Academy of Sciences, L'viv, 290601, Ukraine

Abstract. A mathematical model has been proposed to investigate of dynamics of a musculo-skeletal system (MSS) of a man both in norm and with a below-knee prosthesis. Based on this model computer program(CP) has been composed. By means of CP composed the series of problems of dynamics of a two-leggal walking, calculation and optimization of consumption of energy on motion , optimization of parameters of the construction of artificial lower extremities of a man has been solved.

1. INTRODUCTION

To solve the problems of improvement of the existent and creation of new efficient lower limb prostheses it is expedient to use wide capabilities of mechano - mathematical modeling of a human walk process on a prosthetic limb. There were proposed earlier mathematical models of biped walk having different degrees of adequacy in multitude of works [1-7]. These models were used for different purposes: investigation of kinematic, dynamic and energetic characteristics of a human gait, an improvement of prostheses, working out and creation of anthropomorphous walking machines.

But these models don't takes into consideration the principal dynamic difference between an intact and prosthetic limbs of muscular-skeletal system of a man.

2. METHODOLOGY

MSS of a man is simulated by a plane multisegmental system of rigid weighty body. This system consists of an inertial body (trunk) and two legs. Each leg consists of three elements. The two elements with weight and inertia model the thigh and shin, while the third weightless and inertiafree element models the foot. In addition to the weights of its component parts (the trunk, thighs, and shins), the external forces acting on the mechanism include the interaction forces between the feet and the surface, which we replace by principal vector of the plane system of reaction forces of the support and principal moment of reaction forces of the support, referred to the point of the ankle joint. It is assumed that control moments act in the thigh and knee joints, these moments being treated as internal forces. Describing controlled mechanical system is used for modeling of human locomotion over a horizontal surface. The equations of motion of this controlled system is written in the form of Lagrange equations of the second kind. To receive a complete system of expressions describing the dynamics of the MSS of a man with below-knee prosthesis the Lagrange equations has been added by conditions of a kinetho-static balance of feet under the action of ankle moments and the forces of reactions of the support.

3. RESULTS AND DISCUSSION

There has been set up a successive algorithm of solution of the problem of a human gait dynamics with a below-knee prosthesis on a horizontal surface. This algorithm has the next important feature. It directly takes into consideration the principal dynamic difference between an intact and prosthetic limbs of a MSS of a man. This is achieved by the way of refusal from the assumption that the controlling force moments at all the joints of a prosthetic limb are the active ones. In the given paper a mathematical modeling of human gait with a shank prosthesis is considered based on a supposition that a force moment at the prosthetic ankle joint, is a passive one. The value of thus moment depends not only from the gait pattern of a man but from a prosthetic construction as well. In the algorithm proposed this has been shown in such a way that the moment at the ankle joint of the prosthesis (function p(t,C)) is assumed as preset one in advance. Thus if to select one or another kind of the function p(t,C) with a help of the built algorithm it is possible to study the motion of a man with below-knee prosthesis of different constructions. Based on this algorithm proposed CP has been composed in C-language. This CP makes it possible:

1.to simulate a human locomotion both normal and on artificial lower extremities;

2.based on preset anthropometric data and kinematics of MSS to calculate the forces of ground reaction and the momens of forces acting at the joints of legs during the motion of a man along a horizontal surface;

3.to calculate the energy costs required for the preset motion of a human MSS both normal and on a below-knee prosthesis;

4.to solve the problems of optimization based on a minimum of of the energy consumption of the laws of human motion with a below-knee prosthesis;

5.to solve the problems of optimization of linear, massinertial, resilient and other constructive parameters of below- knee prostheses aimed of reduction of energy costs for ambulation of a man with prosthetic limb.

Let's take some results of the use of the developed CP for the study of a task of optimization of resilient characteristics of the shank prosthesis construction. Let the construction of a shank prosthesis is designed in such a way that a moment at the ankle joint of a prosthetic leg may be represented as $p(t)=Ca(t)+Ka'(t)$, where C ,K are the parameters characterizing concentrated elasticity and viscoelasticity of the prosthesis, respectively, a(t), a'(t)- angle of deviation of the shin from a vertical axis and its angular velocity, respectively.
The following problem is formulated. To define parameters of the prosthesis $C=C_*$, $K=K_*$ making possible minimal energy costs during human motion with a preset gait. To estimate the energy consumption during motion of a man with a shank prosthesis was accepted the functional [1,2].

With the help of the CP developed a formulated problem of optimization of resilient characteristics of below-knee prosthesis had been decided for a number of gait patterns. The analysis of numerical results has shown that at a preset gait there exist optimal meanings of the parameters C, K, which require minimal energy costs during ambulation of a man on a below-knee prosthesis. With it predicted kinematic and dynamic characteristics of motion of a man on a below - knee prosthesis are well agreed quantitatively and qualitatively with experimental data of a biped walk [2].

The CP developed makes it possible to display the film of motion of a man, graphic dependencies of kinematic dynamic and energy characteristics of a two-leggal walking. The user can use efficiently the data bank on the existing prosthetic appliance and formulate the recommendations on improvement the construction of artificial extremities according to the results of solution of optimization problems. The efficiency of CP is verified by comparison of the results of simulation of walking of a man both normal and on an artificial shin with corresponding experimental data. The proposed CP can be used, in particular, for development of Computer Work Stations for solution of the problems of biomechanics and construction of prosthetic appliance. It can be utilized for development of expert systems in making of prosthetic appliance.

This paper is an extension of the research into bipedal locomotion that was undertaken in [1-3].

4. ACKNOWLEDGEMENTS

This research is partially supported by Moscow Central Scientific Research Institute of Prosthetic and Construction of Prosthesis.

The author thank M.V.Demidyuk and G.F.Ivakh for their participation in the development of CP.

5. REFERENCES

1. Beletskii,V.V., Berbyuk,V.E. and Samsonov,V.A., Parametric Optimization of Motion of a Bipedal Walking Robot. Izv. AN SSSR, MTT, [Mechanics of Solids], Vol.17, No.1, (1882), 24-35.
2. Berbyuk,V.E., Dynamics and Optimization of Robototechnical Systems. Naukova Dumka, Kiev, 1989.
3. Berbyuk,V.E., Farber, B.S., Computer Modeling For Locomotion Of Human With Artificial Leg. In: Proceedings of the IVth International Symposium on Computer Simulation in Biomechanics, Paris, 1993.
4. Cappozzo, A., Leo, T.,and Pedotti, A., A General Computing Method For The Analysis of Human Locomotion. J.Biomech. Vol.8,No.5, (1975).307-320.
5. Chow, C.K., Jacobson, D.H., Studies of human locomotion via optimal programming. Math. Biosciences, 10,(1971),239-306.
6. Hatze,H., The Complete Optimization of a Human Motion. J. Mathematical Biosciences, 28 (1976), 99-135.
7. Hatze,H., A Comprehensive Model For Human Motion Simulation And Its Application To The Take-Off Phase Of The Long Jump. J. Biomechanics, 14,No.3, (1981), 135-142.

MATHEMATICAL MODELLING OF HEAT AND MASS TRANSFER IN LIVING TISSUE

Igor Lubashevskii †, Vasil Gafiychuk ‡

†Technological Academy of Russia Federation, 117049, Moscow, Russia
‡Institute for Applied Problems of Mechanics and Mathematics Ukrainian Academy of Sciences, 290601, L'viv, Ukraine

Interest to the problem of mathematical modelling of heat and mass transfer in living tissue is partly caused by development of localized hyperthermia, which is the method of tumor treatment based on selective heating of the diseased region by external or internal sources up to a temperature $T_+ \sim 42 - 44^0C$ and maintaining the desired temperature distribution in the region for a certain time $t \sim 20 - 60$ min. Apparently in that case to optimize the treatment mathematical analysis of the temperature distribution is required.

For last years a number of different approaches to describing heat transfer in living tissue has been proposed [1,2]. Within the framework of these approaches the obtained final bioheat equation may be written in the form:

$$c\rho\frac{\partial T}{\partial t} = \nabla(k\nabla T) - c^*\rho^* j(T - T_a) + q. \tag{1}$$

Here T is the tissue temperature, ρ and c are the density and the heat capacity of the tissue, ρ^* and c^* are the same quantities for blood, k is the thermal conductivity of the tissue, T_a is the temperature of blood in large arteries of systemic circulation and is regarded as a given constant, j is the density of the average blood flow rate per unit volume, and q is the rate of heat generation.

A more accurate description of the heat transfer process in living tissue is obtained if one takes into account the fact that living tissue form an active, highly heterogeneous medium in which blood flow rate is nonlocal and substantially depends from temperature distribution. In this paper we shall formulate a phenomenological model for the thermal response in living tissue under local strong heating that will include these factors. This description can also be obtained by a rigorous analysis of the microscopic relations governing the heat transfer process in living tissue and the response of the vascular network.

The governing equations of this descriptions include the relation between j and true blood flow rate j_t which can be expressed in the form

$$j - \nabla(l_c^2 \nabla j) = j_t \tag{2}$$

where $l_c^2 \approx k/(\rho c j)$.

The evolution equation for true blood flow rate may be represented as

$$\tau \frac{\partial j_t}{\partial t} + j_t = j_0 + \frac{1}{\Delta} \int\limits_{Q_0} dr' G(r, r') j_t(r') \left[T(r') - T_a\right] \tag{3}$$

where the transient term allows for a possible time delay τ in the vascular network response to the temperature variations, j_0 is a uniform blood flow rate when $T = T_a$ Q_0 is the total living tissue domain under consideration, $G(r, r')$ is the kernel, r and r' are victors, and Δ is the length of temperature survival of the living tissue.

If we consider the tumor then the temperature response of the vessels within a tumor is strongly depressed. This allows us to set

$$G(r, r') = 0 \tag{4}$$

where the point r' belongs to the tumor domain Q_t , i.e. $r' \in Q_t$.

The properties of the given model has been analyzed numerically for the ideal thermoregulation when the kernel $G(r, r')$ may be represented approximately by the δ- function

$$G(r, r') \approx \delta(r - r')\Theta_+(r') \tag{5}$$

where $\Theta_+ = 0$ if $r \in Q_t$ and $\Theta_+ = 1$ for $r \notin Q_t$ and $q = q_0 exp(-r/\lambda)$.

Characteristics of the transient process for the system being at the state $\{T = T_a, j = j_0\}$ at the initial time are investigated by solving equation (1)-(3),(5). As it should be expected, when $\lambda \gg 1$ and q_0 is not large enough, the time increase in the tissue temperature is monotony. However, if the quantity q_0 is sufficiently large and the tissue temperature T attains values of order one during a time less than the delay time of vessel response, then the tissue temperature can go out of the temperature survival interval for a certain time and in the given case the time increase in the tissue temperature is nonmonotony. As it should be expected nonmonotony of the transient process becomes more pronounced as the characteristic time τ of the vessel response and the size of the region affected directly increase.

We applied our model to investigation of heat transfer in living tissue containing a tumor. Typically, for a real tumour response of its vessels to temperature variations is depressed (4). By computer simulation we obtained the distributions of the tissue temperature $T(\vec{r})$, the true and averaged blood flow rates $j_t(\vec{r}), j(\vec{r})$.

As it should be expected, under the ideal thermoregulation the tissue temperature can go out of the survival interval in the tumour only, whereas in the normal tissue the increase in the blood flow rate keeps tissue temperature within this interval. The temperature in the large tumour becomes already noticeably different from the normal tissue temperature when the latter attains the value $T \sim 0.5\Delta + T_a$ and in the normal tissue the blood flow rate increases twice. For the small tumour similar difference takes place as the normal tissue temperature comes near to the boundary of the survival interval ($T_+ = T_a + \Delta$) and in the normal tissue the blood flow rate increases by tenfolds.

References

[1] A.Shitzer and R.C.Eberhart, in Heat transfer in medicine and bio logy. Anal. and Appl. edited by A.Shitzer and R.C.Eberhart (Plenum, New-York, 1985) v.1, p.137.

[2] S.Weinbaum and L.M.Jiji. A new simplified bioheat equation for the effect of blood flow on local average tissue temperature. ASMI Journal of Biomechanical Engineering. v.107, 131-139, (1985).

ARE BIOLOGICAL PROCESSES TOO COMPLEX TO MODEL?

Rudibert King
Institut für Mechanik und Regelungstechnik
Universität Gesamthochschule Siegen
FRG

Abstract. Single cells consist of thousands of different compounds and reactions. Evolution theory tells us that none of these compounds or reactions are useless for the cell. Keeping this in mind, the question is addressed on which basis a selection of state variables and relation between state variables is possible. It is shown that only an interdisciplinary approach covering at least biology, chemical engineering and mathematics shows fruitful results.

1. INTRODUCTION

Comparing biotechnical processes with other processes in chemical, electrical or mechanical engineering shows fundamental differences. Even for an apparently simple organism, such as *E. coli*, more than 5000 different intracellular compounds are known. Each of these are necessary for growth, production or for other sometimes unknown purposes. For some growth conditions all of these compounds show the same specific growth rate, i.e. the degree of freedom of the cell reduces to one. So called unstructured models can be used here. For other, very often technically interesting conditions the composition of a cell shows a profound dynamical evolution. Here, the degree of freedom is large. Structured models are needed.

To combine both regimes in one model a consideration of all compounds, reactions, and especially regulations seems necessary. This is impossible for reasons of complexity. Moreover, even the behaviour of *E. coli*, which is addressed in numerous research projects, is only known for a limited number a environmental conditions. For new, technically interesting organisms the situation seems to be worse. However, it is known that the basic principles of the cells chemistry applies to all cells. Therefore, a chance exist that at least for some classes of organisms for limited and well defined conditions a general model structure based on a reduced set of state variables can be given. A selection of these state variables on a rational basis, however, is not possible but involves intuition and most desirably an interdisciplinary cooperation or education.

As an example a highly structured model is outlined with which the very complex behaviour of a bacteria, *Streptomyces tendae*, could be described. A comparison with other Streptomyces strains shows that the basic model assumptions hold true for these bacteria as well.

2. A HIGHLY STRUCTURED MODEL FOR ONE STRAIN

Cultivating the antibiotic producing strain *S. tendae* in a batch culture shows some remarkable patterns of growth. It is known for all organisms, including bacteria, that so called substantial substrates have to be supplied for growth. Among these the carbon, the phosphor and the nitrogen source amount to the largest portions. Whenever one of the sources is exhausted growth is supposed to stop. However, *S. tendae* continue to grow after depletion of phosphor or nitrogen, though with a slower growth rate. This can only be explained by a growth rate depending on intracellular substances, e.g. intracelullarly stored phosphates and N-compounds. As a result an unstructured model is not suitable to describe this strain.

To analyse the complex growth and production behaviour DNA, RNA and proteins are measured. Three distinct phases of growth are found. A first growth phase is characterized by a common exponential growth of all compounds as long as all substrates are available in the fermenter broth. In a second phase the total amount of DNA in the fermenter increases when phosphor is depleted, i.e. there is still a DNA-duplication. In this phase the total amount of RNA is constant, whereas proteins and total cell mass increase, and antibiotic production starts. The last phase is characterized by a decline in the amount of RNA, constant amounts of proteins and DNA, and an increasing biomass. In case of a nitrogen limitation, DNA increases in the second phase too, however, RNA and proteins are degraded, antibiotic production starts. Here, DNA and biomass are constant in the third phase, the degradation of RNA and proteins slows down.

To explain this complex behaviour the central dogma of molecular genetic is considered. It states that the information in a cells goes from the DNA over the RNA to proteins and other compounds, Fig. 1. These macromolecules are synthetised out of precursors which are produced from substrates, respectively.

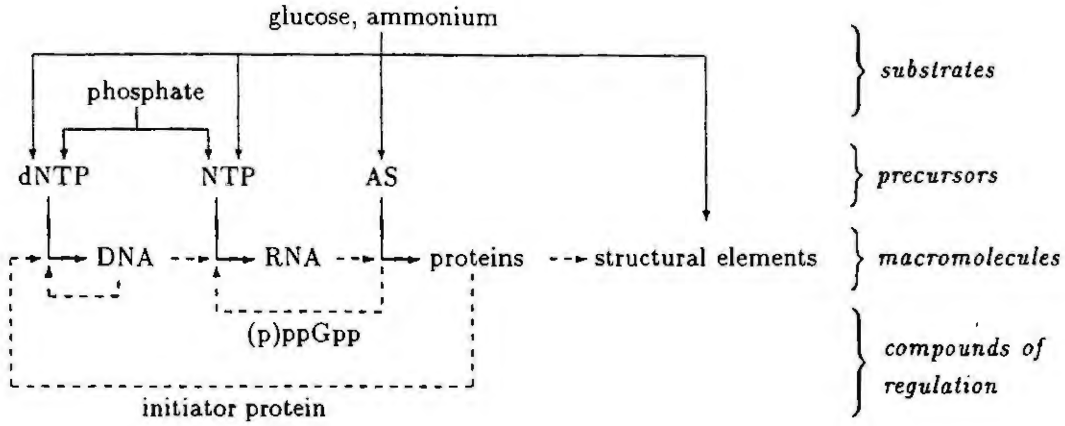


Figure 1: Outline of the flows of compounds and information in a cell. — flow of compounds; - - - flow of information

Jensen and Pederson [1] have shown that the production of the macromolecules in Fig. 1 is highly regulated, e.g. by pppGpp and an initiator protein. Hence, whenever a limitation in the production of the precursors occurs the metabolism is changed very efficiently such that needless metabolic pathways are cut down. As the concentrations of these regulating substances change very fast with respect to the macromolecules, precursors and substrates a quasi steady state assumption can be applied during modeling. In the model all biological information are translated into apropriate material balances for the internal compounds and substrates. The state variables are: D - DNA, R - RNA, Pr -proteins, As - aminoacids, Nu - nucleotides ((d)NTP), U - structural elements, A - ammonium, Ph - phosphate, and C - glucose. A material balance for the antibiotic can be added readily.

$$\frac{d}{dt}\begin{pmatrix} m_D(t) \\ m_R(t) \\ m_{Pr}(t) \\ m_{As}(t) \\ m_{Nu}(t) \\ m_U(t) \\ m_A(t) \\ m_{Ph}(t) \\ m_C(t) \end{pmatrix} = \underbrace{Q_e(t)\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ c_{Ae}(t) \\ c_{Phe}(t) \\ c_{Ce}(t) \end{pmatrix}}_{inlet} - \underbrace{\frac{Q_a(t)}{V(t)}\begin{pmatrix} m_D(t) \\ m_R(t) \\ m_{Pr}(t) \\ m_{As}(t) \\ m_{Nu}(t) \\ m_U(t) \\ m_A(t) \\ m_{Ph}(t) \\ m_C(t) \end{pmatrix}}_{outlet} - \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \mu_E(t)V_x(t) \end{pmatrix}}_{maitnenance} +$$

$$\underbrace{V_x(t)\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & -1 & 1 & -Y_{AsNu} & -Y_{AsU} & 0 & Y_{AsPr} \\ -1 & -1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1-Y_{AsPr} \\ 0 & 0 & 0 & -Y_{AAs} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -Y_{PhNu} & 0 & 0 & 0 \\ 0 & 0 & 0 & -Y_{CAs} & -Y_{CNu} & Y_{AsU}-1 & 0 & 0 \end{pmatrix}\begin{pmatrix} \mu_D(t) \\ \mu_R(t) \\ \mu_{Pr}(t) \\ \mu_{As}(t) \\ \mu_{Nu}(t) \\ \mu_U(t) \\ \mu_{zR}(t) \\ \mu_{zPr}(t) \end{pmatrix}}_{metabolism} \qquad (1)$$

Although model (1) is metabolically motivated in many parts, it is experimental in nature [2]. This is a result of the formal kinetics used for the reaction rates μ_i. As an example, after the quasi steady state assumption for the initiator protein the rate of DNA-duplication looks as follows:

$$\mu_D(t) = \mu_{Dm} \cdot \frac{g_{As}(t)}{g_{As}(t) + K_{PrAs}} \cdot \frac{g_{Nu}(t)}{g_{Nu}(t) + K_{PrNu}} \cdot g_R(t) \cdot g_D(t) \quad , \qquad (2)$$

in which g_i represents the internal concentration of substance i.

The parameters of the model are identified with methods of nonlinear optimization by fitting several different experiments at the same time. Fig. 2 shows a comparison between measured and simulated values. In the meantime this model could be applied with great success in the context of optimal control [3].

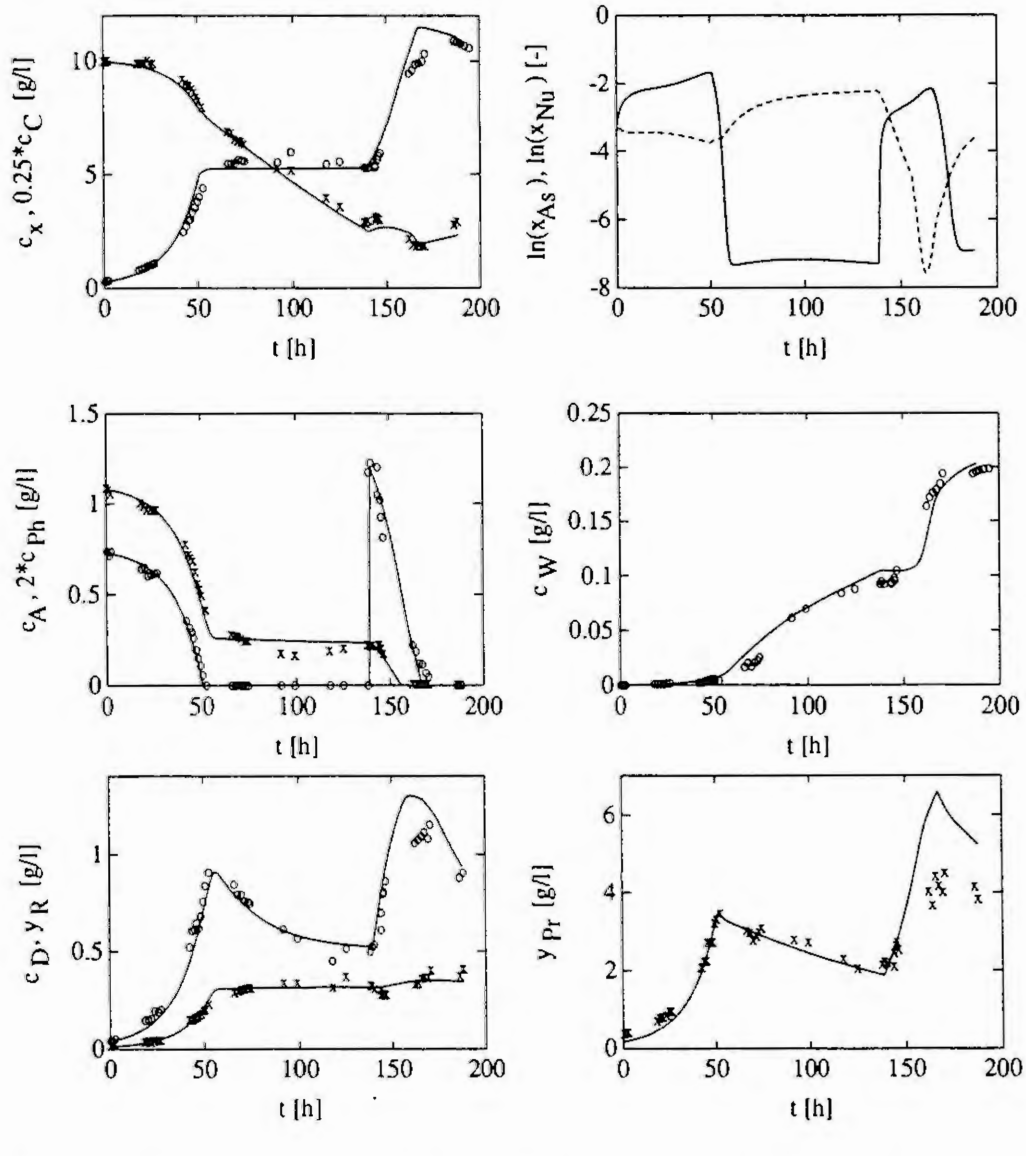


Figure 2: Comparison between measured and simulated values with $c_i = m_i/V$: $i = x$ - biomass (o); C - glucose (x); A - ammonium(o); Ph - phosphate (x); D - DNA (x); W - Nikkomycin (o); with $x_i = m_i/m$: $i = As$ - amino acid (—); Nu - nucleotides (- - -); with $y_i = m_i/V$: $i = R$ - RNA+0.25*DNA (o); Pr - proteins+As (x)

3. OTHER STREPTOMYCES STRAINS

Summarizing the effort needed to build up the outlined model it has to be concluded that a considerable amount of time was necessary to cope with the complexity of the biological system. If this is true for every new strain the financial benefit is a crucial question in an industrial environment. However, it can be shown that the observation of the three different growth regimes with typical patterns of synthesis and degradation of macromolecules can be found for other Streptomycetes too. To illustrate this the time axes of the three phases are normalized in Fig. 3 for different strains individually, such that corresponding phases coincide. From these results it is believed that the basic structure of the proposed model can be applied as well for other strains. The effort of modeling 'reduces' to parameter identification and to minor changes in formal kinetics.

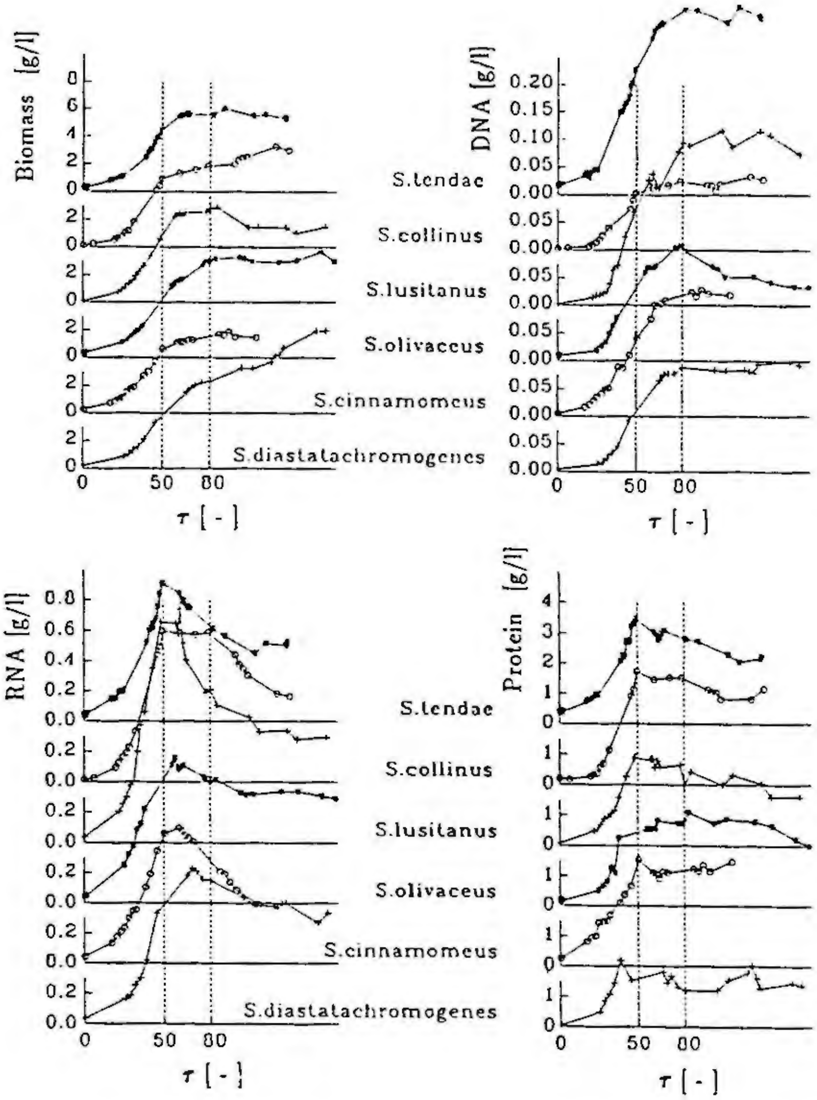


Figure 3: Comparison of different strains. A phosphate limitation takes place at $\tau = .50$ [2].

3. CONCLUSIONS

Biological processes can only be modelled satisfactorily if as much biological information is included as possible. However, the complexity of these systems requires some sort of lumping, for which up to now a rational basis does not exist. Intuition still plays an import role. The results presented above, however, suggest that at least for some classes of biological systems common powerful models can be proposed.

4. REFERENCES

[1] Jensen, K.F. and Pedersen, S., Metabolic growth rate control in *E.coli* may be a consequence of subsaturation of the macromolecular biosynthetic apparatus with substrates and catalytic components. Microbiol. Rev. 54 (1990), 89-100.

[2] King, R., Mathematische Modelle myzelförmig wachsender Mikroorganismen. Habilitationsschrift Universität Stuttgart (1992).

[3] Waldraff, W. et.al., Modelbased control of bioprocesses - Trajectory optimization and control. ECB 6, Florence, 13.-17.6.1993.

TOWARDS A MULTI-REPRESENTATION IN BIOPROCESS MODELLING WITH A VIEW TO PROCESS CONTROL.

J.M. Flaus, G. Romero and A. Cheruy
Laboratoire d'Automatique de Grenoble, UA CNRS 228, BP 46,
F-38402 Saint Martin d'Hères
Fax: (33) 76826388 E-mail: flaus@lag.grenet.fr

Abstract. This paper deals with the study of different ways of describing knowledge about bioprocess that could be of interest for integrated advanced control that includes "classical" control loop and supervisory control. We successively examine physics law based mathematical modelling, linear black box modelling, qualitative physics based modelling, fuzzy modelling, neural network based modelling and rules based modelling or Expert Systems.The main interest of this paper is to present approaches that are not closely related from a unifying control point of view.

Key words: Modelling, control, knowledge representation, bioprocess, neural network, fuzzy and qualitative representations , artificial intelligence, integrated control.

1. INTRODUCTION

The monitoring and control of bioprocesses is difficult. These processes are non-linear, time-varying and living cells are involved. So modelling of these systems is tedious and the lack of sensors for measuring the biological variables makes things still more difficult.

So, to face these problems, the temptation is high to resort to alternative approaches such as Expert systems, neural networks or fuzzy sets. Very often, such approaches are used to try to deal with the lack of knowledge about the system. For example, fuzzy systems are used to model an unprecise understanding of the behaviour of the process.

However, a thorough analysis of these modelling approaches shows that things are a little bit more complex. The goal of this paper is to present various modelling techniques and to try to see how they can improve the representation of the knowledge about a bioprocess. The first point, that we willl discuss, is the monitoring or control purpose for which a non classical modelling approach could seem as being relevant.

Then we will present several techniques ranging from the classical mathematical or black box modelling approaches to some rather recent ones such as neural networks, experts systems , qualitative and fuzzy modelling approaches. As we are interested in the modelling for process control, it seemed important to us to highlight how the state and its evolution can be represented.

After this quick overview, we will propose some ideas that can be considered as a first step to a multi representation approach with bioprocess supervision as an objective.

2. ABOUT MODELS USED FOR BIOPROCESS CONTROL

Bioprocess modelling is one of the main bottleneck for control and monitoring. Models are difficult to build because bioprocesses are complex, time-varying and experiments are difficult to carry out. On the other hand, heuristic knowledge is not always used[1]. An improvement of the performances of the task using the model could be expected if we could use the full knowledge about the process. So the question is :"In what specific area is it possible to improve performances with a better representation of the knowledge ?". We can distinguish three main uses of models in the process control area:

(i) estimation of non measurable variables,

[1] A good example of that is the modelling of the specific biomass growth rate: as expressing them as a function of the biological variables is difficult, an interesting approach has been to consider them as a time-varying parameter that is estimated on-line (1). However, in this approach, it is not possible to represent that if the sustrate concentration is equal to zero, then the growth rate is also equal to zero.

(ii) regulation of some concentration such as the substrate concentration,
(iii) optimisation
(iv) and supervision of the process including fault diagnosis and help to the operator.

For applying estimation techniques, we need a very precise dynamical model handling numerical values. So no improvements can be obtained in using a model that is not into a mathematical form. Much of the same could be said for optimization. For regulation purposes, a less precise model is sufficient, but a good compromise robustness/performances requires a rather good model. So, in our opinion the only area for which an approximate model could be sufficient is supervision. Indeed, that kind of model can be sufficient for keeping the process around the operating what can be very helpful for the operator as soon as the system is complex.

In this study, our goal is to find out a way to represent the process, that can be not exhaustive, but that allows to describe unprecise knowledge of any kinds: qualitative, quantitative, heuristic and logical for example. So with this objective in mind, looking at modelling approaches from various horizons may be fruitful.

3. MATHEMATICAL MODELS

The main motivation to use mathematical models is that these models have real physical significance describing internal physical states of the bioprocess. Every variable has a physical signification and the equations are written to express a physics or chemistry law.

Basically, these models are obtained in writing the mass and energy balances equations. In these equations, two kinds of parameters appear: the yield coefficients, such as Yxs, and the kinetic parameters, such as the specific growth rate Most often, these parameters depend on the state variables according to a law which is difficult to establish.

Representation of the state and its evolution

The state of the system is described by the state variables and its evolution is governed by a set of non linear differential equations. Given an initial state, the set of differential equations and the values of the input that are applied to the system, the evolution of the state variables as a function of time can be computed.

Interest and limitation for process control

This type of model is well suited to the simulation of the behaviour of a system. It can also be used for control design (7), either with predictive control algorithm, or with non-linear control techniques if the model is not too complex. However, as non-linear systems theory is in development, there is no methodology for designing control algorithms based on non-linear systems, that can guarantee the desired performances and basic properties of the control loops, such as stability and robustness. Moreover, if we want to take into account the effect of environment variables such as pH, temperature, which can easily be used to control the process, the complexity of this kind of model will increase quickly, mainly because they act non linearly on the kinetic parameters.

4. LINEAR BLACK-BOX MODELS

To be able to control a system, a basic requirement is to know the effect of a variation of the inputs on the outputs of the system. The understanding of the internal behaviour of the system is not necessarily needed. That is the reason why the black-box models also known as transfer functions have been developed and extensively studied in classical control theory. The generic structure formulation of a black box model is as follows:

$$Y(s)=\frac{b_0+b_1s+\ldots b_ms^m}{a_0+a_1s+\ldots a_ns^n}e^{-\tau s}U(s)\ \textit{(continuous form)}\ \text{ and } Y(z)=\frac{b_0+b_1z+\ldots b_mz^m}{a_0+a_1z+\ldots a_nz^n}z^{-r}U(z)\ \textit{(discrete form)}$$

where Y(.) is the output, U(.) the input, τ (r respectively) is the dead time and the coefficients a_i and b_i are the model parameters. They can be estimated by applying an appropriate technique such as least squares regression, provided sufficient data y(t) and u(t) are available. Once the parameters have been determined and given the appropriate input and output histories, the model can be used to provide predictions of the output. This constitutes a simple, but effective mean of predicting the immediate effects of changing process inputs, based upon knowledge of its dynamics in the vicinity of some operating point.

Representation of the state and its evolution

In the state space formulation of a black-box model:

$x(t+1) = F\,x(t) + G\,u(t)$ and $y(t)= H\,x(t)$

the state of the system is obviously described by the state vector and the evolution can be computed with the transition matrix . In the input-output vector, the state of the system is hidden in the vector of the past inputs and past outputs. Indeed, it can be shown that the state space form of a linear system is equivalent to the input-output relationship, which is the ARMAX model for the discrete case or the transfer function G(s) for the continuous case. So, the state can be computed from the past inputs and outputs, and in the same way, the transition matrix can be obtained from the polynomials A and B.

Interest and limitation for process control.

A very nice property of the black box models is that they are linear. Theory for linear systems is very well developed and a lot of properties, such as stability or robustness, are easy to study. For small deviations around a steady state point, a black box model is well suited to describe the system. Unfortunately, in bioprocess control, we are often interested in batch or fed-batch modes, for which there are no steady state. That is the reason why this kind of model is usually not used and that non-linear models based on mass balances are preferred.

5. QUALITATIVE PHYSICS BASED MODELLING

Qualitative reasoning was introduced as a formalism for reasoning about physical systems that captures many of features of human reasoning. It is especially useful for modelling complex processes where the governing equations are known in sparse detail, such as bioprocesses (7). Qualitative modelling reduces the quantitative precision of real values variables by replacing them with quantitative variables that belongs to a set of linguistic values, usually +, 0 and -. The behaviour of a physical system is described by the qualitative value of the state and of its first and second derivatives. Qualitative operators are introduced for combining qualitative values. The propagation of the state is made based on the Qualitative Differential Equations that can be seen as some constraints to generate all possible successive states from the active state.

This kind of model is built from the mathematical model based on mass balances, where the real values variables are replaced by a qualitative variable and the parameters are replaced by their sign.

Representation of the state and its evolution.

The state of the process is described by the qualitative value of the state variables. Evolution of the system can be predicted with the confluences equations. However, the time is not explicitly handled by this type of model, which gives only the chronology of the evolution, but not the state variables as a function of time.

Interest and limitation for process control.

Qualitative models that we presented are not precise enough and often leads to an very large number of solutions for the system evolution. Some improvements have been proposed such as "order of magnitude reasoning" (11). Qualitative model are not really applied for process control but rather for diagnostic tasks. They can be used to infer a causal explanation of a behaviour with an incomplete knowledge of the behaviour model.

6. FUZZY MODELLING FRAMEWORK

Very often the bioprocesses are characterized by the fact that information available for the description of their behaviour is semiqualitative (experience). Semiquantitative information is something that has not yet been fully accepted by bioengineering science. Probably the main reason for this is the fact that it is very difficult to incorporate semiquantitative information into conventional mathematical models. Fuzzy modelling offer a way to put subjective information into a form acceptable for computers (6) .

The main idea of fuzzy set theory is that the logical values *true* (1) and *false* (0) are replaced by continuously graded values throughout the hole interval (0,1). Fuzzy theory permits a value of a linguistic variable, (e.g. "low", "high") to be represented and processed by a computer. The representation of such values is performed by using membership functions. A membership function of a linguistic variable associates with each possible value of a variable a real number from the interval (0,1). This real number represents the grade of membership of the variable in the particular fuzzy set.

Representation of the state and its evolution

Fuzzy sets modelling are a way to represent the state; instead of using quantitative values, the state variables are represented by grade of membership to a set of fuzzy sets. The transition function can then be given by a table with two entries; the state and the input at time t to which is associated the state at time $t+1$.

Most often fuzzy control do not use an explicit model of process that is to say a model capable of predicting the evolution, but rather an implicit of it namely a model that give the action to applied as a function of the error, (and of its derivate) between a measurement and a setpoint. So fuzzy theory can be seen as a kind of interface between numeric values, (e.g. the output error), and symbolic human knowledge. Add in this case, the state is in a numeric form.

Interest and limitation for process control.

Fuzzy set theory seems to be a nice way of interfacing quantitative and qualitative knowledge. Human knowledge can be expressed in a simple manner through rules involving linguistic variables. But currently fuzzy controller is still limited, for example, process with dead time are not handled correctly.

7. NEURAL NETWORKS BASED MODELS

Neural networks (NN) seems to be an interesting tool for modelling and control purposes because they are able to represent an arbitrary nonlinear mapping (4). The utility of NN in providing viable process models has been demonstrated in the area of chemical systems , where the technique was used to successfully characterise two non-linear systems as well as interpret biosensor data (2). In the area of bioprocess (13) have used a NN to predict the biomass in a continuous mycelial cultivation and in a fed-batch penicilin cultivation (12) used a NN for the on-line prediction of biomass and substrate in a continuous stirred tank reactor.

NN's are made up of a number of simple, highly connected processing elements (PE) called neurons. A neuron has several inputs connections and one or more output connections and typically the NN are organized into layers with each PE in one layer. Associated with each connection is a weight *w* and each PE is a state (usually on or off). Together these weights and states represent the distributed data of the network. The weights of a network together represent an energy surface, and their actual values determine the set of patterns recognisable by the network. During pattern recognition, each PE operates as a simple threshold device. A PE sums all the weighted inputs (multiplying the connection weight by the state of the previous layer PE) and then applies a threshold function, such as a sigmoid function.

The NN must be trained in order to model a system, i.e. the weights *w* must be chosen in a way that the outputs of the network fits with the output of the system at a given input signal. To do this, several procedures are used. One of the most popular is the *back propagation* procedure which adjusts the weight of the connections to minimize the difference between the actual output of the network and the desired input, so the network is said trained if the difference between the actual output of the network and the desired input is acceptable low. The NN is then used for its purpose and the weights are fixed..

The choice of the configuration of the NN depends on the task to be performed. For modelling physical systems, a feed-forward layered network is used which consists in a layer of input neurons, a layer of output neurons and two or more hidden layers (13).

Representation of the state and its evolution

NN's could be seen as the non-linear version of Black-Box modelling. The state is represented by the inputs and outputs while the evolution function is the neural network, which is nothing else than a static non-linear function.

Interest and limitation for process control.

From the control theory view point the ability of NN's to deal with non-linear systems, e.g. to represent non-linear mappings or to model non-linear systems is the characteristic the most important given that bioprocesses are systems strongly non-linear. They can be exploited in the synthesis of non-linear controllers (9). The most important limitation is the lack of a methodology to configurate the NN, e.g. how many layers/PE are required to achieve a specific degree of accuracy for the process to be modeled.Another important limitation is the static characteristic of the NN that prevents to represent well the dynamic systems like bioprocesses.

8. RULES BASED MODELS

Not all bioprocesses can be represented completely by exact mathematical models which in fact, is frequently needed for modern process control. For the control of a bioprocess, a priori knowledge exists which cannot be readily incorporated as a mathematical model but which can be represented by language-based-rules, such as *IF/THEN* conditions (rules). Rules should themselves be capable of expressing dynamic relationship. (8)

An expert system (ES) is a software-based system that describes the behaviour of an expert in some field by capturing the knowledge of one or more experts in the form of rules and symbols. At a minimum, the system has three parts. First it has a collection of rules called the *rule base* which consists in number of production rules of the form IF/THEN. The first half of each rule expresses a condition which, if true, indicates that it is appropriate to perform the action expressed in the second half of the rule. A second part is the collection of *facts*. Sometimes these are embedded in the rule base along with the production rules, and sometimes they are contained in a separate knowledge base. The third part of an ES, the *inference engine* is the active portion of the system. It performs two main functions: it matches the current state of the facts against the conditional part of the rules to generate a list of matching rules; and it selects one or more of these matching rules to "fire" or execute the action part of the rule.

Representation of the state and its evolution

Using the rules, in the condition part of the rules a process state is described. If this state is present (*state*), then the action part (*evolution*) of the rule will be executed . For example, in the next rules we can identify the state of the process and clearly identify the action (*evolution*) in response of this state:

IF *DO gradient > 20% and it is the exponential growth phase and the aeration rate was not increased during the last 15 min* THEN *conclude that there is a bad state and activate induction procedure.*

Interest and limitation for process control.

Only, applications of ES's from research laboratories have been described. In these applications ES's which are rule-based systems containing IF/THEN rules, were combined with others techniques and were used mainly for the supervision, fault diagnostic, optimization and control of bioprocesses (1)(8)(5). Some of these systems permit more than one knowledge representation technique such as objects and procedures. ES approach works well when the problem has a reasonably accessible set of rules, with a clearly identified expert who is available to help to build the system. However, this approach stumbles when the problem cannot easily solved with known rules or when multiple experts disagree on how to solve it. Another limitation in the development of ES's is the problems about the coherence or consistency of the rules base, which at present, are not resolved yet.

9. TOWARDS A MULTI REPRESENTATION APPROACH

From this quick review of some of the main ways to represent knowledge, it appeared to us that there were no approach that would be suited for our needs, that is to say that would be able to represent some quantitative knowledge as well as some qualitative one.

The first idea that would be to replace a classical model by a qualitalive or fuzzy approach is, in fact, not interesting: indeed we have to lose some information that we have with a classical model and that can not be correctly represented with a qualitative model. The same can be said for rule based models: they are clearly devoted to representation of logical information, which is in fact poor. As to neural networks models, they can be seen as a non linear function, and from a process control point of view, we feel that problems that arise from them, in particular about their training, are not completely solved .

So, we feel that progress can be made only in using a multi-representation approach, which would be able to retain the benefits of tradionnal approaches while improving the modelling of unprecise knowledge. There are two ways to imagine such a new appraoch:

(i) either, in designing a new kind of variables, derived from real values, that would able to better represent unprecise information. For example, stochastic modelling is an attempt in that direction.

(ii) or in trying to make several representations coexist. For example, we could imagine to use a classical model for describing the process around the operating point while relying on rules based model far from it.

The first approach has the advantage to be coherent by construction, but possibly could lack of flexibility. The second seems to be more flexible, but could be difficult to manage. For example, problem of switching from a representation to another could be difficult to solve properly. The problem is opened.

10. CONCLUSION

In this paper, we dealt with the problem of the knowledge representation for bioprocesses with a view to process control.

First, we tried to analyze the modelling limitations of tradional approaches. We pointed out that supervision is the only field for which a better organization of the different kinds of the available knowledge about the process could be of interest. Indeed, estimation and control techniques require a rather good model and its quality is closely related to the performances of the algorithms. On the other hand, supervision has as an objective to help the operator and as techniques that are involved are based on the imitation of human reasonning, they could benefit from a better use of the available knowledge.

With this in mind, we have presented some modelling approaches that could seem relevant for improving the knowledge representation. We successively examined mathematical models, linear black-box models, qualitative physics based modelling, fuzzy modelling, neurals network models and rules based models.We tried to unify the presentation of each technique from a control point of view mainly in showing how the state and its evolution was described.

It appeared from this review that a multi-representation for bioprocess modelling could be useful for supervision purposes. We think that such an approach would be especially relevant for biotechnological process and that a lot of progress could be made while opening interesting research perspectives, such as estimation techniques for that kind of models.

11. REFERENCES.

[1] Aynsley M, Peel D, Hofland A.G., Montague G.A. and Morris A.J., (1990). *Real-Time Knowledge-Based Systems in Fermentation Surpevisory Control.* In Knowledge-Based systems for Industrial Control. IEE control Engineering Series 44. Peter Peregrinus Ltd.England

[2] Bastin G., Dochain D. (1990), *On-Line Estimation and Adaptive Control of Bioreactors.* Elsevier, Amsterdam.

[3] Bhat N., Minderman P., McAvoy T. and Wang N. (1989), *Modelling Chemical Process Systems via Neural Computation.* Proceedings of 3rd. Int. Symp. "Control for profit", Newcastle-upon-Tyne, England.

[4] Billings S.A., Jamaluddin H.B. and Chen S. (1990), *Properties of neural networks with applications to modelling non-linear dynamical Systems.* International Journal of Control, 1990 Vol. 55 No. 1, 193-224.

[5] Cooney C.L., O'Connor C.L. and Sanchez Reira F. (1988). *An Expert System for Intelligent Supervisory Control of Fermentation Processes.* 8th International Biotechnology Symposium. Paris

[6] Dohnal M. (1988a) *Fuzzy Bioengineering Models.* Biotechnology and Bioengineering, Vol. 27, 1146-1151.

[7]Hangos K.M., Csaki ZS. and Jorgensen S.B. (1992), *Qualitative Model-Based Intelligent Control of a Distillation Column.* Engineering Applications Artificial Intelligence, Vol. 5 No. 5, pp 431-440

[8] Hitzman B., Lubert A. and Schugerl K. (1992), *An Expert System Approach for the Control of a Bioprocess. I: Knowledge Representation and Processing.* Biotechnology and Bioengineering, Vol. 39, pp. 33-43

[9] Hunt K.J., Sbarbaro D., Zbikowski R. and Gawthrop P.J. (1992), *Neural Networks for Control Systems- A Survey.* Automatica, Vol. 28, No. 6, pp. 1083-1112

[10] Johnson A. (1987), *The control of Fed-batch Fermentation Processes - A Survey* Automatica, Vol. 23, No. 6, pp. 691-705

[11] Mavrovounitiotis M.L. and Stephanopoulos G. (1989), *Order-of-magnitude reasoning with O(M).* Artificial Intelligence in Engineering Vol. 4, pp. 106-114.

[12] Thibault J., Van Breusegem V. and Cheruy A. (1990), *On-Line Prediction of Fermentation Variables Using Neural Networks.* Biotechnology and Bioengineering, Vol. 36, pp. 1041-1048

[13] Willis M.J., Montague G.A., De Massimo C., Tham M.T. and Morris .J. (1992). *Artificial Neural Networks in Process Estimation and Control.* Automatica, Vol. 28, No. 6, pp. 1181-1187

MODELING THE NITROGEN DYNAMICS IN AN ALTERNATING ACTIVATED SLUDGE PROCESS

S.H.Isaacs, H. Zhao, H. Soeberg and M.Kümmel

Department of Chemical Engineering, Bldg. 229
Technical University of Denmark
DK 2800 Lyngby, Denmark

Abstract. The paper presents a relatively simple model describing the nitrogen dynamics in an alternating waste water treatment process. Only two components, ammonia and nitrate plus nitrite, are considered, and reaction rates are described using empirical expressions. Despite its simplicity, the model serves well for two distinct purposes: as a prediction model for use in control strategies and as a steady state (limit cycle) model for process analysis. Nitrification rate limitation at low ammonia concentrations is shown to influence model predictions significantly. A correction factor is proposed, which allows the less computationally intensive case of zero order kinetics to be employed when limitation occurs.

1. INTRODUCTION

Activated sludge processes for nutrient and organic removal from waste waters are best described by relatively complex models of a mechanistic nature. These models subdivide the sludge into several classes of organisms, and subdivide the organic matter entering and within the process into several classes of biodegradability (e.g. [2,5]). Combined with the various nutrient species of significance (ammonia, nitrate and phosphate), the state vector of these models is large and the number of kinetic and stoichiometric parameters are many. Although such models serve well in the design and understanding of nutrient removal processes, their application in control strategies is difficult due to computational complexity and the problems of state and parameter estimation.

One approach to model simplification for control purposes is to freeze the slower dynamics of the process. The microorganisms serving as biological catalysts are not explicitly modeled. The rates of key reactions such as *nitrification* (the conversion of ammonia to nitrate via nitrite by autotrophic bacteria) and *denitrification* (the conversion of nitrate to nitrogen gas) are described with simple empirical expressions whose parameter estimates are updated periodically using on-line measurements. As a result of this simplification, long term effects such as changes in population balance cannot be described. However, the models can be used to control the process response to shorter term disturbances such as diurnal and weekly variations in inlet water quality.

This methodology has been applied to an alternating type process (trademark Biodenipho, see e.g. [1]) designed for the biologically mediated removal of organics, nitrogen and phosphorus from waste waters [3,4,6]. A characteristic of this process is the semi-batch manner in which nitrification and denitrification are performed sequentially in two similar aerobic/anoxic tanks. This imparts the process with a continually excited internal dynamic, which provides a rich source of information for the estimation of rate associated parameters. The models applied in the previous works describe the dynamics of the major nitrogen containing components only, using linear (zero order) expressions for both the nitrification and denitrification rates. This paper provides a general framework for these models, as well as examines the effect of describing the rate of nitrification with nonlinear (limitation) kinetics. A correction factor is proposed which allows the use of the less computationally intensive zero order kinetic model to describe the situation of limitation kinetics.

2. MODEL EQUATIONS

The main components of the alternating process and the basic 4 or 6 phase operation cycle are shown in Fig. 1. The 4 phase cycle consists only of phases 2, 3, 5 and 6, and limits the aeration time of each tank to a maximum of 50% of the cycle. In the current implementation, the aerated phase of each tank begins as indicated in the figure but is terminated when the ammonia concentration in the tank falls below a given setpoint value. The model equations describe the concentrations of nitrogen as ammonia (NH_4-N) and as nitrate plus nitrite (NO_x-N) in tanks T1 and T2, and account for flow associated transport into and out of the tanks, the conversion of NH_4-N to NO_x-N by nitrification, and the disappearance of NO_x-N by denitrification.

The model is compactly expressed in Eqs. (1) to (4) for all flow and aeration patterns shown in Fig. 1. D is the dilution rate for either tank. C denotes concentration where the subscripts a, n, 1 and 2 stand for NH_4-N, NO_x-N, tank T1 and tank T2, respectively. C_a^{AN} is the NH_4-N concentration leaving the anaerobic zone. ζ are switching functions introduced to facilitate this representation. $\zeta_{1,5,6}$, $\zeta_{2,3,4}$, $\zeta_{1,6}$, and $\zeta_{3,4}$ are equal to 1 during those phases whose numbers appear as subscripts and are equal to 0 otherwise. ζ_{O2} is equal to 1 if aerobic conditions exist and is equal to 0 otherwise. r_n and r_d are respectively the rates of nitrification and denitrification, and may be functions of the modeled states (NH_4-N and NO_x-N concentrations) and controlled variables such as the dissolved oxygen concentration applied during aerated periods.

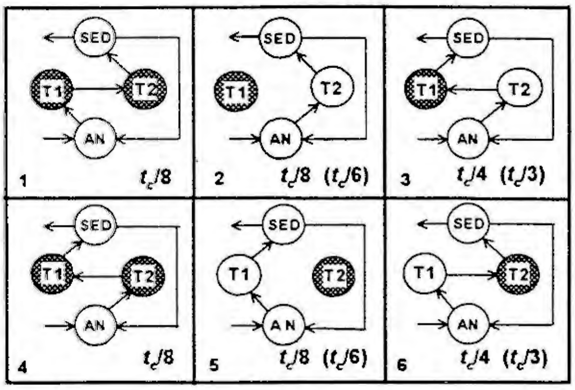


Fig. 1. Phase schedule of the alternating process. AN: anaerobic zone; T1, T2: anoxic/aerobic tanks; SED: final clarifier. The shading indicates when tanks may be aerated. The lower right numbers indicate the phase duration relative to the cycle length t_c for the 6 phase (4 phase) schedule.

In formulating Eqs. (1) to (4), tanks T1 and T2 are considered to be well mixed vessels. The release of ammonia due to cell lysis and hydrolysis, as well as the incorporation of nitrogen into new cell mass, are not accounted for. Also not accounted for are the transition times between aerobic and nonaerobic conditions, and the possibility of simultaneous nitrification and denitrification. Despite its simplicity, the model has been found to describe the essential nitrogen dynamics occurring in tanks T1 and T2 of a 2800 liter pilot scale process reasonably well once suitable expressions for r_n and r_d are provided. These expressions are of an empirical nature since neither cell mass nor organic matter are modeled. Hence, for control purposes, the parameters in these expressions must be updated periodically based on current process information.

$$\frac{dC_{a1}}{dt} = \zeta_{1,5,6} \cdot D \cdot \left(C_a^{AN} - C_{a1}\right) + \zeta_{3,4} \cdot D \cdot (C_{a2} - C_{a1}) - \zeta_{O_2} \cdot r_n \qquad (1)$$

$$\frac{dC_{a2}}{dt} = \zeta_{2,3,4} \cdot D \cdot \left(C_a^{AN} - C_{a2}\right) + \zeta_{1,6} \cdot D \cdot (C_{a1} - C_{a2}) - \zeta_{O_2} \cdot r_n \qquad (2)$$

$$\frac{dC_{n1}}{dt} = -\zeta_{1,5,6} \cdot D \cdot C_{n1} + \zeta_{3,4} \cdot D \cdot (C_{n2} - C_{n1}) + \zeta_{O_2} \cdot r_n - \left(1 - \zeta_{O_2}\right) \cdot r_d \qquad (3)$$

$$\frac{dC_{n2}}{dt} = -\zeta_{2,3,4} \cdot D \cdot C_{n2} + \zeta_{1,6} \cdot D \cdot (C_{n1} - C_{n2}) + \zeta_{O_2} \cdot r_n - \left(1 - \zeta_{O_2}\right) \cdot r_d \qquad (4)$$

Neither the settler nor the anaerobic zone are explicitly described in Eqs. (1) to (4). It is assumed here that no appreciable denitrification occurs in the settler, and that the settler can be described as a well mixed vessel with respect to soluble components. In this case, effluent concentrations can be calculated based on the dynamics of tanks T1 and T2. The NO_x-N content of the inlet sewage water is negligible, and any NO_x-N recycled to the anaerobic zone from the settler is denitrified completely within this zone. Hence the NO_x-N concentration leaving the anaerobic zone is identically zero. Furthermore, the NH_4-N entering with the inlet sewage water (generally in the range of 30 to 50 mg/liter) is far greater than that which is recycled to the anaerobic zone from the settler (normally less than 2 mg/liter). The nitrogen dynamics in the entire process can therefore be described to a good approximation by neglecting the effect of the two tanks' output to the settler on the two tanks' input from the anaerobic zone, or equivalently, by considering only the dynamics of the two tanks as done in Eqs. (1) to (4).

3. COMPUTATIONAL ASPECTS

The model given by Eqs. (1) to (4) is inherently nonlinear due to the dependency of ζ_{O2} on the NH_4-N concentration. An important distinction to be made regarding computational effort is whether the model is linear or nonlinear with respect to the reaction kinetics. The former case reduces to the solution of a set of algebraic equations as illustrated below. For the latter case, more computationally intensive nonlinear integration routines are required. There are two distinct manners in which the model can be employed, either as a prediction model in control algorithms or as a steady state (i.e. limit cycle) model for process analysis. The prediction model involves a forward simulation starting from known (measured) concentrations in both tanks and given (estimated) rate expressions. All or part of the Eqs. (1) to (4) have been employed as a prediction model in three control schemes to date: To determine the desired denitrification rate in a strategy involving the controlled addition of an external organic source to the denitrifying zone [3]; to determine the desired nitrification rate in a strategy involving control of the dissolved oxygen concentration setpoint [4]; and to determine the desired cycle length [6]. The solution to the prediction model is straightforward and has been shown for the case of zero order kinetics and the 4 phase operating schedule in [4].

The steady state model describes the situation when process conditions have been constant long enough so that the nitrogen dynamics have attained a limit cycle. In this case only one of the two tanks need be considered (here, tank T2), since the dynamics in one tank will be identical to the other tank delayed by half a cycle. The limit cycle solution is found by solving the equations for one complete cycle in an iterative fashion until initial and final concentrations are identical. Since tank T2 is decoupled from tank T1 in phases 2 through 5, phase 2 is used as a starting point for each iteration. In phases 6 and 1, the already calculated solutions of tank T2 in phases 3 and 4, respectively, are taken as the solutions for tank T1. For each iteration, the time point of stopping aeration must be found. In the case that zero order kinetics are employed for denitrification, appropriate steps must be taken to avoid unrealistic negative concentrations.

phase	NH_4-N equations		NO_x-N equations	
2	$C_{a2}^{t=t_2} = C_{a2}^{t=t_1} \cdot \Phi(t_2 - t_1) + C_a^{AN} \cdot \Gamma(t_2 - t_1)$	(5)	$C_{n2}^{t=t_2} = C_{n2}^{t=t_1} \cdot \Phi(t_2 - t_1) - \frac{r_d}{D} \cdot \Gamma(t_2 - t_1)$	(11)
3	$C_{a2}^{t=t_3} = C_{a2}^{t=t_2} \cdot \Phi(t_3 - t_2) + C_a^{AN} \cdot \Gamma(t_3 - t_2)$	(6)	$C_{n2}^{t=t_3} = C_{n2}^{t=t_2} \cdot \Phi(t_3 - t_2) - \frac{r_d}{D} \cdot \Gamma(t_3 - t_2)$	(12)
4	$C_{a2}^{t=t_4} = C_{a2}^{t=t_3} \cdot \Phi(t_4 - t_3) + \left(C_a^{AN} - \frac{r_n}{D}\right) \cdot \Gamma(t_4 - t_3)$	(7)	$C_{n2}^{t=t_4} = C_{n2}^{t=t_3} \cdot \Phi(t_4 - t_3) + \frac{r_n}{D} \cdot \Gamma(t_4 - t_3)$	(13)
5	$C_{a2}^{t=t_5} = C_{a2}^{t=t_4} - r_n \cdot (t_5 - t_4)$	(8)	$C_{n2}^{t=t_5} = C_{n2}^{t=t_4} + r_n \cdot (t_5 - t_4)$	(14)
6a	$C_a^{sp} = C_{a2}^{t=t_5} \cdot \Phi(t_n - t_5) + \left(C_a^{AN} - \frac{r_n}{D}\right) \cdot \Gamma(t_n - t_5)$	(9a)	$C_{n2}^{t=t_n} = C_{n2}^{t=t_5} \cdot \Phi(t_n - t_5) + \frac{(r_n - r_d)}{D} \cdot \Gamma(t_n - t_5)$	(15a)
	$+ D \cdot \left(C_a^{t=t_2} - C_a^{AN}\right) \cdot (t_n - t_5) \cdot \Phi(t_n - t_5)$		$+ \left(D \cdot C_{n2}^{t=t_2} + r_d\right) \cdot \tau \cdot \Phi(t_n - t_5)$	
6b	$C_{a2}^{t=t_6} = C_a^{sp} \cdot \Phi(t_6 - t_n) + C_a^{AN} \cdot \Gamma(t_6 - t_n)$	(9b)	$C_{n2}^{t=t_6} = C_{n2}^{t=t_n} \cdot \Phi(t_6 - t_n) - \frac{2r_d}{D} \cdot \Gamma(t_6 - t_n)$	(15b)
	$+ D \cdot \left(C_a^{t=t_2} - C_a^{AN}\right) \cdot (t_6 - t_n) \cdot \Phi(t_6 - t_5)$		$+ \left(D \cdot C_{n2}^{t=t_2} + r_d\right) \cdot (t_6 - t_n) \cdot \Phi(t_6 - t_5)$	
1	$C_{a2}^{t=t_1} = C_{a2}^{t=t_6} \cdot \Phi(t_1) + \left(C_a^{AN} - \frac{r_n}{D}\right) \cdot \Gamma(t_1)$	(10)	$C_{n2}^{t=t_1} = C_{n2}^{t=t_6} \cdot \Phi(t_1) + \frac{(r_n - r_d)}{D} \cdot \Gamma(t_1)$	(16)
	$+ D \cdot \left(C_a^{t=t_3} - C_a^{AN} + \frac{r_n}{D}\right) \cdot t_1 \cdot \Phi(t_1)$		$+ \left(D \cdot C_{n2}^{t=t_3} - r_d\right) \cdot t_1 \cdot \Phi(t_1)$	

in the above, $\Phi(\tau) = \exp(-D \cdot \tau)$ and $\Gamma(\tau) = 1 - \exp(-D \cdot \tau)$

The steady state 6 phase solution to Eqs. (1) through (4) is illustrated in Eqs. (5) to (16) for the case of zero order kinetics, where r_n and r_d are constants. The times t_n and t_1 through t_6 denote when aeration in tank T2 is stopped and the ends of phases 1 through 6, respectively. C_a^{sp} is the setpoint concentration of NH_4-N at which aeration is stopped. Eqs. (5) to (16) specifically apply for when t_n occurs in phase 6 and when NO_x-N does not disappear at any time. If t_n occurs elsewhere or if NO_x-N disappears completely before aeration is due to resume, Eqs. (5) to (16) will take on a different form. Since Eqs. (5) to (10) for NH_4-N are decoupled from Eqs. (11) to (16) for NO_x-N, the solution is found more efficiently by first finding the limit cycle for NH_4-N.

4. ZERO ORDER VS. NONLINEAR KINETICS

The reaction rates in the pilot scale process appear to be well described by the Monod type limitation kinetics given by Eq. (17) for nitrification and Eq. (18) for denitrification, with half saturation constants lying in the range of 0.1 to 0.8 mg/liter for k_{NH4} and 0.1 to 0.2 mg/liter for k_{NOx}. This is illustrated in the comparison between calculated and measured concentrations shown in Fig. 2. A precise value for either half saturation constant is difficult to judge from pilot plant data, as inaccuracies and noise in the measurement system influences strongly their estimation. Hence the rather large range quoted here for k_{NH4}. These value ranges somewhat agree with those suggested in [2].

$$r_n = r_n^{max} \cdot \frac{C_a}{C_a + k_{NH4}} \quad (17)$$

$$r_d = r_d^{max} \cdot \frac{C_n}{C_n + k_{NOx}} \quad (18)$$

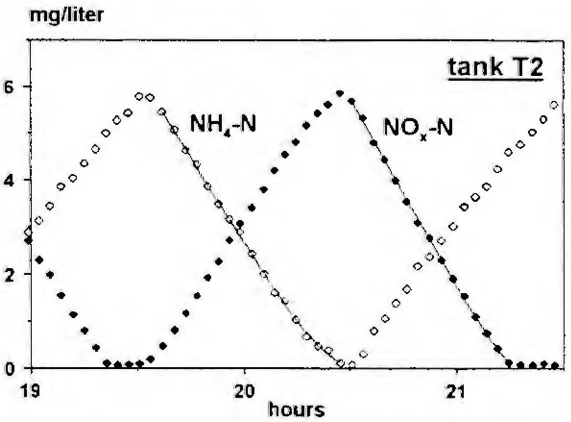


Fig. 2. Pilot plant measurements (symbols) and model simulations using Eqs. (17) and (18) with r_n^{max}= 0.13 mg/l/min, k_{NH4}= 0.35 mg/l, r_d^{max}= 0.14 mg/l/min and k_{NOx}=0.15 mg/l. Here, only phases 2 and 5 were implemented in order to isolate tank T2 from tank T1.

Due to the low value range for k_{NOx} and since model predictions are not very sensitive to k_{NOx}, good results can be obtained using zero order kinetics for the denitrification rate (k_{NOx}=0 in Eq. (18)). On the other hand, k_{NH4} has a strong influence on model predictions even when low values are assumed. This is illustrated in Fig. 3 which shows the average effluent concentrations of NH$_4$-N, NO$_x$-N and their sum as a function of the cycle length, t_c, for several values of k_{NH4}. Fig. 3 was produced from limit cycle solutions to Eqs. (1) to (4) using the 4 phase operation schedule and with parameter values as listed in the figure caption.

An important distinction exists between the use of zero order and limitation kinetics for nitrification. As Fig. 3 shows, there is an optimal cycle length, t_c*, which is a function of inlet and process conditions, and which minimizes the steady state average total nitrogen concentration in the effluent, $\bar{C}_{tn}^{out}$. The value of t_c* is the cycle length for which NO$_x$-N just disappears when aeration is due to resume [4]. At greater cycle lengths, NO$_x$-N disappears sooner, and a period of anaerobic conditions occurs. At shorter cycle lengths, denitrification does not come to completion in the time available and NO$_x$-N is still present in the tanks when aeration resumes.

If the nitrification rate is zero order (k_{NH4}=0 in Eq. (19)), $\bar{C}_{tn}^{out}$ is only a weak function of t_c for t_c<t_c*, where the rise in effluent NH$_4$-N with increasing cycle length is almost equal to a corresponding decrease in NO$_x$-N. In this case, operating the process with the minimum feasible cycle length regardless of inlet and process conditions could perhaps be justified, since this would minimize effluent NH$_4$-N while not being very distant from the minimum value of $\bar{C}_{tn}^{out}$.

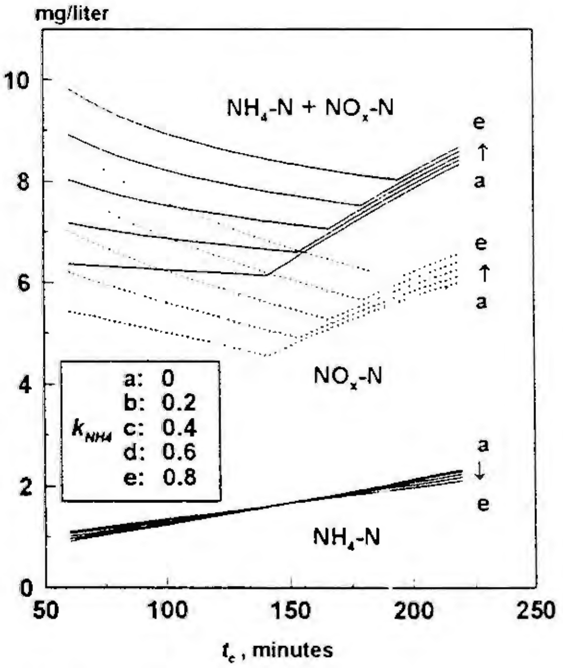


Fig. 3. Average effluent concentrations calculated with D=0.05 min^{-1}, C_i^{AN}=20 mg/l, C_a^{sp}=0.5 mg/l, r_n=0.15 g/l/min, r_c=0.07 g/l/min and k_{NH4} in mg/l as shown.

On the other hand, when limitation kinetics are employed for nitrification, t_c* represents a strong minimum both in the direction of decreasing as well as increasing t_c, and the avoidance of cycle lengths which are too short becomes important. This characteristic becomes more prominent as k_{NH4} increases, and can be explained as follows. The maximum NH$_4$-N concentration attained in the aerobic/anoxic tanks is a monotonic function of the cycle length. Therefore, with rate limitation occurring at low NH$_4$-N concentrations, the nitrification rate averaged over the entire aerobic period is also a monotonic function of the cycle length. Consequently, when denitrification is limited by the time available (i.e. for t_c<t_c*), a progressively lesser fraction of the incoming ammonia can be converted completely to nitrogen gas as the cycle length decreases, and the effluent total nitrogen content rises. Since the aeration time is automatically varied according to when NH$_4$-N in the aerobic/anoxic tanks drops to a fixed setpoint value, k_{NH4} has only a marginal effect on the average effluent NH$_4$-N, with the greatest influence occurring in the average effluent NO$_x$-N.

As Fig. 3 indicates, the value of t_c* and the corresponding minimum value of $\bar{C}_{tn}^{out}$ calculated with the steady state model depend strongly on the value of k_{NH4} employed. The same is true for certain parameters derived using the prediction form of the model. Consequently, in addition to the more easily identifiable maximum reaction rates, a good estimate of k_{NH4} is important. Using a wrong value of k_{NH4} influences model predictions mostly by producing a wrong estimate of the time when anoxic (denitrifying) conditions begin. Hence, in practice, other factors which can lead to a mismatch between the model predicted and actual onset of denitrification (e.g. the transition time between aerobic and anoxic conditions; time delays and errors in the measurement system) should also be accounted for.

A final point to be mentioned here is the choice of the setpoint value C_a^{sp}, which was fixed at 0.5 mg/liter in producing Fig. 3. Increasing (decreasing) C_a^{sp} will tend to counteract the general behavior associated with an increase (decrease) in k_{NH4}. Hence, for a given value of k_{NH4}, process behavior with limitation kinetics will more closely approach that of zero order kinetics as C_a^{sp} increases. However, the concentration of NH_4-N in the effluent is directly proportional to C_a^{sp}, and hence the generally more stringent regulations on effluent NH_4-N content will place an upper limit on the value of C_a^{sp} which can be applied.

5. CORRECTION FACTOR

$$r_n^{cor} = r_n^{max} \cdot \frac{\left(C_a^{max} - C_a^{sp}\right)}{\left(C_a^{max} - C_a^{sp} + k_{NH4} \cdot \ln\left(\frac{C_a^{max}}{C_a^{sp}}\right)\right)} \qquad (19)$$

As indicated above, describing the nitrification rate as zero order when limitation kinetics actually apply can lead to false conclusions regarding process behavior. This is unfortunate since much less computational effort is required by a model which is linear in the kinetics. The zero order kinetic model can, however, provide a reasonably good process description if the maximum nitrification rate, r_n^{max} (for example estimated from data when little limitation occurs), is corrected by a factor according to Eq. (19). r_n^{cor} is the average rate occurring during a batch nitrification (i.e. the nitrifying tank is completely isolated throughout the aerobic period) when the momentary rate is described by Eq. (17). This correction is applied only during phases 1, 5 and 6 for tank T2 (phases 2, 3 and 4 for tank T1), where C_a^{max} is the initial NH_4-N concentration in tank T2 at the start of phase 5 (in tank T1 at the start of phase 2). Due to the inflow of ammonia from the anaerobic zone, the NH_4-N concentration in tank T2 during phase 4 and in tank T1 during phase 1 remains relatively high. Therefore, the nitrification rate in the tanks during these respective phases is described with r_n^{max}, or alternatively, a correction according to Eq. (17) based on the tank's initial NH_4-N concentration when aeration begins. Eq. (19) effectively introduces a nonlinear concentration dependency in the nitrification rate. This, however, is done after the zero order model has been piecewise analytically integrated. For example, r_n^{cor} is substituted for the constant rate r_n appearing in the zero order kinetic model (e.g. Eqs. (5) to (16)) during the calculation procedure, with C_a^{max} set equal to the current value of $C_{a2}^{t=t_4}$.

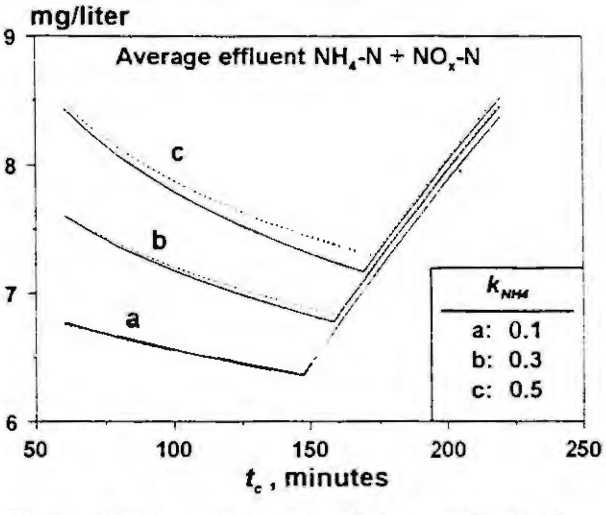


Fig. 4. Steady state model solutions produced from numerical integration of the 4 phase nonlinear kinetic model (r_n described with Eq. (17), dashed curves) and from piecewise analytical integration (r_n described with Eq. (19), solid curves). Employed parameter values were as in Fig. 3.

Fig. 4 shows a comparison between steady state solutions produced from the nonlinear kinetic model and from the zero order kinetic model using Eq. (19). Agreement in the calculated effluent concentration as well as in the optimal cycle length, t_c^*, is good for low values of k_{nh4} but becomes progressively worse as k_{nh4} increases. The discrepancy which occurs is primarily due to neglecting the effects of flow transport in deriving Eq. (19). Of particular significance is the fact that the use of Eq. (19) allows the zero order kinetic model to reflect the behavior caused by rate limitation where there is a significant increase in the average effluent total nitrogen as t_c is reduced below t_c^*.

To point out is the fact that the use of Eq. (19) assumes that good estimates of r_n^{max} and k_{nh4} are available. A potential alternative possibility of using the zero order kinetic model with estimates of the average nitrification rate obtained directly from measurements rather than with Eq. (19) has not been explored by the authors to date.

6. CONCLUSIONS

A model describing the major nitrogen containing components in an alternating type activated sludge process has been presented. Despite its simplicity, the model represents the internal dynamics of this process well, and is currently being used both for process analysis and in control strategies under development. The simplicity of the model avoids many of the problems associated with parameter and state estimation for waste water processes, since the various organic components and bacterial species appearing as states in the more complicated models are not included here. Consequently, the model is not suitable for long term prediction or detailed design. However, applied in an adaptive fashion, by which estimated values of the few parameters contained in reaction rate expressions are periodically updated, the model can be employed in control schemes involving a relatively short (e.g. several hours to a day) prediction window.

Nitrification rate limitation at low ammonia concentrations has been found to influence model predictions significantly. This has been demonstrated here with regards to the optimal cycle length and the total effluent nitrogen concentration determined from steady state model solutions. This limitation should therefore be accounted for even at low values of the half saturation constant in a Monod type reaction rate expression. The computational effort associated with limitation kinetics is, however, far greater than a model employing zero order kinetics, which can be piecewise integrated to form of a set of algebraic equations. To overcome this problem a correction to the nitrification rate occurring in the absence of limitation has been proposed, which allows solutions close to those corresponding to limitation kinetics to be obtained with essentially the same computational effort as with zero order kinetics.

7. LIST OF SYMBOLS

C_{a1}	NH_4-N in tank T1, *mg/l*	k_{nox}	NO_x-N half saturation constant, *mg/l*
C_{a2}	NH_4-N in tank T2, *mg/l*	r_d	denitrification rate, *mg/l/min*
C_{n1}	NO_x-N in tank T1, *mg/l*	r_n	nitrification rate, *mg/l/min*
C_{n2}	NO_x-N in tank T2, *mg/l*	r_n^{cor}	corrected nitrification rate, *mg/l/min*
C_a^{AN}	NH_4-N out of anaerobic zone, *mg/l*	r_n^{max}	maximum nitrification rate, *mg/l/min*
C_a^{sp}	NH_4-N setpoint value, *mg/l*	t_1 - t_6	end of phase times, *minutes*
$\bar{C}_{nt}^{out}$	average effluent total nitrogen, *mg/l*	t_c	cycle length, *minutes*
D	dilution rate, *min^{-1}*	t_n	time of stopping aeration, *minutes*
k_{nh4}	NH_4-N half saturation constant, *mg/l*	ζ	switching function (Eqs. 1 to 4)

8. REFERENCES

[1] Arvin, E. Biological Removal of Phosphorous from Wastewater. *CRC Critical Rev. Environ. Control* 15 (1985) 25-64.

[2] Henze M., Grady C.P.L.(Jr), Gujer W., Marais G v R and Matsuo T. Activated sludge model no.1, *IAWPRC Scientific and technical report No.1,* (1986) IAWPRC, London.

[3] Isaacs, S.H., Zhao, H., Henze, M. Soeberg, H., Kümmel, M. Activated Sludge Nutrient Removal Process Control By Carbon Source Addition, *Proceedings of the 12th World Congress of the International Federation of Automatic Control (IFAC),* Sydney, 19.-23. July, 1993.

[4] Isaacs, S., Kümmel, M. Dissolved Oxygen Setpoint Control of an Alternating Activated Sludge Process. *Proceedings of the 7th Forum for Applied Biotechnology,* Gent, Belgium, September 30 - October 1. To appear in: *Mededelingen van de Faculteit Landbouwwetenschappen, University of Gent,* Coupure 653, B9000 Gent, Belgium, 1993.

[5] Wentzel M.C. Ekama G.A. and Marais G.v.R.. Processes and modelling of nitrification denitrification biological excess phosphorus removal system - A review. *Wat. Sci. Techn.* 25 (1992) 59-82.

[6] Zhao H., Isaacs S.H., Soeberg, H., Kümmel, M. A Novel Control Strategy for Improved Nitrogen Removal in an Alternating Activated Sludge Process. Part I: Process Analysis; Part II: Control Development. To appear in: *Water Research* (1993).

MATHEMATICAL MODELLING OF BAKER'S YEAST PRODUCTION IN AN AIRLIFT TOWER-LOOP REACTOR

K.-H. Bellgardt, G. Strauß
Institute for Chemical Engineering, University of Hannover, Callinstr. 3, D-30167 Hannover

1. Introduction

The kinetics of biotechnological processes are determined by the properties of both, the microorganisms and the reactor. Many effects which can be important for large-scale production processes cannot be covered by unstructured, lumped models. In this paper, a model for baker's yeast fed-batch production in an airlift tower loop reactor is presented. The model is then used to analyse the process and possible directions for optimization.

3. Reactor Model

The reactor model has to describe the time and space-dependent concentrations in the gas and liquid phase of the reactor as a function of initial conditions, manipulating variables, and biological reactions of the yeast cells. The layout of the reactor and the structure of the model are shown in Fig. 1. A distributed model was established under the following simplifying assumptions [7]:

> The hydrodynamics are in steady state, there is churn-turbulent flow, ideal mixing in radial direction, no dispersion in the gas phase, no dispersion over system boundaries of riser, downcomer and head, and a linear pressure drop over the height of the reactor.

The liquid-phase dispersion model for the main components cell mass, substrate molasses, product ethanol, and dissolved oxygen is, in general form, for the concentration c of a component i in the reactor segment j:

$$\begin{aligned}(1-\epsilon_j)\cdot\frac{\partial c_{ij}}{\partial t} &= -u_{lj}\frac{\partial c_{ij}}{\partial z} + (1-\epsilon_j)\cdot(k_l a)_{ij}\cdot(k_{Hi}\cdot P_j\cdot x_{ij}-c_{ij}) \\ &+ E_{lj}\cdot[\frac{\partial A_j}{\partial z}\cdot\frac{1-\epsilon_j}{A_j}\cdot\frac{\partial c_{ij}}{\partial z} - \frac{\partial \epsilon_j}{\partial z}\cdot\frac{\partial c_{ij}}{\partial z} + (1-\epsilon_j)\cdot\frac{\partial^2 c_{ij}}{\partial z^2}] \\ &- (1-\epsilon_j)\cdot c_{Xj}\cdot q_{ij} + \frac{Q_{lFj}\cdot c_{ie}}{H_{Fj}\cdot A_j} - \frac{Q_{lzj}\cdot c_{ij}}{H_j\cdot A_j}\end{aligned} \tag{1}$$

The time and space dependency of the variables was not explicitly stated for sake of simplicity. The boundary conditions are of Danckwert's type. The following boundary conditions were used to describe the coupling of the three subsystems of the reactor at the top of the riser:

$$\frac{u_{lr}(0)}{E_{lr}\cdot(1-\epsilon_r(0))}\cdot(c_{id}(0) - c_{ir}(0)) + \frac{\partial c_{ir}(0)}{\partial z} = 0 \tag{2}$$

$$\frac{u_{ld}(H_d)}{E_{ld}\cdot(1-\epsilon_d(H_d))}\cdot(c_{ik}(0) - c_{id}(H_d)) + \frac{\partial c_{id}(H_d)}{\partial z} = 0 \tag{3}$$

$$\frac{Q_{lu}}{E_{lk}\cdot(1-\epsilon_k(0))\cdot A_k(0)}\cdot(c_{ir}(H_r) - c_{ik}(0)) + \frac{\partial c_{ik}(0)}{\partial z} = 0 \tag{4}$$

In airlift reactors, the hydrodynamics and flow characteristics are controlled by the aeration rate which, therefore, also influences the mass transfer and mixing properties. Some of these parameters were measured by our co-workers [5,6]. The following correlations were used for the dispersion coefficients, transport parameters, and gas holdup:

$$E_l = 1.23\cdot d^{1.5}u_g{}^{0.5} \tag{5}$$

Towel & Ackerman [3]

$$\frac{(k_l a)_{Or}\cdot H_d}{u_{lr}} = 2\cdot\left[\frac{u_{gr}}{u_{lr}}\right]^{0.87}\cdot\left[1 + \frac{A_d}{A_r}\right]^{-1} \tag{6}$$

Bello et al. [2]

$$(k_1a)_{Od} = 0.89\cdot(k_1a)_{Or} \tag{7}$$

$$(k_1a)_{Ok} = 0.011\cdot u_g^{0.82} \qquad \text{Kastanek [3]} \tag{8}$$

$$\epsilon_r(z) = \frac{u_{gr}(z)}{0.24 + 1.35\cdot(u_{gr}(z) + u_{lr})^{0.93}} \qquad \text{Hills [4]} \tag{9}$$

$$\epsilon_d(z) = 0.89\cdot\epsilon_r(z) \qquad \text{Bello et al. [2]} \tag{10}$$

$$\epsilon_k(z) = 0.63\cdot u_{gk}(z)^{0.775} \qquad \text{Weiland [8]} \tag{11}$$

The interesting components of the gas phase are oxygen and carbon dioxide which are advantageously described by their mole fractions. The model equations of the gas phase are

$$\epsilon_j\cdot\frac{P_j}{R\cdot T}\cdot\frac{\partial x_{gij}}{\partial t} = -u_{gj}\frac{P_j}{R\cdot T}\cdot\frac{\partial x_{gij}}{\partial z} - (1-\epsilon_j)\cdot(k_1a)_{ij}\cdot(k_{Hi}\cdot P_j\cdot x_{gij}-c_{ij}) \tag{12}$$

The cell model for the yeast Saccharomyces cerevisiae determines the metabolic activity, expressed by

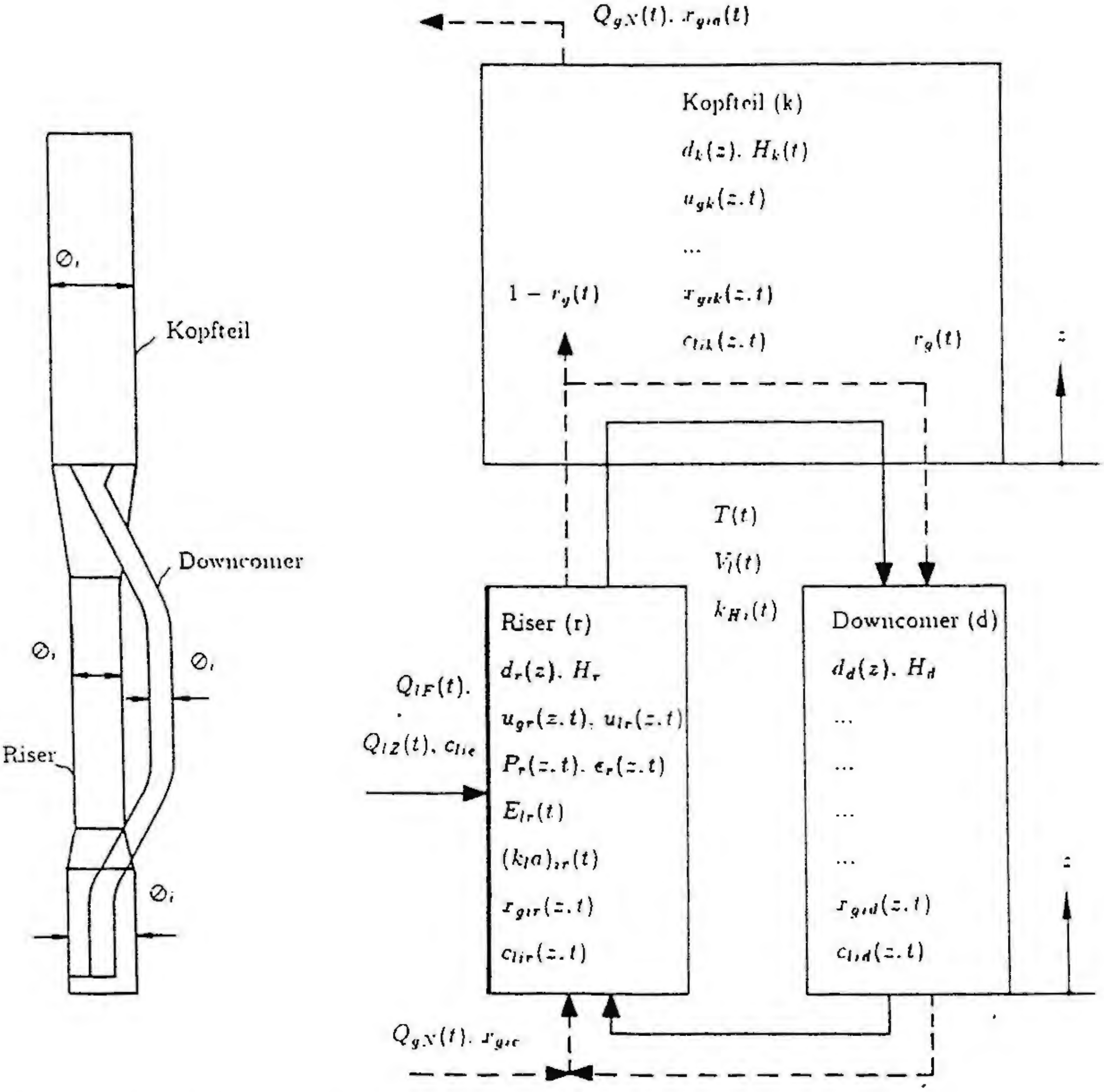


Fig. 1 Schematic diagram of the fed-batch airlift reactor (2 m^3 pilot plant, reactor high 24 m) and schematical representation of the subsystems riser, downcomer, and head

the specific reaction rates $q_{i,j}$ in Eq. 1 as a function of the liquid phase concentrations. For this study, the model of Bellgardt was used [1]. For fed-batch production, the long-term regulation of the pathways for gluconeogenesis and respiration can be neglected.

4. Simulation studies of the fed-batch airlift reactor

After verification of the model with experimental data, simulation studies for the pilot plant were carried out to investigate the influence of non-ideal mixing and concentration gradients. Some results are shown in Figs. 2 and 3. The concentration profiles (not given) are relatively flat and the cell mass could be considered as a lumped variable. Nevertheless, the yeast metabolism switches from fermentative with ethanol production to oxidative with ethanol uptake during each circulation. Under the given operating conditions, ethanol is mainly produced in the head space of the reactor. In downcomer and riser there are both, ethanol production and uptake (Fig. 3); ethanol is produced in the upper parts, where the oxygen transfer is lowest. At these points there is also a slightly higher sugar concentration due to distributed feeding over the height of the reactor. In in the lower parts of the system ethanol uptake can be found. It is clear that such a mixed metabolism influences significantly the overall kinetics of the process.

The analysis reveals that for the given system substrate feeding in the riser is not optimal, because the substrate concentration is highest just in those parts of the reactor where the oxygen supply is low. This limits the productivity, because the substrate flow rate has to be kept low to avoid loss of yield. It can be concluded that, here, a substrate feeding in the downcomer would be favourable. This was confirmed in simulation studies.

5. List of Symbols

A	Area of reactor, m^2	k_H	Henry constant, $mole(Nm)^{-1}$	T	Temperature, K
c	Concentration, kgm^{-3}	k_la	Volumetric mass transfer coefficient, h^{-1}	u	Velocity, ms^{-1}
d	Diameter, m			x	Mole fraction
E_l	Dispersion coefficient, ms^{-2}	P	Pressure, Pa	t	Time variable, h
H	Height, m	q,r	Specific reaction rate , h^{-1}	z	Location variable, m
OTR	Total Oxygen transfer rate, $gm^{-3}h^{-1}$	Q	Flow rate, m^3h^{-1}	ϵ	Relative gas holdup
		R	Gas constant, $Jmol^{-1}K^{-1}$		

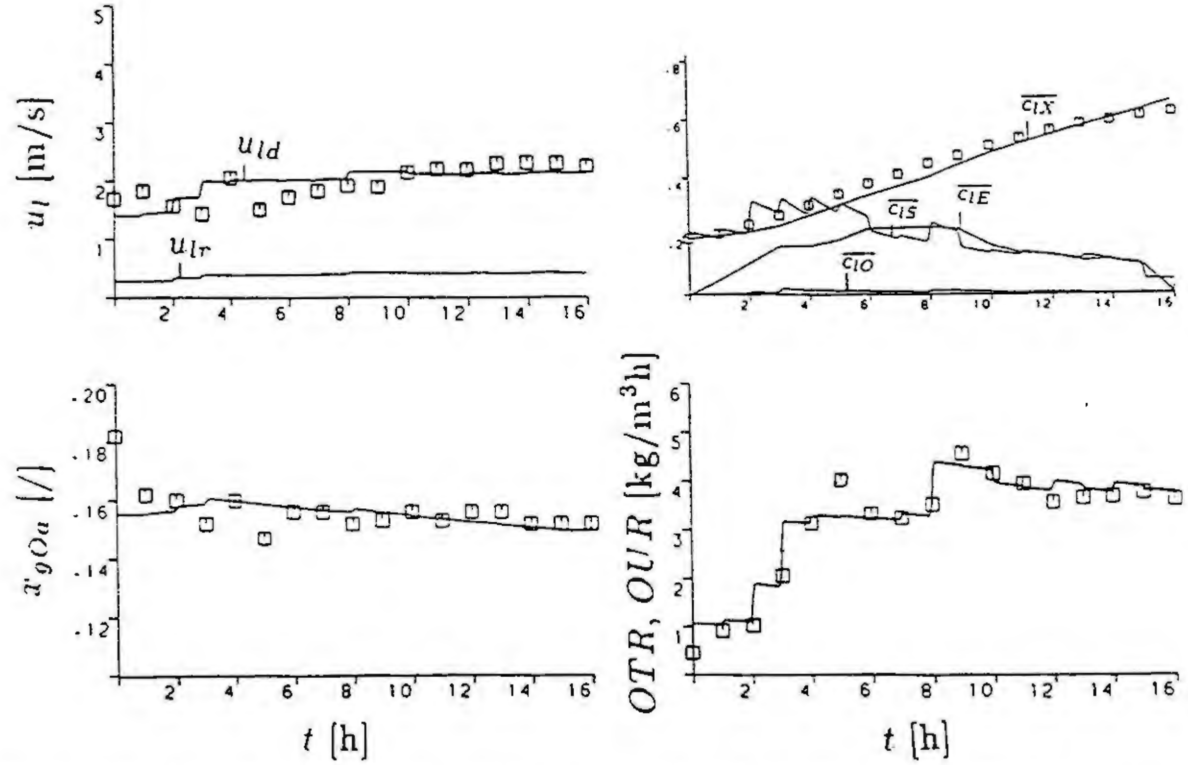


Fig. 2 Simulation (lines) and experimental data (symbols) for a fed-batch cultivation ($\overline{}$ = Averaged value). Scaling: c_O = 0.5mmol/l, c_S = 2mmol/l, c_E = 0.5 mol/l, c_X = 100 g/l

Subscripts

d	Downcomer	i	Substance = (E,O,S,X)	r	Riser
e	Substrate inflow	j	Reactor subsystem = (d,k,r)	S	Substrate
E	Ethanol	l	Liquid phase of the reactor	X	Cell mass
F	Substrate feed	k	Head space	Z	Media additives
g	Gas phase	O	Oxygen		

6. References

[1] Bellgardt, K.-H., Cell models. In: Rehm, H.J., Reed, G. Pühler, A., Stadler, P. (Eds.), Biotechnology, Vol. 4, VCH Weinheim, 1991.

[2] Bello, R.A., Robinson, C.W., Moo-Young, M., Gas holdup and overall volumetric oxygen transfer coefficients in airlift reactors. Biotech. Bioeng., 27 (1985), 369-381.

[3] Deckwer, W.D., Reaktionstechnik in Blasensäulen. Verlag Salle und Sauerländer,1985.

[4] Hills, J.H., The operation of a bubble column at high throughputs. I) Gas holdup measurements. Chem. Eng. J., 12 (1976), 89-99.

[5] Lübbert, A., Korte, T., Larson, B., Simple measuring techniques for the determination of bubble- and bulk-phase velocities in bioreactors. App. Biochem. Biotech., 14 (1987), 207-219.

[6] Lübbert, A., Larson, B., A new method for measuring local velocities of the continuous liquid-phase in strongly aerated gas-liquid multiphase reactors. Chem. Eng. Tech., 10 (1987), 27-32.

[7] Strauß, G. Model building of growth of baker's yeast in fed-batch air-lift reactors. Thesis, University of Hannover, 1991.

[8] Weiland, P., Untersuchung eines Airlift-Reaktors mit äußerem Umlauf im Hinblick auf seine Anwendung als Bioreaktor. Thesis, University of Dortmund, 1978.

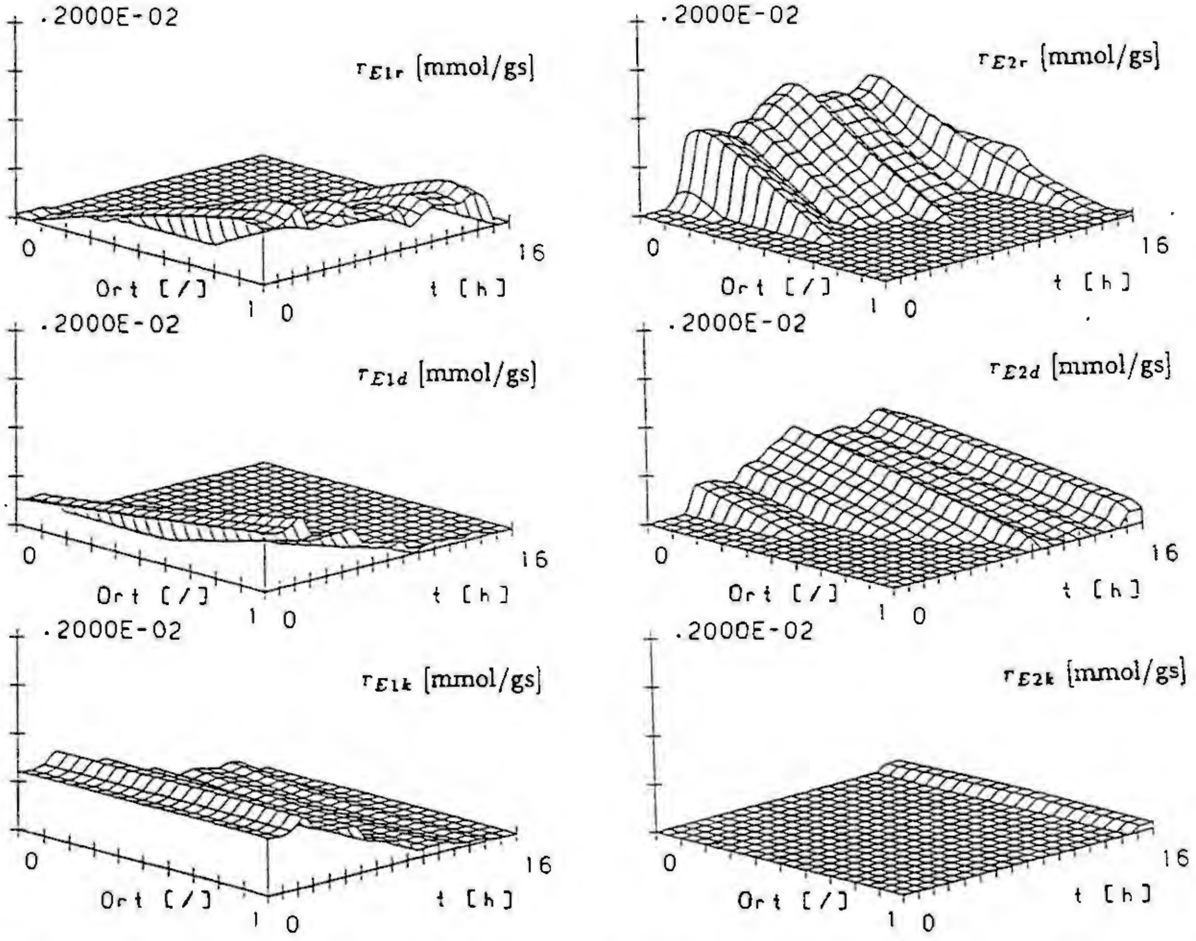


Fig. 3 Simulation results for space and time dependence of the specific ethanol production (left side) and ethanol consumption rate (right side) in riser (top), downcomer (middle) and head (bottom)

A knowledge-based sofware for modelling biochemical processes - a typical application in pH control

Mondher Farza and Arlette Chéruy

Laboratoire d'Automatique de Grenoble, UA CNRS 228, BP 46,
F-38402 Saint Martin d'Hères, France.

Abstract

In this paper, we'll present a knowledge-based software devoted to modelling of biochemical processes. The main functional capabilities of this software are illustrated through a typical example dealing with pH dynamical modelling and control in a microbial culture.

Introduction

Mathematical models are recognized as interesting and useful tools in systems analysis, in process control and in experimental design. Their use in the chemical engineering has met with considerable success, and they are becoming standard tools in process engineering practice. In contrast, the existing mathematical models are rarely fully trusted by investigators in biochemical engineering. This is because the biological behaviour has a complexity unparalleled in other fields: since bioprocesses involve living organisms, their dynamics are often poorly understood, strongly nonlinear and non-stationary. The reproducibility of experiments is uncertain and the model parameters do not remain constant over long periods, due to metabolic variations and physiological modifications. Nevertheless, mathematical models, being the representation of our understanding and the condensed version of our knowledge, are necessary for bioreactor design and the successful formulation of a meaningful process control algorithm so that the final objective of process optimization can be achieved. Therefore, it is interesting to develop software systems for computer-aided modelling devoted to biochemical processes.

The CAMBIO software

CAMBIO, the computer-aided modelling software we propose, is a dedicated workstation providing for easy and interactive modelling and simulation of biochemical processes [4]. It allows expression of the user knowledge in a concrete form which can be built up into a functional diagram. This diagram has to exhibit the most relevant components of the process with their related interactions.

In order to build up this diagram, CAMBIO provides the user with a set of graphical tools and mnemonic icons. It allows the user to choose the elements of the diagram from within menus where graphics symbols are displayed. These symbols are associated to typical components and reactions proposed by CAMBIO. Indeed, CAMBIO classifies all reactants involved in a bioprocess into different typical functions as substrate, biomass, enzyme, etc (Figures 1 and 2).

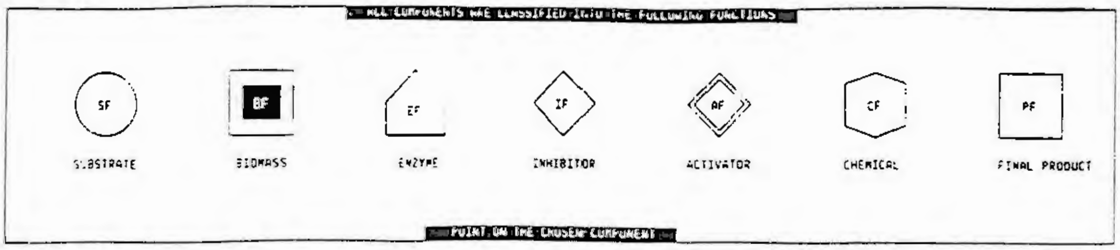

ALL COMPONENTS ARE CLASSIFIED INTO THE FOLLOWING FUNCTIONS
SF
SUBSTRATE
BF
BIOMASS
EF
ENZYME
IF
INHIBITOR
AF
ACTIVATOR
CF
CHEMICAL
PF
FINAL PRODUCT
POINT ON THE CHOSEN COMPONENT

Fig. 1. Typical functions of the relevant components involved in a bioprocess

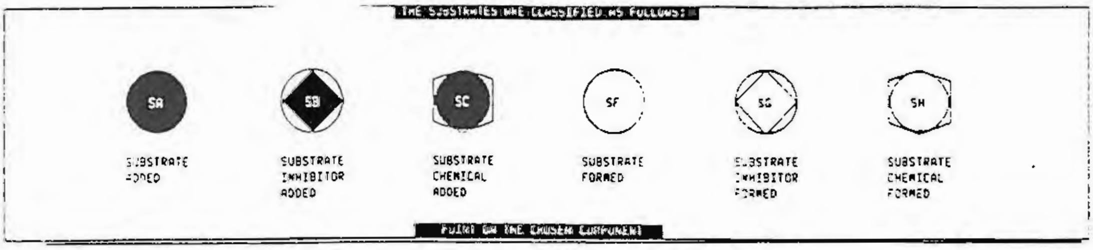

THE SUBSTRATES ARE CLASSIFIED AS FOLLOWS:
SA
SUBSTRATE ADDED
SB
SUBSTRATE INHIBITOR ADDED
SC
SUBSTRATE CHEMICAL ADDED
SF
SUBSTRATE FORMED
SG
SUBSTRATE INHIBITOR FORMED
SH
SUBSTRATE CHEMICAL FORMED
POINT ON THE CHOSEN COMPONENT

Fig. 2. All components involving the substrate function

Similarly, biological and physico-chemical reactions are classified into different types and a graphical symbol is associated to each typical reaction (Figures 3 and 4).

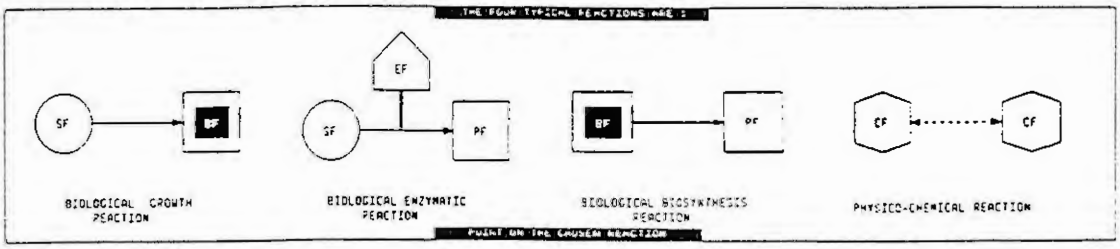

THE FOUR TYPICAL REACTIONS ARE :
SF
BF
BIOLOGICAL GROWTH REACTION
EF
SF
PF
BIOLOGICAL ENZYMATIC REACTION
BF
PF
BIOLOGICAL BIOSYNTHESIS REACTION
CF
CF
PHYSICO-CHEMICAL REACTION
POINT ON THE CHOSEN REACTION

Fig. 3. Typical reactions implied in a bioprocess

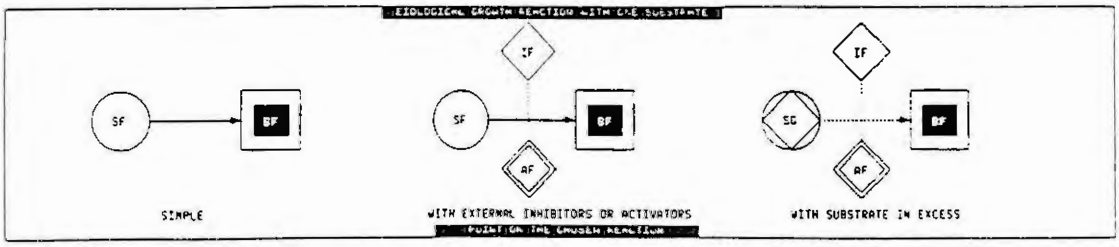

BIOLOGICAL GROWTH REACTION WITH ONE SUBSTRATE
SF
BF
SIMPLE
IF
SF
BF
AF
WITH EXTERNAL INHIBITORS OR ACTIVATORS
IF
SG
BF
AF
WITH SUBSTRATE IN EXCESS
POINT ON THE CHOSEN REACTION

Fig. 4. Different configurations of a biological growth reaction, with one substrate

CAMBIO also offers the possibility of accounting for input and output flow rates of bioreactors (Figure 5). Two kinds of components could be added to the medium: the first consists of components which will be directly implied in a biological reaction (as substrate for example); the second deals with components only implied in acid-base reactions and which are added with a view to pH control [3].

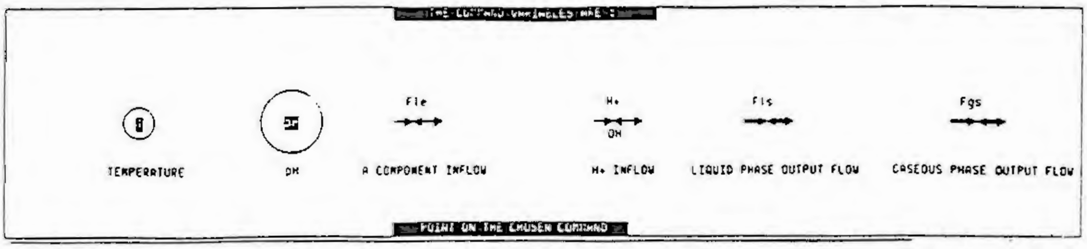

THE COMMAND VARIABLES ARE :
Fle
H+
OH
Fls
Fgs
TEMPERATURE
OH
A COMPONENT INFLOW
H+ INFLOW
LIQUID PHASE OUTPUT FLOW
GASEOUS PHASE OUTPUT FLOW
POINT ON THE CHOSEN COMMAND

Fig. 5. Some command variables proposed by CAMBIO

Thus, the software provides the model designer with all elements he needs to build up a functional diagram: first he has to choose the variables implied in the bioprocess, then organize them into a diagram by using typical reactions. By taking advantage of the knowledge included in the functional diagram, CAMBIO automatically generates the dynamical material balance equations of the process in the form of a mixed algebraic-differential system. Each differential equation corresponds to a material balance associated to a variable which is not an acid neither a base form. The algebraic equations are associated to acid-base reactions which instantaneously reach the steady state, in comparison with biological reactions.

Modelling of a typical biochemical process with CAMBIO

Let's consider an example dealing with a growth reaction i.e. a biomass (BIO) is growing by consuming a substrate (SUB). We suppose that the substrate in the reactor exits in an associated form (ACI) and a dissociated form (BAS) which are respectively an acid and a base form. The predominance of each form varies according to pH. The operating mode of the bioreactor is supposed to be continuous: the reactor is continuously fed with the substrate influent; the outflow rate is equal to the inflow one, and the tank is filled so that the volume of the culture remains constant.
We suppose that substrate is an acid compound and then pH in the reactor becomes lower and lower as the substrate is added. Therefore, in order to keep pH constant, some basic material, in this case NaOH solution, should be added in the reactor.

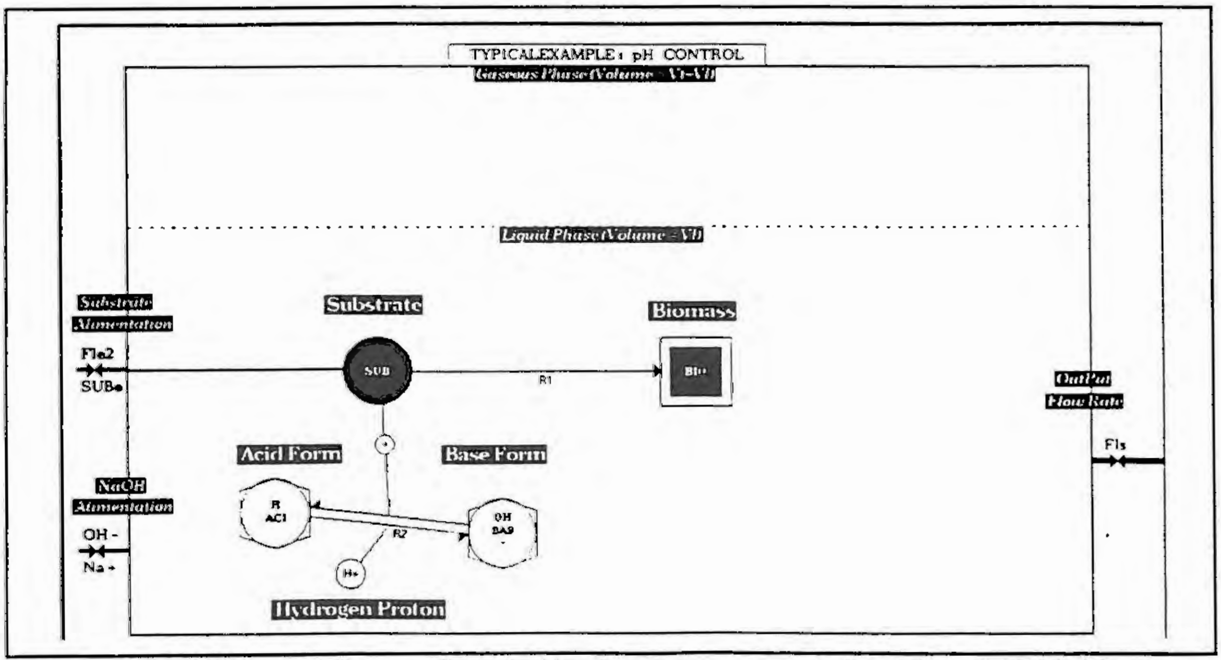


Fig. 6. A functinal diagram of a typical biochemical process as designed through CAMBIO

The functional diagram of such a process as designed with CAMBIO is presented in Figure 6. We note that CAMBIO automatically labelled the acid and base form with the symbols H and OH respectively. Moreover, some symbols are labelled with the signs '+' or '-': these are respectively cations and anions presented in the culture (see [1] and [7]).

From this diagram and by taking advantage of the knowledge it involves, CAMBIO automatically generates a dynamical balance model under the form of mixed differential-algebraic equations. This constitutes a mathematical model framework and can be used for various modelling purposes (simulation, estimation, control,etc). The model associated to the functional diagram of the Figure 6 is generated as follows by CAMBIO:

$$\mathrm{Ka} = \frac{\mathrm{HP}\ \mathrm{BAS}}{\mathrm{ACI}} \qquad (1)$$

$$\mathrm{SUB} = \mathrm{ACI} + \mathrm{BAS} \qquad (2)$$

$$\mathrm{Kw} = \mathrm{HP}\ \mathrm{OH} \qquad (3)$$

$$\mathrm{HP} + \mathrm{Na} = \mathrm{OH} + \mathrm{BAS} \qquad (4)$$

$$\frac{d\mathrm{BIO}}{dt} = \mu\ \mathrm{BIO} - \mathrm{Kd}\ \mathrm{BIO} - \frac{\mathrm{Fle1} + \mathrm{Fle2}}{\mathrm{Vl}}\ \mathrm{BIO} \qquad (5)$$

$$\frac{d\mathrm{SUB}}{dt} = -\frac{\mu}{\mathrm{Y}}\ \mathrm{SUB} - \mathrm{m}\ \mathrm{BIO} + \frac{\mathrm{Fle2}}{\mathrm{Vl}}\ \mathrm{SUBe} - \frac{\mathrm{Fle1} + \mathrm{Fle2}}{\mathrm{Vl}}\ \mathrm{SUB} \qquad (6)$$

$$\frac{d\mathrm{Na}}{dt} = +\frac{\mathrm{Fle1}}{\mathrm{Vl}}\ \mathrm{Nae} - \frac{\mathrm{Fle1} + \mathrm{Fle2}}{\mathrm{Vl}}\ \mathrm{Na} \qquad (7)$$

Equation (1) corresponds to the acid-base dissociation constant; equation (2) states that the substrate SUB is the sum of the acid and the base forms; the electrical equilibrium of water and culture medium are described by equation (3) and (4), respectively. Finally, equations (5), (6) and (7) are the associated material balances to biomass, substrate and Na.

The significance of terms appearing in the generated model is given as follows:

BIO,SUB,Na	: concentration of biomass, substrate and sodium, respectively
BAS, ACI	: concentration of the acid and base form, respectively
HP (OH)	: concentration of hydrogen proton (hydroxylation)
Kw	: water equilibrium constant
Ka	: acid-base dissociation constant relative to ACI and BAS
μ	: specific growth rate of biomass
Kd	: decay rate
m	: maintenance rate
Vl	: liquid volume,of the reactor
Y	: yield coefficient relative to the conversion of substrate into biomass
Fle1	: input flow rate relative to NaOH alimentation
Fle2	: input flow rate relative to substrate alimentation
SUBe	: concentration of the substrate in inlet stream Fle2
Nae	: concentration of Na in inlet stream Fle1

Application in pH control

We will focus in this section on the use of this model in order to design and simulate a pH control. This consists in regulating the pH at a predescribed level pH*, despite the fluctuations of the input substrate concentration, by using the input flow rate relative to NaOH alimentation, Fle1, as control variable.

For simulation purpose, we need an analytical expression of the specific growth rate; CAMBIO offers an interactive user-friendly environment that permits the coding kinetics rates. We will consider the following expression for the growth rate:

$$\mu = \frac{\text{MUMAX SUB}}{\text{KS} + \text{SUB}} \, (\, pH_{max} - pH \,)\,(\, pH - pH_{min}\,) \quad \text{and } pH = -\log_{10}(\text{HP})$$

where:

MUMAX	: maximum specific growth rate of the biomass
KS	: saturation constant
pHmin, pHmax	: constant parameters

Now, we will describe how deriving a linearizing control algorithm directly from the model. From equations (1),(2) and (3), OH and BAS can be explained as functions of HP and SUB and we obtain :

$$\text{OH} = \frac{\text{Kw}}{\text{HP}} \quad (8) \qquad \text{and} \quad \text{BAS} = \frac{\text{Ka SUB}}{\text{HP} + \text{SUB}} \quad (9)$$

By substituting (8) and (9) in (4), we have:

$$\text{HP} = -\text{Na} + \frac{\text{Kw}}{\text{HP}} + \frac{\text{Ka SUB}}{\text{Ka} + \text{HP}} \tag{10}$$

In our application, the values of Ka and pH* are such that the entire quantity of substrate SUB is pratically dissociated and the acid form can be neglected. Thus, equation (10) can be simplified and rewritten as follows:

$$\text{HP} = -\text{Na} + \frac{\text{Kw}}{\text{HP}} + \text{SUB} \tag{11}$$

By differentiating equation (11) with respect to time t, we obtain :

$$\frac{d\text{HP}}{dt} = \frac{-\frac{d\text{Na}}{dt} + \frac{d\text{SUB}}{dt}}{\left(1 + \frac{\text{Kw}}{\text{HP}^2}\right)} \tag{12}$$

with dSUB/dt and dNa/dt given by equations (6) and (7)

We remind that our control objective is the pH regulation by using NaOH alimentation inflow rate as control variable. By applying the principle of linearizing control, the tracking error (pH*-pH) will be governed by a prespecified stable linear differential equation called a reference model [2]. In this example, we select a first order reference model for the regulation error :

$$\frac{d}{dt}(pH^* - pH) + \lambda\,(pH^* - pH) = 0 \tag{13}$$

where λ is a constant positive parameter.

Since dpH*/dt = 0 because pH^* is constant, equation (13) also implies :

$$\frac{dpH}{dt} = -\lambda\,(pH^* - pH) \qquad (14)$$

In term of HP, equation (10) can be rewritten:

$$\frac{dHP}{dt} = K\;\; HP\,(pH - pH^*) \qquad (15)$$

where K is a constant positive parameter (K = λ ln (10))

Now, by subtituting (6), (7) and (15) in equation (12), we obtain:

$$Fle1 = \frac{N}{D} \text{ where}$$

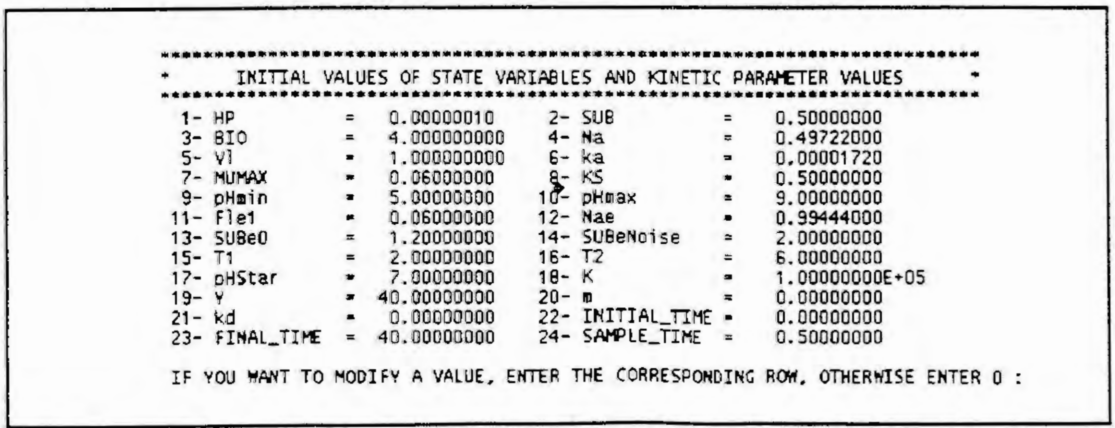
$$N = K\;HP\,(pH - pH^*)\left(1 + \frac{Kw}{HP^2}\right) + \frac{\mu\;BIO}{Y} + \frac{Fle2}{Vl}\;(\,SUB - SUBe - Na\,)$$

$$D = \frac{Na - SUB - Nae}{Vl}$$

We note that Fle1 is a function of Na, SUB, BIO, HP and μ. In practice, all these variables cannot available by measurement but we can cover the lack of measurements by on-line estimation techniques. Let's suppose that only pH is measured. The sodium concentration Na can be direcly deduced by solving equation (7). Then, equation (11) gives an indirect measurement of the substrate SUB. Finally, we can estimate the biomass and the specific growth rate μ from the substrate by using available algorithms described in the litterature [2,3] or some softwares dedicated to bioprocess estimation [4,6].

Simulation results

The controller we propose has been simulated through CAMBIO. The values used in this simulation are given in Figure 7. This Figure corresponds to the screen displayed by CAMBIO before numerical integration of the model.



```
*****************************************************************************
*       INITIAL VALUES OF STATE VARIABLES AND KINETIC PARAMETER VALUES      *
*****************************************************************************
  1- HP            =    0.00000010     2- SUB           =    0.50000000
  3- BIO           =    4.000000000    4- Na            =    0.49722000
  5- Vl            =    1.000000000    6- ka            =    0.00001720
  7- MUMAX         =    0.06000000     8- KS            =    0.50000000
  9- pHmin         =    5.00000000    10- pHmax         =    9.00000000
 11- Fle1          =    0.06000000    12- Nae           =    0.99444000
 13- SUBe0         =    1.20000000    14- SUBeNoise     =    2.00000000
 15- T1            =    2.00000000    16- T2            =    6.00000000
 17- pHStar        =    7.00000000    18- K             =    1.00000000E+05
 19- Y             =   40.00000000    20- m             =    0.00000000
 21- kd            =    0.00000000    22- INITIAL_TIME  =    0.00000000
 23- FINAL_TIME    =   40.00000000    24- SAMPLE_TIME   =    0.50000000

 IF YOU WANT TO MODIFY A VALUE, ENTER THE CORRESPONDING ROW, OTHERWISE ENTER 0 :
```

Fig. 7. Screen displayed before integration routine takes place: values used for simulation

We note that the initial conditions correspond to the steady state on the process; then a square wave of influent substrate has been introduced as a disturbance (Figure 8). The performances of the controller are shown by Figures 9, 10 and 11.

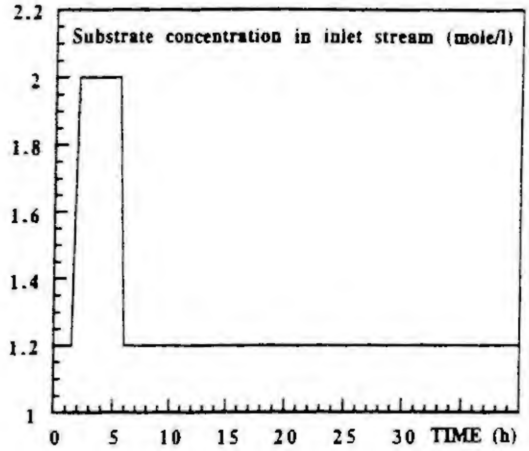


Fig.8. Substrate in inlet stream

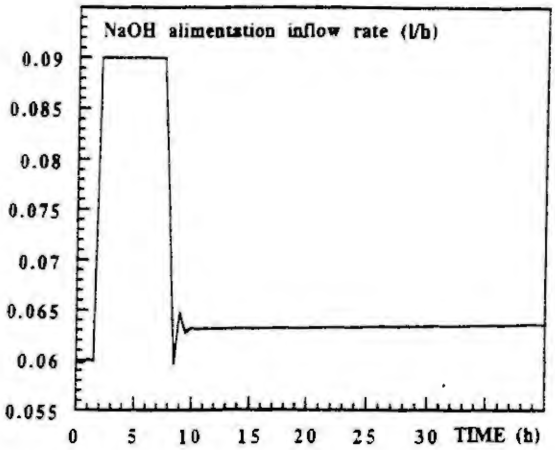


Fig. 9. Inflow rate of NaOH alimentation

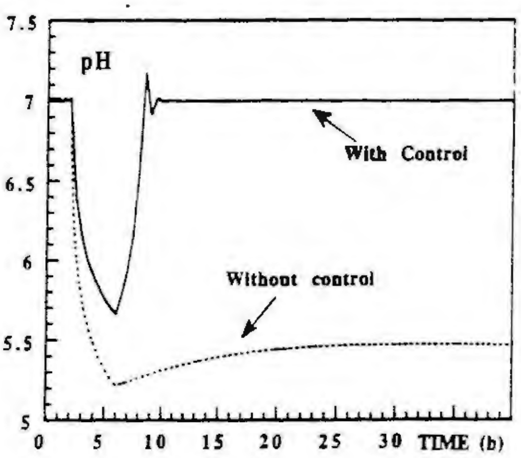


Fig. 10. Evolution of pH

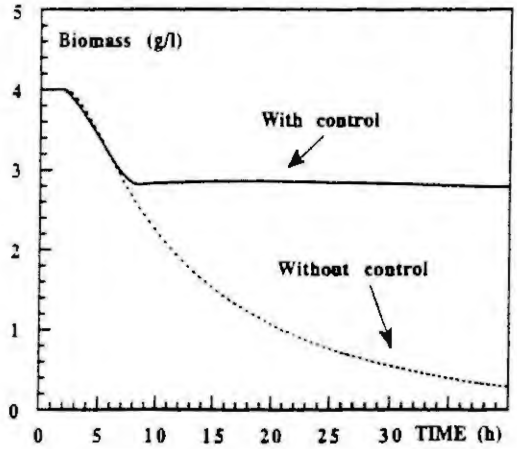


Fig. 11. Evolution of Biomass

Conclusion

The main capabilities of CAMBIO, a knowledge-based sofware for modelling and simulation of bioprocesses, are presented through the modelling and control of a typical biochemical process. CAMBIO allows the user to proceed to a functional analysis of his process in his own technical language, in order to exhibit the relevant variables with their related interactions to be taken into account in the modelling procedure. Indeed, CAMBIO provides the user with a set of design symbols and mnemonic icons in order to interactively design a functional diagram of the process. Then, CAMBIO automatically generates a dynamical balance model by taking advantage of the knowledge included in the diagram. The model can be used in various modelling applications.

Moreover, CAMBIO offers facilities to generate a simulation model (for coding kinetics, introducing auxiliary variables, etc). This model is automatically interfaced with a specialized simulation software which allows visualization of the process dynamical behaviour under various operational conditions, possibly involving feedback control strategies.

References

[1] Bailey, J.E. and Ollis, D.F Biochemical Engineering Fundamentals, 2nd edn., McGraw-Hill, New York, 1986.

[2] Bastin, G. and Dochain, D, On-line estimation and Adaptive control of bioreactors, Elsevier, Amsterdam, 1990.

[3] Bastin, G. and Dochain, D, On-line estimation of microbial specific growth rates, Automatica, Vol. 22, No. 6 (1986), 705-709.

[4] M. Farza, CAMBIO - Récents développements dans l'aide à la modélisation et l'estimation des bioprocédés, PhD thesis, Institut National Polytechnique de Grenoble, Grenoble, France, 1992.

[5] Farza, M. and Chéruy, A., CAMBIO: software for modelling and simulation of bioprocesses, Computer Applications in the BIOSciences, 7 (1991), 327-336.

[6] Farza, M. and Chéruy, A., BIOESTIM - un logiciel pour l'aide à l'estimation des bioprocédés, Récents Progrès en Génie des Procédés. In: Lavoisier (Ed.), Techniques et Documentation, Paris, 1993.

[7] McAvoy, T.J, Hsu, E. and S. Lowenthal, Dynamics of pH in a controlled stirred tank reactor, Ind. Eng. Chem. Process Des. Dev., Vol. 11, No. 1 (1972)

Simulation of Metallurgical Plants with MetaMod

Wilfried Lyhs
Deutsche Voest Alpine Industrieanlagenbau GmbH
Neusserstr. 111, 40219 Düsseldorf / Germany

Abstract

MetaMod is a useful and easy to program tool for the simulation of industrial plants. In the modules of a simulated plant conservation of mass and enthalpy under consideration of equilibria of chemical reactions is automatically done. External programs, which may be user written or standard modules, can be integrated in simulation by adding a simple file interface to them in order to enable data communication with **MetaMod**. This paper presents the concepts of **MetaMod** with a simple example.

Introduction

The Deutsche Voest-Alpine Industrieanlagenbau (**DVAI**) is an engineering company building metallurgical plants such as steel works including rolling mills and melting furnaces all over the world. Together with its Austrian mother company VAI they developed in the middle of the seventies a new process for the production of pig iron from iron ore. In contrary to the well known blast furnace the so called *COREX*-plant is operating without coke and therefore the main advantage of this process is a smaller pollution through offgas and dust.

As most continuous process simulation tools available on the market are designed for the needs of the petrochemical industry they do not satisfy all metallurgical demands. In order to perform layout calculations for metallurgical plants such as *COREX*-plants or to simulate their stationary behaviour in operation DVAI most recently developed a tool called **MetaMod**.

MMCL

The definition of a model is done via a control language called MMCL[1] which is very similar to the control languages of operating systems. Commands formulated in MMCL can be entered interactively or may be written to a *model definition file* and will be interpreted command by command.

The procedure for constructing a model will be illustrated in the following by an example which describes a cyclone and a combustion chamber (cf. *Fig. 1*). With this example the functionality of MMCL-commands will be explained briefly.

In advance here are some common remarks to MMCL syntax. Besides the terms for commands **MetaMod** has several groups of reserved words: auxiliary words, patchwords and technical units.

Patchwords like *are* or *with* do not have a special meaning but contribute to better readability of MMCL-commands and therefore do not alter the interpretation procedure of the commands.

Auxiliary words such as *input, output, temp, press* etc. may occur in a command where they should explain the meaning of user defined words or attributes of streams.

When variables are assigned in a model, different technical units such as *t/h* or *m3/h, kPa* or *bar, MW* or *kJ/h, degC* or *K* can be used.

[1] MMCL: MetaMod Control Language

In order to give a name to the model under which it can be archived in the computer the *model* command is introduced into MMCL.

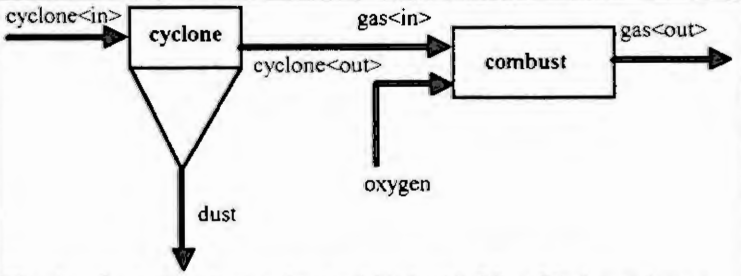


Fig. 1: Part of a plant to be modelled with **MetaMod** consisting of cyclone and combustion camber

model **cyclone+combust;**

Moreover the user has to tell **MetaMod** what chemical elements should be considered in the calculation. This can be done with the *elements* command which for example can look like:

elements are **C O H N Si Fe;**

This model runs with n_{Ele}=6 elements which are carbon, oxygen, hydrogen, nitrogen, silicon, and iron.

The chemical species to be considered in all streams of the model can be declared with the *species* command:

species are **C CO CO2 O2 N2 H2 H2O<g> SiO2[CR]<s> Fe2O3<A><s> Fe<A><s>;**

In our case only n_{Spe}= 10 species are selected from the number of all possible compounds.

In case of species which exist in more than one state of aggregation a suffix in the name of the species like *<s>* or *<g>* indicates solid or gaseous states. Different modifications of a species are distinguished by suffixes like *[CR]* or *<A>*.

Some more user defined attributes such as costs or particle size in the dust load of that gas can be easily added to the streams by the command:

attributes are **costs, fracs<5mm, fracs>5mm;**

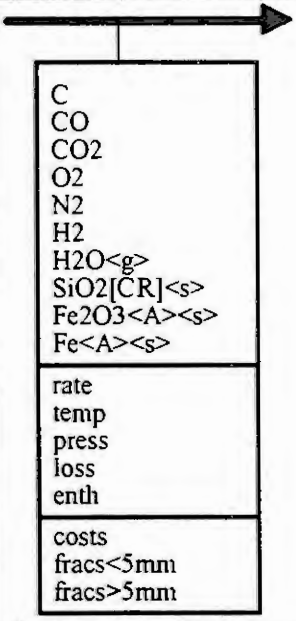


Fig. 2: attributes of streams in model **cyclone+combust**

All streams now consist of three types of attributes: the chemical composition, some standard attributes like flow rate, temperature enthalpy etc and last the user defined attributes. *Fig. 2* shows all attributes of a stream prototype.

Modules which are to be regulated by **MetaMod** can be generated by the *define* command. The module is created simply by typing its name. Moreover its associated input and output streams can be defined in the same statement.

If there are interconnected modules as shown in *fig. 1*, the syntax of the *define* command provides the definition of the network in the following way:

define **combust** ***with input*** **gas<in> = cyclone|cyclone<out>, oxygen**
output **gas<out>;**

A stream named *gas<in>* is the input of the module *combust* and is connected to a stream named *cyclone<out>* coming from a module named *cyclone* modelling a dust separating device.

As *combust* is an internal module, **MetaMod** will automatically solve a set of equations including the conservation of mass and enthalpy.

The *use*-command, which is in its syntax very similar to the *define*-command, has an additional function which is to connect a user written program to a module. Recursive calls of **MetaMod** are also possible with the *use*-command.

use **cyclone *input* cyclone<in>, *output* cyclone<out>, dust = cyc.exe cyclone<in> cyclone<out> dust;**

In the example above the output streams *cyclone<out>* and *dust* are calculated by calling a standard program *cyc.exe* with three parameters identifying the standard input and output of this program with the stream names in **MetaMod**.

The modelling of chemical reactions in thermodynamical equilibrium is done with the *react*-command. For the postcombustion of carbon monoxide in module *combust* $CO + \frac{1}{2}O_2 \Leftrightarrow CO_2$ the corresponding *react*-command is:

react **combust|gas<out>|CO2 - 0.5 O2 -1 CO =kp;**

On the right hand side of the command above the constant of equilibrium **kp** is calculated from a thermodynamical database[2].

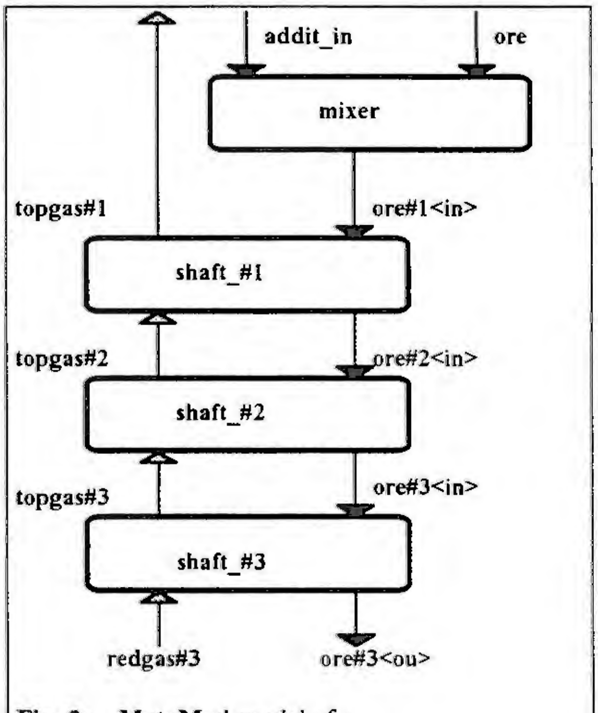


Fig. 3: **MetaMod**-model of a reduction furnace

It is not possible to solve complex calculations of equilibrium composition with the aid of the *react*-command. For those purposes as for the simulation of a reduction shaft furnace shown in *fig. 3* external programs called via the *use*-command may be started.

After all modules and streams are declared **MetaMod** must be instructed which variables are "dependent" and fixed and which are "independent" and therefore should be determined. This can be done with the aid of the *set*-command.

```
set combust|gas<in>|press = 1. bar,
          oxygen|temp   = cyclone<in>,
          oxygen|O2     = 0.95, ...;
```

In the example above the temperature of a stream called *oxygen* is identified with the temperature of a stream *cyclon<in>*, whereas the pressure and the value of chemical analysis O2 are set to fixed values.

The *set*-command provides the retrieval of data from databases e.g. SQL-databases which is important when a **MetaMod**-model is used to control a running plant.

Independent variables can be initialised with starting values by using the *preset*-command which has the same syntax as the *set*-command.

In order to create reports with selected data from calculations the user can define the names of these data within a *report*-command:

report **combust|gas<out>|flow, gas<out>|CO, gas<out>|CO2, cyclon|edust|fracs<5mm;**

[2] Barin, I.: Thermochemical Data of Pure Substances, VCH Verlag, Weinheim 1989

If a mask including text or graphics is created in advance, the denoted data can be copied automatically to it, producing an easy to read report file.

When the model definition file is ready, the MMCL interpreter will produce structure and data files from it which will be used by the calculation part of **MetaMod**. For new calculations with the same model the data file can be modified separately. The data flow for a **MetaMod** calculation is shown in *fig. 4*.

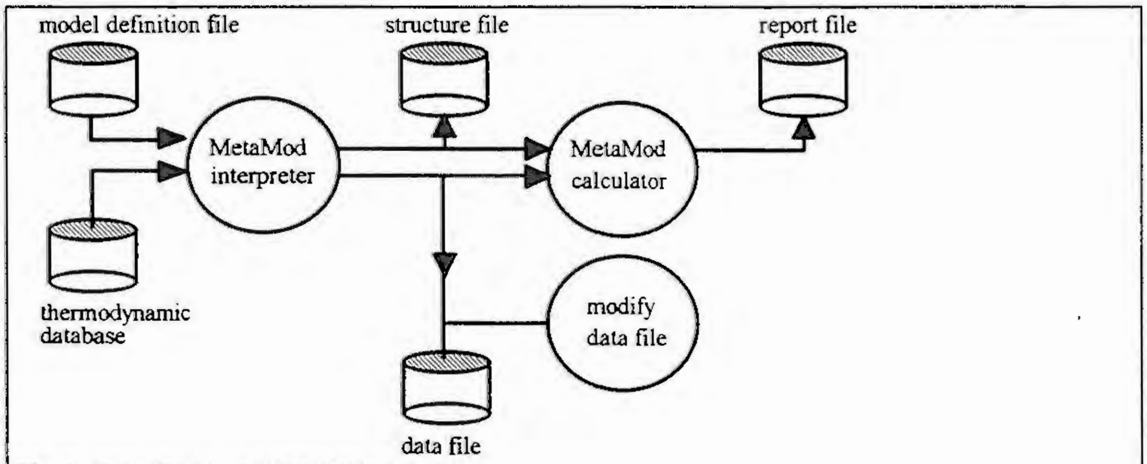


Fig. 4: Data flow in a MetaMod calculation

Summary

MetaMod was designed for the steady state simulation of industrial and especially metallurgical plants. The name **MetaMod** indicates this functionality but gives also a hind to the fact, that a "meta" software is created which can handle existing simulation and standard programs. As a lot of standard modules for cyclones, pressure swing adsorbers, mixers, heat exchangers are adapted ,**MetaMod** is a powerful and easy to handle tool for layout calculations or control purposes within running plants.

Modelling Two Associated Biochemical Pathways

J.-P. Morillon, R. Costalat, N. Burger†, and J. Burger
Institut de biologie théorique, 10 rue Boquel, 49100 Angers (France)
† IRESTE, CP 3003, 44087 Nantes cedex 03 (France)

Abstract. The behaviour of two associated biochemical pathways is modelled by two coupled sets of ordinary nonlinear differential equations. Existence and uniqueness of the steady state are proved. Results on the stability of the system are derived by both analytical and numerical studies. Domains of stability are given for some values of the parameters of the model, and it is proved that the association can increase the domain of stability.

1 Introduction

The most commonly encountered form of regulated biochemical pathway, generally referred to as the Yates-Pardee or Goodwin metabolic pathway, consists of a single pathway of enzymatic reactions, where the last product inhibits the first enzyme (single loop negative feedback) [3] [5]. When the length of a Yates-Pardee metabolic pathway is increased, the stability domain of its unique steady state is decreased [4] [5].

From a different point of view, G. Chauvet has suggested that the association of metabolic pathways can result in an increase in their stability domain [1]; this property can be viewed as non-trivial, because an increase in the complexity of artificial systems often results in a decrease of their stability domain. The question arises whether this unusual property can be verified in large classes of formal or real biological systems.

In the present work, we study the existence, uniqueness, and stability of the steady state of a system of two associated Yates-Pardee metabolic pathways.

2 The model

Let us consider two biological "units" (e.g. cells or organelles), each of which contains a Yates-Pardee metabolic pathway (Fig. 1). Mass-balance and enzyme kinetics lead to write the following dynamical system:

$$\frac{dP_1}{dt} = -\alpha_1 P_1 + f(P_n) + \beta_1 (P_1^* - P_1) \tag{1}$$

$$\frac{dP_i}{dt} = \alpha_{i-1} P_{i-1} - \alpha_i P_i + \beta_i (P_i^* - P_i) \qquad i = 2, \ldots, n \tag{2}$$

$$\frac{dP_1^*}{dt} = -\alpha_1^* P_1^* + f^*(P_n^*) + \beta_1 (P_1 - P_1^*) \tag{3}$$

$$\frac{dP_i^*}{dt} = \alpha_{i-1}^* P_{i-1}^* - \alpha_i^* P_i^* + \beta_i (P_i - P_i^*) \qquad i = 2, \ldots, n \tag{4}$$

where P_i and P_i^* are the respective time-dependent concentrations of a given metabolite in the first and the second unit. The coefficient α_i (resp. α_i^*) is the non-negative kinetic constant of the reaction $P_i \rightarrow P_{i+1}$ in the first unit (resp. $P_i^* \rightarrow P_{i+1}^*$ in the second unit). f and f^* are given reaction functions. In the Yates-Pardee metabolic pathways, f and f^* describe allosteric reactions and can be written as:

$$f(P_n) = \frac{\alpha_0}{1 + K\,(P_n)^{\mu}} \qquad f(P_n^*) = \frac{\alpha_0^*}{1 + K^*\,(P_n^*)^{\mu^*}} \tag{5}$$

where α_0, K, μ, α_0^*, K^* and μ^* are positive constants.
Finally, we assume that association results in passive diffusion between the two units, with constant non-negative coefficients β_i. If all the β_i are zero, the two units are said to be independent or non-associated.

$$\begin{tikzcd}
S_0^* \arrow[r] & P_1^* \arrow[r, "\alpha_1^*"] \arrow[d, shift right, "\beta_1"'] & P_2^* \arrow[r, "\alpha_2^*"] \arrow[d, shift right, "\beta_2"'] & P_3^* \arrow[r, "\alpha_3^*"] \arrow[d, shift right, "\beta_3"'] & P_4^* \arrow[r, "\alpha_4^*"] \arrow[d, shift right, "\beta_4"'] & \cdots \arrow[r, "\alpha_{n-2}^*"] & P_{n-1}^* \arrow[r, "\alpha_{n-1}^*"] \arrow[d, shift right, "\beta_{n-1}"'] & P_n^* \arrow[r, "\alpha_n^*"] \arrow[d, shift right, "\beta_n"'] \arrow[lllllll, dashed, bend right, "\alpha_0^*\ \ \mu^*\ \ K^*"'] & {} \\
S_0 \arrow[r] & P_1 \arrow[r, "\alpha_1"] \arrow[u, shift right] & P_2 \arrow[r, "\alpha_2"] \arrow[u, shift right] & P_3 \arrow[r, "\alpha_3"] \arrow[u, shift right] & P_4 \arrow[r, "\alpha_4"] \arrow[u, shift right] & \cdots \arrow[r, "\alpha_{n-2}"] & P_{n-1} \arrow[r, "\alpha_{n-1}"] \arrow[u, shift right] & P_n \arrow[r, "\alpha_n"] \arrow[u, shift right] \arrow[lllllll, dashed, bend left, "\alpha_0\ \ \mu\ \ K"] & {}
\end{tikzcd}$$

Figure 1. Model of two associated biochemical units.

3 Existence and uniqueness of the steady state

If system (1-4) admits a steady state, it satisfies the following system:

$$(\alpha_1+\beta_1)\,P_1-\beta_1\,P_1^* = f(P_n) \tag{6}$$

$$\alpha_{i-1}\,P_{i-1} = (\alpha_i+\beta_i)\,P_i-\beta_i\,P_i^* \qquad i=2,\ldots,n \tag{7}$$

$$(\alpha_1^*+\beta_1)\,P_1^*-\beta_1\,P_1 = f^*(P_n^*) \tag{8}$$

$$\alpha_{i-1}^*\,P_{i-1}^* = (\alpha_i^*+\beta_i)\,P_i^*-\beta_i\,P_i \qquad i=2,\ldots,n \tag{9}$$

The reaction functions f and f^* are suppose to be continuous, differentiable, and decreasing from $[0,+\infty[$ into $[0,+\infty[$.

3.1 Existence and unicity of the steady state

The system (6-9) can be rewritten, for $\alpha_i \neq 0$ and $\alpha_i^* \neq 0$, $i=1,\ldots,n$, as:

$$M_1\cdot\begin{pmatrix}\alpha_1 P_1\\ \alpha_1^* P_1^*\end{pmatrix}=\begin{pmatrix}f(P_n)\\ f(P_n^*)\end{pmatrix} \quad\text{and}\quad \begin{pmatrix}\alpha_{i-1}P_{i-1}\\ \alpha_{i-1}^*P_{i-1}^*\end{pmatrix}=M_i\cdot\begin{pmatrix}\alpha_i P_i\\ \alpha_i^* P_i^*\end{pmatrix} \qquad i=2,\ldots,n \tag{10}$$

with

$$M_i = \begin{pmatrix}1+\gamma_i & -\gamma_i^*\\ -\gamma_i & 1+\gamma_i^*\end{pmatrix} \qquad \gamma_i=\beta_i/\alpha_i\,,\ \gamma_i^*=\beta_i/\alpha_i^*. \tag{11}$$

So, taking the particular form of M_i into account, P_n and P_n^* are related by a relation that can be written as:

$$\begin{pmatrix}1+d & -d^*\\ -d & 1+d^*\end{pmatrix}\begin{pmatrix}\alpha_n P_n\\ \alpha_n^* P_n^*\end{pmatrix}=\begin{pmatrix}f(P_n)\\ f^*(P_n^*)\end{pmatrix}=M\begin{pmatrix}\alpha_n P_n\\ \alpha_n^* P_n^*\end{pmatrix}. \tag{12}$$

where d and d^* are non-negative constants.
Hence:

$$(1+d)\,\alpha_n\,P_n-d^*\,\alpha_n^*\,P_n^* = f(P_n) \tag{13}$$

$$\alpha_n^*\,P_n^*-f^*(P_n^*)+\alpha_n\,P_n-f(P_n) = 0. \tag{14}$$

Since all coefficients of M^{-1} are positive, the solution (P_n, P_n^*), if it exists, ought to be positive. Let us now put

$$g_n(x) \;=\; \alpha_n\, x - f(x) \qquad \text{and} \qquad g_n^*(x) \;=\; \alpha_n^*\, x - f^*(x).$$

The functions g_n and g_n^* are strictly increasing from $[0, +\infty[$ into $[-f(0), +\infty[$ and $[-f^*(0), +\infty[$ respectively.

Then, with Eq. (14), we have $P_n^* = \varphi(P_n)$, with $\varphi = (g_n^*)^{-1} \circ (-g_n)$ defined on $[0, \bar{x}]$, decreasing with values on $[0, \bar{x}^*]$, with $\bar{x} = (g_n)^{-1}(f^*(0))$ and $\bar{x}^* = (g_n^*)^{-1}(f(0))$.

Let us now define h as :

$$h(x) \;=\; (1+d)\, \alpha_n\, x - d^*\, \alpha_n^*\, \varphi(x) - f(x).$$

The function h is continuous and strictly increasing on $[0, \bar{x}]$ and verifies :

$$\begin{aligned} h(0) &= -d^*\, \alpha_n^*\, \bar{x}^* - f(0) \quad \leq 0 \\ h(\bar{x}) &= (1+d)\, \alpha_n\, \bar{x} - f(\bar{x}) = g_n(\bar{x}) + d\, \alpha_n\, \bar{x} = f^*(0) + d\, \alpha_n\, \bar{x} \quad \geq 0. \end{aligned}$$

The function h equals zero once on $[0, \bar{x}]$. Thus Eq. (13), with $P_n^* = \varphi(P_n)$, admits a unique solution in $[0, \bar{x}]$. We have thus proved that the system (12) admits a unique solution such that $P_n \geq 0$ and $P_n^* \geq 0$. We deduce that the system (6-9) admits a unique steady state such that $P_n \geq 0$ and $P_n^* \geq 0$.

If $f(0)$ and $f^*(0)$ are not simultaneously zero, then the solution $(P_n\,, P_n^*)$ is different from $(0\,,0)$. Moreover, if some α_i equal zero, with $\beta_i \neq 0$, then it can also be proved that the system still admits a positive steady state.

3.2 Positivity of the steady state

Since all coefficients of M_i^{-1}, $i = 1, \ldots, n$, are positive, we deduce that P_1 and P_1^* are positive and from Eq. (10) that all P_i and P_i^* are positive.

Hence, the initial system admits one and only one positive steady solution.

4 Stability analysis

Let us consider the case where the functions f and f^* are defined by Eq. (5). The system is first linearized near the steady state, and the stability of the linearized system is analytically determined by the study of the real part of the eigenvalues. The criterion of Liénard and Chipart [2], applied to the characteristic polynomial, is used for this study, and gives us necessary and sufficient conditions for the stability, i.e. analytical conditions involving the coefficients of the system. In the non-associated case, i.e. when all $\beta_i = 0$, we get explicit conditions for $n \leq 5$. As an example, let us indicate the necessary and sufficient condition for stability for the case where $n = 4$:

$$\alpha_4\, K\, x^{\mu+1} + \alpha_4\, x - \alpha_0 < 0$$

where

$$x = \frac{\alpha_0}{\alpha_4} \left\{ \frac{\mu\, \sigma_4\, s_1^2 - s_1\, s_2\, s_3 + s_3^2 + s_4\, s_1^2}{\mu\, s_4\, s_1^2} \right\}$$

with

$$s_1 = \sum_{i=1}^{n} \alpha_i \qquad\qquad s_k = \sum_{\substack{i_m \neq i_j \\ \forall m \neq j}} \alpha_{i_1}\, \alpha_{i_2} \cdots \alpha_{i_k}$$

In the case of associated units, we get explicit conditions for $n = n^*$ with $n \leq 2$. For greater values of n and n^*, the calculation becomes somewhat tedious. In the particular case where $n = n^* = 1$, the associated system is always stable.

Another method is to compute the eigenvalues of the matrix with scientific software such as MATLAB. This computation requires the knowledge of the steady solution which is calculated by numerically solving the system (6-9). Results are obtained for any value of n and n^*, and it may be noted that no numerical problem generally arises during this computation. However, a systematic study of the stability domains, even for small values of n and n^*, may rapidly become cumbersome.

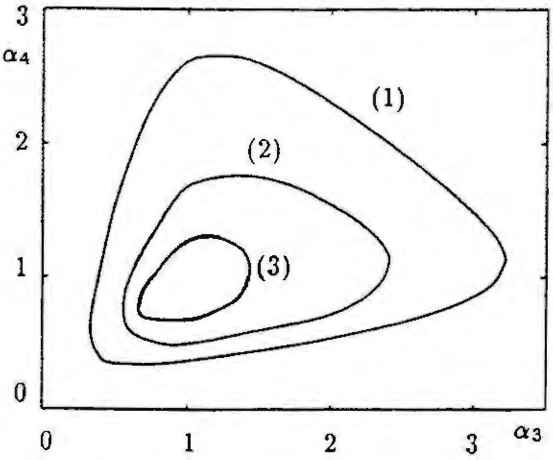


Figure 2. Instability domains.

Fig. 2 shows the instability domains in the plane (α_3, α_4) for two associated units in the case $n = n^* = 4$. The function f corresponds to an allosteric reaction as defined by Eq. (5). The function f^* is considered with $\alpha_0^* = 0$. The values of the coefficients for the curve (1) are $K = 1$, $\mu = 5$, $\alpha_0 = 50$, $\alpha_1 = \alpha_2 = 1$, $\alpha_1^* = \alpha_2^* = 1$, $\alpha_3 = \alpha_3^*$, $\alpha_4 = \alpha_4^*$, and $\beta_i = 0$ for all i. This case corresponds to non-associated units. Curve (2) refers to the same values, except $\beta_3 = 1$, i.e. association occurs. Curve (3) refers to the same values as for curve (2), except $\beta_4 = 1$. The systems are unstable inside the contours. These curves indicate how the association can increase the domain of stability.

A direct solution of the system (1-4) can also be numerically obtained. Apart the fact that such a study may be very long, the question of discerning whether a system is stable or not remains to be answered. However, in all the cases studied, the same results were obtained with the three methods. This means that the local properties derived by linearization are global properties of the system.

5 Conclusion

A formal model of two associated biochemical pathways, consisting of a set of nonlinear coupled first-order ordinary differential equations, has been studied. Existence and uniqueness of the steady state has been proved, so that no bifurcation phenomenon can appear when the Yates-Pardee metabolic pathways are associated. Moreover, analytical and numerical studies of the stability of the system show that the association of two units can lead to an increase of the domain of stability. From a biological point of view, the results suggest that the exchange of matter between compartments (e.g. cells, organelles, ...) may be a source of stability for the cell metabolism.

Acknowledgements. The authors are grateful to the Conseil général du Maine-et-Loire for supporting this research, and to Pr. G.A. Chauvet for helpful discussions.

6 References

[1] Chauvet, G.A., Hierarchical functional organization of formal biological systems: a dynamical approach. I. The increase in complexity by self-association increases the domain of stability of a biological system. Phil. Trans. R. Soc. Lond., B 339 (1993), 425-444.

[2] Gantmakher, F.R., The Theory of Matrices, 2 vols, translated by K.A. Hirsch, Chelsea Pub. Co, 1959.

[3] Goodwin, B.C., Analytical Physiology of the Cells and Developing Organisms. Academic Press, London, 1976.

[4] Rapp, P., Analysis of biochemical phase shift oscillators by a harmonic balancing technique. J. Math. Biol., 3 (1976), 203-224.

[5] Walter, C.F., Kinetic and thermodynamic aspects of biological and biochemical control mechanisms. In: Biochemical regulatory mechanisms in eukaryotic cells, Emest and Santiago Grisolia (ed.), Wiley-Interscience, 1972, 355-489.

CALCULUS OF ITERATIONS AND DYNAMICS OF THE PHYSICOCHEMICAL REACTIONS

V. Gontar

International Group for Scientific and Technological Chaos Studies
Ben-Gurion University of the Negev, P.O. Box 1025,
Beer-Sheva 84110, Israel

ABSTRACT

The analogy between the basic relationships of the π-heorem of the theory of dimensionality and the principle of minimum free energy for complex chemical equilibrium is the basis for formulating new extreme principles and mathematical models for describing chemical reaction kinetics and chemical reactions dynamics.

The monopoly of differential equations for describing all kinds of dynamics was broken by these results, yielding algebraic equations and iterational maps (calculus of iterations) for simulation the temporal and spatial behavior of processes with chemical reactions.

1. INTRODUCTION

Traditional modelling of the kinetics of physicochemical reactions is based on the kinetic mass action law (KMAL) and leads to systems of differential equations (rate equations). New questions have recently arisen in connection with simulations of chaotic behavior in physicochemical systems. How is one to explain the unpredictable, unsteady state solutions of differential equations obtained in the light of the deterministic conception of differential equations? Can one find a reasonable connection linking the set of difference equations ($X_{n+1} = F(X_n)$), which at present involve a not very understandable conception of "discrete" time and generate the same scenario of chaotic dynamic behavior as do differential equations (which of course employ a conception of continuous time and space) – a connection that would have physicochemical meaning? Is it possible to use difference equations in the same way as we use differential equations to simulate dynamics, etc.?

This report is an attempt to formulate possible answers to these questions and to show that there exists an alternative mathematical language based on some physical conception for describing chemical reaction dynamics without using KMAL. Simulation of physicochemical processes in plasma and in chemical reactions with chaotic behavior will be attempted employing this new theory .

2. BACKGROUND

A new extreme principle was formulated [1]: reactions in multicomponent physicochemical systems proceed in such a way that at each instant of time t_q ($q = 1,2 ...$) the functional:

$$F = \sum_{i=1}^{n} X_i (\ln X_i - f_i - 1) \qquad (1)$$

reaches its minimum in the concentration space X_i subject to the law of conservation:

$$\sum_{i=1}^{n} a_{ij} X_i = b_j; \; j = 1,2,...,M; \; i = 1,2,...,N. \qquad (2)$$

Here $f_i = 0$, $i=1,2,...,M$; $f_i = \ln\pi_l$, $i = M + 1, M + 2,...,N$; π_l are the functions dependent on the time t_q, the temperature T_q, the pressure P, the concentrations of the reacting substances at previous instants of time $X_i (t_{q-s}, S = 1, 2 ...)$ and other variables affecting the process, but independent of the initial concentrations of the components; a_{ij} is a matrix defining the quantity of j-components in i-constituents.

In the case of simulation of the spatial distribution of the concentration π_l have a dependence on space coordinates [2].

Finding the minimum (1) subject to constraint (2) is equivalent to solving the system of algebraic equations.

$$\prod_{l=1}^{N} x_i^{\nu_{li}} = \pi_l, \; l = 1, 2,...,(N-M), \; L=N-M, \qquad (3)$$

with the same constraint (2), where $\| \nu_{li} \|$ is a stoichiometric matrix for the system of L reactions.

In the present paper we are treating the elements of the molecular and stoichiometric matrixes not as integers but as real numbers, resulting in a new and interesting variety of oscillations.

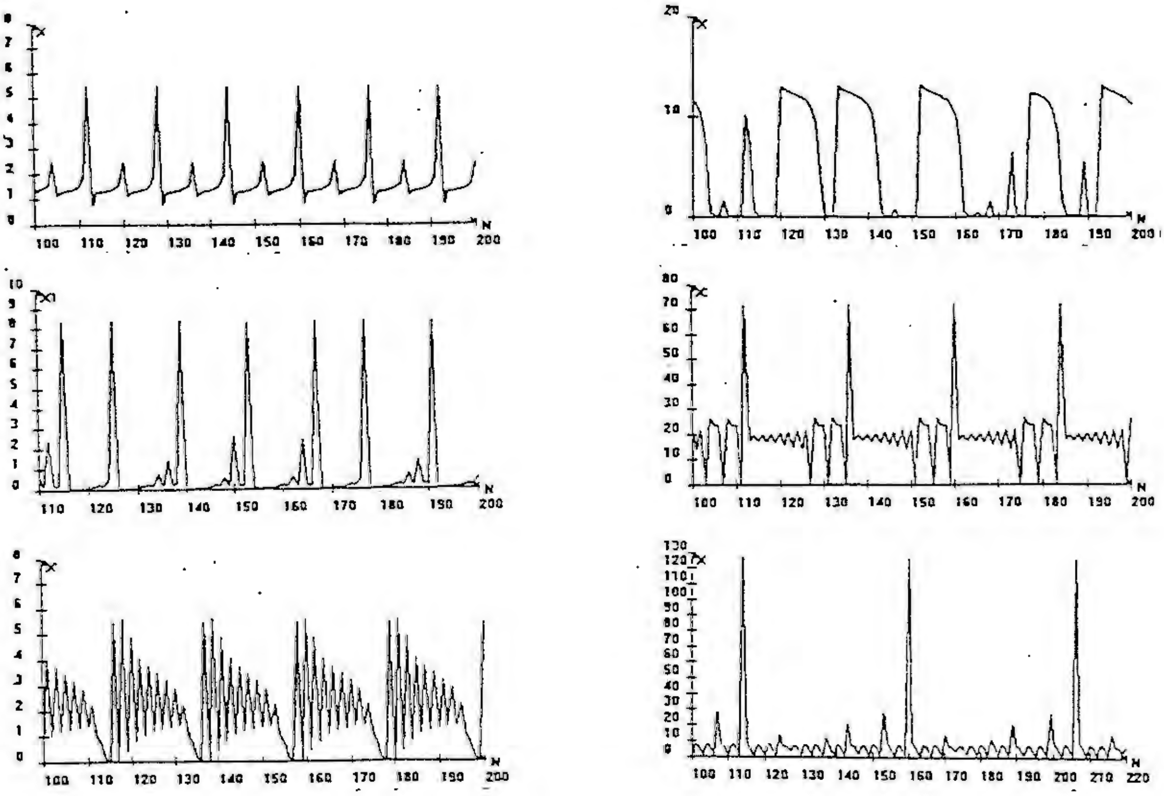
3. RESULTS

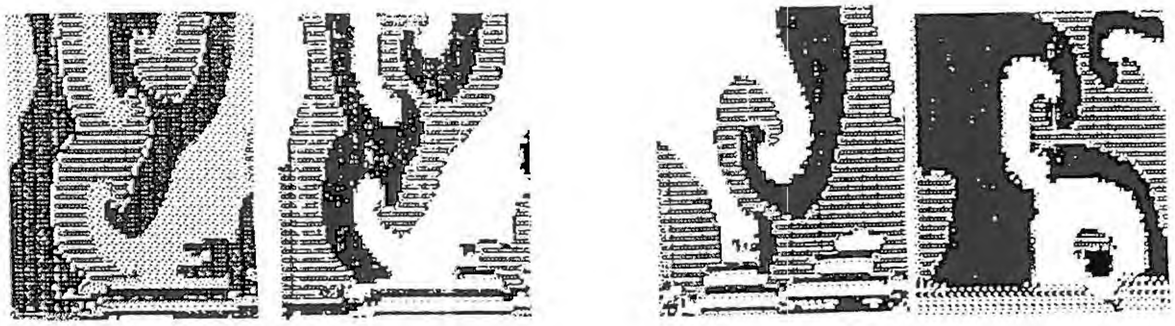
This approach make it possible to reduce the number of equations (3) needed for the simulation complex oscilations and presented in the figure below.

4. CONCLUSIONS

The equations we are suggesting offer extensive opportunities for describing spatially-inhomogeneous "self-organizing" physicochemical systems. Having formulated a new extreme principle as initial postulate for calculating the dynamics of physicochemical conversions in time and space, we suggested and tested a mathematical apparatus for describing the dynamics of all known processes that can be represented by the relation: "reaction system" with the matrix $\| \nu_{li} \|$. Further confirmation may be derived from the analogy between the π theorem of dimensional analysis and the stoichiometry of chemical reactions.

However, a number of fundamental questions should be addressed. First of all, the discrete one-way time of the proposed equations is crucial and requires thorough consideration. In our case time coincides with the traditional





conception of astronomical time only for "simple" closed systems. For "complex" open systems, such as catalytic systems, which "remember" intermediate states, the conception of time in its usual sense disappears. It breaks down into a set of "internal times" dependent on the properties of the system under investigation. The system itself is "the clock", and to adjust the oscillations that arise in such a system to the usual time is merely to pay tribute to the tradition.

In the framework of the approach under development, there is no concept of instantaneous chemical reaction rate, that is, no concept of a limit $\Delta x/\Delta t$ at $\Delta t \to 0$. The lack of derivatives and the discrete time do not require continuous space and time.

These properties of the new method for obtaining the X_i (t) trajectory are fundamental to the mathematical modelling and study of the emergence of unstable and self-organizing systems, where "small" perturbations involve reconstruction of the whole system. Perhaps the "algebraic" method of description of oscillations and waves will usher in the next era in the interpretation and modelling of complex systems. We hope that the "calculus of iterations" represents a new mathematical tool for simulating different types of complex dynamics—including chaotic ones—and will play the same role as the differential equations.

5. REFERENCES

[1] Gontar, V. New principle and mathematical model for formal chemical kinetics. Russian J. Phys. Chem 55, (1981), 2301-2305,

[2] Gontar, V. A new theoretical approach to the description of physicochemical reaction dynamics with chaotic behavior. In: "Chaos in Chemistry and Biochemistry" World Scientific, London, 1993, pp. 225-246.

Methods of System Modelling for Catalytic Reactors and Chemical Engineering Processes

A.G. Abilov

Institute of Petrochemical Processes The Azerbaijan Akad. of Sci. Telnov st. 30
Baku - 370025, Azerbaijan

M. Alpbaz
Chemical Engineering Department,
University of Ankara, Tandoğan
Ankara - 06100, Turkey

Y. Cabbar
Ministry of Environment
İstanbul Cd. 98, İskitler
Ankara - 06060, Turkey

Abstract The paper deals with the methods of system modelling for catalytic reactors and chemical engineering processes on the basis of the system analysis and the computer use. System modelling for chemical engineering processes were done by taking the chemical and physical properties of all the equipments into consideration.

1. Introduction

Improving the existing active technology of chemical processes through advanced analysis systems depends on investigating the theoric feasibility of processes, determining the factors that carry out the processes, selecting the reaction mechanism and forming the kinetic model and researching the consistency of the events at an industrial establishment with physical and chemical laws by means of laboratory and pilot scale experiments.

It is necessary to develope new computer aided analyses models in order to solve the above mentioned problems. These analyses models provide the information required to establish systematic models of reactors. The information provided by the analyses make it possible to set up huge scale reactors, to determine optimal parameters for industrial processes and to desing the control systems for these processes.

2. Fundementals of System Modelling For Chemical Engineering Processes

In this part, matters that are necessary and significant in determining the mathematical modelling and systematic investigation of catalytic processes were given. These are as follows.

— Controlling the experiments by computers; examining the rate of the reaction by analysing the results of the related experiments.

— Determining the reaction mechanism for the process and developing the kinetic model and calculating the parameters for the model by computer.

— Selecting the type of the reactor and determining the profile of the optimum temperature in the reaction medium theoretically. This is done with the help of maximum principle method on the basis of kinetic model.

— Making the mathematical models for physical and chemical processes by taking into consideration the equations of heat, mass and hydrodynamic for the type of the reactor chosen on kinetic model.

— Comparing the results of temperature obtained from the chemical and physical models with the temperature profile at the pilot scale experiment and thus detailing the model.

— Determining the static characteristics of processes from the model and calculating the temperature and conconstration profiles.

— Establishing the dynamic model after working on the physical and chemical models of the reactor. Testing the endurance of the reactor at steady - state and drawing the process curve in dynamic state.

— Developing the mathematical model that include all the equipments such as condencer, reactor, mixer, and seperator which are present in the plan of the technological process.

3. Computer Aided Mathematical Modelling For Catalytic Processes

In order to achieve the methodology mentioned in the second part of this paper algorithums were drawn and some calculation modules were developed. These algorithms and calculation modules were generalized for kinetic and reactor models for catalytic processes. While developing the algorithm the process was regarded from a systematic point of view.

Catalytic process was divided into different parts and studied seperately and modeled and the general models for the process was found out. Then the working conditions of the reactor which were determined earlier were calculated through computer on the model developed. The model was set up regardless of the real sizes of the reactor. The sizes were taken into account at the boundry and initial conditions.

Kinetic models for catalytic processes were linearized. This can be represented as follows. '

$$\frac{dC_i}{d\tau} = f\,(C_1, \ldots.\ C_n, K_1, \ldots\ldots.\ K_s) \quad i = \imath, \ldots\ldots, n \tag{1}$$

Reaction constants (Ks) were calculated in two ways; Kalman Filter Algorithm and recursive approach. The latter provided better results regarding the constants. By adding the hydrodynamic equation and differential temperature and material equations which were derived from the supposition that the catalyst part of the reactor was homogenius, to the Equation - 1 which was developed for system modelling for catalytic processes the following equations were derived.

$$\frac{dC_1}{d\tau} = W_i\ (C_1,\ \ldots\ldots.,\ C_n, T) \tag{2}$$

$$\frac{dT}{d\tau} = \frac{1}{C_pT}\ \Sigma\Delta H_i\ W_i\ + \alpha\ (T - T^*) \tag{3}$$

$$\frac{dP}{d\tau} = \chi \tag{4}$$

Different algorithms were developed so as to find out the optimal values of the parameters of the model, to prove the validity of the hypothesis and to determine the static characteristics of the processes.

As is known catalytic processes are exothermic and heat is released in these reactions. As a result the temperature rises which increases the rate of the reaction. The process goes on in this way and a cycling connection occurs in the reactor. Therefore it is necessary to study the process curves of dynamic and steady states in the reactor.

In making up the dynamic models for catalytic processes, the working conditions of the processes were exemined at different states; steady and transient. states, before sturt up.

Catalytic dynamic models with heat exchanger that are characterized by distributed parameter nature are considered with heat, kinetic and mass balance and also hydrodynamic equations are shown below.

$$\frac{\partial C_i}{\partial t} + \frac{\partial C_i}{\partial \tau} = \sum_{j=1}^{m} K_j \ (C_1, ..., C_j \ T) \qquad (5)$$

$$A_e \frac{\partial T}{\partial t} + B_e \frac{\partial T}{\partial \tau} = \sum_{j=1}^{m} \Delta H_j \ K_j \ (C_1, ..., C_j \ T) \qquad (6)$$

Initial and boundry condtions;

$$C_i (0) = C_{i', in} \qquad T (0) = T_{in} \qquad (7)$$
$$C_{i'} (t, \tau) = C_i (t_b, \tau_{b'}) \quad T (t, \tau) = T_b$$

The dynamic equations system established for tubular countercurrent heat exchanger are given below.

$$\frac{\partial Tg}{\partial t} + V_g \frac{\partial Tg}{\partial x} = r_1 \ (T_{CT} - T_g) \qquad (8)$$

$$\frac{\partial T_{CT}}{\partial t} = r_2 \ (T_g - T_{CT}) + r_3 \ (T_{XB} - T_{CT}) \qquad (9)$$

$$\frac{\partial T_{XB}}{\partial t} - V_{XB} \frac{\partial T_{XB}}{\partial x} = r_4 \ (T_{CT} - T_{XB}) \qquad (10)$$

The equations were solved through the method of finite difference for dynamic and steady states conditions. Figure - 1, shows the solution algorithm, Figure - 2 shows the changes of temperature and concentration of the compenents in the reactor.

4. Results

Computer methods of catalytic reactions experimental study providing wide collection program, analysis and interpretation results of kinetic experiment for establishing the probable mechanizm and composing kinetic models of complex reactions have been developed. The selection of accurate, single - valued and complete kinetic models allows to find theoretically optimum temperature distribution for substantiation of reactor type.

Identification procedures of the mathematical models in staties and dynamics of catalytic reactors have been developed and the calculations of consentrated and temperature fields were made on these bases. Stability conditions of stationary states and transient processes have been determined.

The packages of programmed modules permitting the realization of general database of algorithm automatizing (computerizing) research and modelling of catalytic processes.

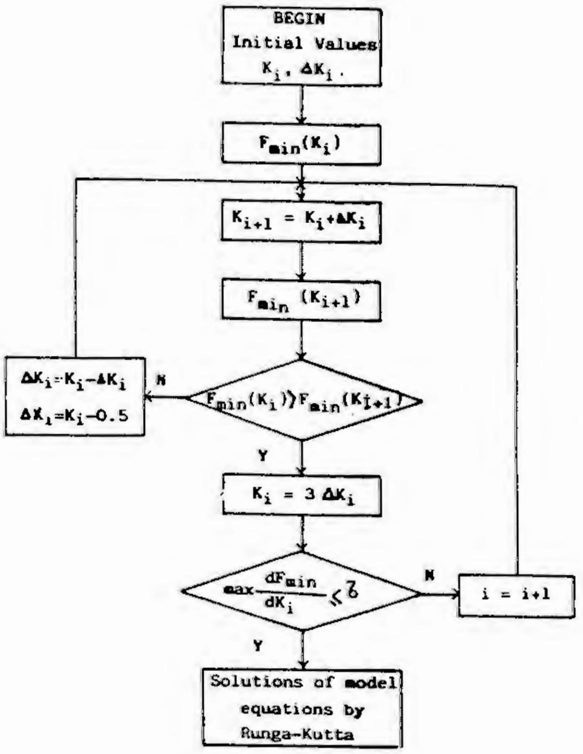

BEGIN
Initial Values
K_i, ΔK_i.
$F_{min}(K_i)$
$K_{i+1} = K_i + \Delta K_i$
$F_{min}(K_{i+1})$
$F_{min}(K_i) > F_{min}(K_{i+1})$
N
$\Delta K_i = K_i - \Delta K_i$
$\Delta K_i = K_i - 0.5$
Y
$K_i = 3\,\Delta K_i$
$\max \frac{dF_{min}}{dK_i} \leq \xi$
N
$i = i+1$
Y
Solutions of model equations by Runga-Kutta

Figure 1- Diagram of the solution algorithm.

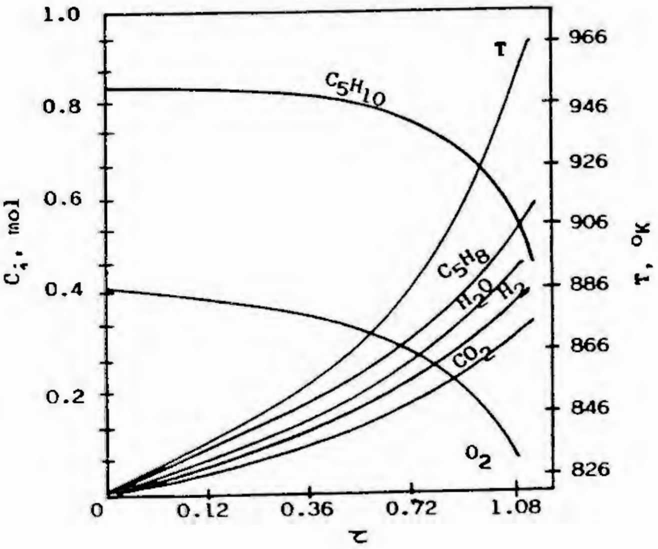


Figure 2- Changes of temperature and concentration in the reactor.

Greek letters and mathematical symbols:

τ	: Time constant
α	: thermal diffusivity
χ	: coefficient that depends on the temperature and the rate of the reaction
C_i	: Concentration of the reacting components
C_p	: Heat capasity of the component
ΔH_i	: Reaction heat
K_s	: Reaction constants
r_1, r_2, r_3, r_4	: constants.
T	: Temperature of the mixture
T_{CT}	: Temperature of cooler
Tg	: Temperature of gas mixture
T_{XB}	: Temperature of heater
Vg	: Lineer velocity of cold stream
V_{XB}	: Lineer velocity of warm stream
W_i	: Reaction rate of the compenent
X_i	: Vertical coordinate of the reactor.

Referances

Abilov, A.G., The Principles of Construction and Architecture of and Automatic Experiment Systemfor Investigation and Development of Chemical Engineering Process., Int. J. of computers and Chemical Engineering, (1980), 1-6.

Effect of Coupling on the Oscillations of a Biochemical Pathway

S. Doubabi[†], J.-P. Morillon[‡], and R. Costalat[‡]

[†]Département de Physique, Faculté des Sciences, Guéliz, B.P. 618, Marrakech Maroc

[‡]Université d'Angers, Institut de Biologie Théorique, 10, rue André Boquel, 49100 Angers France

Abstract. An analytic method has been applied to a biochemical pathway with coupling. The harmonic balancing technique can be used to determine the effects of coupling on the Goodwin metabolic pathway. It is shown that coupling modifies the linear filter such that the amplitude of the oscillations decreases. The results obtained here are compared with previous stability analysis of associated pathways.

1 Introduction and model

In our first paper [2], we showed that the parallel association of two Goodwin-Yates-Pardee metabolic pathways can increase the stability domain of the unique steady state. We here investigate the oscillations of the Goodwin metabolic pathway [1] by means of the harmonic balancing technique [3], with the addition of some passive diffusion coupling (see Fig. 1). More precisely, we assume that (*i*) a cell or an organelle *u* contains a Goodwin metabolic pathway [1]; (*ii*) the metabolite pool P_i, $i = 1, \ldots, n$, can exchange matter with the outside pool P_i^* via passive diffusion, with a non-negative coefficient β_i.

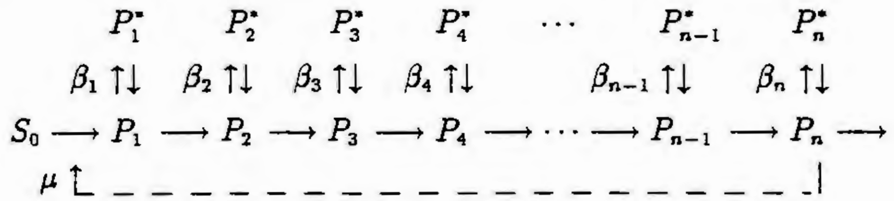


Figure 1. Goodwin biochemical pathway with couplings.

Implementing the transformation of Walter [3], the system can be modelled by the following set of differential equations that describe the time evolution of metabolic concentrations:

$$\begin{aligned}
\frac{dP_1}{dt} &= f(P_n) - b_1 P_1 - \beta_1(P_1 - P_1^*), \\
\frac{dP_k}{dt} &= P_{k-1} - b_k P_k - \beta_k(P_k - P_k^*), \qquad k = 2, \ldots, n, \\
\frac{dP_k^*}{dt} &= \beta_k(P_k - P_k^*), \qquad k = 1, \ldots, n,
\end{aligned}$$

where

$$f(P_n) = \frac{1}{1 + P_n^\mu}, \qquad \mu \in \{2, 3, 4\}.$$

The b_i are positive kinetic constants, the β_i are the non-negative diffusion coefficients, and the function f corresponds to the allosteric feedback inhibition of the first reaction by the last product.

2 Analysis of oscillations

For the analysis of the non-associated system, i.e. when all $\beta_k = 0$, we introduce the linear filter G:

$$G(p) = \left\{ \prod_{k=1}^{n} (p + b_k) \right\}^{-1}$$

where p denotes the differentiation operator. After one or several couplings, the harmonic balance method leads us to consider the following filter:

$$G^*(p) = \left[\prod_{k=1}^{n} \frac{(p+b_k)(p+\beta_k)}{(p+b_k)(p+\beta_k)+p\beta_k} \right] G(p).$$

We can first prove by successive iterations that $G^*(i\omega) = G(i\omega)$ if and only if $\omega = 0$ or $\beta_k = 0$ for all $k = 1, \ldots, n$, and the argument of $G^*(i\omega)$ verifies the following formulae:

$$\text{Arg } G^*(i\omega) = \text{Arg } G(i\omega) - d(\omega), \qquad d(\omega) = \sum_{k=1}^{n} \left[\text{Arctan } \frac{(b_k + 2\beta_k)\omega}{b_k\beta_k - \omega^2} - \text{Arctan } \frac{(b_k + \beta_k)\omega}{b_k\beta_k - \omega^2} \right].$$

The above results state that $d(\omega) < 0$ and Arg $G^*(i\omega) >$ Arg $G(i\omega)$ if ω is large enough. Since $|G^*(i\omega)| \leq |G(i\omega)|$, the $G^*(i\omega)$-curve is nearer the origin in the complex plane than the $G(i\omega)$-curve. The balance method gives us the following first harmonic equation: $G^*(i\omega) = 1/F(x)$ where F is a real function of x, the amplitude of the oscillation, and ω, the frequency. If the $G^*(i\omega)$ and $1/F(x)$–curves intersect in the complex plane, the amplitude x^* of a periodic solution is smaller when coupling exists. Numerical simulations confirm this fact (see Fig. 2).

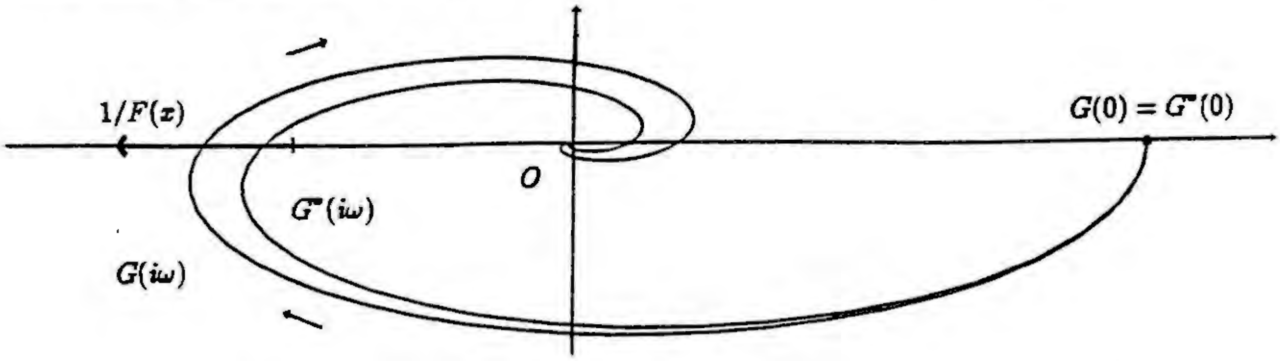


Figure 2. Linear filters in the complex plane. The G^*–curve is nearer the origin than the G–curve.

3 Conclusion

Our results show that coupling with exterior pools of metabolites modifies the behaviour of the oscillations of a Goodwin metabolic system in the following way: (*i*) periodic solutions of the system with coupling have lower amplitudes; (*ii*) coupling can give rise to a steady state, instead of a periodic solution. These observations are consistent with the increase of stability observed when two Goodwin-Yates-Pardee metabolic pathways are associated [2].Thus the stability of the Goodwin-Yates-Pardee metabolic pathway, which is the most commonly encountered form of regulated biochemical pathway, is enhanced when matter can be exchanged with the outside. It may be conjectured that this non-trivial property is one of the reasons why Goodwin metabolic pathways could compete successfully in the natural selection process.

Acknowledgements. This work was supported by the Conseil général du Maine-et-Loire and the Communauté des universités méditerranéennes.

References

[1] Goodwin, B.C., Analytical Physiology of Cells and Developing Organisms. Academic Press, London, 1976.

[2] Morillon, J.-P., Costalat, R., Burger, N., and Burger, J., Modelling Two Associated Biochemical Pathways. This volume.

[3] Rapp, P., Analysis of Biochemical Phase Shift Oscillators by a Harmonic Balancing Technique. J. Mathematical Biology, 3(1976), 203-224.

Computer Algebra Systems for Modeling Complex Processes in Polymer Extrusion

M. Jahnich, K. Panreck and F. Dörrscheidt

University of Paderborn, Department of Control Engineering
Pohlweg 47-49, D-33098 Paderborn, Germany

Abstract. The modeling of dynamic extrusion processes often requires the manipulation of large algebraic expressions, particularly if it is necessary to reduce the order of distributed parameter systems by means of spatial discretization. This paper deals with the application of the computer algebra system *Maple V* for a computer-aided generation of mathematical temperature models used in polymer extrusion. Moreover, it is shown how these models are integrated into an interconnection-oriented modeling concept.

1. Introduction

In order to develop new control strategies for the extrusion process, the modeling and simulation of instationary temperature behavior has become more and more important.

Since extrusion has to be regarded as a complex process, the first step of modeling consists of structuring the process into appropriate subsystems [2,3]. These are connected by means of coupling systems that do represent the physical connection of those subsystems. This procedure leads to an interconnection-oriented modeling concept [3].

The modeling of temperature subsystems often results in distributed parameter models which are usually inappropriate for a system analysis or system synthesis. Therefore, methods of spatial discretization have to be applied to those mathematical models.

Employing these methods, hand computations are quite error-prone and time-consuming, especially for higher dimensional problems. Computer algebra has been applied in order to generate efficient computer code for numerical finite element packages and to derive stiffness matrices symbolically [5]. This paper, however, deals with algorithms creating models of finite dimension, which can be used with the described modeling concept. These algorithms are based on the Galerkin's finite element approach.

2. Modeling strategies for extrusion processes

Single screw extruders are used in different fields of polymer processing. As illustrated in fig. 1, a screw rotates in a cylindrical barrel with a die at the output end. Solid polymer granules are added to the feed hopper and transported by the screw to the die. As a consequence of shearing in the screw channel and heat transfer through the barrel from the barrel heaters the polymer is melted. Afterwards the melt is extrudated through the die where the product is finally shaped.

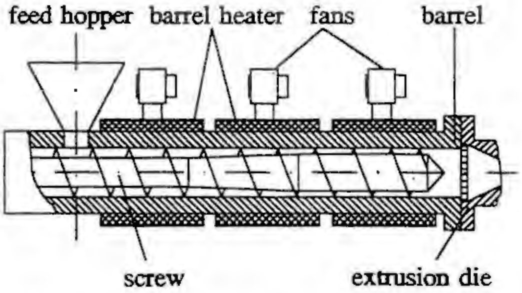


Figure 1. single screw extruder

Since there are different physical phenomena in the components of the extruder and in the die, the extrusion process has to be treated as a complex process. For this reason, an interconnection-oriented concept has been applied to modeling. Herein, the models are composed of subsystems and coupling systems which describe the dynamic behavior of single components and the coupling of the subsystems, respectively.

In order to formulate the different models, the first step of modeling is the structuring of the process. This can be done under phenomical as well as under geometrical aspects and depends basically on the objective of the model [3]. After that, the defined subsystems and coupling systems can be modeled separately. The description of the N subsystems generally leads to N nonlinear models

$$\dot{\mathbf{x}}_i = \mathbf{f}_i(\mathbf{x}_i, \mathbf{u}_{K,i}, \mathbf{u}) \ , \quad \mathbf{y}_{K,i} = \mathbf{c}_{K,i}(\mathbf{x}_i, \mathbf{u}_{K,i}, \mathbf{u}), \ i = 1, ..., N, \tag{1}$$

wherein $\mathbf{x}_i$ is the dynamic state, $\mathbf{u}_{k,i}$ and $\mathbf{y}_{k,i}$ are the inputs and outputs for interconnection and $\mathbf{u}$ is the global input. The coupling systems are described by M algebraic equations

$$\mathbf{0} = \mathbf{g}_{K,j}(\hat{\mathbf{y}}_{K,j}, \hat{\mathbf{u}}_{K,j}, \mathbf{u}) \ , \quad j = 1, ..., M \ , \tag{2}$$

wherein $\hat{\mathbf{u}}_{K,j}$ and $\hat{\mathbf{y}}_{K,j}$ are the input and output vectors of the coupling system, respectively. The subsystems and coupling systems are connected as follows :

$$\hat{\mathbf{y}}_{K,j} = \mathbf{C}_{K,j} \cdot (\mathbf{y}_{K,1}^T, ..., \mathbf{y}_{K,N}^T)^T \ , \quad \hat{\mathbf{u}}_{K,j} = \mathbf{B}_{K,j} \cdot (\mathbf{u}_{K,1}^T, ..., \mathbf{u}_{K,N}^T)^T \ . \tag{3}$$

These equations represent the topology of the model. Thus, the matrices $\mathbf{B}_{K,j}$ and $\mathbf{C}_{K,j}$ only consist of zeros and ones.

The model of the total process forms a differential algebraic equation (DAE). Depending on the nature of the physical connections the coupling equations can be of a "singular" type resulting in a DAE of higher index [2].

3. Computer algebra for modeling temperature processes

The fundamental equation describing the transport of energy in a homogeneous fluid or solid is the equation of energy [1]. For the extrusion process its instationary temperature profiles can often be described by

$$\frac{1}{a}\frac{\partial T}{\partial t} - \Delta T - \frac{1}{\lambda}\Phi_v = 0. \tag{4}$$

Herein, T is the temperature inside a subsystem, a is the thermal diffusivity, λ is the thermal conductivity and Φ_v is the viscous dissipation function. For pure viscous melts Φ_v is a nonlinear function of T and of the second invariant I_2 of the fluid's shear rate tensor $\underline{\dot{\gamma}}$. The convective energy transport is neglected here, this assumption is valid e.g. for melt flows in a long pipe with a constant cross-section (fig. 2).

The modeling of a subsystem using (4) comprises the following steps :

1) introduction of dimensionless variables,
2) spatial discretization (FEM),
3) elimination of expressions which belong to boundary temperatures,
4) introduction of states θ (related temperatures at inner nodes of the domain Ω) and of coupling variables θ_k (related temperatures at nodes on the boundary Γ),
5) computation of coupling equations,
6) computation of output equations.

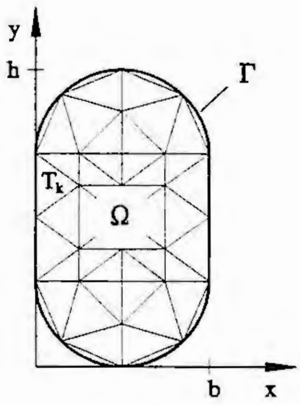


Figure 2. simple extrusion die geometry (cross-section)

Galerkin's finite element approach [4] has been implemented in *Maple V* in order to perform the second step of the modeling procedure. For this reason, (4) has to be rewritten in the weak formulation

$$\int_{\Omega}\left(\frac{1}{a}\,\frac{\partial T}{\partial t}\,\varphi_j+\nabla T\nabla\varphi_j-\frac{I}{\lambda}\,\Phi_v\varphi_j\right)dx\,dy-\oint_{\Gamma}\frac{\partial T}{\partial n}\,\varphi_j ds=0. \tag{5}$$

Employing steps 1 to 4 yields the state space equation

$$\mathbf{r}:=\mathbf{M}\dot{\boldsymbol{\theta}}+\mathbf{A}\boldsymbol{\theta}+\mathbf{B}_1\boldsymbol{\theta}_k+\mathbf{B}_2\boldsymbol{\Phi}_v^*=\mathbf{0}\ , \tag{6}$$

which is computed in residual form.

The boundary integral can be disregarded since it only contributes to expressions which have been elimininated from (6). It will be used for the computation of the coupling equations.

The integrand of (5) has been programmed in a maple procedure as follows:

```
Genergy :=
 proc(TEMP,DISS,FORMFCN,LAMBDA,A);
  diff(TEMP,t)*FORMFCN/A + dotprod(grad(TEMP,[x,y,z]),
  grad(FORMFCN,[x,y,z])) - DISS*FORMFCN/LAMBDA;
end;   #   of procedure Genergy
```

The parameters *TEMP* and *DISS* are local interpolation polynomials in x and y,

$$T(t,x,y)=\sum_{i=1}^{m} T_i(t)\,\hat{\varphi}_i(x,y)\ , \tag{7}$$

on the k-th element T_k (fig. 2); herein T_i and $\hat{\varphi}_i$ are indexed locally in T_k; m is the number of local basis functions. The parameter *FORMFCN* contains the j-th local basis function. Having called *Gener-*

gy, a polynomial in x and y is yielded. Now, dimensionless spatial variables as well as related temperatures θ und a related dissipation function Φ_v^* can be introduced.

After integrating this polynomial over T_k, and after doing this for all $\hat{\varphi}_j$, having a support in T_k, the results are added to the respective elements of the residual vector **r**. The vector **r** contains elements which belong to nodal points on Γ and which will be elimininated because T is supposed to be given here (step 3). The separation of the temperature variables at the inner and outer nodal points provides the state variables $\boldsymbol{\theta}$ and the coupling elements $\boldsymbol{\theta}_k$, respectively (step 4). All in all, the algorithm computes a model as given in (6).

In addition to the computation of a state space equation, the coupling equations have to be generated as well, in order to connect subsystems. These equations for temperature systems are based on continuity requirements for the temperature and for the heat flux at the boundary of the considered domain and arise from the discretization of those boundary conditions

$$\dot{\mathrm{q}} + \lambda \, \nabla \mathrm{T} = 0, \; \mathrm{T_b} - \mathrm{T} = 0 \quad \text{on } \Gamma. \tag{8}$$

The vector $\dot{\mathrm{q}}$ is the heat flux and T_b is the temperature on the boundary Γ. Using the weak formulation, the continuity equation for the heat flux becomes

$$\int_{\Gamma} (\dot{\mathrm{q}}_n + \lambda \, \nabla \mathrm{T} \cdot \vec{\mathrm{n}}) \, \varphi_j \mathrm{d}s = 0 \tag{9}$$

where φ_j varies freely on Γ. The scalar $\dot{q}_n$ is the heat flux normal to Γ. Now, the procedure corresponds to the generation of the state space equation. The functions $\dot{q}_n$ and T_b are interpolated by their values at the coupling nodes q_i and $T_{b,i}$, respectively (fig. 3). After integrating over Γ, the results are of the form

$$\mathbf{C}_{k,1}\mathbf{q} + \mathbf{C}_{k,2}\boldsymbol{\theta}_k = \mathbf{0} \; ; \quad \mathbf{C}_{k,3}\mathbf{T}_b + \mathbf{C}_{k,4}\boldsymbol{\theta}_k = \mathbf{0}. \tag{10}$$

The vectors **q** and $\mathbf{T}_b$ are the ouputs of the subsystem of the interconnection-oriented modeling concept. It is important to notice that the coupling elements do have corresponding dimensions, i.e. the introduction of dimensionless variables is valid only for the considered subsystem; the modeling of the connections refers to its physical values. The interconnection-oriented model (6) and (10) can easily be transformed into a model as given in (1).

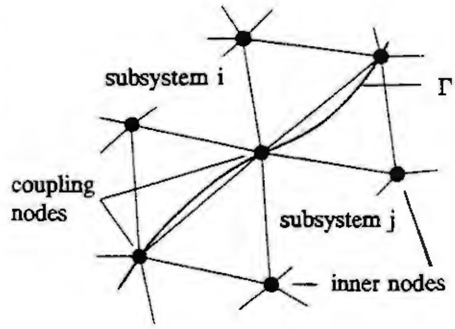


Figure 3. coupling of subsystems

Further aspects of computer algebra in modeling and simulation of extrusion processes can be found in the

- generation of output equations, e.g for the computation of average temperatures or of the average dissipation in a subsystem.
- elimination of coupling variables, which are only involved in linear coupling equations. This reduces the dimension of the process model used in simulation.
- symbolic computation of the jacobian and automated generation of computer code.

4. Conclusions

The modeling of the extrusion process often implies the manipulation of large algebraic expressions, especially if the order reduction of distributed parameter models is involved. Thus, the application of computer algebra is quite useful for the automatic generation of interconnected models of subsystems used in polymer extrusion.

5. Acknowledgements

The authors would like to thank the DFG for supporting the work of their group in this field of research.

6. References

[1] BIRD, R. B.; STEWART, W. E.; LIGHTFOOT, E. N.: *Transport Phenomena*. New York: Wiley & Sons, 1960.

[2] PANRECK, K.; JAHNICH, M.; DÖRRSCHEIDT, F.: Using Differential-algebraic Equations for Interconnection-oriented Modeling of Complex Processes. To be published in: *Automatisierungstechnik - at.*

[3] MARQUARDT, W.; ZEITZ, M.: Rechnergestützte Modellbildung in der Verfahrenstechnik. In: *VDI-Berichte 925: Modellbildung für Regelung und Simulation*. Düsseldorf: VDI, 1992, pp. 307-341.

[4] SCHWARZ, H.R.: *Methode der Finiten Elemente*. 3rd Edition. Stuttgart: Teubner, 1991.

[5] SILVESTER, P. P.: Symbolic Computation as a Basis for Numerical Methods. *IEE Conference Publication* n 350, London, UK, 1991, pp. 1-5.

OPTIMAL YIELD IN CERTAIN CONTINUOUS CHEMICAL REACTIONS

Günther KARIGL

Abteilung für Mathematik in den Naturwissenschaften, Technische Universität Wien
A-1040 Wien, Wiedner Hauptstraße 8/118

Abstract. In this paper a consecutive chemical reaction of type A → B → C is considered where source A is turned to a product B in a catalytic reaction, and B is decomposed to C at the same time. Suppose that A can be supplied in a constant source concentration and B and C can be removed continuously. This continous extraction process is modelled by means of partial differential equations and optimal yield of B is compared for different modelling assumptions. It is shown that the efficiency of the reaction theoretically can be brought up arbitrary close to 1.

1. INTRODUCTION

Consecutive reactions are very common in reaction kinetics. We focus our interest on reactions of type

$$A \rightarrow B \rightarrow C\,, \tag{1}$$

where product B results from source A in a catalytic reaction and is decomposed to product C at the same time. It is our aim to maximize the yield of product B. A typical example for this situation is given by the acid hydrolysis of cellulose to gain glucose and alcohol (cf. Concin [1], Grethlein [2], Saeman [4] among many others).

In a simple time-dependent model the reaction can be described by a system of ordinary differential equations. Since Rakowsky [3] it is well known that the concentration of B runs through a maximum such that optimal yield can be achieved only at a certain time point depending on the reaction parameters. The situation is different, however, if the reaction occurs in a solvent where product A is insoluble and components B and C are soluble. In this case B and C can be removed continuoulsy and the chemical reaction B → C can be stopped. Moreover, the source A can be transported continuously through the solvent such that a constant initial source concentration will be maintained.

The aim of this paper is to analyze the improvement of yield B if all components A, B and C are continuously supplied and removed through the solvent. We consider different possibilities of transportation: a simple time-dependent model for reference and three spatial models. For each model the optimization of yield B is considered using the efficiency w which is defined as

$$w = \text{output of product B / input of source A}\,.$$

Thus the efficiency indicates the fraction of source A being converted into the reaction product B.

2. REFERENCE MODEL

First of all let us consider a simple time-dependent model for the reaction (1). Both subprocesses of (1) are assumed to be first order reactions taking place at the same time. Let $c_A(t)$, $c_B(t)$ and $c_C(t)$ be the concentrations of products A, B and C, respectively, and let k_1 and k_2 be reaction constants. Then the process can be described by the following model equations:

(2) $$\begin{cases} \dot{c}_A = -k_1 c_A \\ \dot{c}_B = k_1 c_A - k_2 c_B \\ \dot{c}_C = k_2 c_B \end{cases}$$

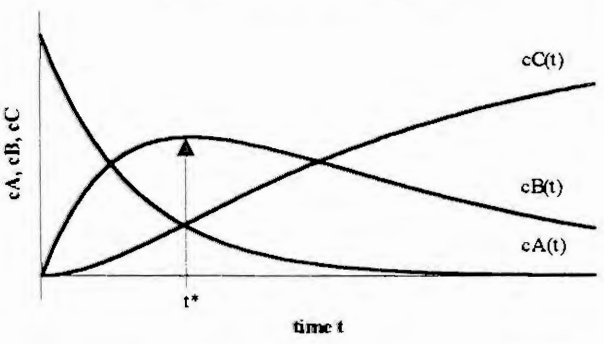
The solution of system (2) is given by

$$c_A(t) = c_{A0}\, e^{-k_1 t}$$

$$c_B(t) = \begin{cases} c_{A0} \dfrac{K}{K-1}\left[\left(1 + \dfrac{c_{B0}}{c_{A0}}\dfrac{K-1}{K}\right) e^{-k_2 t} - e^{-k_1 t}\right] & \text{for } K \neq 1 \\ c_{A0}\left(k_1 t + \dfrac{c_{B0}}{c_{A0}}\right) e^{-k_1 t} & \text{for } K = 1 \end{cases}$$

where $K = k_1/k_2$. (The concentration of C is determined by $c_A(t) + c_B(t) + c_C(t) = c_{A0} + c_{B0} + c_{C0}$.) The qualitative behaviour of these solutions with initial conditions $c_{A0} > 0,\ c_{B0} = c_{C0} = 0$ is illustrated in Fig. 1.

Fig. 1 Concentrations of A, B and C



Next let us consider the function $w(t) = c_B(t)/c_{A0}$, i.e. the relative concentration of B. This function has a unique maximum at

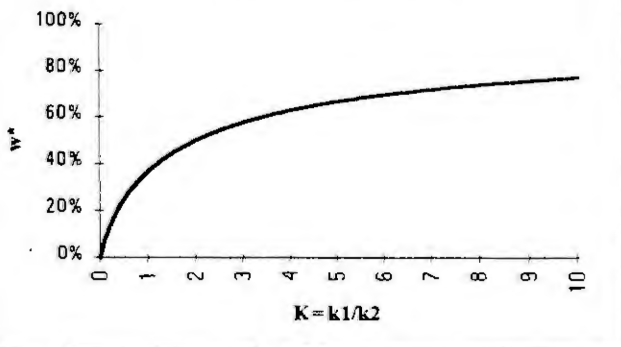
(3) $$t^* = \begin{cases} \dfrac{1}{k_1 - k_2} \ln K \Big/ \left(1 + \dfrac{c_{B0}}{c_{A0}}\dfrac{K-1}{K}\right) & \text{for } K \neq 1 \\ \dfrac{1}{k_1}\left(1 - \dfrac{c_{B0}}{c_{A0}}\right) & \text{for } K = 1 \end{cases}$$

with value

(4) $$w^* = \begin{cases} \left[\left(1 + \dfrac{c_{B0}}{c_{A0}}\dfrac{K-1}{K}\right)^K \Big/ K\right]^{\frac{1}{K-1}} & \text{for } K \neq 1 \\ e^{-1+c_{B0}/c_{A0}} & \text{for } K = 1 \end{cases}$$

Hence in the important case $c_{B0} = 0,\ K \neq 1$ optimal reaction efficiency yields $w^* = K^{-1/(K-1)}$ (cf. Fig. 2 and Tab. 1).

Fig. 2 Optimal efficiency



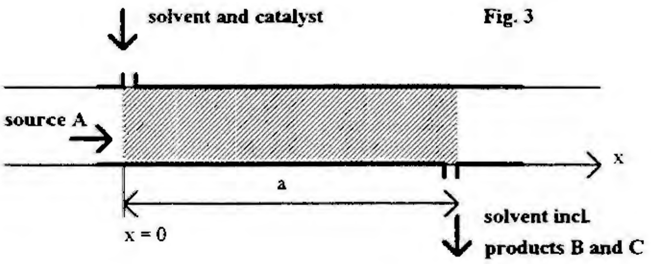
Tab. 1

K	w* (in %)
0,1	7,7
0,5	25,0
1	36,8
2	50,0
5	66,9
10	77,4
100	95,5

3. SIMPLE FLOW MODEL

As a first approach in modelling the consecutive reaction (1) in time and space we consider a situation where all three components underly a continuous movement (say in x-direction) with constant velocity v_x. As before both subprocesses of (1) are inseparable and occur at the same time, because no relative movement between the reaction components takes place. On the other hand, however, now continous processing becomes possible (see Fig. 3). The concentrations c_A, c_B and c_C are considered as functions in x and t. From the continuity equation the following model equations derive:

Fig. 3



(5)
$$\begin{cases} \dfrac{\partial c_A}{\partial t} = -v_x \dfrac{\partial c_A}{\partial x} - k_1 c_A \\ \dfrac{\partial c_B}{\partial t} = -v_x \dfrac{\partial c_B}{\partial x} + k_1 c_A - k_2 c_B \\ \dfrac{\partial c_C}{\partial t} = -v_x \dfrac{\partial c_C}{\partial x} + k_2 c_B \end{cases}$$

At $x = 0$ source A and the solvent are supplied continuously. This idea corresponds to the boundery conditions $c_A(x=0, t) = c_{A0}$ and $c_B(x=0, t) = c_{B0}$ leading to the following time-independent solutions of (5):

$$c_A(x) = c_{A0}\, e^{-k_1 \frac{x}{v_x}}$$

$$c_B(x) = c_{A0} \frac{K}{K-1}\left[\left(1 + \frac{c_{B0}}{c_{A0}} \frac{K-1}{K}\right) e^{-k_2 \frac{x}{v_x}} - e^{-k_1 \frac{x}{v_x}}\right] \quad (K \neq 1)$$

The behaviour of model (5) follows directly from the reference model by substituting t by x/v_x. (This is also true in the special case $K = 1$.)

Another approach to the above solutions is achieved by considering the initial conditions $c_A(x, t = 0) = c_A{}^0(t = x/v_x)$ and $c_B(x, t = 0) = c_B{}^0(t = x/v_x)$, where $c_A{}^0$ and $c_B{}^0$ are the corresponding solutions of the reference model. This means that the initial concentrations at distance x correspond to the concentrations in the reference model which result for time $t = x/v_x$.

As an objective function we choose the efficiency of the reaction $w = c_B(x = a)/c_{A0}$, i.e. the quotient of output of product B and input of product A at the end of the reaction area (at distance $x = a$). After transformation to the dimensionless variable $\xi = k_1 a/v_x$ the optimal solution of the problem $w(\xi) = \max$ is derived from (3) to be

$$\xi^* = \begin{cases} \dfrac{K}{K-1} \ln K \Big/ \left(1 + \dfrac{c_{B0}}{c_{A0}} \dfrac{K-1}{K}\right) & \text{for } K \neq 1 \\ 1 - \dfrac{c_{B0}}{c_{A0}} \quad \text{for } K = 1 \end{cases}$$

This also determines the optimal length of the reaction area $a^* = \xi^* v_x/k_1$ (for a given velocity v_x). The optimal efficiency again is given by (4).

4. PERCOLATION MODEL

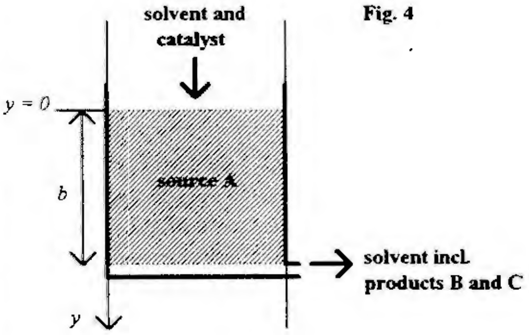


Fig. 4

In this section we consider a variant of process (1) allowing for relative movement between the source product A and the solvent including products B and C (as well as the catalyst). In this case it is possible to separate the two subprocesses of (1) and to improve the reaction's efficiency.

In the sequel we assume that product A stays at rest while the solvent (together with components B and C) undergoes a constant movement with sinking velocity v_y (cf. Fig. 4). The concentrations c_A, c_B and c_C are assumed to depent on height y and time t. These assumptions lead to the system of partial differential equations

$$(6) \qquad \begin{cases} \dfrac{\partial c_A}{\partial t} = -k_1 c_A \\ \dfrac{\partial c_B}{\partial t} = -v_y \dfrac{\partial c_B}{\partial y} + k_1 c_A - k_2 c_B \\ \dfrac{\partial c_C}{\partial t} = -v_y \dfrac{\partial c_C}{\partial y} + k_2 c_B \end{cases}$$

which has the solution

$$c_A(y,t) = c_{A0}\, e^{-k_1 (t - \frac{y}{v_y})}$$

$$c_B(y,t) = c_{B0}\, e^{-k_2 \frac{y}{v_y}} + c_{A0} K \left(1 - e^{-k_2 \frac{y}{v_y}}\right) e^{-k_1 (t - \frac{y}{v_y})} \qquad (t \geq \frac{y}{v_y})$$

where the conditions $c_A(y, t = y/v_y) = c_{A0}$ and $c_B(y = 0, t) = c_{B0}$ have been used.

In this model reaction efficiency, i.e. the ratio of B to A at time t, is given by

$$w = \frac{1}{c_{A0} b} \int_{\frac{b}{v_y}}^{t + \frac{b}{v_y}} c_B(y = b, \bar{t})\, d\bar{t}$$

$$= \frac{1}{\eta} \left[\left(1 - e^{-\eta}\right)\left(1 - e^{-\tau}\right) + \frac{c_{B0}}{c_{A0}} \frac{1}{K} e^{-\eta} \tau \right]$$

where the dimensionless variables $\eta = k_2 b/v_y$ and $\tau = k_1 t$ have been introduced. Moreover, we describe the relative consumption of solvent with respect to source product A by the catalyst ratio v, which is – in contrast to the other models discussed so far – not constant any more but yields

$$v = \frac{v_y\, t}{b} = \frac{1}{K}\frac{\tau}{\eta}\,.$$

The demand for high efficiency leads to the problem w = max which will be discussed in the next section. Observe that τ describes the first and η the second process of reaction (1), such that a certain separation of both processes is possible.

5. TRANSVERSAL FLOW MODEL

A more general situation is described by a model with horizontal source transport and vertical fluid transport as shown in Fig. 5. Let us assume constant transport velocities v_x and v_y in horizontal and vertical directions, respectively. The geometry of the reaction area is determined by the parameters a and b (see Fig. 5). Again a relative movement between the different reacting components takes place, such that a separation of the two subprocesses is possible. In the subsequent analysis all concentrations are assumed to depend on the space variables x, y and on time t.

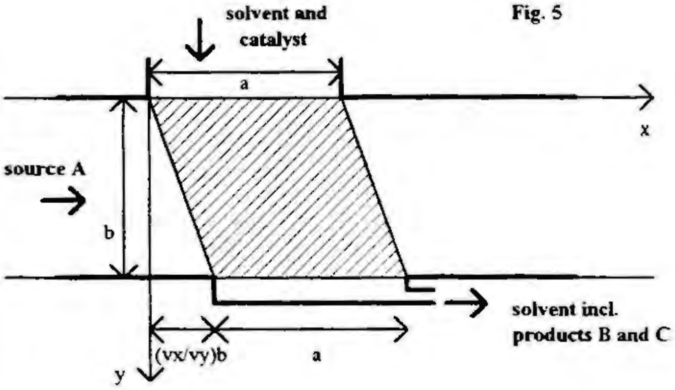


Fig. 5

Analogous to (5) and (6) now we have the system equations

$$(7)\qquad \begin{cases} \dfrac{\partial c_A}{\partial t} = -v_x \dfrac{\partial c_A}{\partial x} - k_1 c_A \\ \dfrac{\partial c_B}{\partial t} = -v_x \dfrac{\partial c_B}{\partial x} - v_y \dfrac{\partial c_B}{\partial y} + k_1 c_A - k_2 c_B \\ \dfrac{\partial c_C}{\partial t} = -v_x \dfrac{\partial c_C}{\partial x} - v_y \dfrac{\partial c_C}{\partial y} + k_2 c_B \end{cases}$$

According to our analysis in the last sections we are interested in solutions with boundery conditions $c_A(x, y = xv_y/v_x, t) = c_{A0}$ and $c_B(x, y = 0, t) = c_{B0}$ which can be shown to be

$$c_A(x,y) = c_{A0}\, e^{-k_1(\frac{x}{v_x} - \frac{y}{v_y})}$$

$$c_B(x,y) = c_{B0}\, e^{-k_2\frac{y}{v_y}} + c_{A0} K \left(1 - e^{-k_2\frac{y}{v_y}}\right) e^{-k_1(\frac{x}{v_x} - \frac{y}{v_y})}$$

(Observe that these solutions again are independent of time.)

In order to determine optimal yield B in this model we assume $c_{B0} = 0$. First we calculate the output of product B

$$\overline{w} = \int_{\frac{v_x}{v_y}b}^{a+\frac{v_x}{v_y}b} c_B(x, y = b) v_y \, dx$$

$$= \frac{c_{A0}}{k_2} v_x v_y \left(1 - e^{-k_1\frac{a}{v_x}}\right)\left(1 - e^{-k_2\frac{b}{v_y}}\right)$$

Next we derive the efficiency w – substituting for $\xi = k_1 a/v_x$ and $\eta = k_2 b/v_y$ – according to

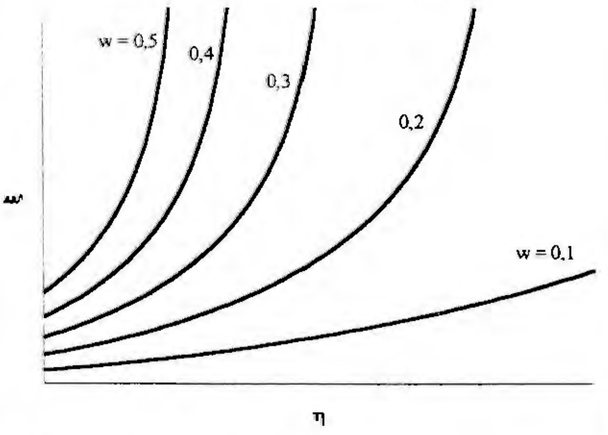
$$w = \frac{\overline{w}}{c_{A0} b v_x} = \frac{1}{\eta}\left(1 - e^{-\xi}\right)\left(1 - e^{-\eta}\right).$$

Moreover, the catalyst ratio is given by

$$v = \frac{a v_x}{b v_y} = \frac{1}{K}\frac{\xi}{\eta}.$$

Obviously the function $w = w(\xi, \eta)$ is increasing in ξ and decreasing in η as visualized in Fig. 6. From this it is clear that efficiency can be brought up arbitrary close to 1 (which involves, however, a dramatically increasing catalyst ratio).

Fig. 6 Reaction efficiency w

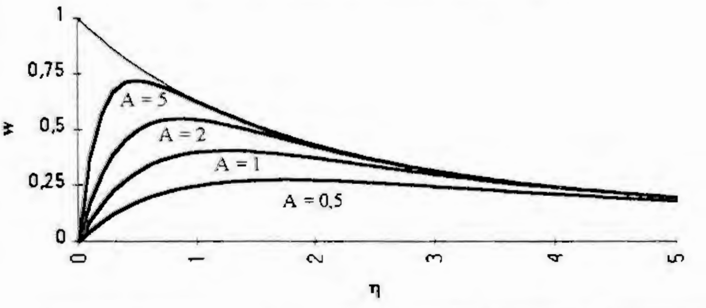


Finally we consider the problem w = max, v = const which leads to the objective function

$$w(\eta) = \frac{1}{\eta}\ 1 - e^{-\eta}\big)\ 1 - e^{-A\eta}\big)$$

with $A = Kv$. The problem $w(\eta)$ = max can be solved by numerical methods only. The qualitative behaviour of $w(\eta)$ for different values of A is indicated in Fig. 7 and shows a concave function with a unique maximum. The value of this maximum is given in Tab. 2. Comparing Tab. 1 and Tab. 2 it becomes evident that already for a catalyst ration of $v = 1$ (which implies $A = K$) the maximal efficiency is higher than the corresponding values for the models of section 2 and 3.

Fig. 7 Efficiency for constant catalyst ratio



Tab. 2

A	w* (in %)
0,1	8,2
0,5	27,5
1	40,7
2	55,1
5	72,2
10	82,1
100	96,9

By a similar approach it is possible to minimize the catalyst consumption for given constant efficiency. This means to consider the problem v = min, w = const which also has a unique solution and can be treated in the same way.

6. REFERENCES

[1] Concin R., New developments in acid hydrolysis. Österr. Chem. Z. 83 (1982), 47-51.

[2] Grethlein H., Chemical breakdown of cellulosic materials. J. Appl. Chem. Biotechnol. 28 (1978), 296-308.

[3] Rakowsky, Kinetik der Folgereaktionen 1. Ordnung, Z. phys. Chemie 57 (1906). 321-340.

[4] Saeman J.F., Kinetics of wood saccharification, Ind. Eng. Chem. 37 (1945), 43-52.

Case Studies in Dynamic Modelling of Large-Scale Wastewater Treatment Plants

Imre Takács[1], Gilles G. Patry[2], Bruce Watson[1], Bruce Gall[1]

[1]Hydromantis, Inc., 1685 Main Street West, Suite 302, Hamilton, Ontario, Canada, L8S 1G5
[2]University of Ottawa, Faculty of Engineering, P.O.Box 450 Station 'A', Ottawa, Ontario, Canada, K1N 6N5

ABSTRACT

Several large-scale wastewater treatment plants have been modelled successfully using the General Purpose Simulator, GPS-X. The program was found to be an invaluable tool in different areas, such as: a) analysis of operational scenarios; b) determination of sustained and peak capacity of plants (rated plant capacity) and c) investigation of plant expansion scenarios. Four cases involving simulation studies on large-scale plants are discussed.

INTRODUCTION

In the early stages of mathematical modelling of wastewater treatment plants research was oriented towards simulation of isolated unit processes. The emphasis of researchers and modellers has been placed mostly on the two key processes - activated sludge and final settling -, but other primary, tertiary, alternative or biosolids processes have also received considerable attention since. Therefore, it is now possible to model a full-scale wastewater treatment plant from headworks to effluent discharge, including process interactions. Is it feasible or practical though? What accuracy can one expect from such an exercise? Are these models applicable in performance analysis, optimization of operation, operator training, automation, planning or design?

These questions were motivating the development and application of the General Purpose Simulator (GPS-X) to full-scale plants. In the past few years, the program has been applied to more than 20 plants worldwide, including plants in Canada, U.S.A., Mexico, Venezuela, the United Kingdom, Sweden, and Japan. Technologies simulated include mostly conventional activated sludge, biological phosphorus removal and sequencing batch reactor processes. Some of the cases in question involve proprietary technologies and cannot be published. Four conventional, large wastewater plants were selected to demonstrate the benefits and challenges of such an exercise.

MODELLING FULL-SCALE TREATMENT PLANTS

When modelling a large-scale treatment plant, the modeller is faced with special challenges which do not exist in a controlled laboratory setting, or even when studying an isolated process on a full-scale plant. Some of the problems stemming from field conditions include:

- *Data requirements.* Plants rarely monitor or sample all important streams and components. The accuracy of flow measurement is usually hard (expensive) to verify due to the size of the plant. Monitoring equipments break down, power failures occur much more often than in the laboratory. Samples get lost or worse, sampled incorrectly. The modeller has to deal with different operational personnel who is changing several times during the experiment, as opposed to a familiar laboratory assistant.
- *Unknown interactions between processes.* When studying an isolated process, inevitable errors will disappear harmlessly at output points. In a large-scale treatment plant, one process influences the other, (e.g. the effect of sludge lines on influent loading), and the complexity of the simulation task is largely increased.
- *Plant has to stay in compliance while being tested.* In case of a real plant subject to strict effluent guidelines it is sometimes impossible or takes careful planning to perform experiments, stress tests, which are easy to perform on pilot scale units or isolated processes.

- *Unpredictable load and operation.* The plant is subjected to storms and other unforeseeable events which will play havoc with experimental plans and the performance of the plant.
- *Extra work load on operators.* Operators are requested to do extra work in addition to their usual job requirements and experience, sometimes without extra compensation.
- *Lack of modelling expertise.* In order for the plant model developed in a study with limited timeframe to become a valuable tool for planning and process analysis, the plant has to assign an engineer to the ongoing modelling task. Mathematical modelling requires some very specialized knowledge which is not readily available on a plant. On a large plant maintaining and operating the model is a substantial part time or full time job.
- *Budget.* Extra sampling to support modelling activities, additional monitors, the time of the process engineer committed to modelling require careful cost-benefit analysis on part of the plant, particularly smaller ones to be able to justify the investment in this high-end technology.
- *Hardware and software* also plays a key role in the implementation of mathematical models as computer programs. A full-scale plant potentially has a large number of unit processes which may require a powerful platform. Versatility is also a key requirement - the effective simulation of a large number of what-if scenarios on a large-scale plant demands a modular approach from the simulation vehicle.

CASE-STUDIES

The size of plants GPS-X was applied to ranges from 12 to 650 Ml/d. Out of more than twenty cases four were selected for discussion based on the results and conclusions which can be generalized for other similar plants. The four plants are listed in Table 1. All the selected plants are conventional biological technologies, consisting of primary clarification and the secondary activated sludge process, in all cases involving several parallel trains.

Table 1: GPS-X Case Studies

Plant name	Location	Design flow [Ml/d]	Technology		
			Step-feeding	Nitrification	Denitrification
Gold Bar	Edmonton, Canada	310	Yes	No	No
Coleshill	Coleshill, UK	55	No	Yes	Yes
Main	Toronto, Canada	650	Yes	No	No
Woodward	Hamilton, Canada	340	Yes	Partial	No

Gold Bar Wastewater Treatment Plant, Edmonton, Alberta, Canada. This facility is a typical non-nitrifying activated sludge plant with 8 parallel trains. In 1990, the average flow to the plant was 278 Ml/d, while the primary plant had a rated capacity of 950 Ml/d, and the peak capacity of the secondary was 430 Ml/d. The objective of the study was to develop a dynamic model of the plant for use in assessing the plant capacity as well as to identify effective operational strategies.

Three plant layouts of different complexity were developed using GPS-X. In the full layout, all eight individual processes were simulated separately, but simultaneously. Hydraulic information to support process modelling was developed from a plant survey and coded into GPS-X in the form of correlation equations. In the intermediate layout (as shown in Figure 1 for illustration), three process trains were simulated, according to the three different construction stages, while in the simplified one the whole plant was lumped into one process. Process models applied in this study were a temperature sensitive version of the IAWQ model [1], and a one-dimensional flux settler [2]. The models were calibrated for a number of flow conditions, including diurnal and extreme flow conditions. Stress tests were performed on specific unit processes to determine model parameters under critical loading conditions, including: a, primary clarifier stress test; b, final clarifier stress test; and c, step-feed test. Calibration of the model for carbonaceous BOD and suspended solids removal was successful. In general, the results agree very well with plant data, as shown for diurnal effluent suspended solids variation in Figure 2.

As part of this study, several changes in operational strategies were investigated using the calibrated model, including:

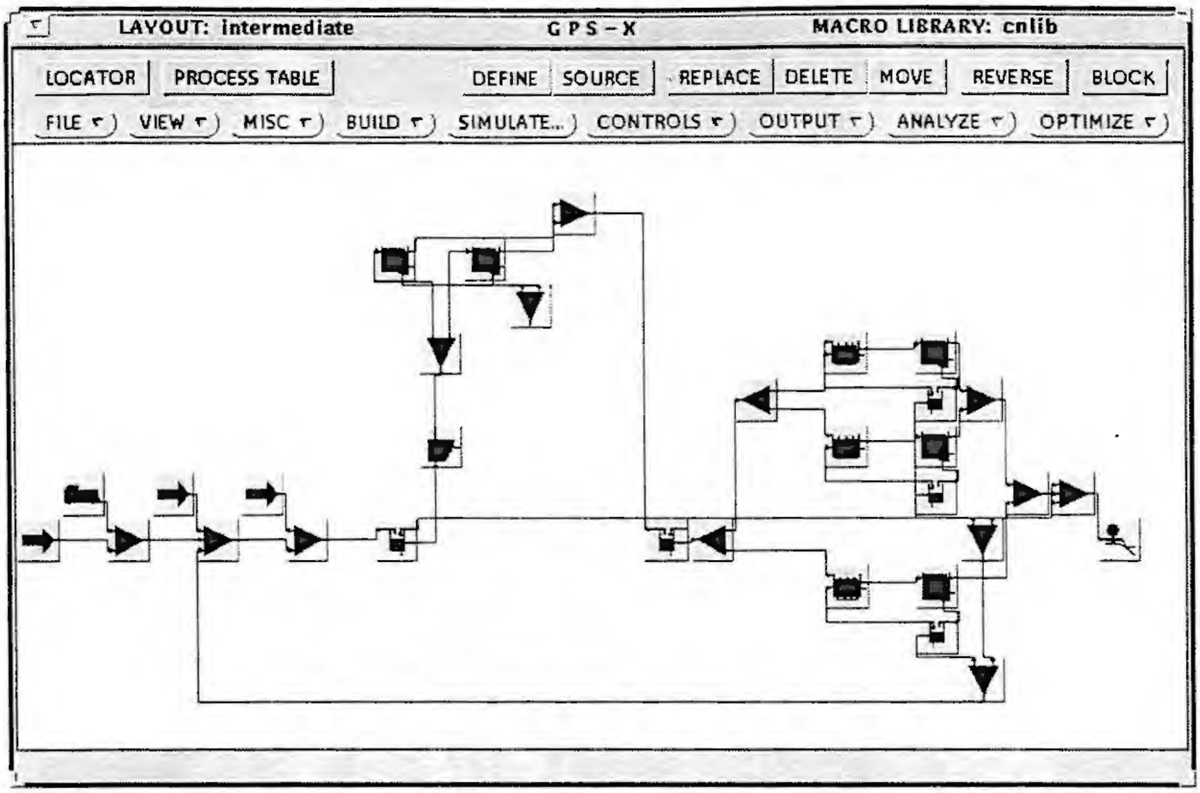


Figure 1. Intermediate layout (Gold Bar WWTP)

- the effect of hydraulic and organic load increases over the next 20 years and the necessary plant expansion,
- rising sludge problems in the final clarifiers,
- step-feed vs. plug-flow operation,
- the effect of even flow distribution on the process,
- the generation and use of operational charts using GPS-X,
- nitrification-denitrification operation.

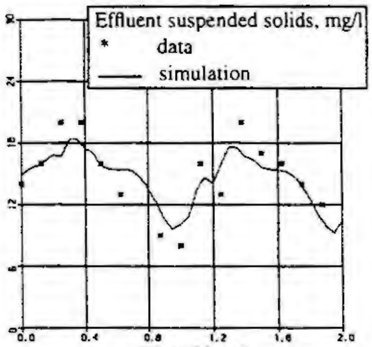


Figure 2. Diurnal calibration

A number of interesting conclusions were drawn from the case studies identified previously:

1, If the current SRT setpoint of 4.2 days is maintained, then 3 new biological reactors and secondary settlers will be required by 1997 and again by the year 2008 to prevent the settlers from overloading. These conclusions were reached by assuming a constant influent concentration under increasing hydraulic load. Settler failure could be prevented, and plant expansion delayed slightly, by decreasing the SRT of the plant. However, this strategy can have serious drawbacks. The mixed liquor suspended solids concentration can become so low in the aeration tanks that no proper flocculation and settling will take place in the settlers. This phenomenon is further augmented by step-feeding operation, at very low SRT's, BOD breakthrough might occur, resulting in serious excursions beyond the prescribed effluent limit.
2, The current operational practice to prevent denitrification in the settlers (low SRT, increased rake speed) is optimal. Final solution to the problem will be provided by an efficient nitrification-denitrification operational mode in the future.
3, Based on dry-weather flow loading conditions, a maximum of two of the aeration tanks and one clarifier can be taken off service without jeopardizing process performance. Ideally all eight final clarifiers should be used with six or seven aeration tanks, but current piping makes this choice impossible. Important energy savings can be realized by the optimal use of biological reactors.

4, Since the study was conducted during dry weather winter conditions, no data was available for the calibration and verification of the model under storm flow conditions. However, for the purpose of this investigation, it was decided to simulate the performance of the plant under heavy storm flow conditions (1200 Ml/d). The results show that the current bypass primary and secondary plant loading limits before bypass of 950 and 430 Ml/d, respectively, represent the maximum which the plant can handle without an operational upset.

5, Even though imbalances in flow distribution between the individual trains were significant (10-15%), the difference between the combined effluent suspended solids of the plant and that of the aggregated (single process) model was negligible (less than 1 mg/l).

Generally, it was found during the study that data collection is very expensive without online monitoring equipment. To reduce data requirements, aggregation of different process trains is a viable approach even when the flow distribution is significantly different of individual trains.

Coleshill Wastewater Treatment Plant, UK. Severn Trent Water Ltd. (STW) operates about 70 activated sludge plants. In the future some will have to meet COD, total N, and total P consent as well as BOD, SS, and NH4-N requirements. STW recognizes that computer simulation can play an important role in understanding the operation of large-scale sewage works, allowing them to meet consents more effectively.

The Coleshill sewage works is a 55 Ml/d conventional diffused air nitrifying activated sludge plant. The plant meets an effluent requirement of 25/45/20 (BOD/SS/ NH_4^+-N) based on a 95 %-ile and 3 samples per week. Coleshill is probably one of the better instrumented plants in the world with over 150 analogue inputs to the site monitoring system, including:

8 ultrasonic flow level sensors,
16 MADOS III dissolved oxygen probes,
4 mixed liquor suspended solids probes,
8 sludge blanket probes,
4 MSL respirometers,
3 effluent suspended solids probes,
3 ion-selective ammonia probes,
3 pH sensors, and
3 temperature probes.

The last four parameters are measured on the settled sewage (East and West) as well as the final effluent.

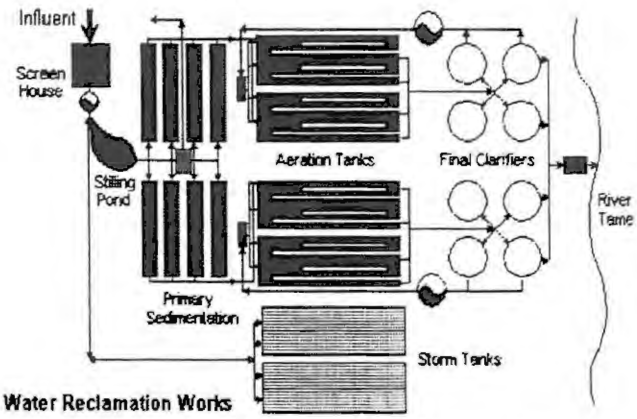


Figure 3. Coleshill Water Pollution Control Plant

The purpose of this investigation was to develop a calibrated model of the Coleshill sewage works and to link the model to the site monitoring system for operational control purposes. A process flow diagram derived from the physical layout of the plant was developed using the GPS-X (see Figure 3).

Following a review of historical plant performance, the model was calibrated to a number of flow conditions in order to assess the stability and sensitivity of model parameters. Operational parameters such as waste and return flowrates, DO setpoints, etc. were identified and specified as inputs to the model. A steady-state solution of the model calibrated to average historical plant records allowed for the identification of the preliminary set of model parameters. Results of this analysis are shown in Table 2. A number of simulations under dynamic conditions were performed. The results of a four day simulation on the effluent ammonia concentration are shown in Figure 4. In this example, the first 2 days were used for model calibration, while the next 2 days were used for model verification. The discontinuities in the observed data (every 12 hours) are associated with the self-calibrating feature of the instrument. It is interesting to note that, following calibration, the instrument typically returns to the simulated

Table 2: Steady-state results

Parameters	Observed	Simulated
Primary effluent		
SS (mg/l)	115	115
BOD_5 (mg/l)	110	108
NH_4^+-N (mg N/l)	27	27
Aeration tanks		
MLSS (mg/l)	2550	2540
DO (pass 3) (mg/l)	3.7	3.7
Final effluent		
SS (mg/l)	3.5	3.8
BOD_5 (mg/l)	7.5	6.7
NH_4^+-N (mg N/l)	9.6	9.5

value. This finding can be used as a warning to either update the calibration of the simulator or the monitoring instrument for fouling.

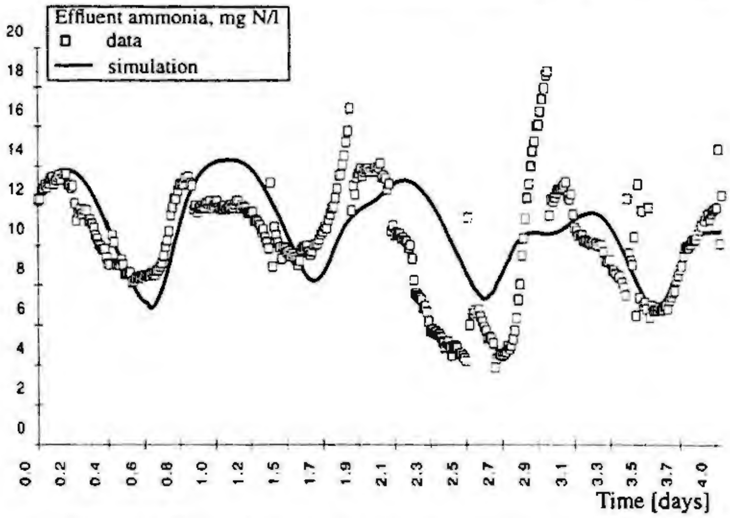


Figure 4. Diurnal ammonia variation

Main Treatment Plant, Toronto, Canada. The GPS-X was used to develop a dynamic mathematical model of the largest wastewater treatment plant in Canada. The plant treats an average of 650 Ml/d influent from Metropolitan Toronto, by primary sedimentation and the basic activated sludge process (non-nitrifying). Only the liquid stream processes were simulated in this study, however, the effect of sidestreams from sludge treatment can be simulated. The model was calibrated and verified based on the results of a 9 month long audit undertaken by CH_2MHill. Initially, several monthly steady-state calibrations were performed in order to find the range of model parameters. Then, several diurnal and storm events were selected from a database containing almost one year of continuous flow and quality data. During the calibration it became apparent that in some of the stress test cases the one-dimensional model could not handle the effect of extreme storm flows on clarifier performance. A two-dimensional fluid dynamic model was incorporated in GPS-X and has been used to determine the flow pattern in the rectangular final clarifiers. Long-term continuous simulations were also performed with a duration of three to nine months to check the model's capability in matching trends in the data. MLSS concentration in a typical three-week period is shown in Figure 5. This specific graph pointed to potential suspended solids probe malfunction during the first 10 days which was confirmed by grab samples and mass balance calculations. Once the probe was recalibrated, the measured data fits the simulation reasonably well. The output from these runs was also processed in probability distribution form.

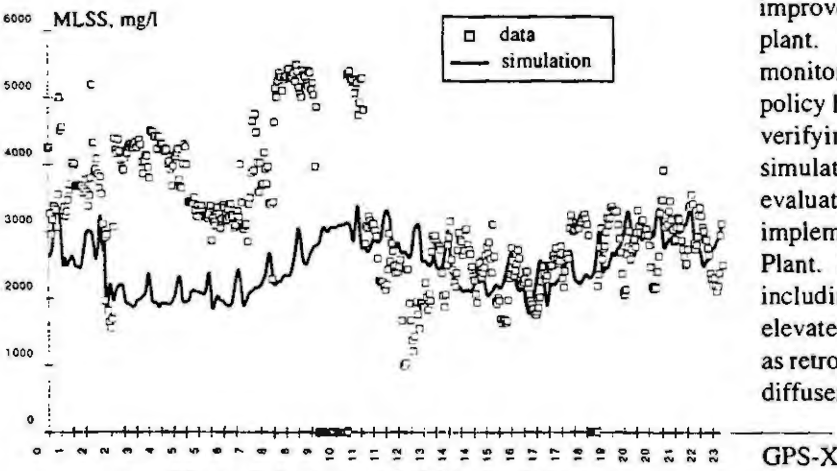


Figure 5. Long-term MLSS simulation

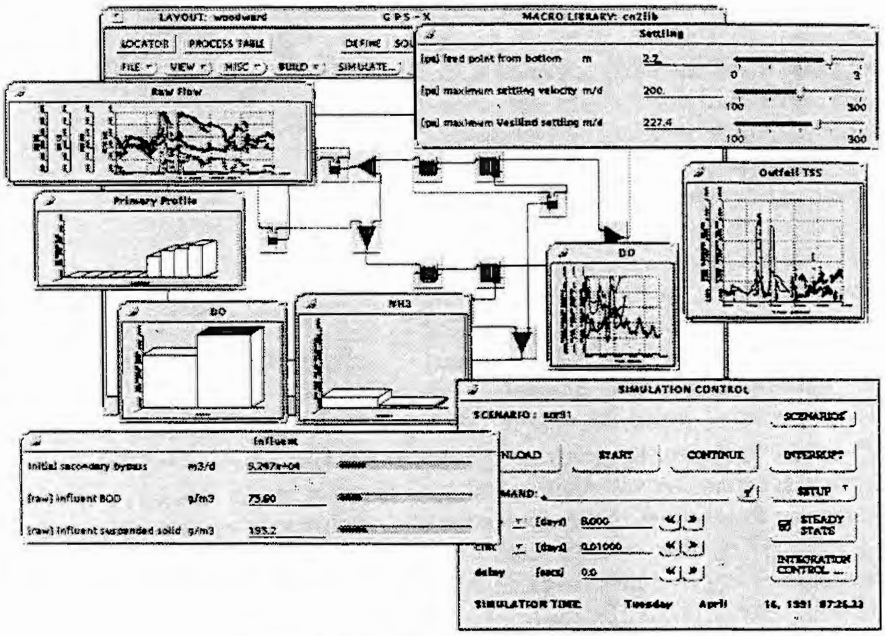
The study pointed to several possible improvements in the daily operation of the plant. Specifically, inadequate flow monitoring and the improvement of bypassing policy has been raised by studying the data and verifying its consistency by dynamic simulations. The model was also used to evaluate different expansion scenarios, e.g. the implementation of nitrification on the Main Plant. Several alternatives were evaluated, including the feasibility of maintaining an elevated MLSS level to increase SRT, as well as retrofitting the aeration system for fine-pore diffusers.

GPS-X has been installed on the plant and since then has been used to determine best operational practises during emergency pump shutdowns and construction periods. In one instance, e.g., it provided maintenance with an estimation of the time-frame while a recycle pump can be safely taken out of service. Analyzing the effect of this operation in GPS-X, it was determined that there is a four hour window while the maintenance or replacement can be performed without hampering effluent quality.

Woodward Avenue, Hamilton, Ontario, Canada. The purpose of this project was to develop a dynamic model of the Woodward Avenue treatment plant. The model was used to provide support information for the development of a Facility Plan Report, addressing the long-term development needs of the plant. The calibrated model has been installed at the plant site.

The Hamilton Woodward treatment plant is a conventional activated sludge plant that is currently designed to treat an average combined sewer flow of 340 Ml/d. The plant is also able to cope with a 600 Ml/d short term peak flow. All flows above 600 ML/d bypass the entire plant. Flows above 410 Ml/d, and below 600 Ml/d, will be bypassed around the secondary facility by operators if the final clarifier sludge blanket is too high. Another unusual aspect of the plant is that the primary clarifiers have a maximum hydraulic capacity of 275 ML/d. This means that all flow in excess of 275 Ml/d bypasses primary treatment and receives only secondary treatment.

The complex plant operation, which includes several by-passes, two different technologies from two plant expansions, and the shallow rectrangular clarifiers equipped with a circular sludge scraper arm proved to be a challenge during plant calibration. GPS-X was used to organize the large amount of data generated by the two process trains visually to aid in calibration. A typical screen including the plant layout, manual and automatic controllers, and linear as well as histogram output of the simulation results along with monitored plant data during the calibration process is shown in Figure 6. for illustration.



Settling

(pa) feed point from bottom m 2.2

(pa) maximum settling velocity m/d 200.

(pa) maximum Vesilind settling m/d 227.4

Influent

initial secondary bypass m3/d 5.247e+04

(raw) influent BOD g/m3 75.60

(raw) influent suspended solid g/m3 195.2

SIMULATION CONTROL

SCENARIO : apr91

[days] 8.000

[days] 0.01000

delay [secs] 0.0

SIMULATION TIME Tuesday April 16, 1991 07:26.33

Figure 6. GPS-X interactive simulation interface.

Selected results of a dynamic simulation for the period of April 13 to 18, 1991 are shown in Figure 7. The Figure includes flow to the north side aeration basins (which provides approximately 66% of the biological treatment), and north side effluent suspended solids (actual and simulated). Simulated north side flows are based on total plant influent flow, flow splits between the north and south plant, and the effects of operator control rules programmed into the simulation code. During periods of high flow to the north side of the plant, the effluent suspended solids increases significantly. There is good agreement between simulated and actual effluent suspended solids.

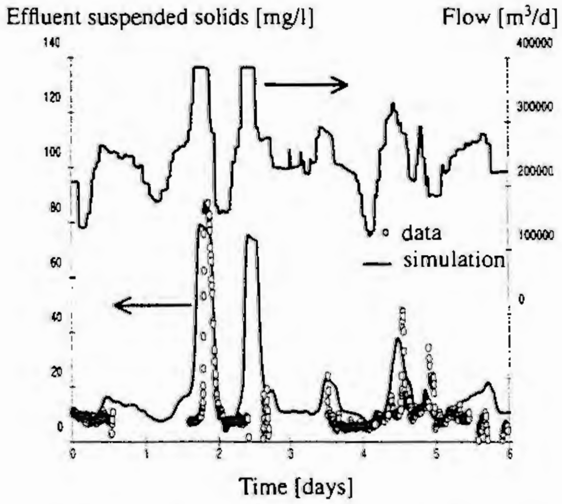


Figure 7. Effect of storm flow on suspended solids

The calibrated model was used to evaluate expansion scenarios for two different expected flow levels (400 and 600 Ml/d), and three different effluent standards. Necessary expansion of primary and final settling capacities was established by traditional design methods and verified with the model. The proposed conversion of the aeration tanks to plug-flow reactors equipped with an anoxic zone was also simulated, with special attention of the required aeration capacity and taper.

CONCLUSIONS

Out of more then twenty case studies four was selected to demonstrate the challenges and benefits of simulation as a tool on large-scale wastewater treatment plants, It was found that activated sludge models and primary and final clarification models are robust enough today to be used in an industrial environment even when full identification of every model parameter is not possible or feasible.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the management and the team of workers at the Gold Bar Wastewater Treatment Plant, Coleshill Sewage Works, Main Plant and Woodward Avenue Wastewater Treatment Plant. Without the help of these individuals successful completion of the above studies would not have been possible.

REFERENCES

[1] Henze, M., Grady Jr., C.P.L., Gujer, W., Marais, G.v.R., Matsuo, T. (1987). Activated Sludge Model No. 1. *IAWPRC Task Group on Mathematical Modelling for Design and Operation of Biological Wastewater Treatment*, IAWPRC, London, England.

[2] Takács, I., Patry, G.G. and Nolasco, D. (1991). A Dynamic Model of the Clarification-Thickening Process. *Wat. Res.* **25**, No 10. 1263-1271.

MODELLING AND SIMULATION OF FINAL CLARIFIERS IN WASTEWATER TREATMENT PLANTS

M. KOEHNE, K. HOEN and M. SCHUHEN

ZESS Zentrum für Sensorsysteme und IMR Institut für Mechanik und Regelungstechnik

Universität Siegen, D-57068 Siegen

Abstract. The complex dynamic process of clarification, settling and thickening in final clarifiers of wastewater treatment plants is very sensitive to changes in hydraulic and organic loading. Vitasovic's solid flux model and several modifications of the sludge settling velocity in this model are investigated and applied to the sewage plant of Siegen. The underflow and effluent suspended solids concentrations are used for comparison of simulation results. Hydraulic overloading during rainfall periods are modelled very well. However, the effluent suspended solids concentration during dry weather periods with changes in organic loading is not yet sufficiently modelled by all the proposed modifications of the sludge settling model.

1. INTRODUCTION

The activated sludge process (Fig. 1) is currently the most widely used biological wastewater treatment process in the developed world /Gray90/. In this process a large population of micro-organisms is maintained in suspension in a tank through which wastewater passes continuously (with volumetric flow rate Q). Air or oxygen is supplied and purification takes place in a series of steps. Bacteria utilize the organic material to yield new cells and provide energy /Jame84/. The synthesized cells (sludge) are separated from the clarified effluent by means of sedimentation in a solids-liquid separator (secondary settler or final clarifier). The main portion of the active bacteria (activated sludge) is recycled to the inlet of the aeration tank (flow rate R) and the smaller portion is removed (excess sludge or secondary sludge, flow rate S).

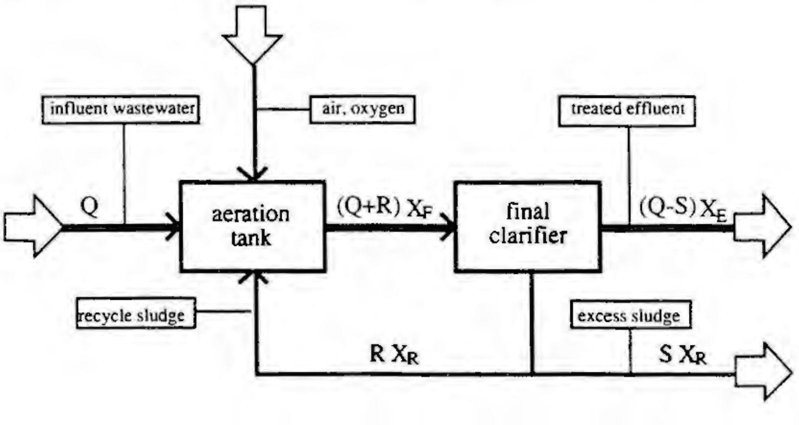


Fig. 1:
Flow scheme of the activated sludge process.

The micro-organisms are maintained in the aeration tank in the form of flocs, which are dispersed throughout the liquor (mixed liquor suspended solids MLSS). Purification is achieved as a result of micro-organism metabolism, which depends on the presence of sufficient oxygen. A portion of the carbonaceaous organic waste is converted to inorganic material by oxidation. The other part of the waste is converted into new cells. In addition nitrifying bacteria will convert amonia to nitrates. The rate depends on the retention characteristics of the particular process /Jame84/. In other words: The very operation of an activated sludge plant is based on the transformation of *soluble organic* matter into *settleable suspended* matter /TPN90/. Therefore, the separation of solids and fluid is undoubtedly one of the most important processes in wastewater treatment. Final clarifiers should produce an effluent with a very low suspended solids concentration and an underflow or recycled sludge with a high concentration of biomass for use in the aeration tank (biochemical reactor).

In recent years numerous mathematical models of the complex biokinetic processes in activated sludge plants have been developed. The most wellknown and widely used model is the IAWPRC model No. 1 /HGGM86/, which also considered in this paper. However, this model takes not into account the settling characteristics of activated sludge in final clarifiers. Therefore, the IAWPRC model has been combined with the multi-layer flux model of Vitasovic /Vita89/. Several modifications of this dynamic model of the clarification-thickening process by Takács /TPA90, /TPA91/, Otterpohl /OtFr92/, as well as results of an extended new activated sludge model of Gujer /GuKa92/ are considered.

The aim of this paper is two-fold: Comparison of the different modifications of the flux model and simulation of the final clarifiers of the wastewater treatment plant of Siegen. The capability of predicting or simulating the observed and measured settler failures is of specific interest. In the following chapters the flux model and the different modifications of the settling velocity in this model are briefly discussed. Simulation results are presented for comparison. Both influent flow situations are considered, dry weather and storm water influent.

2. MODELLING OF FINAL CLARIFIERS

In this paper, final clarifiers with vertical flow characteristics are considered (Fig. 2). One-dimensional models are able to describe the space dependent sludge concentration profile along the vertical coordinate z, which can be measured with vertically moving turbidity sensors. A typical measured profile is shown in Fig. 2. The sludge blanket height and the increasing concentration can be observed.

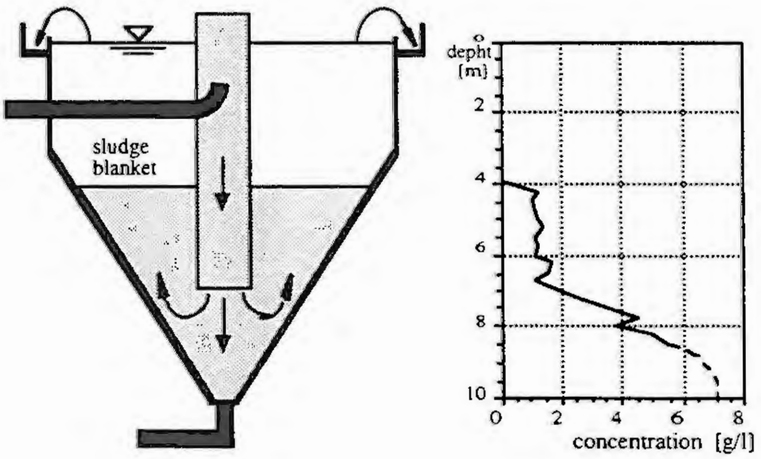


Fig. 2:
Final clarifier of the wastewater treatment plant of Siegen (squared pyramid) and a measured sludge profil.

Sewage plant operators have observed that the settling and clarification process is very sensitive to changes in hydraulic loading and changes in organic loading or the characteristics of biomass. The hydraulic conditions can be accounted much more readily in mathematical modelling than the characteristics of biomass, which depend on the layout of the plant, food to mass ratio, oxygen concentration in the aeration tank, type of organics in feed etc. /TPN90/.

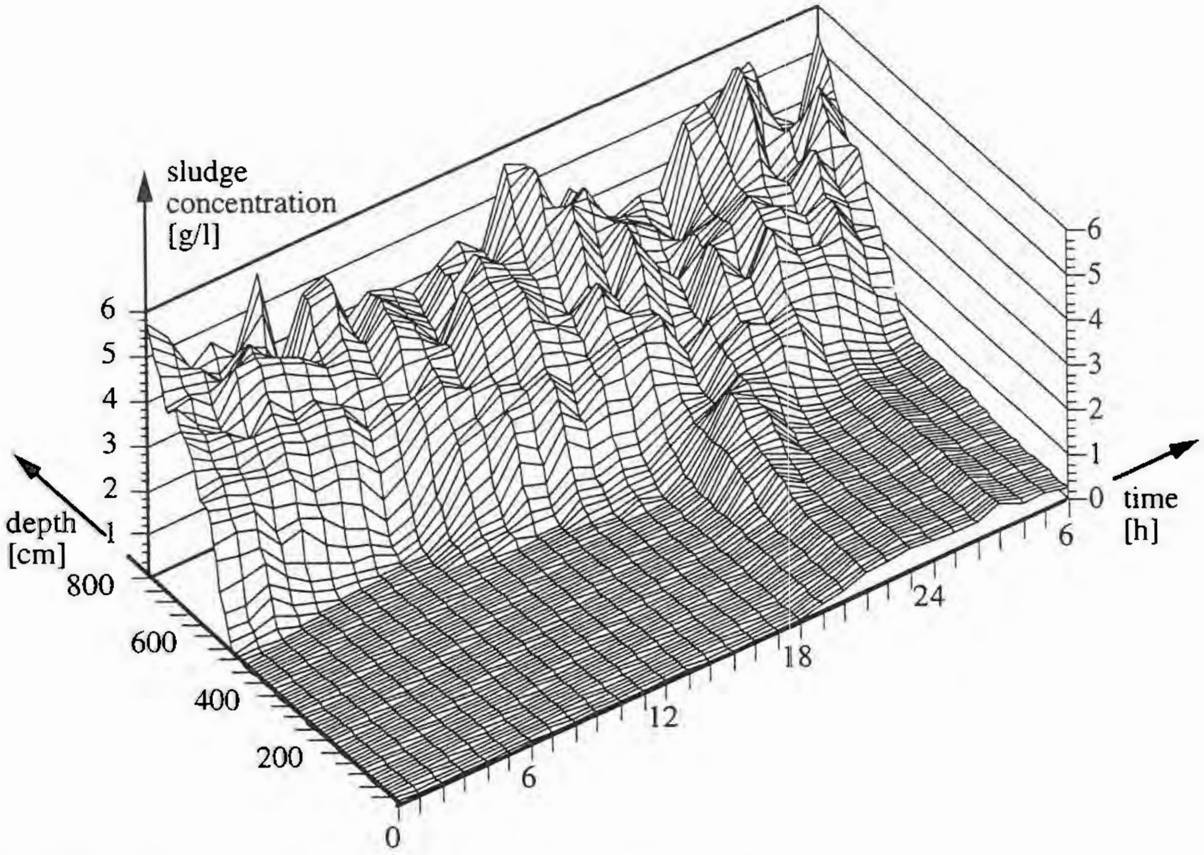


Fig. 3: Hourly measured sludge concentration profiles in a final clarifier.

A typical failure of a final clarifier due to hydraulic overloading can be observed in Fig. 3, where the concentration profiles are measured every hour. High effluent suspended solids concentrations occur after a heavy rainfall (storm water influent flow).

The model of Vitasovic /Vita89/ is based on the solids flux theory and provides a comprehensive description of the observed thickening function of the final clarifier, which is split into a finite number of uniform horizontal layers. See /Vita89/ and /TPN91/ for details. The solids concentration X_i in each layer of constant height h can be calculated from a simple mass balance around each layer:

$$h\frac{dX_i}{dt} = \underbrace{\frac{Q_i}{A_i}(X_{i-1} - X_i)}_{\text{bulk movement}} + \underbrace{\min(v_{i-1}X_{i-1}, v_iX_i) - \min(v_iX_i, v_{i+1}X_{i+1})}_{\text{gravity settling}} \qquad (1)$$

with Q_i = volumetric flow rate, A_i = cross sectional area, v_i = settling velocity, i = 1,2,...n. Modifications of this equation are necessary for the top layer (i = 1, $x_1 = x_E$ = effluent suspended solids concentration), the bottom layer (i = n, $x_n = x_R$ = recycled sludge concentration), and the feed layer (i = m, feed concentration x_F and feed flow rate $Q_F = Q + R$). Essential are the assumptions about the settling velocity in different models. In /Vita89/ a widely accepted empirical equation is used, which was originally proposed by Vesilind:

$$v_i = v_0 e^{-aX_i} \qquad \text{(Model of Vitasovic, Vesilind)} \qquad (2)$$

where the main maximum settling velocity v_0 and the exponential decreasing parameter a are empirical constants. In /TPN90/ and /TPN91/ it has been assumed by Takács et al. that the mixed liquor solids entering the final clarifier in the feed layer can be thought to consist of unsettleable, slowly settling, and rapidly settling fractions. A double exponential settling velocity model is proposed to account for the three settling fractions:

$$v_i = v_0\left[e^{-r_h(X_i - X_{min})} - e^{-r_p(X_i - X_{min})}\right] \qquad \text{(Model of Takács et al.)} \qquad (3)$$

where X_{min} = $f_{ns}X_F$ = minimum attainable effluent suspended solids concentration
f_{ns} = non-settleable influent suspended solids fraction
r_h = settling parameter characteristic of hindered settling (rapidly settling)
r_p = settling parameter characteristic of low solids concentration (slowly settling) .

The two exponential settling velocity models (2) and (3) are compared in Fig. 4.

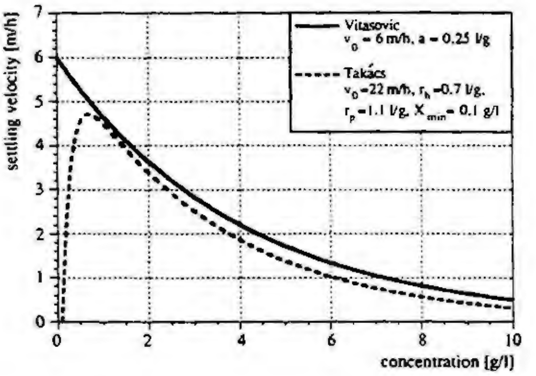


Fig. 4:
Comparison of two settling velocity models.

Another modification is proposed in /OtFr92/. The settling flux terms in equation (1) are multiplied by a so-called "Ω-function", which has been originally introduced by Härtel /Härt90/. This function is dependent on the depth z, sludge volume index SVI, influent solids concentration X_F, and few geometrical parameters:

$$\Omega = f(z, SVI, X_F) \rightarrow \Omega_i = f_i(SVI, X_F). \qquad \text{(Härtel's } \Omega\text{–function)} \qquad (4a)$$

The solids in the secondary clarifier are divided into two components: macro flocs and mall solids. Macro flocs are assumed to settle according to the function of Härtel, and small solids are assumed to settle with a very small constant velocity:

$$v_{i,Floc} = v_0(SVI)e^{-a(SVI)X_i} \qquad \text{(Model of Otterpohl et al.)} \qquad (4b)$$

$$v_{i,Sol} = \text{constant}. \qquad (4c)$$

The parameters v_0, a, depend on the SVI. The influent concentration X_F has to be divided into two fractions

$$X_{F,Floc} = (1-f)X_F \qquad \text{and} \qquad X_{F,Sol} = fX_F. \qquad (4d)$$

The parameter f has to be suitably chosen. See /OtFr92/ for more details.
Gujer et al. /GuKa92/ extended the activated sludge model Nr. 1 /HGGM86/ by introducing flocforming heterotrophic organisms (X_{Flo}), filamentous heterotrophic organisms (X_{Fil}), and "nocardia" type organisms (X_{Noc}) instead of the usually assumed single population of heterotrophic organisms (X_H). The latter is not essential with respect to the modelling of final clari-

fiers and is therefore neglected in this paper. Starting from this new biokinetic model /GuKa92/, Hoen and Schuhen proposed another model of the settling velocity in the final clarifier, which takes into account both filamentous and flocforming heterotrophic organisms:

$$v_i = v_0 e^{-aX_i}\left[(1-b)e^{-k\frac{X_{i,Fil}}{X_{i,Flo}}} + b\right] \qquad \text{(Model of Hoen, Schuhen)} \qquad (5)$$

with the empirical parameters v_0, a, similar to equation (2) and the additional parameters k and b, where b describes the maximum settling velocity of filamentous organisms. In the case $X_{i,Flo} >> X_{i,Fil}$ the model (5) converges to model (2).

3. SIMULATION OF FINAL CLARIFIERS

During 1985 - 90, one third of the sewage plant of the city of Siegen has been used as full scale pilot plant for research and experiments. Suitable measurements are available for model parameter adjustment, model verification and validation. This plant is highly loaded with respect to substrate concentration (BOD or COD). A group of four final clarifiers with quadratic cross sectional area (squared pyramid) and vertical flow characteristics are hydraulically overloaded during storm water influent flow. High effluent sludge concentrations are observed (Fig. 2). Models of final clarifiers should describe both, stormwater and dry weather influent flow situations.

The activated sludge process is modeled by a simplified version of the IAWPRC model /HGGM86/, without nitrification and denitrification, considering the readily biodegradable substrate S_S and the active heterotrophic biomass $X_H = X$ as main state variables. In combination with equation (5) of the settling velocity, the biokinetic model of Gujer et al. /GuKa92/ with flocforming and filamentous organisms, $X = X_{Flo} + X_{Fil}$, is considered. It is assumed that these two organisms can grow on two substrates, inflowing soluble COD, S_S, and soluble hydrolysis products, S_H.

Vitasovic's final clarifier model with n = 10 layers of constant thickness h = 1.0 m, variable area A_i, and influent in the fourth layer is assumed. A new wastewater treatment plant simulation system KSIM /Scuh93/, developed at the Institute of Mechanics and Control Engineering (IMR) of the University of Siegen, has been applied for simulation of different influent flow situations. KSIM is based on the commonly used programs MATLAB and SIMULINK. KSIM allows for easy interactive choice of different activated sludge models, different models of primary and secondary clarifiers, several flowschemes, process control strategies etc.

Fig. 5 shows simulation results of the storm water situation. Two and a half days (60 hours) have been considered for simulation. Two heavy rainfalls can be observed from the influent flow rate Q(t) after 20 and 40 hours. The measured activated sludge concentration of the aerator (biochemical reactor) has been used as feed concentration of the final clarifier. The effluent suspended solids concentration $X_E(t)$ and the recycled sludge concentration $X_R(t)$ have been chosen for valuation of the different models. The measurement range of X_E was restricted to 25 mg/l, therefore, the high effluent concentration due to storm water influent can only be observed from simulation. The simulated underflow sludge concentration $c_R(t)$ corresponds sufficiently well with the measured values. The resultes of Fig. 5 have been obtained with the multilayered clarifier model of Vitasovic (Equ. 1) and the settling velocity model (2). Equivalent results have been obtained with the other velocity models (3) - (5).

The simulations results are quite different in the dry weather situation. None of the four settling velocity models combined with Vitasovic's layered model of final clarifiers is able to simulate effluent suspended solids concentrations $X_E(t)$ which are in sufficient agreement with measurement results (Fig. 6). However, is should be regarded, that the concentration level is rather low (between 1 and 3 mg/l) in the dry weather situation.

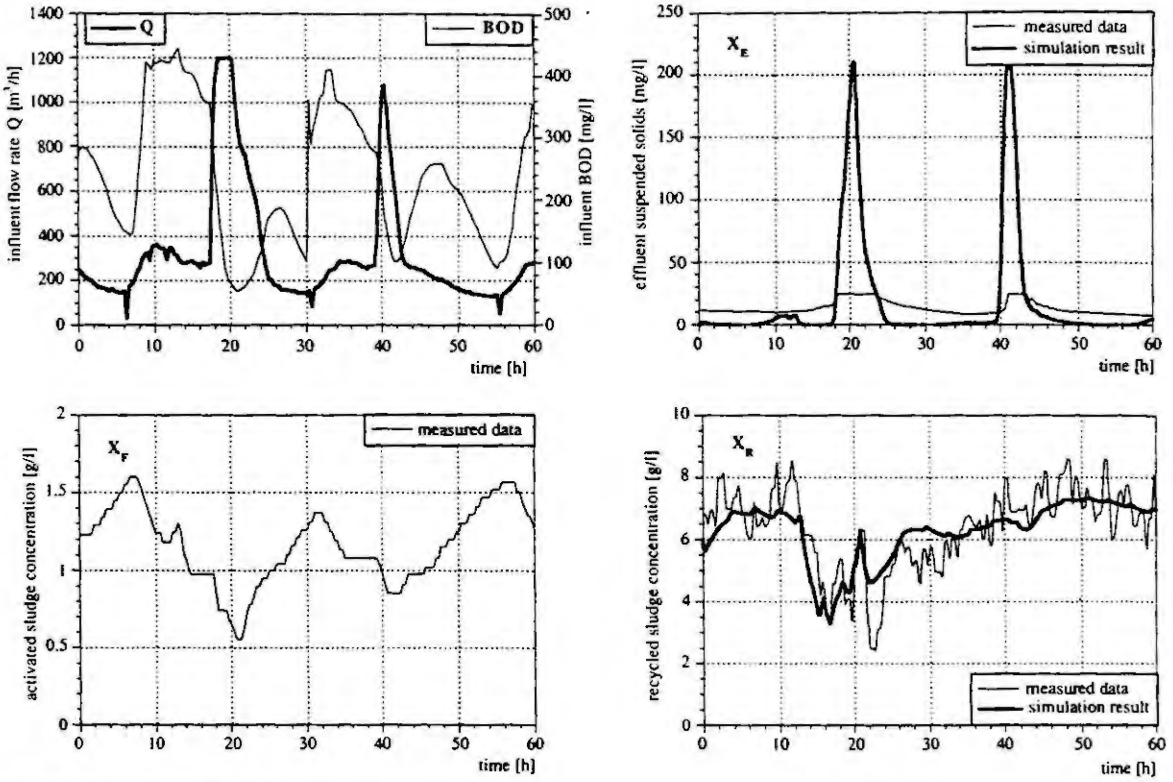


Fig. 5: Simulation of the strom water situation

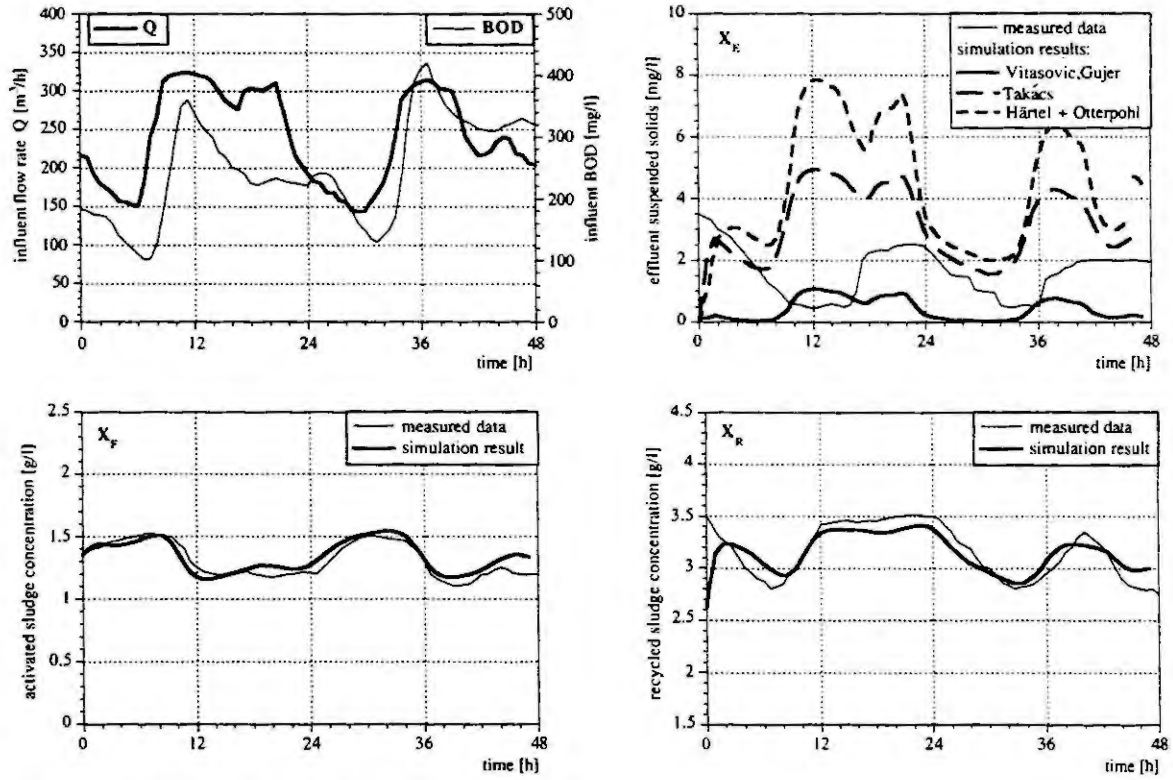


Fig. 6: Simulation of the dry weather situation.

4. RESULTS

Models for complex acitvated sludge processes are available, which can simulate the dynamic behavior of a large variety of activated sludge systems /HGGM86/, /GuKa92/. Models, which are able to simulate the settling characteristics of the activated sludge in final clarifiers are still in an early phase of development. In this paper the layered model of Vitasovic /Vita89/ and several models of the sludge settling velocity have been investigated and compared. Measured data from a full scale pilot plant have been used for model verification and validation. From simulation results it can be concluded, that storm water flow situations (hydraulic overload) are modelled very well by all considered models. But none of the models was able to simulate dry weather flow situations (organic overload) with sufficient accuracy. Therefore, future research activities should be directed to improve the models of final clairifiers, espicially modells for circular clarifiers with horizontal flow characteristics are needed.

5. REFERENCES

/Gray90/ Gray, N. F.: Activated sludge, Theory and Practice
Oxford University, 1990

/GuKa92/ Gujer, W.; Kappeler, J.: Modelling population dynamics in activated sludge systems
Wat.Sci.Tech. 25(1992), No. 6, pp. 93-105

/Härt90/ Härtel, L.: Modellansätze zur dynamischen Simulation des Belebtschlammverfahrens
Dissertation, TH-Darmstadt, WAR-Schriftenreihe, Band 47, 1990

/HGGM86/ Henze, M., Grady, C.P.L, Gujer, W., Marais, G.v.R, Matsuo, T.:
(IAWPRC Task Group on Mathematical Modelling for Design and Operation of Biological Wastewater Treatment)
Activated Sludge Model No.1, Report, 1986

/Jame84/ James, A. (Ed.): An Introduction to Water Quality Modelling
John Wiley, Chichester, 1984, pp. 182-196

/OtFr92/ Otterpohl, R., Freund, M.: Dynamic Models For Clarifiers Of Activated Sludge Plants With Dry And Wet Weather Flows
Wat. Sci. Tech. Vol. 26, No 5-6, pp 1391-1400, 1992

/Scuh93/ Schuhen, M.: Handbuch für das Simulationssystem KSIM
IMR-Bericht 14-93, Diplomarbeit UGH Siegen, 1993 (unpublished)

/TPN90/ Takács, I., Patry, G.G., Nolasco, D.: A Generalized Dynamic Model of the Thickening/Clarification Process
Advances in Water Pollution Control Series, 10,
IAWPRC Workshop, Elmswood, 1986, pp.487-493

/TPN91/ Takács, I., Patry, G.G., Nolasco, D.: A Dynamic Model of the Clarification-Thinkening Process '
Wat. Res. Vol. 25, No.10, pp. 1263-1271, 1991

/Vita89/ Vitasovic, Zdenko: Continuous Settler Operation: A Dynamic Model
Dynamic Modeling and Expert Systems in Wastewater Engineering
edited by Patry G.G. and Chapman D., Lewis Publishers, Michigan, 1989

MODELING AND SIMULATION OF A MUNICIPAL WASTE WATER TREATMENT PLANT BY ACTIVATED SLUDGE

M.N. PONS[†], N. ROCHE[†,*], O. POTIER[†], R. BENDOUNAN[†],
L. PEREIRA[†], C. PROST[†,*] and J.P. CORRIOU[†]
[†]Laboratoire des Sciences du Génie Chimique, CNRS-ENSIC-INPL
BP 451, F-54001 Nancy Cedex, France
[*]Université Nancy I, Le Montet
F-54601 Villers-les-Nancy Cedex, France

Abstract. A software has been developed for the simulation of a municipal waste water treatment plant by activated sludge including a primary settler, an aerator of the canal type and a secondary settler with sludge recycle. The daily variations in load and flowrate are taken into consideration.

1. INTRODUCTION

Waste water treatment plants by activated sludge are the most used systems for the treatment of domestic waste waters, mixed or not with industrial effluents. The quality of the purified water should be maintained within strict limits in spite of variations (periodical or not) in load and flowrate, of biomass variability, and of hydrodynamical characteristics of the plant. Large quantities of oxygen should be supplied, either by gas injection (air or oxygen) or by surface aeration, to avoid any limitation of microbial activity. On the other hand, sludge overproduction should be avoided as their elimination is costly.

The design of new plants, the development of new control strategies and the training of operators would be improved by the availability of a dedicated simulation software. The aim of this paper is the description of such a tool, devoted to the simulation of a plant including a primary settler, an aeration tank of the canal-type and a secondary settler with sludge recycle. The software has been developed using experience gained on the waste water plant of a large french city (300 000 equiv. inh.).

2. MATERIALS AND METHODS

The actual waste water plant includes three primary settlers (total volume ≈ 12 000m^3), three aeration tanks with air injection (total volume ≈ 9 000 m^3) and three secondary settlers (total volume ≈ 15 000 m^3). One-week long monitorings have been previously conducted [4]. Turbidity and UV-light and visible light absorption have been monitored on water at the plant inlet, at the primary settlers outlet and at the plant outlet over a 8-month period. Daily average pollution values have been provided by the Plant Managing Company. Weather data have been provided by the National Weather Agency and Nancy District Authority.

Residence time liquid phase distributions have been measured on one of the primary settlers and on the sub-system aeration tank-secondary settler. The tracer was lithium chloride. Lithium was measured by atomic absorption after sample filtration.

3. MODELING

The simulated plant includes a primary settler, an aeration tank and a secondary settler with cell recycle and represents one third of the actual plant. Water flows by gravity and the volume of each unit is constant. Detailed equations are here omitted for sake of simplicity.They are based on simple mass balances.

3.1. Biological model

The pollution is essentially of domestic nature. It is usually divided into three parts, according to its biodegradability and solubility [5]:

- biodegradable polution
- non biodegradable non soluble (S_{snsi})
- non biodegradable soluble (S_{ssi})

The biodegradable pollution is divided into a soluble part (Ss), which is rapidly metabolized, and a non soluble part (S_{sns}), which is hydrolyzed into a reserve substrate (S_r) transfered across the cell wall before consumption.

The biological model is based on IAWPRC model [1, 3]. Nitrification and denitrification processes are not taken into account. Biomass consumes the biodegradable pollution. Biomass death, favored by oxygen limitaiton or excess, produces a new pollution, more or less biodegradable. Monod's type kinetics are used with constant yields.

3.2. Aeration tank model

After analysis of the hydraulic residence time distributions, the aeration tank is modeled by a series of n units: each unit is built from a well-mixed zone, with backmixing (factor α), and a dead zone [7]. α varies with flowrate according to the following relationship:

$$\alpha = 8.5 - \left(81.25 - \frac{162}{J}\right)^{1/2}$$

with $J = 1 + 1000/\tau$ where τ is the aeration tank global residence time. Biological reaction is taking place only in the well-mixed zone which is aerated. Aeration is represented by a mass transfer coefficient (K_la) which can have a different value in each unit.

3.3. Settlers model

The primary settler retains about 50% of the non soluble polllution entering the plant and the secondary settler retains about 90% of the insoluble pollution entering the aeration tank. The same model structure of layers of annular zones of equal surface has been selected for both settlers. The model is derived from Kabouris and Georgakakos'work [2]. The differences between both settlers are:

- for the settling velocity: the average settling velocity is taken as 0.8 m/h.Vesilind's relationship is used for the secondary settler [8]:

$v = k \exp(-nX_s)$

where X_s is the total concentration in suspended matter. Typical value for k and n are respectively 6.4 (m/h) and 0.4 but they can be varied according to sludge settleability.

- for hydrodynamics: the average wastage rate of the primary settler is 0.01. The recycle flowrate of the secondary settler is equal to the water flowrate (recycle rate β = 0.98).

3.3. Model perturbations

The model perturbations are derived from Neveux *et al.*' work [4]. Pollution and flowrate are represented by functions of the type:

$$Y(t) = \left(Y_{moy} + y_1 \cos\left(\frac{t}{12\pi} - \theta_1\right) + y_2 \cos\left(\frac{t}{6\pi} - \theta_2\right)\right)(1 + r(t)^*\nu)$$

where r(t) is a random parameter varying between -0.5 and 0.5 and ν, the noise amplitude. During heavy rain periods only a part of the total flowrate is treated.

3.4. Actual perturbations

Data files produced by the on-line monitoring computer can be coupled to the software. A calibration curve has been established to relate the soluble pollution with the UV light absorbance and the non soluble pollution to the turbidity. Actual flowrate is measured every 5 min and each water is analyzed every 15 min.

3.5. Implementation

The software is written in FORTRAN 77 and is implemented on a SPARK1 SUN™ workstation. The differential equations are integration by a fourth-order Runge-Kutta routine with a time step of 0.001 h. Hydraulic parameters have been determined from the tracing experiments, the geometrical characteristics of the actual plant and its operation characteristics (flowrate, recycle rate, wastage rates, aeration).

4. RESULTS

The following figures illustrate an example of simulation obtained with the software. The primary settler has a volume of 4000 m^3 and an height of 2.5 m. The aeration tank has a volume of 3330 m^3. The well mixed-zone occupies 90% of each unit. The oxygen mass transfer coefficient is 4 hr^{-1} in each unit. The secondary settler has a volume of 5000 m^3, a height of 3.6 m. Each settler is represented by 5 layers of 5 annular zones. The upper layer is higher than the others. Figure 1 shows the flowrate and pollutions (soluble and insoluble) at the plant inlet. Figure 2 shows the efficiency of the primary settler. Figure 3 shows the residual pollution which is rejected to the river. The biomass concentration at the aeration tank is about 5 g/L when the total solid concentration in the recycle oscillates between 9 and 10 g/L. There is an oxygen profile across the aeration tank with a strong consumption in its first third.

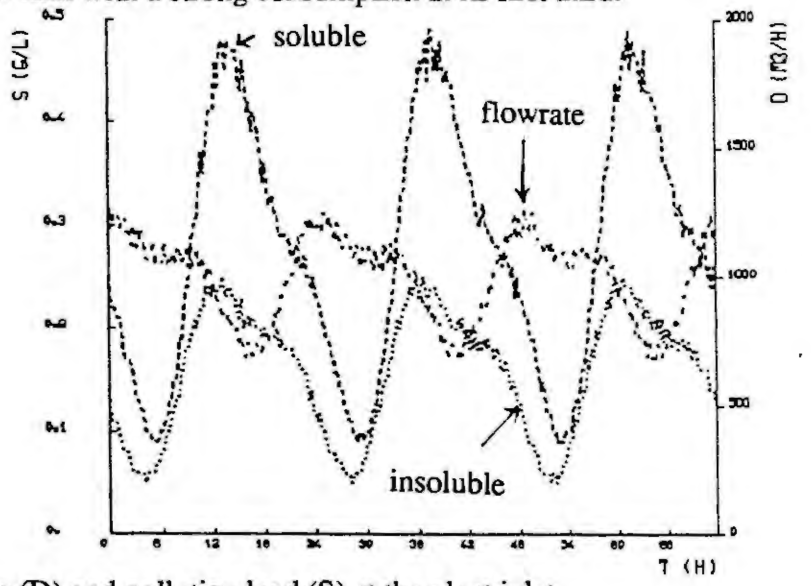


Figure 1: Flowrate (D) and pollution load (S) at the plant inlet

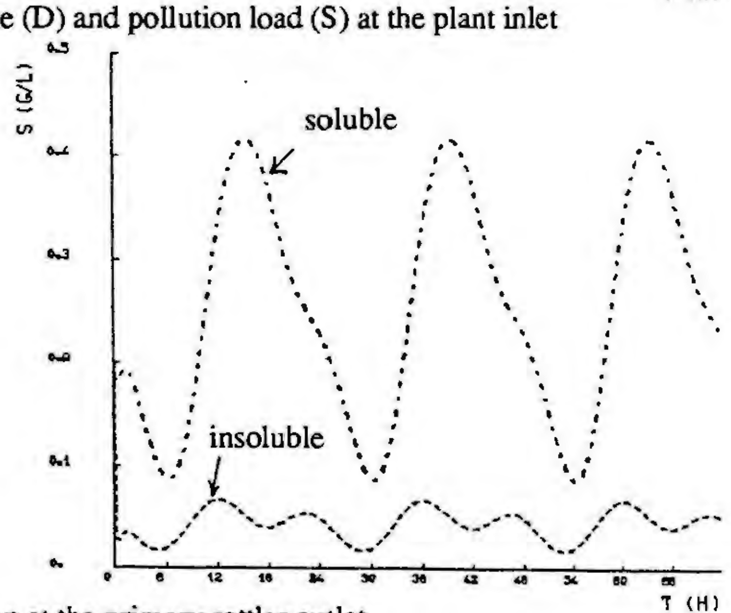


Figure 2: Pollution at the primary settler outlet

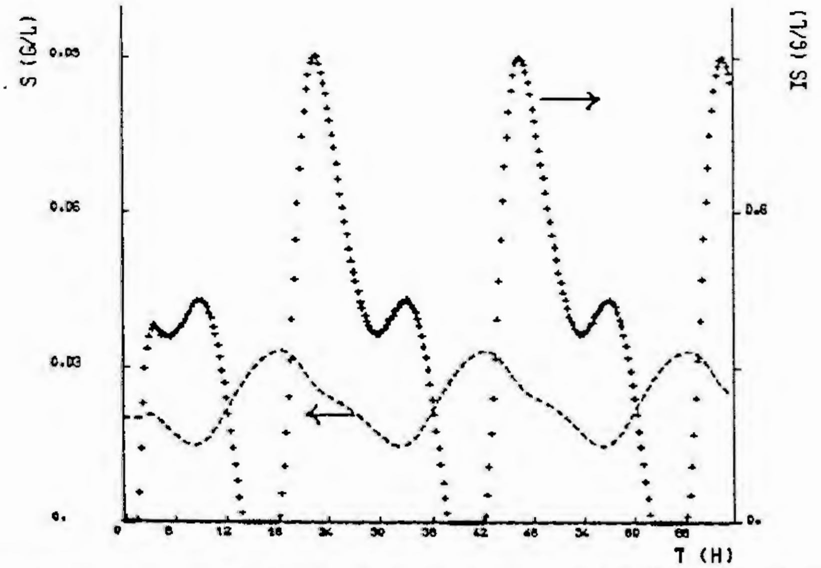


Figure 3: Residual pollution at the plant outlet: soluble pollution (S) and total solids (XS)

The software produces results which are in agreement with the experimental values obtained on the actual plant. Many improvements are still possible concerning for example sludge settleability and compression, nitrification and denitrification reactions, etc. These improvements require more experimental work on the plant and the advice of wastewater treatment experts.

5. ACKNOWLEDGEMENTS

The authors are thankful to the Nancy District Authority for its help in the project.

6. REFERENCES

[1] Dold, P.L. et G.v.R. Marais . Evaluation of the general activated sludge model proposed by the IAWPRC Task Group., *Wat. Sci. Tech.*, **18**, 63-89, (1986).

[2] Kabouris, J.C. et Georgakakos, A.P. Optimal control of the activated sludge process, *Wat. Res.*, **24**, 10, 1197-1208, (1990).

[3] Henze, M., C.P.L. Grady Jr., W. Gujer, G.v. R. Marias et T. Matsuo . Activated sludge model n° 1., IAWPRC Task Group Report, (1986)

[4] Neveux, S., M.N. Pons, M., Sardin, C. Prost et F. Colin . Etude quantitative de la dynamique de variation des effluents urbains lors de leur traitement. *Récents progrès en Génie des Procédés*, **3**, 9, 144-149, (1989).

[5] Nicol, J.P., L.D. Benefeld, E.D. Wetzel et J.A. Heidman . Activated sludge systems with biomass particle support structures. *Biotechnol. Bioeng.*, **31**, 682-695, (1988).

[6] Pons, M.N., O. Potier, N. Roche, F. Colin et C. Prost, Simulation of municipal wastewater treatment plants by activated sludge, ESCAPE 2, Toulouse, CACE 17-Supplement S227-S232 (1992).

[7] Roche, N. . Influence de l'hydrodynamique des bassins d'aération sur la décantabilité des boues activées. Thèse INPL, Nancy, France, (1989).

[8] Vesilind, P.A. . Discussion de "Evaluation of activated sludge thickening theories" R.I. Dick et B.B. Ewing. *J. sanit. Engng. Div., Am. Soc. civ. Engrs.*, **94**, 185-191, (1968)

SIMULATION OF THE MICROKINETICS OF BIOLOGICAL PHOSPHORUS REMOVAL

Angela Ante, Harald Voss

Department of Biochemical Engineering, Technical University Aachen, RWTH;
Worringer Weg 1; D - 52074 Aachen, FRG

Abstract. The simultaneous biological removal of nitrogen and phosphorus is a process of considerable technical and biological complexity owing to the mixed substrates and the diversity of microorganism populations. We have developed a mathematical model based on IAWPRC-model № 1 for this process scenario which describes the microkinetics of enhanced biological phosphorus elimination in addition to the degradation kinetics of carbonaceous components and nitrogen.

1. INTRODUCTION

A partial recycle of sludge is necessary for the phosphorus elimination and a partial recirculation of water for the denitrification, which is usually carried out upstream. So the observed conversion rates of some fundamental processes are mainly controlled by the kinetics of subsequent processes.Therefore and in consequence of fluctuating rates and concentrations in the influent it is very difficult to estimate the process kinetics without construction and dynamic solution of the adequate interconnected reactor balances. Fig. 1 shows the scheme of a wastewater treatment plant with nitrogen and phosphorus removal and the balancing units.

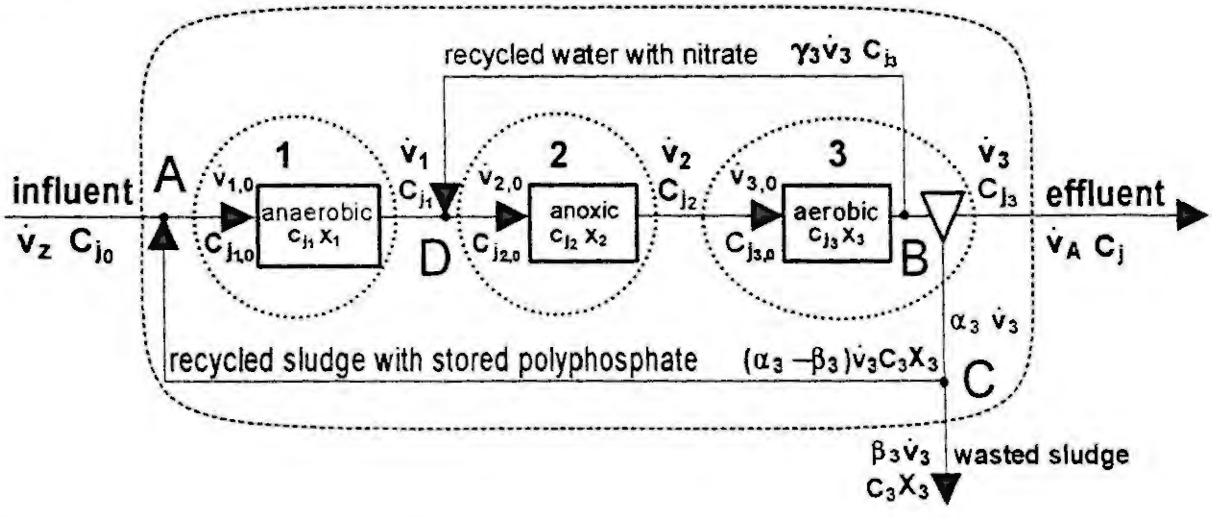


Fig 1: Scheme of a wastewater treatment plant with nitrogen and phosphorus removal and the balancing units; e.g. Phoredox design (anaer. = anaerobic)

For this general process situation a mathematical model has been developed which is based on IAWPRC model № 1 (Henze et al. 1987) describing the kinetics of the biological processes involved in accordance with the results of investigations.

2. THE MATHEMATICAL MODEL OF THE OVERALL PROCESS

The model allows to calculate the concentrations of soluble as well as particulate components including biomass fractions summarized in table 1.

Table 1: Definition of component symbols

No	symbol	definition	unit
Substrate components			
1	S_S	*Readily biodegradable substrate*	*- g (COD) m^{-3}*
2	S_{Ac}	*Low fatty acids*	*- g (COD) m^{-3}*
3	X_F	*Particulate substrate of lower molecular weight*	*- g (COD) m^{-3}*
4	X_S	*Slowly biodegradable substrate*	*- g (COD) m^{-3}*
Biomass fractions			
5	X_{BH}	*Active heterotrophic biomass*	*- g (COD) m^{-3}*
6	X_{BP}	*Active phosphorus-accumulating biomass*	*- g (COD) m^{-3}*
7	X_{BA}	*Active autotrophic biomass*	*- g (COD) m^{-3}*
Intracellular components of the phosphorus-accumulating microorganisms			
8	X_{PP}	*Intracellular stored polyphosphate phosphorus*	*- (P) m^{-3}*
9	X_{PB}	*Intracellular stored polyhydroxybutyrate (PHB)*	*- g (COD) m^{-3}*
Inert component			
10	X_X	*Inert organic matter*	*- g (COD) m^{-3}*
Oxygen			
11	S_O	*Oxygen*	*- g (neg. COD) m^{-3}*
Nutrient components			
12	S_P	*Soluble ortho-phosphate phosphorus*	*- g (P) m^{-3}*
13	S_{NO}	*Nitrate and nitrite nitrogen*	*- g (N) m^{-3}*
14	S_{NH}	*Ammonium and ammonia nitrogen*	*- g (N) m^{-3}*
15	X_{PD}	*Particulate biodegradable organic phosphorus*	*- g (P) m^{-3}*
16	X_{ND}	*Particulate biodegradable organic nitrogen*	*- g (N) m^{-3}*

The model in form of matrices of stoichiometry and kinetics of the overall phosphorus removal process are published by Ante et al. in /1/. These matrices allow to set up the appropriate mass balance equations for each component in each reactor of the plant and each actual reaction condition unit including the feed flows by selecting the relevant individual reaction mechanisms.

The following balance equations consider the component j in the first, anaerobic reactor (= balancing unit 1 of fig. 1) as examples of the use of the model.

The concentration of a soluble component in the influent is defined by the feed and recycle stream:

$$C_{j_{1,0}} = \frac{C_{j_0}\dot{v}_z + C_{j_3} \cdot (\alpha_3 - \beta_3)\dot{v}_3}{\dot{v}_z + (\alpha_3 - \beta_3)\dot{v}_3} \qquad [\mathrm{g/m^3}] \quad (1)$$

The volume flow rate of the influent is given by:

$$\dot{v}_{1,0} = \dot{v}_z + (\alpha_3 - \beta_3)\dot{v}_3 \qquad [\mathrm{m^3/d}] \quad (2)$$

The dynamic change of the concentration of soluble component j in the first reactor with volume V_1 is described by the convection transport expressions and the conversion rates:

$$\frac{dC_{j_1}}{dt} = \frac{C_{j_{1,0}} \cdot \dot{v}_{1,0}}{V_1} + \sum_j (v_{ij} \cdot r_i)_1 - \frac{C_{j_1} \cdot \dot{v}_1}{V_1} \qquad [\mathrm{g/(m^3 \cdot d)}] \quad (3)$$

For example, the observed conversion rate for phosphorus (component 12) under oxygen-free conditions is to be selected from the matrix for oxygen-free conditions. Summarizing the products of stoichiometric parameter v_{ij} and each process rate r_i allows to formulate the observed conversion rate:

$$\sum_j (v_{ij} \cdot r_i)_1 = v_{12,4} \cdot r_4 + v_{12,5} \cdot r_5 + v_{12,6} \cdot r_6 + v_{12,8} \cdot r_8 + v_{12,\,10b} \cdot r_{10b} + v_{12,12} \cdot r_{12} \qquad [\mathrm{g/(m^3 \cdot d)}] \quad (4)$$

The total phosphorus balance in the anaerobic re
actor is:

$$\frac{dS_P}{dt} = \frac{S_{P_{1,0}}}{V_1} - \frac{S_{P_1}\dot{v}_1}{V_1} - j_B \left(\frac{S_S}{K_S + S_S}\right)\left(\frac{S_{NO}}{K_{NO} + S_{NO}}\right)\left[\hat{\mu}_H \eta_D X_{BH} + \hat{\mu}_P \eta_P X_{BP}\right] - j_B \left(\frac{S_S}{K_S + S_S}\right)\left(\frac{K_{NO}}{K_{NO} + S_{NO}}\right) \hat{\mu}_G \eta_G X_{BH}$$

$$+ r_{17} \frac{X_{PD}}{X_S} + \rho_R \left(\frac{S_{Ac}}{K_{Ac} + S_{Ac}}\right)\left(\frac{K_{NO}}{K_{NO} + S_{NO}}\right)\left(\frac{X_{PP}}{X_{BP}}\right)\left(1 - \frac{X_{PB}}{X_{BP} T_{PB}}\right) X_{BP} + b_P X_{PP} \qquad [\mathrm{g/(m^3 \cdot d)}] \quad (5)$$

For particulate components X it is necessary to take into account the concentration in the recycle stream from the clarifier. For the influent concentration follows:

$$C_{j_{1,0}} = \frac{C_{j_0}\dot{v}_z + C_3 \cdot X_3 \cdot (\alpha_3 - \beta_3)\dot{v}_3}{\dot{v}_z + (\alpha_3 - \beta_3)\dot{v}_3} \qquad [g/m^3] \quad (6)$$

For example, the observed conversion rate for phosphorus accumulating microorganisms (component 6) under oxygen-free conditions is given by:

$$\sum_j (\nu_{ij} \cdot r_i)_1 = \nu_{6,5} \cdot r_5 + \nu_{6,10a} \cdot r_5 \qquad [g/(m^3 \cdot d)] \quad (7)$$

The balance for phosphorus accumulating microorganisms in the anaerobic reactor is:

$$\frac{dX_{BP}}{dt} = \frac{X_{BP_{1,0}}\dot{v}_1}{V_1} - \frac{X_{BP_1}\dot{v}_1}{V_1} + \hat{\mu}_H \cdot \left(\frac{S_S}{K_S + S_S}\right)\left(\frac{S_{NO}}{K_{NO} + S_{NO}}\right) \cdot X_{BP} - b_P X_{BP} \qquad [g/(m^3 \cdot d)] \quad (8)$$

Most of the values of the parameters were found in the literature. The values of the remaining parameters have been identified by modelling the steady state of a lab-scale plant. These values are published in tables 5 and 6 in /1/.

3. APPLICATION OF THE MATHEMATICAL MODEL

Simulations of the measured concentration curves from batch experiments carried out in our laboratory have been made with the identified parameters. The experiments delivered the microkinetics of the fermentation process and the production of low fatty acids. Before starting the experiments the centrifuged sludge has been saturated with acetate in order to fill up the organic storage and to empty the phosphorus storage of the phosphorus accumulating microorganisms so that it was possible to follow the time course and the kinetics of the production of organic acids by the facultativly anaerobic heterotrophs. The experimental conditions are summarized in table 2. The simulation fits well with the measured data as shown in fig. 2. Therefore it has been concluded that the model taken as basis is adequate and the identified parameters are relevant for the intended complete model of enhanced biological phosphorus elimination.

Table 2: Experiment conditions

Experiment conditions		Medium	[g/l]
Temperature	23°C	Glucose	2,5
Mixer velocity	150 min^{-1}	$(NH_4)_2SO_4$	0,104
PH	7,0	Na_2HPO_4	0,08
Correction medium	1 n NaOH	Resazurin	trace (redox indicator)

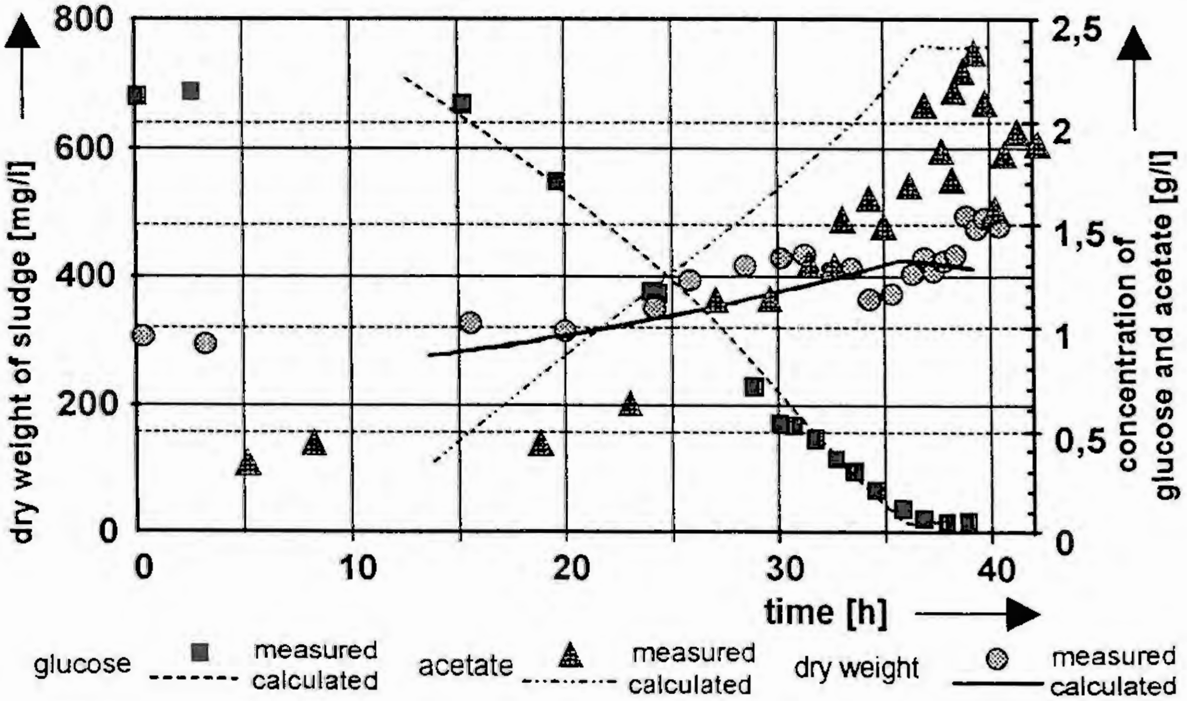


Fig. 2: Comparison of experiment and simulation

For this reason we used this mathematical model to design and optimize a treatment plant by simulating a modified configuration and by testing the possible influence of process parameters.

Fig. 3 shows the influence of a modified configuration calculated by the model. As an example the size of the second anaerobic reactor of the lab-scale treatment plant which we used for identification of the model parameter has been varied.

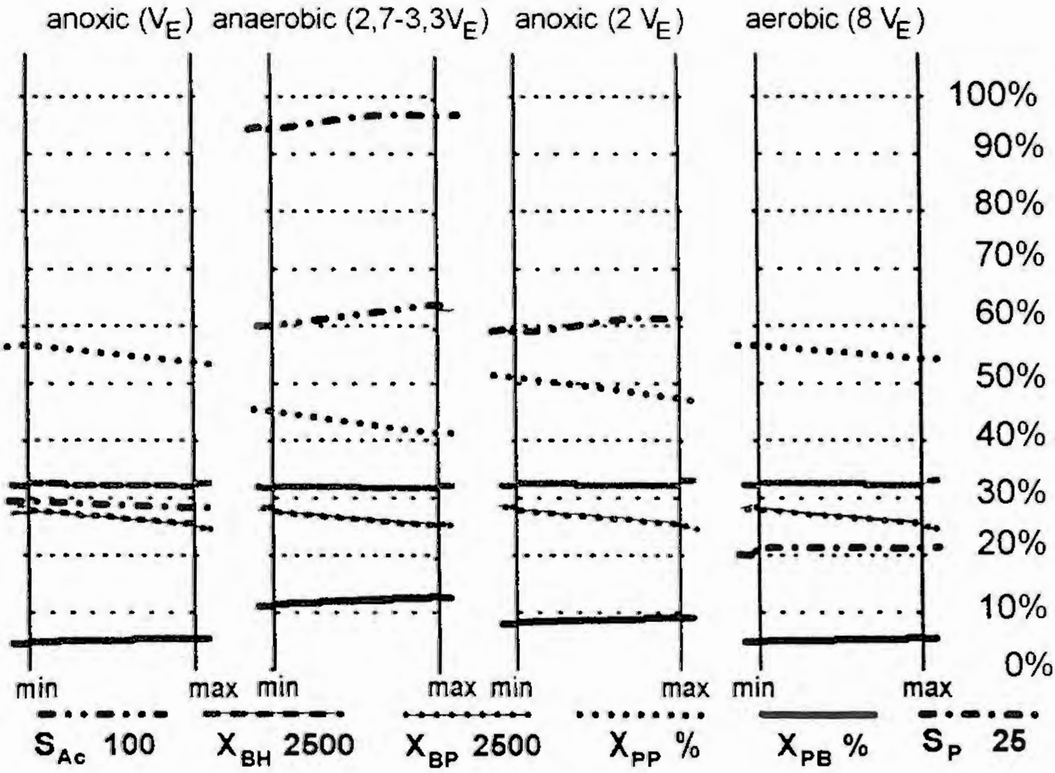


Fig. 3: Influence of the modification of the plant configuration by varying the size of the anaerobic reactor At the position "min" the iteration has been carried out with the minimum size, then the reactor volume has been continuously enlarged to "max", finally for some time steps the maximum size has been used (V_E = unit of volume); the different concentrations of the components corresponding to 100% of the y-achsis are indicated below the graph

At present the experimental investigations are continued in a lab scale unit plant of three combined bioreactors in order to obtain informations about the influence of the essential technical parameters e. g. mean cell residence time, water and sludge recycle rate, possible intermediate feeding and also about the influence of the process configuration such as number, design, size and interconnection of the reactors.

4. CONCLUSION

Efficient mathematical models are a convenient instrument for effective planning of biotechnological processes. The development and application of corresponding simulation programs allows to increase understanding of the interactions of technical parameters and the microbiological activities in the complex processes involved in enhanced biological phosphorus removal. Moreover, the presented semistructured model permits to minimize the total costs and energy and chemical demands by individual sizing of system units for a selected system configuration. Our mathematical process model shows an excellent agreement between experimental and simulated results even in the situation of incomplete informations about the details of the processes involved.

5. REFERENCES

/1/ Ante, A.; Besche, H. U.; Voss, H. (1993) Modellierung der Mikrokinetik der erhöhten biologischen Phosphorelimination. *Schriftenreihe "Biologische Abwasserreinigung 3"; Biologische Phosphorelimination aus Abwässern,* Kolloquium an der TU Berlin, 27./28. September

/2/ Henze, M.; Grady, C. P. L.; Gujer, W.; Marais, G.V.R.; Matsuo, T. - IAWPRC Task Group. (1987). Activated sludge model No. 1. *IAWPRC Scientific and Technical Report No. 1,* IAWPRC, London

SIMULATION OF INDUSTRIAL WASTEWATER TREATMENT PLANT

Milan Králik, Jan Derco, Jozef Antoni, Alexander Kovacs

Faculty of Chemical Technology, Slovak Technical University,
Radlinského 9, 812 37 Bratislava, Slovakia

Abstract

Results of an identification and a description of the computer program for a simulation of the Industrial Wastewater Treatment Plant (IWTP) behaviour are presented. The attention has been focused on the simultaneous removal of organic carbon, nitrates and ammonium impurities in a Carrousel system, [1]. The IAWPRC model, [2], was taken as a basis for the description of kinetics of biological processes. The complete-mixing reactor (CMR) with an aerobic and an anoxic zone, the CMR with the intermittent aeration and the complete-mixing tank in series model have proved to be a good approximation of the real system. Mathematical models together with the modules for an estimation of model parameters, were incorporated into the generally usable computer program which enables to simulate dynamic behaviour of the IWTP and thus, it gives a possibility to select the optimal regime of IWTP operation.

1. INTRODUCTION

Nowadays, a biological treatment of wastewaters is the most widely spread way for the removal of organic carbon, nitrates, ammonium nitrogen and phosphorus. While, for domestic wastewater, there are available relatively good descriptions of processes occurring in such systems, [3], the description of IWTP is not so simple, [4]. In the comparison with domestic wastewaters, industrial wastewaters have a different composition, the arrangement of equipment in IWTP is usually more sophisticated (e.g. some special chemical stages for treatment of a slowly biodegradable substrate) and also variations of inputs are more irregular than in domestic WTP. The aim of this paper is to bring a contribution to the solution of this problem by presenting some experiences from the investigation of the IWTP in east Slovakia - Chemko Strazske.

2. THE IWTP IN CHEMKO STRAZSKE

The IWTP in Chemko Strazske consists of mechanical, chemical and biological stages, whereas biological ones are represented by 2 conventional aeration tanks and by 2 Carrousel plants. One of the Carrousel has the volume of 32500 m^3, another one 16000 m^3. Hydraulic retention time in these systems depends on the flow rate and the way of Carrousel plants combination (parallel or in series); the average value is about 4 days. For the period, the data for a treatment were considered, the average flowrate of wastewater was 250 m^3/h, the input COD 4000 g m^{-3}, nitrogen in the form of NH_4^+ 200 g m^{-3} and nitrogen in the form nitrate 100 g m^{-3}. Wastewater in this enterprise usually contains a significant amount of slowly biodegradable and nondegradable compounds, such as hexamethylenetetramine. The chemical decomposition of this compound to ammonium and formaldehyde is performed prior entering the biological stages. The effectiveness of IWTP with respect to organic and nitrogen impurities is about 90 %. On the other hand, operational costs (the majority is given by the operation of aerators) have not been optimal.

3. MATHEMATICAL MODELS

One of the ways how to optimize a regime of WTP is by a utilization of dynamic mathematical models (simulators) which can help to find the best operational conditions, i.e the sufficient efficiency of WTP under minimal costs. The black box (also called statistical or empirical) and/or physical-chemical (also called mechanistic) models of individual apparatus are involved into a simulator, [4]. The physical-chemical models are based on conservation equations of species, equations for chemical and biochemical reaction rates, and mass transfer rate equations. Black box models are based on the approach considering only a dependence of outputs on inputs. Physical-chemical models are more suitable for an optimization of WTP, but usually they are more complex than the black box approach models (higher computational time) and an estimation of model parameters is more difficult (combination of experiments in the laboratory and on the plant, more sophisticated procedure of the data treatment).

In order to derive the physical-chemical model of the equipment, a knowledge of the fluid flow pattern is needful. The approximation by a complete-mixing vessel is used in the majority of available simulators (e.g. IAWPRC, SSSP, ASIM, UCTOLD, [3]). The description of biochemical reaction kinetics and stoichiometry is usually based on the IAWPRC model, [2].

The differences among the simulators mentioned above are also in the possibility how to connect individual units of the WTP, distribution of the feed, a possibility to consider kinetic parameters as temperature dependent, time range for the performance of a dynamic simulation. For example, all these simulators use the connection of units in series. In the standard version of simulators, the number of apparatus is limited (SSSP: 10, ASIM: 6, IAWPRC:12, UCTOLD: 12). The temperature dependence of kinetic parameters is by the Arrhenius equation (UCTOLD, IAWPRC) or through the parameter values for some temperatures (ASIM). The 24 hours cycle is used as a time range for the simulation with SSSP, UCTOLD and IAWPRC programs, the ASIM has an optional length of the simulation period. The other limitation of these simulators is that they do not have standard modules for an estimation of model parameters. In spite of the restrictions mentioned earlier, an application of these simulators is very useful. The simulators work in the "user friendly" mode and they are suitable for a deeper acquaintanance with processes in WTP, and can be also utilized for the simulation of individual units of the IWTP with more complex technological scheme.

Based on a description given in the book of Andrews, [4], the GPS (Generally Proposed Simulator) could be very suitable for application in large IWTP control systems. Unfortunately, we have not had an opportunity to test this simulator.

4. DYNAMIC SIMULATOR OF WASTEWATER TREATMENT PLANT - DSWTP

The limitations of the application of the simulators tested in our laboratory have led us to create our own simulator - DSWTP. The development of this simulator started with the models we have used for an identification of the IWTP equipment. Our attention has been focused mainly on the Carrousell system because of the highest influence on the quality of the outflow wastewater. Generally, the best model of the Carrousel system would be the axial dispersion flow reactor with the axial dispersion coefficient dependent on the axial coordinate with respect to aerators under operation, e.g. the value of the axial dispersion coefficient is very high in the space where an aerator is switched-on and this value is much lower in some distance (more than 10 m) from this aerator. However, such model is of the high complexity and it is not easy to adapt model parameters to the Carrousel plant. Another problem deals with a suitable algorithm for an efficient solution of partial differential equations with space and time variable parameters. Therefore, we have used much simpler approximation of the Carrousel system than the axial dispersion reactor.

Complete-mixing aerobic and anoxic reactors-in-series were used as the first approximation of the Carrousel system. We modified the IAWPRC biokinetic model in such a way that switching terms dependent on a concentration of oxygen were set to zero or to one according to aerobic or anoxic conditions. This procedure has resulted in one set of biokinetic rate equations for oxic conditions and another one for anoxic conditions. The number and positions of the aerators under operation were taken into account by the number and position of the reactors with the oxic regime. The treatment of the one year data from the IWTP in Chemko Strazske showed a very good ability of this model for the dynamic simulation, [5].

The internal recirculation ratio in the Carrousel is very high (more than 40) and in consequence, concentrations of all species except oxygen are in each point of the Carrousel system practically the same, so the hydraulic regime is near to the CMR. This approximation was utilized in our second model. The extents of aerobic and anoxic processes were considered by the weights on oxic and anaerobic biochemical reactions. These weights have been set as a function of the number of aerators switched-on. The identification of IWTP confirmed validity of this model, [6]. Also, a significant reduction of the computational time due to the less number of differential equations, was observed in the comparison with an application of the tank-in-series model.

Our third approach to modelling of the Carrousel system is based on a transformation of the coordinate along the direction of the sludge flow (axial coordinate), to the time coordinate. This transformation leads to a reactor with an intermittent supply of reactants - in our case: the supply of oxygen (similar approach e.g [7]). The main advantage of this model in the comparison to the two models mentioned above is that a transport of oxygen from gas to liquid phase can be included into the dynamic description. The intermittent CMR model is of a comparable complexity as the simple CMR, computational requirements are low and an incorporation into a simulator is easy. On the other hand, this model is much simpler and less computational time is necessary than in the case of the tank-in-series model. The other advantage is a possibility to consider the mass transfer coefficient as a function of the organic substrate concentration which can significantly improve the description of the real system behaviour, [8].

The complete-mixing vessels are applied in DSWTP to approximate a behaviour of all other units involved in IWTP (homogenization tanks, chemical reactors, settlers). If a chemical reaction is performed the total conversion of species which are present in less stoichiometric amount, is considered. As for settlers, one can consider more sophisticated approach to describe dynamics, e.g. the model proposed by Marsili-Libelli [9]. However, in the case of Carrousel systems, hydraulic retention time in the bioreactor is much more higher than in the settler (approx. 10:1) so, the inacuracy of the approximation by a complete- mixing vessel is not significant.

The other features of the DSWTP are as follows:

i) the number of units (models) is limited only by the capacity of a computer applied (but if the total number of ordinary differential equations is higher than 500 some problems with the numerics can occur)

ii) the connection of the units in a IWTP is arbitrary, inputs and otputs can be directed to and from each unit

iii) time variations of wastewater flow and concentrations can be considered, the output flows from the units are controlled by distribution coefficients, so the time variations of the feed flow rate are automatically reflected

iv) the volume of liquid in homogenization tanks can vary with time, i.e. the output flow rate from a homogenization tank need not to be equal to the sum of input flow rates; the volume is checked to minimal and maximal values

v) during the simulation period, the concentrations of components in the output streams are checked by the prescribed limits

vi) the simulation time and frequency of the tabulation of results is limited only by the capacity of a computer.

DSWTP creates and modifies global model in a dialog with a user. The Runge-Kutta-Hanna method, [10], is applied for a solution of the set of ordinary differential equations. DSWTP operates above the database updated by programs for a collection of data from the IWTP and programs for an estimation of model parameters used in a simulation. Values of model parameters are calculated on the basis of IWTP dynamic data and results from laboratory kinetic tests, [11], which are used as final value of the model parameters or as the starting values in an optimization procedure. The least square objective function and Nelder Mead method for a search of a minimum, [12], are utilized to find model parameters.

5. RESULTS

DSWTP has been developed for the PC 286 computers and their analogues. The experience with the first version of our simulator showed the utility and the potential of this program in the improvement of the IWTP economics.

6. REFERENCES

[1] Glysson, E.,A., Swan, D.,E., Way, E.,J., Inovation in the Water and Wastewater Fields. Buterworth Publishers, Toronto 1985

[2] IAWPRC, Scientific and Technical Reports No.1 Activated Sludge Model No.1 by IAWPRC Task Group on Mathematical Modelling for Design and Operation of Biological Wastewater Treatment, 1987

[3] Henze, M., Nutrient Removal Computer Models. Nutrient Removal Newsletter, 2/92 (1992) 5

[4] Andrews, J.,F., Dynamics and Control of the Activated Sludge Process. Technomic Publishing Company, Lancaster, 1992

[5] Hutňan, M., Derco, J., Králik, M., Modelling of Carrousel Extended Aeration Plant. In: Proceeding of 6th IAWPRC Conference on Design and Operation of Large Wastewater Treatment Plants, Prague (1991), 400-405

[6] Derco, J., Králik M., Hutňan, M., Drtil, M., Modelling of Wastewater Treatment Plant. In: J. Arnaldos & P. Mutje (Ed.), Chemical Industry and Environment, Vol.2: Water, Girona-Spain, 1993

[7] Baykal, B.,B., Orhon, D., Artan, N., Implications of the Task Group Model-II. Response to Intermittent Loadings. Wat. Res. 24 (1990), 1259-1268

[8] Derco, J., Králik, M., Černák, R., Berešíková, Z., Modelling of Carrousel Plant by Intermittent Aeration Activated Sludge Process. In: Proceedings of the ChISA'93, Prague, 1993

[9] Marsili-Libelli, S., Modelling, Identification and Control of the Activated Sludge Process. In: A.Fiechter (Ed.), Advances in Biochemical Engineering/Biotechnology, Vol.38, Springer Verlag Berlin Heidelberg, 1989

[10] Kubíček, M., Numerical Algoritms of Chemical Engineering Problems Solution. Publishers of Technical Literature, Prague (1993) (czech)

[11] Bodík, I., Drtil, M., Derco J., Estimation of the Kinetic Parameters of Heterotrophs and Autotrophs. In: V.Bales (Ed.), Modelling for Improved Bioreactor Performance, Papiernička-Slovakia, 1993

[12] Bounday, D., B., Basic Optimization Methods. Edward Arnold Publ., London, 1984

SIMULATION AND CONTROL OF THE ACTIVATED SLUDGE PROCESS – A COMPARISON OF MODEL COMPLEXITY

U. JEPPSSON

Department of Industrial Electrical Engineering and Automation (IEA)
Lund Institute of Technology (LTH)
P.O. Box 118, S-221 00 Lund, Sweden
E-mail: Ulf.Jeppsson@indeltek.lth.se

Abstract. Automatic control systems relies on modelling to predict the near future and identification algorithms to adapt to changing process behaviour. The traditionally highly complex models of the activated sludge process developed for scientific purposes, cannot be identified from on-line measurements and are not suited for process control purposes in their present form. Model decoupling based on the very different time scales of the dynamic processes is one possible way of attacking this problem. It allows the implementation of more simple and realistic controllers in combination with predictions based on simplified models in a hierarchical control structure. This paper discusses these concepts and presents a reduced order model describing carbonaceous removal, nitrification, and denitrification in a medium time scale (several hours/days). The model parameters are identifiable from available on-line measurements and the dynamic behaviour is verified against computer simulations of the IAWQ model no. 1.

Key words – wastewater treatment; activated sludge; dynamic modelling; model complexity; model reduction; model identification; automatic control.

1. INTRODUCTION

A wastewater treatment (WWT) process is a highly complex system characterized by large, uncontrollable input disturbances. With a limited number of control possibilities and available on-line measurements, the main objective of the operation is to maintain a consistently high quality process output during transient process behaviour. The obvious problem stress the need for better understanding of the dynamics, operation, and control of WWT systems.

WWT plants are practically never in steady state, mainly due to load variations. As a consequence the dynamical properties of the process have to be fairly well known if a plant is going to be consistently controlled towards a desired result. Since the knowledge of the plant dynamics can be incorporated in mathematical models, it is crucial to develop models that are accurate and suitable for their intended purpose. The most sophisticated models available today, describing the structured behaviour of activated sludge plants in low level scientific terms, e.g. [1, 3, 13], are far too complex to be used for operational purposes.

The high complexity of such models causes severe problems of identifiability and verifiability [2, 4]. Usually the models are derived from simpler unit operations and later combined to large plant models. Although the qualitative behaviour is most probably the same, the identification and verification become an awkward task. A number of parameter combinations can often explain the same dynamical behaviour. This is further accentuated when the influent wastewater composition is taken into consideration; a change in its characteristics can frequently be explained by kinetic parameter changes. An adequate model for predicting and controlling WWT plant behaviour should be verifiable and all model parameters possible to update in a unique way from on-line measurements. Such simple, dynamical models are the aim for this work.

Advanced control systems of WWT plants are still in the development stage but recent advancements in technology, e.g. sensors and computers, have been rapid. Therefore automatic control of WWT systems are now becoming feasible. Economical and environmental incentives are also strong driving forces for improving plant performance by means of control. From the standpoint of process control, a WWT process has the following characteristics. Firstly, all wastewater flowing into the plant must be treated and the consistency of the operation must be guaranteed. Secondly, the quantity and quality of the wastewater inflow cannot be adjusted. Moreover, WWT processes are subject to large disturbances and process parameters change over time due to the adaptive behaviour of the microorganisms. The residence time for these processes is long and the treatment systems suffer from long lag times and sudden, abrupt failures. In addition, there are a number of internal feedback loops, such as return flows from sludge treatment processes. Under these circumstances the process is hard to control efficiently, especially as another important control objective is to minimize operational costs, which may lead to contradictory performance criteria.

With an increasing number of subprocesses (e.g. simultaneous biological carbon, nitrogen, and phosphorus removal) within a treatment plant, the complexity of the control task will increase further in order to maintain consistently good operations. It is possible to take advantage of the fact that the dynamics of an activated sludge system spans over a wide spectrum of response times – from seconds to months. This allows many control actions to be decoupled, i.e. several variables may be controlled by separate, local controllers by assuming fast dynamics to be instantaneous while controlling slow phenomena and controlling fast varying variables while considering the slow modes to be constant. Consequently it is possible to apply simplified dynamic models with different predictive time horizons. Therefore it is believed that control systems for the activated sludge process should be based on a hierarchical structure. An automatic control system may be developed on at least three different levels with respect to different time variations of the process.

Finally, the necessary inclusion of the clarification and thickening processes will complicate the control issue, although a few structured models of the settler behaviour do exist, e.g. [10, 12]. This is mainly due to the poor knowledge of the interactions between the settling parameters and the biological conditions of the sludge.

2. MODEL REDUCTION

The activated sludge process is recognized as the most common and major unit process for reduction of organic waste. The main feature of the process includes degradation of influent biodegradable pollutants, containing both organic carbon, nitrogen, and phosphorus by use of microorganisms. The organisms form flocs which are separated from the treated wastewater by means of gravity settling and recirculated back to the reactors. Basically, organic material is transformed into water and carbondioxide by heterotrophic organisms while consuming oxygen (aerobic conditions) or nitrate (anoxic conditions). Nitrogen – in the form of ammonia – is transformed into nitrate and water by autotrophic organisms while utilizing oxygen (aerobic conditions). However, the carbonaceous and nitrogenous material are present in many different forms, which the microorganisms react differently to and they are also transformed between forms through hydrolysis. Moreover, a large number of different organism species exist within a WWT plant and the dominating population may change over time. Consequently the behaviour of a plant varies. The process is also affected by the oxygen requirements, hydraulic flow schemes, the behaviour of the settler and clarifier, etc., as well as secondary parameters, such as temperature, pH, and toxic or inhibitory substances.

Learning from other types of complex processes it should be clear that *one* single model of the activated sludge process will never be sufficient to describe it fully. Instead, every model must be adjusted for its intended goal of application, whether it is a model for design, research, or control. The purpose affects the complexity, structure, and predictive capabilities of a model.

Improved biological insights have led to increasingly complex, structured compartment models to describe the bioprocesses and summarize the level of knowledge. They are usually structured in the sense that the activity of the biomass as well as the mixture of substrates are described by several variables. This kind of structured modelling is the ultimate way of moving from a data description towards a process description. If possible, the use of such structured models should be preferred over unstructured ones because of their higher predictive value. However, it is extremely important not to mistake high model *complexity* for model *accuracy*. A model of sufficiently high order may provide a perfect fit to available data but have no real predictive power.

The majority of the available models describing the activated sludge process rely heavily on off-line analysis for identification. Consequently, these models cannot be used to devise practically implementable automatic control systems. In contrast, such models can still be used in simulations to gain insight on the principle behaviour of WWT plants under various conditions and to investigate the potential of different control strategies.

The difficulty of applying highly complex models may be exemplified by the state-of-the-art NDBEPR model [13]. It incorporates most of the current knowledge for the activated sludge process gained from extensive interpretation of experimental data. Consisting of no less than 19 state variables, 25 reaction processes, and 51 parameters it is obvious that available monitoring equipment is completely insufficient to support any control system based on this model. On the other hand, as a research model it is an extraordinary result of profound process knowledge, biological insight, hard work, and possibly even intuition.

The traditional 'bottom-up' method for research modelling – i.e. identifying each single reaction mechanism under extremely well controlled conditions and later putting them together to form a complete model – is not the way to proceed for the goals aimed at in this work. Instead a 'top-down' method is suggested, i.e. starting with an extremely simplified model where many state variables and parameters have been lumped together and structural difficulties have been eliminated, and step by step increase the complexity. A choice of a proper structure is based on criteria that include complexity, model fit (dynamic behaviour), identifiability, available on-line measurements, etc. It is also possible to apply general methods for model reduction although the author believes it important to make use of physical insight of the process behaviour. This means that major phenomena should be described in a physically reasonable manner and parameters have some physical interpretation.

In this paper, the IAWQ model no. 1 [3] is used as the basis for the reduction procedure. The aim of the reduced model is to adequately represent both carbonaceous and nitrogenous activities with sufficient accuracy for the purpose of control. Some problems which exist with the IAWQ model, are focused on in the development of the reduced model, e.g.:

- lack of identifiability;
- troublesome nonlinearities, e.g. the Monod and the switching functions;
- difficult estimation and updating of time varying parameters;
- complicated characterisation of the incoming wastewater;
- mix of slow and fast dynamics;
- limited usefulness for automatic control.

The result of the reduction procedure is summarized in Figure 1. This reduced model describes the biological behaviour of an activated sludge process in a medium time scale (days). Therefore it does not take rapid uptake phenomena into consideration (only one type of modelled carbonaceous substrate and consequently no hydrolysis is incorporated) and dissolved oxygen (DO) is excluded as a state variable. DO is assumed to be controlled separately in a faster time scale where these measurements may be used as the basis for estimation of both the respiration rate and the short-term BOD by means of a respirometer [8, 9, 11].

Only three quality variables are assumed to be measurable on-line for keeping the model parameters updated – biodegradable organic substrate, ammonia nitrogen, nitrate nitrogen – on top of respiration measurements. Moreover, the number of parameters are greatly reduced, the switching functions are eliminated, and the Monod model is simplified by a first order

Reduced model		Anoxic environment						Aerobic environment					
Component ➤ i		1	2	3	4	5	Process rate	1	2	3	4	5	Process rate
j	Process ↓	X_{COD}	S_{NH}	S_{NO}	$X_{B,H}$	$X_{B,A}$	ρ_j [ML^{-3}T^{-1}]	X_{COD}	S_{NH}	S_{NO}	$X_{B,H}$	$X_{B,A}$	ρ_j [ML^{-3}T^{-1}]
1	Growth of heterotrophs	$-\frac{1}{Y_H}$	$-i_{XB}$	$-\frac{1-Y_H}{2.86\,Y_H}$	1		$r_H\,X_{COD}\,X_{B,H}$	$-\frac{1}{Y_H}$	$-i_{XB}$		1		$r_H\,X_{COD}\,X_{B,H}$
2	Growth of autotrophs								$-i_{XB}-\frac{1}{Y_A}$	$\frac{1}{Y_A}$		1	$r_A\,S_{NH}\,X_{B,A}$
3	Decay of heterotrophs	1	i_{XB}		-1		$b\,X_{B,H}$	1	i_{XB}		-1		$b\,X_{B,H}$
4	Decay of autotrophs	1	i_{XB}			-1	$b\,X_{B,A}$	1	i_{XB}			-1	$bX_{B,A}$
Conversion rates [ML^{-3}T^{-1}]		$cr_i = \Sigma \nu_{ij}\,\rho_j$											
Stochiometric parameters Heterotrophic yield: Y_H Autotrophic yield: Y_A M(N)/M(COD) in biomass: i_{XB}		Biodegradable organic matter [M(COD)L^{-3}]	Ammonia nitrogen [M(N)L^{-3}]	Nitrate nitrogen [M(N)L^{-3}]	Active heterotrophic biomass [M(COD)L^{-3}]	Activa autotrophic biomass [M(COD)L^{-3}]	Parameters to estimate Anoxic: r_H, Y_H Aerobic: r_H, r_A, Y_H, Y_A Common: b	Biodegradable organic matter [M(COD)L^{-3}]	Ammonia nitrogen [M(N)L^{-3}]	Nitrate nitrogen [M(N)L^{-3}]	Active heterotrophic biomass [M(COD)L^{-3}]	Active autotrophic biomass [M(COD)L^{-3}]	Kinetic parameters Heterotrophic reaction rate: r_H Autotrophic reaction rate: r_A Heterotrophic and autotrophic decay rate: b

Figure 1. Matrix representation of a reduced order model for daily variations [4].

approximation. Of the eight basic processes in the IAWQ model only four remain. Parameters shown in bold in the process matrix (seven for the complete system) are assumed variable and should consequently be identified and estimated on-line. Only the parameter i_{XB} is considered known and constant due to the low sensitivity of the model to this parameter. An analysis of the simplifying assumptions, motives, and consequences is given in [4, 5].

3. IDENTIFICATION AND CONTROL

The issues of identification and control are very closely linked for the activated sludge process since the behaviour (i.e. process parameters) change over time. A model based process control scheme for a WWT plant must be *adaptive, predictive*, and *efficient*. Adaptiveness means that the model is capable of adjusting itself to changes both in the input and to variations of the parameters that characterize the process in order to minimize or eliminate any instability or other undesired behaviour. Predictiveness implies the capability of forecasting and anticipating changes before they occur and to use this information to eliminate process disturbances. Finally, efficiency means running the process to its full capacity and take advantage of its whole potential according to the applied control objectives.

The overall goals of WWT plant operations can be summarized as:

- keep the plant running and avoid large failures;
- comply with the effluent quality requirements;
- minimize the cost (energy, chemicals, sludge production, etc.).

Traditionally, the first two goals have been accomplished by means of oversizing WWT plants which made them practically 'self-controlled'. No incentive was given for operation and control. Today, many plants are approaching their maximum capacity in this sense due to increased inflows, stricter effluent regulations, etc., and are considered for expansion. However, a conversion of operation to a dynamic control scheme with no or minimal structural modification will most likely be an economical and more efficient alternative to a program of massive structural intervention. Such an approach would also provide means for reducing operational costs (the third objective) as well as preparing plants for the increased process complexity expected in the future, which will rely heavily on control. Plants must, however, be modified to include a larger degree of flexibility, e.g. step-feed (for influent and recirculated streams), variable volumes and air distribution, flow equalisation, thereby strengthening the control authority of the process. Moreover, a future WWT plant will not be viewed as a separate process, instead the sewer system, the plant, and the recipient will be interconnected – not only in reality but also by means of control strategies and actions.

Process control based on predictions from models is subjected to uncertainty. However, unlike design, it is possible in many cases to correct for this uncertainty by the use of automatic feedback control in which the amount of control exerted depends upon the difference between the desired and observed performance. This means that a reasonable amount of uncertainty can be tolerated in a dynamic model for process control, and reduced order models with considerable error in the prediction may thus prove useful.

The earlier discussed widely different time scales for various dynamics of the activated sludge process significantly improve the possibilities for successful control by means of process decoupling. Thereby a high order multivariable control system where all measurements and control variables are connected to one very complex controller can be avoided. Instead many control actions can be based on a single measurement and a single control signal. If a control system cannot perform well enough on measured information by direct feedback of observed variables, then one can seek to incorporate a model of the system which is being controlled. Such modularization and hierarchisation are highly effective ways of managing process complexity. For example can the following processes be identified with respect to different time constants under normal plant operation: respiration rate changes, hydraulic phenomena caused by flow rate changes, dissolved oxygen dynamics – minutes; concentration changes caused by flow rate variations – hours; cell growth, cell decay – days/weeks; temperature changes – months. Naturally, the time scale of the model must also be related to the time scale in which the controlled variable can influence the process.

A computer control system has several tasks, such as disturbance detection, measurement noise filtering, calculation of indirect variables, and calculation of control strategies. By combining it with methods for real time state and parameter estimation (e.g. an extended Kalman filter) the modelled process behaviour can be updated and thereby the predictive capabilities are enhanced. This principle *requires* fairly simple, dynamical models which are identifiable from on-line measurements. The structure of such a system is outlined in Fig. 2.

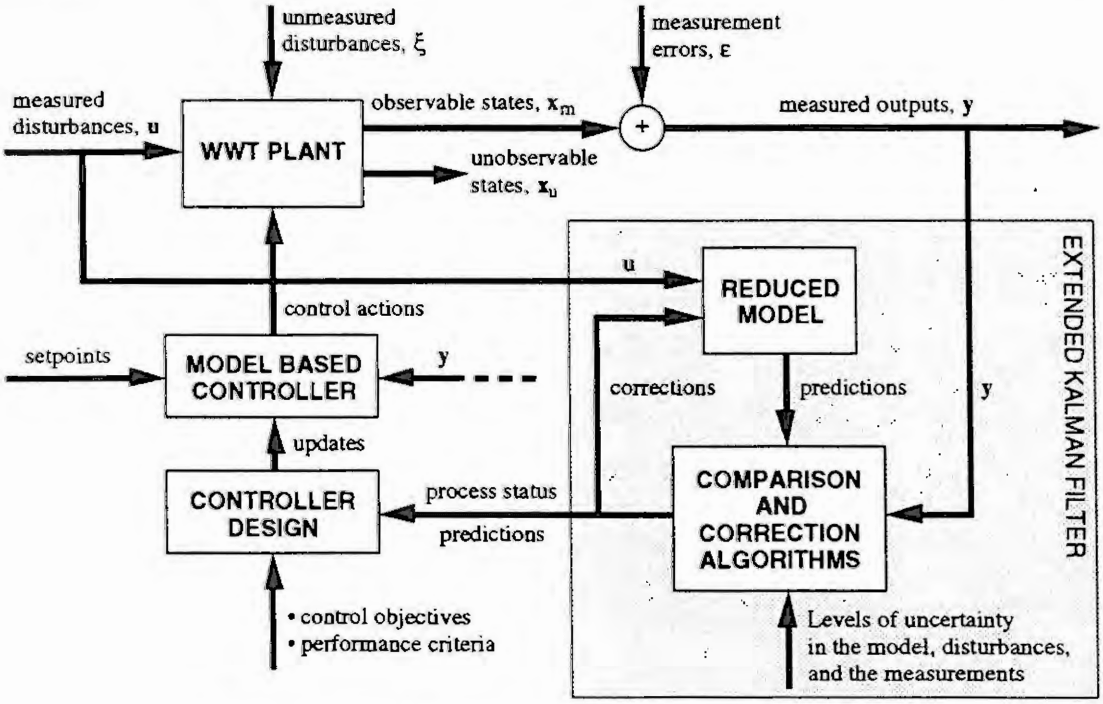

unmeasured disturbances, ξ
measurement errors, ε
measured disturbances, $\mathbf{u}$
WWT PLANT
observable states, $\mathbf{x}_m$
+
measured outputs, $\mathbf{y}$
unobservable states, $\mathbf{x}_u$
EXTENDED KALMAN FILTER
u
REDUCED MODEL
control actions
setpoints
MODEL BASED CONTROLLER
y
corrections
predictions
y
updates
CONTROLLER DESIGN
process status
predictions
COMPARISON AND CORRECTION ALGORITHMS
• control objectives
• performance criteria
Levels of uncertainty in the model, disturbances, and the measurements

Figure 2. Structured on-line identification and model based control.

4. MODEL VERIFICATION BY SIMULATION

Verification is an important aspect of all modelling work. It is not only a question of fitting a model to a set of data but also of identifiability, stability, sensitivity, complexity, etc. For on-line systems the verification procedure is even more critical since such systems directly interact with a real process. For many processes (such as WWT) the models must incorporate methods to update the parameters in a unique fashion as operating conditions change over time. The ultimate verification can only be accomplished by applying a model in practice, monitoring the results, and determine if the purpose of the model has been accomplished.

The reduced order model (Fig. 1) is verified against the IAWQ model to investigate if the same *dynamical* behaviour of importance in the actual time scale is incorporated. The IAWQ model with default parameter values for 20° C is used to simulate a completely mixed, single sludge WWT plant performing both carbonaceous and nitrogenous removal with an anoxic zone for the predenitrification. The hydraulic retention time is ten hours, the sludge retention time is ten days, and both internal and external recirculations exist. More details of the plant conditions are described in [4]. The parameters of the reduced order model are first identified under perturbed steady state conditions from the earlier discussed basic quality measurements, using an extended Kalman filter. The physical outline of the plant is naturally identical for both models. The reduced model is then simulated using the obtained parameter values and directly compared to simulations of the IAWQ model, exposed to the same influent characteristics. A more thorough analysis of the identification methods and the verification results is given in [4].

For both models, the thickener is modelled as a constant compaction ratio [6]. This ratio is in turn computed from the sludge retention time. All COD is considered part of the floc in the thickener. The settler retention time is taken into account by a subsequent time lag. The clarifier behaviour is excluded from this part of the model verification.

In this brief comparison between the two models, pulse disturbances (50% increase) of the influent flow rate, biodegradable organic substrate concentration (the sum of readily and slowly biodegradable substrate for the IAWQ model), and ammonia concentration are considered. Since the reduced model has a different behaviour for fast transients – not the modelled time scale – the pulse transients appear to be the most decisive model test for the dynamics. Some results are presented in Figure 3.

It is clear that the qualitative behaviour of the models are quite similar. Some differences are apparent and can be accounted for [4]. The predicted concentrations of active biomass suffer from significant offsets compared to the results from the IAWQ model. This is because the parameter identification was not based on any measurements of these quantities. On the other hand, active biomass concentration is extremely difficult to determine accurately in a WWT process and consequently the values predicted by the IAWQ model may also be uncertain. Important to realize is the fact that for control purposes it is not always necessary to know the correct absolute value of certain variables but more often their relative change. Control strategies can also be based on models which can only predict the qualitative behaviour of a process. A compromise must be made between a model sufficiently complex to describe the major phenomena and a model simple enough to allow its parameters to be updated from on-line measurements.

5. CONCLUSIONS

From a control point of view, the activated sludge process may be characterized as a *disturbance rejection problem.* The objective is to produce a consistently high quality output while exposed to large input and internal process disturbances. This may be accomplished with automatic control systems which are both predictive and adaptive. The high complexity of the process creates a demand for modularisation and hierarchisation which can be based on the very different time scales of the process dynamics. Such decoupling enhances the possibility of applying more simple and realistic controllers and predictions based on reduced order models. In an effectively functioning hierarchy, the interactions between systems at lower levels are such as to create a reduced level of complexity at the level perceived above, i.e. the externally perceived complexity is reduced.

The problem of overparametrization in complex models has been approached. Existing complex models of the activated sludge system dynamics do not have a unique set of parameters which can explain a certain behaviour. An attempt has been made to derive a reduced order model with less number of states and parameters that still preserves the main physical interpretations. The parameters may be uniquely updated from available on-line measurements. Simulation comparisons between the IAWQ model and the reduced model indicate that the main features of the dynamics have been retained.

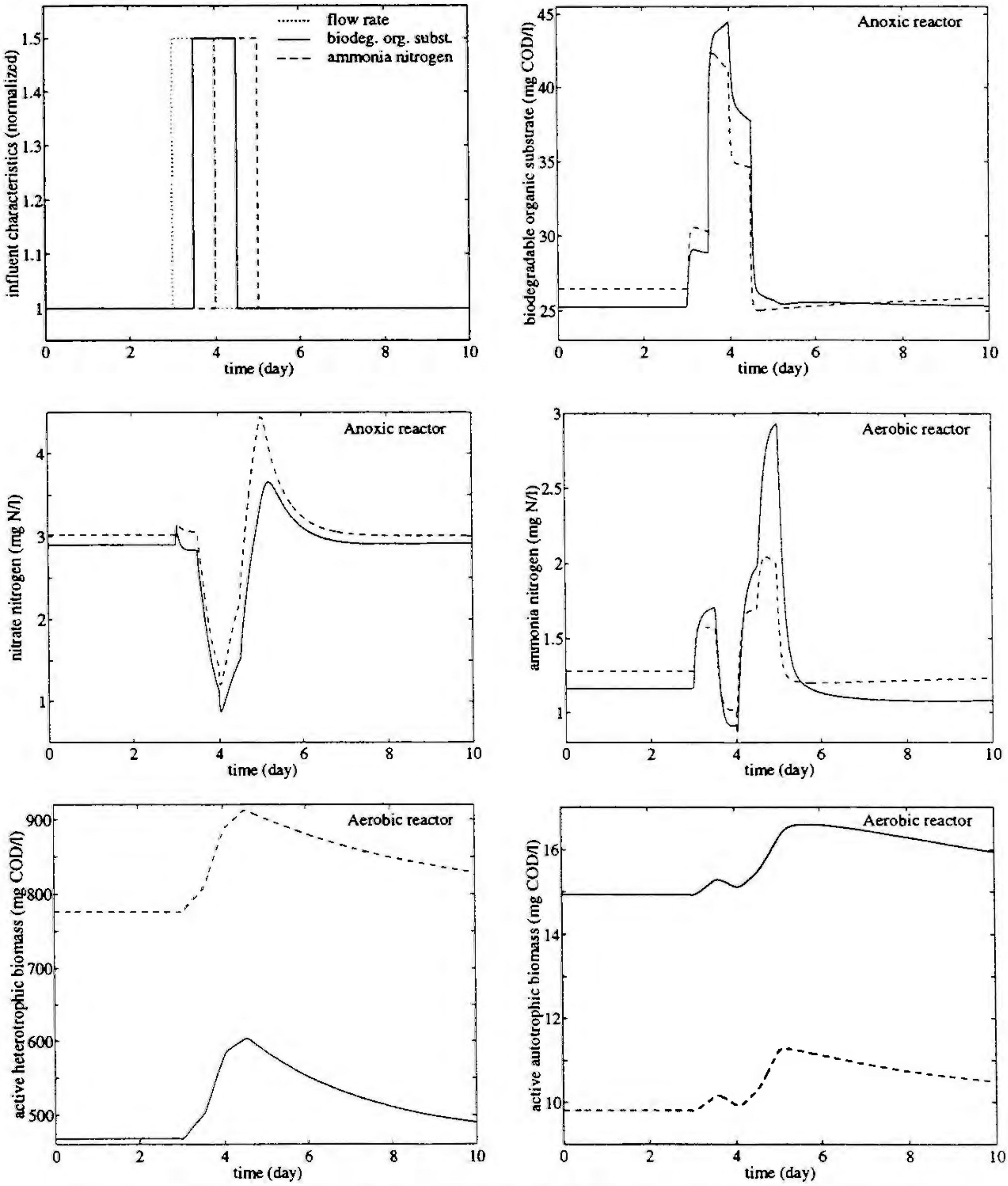


Figure 3. Transient responses for a combined influent flow rate, organic substrate, and ammonia disturbance for the IAWQ model (solid) and the reduced order model (dashed).

6. ACKNOWLEDGEMENTS

This work was financially supported by the Swedish National Board for Industrial and Technical Development (NUTEK) and performed within the Swedish STAMP (Swedish abbreviation for *Control of WWT Plants – Methods and Processes*) research group [7].

7. REFERENCES

[1] Dold, P. L., Activated Sludge System Model Incorporating Biological Nutrient (N & P) Removal. Dept. of Civil Eng. & Eng. Mech., McMaster University, Hamilton, Ontario, Canada, 1992.

[2] Godfrey, K. R. and J. J. DiStefano, Identifiability of Model Parameters. Proc. IFAC Identification and System Parameter Estimation, York, UK, 1985.

[3] Henze, M., C. P. L. Grady Jr., W. Gujer, G. v. R. Marais and T. Matsuo, Activated Sludge Model No. 1. IAWQ Sci. and Tech. Report No. 1, IAWQ, London, UK, 1987.

[4] Jeppsson, U., On the Verifiability of the Activated Sludge System Dynamics. Licentiat thesis, Dept. of Ind. Elec. Eng. & Automation, Lund Institute of Technology, 1993.

[5] Jeppsson, U. and G. Olsson, Reduced Order Models for on-line Parameter Identification of the Activated Sludge Process. In: B. Jank (Ed.), 6th IAWQ Workshop on Intrumentation, Control and Automation of Water & Wastewater Treatment and Transportation Systems. Banff and Hamilton, Canada, June 17-25, 126-136, 1993.

[6] Olsson, G. and J. F. Andrews, The Dissolved Oxygen Profile - a Valuable Tool for Control of the Activated Sludge Process. Water Research, 12 (1978), 985-1004.

[7] Olsson, G., Operations and Control in Wastewater Treatment - some Swedish Experiences. Wat. Sci. Tech., 24(6) (1991), 193-200.

[8] Spanjers, H. and G. Olsson, Modelling of the Dissolved Oxygen Probe Response in the Improvement of the Performance of a Continuous Respiration Meter. Water Research, 26(7) (1992), 945-954.

[9] Spanjers, H., G. Olsson and A. Klapwijk, Determining Influent short-term Biochemical Oxygen Demand by Combining Respirometry and Estimation. In: B. Jank (Ed.), 6th IAWQ Workshop on Intrumentation, Control and Automation of Water & Wastewater Treatment and Transportation Systems. Banff and Hamilton, Canada, June 17-25, 249-263, 1993.

[10] Takács, I., G. G. Patry and D. Nolasco, A Dynamic Model of the Clarification-Thickening Process. Water Research, 25(10) (1991), 1261-1273.

[11] Vanrolleghem, P. A. and W. Verstraete, Simultaneous Biokinetic Characterization of Heterotrophic and Nitrifying Populations of Activated Sludge with an On-line Respirographic Biosensor. In: B. Jank (Ed.), 6th IAWQ Workshop on Intrumentation, Control and Automation of Water & Wastewater Treatment and Transportation Systems. Banff and Hamilton, Canada, June 17-25, 227-237, 1993.

[12] Vitasovic, Z. Z., An Integrated Control Strategy for the Activated Sludge Process. Ph.D. Dissertation, Rice University, Houston, Texas, pp. 288, 1986.

[13] Wentzel, M. C., G. A. Ekama and G. v. R. Marais, Processes and Modelling of Nitrification Denitrification Biological Excess Phosphorus Removal Systems - a Review. Wat. Sci. Tech., 25(6) (1992), 59-82.

MODELLING AND SIMULATION OF WASTEWATER TREATMENT PLANTS BASED ON BIOFILTERS

J.M. LE LANN[1], J. JACOB[1], B. KOEHRET[1], B. CAPDEVILLE[2], K.M. N'GUYEN[2],
J.P. BABARY[3], S. BOURREL[3]

1. LEAP/Groupe d'Analyse Fonctionnelle des Procédés
ENSIGC (INPT), 18 chemin de la loge, 31078 Toulouse Cedex, France
2. INSAT/Unité de Recherche Traitement Biologique
Complexe scientifique de Rangueil. Département GPI, 31077 Toulouse Cedex, France
3. LAAS-CNRS/Equipe de Recherche Conduite de Procédés Biotechnologiques
7 avenue du Colonel Roche, 31077 Toulouse Cedex, France

ABSTRACT

In this paper, a general mathematical model describing the dynamic behaviour of a submerged granular bed biofilter is presented as well as simulation results obtained by two numerical approaches (a global approach using finite difference method connected with a Gear's integration method for DAE systems and an orthogonal collocation approximation approach) with comparison with pilot plant data.

1. INTRODUCTION

Most of the conventional wastewater plants are based on activated sludge process and a great number of contributions are devoted to the modelling and simulation of such systems in view to their automatic control. More recently, contributions have appeared in the literature on the modelling and dynamic simulation of biological wastewater treatment plants based on a new type of reactors, the so-called biofilters. Their main advantages are their high compacity and efficiency for biological epuration (carbon, nitrogen), their good integration in the environment (no noise, no smell), their low energy consumption and sludge production and the fact they don't need secondary clarifiers.

The dynamic simulation of these kind of process is very useful for model validation, performance investigation, parameter estimation and control strategies development with automatic process management in mind. In this context, a dynamic mathematical model for submerged granular bed biofilters is proposed here. It is presented in a general form and can be used for many different applications (denitrification, carbon removal, nitrification, denitritation...). Knowing the basic biological and physical phenomena, macroscopic balances equations are written and lead to a system of non-linear partial differential and algebraic equations.

First of all, the simulation is performed with a strategy based on the method of lines connected with an adaptated form of Gear's integrator for DAE systems with automatic state event detection. Moreover, in view to solve different problems attached to process control (estimation of unknown state variables, optimization of control for on line applications,...), the full model has been reduced to a finite dimensional state equation system using a functional approximation scheme : the orthogonal collocation method. These methods are used for two applications of the biofilters (denitritation and denitrification) and the results obtained are compared with pilot plant data.

2. MODEL DEVELOPMENT

Basic assumptions :

- The liquid is in plug flow.
- The gazeous phase is not taken into account because the gazeous hold-up may be neglected compared to the liquid one.
- The suspended particles in the influent are taken into account.

- The initial amount of biomass is considered as uniform in the filter.
- The decay of biomass is neglected (the residence time of microorganisms never exceeds a few days because of washing due to filter clogging).
- The detachment of biofilm and particles retained by filtration is neglected (laminar flow).
- The temperature and the PH (close to 7) are constant.

Unlike many biofilm models, the diffusion phenomena across the biofilm are not taken into account because different works done on aerobic [6] and anaerobic biofilms show that the biological reaction occurs only at the edge of the biofilm.The active/desactivated biomass concept [6] is used to describe the phenomena : when the support is fully settled by microorganisms, the active biomass amount stays constant while there is accumulation (due to inhibiting products) of desactivated biomass, keeping on growing and contributing to the filter clogging.

The rates equations are based on the IAWPRC model (for activated sludge) [2] with double substrate limitation characterized by a double Monod-type kinetic law for the electron donor D (organic substrate...) and the electron acceptor A (oxygen, nitrate...). The growth rates for active and total biomass are :

$$\frac{\partial C_B}{\partial t} = \frac{\partial C_A}{\partial t} + \frac{\partial C_D}{\partial t} = \mu_{max} \frac{S_D}{S_D + K_D} \frac{S_A}{S_A + K_A} C_A$$

The stœchiometric coefficients associated to these rates are chosen to satisfy the reduction degree balance and the organic compound concentrations are given in terms of chemical oxygen demand (COD).

Unlike others biofilter models, the retention (deep filtration) of the suspended particles in the filter is taken into account (suspended particles Ml → retained particles Mr). The retention kinetic is written as :

$$\frac{\partial C_{Mr}}{\partial t} = \frac{Q\, k\, X_{Ml}}{\Omega}$$

where k is the filtration coefficient which may be written by using different ways depending on the retention and was, after different trials, considered as constant.

A material balance is written for each component involved in the biological or filtration physical reaction, for a liquid in plug flow :

$$\text{accumulation} = \text{input} - \text{output} \pm \text{generation}$$

$$\frac{\partial ([\,]dV)}{\partial t} = Q([\,]_z - [\,]_{z+dz}) + \nu\, r\, dV$$

A term for oxygen transfer in the liquid phase has to be added in the oxygen balance for an aerobic reaction. The pressure drop in the filter is also estimated by using the Blake-Kozeny correlation and, in the case of a floating bed, an algebraic equation may be added to describe the bed rising in the filter. Thus the full dynamic model leads to a partial differential and (or) algebraic system (PDAE) of (n=9) equations [4].

2. GLOBAL RESOLUTION USING THE METHOD OF LINES

The problem to be solved may be written in vectorial form with adequate initial and boundary conditions :

$$\mathbf{D}\,(\mathbf{s},\, \mathbf{t}) \frac{\partial \mathbf{s}}{\partial t} = \mathbf{f}\,(\mathbf{s},\, \mathbf{t},\, \mathbf{u},\, \mathbf{p}) + \mathbf{g}\,(\mathbf{s},\, \mathbf{t},\, \mathbf{u},\, \mathbf{p}) \frac{\partial \mathbf{s}}{\partial z}$$

With generalization in mind, a numerical strategy inspired by differential and algebraic equations (DAE) treatment associated to a space discretization using the method of lines is applied. It allows :
- a global treatment of the system without discrimination between the variables (especially for algebraic ones).
- a treatment of the physical model in its original form, as it was written, without preliminary manipulations.
- the use of sophisticated methods which has been developed for initial value differential-algebraic equations.
- the possibility, for further works, to use the developments involved in DAE systems (parameter estimation,...).

The main failures of the method are the large size of the resulting DAE system obtained after discretization (this problem can be reduced by using adequate numerical conditionment of the system) and the rigidity of the fixed grid discretization scheme which is not really a problem in our case (because their is no sharp space profiles) and could be solved by using a moving grid approach.

The numerical method of lines [7] consists of a discretization for the spatial variable (vertical position z) in N discretisation points. Each state variable s is transformed into N variables corresponding to its value at each discretization point. The spatial derivatives are approximated using finite difference formula on 3, 5 or 7 points (the best results for accuracy and computation time efficiency were obtained with 5 points).

The resulting N x n DAE system is solved by Gear's multistep and multi-order implicit method based on a predictor-corrector scheme [3]. At any time step, a non-linear system has to be solved in the corrector loop. The Newton-Raphson method is used for a quick convergence and because we have a good initialization with the predictor. The derivatives are computed analytically in the dynamic operator which has a block structure. Each block is an n x n matrix corresponding to the model equations at one discretization point. The extra-diagonal terms come from the spatial derivatives approximation. Thus the dynamic operator is a multi-diagonal block matrix and we gain memory space accuracy and computation time saving by treating it like a banded matrix, transforming it to a rectangular matrix for which the pivot research is done vertically.

As the biofilter is cleaned when the pressure drop reaches an upper limit, a procedure to detect automatically state events has been implemented. In case of time events, polynomial approximations are used to simulate flow-rate and concentration input variations without introducing discontinuities.

3. RESOLUTION WITH AN ORTHOGONAL COLLOCATION APPROXIMATION APPROACH

In the field of process control based on the use of infinite dimensional models, a compromise between the low model complexity and the high solution accuracy has to be found. That's why, after the global resolution method presented before, a functional approximation method is used to obtain a state representation with a low finite dimension with a satisfying accuracy.

As presented previously, the dynamics of the process is described by a non-linear distributed parameter model. The partial differential equations of the model are reduced to ordinary differential equations (solved by using Runge-Kutta integration method) by using an orthogonal collocation method [8]. The choice of this method for the space discretization of the biofilter model has been dictated by two main reasons. First of all, this method is largely used and accepted in chemical engineering [5] for the reduction of dynamic models of tubular reactors ; secondly orthogonal collocation is known to be an efficient and powerful method, provided some precautions are respected about the choice of base functions and others parameters. Moreover, it offers the advantages that its implementation is easier, that the nature and the dimension of the state variables remain unchanged after the reduction procedure and that it is conservative for mass and heat balances.

The collocation method consists of expanding the variables as a finite sum of products of time functions and space functions.

$$s\ (z,t) \cong \sum_{i=0}^{N+1} f_i(z)\ .\ C_i\,(t) \qquad \text{with N number of collocation points}$$

The first question to be put is : how can we choose the base functions $f_i(z)$ and the right value for N ? Polynomials are the most usual choice, preferably orthogonal polynomials, avoiding ill-conditionned matrix in the resolution. In what follows we choose the Lagrange interpolation polynomials, defined through (N+1) points and satisfying an orthogonality condition.

It has been shown that the best approximation is obtained when the collocation points are used as interpolation points. Moreover the best collocation point location corresponds to the zeros of orthogonal polynomials such as Jacobi polynomials $P_N^{(\alpha,\beta)}$:

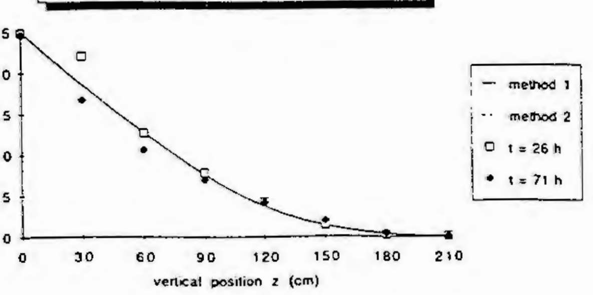
$$\int_{z_0}^{z_L} (z - z_0)^{\beta}\,(z_L - z)^{\alpha}\, z^j\, P_N^{(\alpha,\beta)}\, dz = 0$$

Both parameters α and β are considered as optimization parameters of the collocation point location. The best choice depends on the nature of the model. It has been shown [1] that, for non-linear systems, it's better to place the collocation points where the non-linearity is more pronounced. In general, the approximate solution accuracy increases with the collocation point number. In fact, we showed that the model accuracy depends more on the collocation point location than on their number; inappropriate values of α, β and N lead to numerical instabilities.

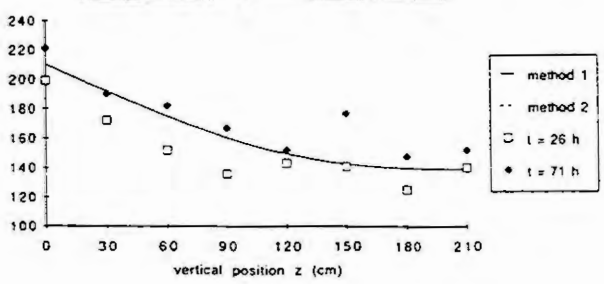
4. APPLICATION TO A DENITRITATION FILTER ($NO2^- \rightarrow N_2$)

The experiments were lead on a synthetic water and the data obtained are compared with the results of the simulations performed with the numerical approaches (meth. 1 : method of lines ; meth. 2 : collocation method).

NITRITE CONCENTRATION IN THE FILTER (g(N-NO2)/m3)



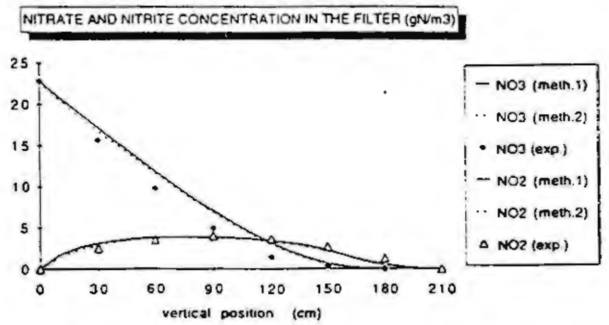
CARBON CONCENTRATION IN THE FILTER (gDCO/m3)

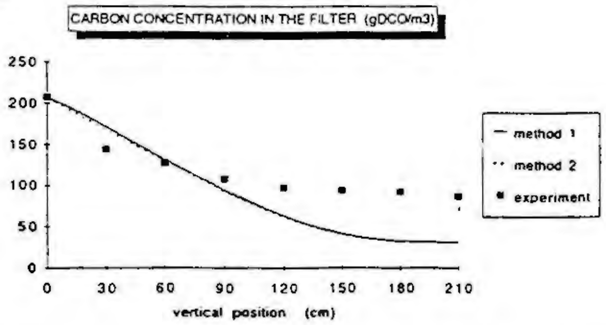


The two methods give equivalent results and a good simulated/experimental data agreement for nitrites and carbon concentrations is obtained.

5. APPLICATION TO A DENITRIFICATION FILTER ($NO3^- \rightarrow NO2^- \rightarrow N_2$)

In order to dissociate nitrates and nitrites, two distinct biological reactions are considered in the model : denitratation ($NO3^- \rightarrow NO2^-$) and denitritation ($NO2^- \rightarrow N_2$).





The agreement between the two methods is good but less perfect than in the case of denitritation : this is probably due to harder integration conditions (zeros initial nitrites concentration). However good simulated/ experimental data agreement is obtained for nitrites and nitrates concentrations. The results are less satisfying for carbon but it has to be precised that the simulations were performed without real estimation of kinetic and stœchiometric parameters : some comes from the literature, some were measured experimentally (Y_H, the biomass yield), others (C_{Amax}, the maximum active biomass concentration) were chosen after a brief trial/error strategy. A better fit should be obtained by using a rigorous parameter estimation procedure, currently in development.

5. CONCLUSION

A general model has been developed for the dynamic simulation of wastewater treatment units by submerged biofilters. The experimental/simulated data agreements obtained are satisfying and allow to strengthen the assumptions and to validate the biological part of the model. On a numerical point of view, the two methods used to reduce the distributed parameter model to a DAE system (method of lines and orthogonal collocation) give similar results, in an equivalent CPU time.

For further control development, the collocation method, leading to a smaller dimension system, will be used but the global method will remain necessary to validate the model reductions and to know how far it may be reduced without losing in consistency : the problem consists in finding a good compromise between low complexity and high accuracy of the model.

6. REFERENCES

[1] Cho, Y.S., B. Joseph, Reduced-order steady-state and dynamic for separation processes, AIChE J., 9 (1992), 873-883.

[2] Henze, M., C.P. Leslie Grady, W. Guyer, G.V.R. Marais, T. Matsua, (IAWPRC task group), A general model for single sludge wastewater treatment systems, Wat. Res., 21 (1987), 505-515.

[3] Hindmarsh, A.C., LSODE and LSODI, two new initial value ordinary differential equation solvers, ACM SIGNUM Newsletter, 15 (1980), 10-11.

[4] Jacob, J., J.M. Le lann, H. Pingaud, B. Koehret, K.M. N'Guyen, B. Capdeville, Dynamic simulation of submerged packed biofilters in wastewater treatment plants, Comp. Chem. Eng., 18 suppl. (1994), s639-s643

[5] Jorgensen, S.B., Fixed bed reactor dynamics and control - a review, Proc. IFAC Control of Distillation Columns and Chemical Reactors, Pergamon, Oxford, (1986), 11-24.

[6] N'Guyen, K.M., Description et modélisation des films biologiques aérobies, Thèse Doctorat INSAT, n°96 (1989).

[7] Schiesser, W.E., An introduction to the numerical method of lines integration of partial differential equations, Lehigh universities and naval air development center (1977).

[8] Villadsen, J.V., M.L. Michelsen, Solution of differential equation models by polynomial approximation, Prentice-hall International, Englewood Cliffs, NJ. (1978).

GPS-X: A Wastewater Treatment Plant Simulator

Gilles G. Patry[1], Imre Takács[2]

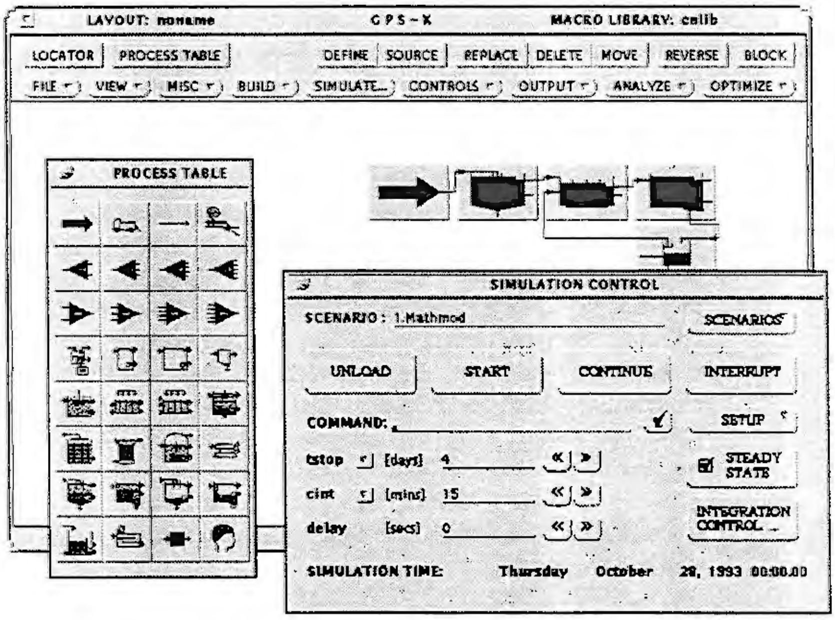
[1] *University of Ottawa, Faculty of Engineering, P.O.Box 450 Station 'A', Ottawa, Ontario, Canada, K1N 6N5*
[2] *Hydromantis, Inc., 1685 Main Street West, Suite 302, Hamilton, Ontario, Canada, L8S 1G5*

ABSTRACT

The General Purpose Simulator (GPS-X) is an object-oriented dynamic simulation package that includes a wastewater toolbox. Its library, containing many of the most popular wastewater treatment unit processes enables the user to handle the simulation of virtually any wastewater treatment plants from headworks to effluent discharge. In addition, GPS-X is a flexible, powerful modelling environment, where users can develop their own mathematical models taking full advantage of the interactive and versatile graphics GPS-X offers. The program has been applied to more than twenty (20) large-scale wastewater treatment plants worldwide.

INTRODUCTION

In the last few years there has been considerable progress in the area of mathematical modelling of wastewater treatment processes. The turning point is undoubtedly the release of IAWQ Activated Sludge Model No. 1. [4] in 1986. Since then, a whole family of activated sludge models (including biological nutrient removal (BNR) technologies) was published [2,8], along with settling (thickening and clarification) [7], biofilm [1], anaerobic [5] and other process models. In fact, it is now possible to simulate the entire sewage works from headworks to effluent disinfection.


LAYOUT: noname

MACRO LIBRARY: cnlib

SCENARIO : 1.Mathmod

COMMAND:

tstop [days] 4

cint [mins] 15

delay [secs] 0

☑ STEADY STATE

SIMULATION TIME: Thursday October 28, 1993 00:00.00

Figure 1. GPS-X: A simple plant layout and the Toolbox

In spite of the development in dynamic modelling, there was practically no progress on the tools (computer languages) used to implement the algorithms. Most of the programs in this field still use Fortran, Turbopascal or other similar low level general purpose languages to create a working environment where the models can be tested and used for research, planning, design and/or operational control. A significant part of coding is spent on developing routines for file I/O, menus, graphics and numerical solutions. To alleviate these problems and to bring efficient, high-power computing into the reach of practising engineers, the General Purpose Simulator (GPS-X) was created utilizing the new low-cost, high-power workstation platform which has become available in the last few years. GPS-X is both a modelling environment for any type of dynamic process, and an extensive, modifiable library of most of the process models available today in this field.

THE STRUCTURE OF GPS-X

GPS-X (from a program organization standpoint) consists of a graphical user interface (GUI) written in C, and simulation code (including the process library) which is written in a high level specialized language. GPS-X uses the *Advanced Continuous Simulation Language* (ACSL) [6] as its programming vehicle. During an *interactive simulation session* with GPS-X, both modules are active in memory and the GUI component acts as a human language interpreter and a graphical output device for the cryptic simulation module. Some of the most interesting features of both modules will be discussed below.

ACSL LANGUAGE FEATURES AND ACCESSORIES

ACSL features an extensive library of numerical methods, including numerical integrators, steady-state solvers and nonlinear optimization algorithms. There are seven *dynamic solvers* as shown in Table 1. During integration, ACSL can handle error limits individually for state variables. Our experience has shown that most models in the wastewater treatment area are stiff, i.e. some variables change on a timescale which is three to four orders of magnitude shorter, than others (e.g. dissolved oxygen against inert sludge constituents). The effect of using an improper solver (fixed step or a variable step with too large error criteria) is illustrated in Figure 2. Here, in the first day, the dissolved oxygen (DO) concentration in the four passes of an aeration tank was integrated using the Euler algorithm (1 minute timestep), then the solver was switched to Gear's method, with an error criteria of 10^{-4}. Dynamic solvers can be changed interactively during runtime with some exception.

Table 1: Available integration algorithms in ACSL

Algorithm	Step	Order
Euler	fixed	first
Runge-Kutta	fixed	second
Runge-Kutta	fixed	fourth
Runge-Kutta-Fehlberg	variable	second
Runge-Kutta-Fehlberg	variable	fifth
Adams-Moulton	variable	variable
Gear's stiff	variable	variable

Available *steady-state iteration algorithms* are Newton-Raphson, steepest descent and a linear decoupled method, developed by Hydromantis specifically for the wastewater engineering field. The first two methods are very efficient algorithms for simple ordinary differential equation (ODE) systems, and will converge within a few dozen iterations. Some of the iterations include calculating the Jacobian matrix, which carries a heavy load of derivative evaluations, usually two or three times the number of states in the system. The real problem though with the Jacobian is that mathematical loops and controllers invalidate it by changing its elements within one iteration. Mathematical loops are inevitably present in a wastewater treatment plant model due to physical "loops", usually recycles. Also, DO, mixed liquor suspended solids (MLSS), etc. controllers are typical in models. The alternative solution in GPS-X is to make use of a linear decoupled iteration method, which requires larger number of iterations, but each iteration involves only one derivative evaluation, and the method is not sensitive to the perturbations occurring in a complex realistic system. This method can even handle parameter changes during iterations, greatly reducing analysis time for the engineer.

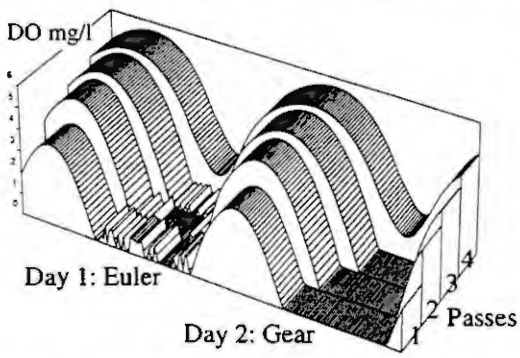


Figure 2. Euler vs. Gear: Numerical instability

There are two advanced *optimization algorithms* available coded in ACSL. The Nelder-Mead simplex method is a direct search method while an indirect quasi-Newton method called BFGS [3] is also accessible through the GPS-X GUI. Both optimization methods are multidimensional, so technically it is possible to optimize any number of parameters. Flexible objective function definition is provided through the GPS-X. Practical experience has shown that applying engineering "common sense" in selecting a few key parameters and defining reasonable limits will speed up the iteration process.

Another useful feature of ACSL is the *explicit structure* designed specifically for dynamic simulations. Table 2. lists all available sections and their use during the simulation . This pre-defined structure essentially frees up the user from the tedious task of organizing loops in the program and ensures that all subroutines are called when appropriate.

Table 2: ACSL code structure

Code segment	Executed	Content
Preinitial	during loading of the program	definitions, default values
Initial	before the dynamic runs starts, at time < 0	steady-state routines
Derivative	every integration timestep	differential equation system
Dynamic	every communication interval	communication with GPS-X
Discrete	at specified frequency	discontinuous code, e.g. controllers
Terminal	after completing the dynamic run	post-processing, e.g. probability distribution

Automatic sorting is another feature which makes ACSL especially powerful for object-oriented code development. The ACSL translator will sort the user specified source code for correct execution order. In effect this means that no attention needs to be paid for the order and placement of object code.

Finally, it is worthwhile to note that the simulation code is *platform independent*, i.e. code developed on one particular type of computer can be ported to practically any other hardware without change.

GPS-X FEATURES

The graphical user interface (GUI) module of GPS-X is what most users see exclusively when using the simulator. GPS-X contains a wastewater toolbox which currently consists of 32 objects common in the wastewater treatment plant as shown in Figure 1 (Process Table). The objects include influents, flowsheet development tools (splitters and combiners), primary, secondary and tertiary treatment process units. The user places these objects on a drawing board to build the treatment plant. Taking advantage of the set of splitters and combiners, the interface provides free flowsheeting, i.e. enables the user to simulate any possible plant layout. In a real plant model it is crucial to consider process interactivity (e.g. effect of sidestreams) in addition to the usual process modelling of the aeration tank or final settler.

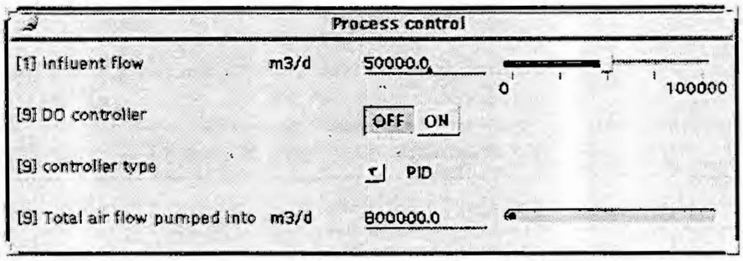
The process icons provide a graphical access to an extensive set of predefined models, currently featuring more than 100 different model variations in four different libraries. A library is defined by a fixed set of state variables, which is a convenient and efficient way of handling model compatibility. The four available libraries are described in Table 3. These libraries contain customized versions of the most important mathematical models available today, including carbonaceous, nitrogen-removal and bio-P technologies, one- and two-dimensional reactive and non-reactive settler models, and many more. Existing models are user modifiable on the source code level, and new models can easily be inserted by the user.

Table 3: GPS-X libraries

Library	Description
CN	basic carbonaceous and nitrogen removal processes
CNP	CN plus bio-P processes
IP	industrial pollutant library
CN2	advanced carbonaceous - nitrogen removal processes

In addition to the usual steady-state and dynamic numerical capabilities, GPS-X features a set of advanced functions such as built in parametric or Monte Carlo sensitivity analysis, nonlinear numerical optimization capabilities, and interactive access to several types of data bases. Data can be extracted and visually presented both for the purpose of driving functions (e.g. influent flow or composition) or calibration (e.g. effluent data).

The primary purpose of GPS-X is to handle the large amount of information which the user encounters during the simulation of large-scale wastewater treatment plants as efficiently as possible. Both inputs and outputs are handled in an innovative, interactive way. Parameters to be changed during the run, either manually, or for automated analysis or optimization purposes are controlled by definable GPS-X controllers. Four different types of controllers of the many available, placed on a Control Panel are shown in Figure 3. for illustration. GPS-X offers a variety of on-line graphs and analysis tools to enhance data visualization. In addition to the usual X-Y (linear) graph type (also available in GPS-X in a scrolling format), which is most often used for analysis of temporal variation, the software places emphasis on the ability to display spatial variations as well. Figure 4 depicts a typical histogram, where the concentration profile of a hypotethical inert component "A" developing in a plug flow reactor (70 mixed tanks in series) is shown after a spike in the influent. Steady-state values of variables can be accessed on-line in digital form, as well, and GPS-X has extensive reporting capabilities to help the user keep organized.


Process control

[1] influent flow m3/d 50000.0

[9] DO controller OFF ON

[9] controller type PID

[9] Total air flow pumped into m3/d 800000.0

Figure 3. GPS-X controllers

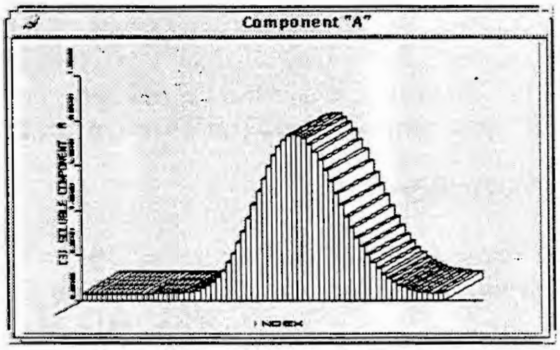


Figure 4. Concentration profile in a plug-flow reactor

CONCLUSIONS

The General Purpose Simulator described in this paper is an open-architecture object oriented software designed to facilitate dynamic modelling of large-scale wastewater treatment plants. The program has an extensive library of numerical methods and process models to reduce program development time for practising engineers and features a variety of advanced features including interactive controllers and graphics, built in analysis and optimization tools. The program has been applied to more than twenty wastewater treatment plants worldwide.

ACKNOWLEDGEMENTS

Development of the GPS-X was partially founded by Environment Canada.

REFERENCES

[1] Arvin, E., Harremoës, P. (1990). Concepts and Models for Biofilm Performance. *Wat. Sci. Techn.*, **22**, 171.

[2] Dold, P.L. (1990). Incorporation of Biological Excess Phosphorus Removal in a General Activated Sludge Model, *Proc. 13th Int. Symposium on Wastewater Treatment*, Montreal, Canada, 83-113.

[3] Edgar, T.F. and Himmelblau, D.M. (1988). Optimization of Chemical Processes, McGraw-Hill, New York.

[4] Henze, M., Grady Jr., C.P.L., Gujer, W., Marais, G.v.R., Matsuo, T. (1987). Activated Sludge Model No. 1. *IAWPRC Task Group on Mathematical Modelling for Design and Operation of Biological Wastewater Treatment*, IAWPRC, London, England.

[5] McCarty, P.L., Mosey, F.E. (1991). Modelling of anaerobic digestion processes. *Wat. Sci. Techn.*, **24**, 17-33.

[6] Mitchell and Gauthier Associates (1987). Advanced Continuous Simulation Language (ACSL). *Reference Manual*. Concord, Mass.

[7] Takács, I., Patry, G.G. and Nolasco, D. (1991). A Dynamic Model of the Clarification-Thickening Process. *Wat. Res.* **25**, No 10. 1263-1271.

[8] Wentzel, M.C., Ekama, G.A., Marais, G.v.R. (1992). Processes and Modelling of Nitrification-Denitrification Biological Phosphorus Removal Systems - A Review. *Wat. Sci. Tech.* **25**, No. 6, 59-82.

MODELING AND SIMULATION OF SEDIMENTATION PROCESSES IN A LOWLAND RIVER

Christof ENGELHARDT, Dieter PROCHNOW, Heinz BUNGARTZ
Institute of Freshwater and Fish Ecology
Rudower Chaussee 5
12484 Berlin, Germany

Abstract. In a case study a sedimentation process observed in a tidally unaffected reach of the lower Elbe River was simulated using a two-dimensional vertically integrated model and a three-dimensional model to predict the distribution of particulate matter in the turbulent flow. The observed sedimentation process could bė confirmed by model simulation under certain conditions.

1. INTRODUCTION

Sediment load is an ecological criterion for the assessment of water quality as well as dissolved or particle-bound pollutants, governed by sediment transport. In a section of the Elbe River upstream from Geesthacht a decreasing concentration of suspended particle load was observed during several measuring campaigns and for a few discharge situations [5]. The reduction of suspended sediment seems to be caused by a selfpurification process due to particle settling. To verify this assumption the turbulent sediment transport in this Elbe River reach was simulated applying two-and three-dimensional mathematical models [1,2,3,4,6].

2. EQUATIONS

The transport in rivers can be modeled by the turbulent momentum equations (Reynolds' equations) to determine the velocity field and by special convection-diffusion equations to calculate concentrations of different particle fractions each characterized by a mean settling velocity. The governing equations of a depth-integrated (2D) model, see [2], are given as follows (sum convention on j = 1,2):

Momentum equations

$$V_j \frac{\partial V_i}{\partial x_j} - \frac{\partial}{\partial x_j}\left(D_H \frac{\partial V_i}{\partial x_j}\right) + g\frac{\partial \eta}{\partial x_i} = -\frac{D_V}{H^2} V_i \qquad i=1,2$$

$$\frac{\partial (H V_j)}{\partial x_j} = 0$$

$$V_i = \frac{1}{H}\int_0^H v_i dx_3 \quad , \quad H = h + \eta$$

in which:

v_i	component of the turbulent velocity in i-direction, i = 1, 2, 3
V_i (x)	depth-averaged velocity in i-direction, i = 1, 2
H (x)	water depth
x = (x_1, x_2)	horizontal cartesian coordinates
D_V	vertical eddy viscosity
D_H	horizontal eddy viscosity
g	gravity
η	vertical deviation of water surface above level h

Boundary conditions

$V_i(x) = V_i^{B}(x)$ or $(\partial V_i/\partial x_1)\, n_1 + (\partial V_i/\partial x_2)\, n_2 = 0$

V_i^B boundary value
$n = (n_1, n_2)$ outward directed normal vector

Transport equation

$$V_1 \frac{\partial c}{\partial x_1} + V_2 \frac{\partial c}{\partial x_2} - \frac{\partial}{\partial x_1}\left(\frac{D_H}{sz}\frac{\partial c}{\partial x_1}\right) - \frac{\partial}{\partial x_2}\left(\frac{D_H}{sz}\frac{\partial c}{\partial x_2}\right) = -\frac{1}{H} f_s$$

in which:

$c\,(x\,)$ concentration of particulate/dissolved matter
f_s sedimentation rate
sz turbulent Schmidt number

Boundary conditions

$c\,(x) = c^B(x)$ or $(\partial c/\partial x_1)\, n_1 + (\partial c/\partial x_2)\, n_2 = 0$

c^B boundary value

A corresponding three-dimensional model is given in [3].

3. COMPUTATIONAL DOMAIN AND INPUT DATA

The area of interest is a tidally unaffected eight km section of the Elbe River (from km 577 to km 586) upstream the Geesthacht weir (Fig. 1). Neither flow nor concentration sinks or sources were observed. The lock was assumed to be closed.

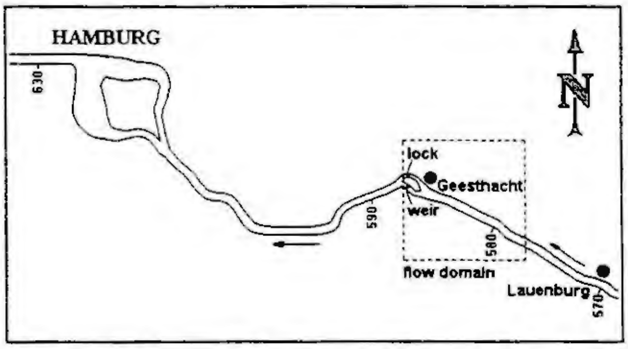


Fig.1 Map of the lower Elbe River with simulated reach

Aside from concentration measurements for this river section a few settling velocity spectra were measured simultaneously. Together with bathometric plans of the Elbe River these experiments permit a model simulation with following input values: constant inflow velocity v_i = 0,4 m/s (discharge Q = 286 m^3/s , inflow cross-section area A = 715 m^2); eddy viscosities D_H = 1 m^2/s, D_V = 0,0015 m^2/s; inflow suspended sediment concentration c = 35 mg/l; turbulent Schmidt number sz = 0,5.

4. SIMULATION AND RESULTS

To simulate the suspended sediment load in this Elbe River reach, the program package SEDIFLOW was used [3,4]. In a first step the three-dimensional version of SEDIFLOW was applied to check the concentration distribution for the transport problem given above (for turbulent flow field see Fig. 2a,b). Because of the almost uniform concentration profiles in vertical direction (see Fig.'s 4a,b,c) in a second step, the same problem was solved by the vertically integrated model (see Fig. 3) .

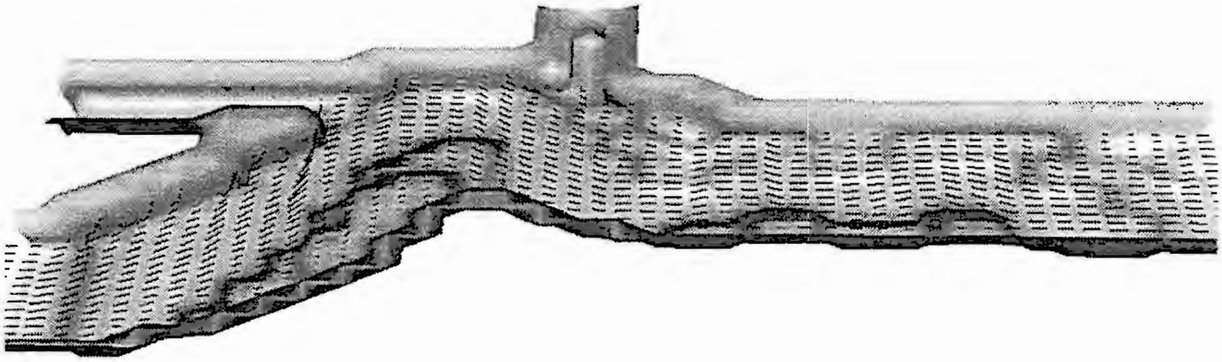


Fig.2a Turbulent flow near surface (3D-SEDIFLOW simulation)

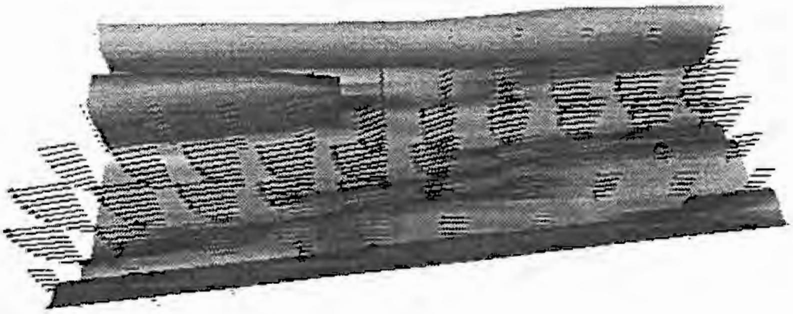


Fig.2b Three-dimensional flow in detail (3D-SEDIFLOW simulation)

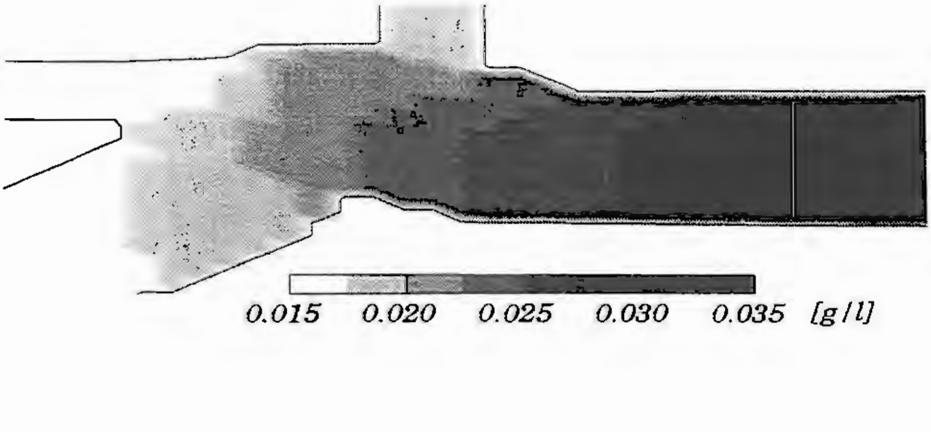


Fig.3 Depth-averaged distribution of suspended sediment concentration (2D-SEDIFLOW simulation)

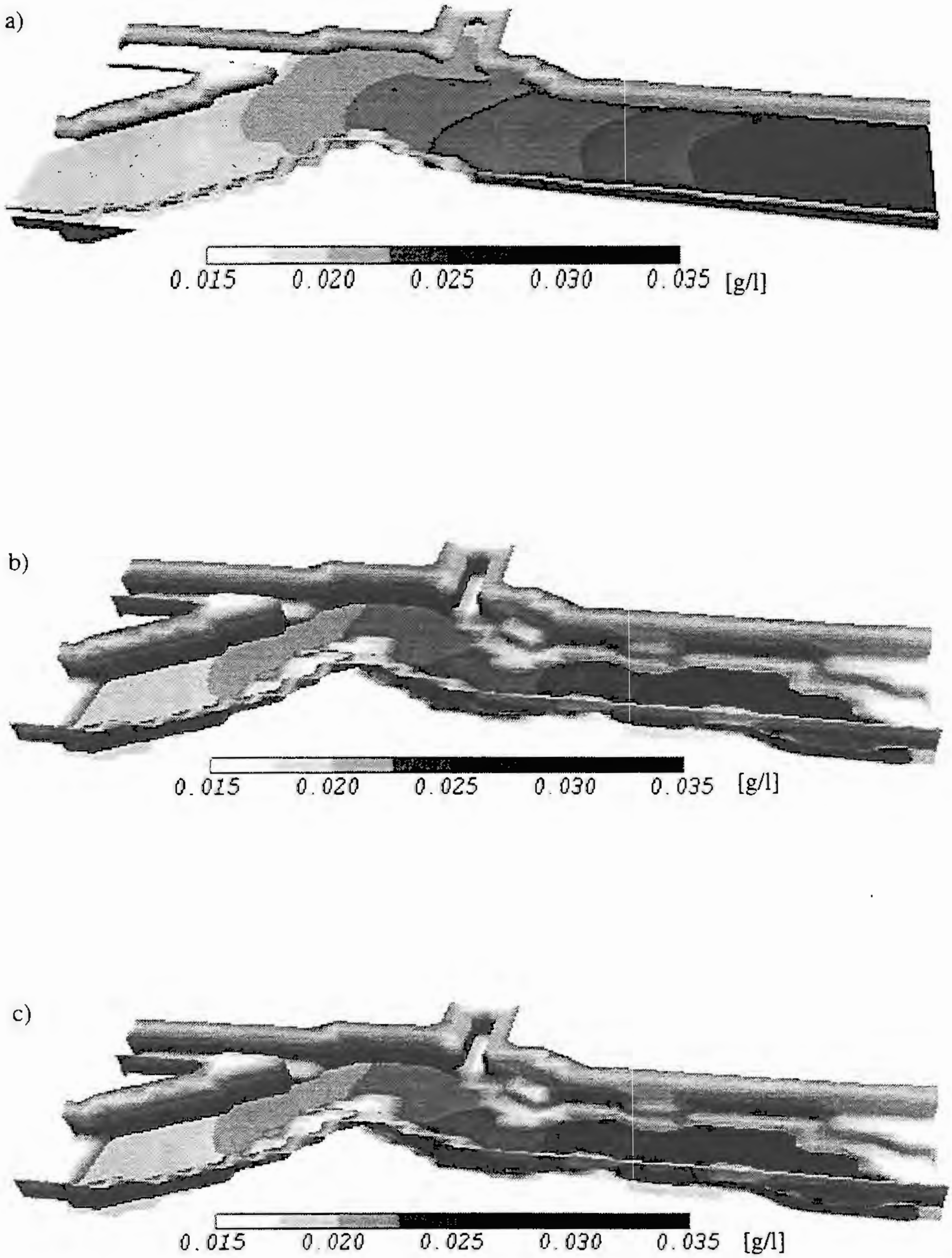

a)
0.015 0.020 0.025 0.030 0.035 [g/l]
b)
0.015 0.020 0.025 0.030 0.035 [g/l]
c)
0.015 0.020 0.025 0.030 0.035 [g/l]

Fig.4 Three-dimensional distribution of suspended sediment concentration (3D-SEDIFLOW simulation); horizontal cross-section at surface (a), in 2m depth (b) and in 4m depth (c)

In SEDIFLOW the sedimentation rate f_s is modeled by

$f_s = s\,(c - c_{eq})$	sedimentation rate
s	mean settling velocity of particulate matter
$c_{eq} = \begin{cases} c\,, & v_\tau > v_{CR} \\ 0\,, & v_\tau \le v_{CR} \end{cases}$	equilibrium concentration (written in case of non-erodible bed)
$v_{CR}^3 = 0{,}15\; g\; \upsilon\; (\varrho_p/\varrho_w - 1)$	critical bed shear velocity in which: υ - kinematic viscosity; ϱ_p, ϱ_w - densitiy of particles and water, respectively

The measured settling velocity spectra yield a characteristic value $s = 10^{-4}$ m/s of the settling velocity [5]. Using this value and the observed [5] typical particle diameter of 25 µm a mean particle density $\varrho_p = 1320$ kg/m^3 results by applying Stokes' law. From here a critical shear velocity of $v_{CR} = 0{,}078$ m/s is received (gravitational acceleration $g = 9{,}81$ m/s^2; kinematic viscosity $\upsilon = 10^{-6}$ m^2/s; water density $\varrho_w = 1000$ kg/m^3). The bottom shear velocities obtained from steady state solution of the three-dimensional problem are almost everywhere below the critical value. Thus here the suspended sediment transport was dominated by deposition. In two-dimensional simulation the flow solution was forced to yield bed shear velocities similar to those of three-dimensional simulation. Fig. 5 depicts the surface suspended sediment concentration at several river cross sections computed by the 3D-and 2D-SEDIFLOW model in comparison with the experimental results of Puls and Kühl [5].

It is shown that the observed sedimentation process in a real river reach could be confirmed with sufficient accuracy by model simulation.

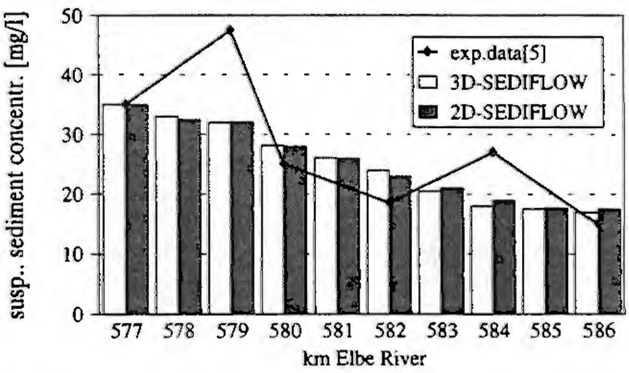


Fig.5 Comparison of 3D-SEDIFLOW and 2D-SEDIFLOW simulation with suspended sediment concentration data [5] observed in the Elbe River

5. ACKNOWLEDGEMENTS

The work reported herein was supported by the GKSS Forschungszentrum Geesthacht GmbH.

6. REFERENCES

[1] Celik, I; Rodi, W.: Mathematical Modelling of Suspended Sediment Transport in Open Channels, 21st Congress, Int. Association for Hydraulic Research, Melbourne, Australia, August 1985

[2] Chu, W.; Liou, I.-Y.; Flenniken, K.D.: Numerical Modeling of Tide and Current in Central Puget Sound: Comparison of Three-Dimensional and Depth-Averaged Model, Water Resour. Res. 24(1989),721-734

[3] Prochnow, D.; Bungartz, H.; Friedrich, H.-J.: Ein modifizierter Chorin'scher Algorithmus zur Berechnung von Gewässerströmungen, Acta hydrophys. 34 (1990), 97-129

[4] Prochnow, D; Bungartz, H; Engelhardt, Ch.: Modeling and Simulation of Contaminant Transport and Sedimentation Processes in Fluvial Systems. In: R. Vichnevetsky, J.J.H. Müller (eds.): Proceedings of the 13th IMACS World Congress on Computation and Applied Mathematics, Criterion Press, Dublin 1991, vol. 4, 1970-1971

[5] Puls, W.; Kühl, H.: Die Geschwindigkeit von Elbeschwebstoff bei Lauenburg und Bunthaus, GKSS 89/E/54 (1989)

[6] van Rijn, L.C.: Sediment Transport, Part II: Suspended Load Transport, J. of Hydraulic Eng. 110 (1984), 1613-1641

APPLICATION OF A PARTICLE SIMULATION MODEL IN RADIATION PROTECTION

G. WIESNER, S. GOTTWALD, U. NIELSEN, M. v. REKOWSKI
Hahn-Meitner-Institut Berlin GmbH
Glienicker Str. 100
D–14109 Berlin

Abstract. As a part of a new radiation monitoring system at the Hahn–Meitner–Institut Berlin a software system for determining short–range atmospheric dispersion processes was implemented. This system is intended to calculate the impact of released radioactive material from a nuclear facility and is based on a Lagrangian random walk model simulating the trajectories of a large number of released particles.

1. INTRODUCTION

Within the scope of environmental protection the dispersion behavior of pollutants released into the atmosphere is of great interest. To allow the calculation of pollutant concentrations or for estimating their influence a multitude of atmospheric dispersion models has therefore been developed respectively realized in practical applications. These models can be classified into three groups [3]: K– or Eulerian models, which solve numerically with finite element methods the advection–diffusion–equation, describing transport and diffusion of pollutants in the atmosphere. Gaussian models, which are based on an analytical solution of a strongly simplified advection–diffusion–equation. Lagrangian random walk models, which are derived from the theory of statistics.

The advantage of models from the last group is an easier consideration of varying meteorological and source conditions and complex terrain structures. On the other hand Lagrangian random walk models are very computer time–consuming, however the involved algorithms allow parallelization to a high degree. Thus, in the last years the use of these models gains more and more in significance.

From the variety of atmospheric dispersion problems the described application of a Lagrangian walk model is restricted to short–range ($\leq$ 20 km) dispersion processes caused by short–time releases of inert substances from one or few sources. Especially the release of radioactive material from a nuclear facility is considered. Within the scope of radiation protection it is required in case of an accidental release to predict the dispersion of the radioactive pollutant cloud for the next hours in a few minutes. Besides calculation of the radionuclide concentrations in the atmosphere it is necessary to take into account radioactive decay just as fall–out and wash–out with dry and wet deposition on the ground and to compute different radiation doses caused by radioactive beta– or gamma–radiation from ground and cloud or by inhalation of radioactive material. To determine the gamma–radiation from the cloud a three–dimensional integral requiring time–integrated concentrations has to be solved. These additional points — radioactive decay and dose calculations — make a difference to "normal" pollutants in practical dispersion models, where only calculation of concentrations is of interest.

In chapter 2 an overview of the chosen simulation model and the integration algorithm for dose calculations is given. It is followed by some technical details of the implemented dispersion model as part of the new radiation monitoring system at the Hahn–Meitner–Institut Berlin. Some concluding remarks complete the paper.

2. THEORETICAL OVERVIEW

2.1 Transport Model

In Lagrangian random walk models the main idea is tracking the trajectories of individual released particles. These particles are considered as representatives of a pollutant (center of gravity of a pollutant ensemble) emitted from a specified source. The trajectory of a particle is determined by two components, the transport by mean wind and the transport by turbulences of the air. The influence of air turbulence on the particle trajectories is simulated by random processes.

In the simulation algorithm the position of a particle at time $t_n + \Delta t$ is computed from the position at time t_n by

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \Delta t \cdot \mathbf{u}_n$$

where the velocity $\mathbf{u}_n$ of a particle at time t_n is composed of the advective velocity $\bar{\mathbf{u}}_n$ and the turbulent velocity $\mathbf{u}'_n$:

$$\mathbf{u}_n = \bar{\mathbf{u}}_n + \mathbf{u}'_n.$$

The turbulent velocity $\mathbf{u}'_n$ is subdivided into a random and a correlation component. The correlation part realizes that particles cannot alter their turbulent velocity arbitrary due to the inertia of the surrounding air. They have a "memory", so–called Lagrangian correlation time, which varies from some seconds to a few minutes in the lower planetary boundary layer. In order to have a correlation between the turbulent velocities $\mathbf{u}'_n$ and $\mathbf{u}'_{n+1}$ the time step Δt has to be chosen smaller than the Lagrangian time scales. The random component consists in each coordinate direction (u, v, w) of a $N(0,1)$ random number r and the standard deviation σ of the particular turbulent velocity.

In the applied dispersion model [2] the following equations build the basis for calculating the advective and turbulent velocities for the particle simulation:

$$x_{n+1} = x_n + \Delta t \cdot (\bar{u}_n + u''_n) \qquad y_{n+1} = y_n + \Delta t \cdot (\bar{v}_n + v''_n) \qquad z_{n+1} = z_n + \Delta t \cdot w'_n$$

$$\bar{u}_n = |\bar{\mathbf{u}}_n| \cos(270^\circ - \alpha_n) \qquad \bar{v}_n = |\bar{\mathbf{u}}_n| \sin(270^\circ - \alpha_n) \qquad (\bar{w}_n = 0)$$

$$u''_n = u'_n \cos(270^\circ - \alpha_n) - v'_n \sin(270^\circ - \alpha_n) \qquad v''_n = u'_n \sin(270^\circ - \alpha_n) + v'_n \cos(270^\circ - \alpha_n)$$

$$u'_n = a_u\, u'_{n-1} + b_u\, \sigma_u\, r_u \qquad v'_n = a_v\, v'_{n-1} + b_v\, \sigma_v\, r_v \qquad w'_n = a_w\, w'_{n-1} + b_w\, \sigma_w\, r_w + c_w$$

$$\text{with } a_i = e^{\frac{-\Delta t}{T_i}}, \qquad b_i = \sqrt{1 - a_i^2}, \qquad i = u, v, w$$

(α_n wind direction in meteorological orientation, c_w correction term, T_i Lagrangian time scales).

2.2 Deposition

For modelling of realistic dispersion processes the simple algorithm given in the previous section has to be modified respectively extended to match the special problem. Examples are the modelling of fall–out and wash–out — outlined in this section —, the consideration of chemical processes or radioactive decay as well as different release conditions like source geometries, release temperatures or varying source intensities.

In practical applications especially the consideration of fall–out and wash–out respectively dry and wet deposition of the released substances on the ground is of some importance. Particle simulation models as described can be extended in a relative simple way to meet these requirements.

The fall-out of aerosols or dust caused by gravitation is modelled by adding a sinking velocity to each simulation particle. This gravitational settling speed v_g is mainly a function of the diameter of released aerosol or dust particles. Appropriate tables or formulas can be found in the literature [4].

During the simulation process it can happen that the computed trajectory of a particle reaches the ground $z = 0$. In this case the dry deposition of the released material has to be taken into consideration. This deposition can be described by a so-called deposition velocity v_d, which depends on a number of parameters like particle diameter, sinking velocity, atmospheric humidity and friction at the ground. As complete models for determining v_d do not exist, simplified problem-oriented attempts for v_d are in use. In the particle simulation model each particle can be supplied with a statistical weight of 1 at starting time, which is reduced by a deposition probability W_d at each ground contact of the particle. The deposited material at this ground contact point corresponds to the value of W_d. In the implemented model W_d is described by a function of v_d, v_s and $\sigma_w(0)$, where $\sigma_w(0)$ is the standard deviation of the vertical turbulent velocity on the ground.

For modelling wash-out respectively wet deposition the wash-out rate ψ as a function of precipitation (especially rain) intensity, mean raindrop size and the collision efficiency between pollutant particle and raindrop can be used. As wet deposition concerns all released particles the statistical weight of each simulation particle can be reduced after every time step Δt by the fraction $\Delta t \cdot \psi$. This fraction corresponds to the wet deposition on a ground point directly under the particle position. Other approaches for handling wash-out in a particle simulation model are also possible.

2.3 Dose Calculation

As mentioned in the introduction the fast calculation of gamma-radiation dose from a radioactive pollutant cloud is of great importance for emergency protection in case of accidental releases from a nuclear facility. The radiation exposure S at fixed points (X, Y, Z) is defined by the integral

$$S(X,Y,Z) = \int_{0}^{T} \int_{R^3} \frac{e^{-\mu r} B(\mu r)}{r^2} \, K(x,y,z) \, C(x,y,z,t) \, d(x,y,z) \, dt$$

with $r^2 = (x - X)^2 + (y - Y)^2 + (z - Z)^2$, μ a material constant, B a so-called build-up factor, K a correction factor for ground radiation and C the radionuclid concentration.

In this particle simulation model time-integrated concentrations will be determined by accumulating the dwelling times of each particle in the cells of a three-dimensional rectangular grid. Thus, the above integral reduces to a volume integral which can be solved elegantly by an adaptive method of integration [1]. The main idea of this method is the precision-dependent decomposition of the integration domain. Near a ground point the calculation is done with relativ accuracy by using spherical coordinates whereas in the remaining domain the center point formula will be applied. Another approach in calculating the gamma-radiation dose by considering each simulation particle as a single moving radiation source is outlined in [1].

3. IMPLEMENTATION

The Hahn-Meitner-Institut Berlin carries out basic research in physical and chemical areas and has for this purpose besides other ray sources a 10 MW reactor as a neutron source in operation. To guarantee the supervision of the institute and the surroundings with respect to radiation protection a new radiation control and monitoring system was installed in 1992. This system is based on VAX computers and was developed in cooperation with Landis & Gyr corporation, Switzerland/Germany. It performs continuous acquisition, processing and archiving of radiological and meteorological data and allows their presentation in various kinds of installation pictures, curves and tables. All measured data are stored in databases which can be accessed by other programs using a set of interface routines.

In addition to the monitoring system the main attention of the radiation protection department was directed to special software which computes — in case of an accidental release — the atmospheric dispersion of radioactive pollutants and several radiation doses mentioned in chapter 1. To meet the required time constraints a Lagrangian random walk model was implemented using a transputer cluster from Parsytec Ltd., Germany with up to now 29 processors T800 and the distributed operating system Helios V1.2.1. The transputer cluster is connected to the monitoring system via ethernet.

The whole dispersion software system is organized like a "double farm" which is controlled by one master task, i.e. the master controls several worker tasks for particle simulation and for integration which operate independently without exchanging data. At the time one simulation worker and one integration worker are mapped on one transputer. Though the farm concept is not well-suited for massiv-parallel systems it can be used here successfully because of the small number of available transputers. Tests on a larger transputer cluster (128 processors) have shown that the concept is limited to 60 - 70 processors for this kind of software system.

The dispersion software has a modular structure and is controlled by a number of input files. Its operation is completely integrated into the user interface of the monitoring system. To perform a special dispersion process a number of parameters has to be defined which are arranged in several control files. Providing a set of predefined control files with prognostic data as release profiles and meteorological data the simulation of different dispersion processes, e.g. for classes of accidental releases, can be performed. The access to the actual radiological and meteorological data measured by the radiation control and monitoring system also allows accompanying online dispersion calculations. Important for the particle simulation model is the availability of wind field data. Wind velocity, wind direction and turbulence data are measured by a SODAR (SOnic Detection And Ranging) system up to a height of 500 m. These data can be used for computing a diagnostic wind field for the vicinity of the institute [5]. A prognostic wind field model is not available.

4. CONCLUSION

The implementation of dispersion software based on a Lagrangian random walk model has shown that a number of phenomena connected with pollutant dispersion in the atmosphere (deposition, radioactive decay, variable source intensities, etc.) can be treated in a relative simple way. The independent calculation of particle trajectories allows an easy use of parallel computer architectures thus guaranteeing time-limits as required in radiation protection or similar applications.

5. REFERENCES

[1] Gottwald, S., Das Divide&Conquer-Prinzip für massiv-parallele numerische Algorithmen. To appear: Fachbereich Mathematik und Informatik, FU Berlin, 1993.

[2] Janicke, L., Programmsystem LASAT zur Berechnung von Aerosol-Transport-Vorgängen. Ingenieur-Büro L. Janicke, Überlingen, 1992.

[3] Martens, R., Maßmeyer, K., Pfeffer, W., Haider, G., Morlock, G., Bestandsaufnahme und Bewertung der derzeit genutzten atmosphärischen Ausbreitungsmodelle. Gesellschaft für Reaktorsicherheit, 1987.

[4] Slinn, W. G. N., Precipitation Scavenging. In: Atmospheric Science and Power Production. US Department of Energy, 1984.

[5] Steffany, F., Anwendungen eines massenkonsistenten Strömungsmodelles im "Microscale". Diplomarbeit Institut für Geophysik und Meteorologie, Uni Köln, 1991.

Ozone Analysis in the Area of Berlin by Means of Parallel Simulation

S. Unger, P.Mieth, A. Sydow
GMD - Research Institute for Computer Architecture and Software Technology
Rudower Chaussee 5
12489 Berlin
Germany

Abstract

A concept for a simulation environment for local authorities is presented consisting of the necessary data bases, a mesoscale meteorological model, an air chemistry model, and decision support tools including result visualization. It is pointed out that a set of such numerical models is very complex. Simulation runs and scenario analyses take hours of computing time, even on today's supercomputers. Therefore a strategy for model decomposition and implementation on massively parallel computers is described.

1. INTRODUCTION

Air pollution problems are of great importance in our time. Although in recent years a variety of numerical models for meteorology, air pollutant transport and air chemistry has been developed, there is a lack of simulation environments for local authorities to support their decision making activities. In order to close this gap a combination of multidisciplinary subjects such as atmospheric physics, meteorology, air chemistry, ground-based sensing, mathematical modelling and computer science is required.

The basic modules of a simulation environment for air pollution transport over conurbation areas with the aim of supporting efficiently the forecasting of smog situations, operational management and urban planning have been described in [7]. Here we want to stress the special importance of parallel computation for such simulations.

In section 2 the concept of a simulation environment for air pollution transport under consideration of urban decision making is presented. The chosen model domain is the region of Berlin. In section 3 the aspects of parallel model implementation are described. In section 4 some remarks about simulation results and verification are given.

2. SIMULATION CONCEPT

This section is only a short overview. For more detailed information see [1], [5] and [8].

The simulation environment consists of the following parts:

- assimilation of input data.

As input data serve orographic and land utilization data as well as emission data, which have to be prepared in close contact with local authorities.

- meteorological model.

As meteorological models we use the 3-layer-model REWIMET and, in future, the fully 3-dimensional model GESIMA (cf.[3] and [4]). Both models are so-called mesoscale models, which can be used for the simulation of regions of about 100*100 km extension with grid spaces of 1...5 km.

In the 3-layer model REWIMET the atmosphere is subdivided into three layers, defined according to the physical structure of the atmosphere. An explicit numerical solver with forward time steps and central space differences is used with the exception of the advection terms, where upstream differences are applied (method by

Smolarkiewicz). The *Courant* stability criterion is proven at each iteration and defines the new time step. The gradient of all variables at horizontal boundaries is set equal to zero.

In the fully 3-dimensional model GESIMA the model area is discretisized in all 3 dimensions. To solve the equation of motion a *MacCormack* scheme is used, the resulting *Poisson*-equation is solved by means of an IGCG (Idealized Generalized Conjugated Gradient) method (cf. [4]). A cloud physics and radiation model is part of GESIMA and allows to compute radiation fluxes, cloud cover, and other important characteristics for motion and pollutant transport.

- air chemistry model.

The scheme of chemical reactions leads to a system of ordinary nonlinear differential equations. Because of the huge differences in reaction kinetics such systems are extremely stiff and time steps according to the quickest reaction must be used for integration. A special algorithm has to be selected to ensure stability and reasonable numerical results. The chemical part has to be solved for every grid point in each time step. So even the fastest sequential computers need several hundred hours for a one day simulation in three dimensional space for 200x200 horizontal grid points. In any case, the most time consuming part is the computation of chemistry.

To overcome these difficulties a lot of compressed chemical reaction schemes have been developed. For our air pollution model we have chosen the CBM-4 model (Carbon Bond Mechanism [2]) to compute the near surface concentrations of ozone, peroxyacetyl nitrate (PAN) and nitrogen oxides. There is no need for the definition of an average molar weight so that this mechanism is mass balanced. Some species are handled explicitly because of their special character in the chemical system (for example isoprene which is the most emitted biogenic species). The mechanism involves 34 species and 82 reactions and contains 9 primary organic compounds. To profit from the features of the CBM-4, detailed information of the hydrocarbon mixture is necessary.

The CBM-4 has been extensively tested against more than 170 smog chamber experiments with good results. Its is proposed from the Environmental Protection Agency of the U. S. A. for usage in air pollution models [2].

Time steps, required for the integration of ordinary differential equation systems representing chemical changes, are much shorter than every other existing time scale in the dispersion equation. Thus the method of operator splitting can be applied. For an integration of the extraordinary stiff system the conventional method according *to Gear* is used in order to minimize the error and to ensure stability. Of course, there are much faster solvers (up to a factor of 5), but they cannot compete concerning to precision and stability. Usually, they do not work well for broader ranges of parameters and inputs and should only be applied with an extensive testing.

- decision support and visualization.

A decision maker in an environmental agency who is responsible for air pollution management commonly has to solve 3 kinds of problems:

- approval of industrial facilities,
- environment compatibility tests, and
- precautionary measures to prevent smog situations.

Today he has little support for these tasks. The calculations for the approval of industrial facilities and environment compatibility tests are carried out with *Gaussian* models representing the state of the art of the 1960s. Up to now measures against smog can only be taken when ground-sensing networks indicate that concentration limits are exceeded.

The final version of the described simulation environment should support the user, mainly, in 2 ways:

- by graphic visualization of simulation results for air pollution transport processes with complex atmospheric models, and
- by recommending measures derived with inference algorithms from a knowledge base of an expert system.

Besides the graphical output the simulation results are used to build up a knowledge base.

3. PARALLEL MODEL IMPLEMENTATION

Numerical models for air pollution transport linked with air chemistry have a huge demand of computing time.

Even on today's sequential supercomputers the time for scenario analysis and case studies is considerably too long. Thus, the simulation of one hour real time with the model REWIMET including air chemistry for our example Berlin (30*24 points in the horizontal plane, 3 layers in the vertical) requires about half an hour computing time on a SUN SPARC 10. More than 90% of the CPU-time is used for calculation of air chemistry and transport. In case of the simulation model GESIMA the situation is much worse. Just to compute velocities and cloud physics for the same simulation period this model needs about 10 minutes on a CRAY YMP, on a SPARC several hours. Including air chemistry, this time increases at least by a factor of 4 or 5, consequently a real time computation even on a CRAY seems to be very difficult. In addition, the mentioned factor is even larger on a CRAY because of bad vectorization performance of the *Gear* chemistry solver with a lot of recurrences and short cycles. In addition, we want to simulate not only the area of Berlin, but we also want to show the consequences of reduction measures in Berlin. Therefore we have to compute the whole surrounding region of Brandenburg, resulting in a mesh of 50*50 or more.

To overcome these problems we have used parallel computers of MIMD structure according to *Flynn's* classification. A macro package for Portable Parallel Programming using the Message Passing Programming Model (PARMACS) or the PVM (Parallel Virtual Machine) concept guarantees a high software portability. On the basis of the current PARMACS release the simulation with REWIMET runs on the following hardware platforms: Intel iPSC/2, Intel iPSC/860, nCUBE2, SUPRENUM and Meiko (both transputer and i860 versions). The GESIMA-model, in a first step, has been parallelized with PVM and runs on a network of workstations.

We here want to point out some details of the parallelization of GESIMA, concerning REWIMET cf. [1].

Vertical direction plays a particular role in atmospheric simulation. There is a close coupling of some processes in vertical direction only (e.g. radiation). Consequently, it makes no sense to decouple or decompose the model vertically because of the large amount of communication required in such a case. Therefore, GESIMA (and REWIMET too) only have been decomposed horizontally, doing the vertical operations in all points sequentially.

There is one big difference in the solution method between GESIMA and REWIMET: As a consequence of the nonhydrostatic formulation of the equation of motion, the time-discretization no longer is explicit. Thus, a *MacCormack* scheme is used resulting in a need for solving a *Poisson* equation for the pressure. This leads to a strong coupling over all points of the mesh and, therefore, a lot of communication is required. Consequently, a good balance between calculations and communication has to be found. For this reason we have used a special type of the *Crout*-factorization, proposed in GESIMA for vectorization. Factorization is done in such a way, that multiplication with upper and lower triangular matrices is decomposed in one horizontal direction. As a consequence of this recurrences in this direction are cancelled. We need, at every iteration step of the conjugated gradient method, only one global communication to compute norms and one local communication to interchange some boundary values between processors.

Now, the state of the art is the following: REWIMET runs with air chemistry on various architectures and is used for case studies and scenario analyses. Up to now GESIMA runs on a network of workstations, connected by Ethernet, under PVM without air chemistry. At the conference we hope to present first results of the winter-smog simulation (without air chemistry) and, eventually, the sommer-smog simulation (with air chemistry) by means of GESIMA.

At the moment we have achieved a speedup of more than 3 for a network of 4 workstations. But, if more workstations are used, the speedup rapidly decreases because of the slow communication. Presently we are porting the model to the MANNA-system, a MIMD-computer developed in our institute [6]. Results are presented on the conference.

4. Simulation results

To verify the computational results it primarily is necessary to prove the meteorological and dispersion part of the model without chemistry in order to be sure that the model describes well the transport of an inert species. It should be mentioned that there always are difficulties to compare measured and computed data. Experimentally estimated wind velocities or concentrations are point measurements, but numerical data are spatially averaged over the model box volume. Point measurements can be influenced very strongly by local properties, whereas box averages do not realize these microscale effects.

We have selected sulfur dioxide as indicator for an inert species. At the same time, the model is able to run as an estimation tool for the computation of near surface concentrations during a winter smog episode. The data for sulfur dioxide for industry and households are quite well documented and can be found in nearly every local environmental office. The selected smog episode has been in winter '89 during a high pressure period. In case of relatively strong geostrophic wind the transport of a substance is dominated by advection.

Good correspondence between the computed wind field and measurements has been achieved during a 15 hour simulation period. Wind field computation under normal high pressure conditions is represented sufficiently by the model.

The full model consists of meteorological, transport and chemical modules and has been implemented . A lot of work has been concentrated on obtaining enough data in the necessary resolution and with realistic daily duration.

Simulation results can be visualized as coloured maps of the concentrations over the simulation area by means of an MOTIF environment. A lot of scenario analyses and case studies for various reduction measures have been done by means of the model REWIMET and are presented on the conference together with first results of the model GESIMA.

Our simulation package coupled with an actual emission data base and meteorological input from the synoptic weather forecasting service may help to find strategies for ozone management. Parallelism in computing is an essential feature to obtain reasonable computation times, hoping, that in future such models may be used to forecast critical concentrations of photochemical oxidants and to develop a management to prevent it.

5. REFERENCES

[1] Gerlach, J., Schmidt, M., Parallele Implementierung von Modellen für die Luftschadstoffanalyse. In: Sydow, A. (ed.), Proc. of the 8. Symposium Simulationstechnik, Berlin, September 1993, Vieweg 1993, pp. 453-457.

[2] Gery, M. W., Whitten, G. Z., Killus, J. P., Development and Testing of the CBM-4 for Urban and Regional Modeling. EPA-600/3-88-012, U.S. EPA, 1988.

[3] Heimann, D., Ein Dreischichten-Modell zur Berechnung mesoskaliger Wind- und Immissionsfelder über komplexem Gelände. Dissertation, University Munich, 1985.

[4] Kapitza, H., Das dynamische Gerüst eines nicht-hydrostatischen Mesoskalen-Modells der atmosphärischen Zirkulation. Dissertation, GKSS-Forschungszentrum Geesthacht GmbH, Geesthacht 1987.

[5] Mieth, P., Unger, S., Ozonanalyse im Berliner Raum mittels paralleler Simulation. In: Sydow, A. (ed.), Proc. of the 8. Symposium Simulationstechnik, Berlin, September 1993, Vieweg 1993, pp. 471-474.

[6] Montenegro, S., Massiv parallele Architektur für nicht-numerische Anwendungen. GMD-Spiegel, Heft 3/4, 1992, Sankt Augustin.

[7] Sydow, A., Schmidt, M., Unger, S., Lux, T., Mieth,P. and Schäfer,R.-P., Parallel Simulation of Air Pollution Transport for Urban Decision Making. In: Proc.of the 12th World Congress of IFAC, Australia 1993, Vol. 3, pp. 419-422.

[8] Sydow, A., Lux, Th., Sadykowa, L., Schäfer, R.-P., Konzepte für die parallele Simulation der Ausbreitung und chemischen Umwandlung luftgetragener Schadstoffe. In: Sydow,A. (ed.), Proc. of the 8. Symposium Simulationstechnik, Berlin, September 1993, Vieweg 1993, pp. 491-494.

Modelling and Simulation of Processes in the Groundwater Zone

Peter-Wolfgang Gräber
University of Technology Dresden, Institute of Groundwater Management
Mommsenstraße 13, D-01062 Dresden, BRD

Abstract **To control and monitor pumping wells in water works, drainage systems in mines and building pits, deposits and industrial and agricultural contaminations, it is necessary to know the processes in the soil and groundwater zone. There are physical or chemical, and biological processes. It is possible to describe these processes with mathematical models, using the heat-conductivity-equation and the convection-diffusion-equation. The simulation and visualisation of the processes is necessary and helpful making decisions for operating the system. The use of parallel computers architecture, super computer/vector processor as well as transputer networks is more effective then the use at serial computers. The compute time, the number and speed of iterations, the solution stability and the discretisation error improve, using these systems.**

1. INTRODUCTION

In the soil and groundwater zone, physical or chemical, and biological processes take place. It is necessary to control and monitor these processes at

- **pumping wells in water works,**
- **drainage systems in mines and building pits,**
- **sanitation of deposits and industrial as well as agricultural contaminations.**

The simulation and visualisation is necessary and helpful when making a decision to optimise the operating strategy.

The complicated properties of the models are the following:

- complexity and strong non linearity of **Large Systems,**
- bad condition and strong different time constant of, **Stiff Systems,**
- inaccurate destination of parameters of **Fuzzy Systems.**

The technology of parallel information processing based on digital computers, has been developed competitively during the last few years to sequential computers. The impulse for the research come from necessity to increase the computer speed.

Already since 1979, the advantage of computers based parallel information processing for the simulation of processes in the soil and groundwater zone has been discussed by GRÄBER. Based on positive experience and on the new possibility of technology, the simulation has been analysed and adapted to current developments of devices and algorithms.

2. HYDROGEOLOGICAL FOUNDATION

To describe the dynamic flow and the coupled energy- and mass transport processes as well as the energy and mass conversion in the soil and groundwater zone the following equations are used:

- the dynamic basis equation of the mass flow $\vec{v} = k \text{ grad } h,$ (1)
- the mass balance equation $\text{div } \vec{v} = S \frac{\delta h}{\delta t} - w,$ (2)
- the boundary conditions.

For each phase (air, water and soil) and for each migrant the following equations:

- the dynamics base equation
  - transport by dispersion $\vec{g_1} = \vec{\vec{D}} \text{ grad } P,$ (3)
  - transport by convection $\vec{g_2} = \vec{v}\, P\,,$ (4)
- the balance equation $\text{div } \vec{g} = (n_0 + \alpha) \frac{\delta P}{\delta t} - w_g,$ (5)

- the boundary conditions.

In addition to these basic equations, chemical reactions (mass conversion) and biological processes might to be considered in sink/source terms.

The mathematical model consists of a system of ordinary and partial differential equations (PDE) and algebraic equations, whose coefficients are usually a function of space, time and the potential. Therefore the system is non-linear and variant in time and space. The combination of the equations results in two non-linear second order partial differential equations:

- the head-conduction-equation (parabolic PDE) $\text{div } (T \text{ grad } h) = S \frac{\delta h}{\delta t} - w,$ (6)
- the convection-diffusion-equation (hyperbolic PDE) $\text{div } (\vec{\vec{D}} \text{ grad } P - \vec{v}\, P) = (n_0 + \alpha) \frac{\delta P}{\delta t} - w_g.$ (7)

The coupling of the mass and quality flow can be done, using the properties of the water quality (e.g. temperature, mass concentration, kinematical viscosity and density) and the properties of the groundwater flow (speed, storage change, internal sinks and sources).

In these equations are:

$\vec{v}$	speed,	$\vec{\vec{D}}$	dispersion,
k	conductivity coefficient,	P	quality potential,
h	water level,	α	sorption coefficient,
S	storage coefficient,	w, w_g	source / drain intensity.
$\vec{g_1}, \vec{g_2}, \vec{g}$	specific quality flow,		

3. MOTIVATION FOR PARALLELISMEN

For the processes in the soil and groundwater zone large time constant and high costs per measurement point, and therefore low density of measurement points in the nature, as well as possible irreversible effects of each operations in the system are characteristics. A CAE-system can help the hydrologist to estimate the influence of the ecology system.

The parallelisation of the necessary simulation by using transputer, will at first, accelerate the mathematical algorithm and therefore shorten the response time. At second, the real processes in the nature which are parallel, are better reflected by parallel modelling. Under such conditions simplified and more stabile algorithms originate. The different parts of these algorithms are divided into coupled sequential processes (similar to CSP-model of HORARE).

The use of transputer networks requires another solution of the simulation, then the super computer. The solution with super computer was a closed task. At the strong parallelity it is possible to used several closed tasks, which work parallel at single transputers. Two ways are possible, the partitions of space and the partitions of processes.

The partitions of space

The basis algorithm of this method is, the simulation area is divided into several parts and these parts are simulated by traditional methods for example the finite-difference-(FDM), finite-elements-(FEM), or the boundary-elements-method (BEM). The main problem of this method is the data exchange between the partitions. Between the transputer as informationslines with interface need to build as lines of the discretisation grid are cut. Using FDM, many lines at the discretisation grid need to be cut, due to its properties.

Using FEM this can be minimisation, by arranging large meshes the border of each part. Using BEM the data exchange is zero along the boundary elements. Therefore, no data exchange between the transputers is necessary. If it is possible, to divide the area on aquipotential line or a first order boundary condition, the necessary effort is also minimal for FDM or FEM.

The partition of processes

The partition of processes goes out from the real processes in the nature, which are parallel. The nature´s processes are reflected as a task of the transputer. For the processes in the soil and groundwater zone this mean, the

- processes of the quality and mass flow,
- the different phases (air, water, soil),
- the different migrants
- biological processes and so on

are parallel. The data exchange between the single processes is minimal in comparison with the partition of area. Each partial differential equation is realised by a another compute part, for example another transputer.

4. RESOLUTIONS METHOD

For the acceleration of the solution of the problem there are two principle ways. The existing sequential algorithm can work at a convential, but extremely fast, "von-Neumann"-architecture. The technology of such computer sets limits for the acceleration, doe to the clock rate and the memory access time, which can be increased only limited and with high technical efforts. But then the cost of the hardware grows exponentially with the increasing of speed. Only real parallel working systems can significantly increase the rate of data processing. These systems require a complete reworking of the algorithms. Under specific circumstances totally new resolution methods are necessary. In our investigations for the use of supercomputers for the simulation of groundwater flow, transputer networks and vector computers are used. Transputer systems offer the advantage of a scalable, relative inexpensive productivity increase.

They are suitable for the construction of a CAE-system, due to the possibility to use these system directly on the working place (using PC-plug-in-unit or auxiliary device). The disadvantage of such systems are communications and synchronisation problems, the same as on read distributed and parallel systems. It is inevitable to redesign all used algorithm.. The existence of distributed memory involves a large communication effort pre or after the central computation part.

The vector computer is, in contrary to a transputer system, a sequential working device, however specialised in the processing of large data streams. It is characterised by vector registers, stream buffers, cache memory and a pipeline architecture. According to the type, sequential or vectored processing can realised also different processors. With it a real parallelity of both processors can be obtained. An essential advantage of that system is,, that the complete redesign of the used algorithms is not necessary. Only an optimisation of the programs regarding the vectorisation is necessary. The vector computers remain transparently for the programmer in essential. So far such computers are available only as main frame computers in computer centres.

The distribution of the problems at the transputer network can be done in two ways. For the mass flow, which can be described by simple matrix equations of the FE- or FD-method, the partitions of space can be used. The total field is divided in single partitions, which are calculated on a computation node. The necessary data exchange represent a special problem, which depends on the discretisation method. For the simulation of migrations processes, which are described by the two coupled non-linear partial differential equations, the partition of processes can be used. The efforts of the data exchange is smaller compared with the space partitioning. The coupling of the equation is realised by communication via data channels.

Using a Windows environment, where service and visualisation function are already available, the user can communicate with the transputer network, using a special file server, as well as with a normal PC. Additional it is planed to implement a file interface to the vector computer VP 200 EX in the compute centre of University of Technology Dresden.

While we test the algorithm at a Supercluster-256 from PARSYTEC at the Rheinisch-Westfälische-Hochschule Aachen, the final system will consist of a PC with a Multi-Transputer-PC-plug-in-unit with 4 T800-Processors from INMOS Coop. and a link switch unit C004 also from INMOS. Such a system can be used easier in practical applications.

5. REALISATION

So far the algorithm to simulate the mass flow is implemented on a quadratic transputer network. The program is written in the language 3L-C. The operating system HELIOS has not been used intentionally, because the simulation of this problem uses only a small part of the services offered by HELIOS. The advantage of the chosen language is the existent debug modus and the large number of library tools for parallel of processing. At present we are working on the partitioning for the INMOS-ANSI-C-Toolset. The software technology is similarly for both environments. In addition to the normal programming steps like compiling and linking, at transputer networks it is necessary the known produce a special loadable file. In this file the run time code is stored according how to the partitions of the transputer network. This file also contains, all hardware and software descriptions and there links. Additionally it necessary to produced the hardware links in a approximate way. Normally these takes place with a link switch unit for which also a configuration file must be created (HELIOS resource maps, OCCAM programs, and so on).At present the hardware is a Supercluster-256 from PARSYTEC, at the Rheinisch-Westfälische-Technische-Hochschule Aachen. During the work as the simulation of the mass flow experiences has been made. There with be discussed in the following.At distributed applications it is usefully to divide the working processes at each node in computation processes and in a message handling process. With that the algorithm can be freed of all the message exchange processing. So the coding is simplified and the reading and the services is easier. Furthermore such a message handler can be used for other applications. If message exchange becomes to complex, it could be better to use a operating system. The precision of a parallel program is shown not only in the static precision but also in it is dynamic accuracy, so called liveliness.

Each structure of parallel working, interconnected, relative independent programs, tends to a loss of liveliness, which also called as jamming. That show up for example in the construction of communication loops. In such case several processes of the ring transmit a message, but no processes can received the message, doe to the unbuffered communication. A loss of liveliness can be prevented, when a central controlling for the communication is introduced. Also the introduction of buffers and the producing of additional parallel processes as sub processes of the message handling can prevent this effect. In our implementation we decided to use the latter variant, because the performance does not decrease and the concept of channel communication is not violated.

The used message handler can only received messages. If a message needs to be transmitted, an additional process is started, working parallel in the same memory and the same transputer, as the message handler. This process then transmits and terminates the message.

On large problems and such with an intensive data exchange the computation time depends in essentially on the efficiency of the communication. Figure 4 shows this fact. On our implementation the relation between the total runtime and the time of the computation task is measure using different numbers of transputer, which are placed in a quadratic network. A simple source/sink field problem is the bases of these measurements. It can be seen clearly, that the relation does not go over 40 per cent and decreases with an increasing number of grid points and transputer nodes. This behaviour is based on the increase of communication with larger or finer distributed problems.

The present work on these problems shows, that this way for acceleration of the simulation is right and sensible. Additional work must be done toward the apply caution of more effective equation solvers for the partitions of space and the accelerations of the communication between the transputer. Also the work on the partitions of space needs to be increased. For practical usage a coupling with the corresponding user environment must be realised.

6. CONCLUSION

The simulation of processes of the soil and groundwater zone by means of parallel computer architecture, like super computer/vector processor and the transputer networks are very effective. The computation speed and the stability increase while the number of iteration and the discretisation error decrease. In the future, the acceptance of the developed simulation programs for the solution of water management projects must increase. Also it is necessary to do additional theoretical research on these problems. The most important problems are of increasing of both the effectiveness, and the parallelity of the programs, and the use of a new transputer generation.

Hierarchical Modeling Approach for Smog Management

E. SZCZERBICKI and A. SYDOW

GMD FIRST, Rudower Chaussee 5 (13.7), 12489 Berlin, Germany

Abstract. Simulation environment for parallel implementation platform to predict and manage smog situations is currently under development. The environment includes the necessary data bases, a mesoscale meteorological model, and air chemistry model. This presentation outlines the concept of employing in the framework of smog management simulation environment the decision support tools based on hierarchical distributed modeling.

1. INTRODUCTION

Smog management system is a set of tools that support the forecasting of smog situations, the operational management of an environmental agency, and the urban planning. It consists of two basic mechanisms of support: modeling support and decision-making support. Both mechanisms are of hierarchical and distributed character and are planned to be implemented in smog simulation environment that is currently under development at GMD FIRST, Berlin [5,6].

Decision-making support is based on problem-solving agents that are able to capture the nature of environmental agency functioning. The approach outlined in the next Section has been successfully implemented in different domains with complex, hierarchical modeling and decision-making characteristics. It is proposed to deal with the similar complexity of smog management related decisions.

2. DECISION MAKING AGENTS

Because of the increasing importance and complexity of smog related problems tools are needed to manage the decision making processes in environmental agencies to maximize their performance and effectiveness. The tools should support the agency in reaching the following goals: (i) how to decompose complex smog related problems and distribute responsibilities and tasks among a number of problem-solving agents, (ii) how to coordinate these agents to solve problems cooperatively, and (iii) how to measure the effectiveness of the agency in meeting its goals. Distributed Artificial Intelligence research, that is concerned with the study and development of computerized problem solvers and decision-makers can provide such tools [1]. The realization that traditional sequential approaches cannot deal with complex decision making problems (and smog management is such a problem) has led to increasing research in distributed and hierarchical systems. This Section describes the experimental hierarchical architecture that is planned to facilitate the decision making area of smog management related problems. Such an architecture should support the basic multi-agent planning actions of task decomposition, task distribution, and result integration, and allow control of agents in the environmental agency to be centralized, partially centralized, or completely distributed. Smog management is a problem with distributed hierarchical structure typical of most cooperative problem-solving and decision-making situations. The operations of an environmental agency involve interactions between many classes of agents, that can be divided into two categories: a kernel of agents that provide fundamental problem-solving activities (e.g. decision-makers in an agency), and a group of peripheral agents that provide ancillary services (e.g. other agencies, urban planning, industrial planning). The organization of these agents may be partially hierarchical but also partially linear. The focus of the proposed research in the nearest future is on the design on control regimes and knowledge representations that allow agents to reason about local environmental activities and interact with other agents coordinating global activities of an environmental agency. Agents will be implemented as two-element entities as depicted in Figure 1.

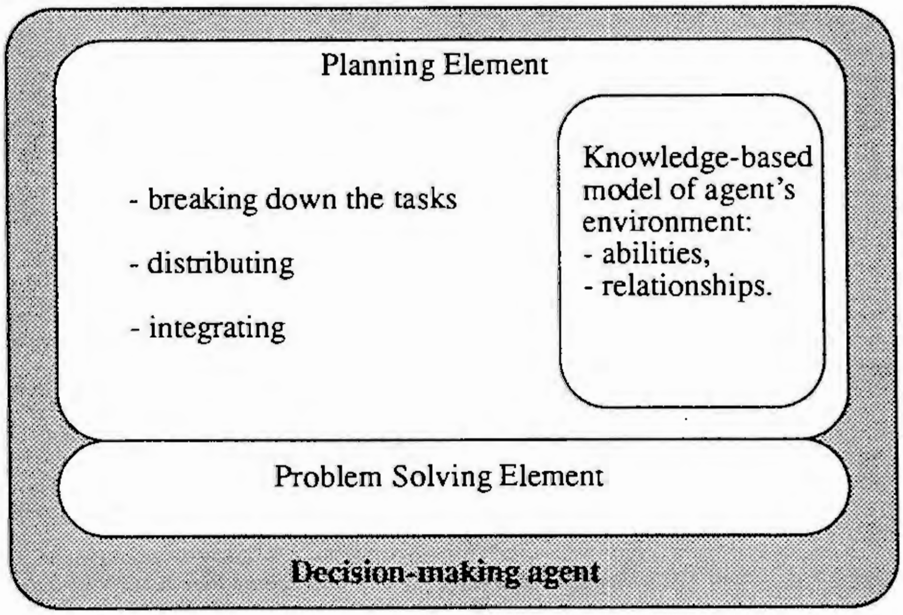


Figure 1. Decision-making agents implementation

The problem-solving element addresses the tasks that are assigned to the agent. It contains the knowledge necessary to perform the tasks required of the agent, the inference mechanism necessary to represent that knowledge, and the interfaces necessary for the agent to interact with the outside world. The planning element includes a knowledge-based model of the environment in which the agent operates, i.e. agent own abilities, abilities of other agents in the environment, relationships between agents. The problem-solving element performs tasks that the agent is able to solve itself. The planning element acts as an intelligent coordination interface that determines how tasks the agent is to perform may be broken down, distributed, and integrated. Agents may be grouped into subsystems, and each agent may be a member of many groups simultaneously as depicted in Figure 2. Communication between agents may be of direct or indirect nature. Agents may also be related to one another through lines of authority. Authority is used in a distributed problem solving environments to assist in negotiations between agents with opposed interests. The main issue that is to be addressed while developing the proposed approach for smog related problem solving is the determination of knowledge sources within agents and knowledge representation. The importance of this is stressed by the fact that any environmental agency will have to cope with situations in which much of the knowledge will be incomplete and possibly in conflict with other agents within the agency.

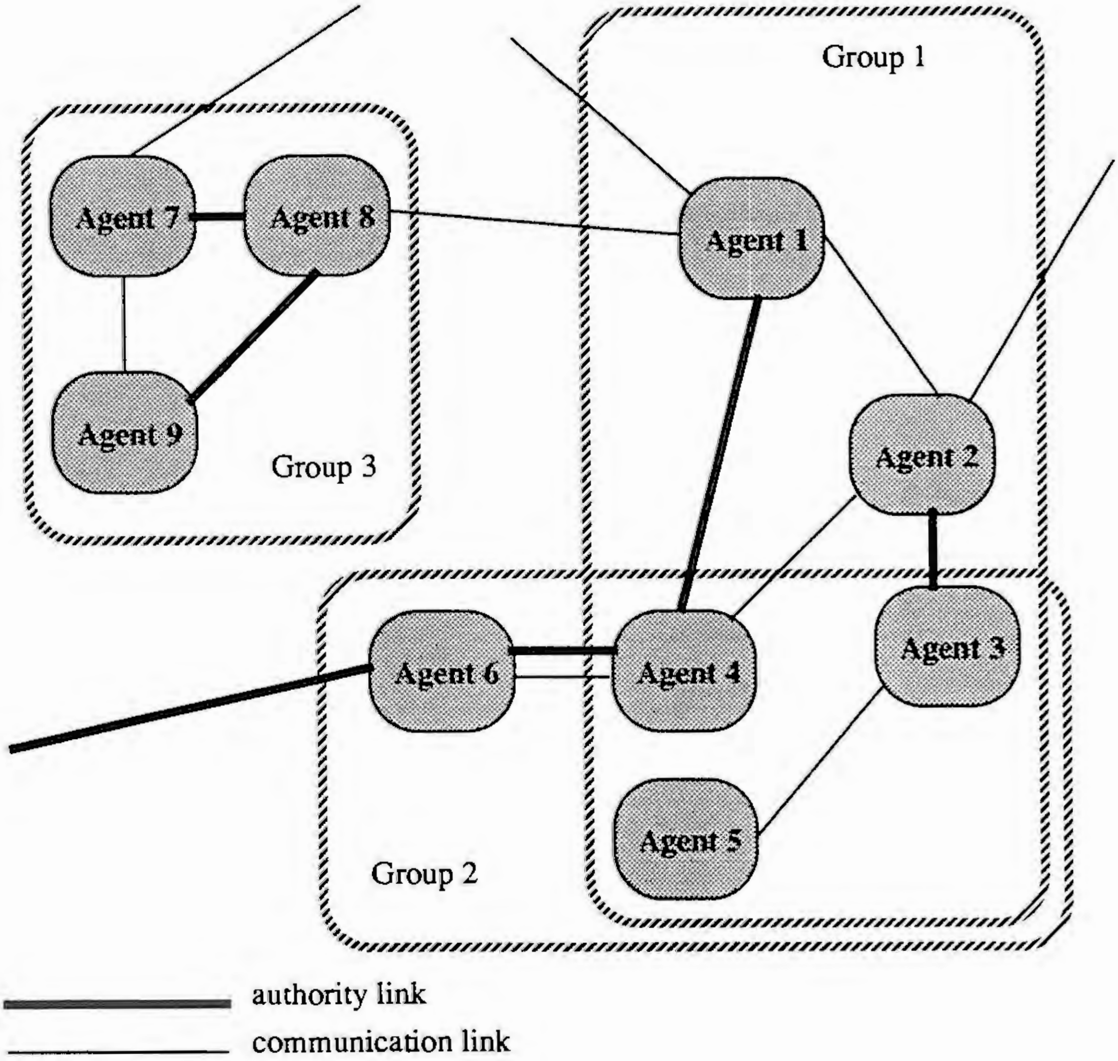


Figure 2. Agents in a multi-agent problem solving environment

Each agent should posses knowledge describing its view of the functioning of the environmental agency. This knowledge may be represented by four different types of knowledge sources within each agent. First, urban planning knowledge consists of skeletal plans which prescribe ways of decomposing problems faced by the agency and coordinating the integration of results. Second, task knowledge describes agents (or groups of agents) in the agency that are capable of carrying out tasks specified within plans. Third, agent knowledge describes rules to be used while interacting with other agents when distributing tasks. Last, coordination knowledge describes an agent role within the agency, its authority over other agents, and allocation of information. The consideration of environmental issues (e.g. smog emission and transportation) in urban planning very often leads to conflicts and informational inconsistencies. In such situations the knowledge representation as sets of constraints and corresponding constraint relaxations is recommended [2,3]. The relaxations can be applied (through negotiations) when conflicts arise. For example agent knowledge sources can include constraints such as the data on simulation of smog transportation including specified pollutants that is needed to perform a given task or limitations inherent in an agent ability to perform a task. Coordination regimes should be defined in the proposed architecture to allow each agent to reason about local and global planning by selecting applicable knowledge sources and by satisfying relevant constraints. In conflict situations agents should interact with other agents to redo problem decompositions, task descriptions and distributions, and the integration of results. The representation of a distributed problem-solving activities within an environmental agency in terms of constraints may yield, if successfully implemented, many advantages. For example, constraint directed representation makes knowledge easy to understand, organize, and manipulate. Constraint relaxations make for easy selection of alternatives in situations when constraints can not be satisfied.

3. CONCLUSION

In this presentation a framework of the decision making support for smog management was outlined. The proposed methodology, if successfully applied, may result in an architecture supporting the analyst in a smog management system development including decision making process. To accomplish this the topographical, meteorological, emission, and chemical knowledge should be integrated within the environmental agency with urban planning knowledge. It can be clearly seen that the proposed architecture takes into consideration the following assumptions (i) distributed representation of gents with various levels of detail is suitable for modeling smog related decision-making problems, and (ii) knowledge representation within a problem-solving agent of an environmental agency in terms of constraints and relaxations is suitable for planning, decomposition, and execution of tasks of the agency. The proposed modeling approach has been successfully applied in different domains and is suitable for computer implementation especially in an object oriented computing environments. It uses hierarchical structures introduced in [4], the model base concepts proposed in [7], and knowledge representation proposed in [3].

4. REFERENCES

[1] Bond, A., Gasser, L. (1988), Readings in Distributed Artificial Intelligence, Morgan Kaufmann: San Mateo.

[2] Evans, M., Anderson, J. (1990), An analysis of constraints for multi-agent problem solving, Canadian Artificial Intelligence Conference (CSCSI-90), Ottawa, Ontario.

[3] Evans, M., Anderson, J. (1990), Constraint-directed intelligent control in multi-agent problem solving, in Zeigler, B., Rozenblit, J. (Eds) AI, Simulation and Planning in High Autonomy Systems, IEEE Computer Society Press: Los Alamitos.

[4] Szczerbicki, E. (1993), Rule-based integration of autonomous multi-agent systems, International Journal of Systems Science, Vol. 23, No. 3.

[5] Sydow, A., Schmidt, M., Lux, Th., Schafer, R.-P., Mieth, P. (1992), Air pollution modelling and simulation (an approach of parallel simulation), EUROSIM Simulation Congress, Capri.

[6] Sydow, A., Schmidt, M., Unger, S., Lux, Th., Mieth, P., Schafer, R.-P. (1993), Parallel simulation of air pollution transport for urban decision making, IFAC World Congress, Sydney.

[7] Zeigler, B.P. (1984), Multifacetted modelling and discrete event simulation, Academic Press: New York.

An Improved Model of a Propeller Aircraft

W. Dunkel
Technical University of Braunschweig
Institute for Flight Guidance and Control
Rebenring 18, D-38106 Braunschweig

1. INTRODUCTION

To develop models for flight simulators and numerical flight simulations [4] as well as for the design of automatic flight control systems (e.g., autopilots) [2] a sufficient knowledge of the process is essential. Often a significant mismatch is encountered between flight test data and computer simulations based on theoretical models [1] of the aircraft motion. This mismatch may be caused by systematic measurement errors or by inadequate modeling of the aircraft and/or of the measurement system. The system identification approach is a powerful tool for the detection and correction of these errors [6,7]. For the identification of nonlinear aircraft models, flight-test data from a twin engine research aircraft DORNIER DO128 are used.

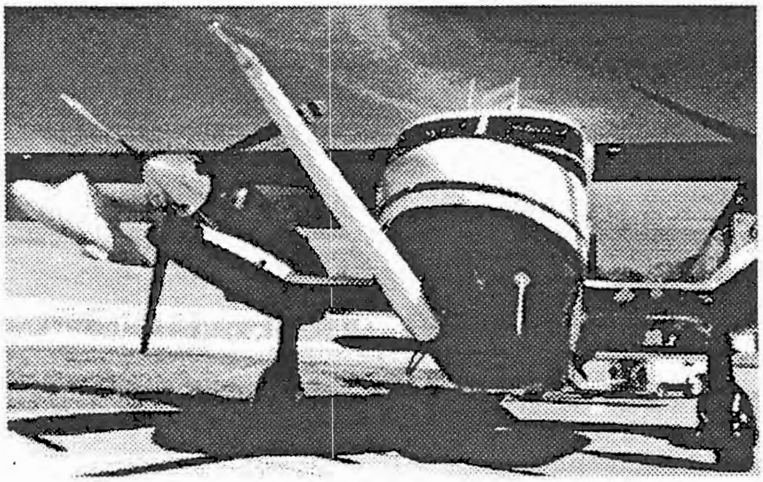


Fig. 1: The research aircraft DORNIER DO 128 - 6

2. IDENTIFICATION

The off-line identification is preferably done in two main steps according to the "Estimation Before Modeling" technique [7] which uses a Maximum-Likelihood identification. In a first step the well-known differential equations of the kinematic airplane model are used to estimate the unknown parameters of the measurement model and of the quasi static wind model. In the second step the propulsion system model and aerodynamic model are used to estimate the unknown aircraft-specific parameters. Advanced sensor systems allow to improve both parts of the model.

2.1. Expansion of the kinematic model

The measurement check as the first step of the identification makes use of a new satellite navigation system (Global Positioning System, GPS) with its output values latitude, longitude and altitude. In differential mode (DGPS) positions are determined with an uncertainty of less than one meter [5, 8]. If the model is extended accordingly this DGPS-data can be used to improve the accuracy of the measurement check (e.g., bias estimation of the accelerometers). To obtain the required accuracy of the identification model the equations in [7] have to be expanded to the description of the earth as a rotating ellipsoid.

Now the inertial system is the WGS 84 [9]. Again all measurements are assumed to be affected by bias, scaling factors and time shifts. These parameters are estimated by integrating the measured body-fixed angular velocities and accelerations and comparing the estimated and measured output quantities (Φ, Θ, Ψ, χ, V_{GS}, $\dot{h}$, h, φ, λ, V_A, α, β) as described below.

The measured angular velocity $\underline{\Omega}_b^{ib}$ (between inertial- and body-fixed coordinate system: index superscript ib, measured in body-fixed coordinates: index subscript b) is measured by laser gyros. This velocity includes the angular velocity between the inertial system and the earth-fixed coordinate system $\underline{\Omega}_g^{ie}$ and the angular velocity between earth-fixed and geodetic coordinate system $\underline{\Omega}_g^{eg}$. The transformation matrix $\underline{M}_{bg}$ from geodetic to body-fixed coordinates (a function of the Euler angles) leads to $\underline{\Omega}_b^{gb}$ and to the equation

$$\partial \underline{\Phi} / \partial t \; = \; \underline{\Omega}_b^{gb} \; = \; \underline{\Omega}_b^{ib} - \underline{M}_{bg} \cdot (\underline{\Omega}_g^{ie} + \underline{\Omega}_g^{eg}) \tag{1}$$

with

$$\underline{\Omega}_g^{ie} = \left[\Omega_e \cdot \cos\varphi \, , \; 0 \, , - \Omega_e \cdot \sin\varphi \; \right]_g^T \tag{2}$$

$\Omega_e \; = \; 7292115 \cdot 10^{-11} \text{rad s}^{-1}$; [9] is the angular velocity of the earth

$$\underline{\Omega}_g^{eg} = \left[\dot{\lambda} \cdot \cos\varphi \, , \; -\dot{\varphi} \, , \; -\dot{\lambda} \cdot \sin\varphi \right]_g^T \tag{3}$$

for determining the vector of the Euler angles $\underline{\Phi}$ (roll angle Φ, pitch angle Θ, yaw angle Ψ)

$$\underline{\Phi} \; = \; \left[\Phi \, , \; \Theta \, , \; \Psi \right]_b^T \quad . \tag{4}$$

The acceleration $\underline{a}_b^{ib}$ which can be measured in the aircraft's center of gravity includes the acceleration from earth mass attraction $\underline{G}_g$, centripetal and Coriolis acceleration. Therefore the transport acceleration $\partial \underline{V}_{Kg} / \partial t$ can be written as

$$\partial \underline{V}_{Kg} / \partial t \; = \; \underline{M}_{gb} \cdot \underline{a}_b^{ib} + \underline{G}_g - \underline{\Omega}_g^{ie} \times (\underline{\Omega}_g^{ie} \times \underline{r}_g) - 2 \underline{\Omega}_g^{ie} \times \underline{V}_{Kg} - \underline{\Omega}_g^{eg} \times \underline{V}_{Kg} \tag{5}$$

with

$$\underline{G}_g = \left[0, \; 0, \; \mu \cdot (R_M + h)^{-2} \right]_g^T \tag{6}$$

$\mu \; = \; 3986001.5 \cdot 10^8 \, m^3 s^{-2}$; [9] is the earth's gravitational constant

$$\underline{r}_g \; = \; \left[0, \; 0, \; (R_M + h) \right]_g^T \tag{7}$$

which leads to the transport velocity in cartesian coordinates

$$\underline{V}_{Kg} = \left[u_K, v_K, w_K \right]_g^T \tag{8}$$

and further to the ground speed V_{GS} and the true track χ

$$V_{GS} \; = \; ({u_{Kg}}^2 + {v_{Kg}}^2)^{1/2} \tag{9}$$

$$\chi \; = \; \arcsin (v_{Kg} \cdot ({u_{Kg}}^2 + {v_{Kg}}^2)^{-1/2}) \; . \tag{10}$$

If the acceleration cannot be measured in the aircraft's center of gravity the distance from the center of gravity to the accelerometers $\underline{X}_b^m$ must be taken into account as follows

$$\underline{a}_b^{ib} \; = \; \underline{a}_b^m \; - \; \underline{\Omega}_b^{gb} \times \left[\; \dot{\underline{X}}_b^m \; + \; \underline{\Omega}_b^{gb} \times \underline{X}_b^m \; \right] - \dot{\underline{\Omega}}_b^{gb} \times \underline{X}_b^m \quad . \tag{11}$$

Normally the variation of the distance e.g. by fuel consumption is negligible $\dot{\underline{X}}_b^m \approx 0$.

The position of the aircraft in earth-fixed coordinates (latitude φ, longitude λ, altitude h) can be obtained by

$$\partial\varphi / \partial t \; = \; u_{Kg} \cdot (R_M + h)^{-1} \tag{12}$$

$$\partial\lambda / \partial t \; = \; v_{Kg} \cdot ((R_P + h) \cdot \cos\varphi)^{-1} \tag{13}$$

$$\partial h / \partial t \; = \; - w_{Kg} \quad . \tag{14}$$

The up to now unknown quantities in the equations above are the local earth's meridian radius of curvature R_M and the local earth's transverse radius of curvature R_P

$$R_M \; = \; a \cdot (1 - e^2) \cdot (1 - e^2 \sin^2\varphi)^{-3/2} \tag{15}$$

$$R_P \; = \; a \cdot (1 - e^2 \sin^2\varphi)^{-1/2} \tag{16}$$

with the earth's semimajor axis a (= 6378145 m) and the first eccentricity of the earth e (= 0.0818191908426) [9].

To demonstrate the effect of the expanded kinematic model the old model (with the earth described as a flat non-rotating system) and the model described here (the earth is a rotating ellipsoid) are compared. A simulation result of the determined earth-fixed positions of both models can be found in Fig. 2. The simulation uses a DO128 aircraft model without sensor errors and without wind. As an input signal an aileron doublet with an amplitude of 5 degrees and a duration of 2 seconds is used. This results in a total position error of 9.25 m after 100 seconds and of already 618.51 m after 200 seconds.

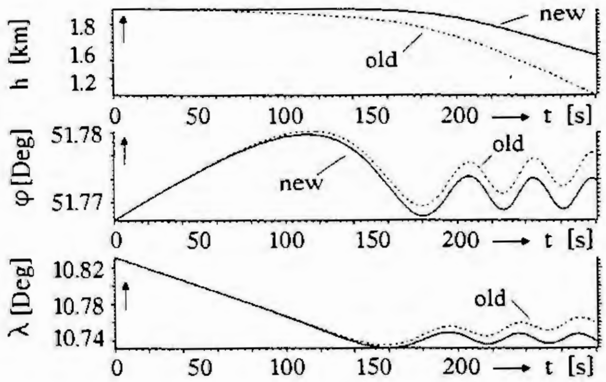


Fig. 2: Comparison of new and old model

2.2. Parameter identification of the aerodynamic and propeller model

So far the major problem of the second step of the identification is the lack of experimental data to validate the models of the coupled aerodynamic and propeller processes in a normal aircraft. It is merely possible to measure some input signals as there are the control deflections of the elevator η, the rudder ζ, the aileron ξ and the shaft power P of the engine as well as the aircraft velocity $\underline{V}_A$ (airspeed V_A , angle of attack α, angle of sideslip β). The forces $\underline{R}$ and moments $\underline{Q}$ of aerodynamic and propulsion system are not available through direct measurement. The closest measurable output quantities are the accelerations $\underline{a}$ and the angular velocities $\underline{\Omega}_K$ (see 2.1.), which are state variables of the kinematic model. There are no directly measurable aerodynamic and propeller output values to strictly separate the effects of both systems. To validate this part of the aircraft model it is advantageous to measure additional quantities in the research aircraft like the pressure behind the propeller (output value of the propulsion system) and the wing pressure (variable of the aerodynamic; a function of C_L). This paper will concentrate on the additional measurement of the pressure/airspeed behind the propeller by a Pitot tube.

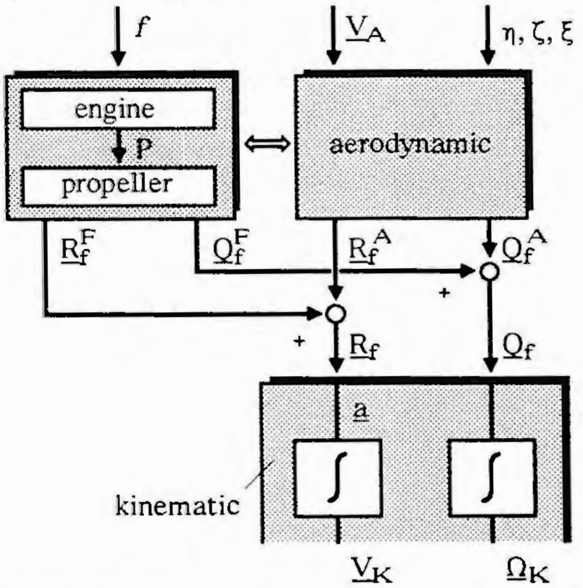


Fig. 3: Block diagram of the aircraft model

The connection to the equations in chapter 2.1. is given by the equation

$$\partial \underline{V}_{Kg}/\partial t = \underline{M}_{gb}\cdot m^{-1}\cdot(\underline{R}_b^A+\underline{R}_b^F) \qquad (17)$$

with the aerodynamic force (drag coefficient C_D, side force coefficient C_Q, lift coefficient C_L /see [3] / and the wing area S)

$$\underline{R}_b^A = \underline{M}_{ba}\cdot(0.5\cdot\rho\cdot V_{Ab}^2)\cdot S\cdot\left[-C_D, C_Q, -C_L\right]_a^T \qquad (18)$$

and the propeller force (σ is the angle between the x-axis of the airplane and of the engine)

$$\underline{R}_b^F = (P\cdot V_{Ab}^{-1})\cdot\eta_{prop}\cdot\left[\cos\sigma,\ 0,\ -\sin\sigma\right]^T \ . \qquad (19)$$

In this equation the term $(P\cdot V_{Ab}^{-1})\cdot\eta_{prop}$ represents the thrust F. The propeller efficiency η_{prop} is given by the propeller charts of the manufacturer as a function of airspeed, air density and speed of the propeller. The deterioration of the propeller efficiency caused by the installation of the propeller (e.g. by drag in the propeller flow) is neglected.

Model Extensions for describing Properties of a Flight Test Measurement System

H. Göllinger
Technical University of Braunschweig
Institute for Flight Guidance and Control
Rebenring 18, D-38106 Braunschweig

Abstract

In this paper model extensions are treated that have to be made to cope with the properties of the measurement system aboard the Dornier DO 128 research aircraft of the Institute of Flight Guidance and Control. The sensors which measure the vertical motion of the aircraft have different time delays due to data preprocessing. In order to obtain consistent on-line data that could be used for analytical redundancy purposes and on-line fault detection, a continuous-discrete Kalman filter and Padé-approximants of the time delays are used. An adaption scheme is developped that improves accuracy.

1. MEASUREMENT OF THE VERTICAL MOTION OF AN AIRPLANE

1.1 Introduction

The vertical motion of the research aircraft of the Institute of Flight Guidance and Control is measured by three different sensor systems which are presented in chapter 1.2. One of their main characteristics are internal time delays that are due to data preprocessing. A continuous- discrete Kalman filter algorithm [1] is used for the estimation of these delays. In the Kalman filter, the delays are replaced by Padé-approximants which are presented in chapter 1.3. In chapter 2, an adaptive algorithm is presented that improves the performance of the Kalman filter. Chapter 3 shows results of simulations. A summary is given in chapter 4.

1.2 Measurement of the vertical motion

The description of the vertical motion is based on the kinematic differential equations of motion which describe the relation between the three-dimensional vectors of acceleration, velocity and position. They are the same for all airplanes [2] and are used here in the case of vertical motion. Assuming a flat non-rotating earth, the vertical movement is described by the relation between the altitude H and the acceleration in vertical direction a_{zg}:

$$\ddot{H} = a_{zg} .$$

There are several sensors in the research aircraft that can be used for the determination of the vertical motion: an Inertial Navigation System (INS), a barometric altimeter and a satellite navigation system (GPS).

The INS measures the acceleration in body-fixed coordinates and the Euler angles. With these data, the earth-fixed vertical acceleration can be computed. The measured acceleration a_{zgINS} is delayed due to the internal calculations

$$a_{zgINS}(s) = a_{zg}(s)\, e^{-sT_{tINS}} .$$

The altitude is measured by the altimeter as a function of the static air pressure. This sensor is dynamically slow but no delay due to internal calculation has to be considered. So the measured signal H_{alt} is

$$H_{alt}(s) = \frac{1}{1 + T_{alt}\, s}\; H(s)\ .$$

The GPS uses the runtime differences of signals from several satellites to calculate the position of the receiver. It is assumed that the GPS is used in differential mode for high accuracy [3]. The information is delayed due to the internal calculations. Additionally, the GPS gives valid informations only every 0.6 sec., instead of every 0.04 sec. as the other sensors do.

$$H_{DGPS}(t_k) = H(t_k - T_{tDGPS}) \qquad (t_k = 0 \text{ sec., } 0.6 \text{ sec., } 1.2 \text{ sec. } ...)$$

This behaviour is modelled using two sets of output equations within the Kalman filter. Dependent on the availability of a DGPS altitude information this measurement is included in the output equations or not.

1.3 Padé-approximants of time delays

The Padé-approximants [4] represent a power series by the ratio of two polynominals. With the given power series expansion of the time delay $f_T(s)$

$$f_T(s) = e^{-sT} = 1 + \frac{1}{1!}(-sT) + + \frac{1}{2!}(-sT)^2 + ... + \frac{1}{n!}(-sT)^n \; ...$$

the Padé-approximants give an optimum choice of the coefficients a_i, b_i of the corresponding transfer function g(s)

$$g(s) = \frac{b_0 + b_1\, s + ... + b_m\, s^m}{a_0 + a_1\, s + ... + a_n\, s^n}$$

Table 1 shows approximants of various numbers of m and n, the highest power of the numerator and denominator polynominal.

	0	1	2	m ·
1	$\frac{1}{1 + Ts}$	$\frac{1 - sT/2}{1 + sT/2}$		
2	$\frac{1}{1 + sT + (sT)^2/2}$	$\frac{1 - sT/3}{1 + 2sT/3 + (sT)^2/6}$	$\frac{1 - sT/2 + (sT)^2/12}{1 + sT/2 + (sT)^2/12}$	
n				

Table 1 : Padé-approximants to e^{-sT}

2. ADAPTIVE ESTIMATION OF TIME DELAYS

Many facts have to be considered when choosing a Padé-approximant that replaces a delay. From the theoretical point of view a numerator and denominator polynominal of high order would be appreciated. But from the practical point of view one tends to use transfer functions of small order because of the known problems of the Kalman filter algorithm with large matrices and the limitations of computing power on board the aircraft. Another problem is the influence of the approximant on the stability of the Kalman filter.

The use of an approximant is first investigated using a simplified example. It consists of a low pass filter of 1st order and a time delay T_t. Estimation of the state and the time constant of the delay is done using a continuous-discrete Kalman filter algorithm. In the Kalman filter, the delay is replaced by a Padé-approximant with m=1 n=1 (Table 1).

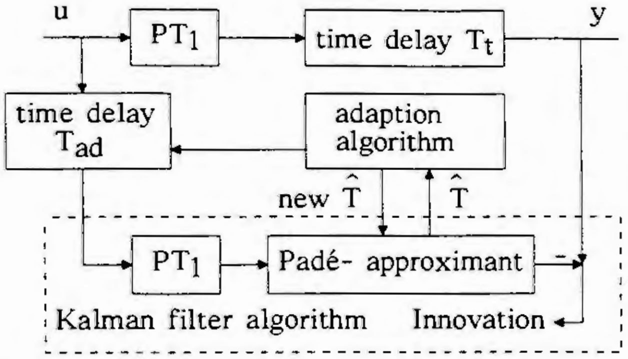
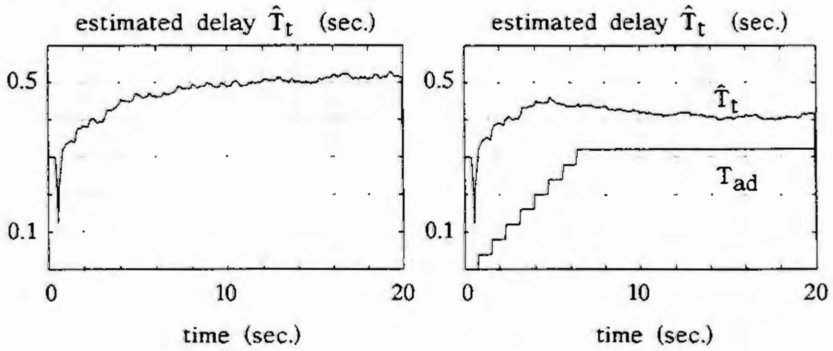


Fig. 1 Block diagram of a simple example

As shown in Fig. 2a, the delay time is estimated too high without adaption algorithm because there is a significant difference between the delay and the Padé-approximant if the delay time to be estimated is too high. The quality of the approximation increases as the time delay to be estimated is reduced. Therefore an adaption algorithm is introduced that uses the delay T_{ad} to delay the input signal of the Kalman filter, if the estimated delay $\hat{T}$ is not within given boundaries. The increase in performance is shown in Fig. 2b. The total estimated delay is the sum of T_{ad} and the delay estimated in the Kalman filter $\hat{T}$.



$$\frac{y_1(s)}{u(s)} = \frac{1}{1 + Ts}$$

$T = 0.3$ sec.

$T_t = 0.4$ sec.

$\sigma_u = 0.1$

$\sigma_y = 0.1$

Fig. 2 Estimated time delay without (a) and with (b) adaption algorithm

The technique described above is now used to estimate the time delays in the sensors that measure the vertical motion of the aircraft.

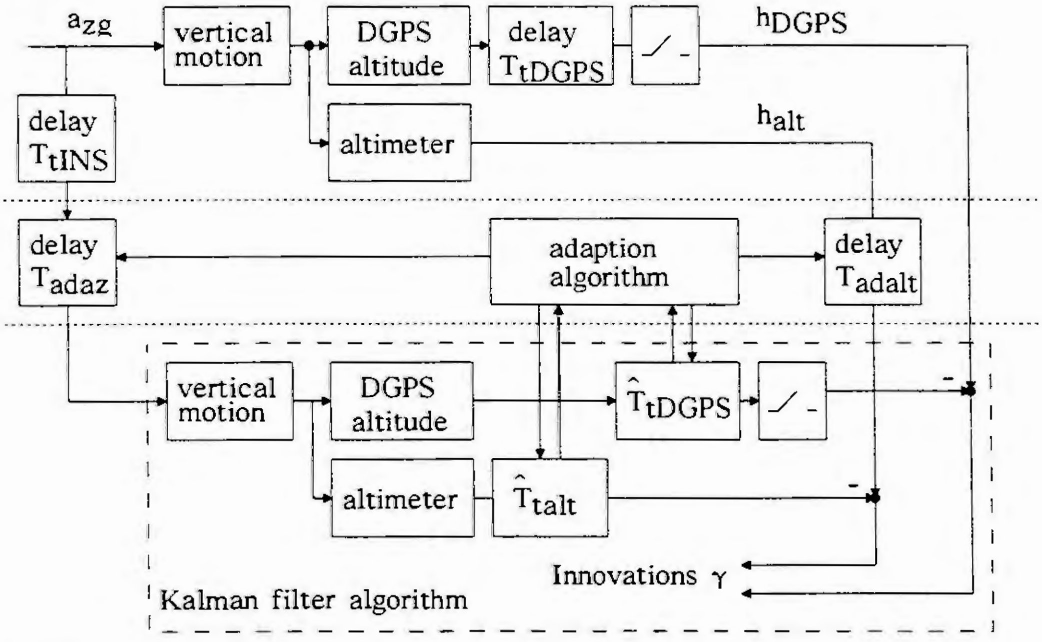


Fig. 3 Block diagram of the Kalman filter with the adaption algorithm

In order to show the principles of the estimation including the adaption algorithm (Fig. 3), some parameter values are assumed. Let's take T_{tDGPS} = 1.2 sec. and T_{tINS} = 0.12 sec. Then, the Kalman filter estimate should be $\hat{T}_{tDGPS}$ = 1.08 sec. This is too much to be accurately estimated, so the adaption algorithm decides to increase the input delay: T_{adaz} = 0.88 sec. But at the same time, the adaption algorithm also has to increase T_{adalt} to assure that the Kalman filter estimates a (positive) time delay $\hat{T}_{talt}$.

The delays in the sensors could be determined to be:

$$T_{tINSe} = T_{adalt} - T_{adaz} - \hat{T}_{talt}$$

$$T_{tDGPSe} = T_{adalt} - \hat{T}_{talt} + \hat{T}_{tDGPS}$$

3. RESULTS

In fig. 4, results of simulations are presented that were done using the parameters introduced in chapter 2. The delays T_{tINSe} and T_{tDGPSe} are calculated using the equations given above. They are shown with these parameters:

a. before the adaption process : T_{adaz} = 0.0 sec. T_{adalt} = 0.0 sec.

b. after the adaption process : T_{adaz} = 0.88 sec. T_{adalt} = 1.2 sec.

After increasing the values of T_{adaz} and T_{adalt} the estimation results are more exact. The estimated value of T_{tINSe} shows the increase in performance: the correct value is 0.12 sec., the estimated value is about 0.1 sec.

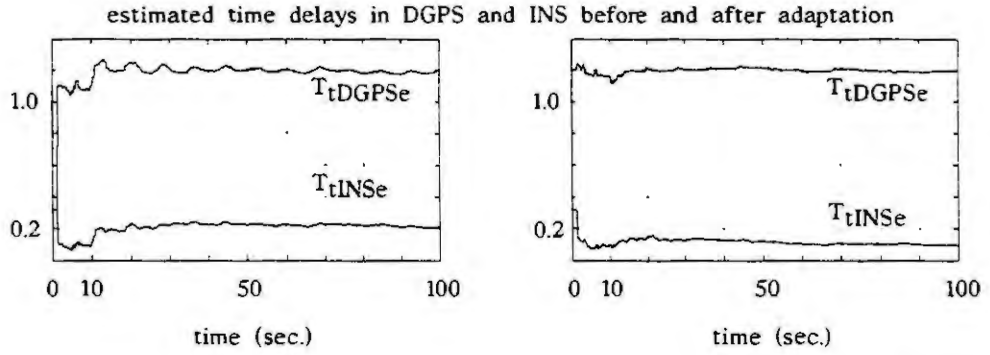


Fig. 4 T_{tDGPSe} and T_{tINSe} before (a) and after (b) the adaption process

4. SUMMARY

It is shown that it is possible to estimate time delays in the measurement of the vertical motion of a research aircraft. Different output equations were used in the Kalman filter algorithm according to the signals measured at the corresponding point of time. The delays were described using Padé-approximants. An improvement of the estimated delays was reached using an adaption algorithm.

REFERENCES

[1] Krebs, V. Nichtlineare Filterung, Oldenbourg, München 1980

[2] Brockhaus, R. Flugregelung, Springer, Berlin 1994

[3] Jakob, T. Beitrag zur Präzisionsortung von dynamisch bewegten Fahrzeugen
Dissertation, TU Braunschweig 1992

[4] Baker, G.A., Padé- Approximants
Graves- Morris, P. Encyclopedia of Mathematics and its Applications, Vol. 13
Addison- Wesley, London 1981

STABILIZATION OF CONSTRAINED NONLINEAR TIME-DELAY SYSTEMS

M. DAMBRINE, J. P. RICHARD

Ecole Centrale de Lille, LAIL-URA CNRS D 1440
B.P. 48, 59651 Villeneuve d'Ascq Cedex - FRANCE

Abstract. This paper is concerned with the problem of stabilizing a class of nonlinear, time-varying, differential-difference systems in presence of constraints on both control and state vectors. Our main tool is the notion of comparison system linked to the concept of vector norm. Then, sufficient conditions are given for a linear state feedback control law to satisfy the asymptotic stability of the system and an invariance property of the domain of constraints.

1. INTRODUCTION

Any real system is subjected to constraints which are consequences of physical limitations such as, for example, limitation of the amplitudes or response velocity of actuators, and in many cases, it must be taken into account for the control.

The design of constrained controllers for systems governed by linear ordinary differential equations can be based on two different approaches. The first one consists in the application of the optimal control laws (minimization of a given performance index under the constraints). The second approach is based on the concept of positive invariance. It has been applied for both continuous and discrete-time linear systems in [9], [10]. Coupled with the comparison method, this last approach provides interesting solutions for nonlinear systems [4], [8].

In this paper, the considered systems are mathematically represented by differential-difference equations. The main tools are the notion of comparison systems linked to the concepts of vector norms and time-delay overvaluing systems. Recent results ([1], [2]) on determination of invariant sets for linear time-delay systems permit us to find a linear state feedback which stabilizes the nonlinear system and satisfies the constraints imposed on the state and/or the control. Our results are an extension of Radhy's study (see [4], [5]) to the delay case.

Notations

Throughout the paper, $D^+p_i(x_i)$ represents the right-hand derivative of $p_i(x_i)$ taken along the motions of (1); $\mathcal{T}_0$ denotes the interval $[t_0\ , +\infty)$; $\mathcal{D}$ is a region of $\mathbb{R}^n$ containing a neighbourhood of the origin; $C(\mathcal{D}) = C([-\tau\ , 0], \mathcal{D})$ denotes the set of continuous functions that map the interval $[-\tau\ , 0]$ into $\mathcal{D}$. $x_t \in C(\mathbb{R}^n)$ is defined by $x_t(s) = x(t+s), -\tau \leq s \leq 0$. In section 4, the sets $I(v)$ and $\mathfrak{I}(v)$ are defined by: $I(v) = \{z \in \mathbb{R}^k : [|z_1|, |z_2|, \dots, |z_k|]^T \leq v\}$, and $\mathfrak{I}(v) = C(I(v))$, where v is a k-vector with positive components. At last, every vector or matrix inequality $A \leq B$ is to be understood for each corresponding component, and the abbreviation $M(.)$ stands for $M(t, x(t), x(t-\tau))$.

2. PROBLEM FORMULATION

The systems considered in this paper are described by the vector equation:

$$\dot{x}\,(t) = A_0(t, x(t), x(t-\tau))\, x(t) + A_1(t, x(t), x(t-\tau))\, x(t-\tau) + B(t, x(t), x(t-\tau))\, u(t), \tag{1}$$

where $x \in \mathbb{R}^n$ (the state vector), $u \in \mathbb{R}^m$ (the control vector), τ a positive number (the delay), $A_0(.)$ and $A_1(.)$ are $n \times n$ matrices :

$$A_0,\ A_1\ :\quad \mathcal{T}_0 \times \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^{n\times n}\ ,\ \text{where } \mathcal{T}_0 = [t_0\ , +\infty[,\ t_0 \in \mathbb{R},$$

and $B(.)$ is an $n \times m$ matrix : $\quad B \; : \quad \mathcal{T}_0 \times \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^{n \times m}$.

The control vector $u(t)$ is subject to linear constraints of the following symmetrical form:

$$-d \leq u \leq d, \tag{2}$$

where d is a real m-vector with positive components.

The state vector x is constrained to belong to the set:

$$\mathcal{G} = \{x \in \mathbb{R}^n : \; -w \leq x \leq w\}, \tag{3}$$

where w is an n-vector with non-negative components.

The problem studied here is the determination of a linear state feedback control law

$$u(t) = K\,x(t) \tag{4}$$

such that the solution of (1) for any initial state function satisfying $x_{t_o}(s) \in \mathcal{G}$ for all s in $[-\tau \;, \, 0]$ converges towards the origin while the control vector satisfies condition (2) and the state vector remains in the set $\mathcal{G}$. For this purpose, the notion of overvaluing systems obtained by use of vector norms is introduced in the following section.

3. VECTOR NORM AND COMPARISON LEMMA

3.1. Vector norms concept

The vector norm concept was first introduced by Robert [6] in order to solve numerical analysis problems linked to linear recurrences. We recall this concept here:

Consider the following partition of $\mathbb{R}^n$:

$\mathbb{R}^n = \mathrm{E}_1 \oplus \mathrm{E}_2 \oplus \ldots \oplus \mathrm{E}_k$, where $\oplus$ denotes the direct vector subspaces sum.

Let P_i denote the projection operator from $\mathbb{R}^n$ into E_i, and x be a vector of $\mathbb{R}^n$. The projection of x into E_i is x_i, so $x_i = P_i\, x = P_i\, x_i$.

Let p_i be a norm defined on the subspace E_i (i = 1, ..., k). Then the vector function $p \; : \; \mathbb{R}^n \; \rightarrow \; \mathbb{R}^k_+$ whose $\mathrm{i^{th}}$ component is defined by: $p_i(x) = p_i(x_i)$ is a regular vector norm (VN) of dimension k.

3.2. Overvaluing systems

The use of a vector norm considered as a special Lyapunov vector function enables us to define a special class of comparison systems.

Definition 1 : The matrices $M,\, N : \mathcal{T}_0 \times \; \mathcal{D} \; \times \; \mathcal{D} \rightarrow \; \mathbb{R}^{k \times k}$ define an *overvaluing system* of (1) with respect to the VN p and the domain $\mathcal{D}$ if and only if :

1) the following inequality is satisfied along every motion of (1) and for each corresponding component :

$$D^+p(x(t)) \leq M(t,\, x(t),\, x(t-\tau))\; p(x(t)) + N(t,\, x(t),\, x(t-\tau))\; p(x(t-\tau)), \tag{5}$$
$$\forall\; t \in \mathcal{T}_0 \;, \forall\; x_t \; \in \; \mathrm{C}(\mathcal{D}),$$

where $M(.) = \{\mu_{ij}(.)\}$, is such that its off-diagonal elements are non-negative, and $N(.) = \{\nu_{ij}(.)\}$ is a non-negative matrix,

2) the system (C) $\;\dot{z}\,(t) = M(.)\; z(t) + N(.)\; z(t-\tau)$ has time-continuous solutions. ■

For usual norms, the expressions of the natural overvaluing system are given in an explicit form. For example, if $p_i(x_i)$ is the "max" norm (maximum of the modulus of each component of subvector x_i), then let a partition of the space $\mathbb{R}^n$ define a block partition of matrix A. $\mathrm{I_i}$ and $\mathrm{I_j}$ represent the sets of indices of rows and columns, respectively, of block A_{ij}.

$$\begin{aligned}
\mu_{ii}(.) &= \max_{r \in \mathrm{I_i}} \Big[a_{rr} + \sum_{\substack{s \in \mathrm{I_i} \\ s \neq r}} |a_{rs}|\Big], && \forall\; i \;= 1,\, \ldots,\, k \\
\mu_{ij}(.) &= \max_{r \in \mathrm{I_i}} \Big[\; \sum_{s \in \mathrm{I_j}} |a_{rs}|\Big] \;, && \forall\; i\;, j = 1,\, \ldots,\, k \quad (i \neq j) \\
\nu_{ij}(.) &= \max_{r \in \mathrm{I_i}} \Big[\; \sum_{s \in \mathrm{I_j}} |b_{rs}|\Big] \;, && \forall\; i\;, j = 1,\, \ldots,\, k
\end{aligned} \tag{7}$$

The dual norm of the max norm (i.e. the modulus norm) leads to equations analogous to the precedent ones but inverting $s \in \mathrm{I}_j$ and $r \in \mathrm{I}_i$.

3.3. Comparison lemma

Let $M(.)$ and $N(.)$ define an overvaluing system of (1) with respect to a regular VN p and to the region $\mathcal{D}$. Then, the system

$$\dot{z}\,(t) = M(t,\,x(t),\,x(t-\tau))\;z(t) + N(t,\,x(t),\,x(t-\tau))\;z(t-\tau) \tag{8}$$

is an *overvaluing comparison system* of (1) in the sense that as long as $x(t)$ remains in $\mathcal{D}$, the inequality

$$z(t) \geq p(x(t)) \tag{9}$$

holds as soon as $z_{t_0}(s) \geq p(x_{t_0}(s))$ for all s in the initial interval $[-\tau\,,\,0]$. ■

Then, the stability (respectively the asymptotic stability) of the solution $z = 0$ of (8) implies the stability (resp. the asymptotic stability) of the zero solution of (1).

4. MAIN RESULTS

It is assumed in this section that with the closed-loop system (1) with (4), i.e. :

$$\dot{x}\,(t) = [A_0(.) + B(.)K]\;x(t) + A_1(.)\;x(t-\tau), \tag{10}$$

it can be associated a linear overvaluing comparison systemwith respect to a regular VN p:

$$\dot{z}\,(t) = M(K)\;z(t) + N\;z(t-\tau). \tag{11}$$

Then, the following theorems give sufficient conditions for the existence of a linear control law $u = Kx$ which stabilizes (1) under conditions (2) and (3).

Theorem 1 : The zero solution of system (11) is asymptotically stable independent of delay if and only if $M(K) + N$ is the opposite of an M-matrix.

Theorem 2 : It is assumed that the zero solution of system (11) is asymptotically stable independently of delay. Then, the set $\mathcal{S}(v)$ is a positively invariant set of (11) if and only if

$$(M(K) + N)\;v \leq 0 \tag{12}$$

Proofs : The sufficient condition of theorem 1 is proved in [7], and by taking $\tau = 0$, and by applying Kotelianski's conditions, it is proved that it is also a necessary one. Theorem 2 is based on results in [2].

Remarks: - An M-matrix is a matrix with positive off-diagonal elements such that all principal minors are non-negative ([3]).

- If $\mathcal{S}(d)$ is a positively invariant set of the overvaluing comparison system (11) then

$$\{\varphi \in C(\mathbb{R}^n) : p(\varphi(s)) \leq d,\; -\tau \leq s \leq 0\} \tag{13}$$

is obviously a positively invariant set of (1).

Theorem 3 : If there is a feedback matrix K such that $M(K) + N$ is the opposite of an M-matrix, the constrained regulation problem is solved, i.e. there is a vector v such that the set $\{x \in \mathbb{R}^n : p(x) \leq v\}$ is included in $\mathcal{G}$ and in the set $\{x \in \mathbb{R}^n : -d \leq Kx \leq d\}$.

5. EXAMPLE

Let us consider the following two-dimensional nonlinear delayed system

$$\dot{x}\,(t) = \begin{bmatrix} -4 + \sin x_1(t) & 3 \\ 2 + \cos t & -3 + x_2(t-2) \end{bmatrix} x(t) + \begin{bmatrix} x_1(\mathrm{t}) & 0 \\ 1 & 1 \end{bmatrix} x(t-2) + \begin{bmatrix} 2 + x_2(t) \\ 3 \end{bmatrix} u(t), \tag{14}$$

and a linear control law:

$$u = K\,x, \quad K = [k_1\;\;k_2]. \tag{15}$$

The state is constrained to stay in:

$$\mathcal{G} = \{x \in \mathbb{R}^2 : [-1\quad -1] \leq x^{\mathrm{T}} \leq [1\quad 1]\}, \tag{16}$$

and the constraint on the control u is: $|u(t)| \leq 3, \forall\, t \in \mathcal{T}_0.$ (17)

The closed-loop system is given by:

$$\dot{x}(t) = \begin{bmatrix} -4 + \sin x_1(t) + (2 + x_2(t))\, k_1 & 3 + (2 + x_2(t))\, k_2 \\ 2 + \cos t + 3\, k_1 & -3 + x_2(t-2) + 3\, k_2 \end{bmatrix} x(t) + \begin{bmatrix} x_1(t) & 0 \\ 1 & 1 \end{bmatrix} x(t-2). \qquad (18)$$

A comparison system of (14) valid in $\mathcal{G}$, associated with the VN $p(x) = [|x_1|\ \ |x_2|]^T$ is given by:

$$\dot{z}(t) = \begin{bmatrix} -3 + 2\, k_1 + |k_1| & |3 + 2\, k_2| + |k_2| \\ |2 + 3\, k_1| + 1 & -2 + 3\, k_2 \end{bmatrix} z(t) + \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} z(t-2). \qquad (19)$$

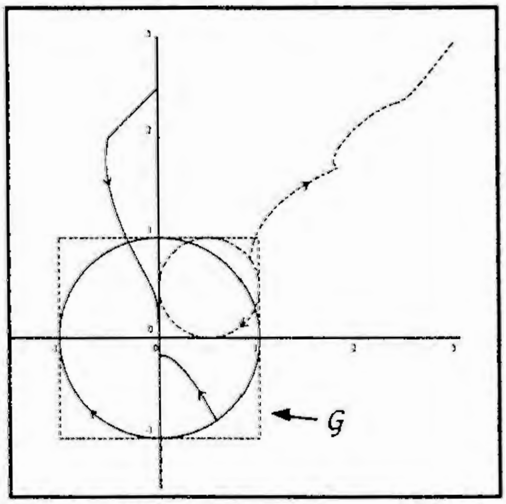
A sufficient condition of asymptotic stability of (19) and of positive invariance of $\mathcal{G}$ is:

$$-2 + 2\, k_1 + |k_1| + |k_2| + |3 + 2\, k_2| < 0 \text{ and } 1 + 3\, k_2 + |2 + 3\, k_1| < 0. \qquad (20)$$

An improvement of the dynamic is obtained by choosing $K_{opt} = [-1.25\ , -1.5]$ which minimizes the criterion

$$\max\ (-2 + 2\, k_1 + |k_1| + |k_2| + |3 + 2\, k_2|,\ 1 + 3\, k_2 + |2 + 3\, k_1|), \qquad (21)$$

then system (19) is asymptotically stable with the decay rate $\alpha = 0.368$.



simulation of solutions of (18) for different initial functions with K = K_{opt} (continuous lines), and K = 0 (dashed line).

6. REFERENCES

[1] Dambrine M., Richard J. P., Stability analysis of time-delay systems. Dynamic Systems & Applications. Vol 2, No. 3 (1993), pp. 405-414.

[2] Dambrine M., Richard J. P., Estimation of stability domains of nonlinear differential-difference equations. To appear in: Proc. Intern. IEEE/SMC'93, Le Touquet, 17-20 October 93.

[3] M. Fiedler and V. Ptak, "On matrices with non-positive off-diagonal elements and positive principal minors", Czec. Math. J., vol 12, No. 87, pp. 382-400, 1962.

[4] Radhy N.E., Borne P., Stabilizing regulators for constrained nonlinear time-varying continuous systems. Proc. of IMACS-MCTS Symposium, Vol 2, Lille, France (May 1991) pp. 350-356.

[5] Radhy N.E., Borne P. and Richard J.P., Regulation of nonlinear time-varying continuous systems with constrained state, In P. Borne and V. Matrosov (Eds), The Lyapunov functions method and applications. J.C. Baltzer AG, Scientific Publishing Co., 1990.

[6] Robert F. , "Normes vectorielles de vecteurs et de matrices." Rev. Fr. Trait. Inf. (Chiffre) vol 17, 4 (1964), pp. 261-269.

[7] Tokumaru H., Adachi N., Amemiya T., "Macroscopic stability of interconnected systems", Proc. of IFAC 6th World Congress, Boston, Paper 44.4, 1975.

[8] Vassilaki M., Application of the method of Lyapunov functions to the design of constrained regulators. In P. Borne and V. Matrosov (Eds), The Lyapunov functions method and applications. J.C. Baltzer AG, Scientific Publishing Co., 1990.

[9] Vassilaki M., Hennet J. C. and Bitsoris G., Feedback control of linear discrete-time systems under state and control constraints. Int. J. Control, Vol 47, No. 6 (1988), pp. 1727-1735.

[10] Vassilaki M., Bitsoris G., Optimum algebraic design of continuous-time regulators with polyhedral constraints. Preprints of AIPAC'89, Nancy, France, IFAC, Vol 1 pp. 61-64, 1989.

A LINEAR STATE SPACE APPROACH TO A CLASS OF DISCRETE–EVENT SYSTEMS

Dieter FRANKE
Universität der Bundeswehr Hamburg
Fachbereich Elektrotechnik
D – 22039 Hamburg

Abstract. The theory of finite automata provides an adequate access to modelling, analysis and control of discrete–event dynamical systems arising in numerous engineering applications like transport processes and production lines. This paper shows that finite automata are much closer related to discrete–time systems than assumed in the past. To this end a novel representation of Boolean functions is introduced similar in some sense to the canonical form of SHEGALKIN Polynomials. As a special class, systems which are linear in the sense of common algebra are considered in some detail.

1. INTRODUCTION

The state equations of a finite automaton [2], [3]

$$\mathbf{x}(k+1) = \mathbf{f}[\mathbf{x}(k), \mathbf{u}(k)], \tag{1}$$

$$\mathbf{y}(k) = \mathbf{g}[\mathbf{x}(k), \mathbf{u}(k)], \tag{2}$$

with input $\mathbf{u}$, state $\mathbf{x}$ and output $\mathbf{y}$, are resembling those of a classical discrete–time system [1], [5]. The main difference is that in a finite state machine these equations are defined over a Galois field rather than the field of real or complex numbers. In this paper we restrict ourselves to Boolean vectors $\mathbf{u}$, $\mathbf{x}$ and $\mathbf{y}$ whose components can only take the logical values 0 and 1. This type of automata is of great importance in binary process control [4] and discrete–event dynamical systems.

2. AN ARITHMETIC REPRESENTATION OF BOOLEAN FUNCTIONS

We recall the canonical form of *Shegalkin* Polynomials [6] which allows the representation of any Boolean function $y = f(\mathbf{x}) = f(x_1, ..., x_n)$, using only the conjunction and the antivalence operation. Now by keeping the general multilinear structure of Shegalkin polynomials and by replacing conjunction and antivalence by *multiplication and summation in the common arithmetic sense*, one obtains the following representation as a multilinear arithmetic polynomial:

$$y = f(\mathbf{x}) = f(x_1, ..., x_n) = a_0 + \sum_{i=1}^{n} a_i x_i + \sum_{j=2}^{n} \sum_{i=1}^{j-1} a_{ij} x_i x_j + \\ + \sum_{k=3}^{n} \sum_{j=2}^{k-1} \sum_{i=1}^{j-1} a_{ijk} x_i x_j x_k + ... + a_{123\cdots n} x_1 x_2 x_3 ... x_n. \tag{3}$$

The number of a–coefficients appearing in this equation is $N = 2^n$ which complies with the number of rows of the sequence table of $y = f(\mathbf{x})$. Therefore, given any *completely specified* sequence table, the coefficients are determined uniquely by solving a set of *linear* algebraic equations.

Consider for example the simple case $n = 2$, hence

$$y = f(x_1,x_2) = a_0 + a_1 x_1 + a_2 x_2 + a_{12} x_1 x_2. \tag{4}$$

Let the switching table 1 be given,

x_2	x_1	y
0	0	$y^{(1)}$
0	1	$y^{(2)}$
1	0	$y^{(3)}$
1	1	$y^{(4)}$

TABLE 1: Switching table for $y = f(x_1,x_2)$

with given binary values $y^{(1)}$, ..., $y^{(4)}$. Here the a–coefficients must satisfy the equation

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_{12} \end{bmatrix} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ y^{(3)} \\ y^{(4)} \end{bmatrix}, \tag{5}$$

which has the unique solution

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_{12} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 1 & -1 & -1 & 1 \end{bmatrix} \cdot \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ y^{(3)} \\ y^{(4)} \end{bmatrix}. \tag{6}$$

In the past, this type of modelling Boolean functions has only been used in Boolean reliability theory [7]. It does, however, also apply to functions $f(\mathbf{x},\mathbf{u})$ and $g(\mathbf{x},\mathbf{u})$ appearing on the right hand side of equations (1) and (2), respectively. This makes finite state machines much closer related to classical discrete–time systems than assumed in the past and provides a novel access to analysis and design of binary dynamical systems.

3. ARITHMETICALLY LINEAR BOOLEAN FUNCTIONS AND AUTOMATA

There is an appealing class of special systems which have a simple structure even in higher dimensions: *arithmetically linear systems*. The state equations of an arithmetically linear automaton obviously have the general form

$$\mathbf{x}(k+1) = A\,\mathbf{x}(k) + B\,\mathbf{u}(k) + \mathbf{a}_0, \tag{7}$$

$$\mathbf{y}(k) = C\,\mathbf{x}(k) + D\,\mathbf{u}(k) + \mathbf{c}_0, \tag{8}$$

where A, B, C and D are constant matrices of appropriate dimensions, and $\mathbf{a}_0$, $\mathbf{c}_0$ are n–dimensional constant vectors.

The arithmetically linear automaton just introduced is *completely specified*. Its linearity results from the special feature that all coefficients of multilinear terms in equation (3) vanish. There is, however, another important class of automata, namely *incompletely specified automata*, which sometimes can be made algebraically linear by setting certain coefficients equal to zero.

Incompletely specified Boolean functions arise whenever one or more rows of the sequence table can be excluded by knowledge or is forbidden to occur. As an example consider table 1 with the first row being excluded. Based on the notation used in equation (4), equation (6) does not directly apply, since $y^{(1)}$ is unspecified. However, the solution of the underdetermined set of equations can be made unique by setting $a_{12} = 0$. This makes on the one hand $y = f(x_1,x_2) = a_0 + a_1 x_1 + a_2 x_2$ strictly *linear*, and on the other hand specifies $y^{(1)} = y^{(2)} + y^{(3)} - y^{(4)}$. The value of $y^{(1)}$ will in general not be binary. For the example $y^{(2)} = y^{(3)} = 1$, $y^{(4)} = 0$ we obtain $y^{(1)} = 2$ and hence $a_0 = 2$, $a_1 = a_2 = -1$. Therefore,

$$y = f(x_1,x_2) = 2 - x_1 - x_2$$

in this example.

The automaton acc. to equations (1), (2) with dimensions n, p and q for the state $\mathbf{x}$, control $\mathbf{u}$ and output $\mathbf{y}$, respectively, is said to be incompletely specified, if only a subset of rows is admitted in the sequence table.

An important class of incompletely specified automata are those, which allow only a subset of the 2^n entries of $\mathbf{x}$. In such an automaton, control $\mathbf{u}(k)$ is said to be *applicable* to state $\mathbf{x}(k)$ if $\mathbf{x}(k+1) = \mathbf{f}[\mathbf{x}(k), \mathbf{u}(k)]$ also belongs to the allowed subset.

4. EXAMPLES

4.1 Transportation process

An example of practical importance is the process of transportation of discrete parts A($\hat{=}$ 1) and B($\hat{=}$ 0), which can be written in matrix notation:

$$\mathbf{x}(k+1) = \begin{bmatrix} 0 & 0 & 0 \cdots\cdots & 0 \\ 1 & 0 & 0 \cdots\cdots & 0 \\ 0 & 1 & 0 \cdots\cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 \cdots 1 & 0 \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} u(k)\,. \tag{9}$$

Here, x (k) and u (k) are the binary state vector and control input at step k, respectively. Such a model is suitable presumed that two different types of pieces are to be transported, e.g. raw materials for a production line.

So far this is a completely specified automaton. In some applications a specific loading of the transportation line may be required, e.g. in such a way that at each step k at most one part B ($\hat{=}$ 0) is allowed on the transportation line, since mainly parts A ($\hat{=}$ 1) are requested at the end of the line. This gives rise to an incompletely specified machine, and the binary control u(k) has to be selected with care such that it is applicable in the above defined sense.

4.2 Tank with inflow and discharge

Consider a tank with valve 1 for inflow of a liquid and value 2 for possible discharge. The system will be modelled as a finite state machine, and therefore binary controls

$$u_i = \begin{cases} 1, \text{ valve open}, \\ 0, \text{ valve closed}, \end{cases} \qquad i = 1, 2,$$

and binary state

$$x = \begin{cases} 1, \text{ tank full}, \\ 0, \text{ tank empty}, \end{cases}$$

will be used. From the corresponding sequence table (TABLE 2) it can be seen that this is an example of an incompletely specified machine because four rows are not allowed for physically evident reasons. Now by means of the procedure of non–binary completion proposed in Chapter 3, the degrees of freedom in table 2 can be utilized such as to make the state equation of the binary process arithmetically *linear*. The result is

$u_2(k)$	$u_1(k)$	$x(k)$	$x(k+1)$	
0	0	0	0	
0	0	1	1	
0	1	0	1	
~~0~~	~~1~~	~~1~~	$x^{(4)}$	not specified
~~1~~	~~0~~	~~0~~	$x^{(5)}$	
1	0	1	0	
~~1~~	~~1~~	~~0~~	$x^{(7)}$	not specified
~~1~~	~~1~~	~~1~~	$x^{(8)}$	

TABLE 2 – Sequence table for tank

$$x(k+1) = x(k) + u_1(k) - u_2(k), \tag{10}$$

and it is emphasized that a classical continuous–time or discrete–time model of this process would by no means be linear.

5. CONCLUSIONS

Motivated by the general multilinear structure of Shegalkin polynomials for the canonical representation of Boolean functions, a novel representation of these functions has been introduced. It is based on common arithmetic operations rather than Boolean algebra. This type of mathematical models for binary dynamical systems can be utilized for the design of binary feedback control in a close relationship to classical discrete–time control systems. Details will be reported elsewhere.

6. REFERENCES

[1] Ackermann, J., Abtastregelung, Band I. Springer–Verlag, Berlin, Heidelberg, New York, 2. Auflage, 1983.

[2] Bochmann, D., Einführung in die strukturelle Automatentheorie. VEB Verlag Technik, Berlin, 1975.

[3] Booth, T.L., Sequential Machines and Automata Theory. J. Wiley and Sons, New York, London, Sydney, 1967.

[4] Fasol, H.K., Binäre Steuerungstechnik. Springer–Verlag, Berlin, Heidelberg, New York, 1988.

[5] Föllinger, O., Lineare Abtastsysteme. R. Oldenbourg Verlag, München, Wien, 2. Auflage, 1982.

[6] Shegalkin, I.I., Die Arithmetisierung der symbolischen Logik. Mat. $c\sigma$, T. 35 (1928), 311–377.

[7] Störmer, H., Mathematische Theorie der Zuverlässigkeit. Akademie–Verlag, Berlin, 1970.

LINEAR QUADRATIC MODELLING IN MULTILEVEL SYSTEMS

K. Petrova and T. Stoilov,

Bulgarian Academy of Sciences, Central Laboratory of Automation,Acad.G.Bontchev str., bl.105, 1113 Sofia, Bulgaria, tel.(02) 713 2774, Fax 0359 2 723787

Abstract: Coordination strategy in multilevel control systems is considered. Noniterative computations and data transfer between the levels are achieved. Linear quadratic approximation of a two control level policy is introduced. The lack of iterative computations benefits the real time implementation of such two control level procedure.

1. INTRODUCTION

The multilevel approach was sucessfully applied for mathematical modelling and problem solution for large scale problems [Bradys and Roberts 1986]; nonlinear problems with time delays [Wang and Jamishidi 1984]; stair case problems with angular overlapping structures [Takashi Shima and Haimes, 1984]; optimal control and parameter estimation [Michalska at al.1985]. These applications concern the mathematical modelling of multilevel systems, decomposition into subproblems between layers, coordination of the subproblems which solutions give the solution of the initial optimiation problem. The main coordination strategies, goal coordination and the predictive coordination perform iterative information transfer between the levels before achieving the global optimal solution [Mesarovich at al.1970]. These iterations are not friendly related to the real time control processes and control systems. In a case of distributed control system these iterations increase the time for control influence evaluation, as additional communication and information transfer is included between the supremal and infimals control levels, fig.1. The multilevel theory is not practicle applicable in real time applications, due to the iterative manner of operation among the computational levels.

2. CASE STUDY

This research introduces a linear quadratic mathematical model of two control levels system which decreases the iterative rules of computations up to one. The infimal control level computes their local control prepositions. The supremal level (coordinator) agrees or corrects these prepositions and the global optimization control problem is solved without iterative computations. The multilevel system operates in a way:

1. Independant computations of the local solutions $x_i(\lambda)=x_i(0)$, by the Local Control Units **LCU** / the Local Subproblem **SP** is defined according the Subsystem **SS** definition and the **LCU** performance index/ assuming lack of coordination influence $\lambda=0$ from the Coordination Unit **CU**.

2. The coordination unit agrees or corrects the prepositions $x_i(0)$ by means of determining the global optimal solution x^{opt}. Hence this coordination strategy consists global optimal evaluation of x^{opt} as a function η of the local prepositions

$$(1) \qquad x^{0pt} = x(\lambda^{opt}) = \eta(x\ (0),\ i=1,n\ .$$

To perform this noniterative strategy (1), an explicit analytical model of η is proposed.

3. LINEAR QUADRATIC APPROXIMATION

The global optimization problem, resolved by the multilevel system is assumed in a block diagonal static form

$$(2) \qquad \mathbf{min} \left\{ F(x_1,\dots,x_n) = \sum_{i=1}^{n} F_i\,(x_i\,)\ \right\},$$

$$g_1(x) = 0 = \sum_{i=1}^{n} g_1^{(i)}\,(x_i\,), \qquad\qquad g_m(x) = 0 = \sum_{i=1}^{n} g_m^{(i)}\,(x_i\,)$$

$$h_1(x_1)=0 \qquad\qquad h_m(x_n)=0,$$

$x = (x_1,..,x_n)$, $g = (g_1,\dots,g_m)$, F,F_i,g_i,g_{ij},h_i-scalar functions.

Applying the goal coordination approach from the Lagrange problem

$$(3) \qquad \mathbf{\underset{x}{min}} \left\{ L(x,\lambda,\psi) = F(x) + \lambda^T g(x) + \psi^T.h(x) \right\}$$

a nonlinear equation system is obtained

$$(4) \qquad \partial L/\partial x_i = 0 = \partial F_i/\partial x_i + \sum_{j=1}^{m} \lambda_j.[\partial g_j^{(i)}(x_i)/\partial x_i] + \psi_i \partial h_i/\partial x_i,\ \ i=1,n.$$

The infimal solutions with lack of coordination influences $x_i(0,\psi^*)$ are evaluated from the system

$$(5) \qquad F'_{i,x} + \psi_i . h'_{i,x} = 0, \qquad i = 1,n.$$

To obtain an explicit analytical relation of $x_i(\lambda,\psi)$ Taylor series are involved. Hence $x(\lambda,\psi)$ is worked out in the form

$$(6) \qquad x(\lambda,\psi) = x(0,\psi^*) + x'_\lambda\Big|_{0,\psi^*} . \lambda + x'_\psi\Big|_{0,\psi^*} (\psi - \psi^*),$$

where the symbol $\Big|_{0,\psi^*}$ denotes evaluation of x'_λ and x'_ψ at the point $(0,\psi^*)$, respectively $x(0,\psi^*)$ and the derivates x'_λ and x'_ψ are evaluated from (4) by differentiating to λ and ψ

$$(7) \qquad \left[x'_\lambda \right]_{mxn} \Big|_{(0,\psi^*)} = \left[\frac{-g'_x}{F''_{xx} + \psi^*_n . h''_{xx}} \right]_{mxn} \Big|_{(0,\psi^*)},$$

$$(8) \qquad \left[x'_\psi \right]_{nxn} \Big|_{(0,\psi^*)} = \left[\frac{-h'_x}{F''_{xx} + \psi^*_n . h''_{xx}} \right]_{nxn} \Big|_{(0,\psi^*)},$$

The optimal coordination values λ^{opt} are evaluated from the inverse Lagrange unconstrained problem

$$(9) \qquad \max_{\lambda,\phi} \{ H(\lambda,\psi) = L(x(\lambda,\psi),\lambda,\psi) \},$$

where $H(\lambda,\psi) = F(x(\lambda,\psi)) + \lambda^T . g(x(\lambda,\psi)) + \psi^T . h(x(\lambda,\psi))$

This problem is worked out in linear equation form

$$(10) \qquad \frac{dH}{d\lambda} = g(0,\psi^*) + (H''_{\lambda\lambda}\Big|_{(0,\psi^*)} + H''^T_{\lambda\lambda}\Big|_{(0,\psi^*)})\lambda + H''_{\lambda\psi}\Big|_{(0,\psi^*)}(\psi - \psi^*) = 0$$

$$\frac{dH}{d\psi} = h(0,\psi^*) + (H''_{\psi\psi}\Big|_{(0,\psi^*)}(\psi - \psi^*) + H''_{\lambda\psi}\Big|_{(0,\psi^*)}(\psi - \psi^*) = 0$$

with the solution

$$(11) \qquad \lambda^{opt}, \psi^{opt} = \arg \{ H'_\lambda = 0, H'_\psi = 0 \}$$

Using (11) in (6) explicit analytical description

$$(12) \qquad x^{opt} = x\,[\, x(0,\phi^*), F, g, F'_x, g'_x, F''_{xx}]$$

is obtained between the local prepositions $x_i(0)$ and the system parameters F, g, F', g', F". The explicit analytical function (12) allows to perform new coordination strategy for real time control in multilevel systems. It introduces one step coordination procedure and overcomes the iterative manner of coordination computations. It is benefitial to apply this one step coordination in geographicaly distributed systems in regards of decreased information transfer in the control system.

4. REFERENCES

Bradys M., P.D.Roberts, 1986. Optimal structures for steady - state adaptive optimizing control of large scale industrial processes. Int. J. Systems Science, vol.17, N1, p.1449-1474.

Mesarovic M., Macko D. and Takahara Y., 1970. Theory of hierarchical multilevel systems. Academic Press, New York.

Michalska H., I.E.Ellis, P.D.Roberts, 1985. Joint coordination Method for the Steady - state Control of Large Scale Systems. Int. J. Systems Sci., vol.16, N5, p.605 - 618.

Takashi Shima, Y.Haimes, 1984. The Convergence Properties of Hierarchical Overlapping Coordination. IEEE Transactions on Systems, Man and Cybernetics, vol.SMC-14, N1.

Wang C., M.Jamshidi, 1984. Optimal Control of Large Scale Nonlinear Systems with Time Delay. International Journal of Control, vol.39,N4.

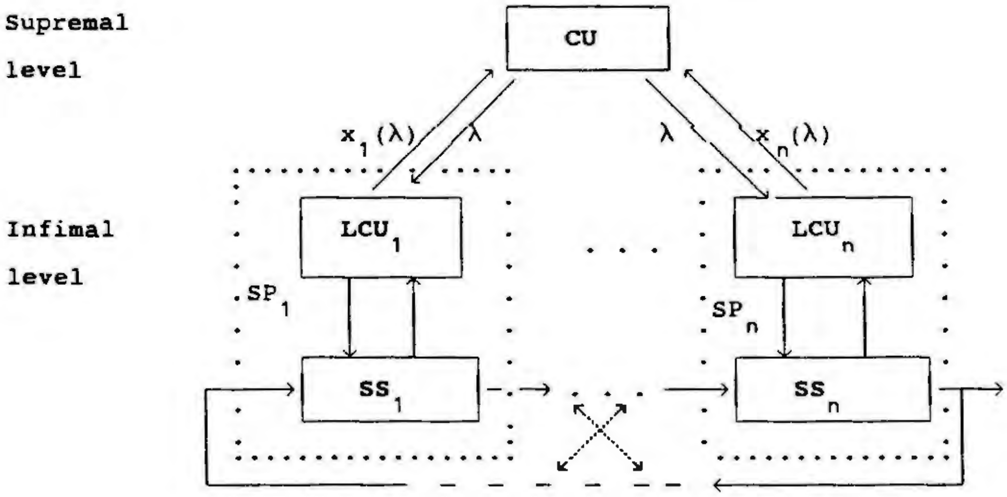


Fig.1. Two levels hierarchical control system

AN EXAMPLE OF PROCESS MODELLING AND REAL-TIME SIMULATION FOR A MULTIVARIABLE CONTROL ALGORITHM TESTING

Juš Kocijan, Rihard Karba
Faculty of Electrical and Computer Engineering
Tržaška 25, 61000 Ljubljana, Slovenia

Abstract. In the paper an example of process, namely semibatch distillation column, modelling and real-time simulation with hardware-in-the-loop for multivariable control algorithm testing is presented. Descriptions of the process model, approach to selected control design and implementation of control algorithm in state space are given. Simulation results are presented in the paper and a comment is given about the used method for the control design evaluation.

1. INTRODUCTION

Eventhough various control algorithms with possibility to satisfy different performance specifications exist, in practice simple PID controllers are frequently used. The latter are so frequent because they are easy for tuning since each part of them causes well known responses in closed-loop behaviour. Special stability tests or theoretical analysis do not need to be performed and on-line tuning on the real plant gives satisfactory results for majority of industrial plants. However, there is a certain class of plants where on-line tuning is not applicable, because of plant hazardous properties or because relatively poor PID controller structure can not assure specified control performance. In this case more sophisticated control algorithms should be used. Use of the latter means that on-line tuning is not or is at least hardly applicable and controller hardware should be more capable. One possible solution of this problem is the test of control algorithm and hardware with real-time simulation (hardware in the loop) for which and adequate process model is needed.

In our paper an example of multivariable control test with real time simulation of a plant mathematical model is enlightned through a case study. Multivariable processes certainly represent a class of problems where on-line tuning can hardly be used. On-line tuning can be used, as it is case with univariable control, mainly for PID control algorithms, while other more advanced approaches for control design should be used for other ones. Multivariable controller hardware in process control is limited to industrial PC computers and microcomputer multiloop controllers which enable implementation of complex control algorithms. In our case an evaluation of the multivariable control of semibatch destillation column with LQG/LTR (Linear Quadratic Gaussian / Loop Transfer Recovery) controller is studied. The column model, obtained with mathematical modelling and measured data curve fitting, is presented in the following section where the implementation of the control algorithm, given in the state space, with multiloop microcomputer controller and evaluation with real-time simulation by the aid of continuous simulation language SIMCOS is given in the third section.

2. DISTILLATION COLUMN MODEL

Distillation is a typical energy-consuming process. Even for a nonproblematic distillation, the successful control of the compositions of both top and bottom product can yield substantial profit resulting from the potential saving in utility cost. For the separation of components on the basis of volatility different kinds of distillation are used in the chemical industry [5, 6]. The general distillation column consists of a reboiler, where approximately constant vapor flow rate is ensured through the corresponding heating. A vapor from the reboiler goes up through the column, gives up part of its energy at each plate, and helps the vaporization of more volatile component. The role of the distillation column is to separate the mixture so that the distillate has the prescribed concentration. In the condenser such heat exchanging must ensure that all vapor will condense. However, there is no need for cooling of the distillate, because part of the liquid is returned to the top of the column as the reflux flow with a temperature which is near to the boiling point. This is done by the reflux distributor which returns one part of the liquid from the condenser to the column and the other part is drawn as a top product.

The distillation devices which can in general separate two (binary distillation) or more components are in most cases used as *continuous distillation columns* where the mixture is fed continuously into the column in the prescribed place somewhere along the column. In the case of small production and often changing mixture, however, the *batch rectification* (distillation is often called rectification for the batch type regime of column operation) is suitable. Here the mixture to be distillated is put into the reboiler and the procedure of separation lasts until the concentration and/or mass of the mixture in the reboiler become so small that the prescribed quality of the distillate can no longer be attained. The fact that the possible impurities in the mixture remain in the reboiler and thus the column is not soiled, as in continuous operation, can be also regarded as an advantage. In some cases, this type of distillation can be improved by the use of *semibatch rectification*. In distinction from the batch operation here the reboiler is continuously fed with the same flow of the mixture as it is the flow of the distillate which is fed into the accumulator. The advantage of such a type of distillation is not only the enlarged mass of the final product but also the constant level of the mixture in the reboiler which as the consequence ensures approximately constant vapor flow rate.

A semibatch destillation column was considered in our case. In the process of dissolvents regeneration the task was to separate methanol from water. These components are usually mixed in an impure lye and the semibatch distillation should enable, due to its character, the successful regeneration of methanol. The repeated use of the dissolvents makes for a substantial profit for the industry. In our case, the rectification device contained eight plates including the condenser and the reboiler ($n = 8$). In the procedure of modelling, with the goal of developing a model usable for the corresponding control design, the following additional assumptions were made:

- Liquid holdup is constant for particular plates. It turns out that this assumption is justified especially for describing the column behaviour in the steady state (not in the phase of putting into operation) what is usually the case.
- Vapor flow rate along the column is approximately constant (it depends mainly on the heating in the reboiler if the device is thermically well isolated).
- Liquid flow rate is a function of reflux rate but is in general independent of its location along the column.
- The pressure in the column varies from plate to plate. However, in the model it is pressumed to be constant, while deviations which occur because of this approximation are compensated througb distribution coefficients for the particular plates and time intervals. The distribution coefficients were calculated for the particular plates and for every time interval determined with the measurement data of x_i, which were used for modelling [4], and where they are supposed to be constant. They are obtained from the corresponding vapor-liquid equilibrium table for the mixture methanol-water [2].

Semibatch distillation prolongs the batch process by feeding the mixture in the reboiler, obeying the relation

$$distillate\ flow\ =\ bottom\ inflow\ of\ mixture \tag{1}$$

Therefore, no top product (distillate) is fed into the reboiler but, rather, only the same flow rate of mixture. The nonlinear model of the semibatch distillation column can be described by the following equations [4]:

$$\frac{dx_1(t)}{dt} = \frac{1}{M_1}[k_2x_2(t) - x_1(t)]V(t) \tag{2}$$

$$\frac{dx_i(t)}{dt} = \frac{1}{M_i}\{L(t)[x_{i-1}(t) - x_i(t)] + V(t)[k_{i+1}x_{i+1}(t) - k_ix_i(t)]\} \tag{3}$$

$$\frac{dx_n(t)}{dt} = \frac{1}{M_n}\{[V(t) - L(t)]x_0(t) + L(t)x_{n-1}(t) - V(t)k_nx_n(t)\}. \tag{4}$$

where the equations elements are as follows: L - liquid flow rate leaving plates $[\frac{kmol}{s}]$, V - vapor flow rate leaving plates$[\frac{kmol}{s}]$, x_i - liquid molar composition of more volatile component on the plate i, k_i - distribution coeficient on the plate i, M_i - liquid holdup on the plate i $[kmol]$.

The issues of the modelling for the investigated distillation column are thoroughly described in [4]. Eventhough in the procedure of modelling certain simplifications were assumed it should be pursued that model behaves like the real process in all important details. Otherwise, the purpose of simulation model for control testing is not achieved. Note that especially nonlinearities, constraints and expected high frequency dynamics have to be included in model which is significantly different to the model used for the controller design.

3. CONTROL DESIGN AND REAL-TIME SIMULATION

For the controller design a linear time-invariant process model is usually required. In our case linearization procedure has been done in two steps. The first step was to obtain high order linear plant for every time interval determined with measurement data which resulted in a set of linear time-invariant models. In the second step one low (third) order process model valid for the whole time range was obtained from the set of high order models. The two step procedure of linearization and validation of models is described in [4].

The LQG/LTR control design procedure has been selected for controller design without going into details why the mentioned type of controller has been designed. One of the reasons was that it results in relatively high order controller usually described in state space and is not suitable for on-line tuning on the process. LQG controller is one of the most popular optimal multivariable controller which bases on a minimisation of the criterion

$$J = \int_0^\infty (\boldsymbol{z}^T \boldsymbol{Q} \boldsymbol{z} + \boldsymbol{u}^T \boldsymbol{R} \boldsymbol{u}) dt; \quad \boldsymbol{z} = \boldsymbol{M} \boldsymbol{x} \tag{5}$$

where $\boldsymbol{Q}, \boldsymbol{R}, \boldsymbol{M}$ are constant matrices, $\boldsymbol{x}$ is the process model state vector and $\boldsymbol{u}$ is the process model input vector. The controller consists of optimal state controller and Kalman filter for the states estimation. Each of these two parts has good properties [3] which can be destroyed with joining optimal state controller and Kalman filter. This problem is overcome by the aid of LTR procedure with loop-shaping in the frequency domain. The detailed and transparent description of LQG/LTR design procedure for multivariable processes is given in [3]. The result of control design procedure is a multivariable controller of the fifth order. The chosen controller hardware was multiloop microcomputer controller MMC-90 [1], which is a product of Jozef Stefan Institute, Ljubljana. It enables an implementation of various simple and complex control algorithms by the aid of block schemes which are later automatically converted into machine language. It was chosen as an alternative to industrial PC which are usually more expensive and space consuming.

The MMC-90 controller was connected to PC computer which simulated the process model through A/D and D/A converters. Real-time simulation with hardware-in-the-loop was realized by the aid of simulation language SIMCOS [7]. Regulation control to step disturbance at 5000s on both inputs was simulated. Set point for both outputs was determined as values of process outputs without disturbances. The results of the real-time simulation with hardware-in-the-loop are given in Figure 1.

Some considerations for the described control algorithm testing:

- The process constraints (including actuators and sensors), have to be included into control design.
- Normalizing of input and output controller signals is neccessary.
- A great care has to be performed with the procedure of controller design due to dynamic constraints of controller hardware (sampling time, numerical errors).
- Sampling time has to be longer than communication interval of the real-time simulation.
- Repeating the simulation is not recommendable when simulation runs are very long (in our case app. four hours).

5. CONCLUSION

In the paper an example of process, namely semibatch distillation column, modelling and real-time simulation with hardware-in-the-loop for multivariable control algorithm testing is presented. Descriptions of the process model, approach to selected control design and implementation of control algorithm in state space are included. Simulation results are presented in the paper and a comment is given about presented method of the control design evaluation.

While in the other fields hardware-in-the-loop testing of control algorithms and hardware is not so uncommon, in the process control this approach is not frequently used. However, it gives some advantages

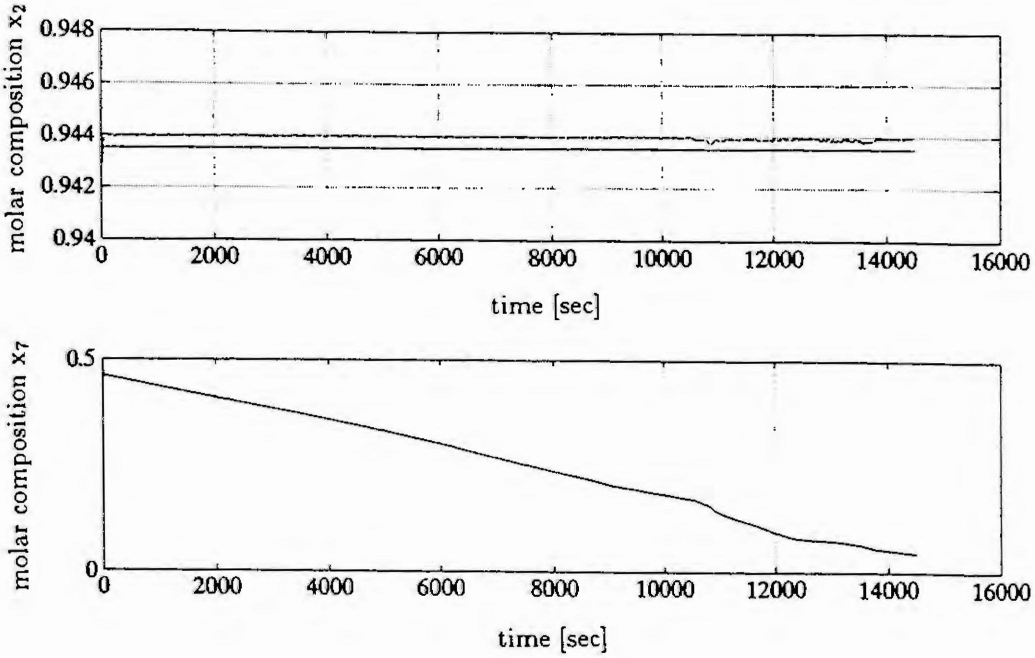


Figure 1: The closed-loop system response to step disturbances on inputs of the nonlinear process model: relative molar composition on the second plate x_2 (dashed line), setpoint value (full line) and relative molar composition on the seventh plate x_7 (dashed line) with setpoint value (full line)

to control design, especially implementation of more advanced control algorithms as it is shown in the paper.

The following conclusions can be drawn from the presented work:

- process model precision is of crucial importance if it is used for control algorithms and hardware testing;
- any kind of advance control algorithm can be tested in this way if controller hardware allows its implementation;
- a lot of implementation issues can be encountered and solved, before the control system is implemented on the real system.

6. REFERENCES

[1] Bitenc, A., et al., Design and application of an industrial controller, Computing & Control Engineering Journal, 3, (1992), 29-34.

[2] Gmehling, J. and U. Onken, Vapor - Liquid Equilibrium Data Collection, Aqueous - Organic Systems, Dechema, Chemistry data series (Behrens D. and R. Eckerman Eds.), Vol.1, Part 1, Germany, 1977.

[3] Maciejowski, J. M., Multivariable feedback design, Addison-Wesley Publishing company, Wokingham, 1989.

[4] Matko, D., B. Zupančič, R. Karba, Simulation and modelling of continuous systems: a case study approach, Prentice Hall International, Oxford, 1992.

[5] Perry, R.H. and C.H. Chilton, Chemical Engineer's Handbook, Mc Graw Hill Book Company, New York, 1973.

[6] Shinskey, F.G., Distillation Control, Mc Graw Hill Book Company, New York, 1984.

[7] Zupančič, B., et al., SIMCOS - digital simulation language with hybrid capabilities, Proc. 4th Symp. Simulationstechnik, Zürich, Switzerland, (1987), 205-212.

Control model for an additional 4-wheel steering of an automobil

Lugner P., Plöchl M.
University of Technology, Vienna

Abstract: To design an effective control for a 4-wheel-steering that also works at critical driving conditions models of different complexity for the plant, the observer and the desired handling properties have to be used. With the utilization of a recognition scheme and an adaptive observer even for an extreme steering manoeuvre and slippery road surface the simulations show essential improvements compared to a conventional, uncontrolled automobile.

1 Introduction

Besides the already applied rear wheel steering in modern passenger cars, mainly Japanese cars, the possibility of an additional steering of all four wheels could be used to improve the driving behaviour especially in critical situations [1]. The design of a control scheme to utilize the possibilities of such a steering has to be based on a nonlinear model for the plant. Since one of the two main state variables cannot be measured directly an observer, a further model of the vehicle, must be introduced to estimate the missing information. An additional simple transfer function model of the vehicle is used to provide a desired handling behaviour.

So the different physical demands and some mathematical restriction with respect to controller design leads to 3 models of different complexity, see Fig.1.

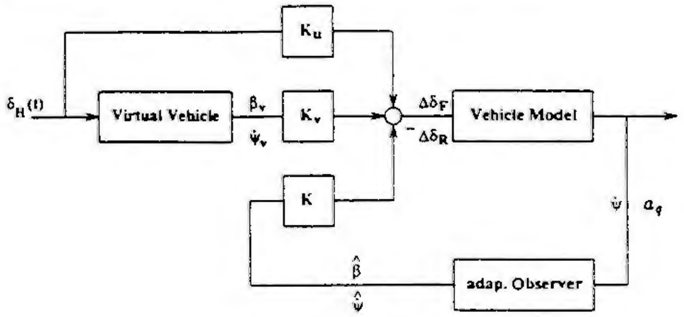
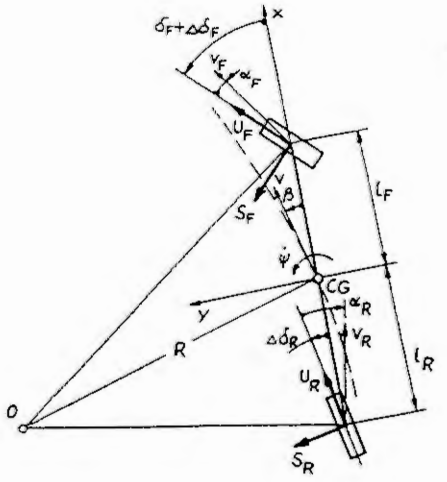


Fig.1 : State model

2 System modelling

Since the main emphases of this investigation is put on the principal possibilities for improvements, even for the plant (vehicle model of Fig.1) simplifications with respect to the vehicle body structure are introduced. So all 3 different models are based on the well documented 2-wheel-vehicle model (and its linearized equations of motion), e.g. [2], Fig.2. The two state variables are the yaw velocity $\dot{\psi}$ and the side slip angle β whereas the additional steer angles $\Delta\delta_F$, $\Delta\delta_R$ are the input quantities. The driver sets the desired path by the steering angle δ_H.

To guarantee sufficient description of the system behaviour for critical situations the nonlinearities of the lateral tyre forces $S_i = S_i(\alpha_i)$, Fig.3, have to be taken into account for the vehicle model.



$$\alpha_F = \delta_F + \Delta\delta_F - \beta - \frac{l_F\dot\psi}{v}, \quad \alpha_R = \Delta\delta_R - \beta + \frac{l_R\dot\psi}{v} \qquad (1)$$

$$ma_q = m \cdot v(\dot\beta + \dot\psi) = S_F + S_R \qquad (2)$$

$$I_{CG}\ddot\psi = S_F l_F - S_R l_R \qquad (3)$$

Fig.2 :
2-wheel-vehicle model (mass m, moment of inertia I_{CG}), kinematics and linearized, simplified equations

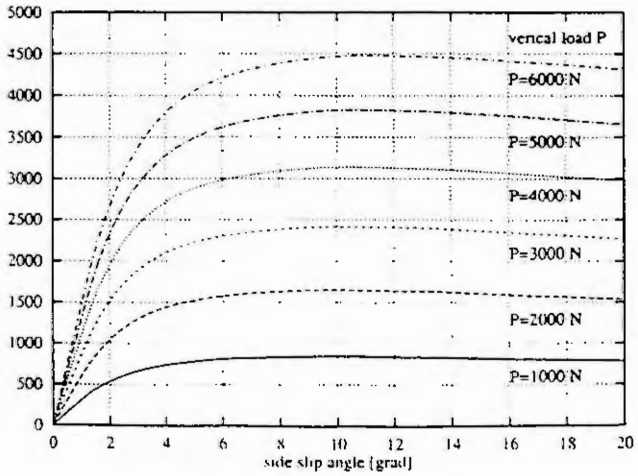


Fig.3 : Lateral tyre force of plant

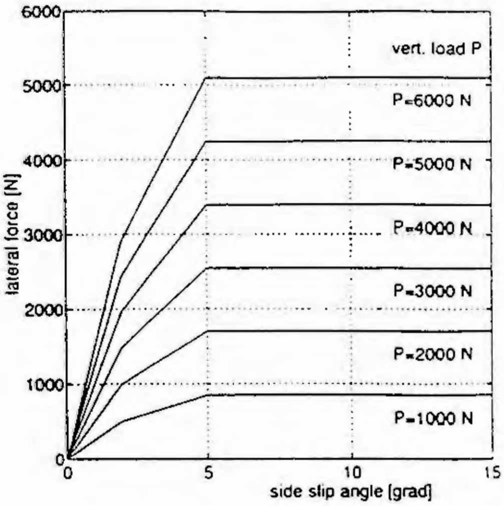


Fig.4 : Lateral tyre force of observer

The yaw velocity $\dot\psi$ can be measured by a gyro-instrument but it is not possible to get reliable information with respect to β. Therefore an estimate $\hat\beta$ must be provided by using an observer [3]. Its mathematical description is derived from the 2-wheel-model now combined with piecewise linearized tyre forces, Fig.4. By this piecewise linear formulation the control feedback K can be calculated by standard MATLAB-software but on the other hand an adaptation scheme, based on the difference of measured yaw velocity $\dot\psi$ and the estimated $\hat{\dot\psi}$, has to select the intervall most suitable to describe the vehicle behaviour.

The virtual vehicle-model, Fig.1, offers the possibility to adapt the general handling behaviour of the vehicle to different desires or demands. For this investigation a driving speed dependent steering ratio was used, so that for steady state cornering the 4WS-car has the desired steering characteristics of a 2WS-car (e.g. a special understeer coefficient k_{us}) [4]. The corresponding reference value $\dot\psi_v$ is calculated using the transferfunction

$$\frac{\dot\psi_v}{\delta_H} = \left[\frac{1}{K_u}\left(\frac{l_F + l_R}{v} + k_{us}\frac{v}{g}\right)\right]^{-1} \qquad (4)$$

that once again is based on the 2-wheel-model, Fig.2, but now with a complete linear description $S_i = c_{\alpha i}\alpha_i$ with the cornering stiffness $c_{\alpha i}$.

For the side slip angle it is assumed, corresponding to experiments [5] that $\beta_v = 0$ is the desired value. The superposition of the filtered (filter K_v) reference values, the handwheel steering input $\delta_H(t)$ transformed with the konstant steering gear ratio K_u and the control feedback quantities provide the additional steering angles $\Delta\delta_F$, $\Delta\delta_R$ for the front and rear wheels.

The goal of the control strategy is to keep $\dot{\psi}$ as close as possible to the reference $\dot{\psi}_v$ and to minimize the side slip angle β if possible to $\beta_v = 0$.

3 Results and Conclusion

As a representative example, for the evaluations a severe lane change manoeuvre on a road with reduced friction $1s$ after the start is chosen. The desired values $\beta_v = 0$, $\dot{\psi}_v$ are assumed to show no influence of this change.

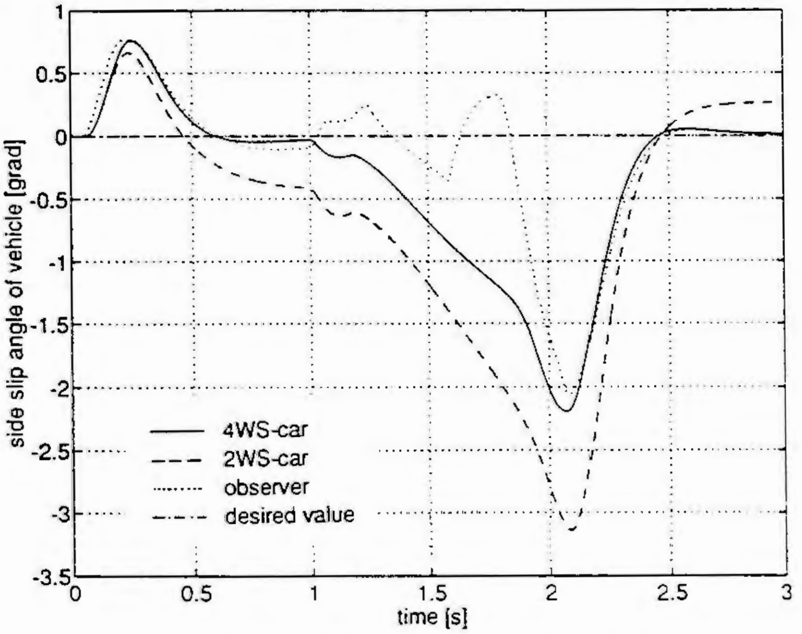


Fig.5 : Side slip angle for a severe lane change manoeuvre

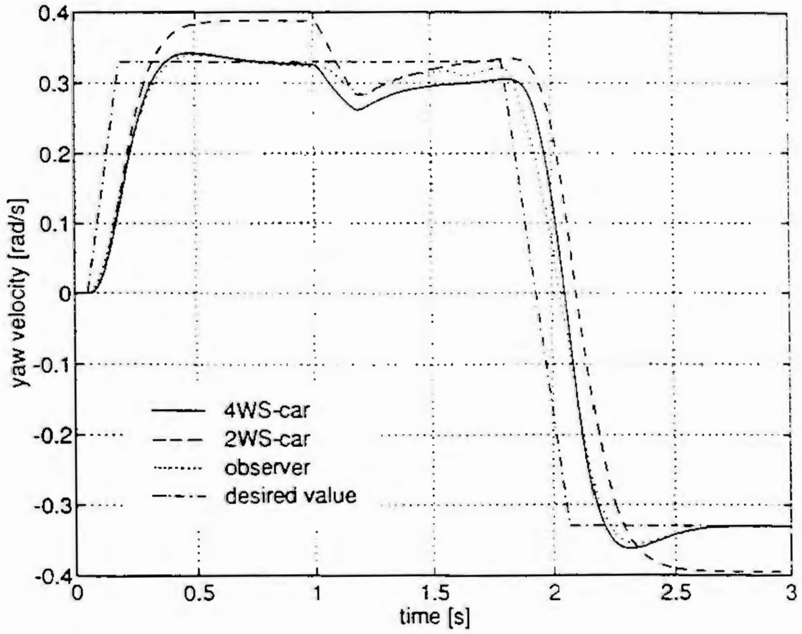


Fig.6 : Yaw velocity

A comparison of Figs.5, 6 shows a reduction in side slip angle β due to control though it is not possible to achieve a β very near or equal to $\beta_v = 0$. Significant improvements especially when the friction changes can also be found for the yaw velocity $\dot{\psi}$.

The estimated values $\hat{\beta}$, $\hat{\dot{\psi}}$ in Fig.6 indicate the switching of the observer characteristics shortly after the vehicle encounters the reduced friction. Greater differences of β to $\hat{\beta}$ result by the kind of the piecewise linearized lateral force, Fig.4, and the applied switching scheme.

Though the control design and the model description of the observer are not fine tuned and optimized, the investigation proves that significant improvements of the vehicle behaviour even for critical conditions are possible. Furthermore a desired vehicle handling can be superimposed — a great advantage for future vehicle concepts.

References

[1] Lugner P., Mittermayr P.: Controlled Additional 4-wheel Steering at Critical Driving Conditions Proceedings of the Int.Symp.on Advanced Vehicle Control, p.245 ff.

[2] Kortüm W., Lugner P.: Systemdynamik und Regelung von Fahrzeugen Springer Verlag, 1994

[3] Föllinger O.: Regelungstechnik, Hüthig Buchverlag, 1992

[4] Senger H.: Dynamik und Regelung allradgelenkter Fahrzeuge, VDI Fortschrittberichte, Reihe 12, Nr.126, 1989

[5] Nakaya H. and Oguchi Y.: Characteristics of the four-wheel steering vehicle and its future prospects Int.J.of Vehicle Design, vol.8, no.3, 1987

MODELLING THE EFFECT OF PROCESS VARIABLES ON SHEET METAL FORMABILITY

W.M. SING and K.P. RAO

Department of Manufacturing Engineering
City Polytechnic of Hong Kong, Kowloon, Hong Kong

Abstract. Theoretical influence of the process variables (strain rate and temperature) on the sheet metal formability near room temperature has been modeled following our recently proposed forming limit stress curve prediction method, which involved the use of Levy-Mises flow law and Hill's anisotropic yield criterion. The results obtained have compared favourably with the experimental observations on a low carbon steel.

1. INTRODUCTION

In deep drawing, experimental observations have found that the fracture position, which occurs usually at the punch radius of a deep drawing cup, can be shifted to nearby position due to a selective cooling of the punch and the drawing ratio can be increased consequently. Also, a positive strain rate sensitivity is favourable in postponing failure after the occurrence of diffuse necking, by reducing the tendency of strain localization. On the other hand, heat generated either due to plastic deformation or intentional external source can accelerate the necking [10]. In sheet metal forming, the heat released from plastic deformation and the heat generated from friction can contribute to the rise in temperature. Especially in high speed forming, temperature increases rapidly when compared to low speed forming. A detailed knowledge of the influence of these processing parameters on the stress and strain relationship is important in the analysis of sheet forming limits.

2. THERMAL HARDENING

Most of the available experimental observations on plain carbon steels have shown that the flow stress as well as drawability decrease with increasing temperature [2,3,7] while the flow stress being dependent on strain rate [8]. Several experimental works have found that selective heating and cooling can improve the drawability. During deep drawing forming, the flange portion can be heated to reduce the flow stress and increase the intrinsic workability which assists the material flow into the cavity [3,9]. Also, higher drawing ratio and elimination of breakage can be achieved by cooling the punch nose [3].

Notation:

a	- Exponent of Hill's yield criterion	$\varepsilon_1,\varepsilon_2$	- Major and minor strains
c	- Specific heat of the material	ε_d	- Diffuse necking strain
ΔE	- Work	$\dot{\varepsilon}$	- Equivalent strain rate
K	- Strength coefficient	$\dot{\varepsilon}_1$	- Strain rate in tension
m	- Strain rate sensitivity	η	- Mechanical equivalent of heat
n	- Strain hardening exponent	ρ	- Density of the material
R	- Anisotropy value	σ	- Equivalent stress
T	- Temperature	σ_1,σ_2	- Major and minor stresses
β	- Temperature hardening coefficient	θ	- Temperature difference
ε	- Equivalent strain	λ	- Stress-strain ratio (Eq.12)

2

Flow stress of steel near room temperature can be expressed [6] using a linear relationship with respect to the isothermal conditions using:

$$\sigma = K\,\varepsilon^{n}\,\dot{\varepsilon}^{m}\,(1-\beta\theta) \qquad (1)$$

where β is a phenomenological coefficient measuring thermal hardening and θ is the temperature increase from a reference value. This equation, though can not be justified over a wide range of temperature, can be used in sheet metal presswork which is usually performed in a limited range around room temperature.

2.1 Influence of temperature on instability

When Eqn. (1) is used in the instability analysis:

$$\frac{d\sigma}{d\varepsilon} = \left(\frac{\partial\sigma}{\partial\varepsilon}\right) + \left(\frac{\partial\sigma}{\partial\dot{\varepsilon}}\right)\left(\frac{d\dot{\varepsilon}}{d\varepsilon}\right) + \left(\frac{\partial\sigma}{\partial T}\right)\left(\frac{dT}{d\varepsilon}\right) \qquad (2)$$

where, $$\frac{\partial\sigma}{\partial\varepsilon} = n\,K\,\varepsilon^{(n-1)}\,\dot{\varepsilon}^{m}\,(1-\beta\theta) \qquad (3)$$

$$\frac{\partial\sigma}{\partial\dot{\varepsilon}} = m\,K\,\varepsilon^{n}\,\dot{\varepsilon}^{(m-1)}\,(1-\beta\theta) \qquad (4)$$

$$\frac{\partial\sigma}{\partial T} = \frac{\partial\sigma}{\partial\theta}\,\frac{\partial\theta}{\partial T} = -K\,\varepsilon^{n}\,\dot{\varepsilon}^{m}\,\beta \qquad (5)$$

Substitute Eqns. (3) to (5) into (2),

$$\frac{d\sigma}{d\varepsilon} = K\,\varepsilon^{n}\,\dot{\varepsilon}^{m}\,(1-\beta\theta)\left\{\frac{n}{\varepsilon} + \left(\frac{m}{\dot{\varepsilon}}\right)\left(\frac{d\dot{\varepsilon}}{d\varepsilon}\right) - \beta\left(\frac{dT}{d\varepsilon}\right)\Big/(1-\beta\theta)\right\} \qquad (6)$$

Following the diffuse instability criterion, the maximum stress in uniaxial tension can be obtained when $d\sigma_1/d\varepsilon = \sigma_1$, and the corresponding strain is:

$$\varepsilon_d = n \Big/ \left\{1 + \beta\left(\frac{dT}{d\varepsilon_1}\right)\Big/(1-\beta\theta) - \left(\frac{m}{\dot{\varepsilon}_1}\right)\left(\frac{d\dot{\varepsilon}_1}{d\varepsilon_1}\right)\right\} \qquad (7)$$

Assuming constant strain rate, Eq. (7) becomes,

$$\varepsilon_d = n \Big/ \left\{1 + \beta\left(\frac{dT}{d\varepsilon_1}\right)\Big/(1-\beta\theta)\right\} \qquad (8)$$

The term $(dT/d\varepsilon_1)$ is positive under adiabatic conditions and normally encourages instability; increasing temperature leading to lower diffuse instability strain value (ε_d). The rise in temperature can be expressed in terms of work:

$$\Delta E = \eta\,\sigma\,\Delta\varepsilon = \rho\,c\,\Delta T \qquad (9)$$

$$\frac{dT}{d\varepsilon} = \frac{\eta\,\sigma}{\rho\,c} \qquad (10)$$

2.2. Case study

Following the FLC prediction proposed by the authors [11], the influence of the variation of temperature on the FLC is demonstrated. Firstly, the tensile properties of AK steel, based on the work of Hecker [5], have been used in this study to obtain the forming limit stress curve (FLSC) for the material. The effect of temperature is introduced into Hill's anisotropic yield criterion:

$$|\sigma_1 + \sigma_2|^{a} + (1+2R)\;|\sigma_1 - \sigma_2|^{a}) = 2\;(1+R)\;\left[K\varepsilon^{n}(1-\beta\theta)\right]^{a} \qquad (11)$$

Levy-Mises equation is,

$$d\lambda = d\varepsilon \Big/ \left(2(1+R)\;\left[K\;\varepsilon^{n}\;(1-\beta\theta)\right]^{(a-1)}\right) \qquad (12)$$

The limit strains have been obtained using the following equations:

$$d\varepsilon_1 = \Big[(1+2R)|\sigma_1 - \sigma_2|^{(a-1)} + |\sigma_1 + \sigma_2|^{(a-1)}\Big] d\lambda \qquad (13)$$

$$d\varepsilon_2 = \Big[-(1+2R)|\sigma_1 - \sigma_2|^{(a-1)} + |\sigma_1 + \sigma_2|^{(a-1)}\Big] d\lambda \qquad (14)$$

The influence of temperature is shown in Fig. 1(a) for a thermal hardening coefficient (β) of 0.0015. At this value of β, all the predicted FLCs, corresponding to various temperatures lie closer to the experimental FLC [5], with higher temperatures decreasing the formability, which is consistent with the predictions of Eqn. (8) and typical experimental behaviour of steels. It should be noted that θ can be positive or negative depending on whether there is a decrease or increase in the temperature. Results in Fig. 1(b) clearly indicate that increasing thermal hardening coefficient leads to increased forming limits.

3. STRAIN RATE HARDENING

In forming processes, the speed of plastic deformation, tool geometry, material surface condition and lubrication can cause the variation of strain rate locally. It is well accepted that higher material thickness can induce higher strain gradient across the thickness leading to increased formability. For a positive rate sensitive material, if an element is straining faster than hardening of an adjacent element, it can reduce the difference of strain and the tendency of strain localization. As a result, the increased strain rate hardening gives a more uniform distribution of strains under stretching and the failure strain level can be extended. Experimental results have shown [4] that the flow stress and total elongation are sensitively dependent on the strain rate, strain rate sensitivity and the type of stretching. Campbell [1] has found that the strain rate sensitivity depends on the local stress and strain values. Because of the close relationship between the position of forming limit curve and the value of n, which in turn has interaction between the material properties and the forming conditions, the strain rate has considerable effect on the formability.

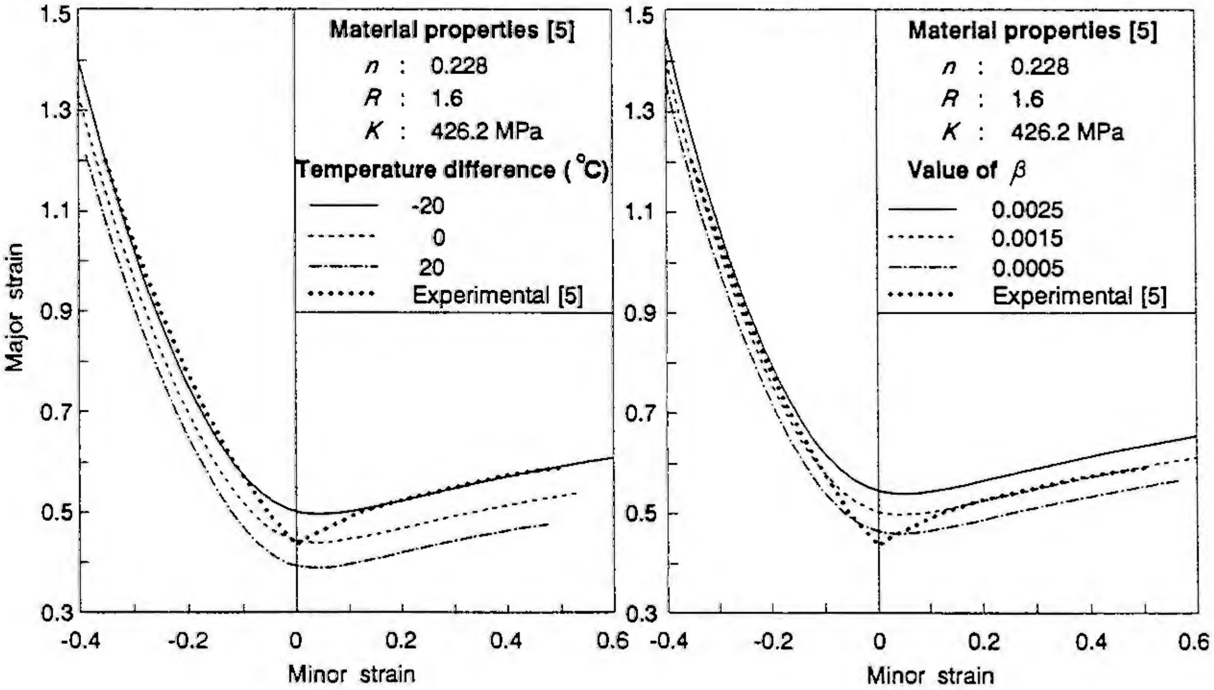


Fig. 1 FLCs based on the FLSC prediction method [11] using a theoretical limit stress of 378.3 MPa, derived from the tensile properties. Influence of (a) temperature difference (θ), and (b) thermal hardening coefficient (β).

3.1. Influence of strain rate on instability

A multiplicative type power strain hardening flow stress relationship, which is valid for deep drawing type mild steel [6], has been adopted in modelling the strain rate associated influence on the forming limits, and has the form:

$$\sigma = K\, \varepsilon^{n}\, \dot{\varepsilon}^{m} \tag{15}$$

Mathematical treatment of this equation yields the following forms:

$$\ln\sigma = \ln K + n\, \ln\varepsilon + m\, \ln\dot{\varepsilon} \tag{16}$$

$$\mathrm{d}\ln\sigma/\mathrm{d}\ln\varepsilon = n + m\, \mathrm{d}\ln\dot{\varepsilon}/\mathrm{d}\ln\varepsilon \tag{17}$$

$$(\mathrm{d}\sigma/\sigma)\, /\, (\mathrm{d}\varepsilon/\varepsilon) = n + m\, \mathrm{d}\ln\dot{\varepsilon}/\mathrm{d}\ln\varepsilon \tag{18}$$

where $$m = \partial\ln\sigma/\partial\ln\dot{\varepsilon} \tag{19}$$

Diffuse instability occurs when the load reaches a maximum in the tensile test. The corresponding strain (ε_d) can be obtained by using Eqns. (15) and (18):

$$\varepsilon_d = n + m\, \mathrm{d}\ln\dot{\varepsilon}_1/\mathrm{d}\ln\varepsilon_1 \tag{20}$$

Theoretically Eqn. (20) indicates that positive strain rate sensitivity and higher strain rate can improve the formability.

3.2. Case study

The effect of strain rate is introduced into Hill's quadratic anisotropic yield criterion by substituting the assumed flow stress relationship (Eqn.15) with the exponent a equal to 2:

$$|\sigma_1 + \sigma_2|^{a} + (1{+}2R)\, |\sigma_1 - \sigma_2|^{a}) = 2\, (1{+}R)\, \left[K\, \varepsilon^{n}\, \dot{\varepsilon}^{m}\right]^{a} \tag{21}$$

The corresponding Levy-Mises equation takes the form:

$$\mathrm{d}\lambda = \mathrm{d}\varepsilon \Big/ \left(2(1{+}R)\, \left[K\, \varepsilon^{n}\, \dot{\varepsilon}^{m}\right]^{(a-1)}\right) \tag{22}$$

The limit strains have been obtained once again using Eqns. (13) and (14).

The influence of the variation of strain rate is shown in Fig. 2(a) for a strain rate sensitivity of 0.006 which is typical for this kind of steel. Obviously, the FLC can be shifted to higher forming limits by increasing the strain rate. However, at this value of m, all the predicted FLCs corresponding to various strain rates are very close to the experimental results. However, when the strain rate sensitivity is increased to 0.018, the predicted FLCs, as shown in Fig. 2(b), clearly demonstrate much larger influence of the strain rate on the material's forming limits. Generally, the predicted FLCs are in good agreement with the experimental observations for mild steel sheet in such a way that higher strain rate and strain rate sensitivity are in favour of higher formability, just as Eqn. (20) predicts theoretically. Because of the inevitable occurrence of the thickness strain gradient during actual forming or on the scribed circle test, there will be some deviation between the experimental results and the present predictions which are essentially based on plane stress condition.

4. CONCLUSIONS

The FLC prediction method used in this study demonstrated the dynamic influence of the material properties on sheet metal forming limits. With the variations in the process parameters during plastic deformation, a forming limit band rather than a single curve can be obtained. Lower temperature during forming can improve the formability where as positive strain rate sensitivity and higher strain rate have apparent favourable influence on the FLC.

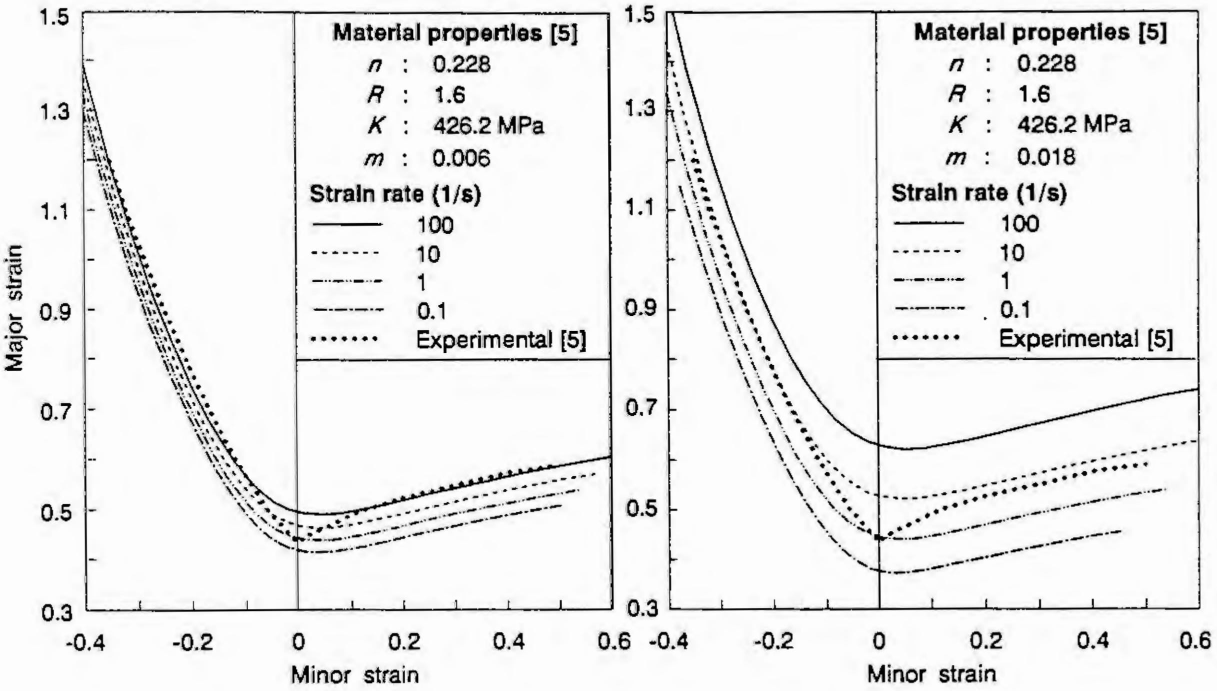


Fig. 2 FLCs based on the FLSC prediction method [11] using a theoretical limit stress of 378.3 MPa, derived from the tensile properties. Influence of the strain rate at different values of the strain rate sensitivity.

5. REFERENCES

[1] Campbell, J.D., Plastic Instability in Rate-dependent Materials, J. Mech. Phys. Solids, 15 (1967) 359-370.

[2] Demeri, M.Y. and H. Conrad, Influence of Temperature on the Material Parameters in the Forming of Sheet Steel, Sheet Metal Ind., 61 (1981), 191-202.

[3] Granzow, W.G., The Influence of Tooling Temperature on the Formability of Stainless Steel Sheets, In: J.R. Newby and B.A. Niemeier (Ed.), Formability of Metallic Materials 2000 A.D., ASTM Publication 753, Philadelphia, 1982.

[4] Green, S.J., J.J. Langan, J.D. Leasia and W.H. Yang, Material Properties, Including Strain-rate Effects, as Related to Sheet Metal Forming, Metall. Trans., 2A (1971), 1813-1820.

[5] Hecker, S.S., Simple Technique for Determining Forming Limit Curves, Sheet Metal Ind., 61 (1975), 671-676.

[6] Korhnonen A.S. and H.J. Kleemola, Effects of Strain Rate and Deformation Heating in Tensile Testing, Metall. Trans. 9A (1978), 979-986.

[7] Liu, Y., J. Zhu and H. Zhou, Variation of Yield Stress with Strain Rate for Three Carbon Steels, J. of Eng. Mat. Techn., 114 (1992), 348-353.

[8] Ohwue, T., M. Usuda and S.Sudo, Cooled-punch Deep Drawing and its Application to Automobile Parts, 17th Biennial Congress 1992 IDDRG, 308-315.

[9] Pearce, R., Sheet Metal Forming, Adam Hilger, Bristol, 1991.

[10] Semiatin, S.L., R.A. Ayres, J.J. Jonas, An Analysis of the Nonisothermal Tensile Test, Metall. Trans., 16A (1985), 2299-2308.

[11] Sing W.M. and K.P. Rao, Prediction of Sheet-metal Formability Using Tensile-test Results, J. Mat. Proc. Tech., 37 (1993), 37-51.

AERODYNAMIC INTERACTION OF HIGHLY COMPLIANT PLATES

Xavier J. R. AVULA
University of Missouri-Rolla
Rolla, Missouri 65401, U.S.A.

Abstract. The problem of a flexible plate moving in a fluid medium is investigated. The classical aeroelasticity problem dealing with small deformations of a plate under the action of aerodynamic forces is revisited by introducing low bending rigidity of the moving plate. Further, the plate is considered as an axially moving material with increasing mass coming into contact with the surrounding air. The deformed configuration and the leading edge trajectory of the plate are calculated and compared with experimental results.

1. INTRODUCTION

Thin plates are used on aircraft structures as deflectors and lift augmentation devices. In recent years, they have been viewed as control surfaces that can be manipulated to achieve stability of moving bodies. They are also used for similar purposes on hydrodynamic machines. Investigations of problems associated with axially moving, highly compliant materials such as strings, bandsaws, belts, and paper sheets have been published by Carrier [1], Wickert and Mote [2]. In this investigation, a mathematical model of a thin plate emerging along its length and subjected to inertial and steady, incompressible aerodynamic forces is constructed and solved by using a semidiscrete finite element method. The plate is considered linearly elastic but highly compliant and capable of undergoing large deflections with very low bending rigidity. The evolution of the deformed configuration of the plate and the trajectory of its leading edge are determined.

2. MATHEMATICAL MODEL

In Fig. 1 are depicted an axially emerging thin plate and its coordinates. The internal and external forces and moments acting on an emerged portion of the thin plate are shown in Fig. 2. The equation of motion in the x-direction is

$$\frac{\partial \overline{H}}{\partial X} - f_d(X,T) + f_{su}\delta(X-L)\cos\beta(L,T) - f_{IM}\cos\beta = mb\frac{\partial^2 \overline{x}}{\partial T^2} \quad (1)$$

where $\delta(\bullet)$ is a Dirac delta function. The equation of motion in the y-direction is

$$\frac{\partial \overline{V}}{\partial X} - mbg + \mathcal{L}(X,T) + f_{su}\delta(X-L)\sin\beta(L,T) - f_{IM}\sin\beta = mb\frac{\partial^2 \overline{y}}{\partial T^2} \quad (2)$$

where $f_d(X,T)$, $\mathscr{L}(X,T)$, f_{su}, f_{IM} are the drag, lift, leading edge suction, and initial impulse, respectively.

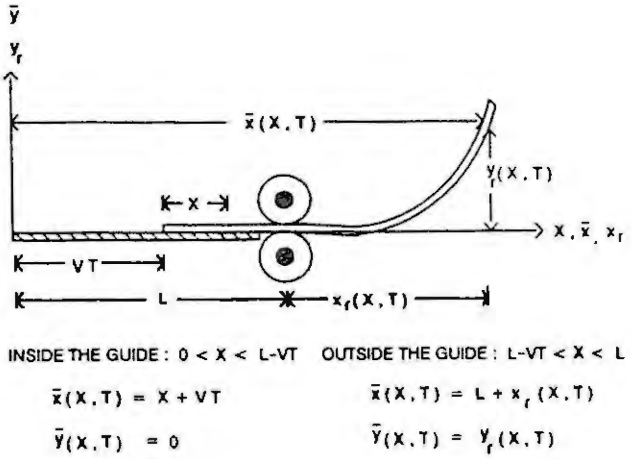


Fig.1. Axially emerging compliant plate.

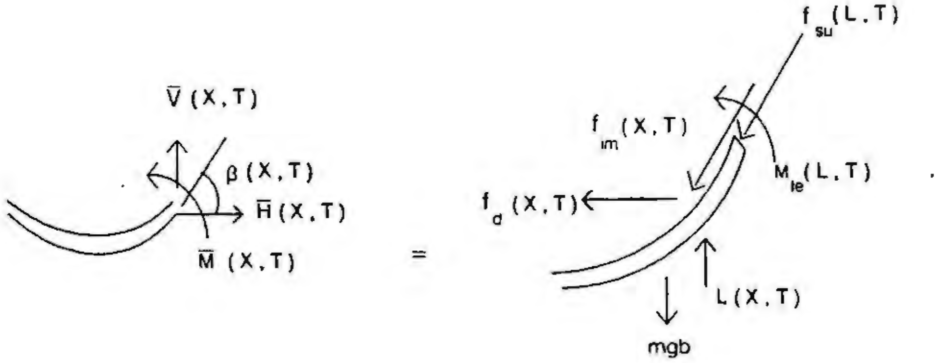


Fig. 2. Free-body diagram of a segment of emerged plate.

Using the notation

$$\frac{\partial(\cdot)}{\partial T} = (\cdot)_T \, , \quad \frac{\partial^2}{\partial T^2}(\cdot) = (\cdot)_{TT} \text{ etc.}$$

Eq. (2) can be integrated and expressed in the form

$$\bar{V}(X,T) = mbg\int_L^X d\xi - \int_L^X \mathcal{L}(\xi,L)\,d\xi + \int_L^X f_{su}\delta(\xi,L)\sin\beta(L,T)\,d\xi + \int_L^X f_{IM}\sin\beta(\xi,T)\,d\xi + mb\int_L^X \bar{y}_{TT}(\xi,T)\,d\xi \qquad (3)$$

where

$$\int_L^X \mathcal{L}(\xi,T)\,d\xi = -m\mu b\left[\int_L^X \bar{y}_{TT}(\xi,T)\,d\xi + 2V\int_L^X \bar{y}_{\xi T}(\xi,T)\,d\xi + V^2\int_L^X \bar{y}_{\xi\xi}(\xi,T)\,d\xi\right] \qquad (4)$$

$$\int_L^X f_{SU}\ \delta(\xi,L)\sin\beta(L,T)d\xi = \frac{mb\mu}{2}\ [Q^2_{XT}(L,T)$$

$$+\ 2VQ_{XT}(L,T)\ \sin\beta(L,T)\ +\ V^2\ \sin^2\beta(L,T)]\sin\beta(L,T) \qquad (5)$$

$$\int_L^X f_{IM}\ \sin\beta(\xi,T)d\xi = mb\mu V^2 k_9 T^{-1/2} \int_L^X \sin\beta(\xi,T)d\xi$$

$$= mb\mu V^2 k_9 T^{-1/2}[Q_X(X,T)\ -\ Q_X(L,T)] \qquad (6)$$

$$mb \int_L^X \overline{y}_{TT}(\xi,T)d\xi\ =\ mbQ_{TT}(X,T)\ . \qquad (7)$$

Introducing the above integrals, the equations of motion, Eqs. (1) and (2) become

$$\overline{H}(X,T)\ =\ mb\Bigg[D_1\ -\ \frac{\mu}{2}\ \{Q^2_{XT}(L,T)\ +\ 2VQ_{XT}(L,T)\ \sin\beta(L,T)$$

$$+\ V^2\sin^2\beta(L,T)\}\ \cos\beta(L,T)\ +\ k_9\mu\ V^2T^{-1/2}\{P_X(X,T)$$

$$-\ P_X(L,T)\}\ +\ P_{TT}\Bigg] \qquad (8)$$

where

$$D_1\ =\ \mu \int_L^X \{k_4\ +\ k_5T\beta^2(\xi,T)\}d\xi \qquad (9)$$

μ is the added mass ratio given by $\pi\rho b/4m$ in which b is the width of the plate, ρ is the mass density of the fluid, and m is the mass per unit area of the plate; P and Q are deformation potentials; and

$$\overline{V}(X,T)\ =\ mb\Bigg[g(X-L)\ +\ (1\ +\ \mu)Q_{TT}\ +\ 2\mu V\ \{Q_{XT}(X,T)\ -\ Q_{XT}(L,T)\}$$

$$+\ \mu V^2\{\sin\beta(X,T)\ -\ \sin\beta(L,T)\}\ -\ \frac{\mu}{2}\ \{Q^2_{XT}(L,T)$$

$$+\ 2VQ_{XT}(L,T)\ \sin\beta(L,T)\ +\ V^2\ \sin^2\beta(L,T)\}\ \sin\beta(L,T)$$

$$+\ k_9\ \mu V^2T^{-1/2}\{Q_X(X,T)\ -\ Q_X(L,T)\}\Bigg]\ . \qquad (10)$$

The initial and boundary conditions are

at $t = 0$, $\beta = 0$

at $X = L - VT$, $\beta = Q_X = 0$, $P_X = L$, and (11)

at $X = L$, $P = Q = 0$, and $\beta_X = \frac{M_{1e}}{B*}$.

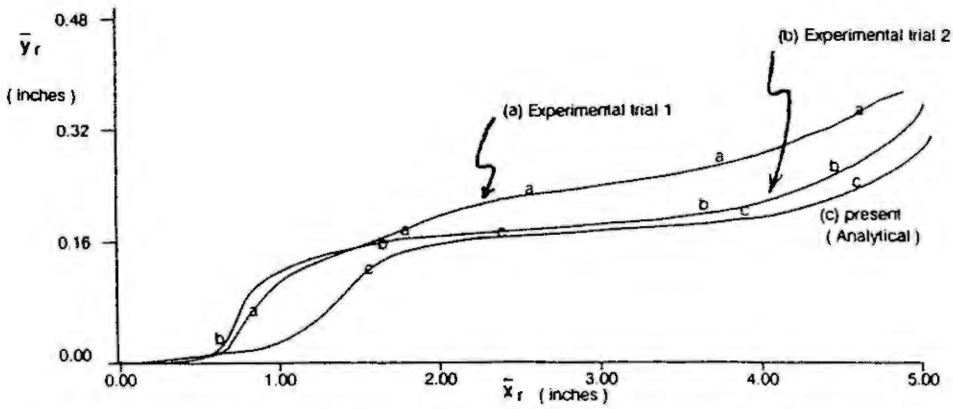


Fig. 3. Leading edge trajectory of a compliant plate under aerodynamic forces.

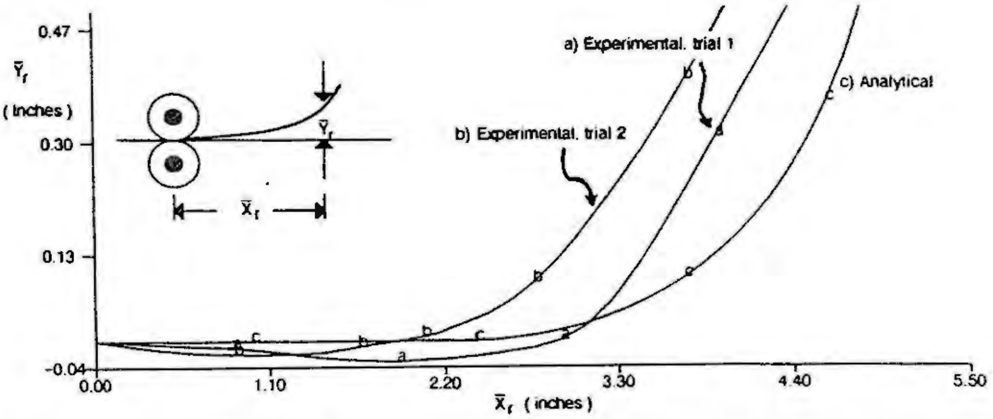


Fig. 4. Deformed profiles of a compliant plate at a fixed time under aerodynamic forces.

SOLUTION AND DISCUSSION

Equations (8) and (10) along with (11) are solved by a semi-discrete finite element method using the finite difference for the time domain and finite elements for space discretization. Limitations on the length of this manuscript preclude a number of details about computations and experimental verification. The motion of a finite strip of paper is considered because the strip has a low bending rigidity and it is easily deformed by the aerodynamic forces. The computed trajectory of the leading edge with respect to time and the complete configurations at a fixed time are shown in Figs. 3 and 4 in comparison with experimentally determined values.

The computational results have established the trends in the deformation characteristics of the compliant plates under aerodynamic forces reasonably well. The leading edge divergence can be attributed to the discrepencies in material charcaterization and nonlinearities.

REFERENCES

[1] Carrier G.F., The Spaghetti Problem. Am. Math. Monthly, 56 (1949), 669-672.

[2] Wickert, J.A. and Mote, C.D.,Jr., Current Research on the Vibration and Stability of Axially Moving Materials. EUROMECH 223, Tampere, Finland, June 16-18 (1987).

MODEL BUILDER'S LOOK AT VARIATIONAL PRINCIPLES IN CONTINUUM MECHANICS

Józef PIETRUCHA
Warsaw University of Technology
Institute of Aeronautics and Applied Mechanics
ul. Nowowiejska 24, 00-665 Warsaw, Poland

Abstract. The primary aim of this paper is to argue into a variational formulation (VF) as an unified approach to the mathematical modelling of complex mechanical phenomena (CMP). Two opposed treatments of a VF (derivation vs postulation) are presented. Existing solutions of the inverse problem of the calculus of variations are discussed. Special attention is paid to the Sedov's variational equation as the basic tool of the CPM modelling. If a phenomenon is complex enough, only the approach based on postulating may be implemented to the formulation of CMP.

1. INTRODUCTION

As is well known, variational principles (VPs) play an important role in classical and quantum mechanics. However, complete agreement has not yet been reached concerning VPs of continuum mechanics (CM). Truesdell and Toupin [1] in 1960 pointed out that "the lines of thought which have led to beautiful variational statements for systems of mass-points have been applied in continuum mechanics also, but only rarely are the results beautiful and useful." Since that time a lot has been done, particularly by Sedov and his collaborators, with great contribution of Berdichevsky [2].

VPs in CM have been studied for a long time. The most efficient numerical methods of solving complicated differential equations in CM are those based upon VF. There are many papers in this area. However, in this paper full attention will be focused on VF as a tool for an elegant and rational creation of causal mathematical models. The main goal of mathematical modelling is to translate a physical phenomenon into a mathematical description, which can take either differential or variational form.

The variational description is very general, because it may be applied not only for systems of mass-points, but also for thermal, electromagnetic, biological systems, etc. The generality of such description makes it possible to treat in a unifying way various disciplines of knowledge, thereby inducing a search for analogies, so important in the times of raging specialization, leading to the disintegration. Thus, the primary aim of this paper is to argue that a VF is a unified approach to the mathematical modelling of CMP.

2. TO DERIVE OR TO POSTULATE ?

The search for a VP is equivalent to so called "*inverse problem of the calculus of variations*": that is, finding the functional the Euler-Lagrange equations ofwhich are equivalent to the governing equations of the given physical problem. But now a fundamental question arises: does such functional exist at all?. It appears, however, that the equivalence of a VP with the original differential equations (and their boundary conditions), cannot be expected in all problems. For example, Finlayson [3] proved that the full Navier-Stokes equation (i.e., with the inertial terms present), cannot be derived from a classical VP (see Chap.3). The present work was motivated by a possibility of another understanding of VF, which is explained below.

If a thought construct were to be recognized as a theory, it must be founded on axioms. Then, through deduction one formulates theorems and conclusions. If these conclusions are in agreement with facts known from other sources, the theory acquires "citizenship rights". From the logic point of view it is quite inessential which theorems are considered as axioms, and which are conclusions derived therefrom. Thus, the VPs can be either derived (approach **D**), or postulated (approach **P**).

When we are dealing with well known problems of classical mechanics the advantage of any of the two approaches cannot be seen. However, in modern mechanics we are often facing multiple coupling phenomena (e.g., active control of aerothermoelasticity effects). In such situations we rather have no choice, as we do not know *a priori* the considered phenomenon. Therefore, we cannot refer to the procedure conforming with the **D** approach.

An elementary example of the **P** approach is discussed in [4]: Snell's law of refraction is deduced from Fermat's principle, which is postulated. A very instructive exercise is to derive the Lamé equation from the postulated Hamilton's principle. A more sophisticated and up to date approach represents the so called "*Sedov's variational equation*", which is considered in Chap.5.

3. CLASSICAL VARIATIONAL PRINCIPLES

As was mentioned in Chap. 2, the problem of existence and formulation of VPs for systems of differential equations belongs to the inverse problem of the calculus of variations. Consider here the differential equation in the form

$$N(u) = 0, \tag{1}$$

which can be nonlinear. Assuming that an operator N is the gradient of a functional F, i.e.,

$$N(u) = \operatorname{grad} F(u), \tag{2}$$

the variation of the $F(u)$ due to a variation in u is

$$\delta F = \int_{\tau} N(u)\, \delta u \, d\tau = 0. \tag{3}$$

Clearly, if $F(u)$ is the functional in a VP, then Eq.(1) will be its Euler-Lagrange equation. Such VP is called *classical*. Thus, the inverse problem is to find $F(u)$ such that $N(u)$ provides the Eq.(1). (Notice: in the above formulation we ignored complications introduced by boundary conditions, which are easily treated in the context of specific applications).

The inverse problem was solved in principle by Vainberg in 1956, but his results were published in English about ten years later [5]. Among nonlinear operators, those for which a VP, not necessarily yielding an extremum, can be built are referred to as "potential operators". The necessary and sufficient condition for an operator to be potential is basically that its Gâteaux differential should exhibit a certain symmetry property. Unfortunately, only a few of the differential equations of interest to scientists and engineers are potential operators. But, if the operator is not potential, one has to spend an additional effort to obtain a VP, see below.

4. MODIFIED VARIATIONAL PRINCIPLES

The term "VP" in the literature has been stretched to include many different classes of functionals, and thus, its meaning has become somewhat fuzzy. We use the term "classical VP" merely for the formalism described in Chap.3. For other cases we will use the term "*modified VP*". For the sake of completeness, we will give brief comments of such VPs here.

Atherton and Homsy [6] showed that even if the operator does not possess a potential, special forms of VP may be formulated. They introduced three types of VPs: *potential*, *alternate potential*, and *composite* ones. Potential principles are those for which the equations, as written, admit a VF. Alternate potential principles are those for which the equations admit a VF only after a differential transformation of variables. Composite principles are those in which, in addition to the original variables, a set of adjoint variables is defined. The most illuminating result is: a composite VP may always be formulated; any differential operator may be recovered as a part of the set of Euler equations for such VP.

Adjoint formulation appeared in the famous PMP - Pontryagin's Maximum Principle [7]. It should be noted, however, that the development of the PMP was a substantial generalization of the early works in the calculus of variations and it is a postulate rather than a VP (in the sense of Atherton and Homsy).

Finlayson [3] has introduced a type of VP, which he called a "*restricted* VP". It differs from an adjoint VP because during variation the time derivative is held fixed. By the way, we would like to pay attention to the interesting Finlayson's result: the restricted VP is equivalent to the Galerkin method and this method is always applicable because it does not depend on the existence of a VP.

Oden and Reddy [8] have developed a theory of *complementary* and *dual* VPs. This principles occupy an important place in mechanics, but mainly due to their utility for establishing bounds on approximate solutions of various class of linear boundary- and initial-value problems. So, they play rather secondary role in the mathematical modelling.

In summary it may be concluded: since all presented modified VPs are the result of the **D** approach, they cannot be treated as the guide of the CMP modelling.

5. SEDOV'S VARIATIONAL EQUATION

First, it should be emphasized that Sedov [9] introduced in 1965 a substantially weaker definition of VP; i.e., the functional equation obtained by assuming equal to zero the sum of volume and surface integrals that include variations of both the integrated functions and the integration regions. Of course, such definition is typical for the **P** approach, and therefore we introduce the notion of *VP in a broad sense*. Now, following the concept of Sedov one can write down his "basic equation" in the form

$$\delta \int_{\tau} L\, d\tau + \delta W^{*} + \delta W = 0 \,, \tag{4}$$

where $d\tau$ is the element of the 4-D Riemannian space-time volume τ bounded by the 3-D surface σ, L is the Lagrangian density, W is the given integral over τ, and W is the unknown integral over σ.

In order to facilitate the interpretation of Eq.(4), let us take into consideration the law of motion of continuous medium in the standard form

$$x^{i} = x^{i}(\, X^{1}, X^{2}, X^{3}, X^{4} \equiv t \,) \,, \tag{5}$$

where spatial and material coordinates are denoted by x^{i} and X^{i}, respectively. Now, the integrated functions have to be specified. For the sake of simplicity let us assume that

$$L = L\,(\, \mu^{k},\, \mu^{k}_{i} \,) \,, \tag{6}$$

$$\delta W^{*} = \int_{\tau} (\, Q_{k} + \frac{\partial Q^{j}_{k}}{\partial x^{j}} \,)\; \delta\, \mu^{k}\, d\tau \,, \tag{7}$$

$$\delta W = \int_{\sigma} P_{k}\; \delta\, \mu^{k}\, d\sigma \,, \tag{8}$$

where μ^{k} are so called *determining parameters*, Q_{k} and Q^{j}_{k} are the mass and surface generalized forces, respectively, and P_{k} are also surface generalized forces, but characterizing the internal interactions.

Performing the variations in the Eg.(4), taking into account the Eqs.(6), (7) and (8), and assuming that $\delta x^{i} = 0$ and $\delta\mu^{k} = 0$ over σ and are arbitrary inside τ, the Euler-Lagrange equations take the form

$$\frac{\partial}{\partial x^{j}} \left(\frac{\partial L}{\partial x^{i}_{l}}\, x^{j}_{l} \right) + \frac{\partial L}{\partial x^{l}_{j}} \frac{\partial x^{l}_{j}}{\partial x^{i}} = Q_{i} + Q_{k}\; \mu^{k}_{i} \,, \tag{9}$$

where

$$Q_{k} \equiv \frac{\partial}{\partial x^{j}} \frac{\partial L}{\partial \mu^{k}_{j}} - \frac{\partial L}{\partial \mu^{k}} \,, \tag{10}$$

and $x^{i}_{j} = \partial x^{i} / \partial X^{j}$ (i,j = 1, 2, 3, 4), and $\mu^{k}_{i} = \partial \mu^{k} / \partial x^{i}$ (k = 5,6,...,n).

For arbitrary variations (but not equal to zero) on any hypothetical surface inside τ we have

$$\mu^{k}_{i} \frac{\partial L}{\partial \mu^{k}_{j}} - x^{j}_{l} \frac{\partial L}{\partial x^{l}_{i}} = L\, \delta^{j}_{i} + P^{j}_{i} + Q^{j}_{i} \,, \tag{11}$$

$$- \frac{\partial L}{\partial \mu^{k}_{j}} = P^{j}_{k} + Q^{j}_{k} \,, \tag{12}$$

where δ^{i}_{j} is the Kronecker delta.

According to Sedov, the basic equation (4) may be used both in Newtonian mechanics and in the theory of relativity. From Eqs.(11) and (12) it is possible to derive, among others, the equations of the classical theory of elasticity and hydrodynamics, as well as the Maxwell equations for electromagnetic field. Moreover, Eqs.(9), (11), and (12) lead to equations of the theory of viscous conducting flow and the theory of irreversible phenomena. Therefore, the Sedov's variational equation can be used as the basic tool for the CMP modelling.

6. RESULTS

1) The lack of classical variational principle (VP) for some physical problem does not preclude the modelling by means of the VP of a broad sense.
2) Solutions of the inverse problem of the calculus of variations, which supply the modified VPs, are not especially valuable for a model builder.
3) In the art of creation of causal mathematical models only the occurrence of the Euler-Lagrange equation is of great interest.
4) The Sedov's variational equation is a convenient tool for the modelling of complex mechanical phenomena (CPM).
5) If we do not know *a priori* the considered phenomenon, only the approach based on postulating may be implemented.
6) Variational principles (VPs) in a broad sense may be used as a unified approach to mathematical modelling of CMP.

7. ACKNOWLEDGEMENTS

This work was supported by grant No. 3 0955 91 01. The author wishes to acknowledge the State Committee for Scientific Research which offered this grant. Author would like to thank Dr. F. Dul for his valuable discussions and many suggestions, and Mr. W. Michalski for his encouragements and criticism.

8. REFERENCES

[1] Truesdell, C., Toupin, R.A., The Classical Field Theories. In: S. Flügge (Ed.), Handbuch der Physik, vol.III/1, Springer-Verlag, 1960, p.595.

[2] Berdichevsky, V. L., Variational Principles of Continuum Mechanics (in Russian). "Nauka", Moscow, 1983.

[3] Finlayson, B.A., The Methods of Weighted Residuals and Variational Principles. Academic Press, New York, 1972, sec. 8.6 and chap. 10.

[4] Arczewski, K., Pietrucha, J., Mathematical Modelling of Complex Mechanical Systems, vol.1. Chichester, E.Horwood, 1993, sec. 4.1.2.

[5] Vainberg, M. M., Variational Methods for the Study of Nonlinear Operators. Holden-Day, San Francisco, 1964, chap. II.

[6] Atherton, R. W., Homsy, G. M., On the Existence and Formulation of Variational Principles for Nonlinear Differential Equations. Studies in Applied Mathematics, LIV (1975), 31-60.

[7] Pontryagin, L. S., et al., Mathematical Theory of Optimal Processes. New York, Interscience Publishers, Inc., 1962.

[8] Oden, J. T., Reddy, J. N., Variational Methods in Theoretical Mechanics. Springer-Verlag, Berlin, 1976, chap. 4.

[9] Sedov, L. I., Mathematical Methods of Building New Models of Continuous Media (in Russian). Progress in Mathematical Sciences, 20 (1965), 121-180.

Nonlinear Free Vibrations of Shallow Shells: A Unifying Formulation

R. HEUER *, H. IRSCHIK ** and F. ZIEGLER *

*Civil Engineering Department, Technical University of Vienna,
Wiedner Hauptstraße 8-10/E201, Austria, A-1040
**Institute of Technical Mechanics and Foundations of Mechanical Engineering,
Johannes-Kepler-University Linz, Linz-Auhof, Austria, A-4040

Abstract. Free large vibrations are studied for the case of shallow symmetrically layered shells. Straight simply supported shell edges are assumed that are prevented from in-plane motions. The influence of geometric nonlinearity is treated using Berger´s approximation. A unifying nondimensional closed form representation for the nonlinear natural vibration periods is given, which is independent of the special planform.

1. INTRODUCTION

Following a procedure developed by the authors for flat and buckled plates [1], [2], this paper gives a unifying presentation of the influence of large amplitudes on the natural vibration periods of shallow shells with arbitrarily shaped, polygonal planform. Shells composed of multiple transversely isotropic elastic layers are considered. Equations of motion according to the dynamic version of the von Karman-Tsien theory [3], modified by Mindlin's kinematic hypothesis [4] are the starting point. The edges are assumed to be simply supported and the in-plane displacements are constrained such that Berger's approximation [5] holds. A multi-mode approach and the Galerkin procedure are subsequently applied to approximately solve the boundary value problem. The result of the projection is a set of nonlinearly coupled ordinary differential equations in time for the generalized coordinates with cubic as well as quadratic nonlinearities. In a single-term approximation, the corresponding solution is in terms of Jacobian elliptic functions which is independent of the special polygonal planform of the shell. For an evaluation of the real-time spectrum of the nonlinear natural fundamental frequency from this unifying similarity solution, only the Dirichlet-Helmholtz-eigenvalue of the corresponding plate must be known.

2. MULTIMODAL APPROACH FOR NONLINEAR NATURAL VIBRATIONS OF SHALLOW SHELLS

Geometrical nonlinearity is considered by means of the kinematic assumptions for the midsurface strains of the shallow shell according to von Karman and Tsien [3]. Taking into account the effect of shear, the distribution of strain through the thickness of the shell is assumed according to Mindlin´s first order shear deformation theory, see [4]. Using these kinematic assumptions when setting up the strain energy the contributions due to the in-plane, bending, and shear deformation become, respectively:

$$U_m = \frac{1}{2} D \int_A I_e^2 \, dA \quad , \tag{1}$$

$$U_b = \frac{1}{2} K \int_A \left\{ \left[\psi^2_{x,x} + \psi^2_{y,y} + \frac{1}{2} (\psi_{x,y} + \psi_{y,x})^2 \right] + \nu \left[2\psi_{x,x} \psi_{y,y} - \frac{1}{2} (\psi_{x,y} + \psi_{y,x})^2 \right] \right\} dA \quad , \qquad (2)$$

$$U_s = \frac{1}{2s} \int_A \left[(w_{,x} + \psi_x)^2 + (w_{,y} + \psi_y)^2 \right] dA \quad , \qquad (3)$$

where w denotes the deflection and ψ_x, ψ_y are the cross-sectional rotations. I_e stands for the first invariant of the midsurface strain tensor. The contribution of the second invariant is neglected in Eq.(1) following Berger [5]: Berger´s assumption is a reasonable well-behaving approximation for structures with immovable in-plane boundary conditions. It renders exact results in the case of a shallow shell strip and has been confirmed for circular plates by Schmidt [6] using a perturbation technique and the von Karman equations. Immovable in-plane boundary conditions are considered throughout the paper. In Eqs. (1) – (3), A denotes the shell area projected onto the (x, y) – plane, D, K, 1/s denote the effective membrane, bending and shear stiffness, respectively, where the case of symmetrically laminated shells composed of transversely isotropic layers is considered:

$$(D, K) = \sum_{k=1}^{N} \int_{z_{k-1}}^{z_k} \frac{E_k}{(1-\nu_k^2)} (1, z^2)\, dz \, , \quad \nu = \frac{1}{K} \sum_{k=1}^{N} K_k \nu_k \, , \quad \frac{1}{s} = \kappa^2 \sum_{k=1}^{N} \int_{z_{k-1}}^{z_k} G_{ck}\, dz \quad . \qquad (4)$$

ν is the effective Poisson´s ratio. The shear factor κ^2 is commonly set to $12 / \pi^2$.

Neglecting rotatory as well as in-plane inertia, the field equations of this shear-deformable Berger-type shallow shell theory have been derived by Heuer [7] using Hamilton´s principle. For thermally stressed and buckled plates see [1], [2]. As a result, the deflection turns out to be governed by the single fourth-order differential equation, μ is the averaged mass per unit of area:

$$K\,(1+sn)\,\Delta\Delta w - n \left[\Delta w - 2(H - Ks\Delta H) \right] - Ks\mu\Delta\ddot{w} + \mu\ddot{w} = 0 \quad , \qquad (5)$$

where n characterizes a time variant isotropic in-plane force, that is constant throughout the plate domain and is related to the deflection by the averaging integral

$$n = -\frac{D}{2A} \int_A w\,(\Delta w - 4H)\, dA \quad , \qquad (6)$$

H is the initial Gaussian curvature of the shallow shell. In a multimodal approach the deflection of the shallow shell is expanded into the orthogonal set of eigenfunctions w_j^* of the corresponding linearized flat plate problem where $n = 0$:

$$w(\mathbf{x},t) = \sum_{j=1}^{N} c_j\, q_j^*(t)\, w_j^*(\mathbf{x}) \quad . \qquad (7)$$

The initial condition of the generalized coordinates are

$$q_j^*(t=0) = 1 \, , \quad \dot{q}_j^*(t=0) = 0 \, . \qquad (8)$$

The coefficients c_j carry the appropriate dimensions and may be chosen freely. The superscript (*) stands for nondimensional quantities. In [8] it has been shown by Irschik, that the eigenfunctions of those shear deformable simply supported plates with polygonal planform are governed by a set of second-order Helmholtz-differential equations with homogeneous Dirichlet's boundary conditions,

$$\Delta w_j^* + \alpha_j w_j^* = 0\ ,\ \ j = 1, 2 ... N \tag{9}$$

$$\Gamma:\ w_j^* = 0\ . \tag{10}$$

For homogeneous plates that are rigid in shear, the eigenvalues are directly proportional to the linear natural frequencies by the common factor $\sqrt{}\rho h/K$, otherwise see Eq. (12) for j = 1. Using Eq. (7) as a Ritz-approximation for the solution of Eq. (5) and running through the Galerkin procedure give the set of coupled non-linear ODEs:

$$q_j^{*''} + a_j^* q_j^* + \delta_{1j} \sum_{k=1}^{N} b_{jk}^* q_k^{*2} + 2\, b_{1j}^* q_1^* q_j^* + q_j^* \sum_{k=1}^{N} e_{jk}^* q_k^{*2} = 0\ \cdot \tag{11}$$

The non-dimensional form of Eq. (11) has been derived by scaling the time with the fundamental natural frequency of the linearized plate problem,

$$t^* = \omega_{1L}^{P}\, t\ , \qquad \omega_{1L}^{P} = \left[K\alpha_1^2 / \mu(1{+}Ks\alpha_1)\right]^{1/2}\quad , \tag{12}$$

where $(\ )' = \frac{\partial}{\partial t^*}$. Furthermore, in the derivation of Eq. (11) it has been assumed that the initial curvature of the shallow shell is proportional to the basic eigenmode of the linearized plate problem,

$$H = \frac{1}{2}\, C\, \alpha_1 w_1^*\ \ , \tag{13}$$

where $C\alpha_1$ measures the intensity of the initial curvature. In that case the coefficients in Eq. (11) turn out to be

$$a_j^* = \alpha_j^{*2} \frac{(1 + s^*)}{(1 + s^*\alpha_j^*)} + C^{*2} D^* \beta_1^* (1 + s^*)\, \delta_{1j}\ \ , \tag{14}$$

$$b_{1j}^* = \frac{1}{2}\, C^* D^* \sqrt{\beta_1^*}\, \alpha_j^* (1 + s^*)\ ,\ \ e_{jk}^* = \frac{1}{2}\, D^* \alpha_j^*\ \alpha_k^* (1 + s^*) \tag{15}$$

with the following similarity numbers:

$$\alpha_j^* = \frac{\alpha_j}{\alpha_1}\ ,\ s^* = Ks\alpha_1\ ,\ D^* = \frac{D\, R_0^2}{K}\ ,\ C^* = \frac{C}{R_0}\ ,\ \beta_1^* = \frac{1}{A}\int_A w_1^{*2} dA\ \ . \tag{16}$$

R_0 denotes a characteristic length. Structures having the same similarity numbers result in an identical nondimensional result. Note that any possibly nonregular polygonal planform of the shell enters only via the parameters α_j^* and β_1^* of the simple linear boundary value problems stated in Eqs. (9) and (10). In order to finally derive the simple form of Eqs. (11), the following choice must be made but without loss of generality:

$$c_j^2 = \frac{R_0^2}{\beta_j^*} \quad . \tag{17}$$

In a single-term approximation,

$$q^{*''} + a^* q^* + 3\, b^* q^{*2} + e^* q^{*3} = 0 \quad , \tag{18}$$

the nonlinear fundamental frequency parameter,

$$\omega_{1N}^* = \omega_{1N} / \omega_{1L}^p \quad , \tag{19}$$

can be determined in terms of the complete elliptic integral of the first kind, compare [9]. ω_{1N}^* turns out to be independent of the special shell geometry. In the single-term approximation, the individual shape of the shallow shell enters the transformation into real-time through the linear natural frequency ω_{1L}^p of the corresponding plate, or, after inverting Eq. (12), through the linear first eigenvalue α_1 of an effectively prestressed membrane of the same planform.

The above similarity solution has been checked by comparison to various particular results from the literature.

References

[1] Heuer, R., Irschik, H. and Ziegler, F., Multi-modal approach for large natural flexural vibrations of thermally stressed plates, *Nonlinear Dynamics* **1**, 1990, 449-458.

[2] Heuer, R., Irschik, H. and Ziegler, F., Multi-modal formulation for free large vibrations of buckled plates, in *Proceedings of the 9th International Modal Analysis Conference*, Florence, Italy, Union College, Schenectady, NY, 1991, 96-100.

[3] von Karman, Th. and Tsien, H.S., The buckling of thin cylindrical shells under axial compression, *Journal of the Aeronautical Sciences* **8**, 1941, 303-312.

[4] Mindlin, R.D., Influence of rotatory inertia and shear on flexural motions of isotropic, elastic plates, *Journal of Applied Mechanics* **18**, 1951, 31-38.

[5] Berger, H.M., A new approach to the analysis of large deflection of plates, *Journal of Applied Mechanics* 22, 1955, 465-472.

[6] Schmidt, R., On Berger´s Method in the Nonlinear Theory of Plates, *J. Appl. Mech.* **41**, 1974, 521-523

[7] Heuer, R., Large Flexural Vibrations of Thermally Stressed Layered Shallow Shells, *Nonlinear Dynamics* **5**, 1994, 1-14.

[8] Irschik, H., Membrane-type Eigenmotions of Mindlin Plates, *Acta Mechanica* **55**, 1985, 1-20.

[9] Weigand, A., Die Berechnung freier nichtlinearer Schwingungen mit Hilfe elliptischer Funktionen, In: *Forschung auf dem Gebiet des Ingenieurwesens* **12**, VDI-Verlag 1941, 274-284.

Functional, Structural and Beavioural models for the design of integrated automation systems.

M.Staroswiecki, J.Ph Cassar, C. Feliot
LAIL URA 1440
Bât. P2-UFR IEEA
Université des Sciences et Technologies de Lille
59655 Villeneuve d'Ascq Cedex

Abstract:

This paper presents a modelization appoach which considers three points of view (functional, structural, behavioural) under which continuous physical processes can be described. First we describe the model, and illustrate our approach on an application example.Then we point out the uses which can be made of such a model in the framework of the design of integrated automation systems.

1. Introduction.

The increasing complexity of production processes, as well as the more and more strict constraints to which they are submitted, do that the automation of such systems can't confine itself to the design of the control process. Consequently, engineers have to design not only digital control systems but merely Real Time Process Operating Systems (RTPOS), covering the operators' needs all along the life cycle of the automated process. This general design approach is called integrated automation [12]. An integrated automation system is based on the relations between three entities :

- a physical process which performs some transformations on the flow of produced goods (the Operative System (OS)).
-an automation system which controls and supervises the operations of the process. It includes hardware (actuators and sensors) and software (the RTPOS) components.
-a team of operators whose tasks are to control, to maintain, to manage the plant and its production.

The set of actuators constitutes the Application System (AS).The RTPOS facilities can be structured as follows :

- a Decision System (DS) which outputs the orders to be applied to the process through the actuators.
- a Communication System (CS) which is in charge of the information exchanges through the system.
-an Information System (IS) which produces the data which will be used by the RTPOS facilities. The IS can be considered as the heart of the RTPOS : it transforms the Raw Data Base (RDB) (the data which are directly accessed through the process or the man/machine interfaces) into a Validated Data Base (VDB) (the data whose pertinence is certified). The fault detection and isolation (FDI) algorithms are the basis of the information system, since they validate the RDB when no fault occurs, and create information which describe the processe's operating ability.

The specification, the design, and the implementation of control algorithms rest on the representation of the system by the means of control models. In the same way the specification, the design, and the implementation of the RTPOS facilities require an adequate modelization of the process.Taking into account the large scale of complex systems, as well as the various types of knowledges which are available upon them, a classical state space description, for instance, is insufficient. In fact large scale systems can be considered as the interconnexion of a given set of basic components. A complete description of the system can be performed through the consideration of the set of these components, answering the three questions :

- what does the system do? (what are the functions of the components)
- how is the system architectured? (what are the links between the components)
- how does the system do ? (what is the components and the overall behaviour)

Such a description can be achieved using functional, structural and behavioural models :

- the functional model describes, independently of the means which are at work, the activities achieved by the system. Each one is supported by a hardware or a software component. Different hierarchical levels can be used for the expression of the functional model. Functional analysis tools exist, which provide both analysis rules and a standard representation of functional models.
- the structural model defines the set of the variables which have been selected to describe the processe's behaviour as well as, at a qualitative level, the constraints between these variables.
Structural properties like observability, controlability, or calculability are accessible. This model can be extended to the whole set of the treatments of the RTPOS.
- the behavioural model expresses the constraint relations which link the variables of the model. These relations can be qualitative (state graph, rules ...) or quantitative (algebraic, differential equations...).

The modelization approach will be illustrated on the following example (figure 1) : two fluids respectively at temperature θ_{e_1} and θ_{e_2} are mixed in a reservoir. Their flows can be controlled so as to maintain the temperature of the delivered fluid as well as the reservoir's level constant.

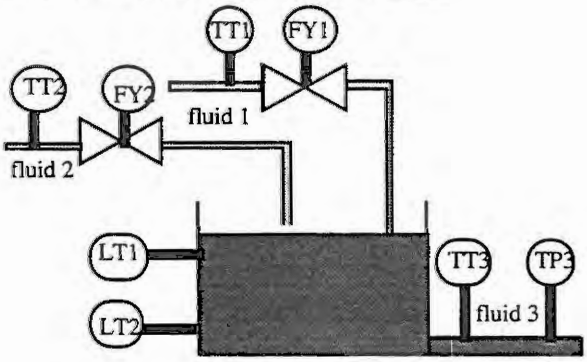


figure 1 : Representation of the example.

2. Functional model.

2.1. Functional analysis methods.

Functional analysis methods provide structured, hierarchical, descendant approaches for the modular analysis, at different abstraction levels of complex systems.

Among the most used methods, we can refer to the following ones :

- the **Flow-model** is a structured method, allowing a qualitative modelization of energy and material flows. The model is based on a graphical representation called flow structure, in which each node represents a function.
Functions are organized into classes (source, transportation, storage, distribution, barrier).
The aim is to provide, according to Lindt[7], a systematic way to identify the objects of the control system as well as a diagnosis assistance.
- the **SADT** method [5] [10] provides both a graphical model and a structured hierarchical approach.
Two points of views are distinguished :
  - the activities of the system are represented by actigrams.
  - the data handled by the activities are represented by datagrams.
- the Structured Analysis methods (SA) are widely used for software design application, **SA-RT** (Structured Analysis and Real Time) allows to introduce temporal considerations.

2.2. SADT approach following product and energy flows.

Some recent works are devoted to functional modelization with standard functions in manufacturing production systems following material's flows [9] [14]. Similarly, our approach identifies a limited number of standard functions. The application to continuous processes calls to a functional representation based on the transformation of energy flows.

SADT covers both the functional and the structural aspects and can be easily connected to classical behavioural modelization tools (transfer functions, state space equations, Bond-Graphs).

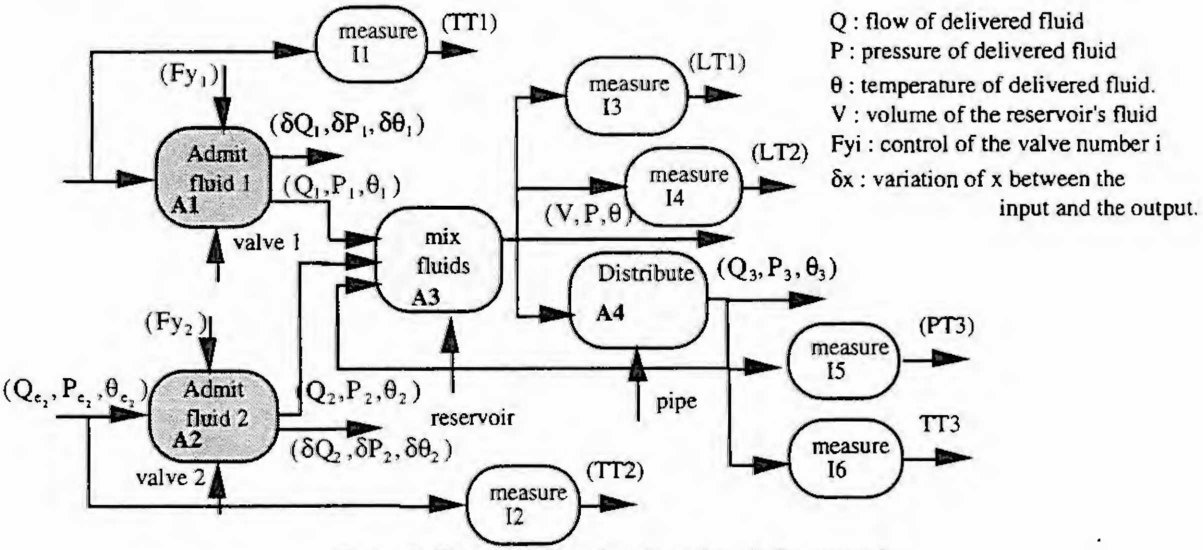
In fact, the energy flows are often connected with material flows. For instance a hot water flow is the support of both an hydraulic and a thermal energy flow. Such an analysis leads to consider, similarly to Flow-Model [7] or Bond-Graph [6] approaches, a limited number of standard functions .

We propose to consider four classes of functions : storage, exchange, power supply, distribution.

In order to represent these phenomenas as well as the coupling between phenomenas of different physical fields [2], we use a vectorial representation, in which the components are power variables (effort/flow) and energy variable (impulsion/displacement) [6] relative to the considered physical fields. Nevertheless this vectorial notation can be extended to other types of variables as the humidity of a gas or a chemical concentration, etc... In this way we are able to represent within the same formalism information flows. The RTPOS facilities can be represented under their functionnal point of view using the same tool. The interfaces between the RTPOS and the physical process are easily represented through the measurement activities (which output Raw data to the information system) and actuation ones (which are controlled by data issued from the decision system). Models of transformation of information into energy and energy into information describe respectively the actuators' and sensors' activities.

2.3. Functional model of the example.

The global function of the system is to provide a delivery of fluid at a given temperature while maintaining constant the level in the reservoir.The functional decomposition of the global function is shown (figure 2).



measure I1
(TT1)
(Fy_1)
$(\delta Q_1,\delta P_1,\delta\theta_1)$
Admit fluid 1 A1
(Q_1,P_1,θ_1)
valve 1
measure I3
(LT1)
measure I4
(LT2)
(V,P,θ)
mix fluids A3
Distribute A4
(Q_3,P_3,θ_3)
(Fy_2)
$(Q_{e_2},P_{e_2},\theta_{e_2})$
Admit fluid 2 A2
(Q_2,P_2,θ_2)
reservoir
pipe
$(\delta Q_2,\delta P_2,\delta\theta_2)$
valve 2
measure I2
(TT2)
measure I5
(PT3)
measure I6
TT3
Q : flow of delivered fluid
P : pressure of delivered fluid
θ : temperature of delivered fluid.
V : volume of the reservoir's fluid
Fyi : control of the valve number i
δx : variation of x between the input and the output.

figure 2: Functional low level model of the example.

The functions A_i are elements of the operative and application system, they represent the activities of the physical system equipped with its actuators. The fonctions I_i are the elements of the information system, they describe the activities of the sensors. The information provided by the sensors and the control signals FTy_1 and FTy_2 (known values of Fy_1 and Fy_2) constitute the Raw Data Base of our system.`

We lead the decomposition so far as to reach the actuator's level. Then all the functions are elements of the classes we've defined earlier, and all the flows are specified, especially the actuators' control signals that we'll have to elaborate.

3. Behavioural model.

A behavioural model expresses a set of relations which specify the constraints between the variables which appear in the functional model.The formal expression of these relations depends on the type of knowledge which is available upon the hardware medium of this activity, and can be analytical, qualitative, rules, numerical tables... To each standard function corresponds a behavioural model. In this way, to each function of the low level functional model, we associate the behavioural model of the component which performs this function.The value of some variables are unknown they're called unknown variables, other ones are measured or known by the system (algorithm's data, control signal...).The behavioural model of a sensor provides a relation between an unknown variable (measured variable) and a known one (measured value), it's called a knowledge relation.

3.1. Behavioural model of the example.

Under the following hypothesis :

- $P_1 = P_2 = P_0$ with P_0 the atmospheric pressure.
- $\delta Q_i = 0$ no leak of fluid accross the valves.
- $\delta\theta_i = 0$ non variation of temperature of fluid across the valves and in the reservoir.

The behavioural model of the system is given by the following set of relations :

function A1 :

$R_{1,1} : Q_{e_1} - Q_1 = 0$

$R_{1,2} : P_{e_2} - P_0 - \delta P_1 = 0$

$R_{1,3} : \theta_{e_1} - \theta_1 = 0$

$R_{1,4} : Q_{e_1} - f(Fy_1) \times \sqrt{\delta P_1} = 0$

function A2 :

$R_{2,1} : Q_{e_2} - Q_2 = 0$

$R_{2,2} : P_{e_2} - P_0 - \delta P_2 = 0$

$R_{2,3} : \theta_{e_2} - \theta_2 = 0$

$R_{2,4} : Q_{e_2} - f(Fy_2) \times \sqrt{\delta P_2} = 0$

Function A3 :

$R_{3,1} : \dot{V} - Q_1 - Q_2 + Q_3 = 0$

$R_{3,2} : c(Q_1\theta_1 + Q_2\theta_2 - Q_3\theta - \dot{V}\theta) = 0$

$R_{3,3} : \dot{V} - \frac{dV}{dt} = 0$

$R_{3,4} : P - f_p(V) = 0$

Function A4 :

$R_{4,1} : Q_3 - f(s) \times \sqrt{P - P_3} = 0$

$R_{4,2} : \theta_3 - \theta = 0$

Function I_i : (knowledge relations)

$Rc_{1,1} : \theta_{e1} - TT_1 = 0$

$Rc_{1,2} : Fy_1 - FTy_1 = 0$

$Rc_{2,1} : \theta_{e2} - TT_2 = 0$

$Rc_{2,2} : Fy_2 - FTy_2 = 0$

$Rc_{3,1} : f_v(V) - LT_1 = 0$

$Rc_{3,2} : f_v(V) - LT_2 = 0$

$Rc_{4,1} : P_3 - TP_3 = 0$

$Rc_{4,2} : \theta_3 - TT_3 = 0$

4. Structural model

The structural model expresses the fact that in the behavioural model a variable appears or not in a relation [11], without taking into account the formal expression of the relation. No restrictions about the type of behavioural models to use is expressed, since the real behaviour can be linear or not, static or dynamic, quantitative or qualitative, while leading to the same structure. The goal of such a representation is to point out the structural properties of the system [9]. The structural model is a bipartite graph which can be represented by its incidence matrix.

4.1. Structural model of the example.

The figure 3 represents the incidence matrix of the system's structure.

	Q_{e_1}	Q_1	P_{e_1}	δP_1	θ_{e_1}	θ_1	Fy_1	P_0	Q_{e_2}	Q_2	P_{e_2}	δP_2	θ_{e_2}	θ_2	Fy_2	$\dot{V}$	Q_3	V	θ	P	P_3	θ_3	TT_1	FTy_1	TT_2	FTy_2	LT_1	LT_2	TP_3	TT_3
$R_{1,1}$	1	1																												
$R_{1,2}$			1	1				1																						
$R_{1,3}$					1	1																								
$R_{1,4}$	1			1			1																							
$R_{2,1}$									1	1																				
$R_{2,2}$								1			1	1																		
$R_{2,3}$													1	1																
$R_{2,4}$									1			1			1															
$R_{3,1}$		1								1						1	1													
$R_{3,2}$		1				1				1				1		1	1		1											
$R_{3,3}$																1		1												
$R_{3,4}$																		1		1										
$R_{4,1}$																	1			1	1									
$R_{4,2}$																			1			1								
$Rc_{1,1}$						1																	1							
$Rc_{1,2}$							1																	1						
$Rc_{2,1}$														1											1					
$Rc_{2,2}$															1											1				
$Rc_{3,1}$																		1									1			
$Rc_{3,2}$																		1										1		
$Rc_{4,1}$																					1								1	
$Rc_{4,2}$																						1								1

figure 3: Incidence matrix of the structure of the exemple.

5. Exploitation of the model.

The three components of the model are connected as shown on figure 4. The structural model constitutes a link between the two other ones. To each low level function corresponds a subsystem of the structural model, which corresponds itself to a set of constraints of the behavioural model.

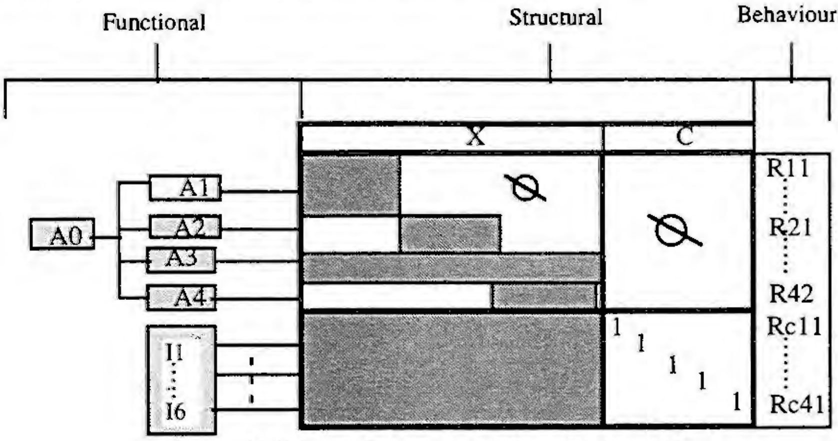


figure 4 : Schematic representation of the model.

5.1. Exploitation of the functional model.

Each standard function is defined by an input and an output interface,whose interconnexions are submitted to constraints. The detection of incompatibilities between the functions' interfaces and the data applied to their input and/or output gives an easy way to evaluate the coherence of the model.

The functional modelization of the operative and application systems expresses all the controls applied to the system. The control flows and the functional tree (issued from the functional decomposition) are helpfull for the specification of the decision system of the RTPOS.

From the functional tree (figure 5a) we infer the functional tree of the decision system (figure 5.b).

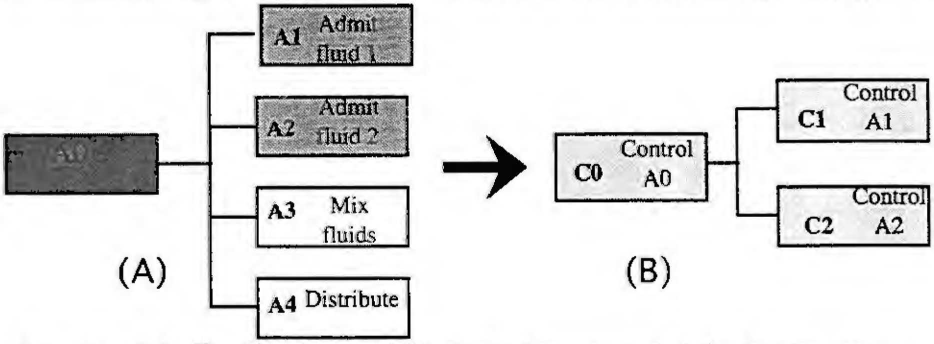


figure 5. a & b : Functional trees of the physical process and of the decision system.

According to the hierarchical organisation of the control (fig 5.b) we get the following functional model (figure 6).

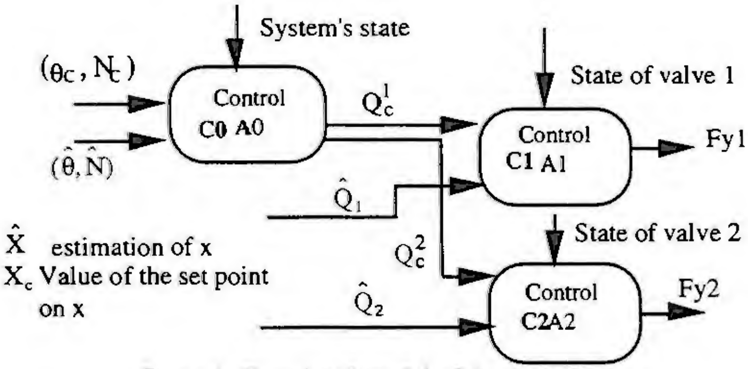


figure 6 : Functional model of the control.

Associated with the definition of the decision functions, we get their information needs that the information system will have to supply.

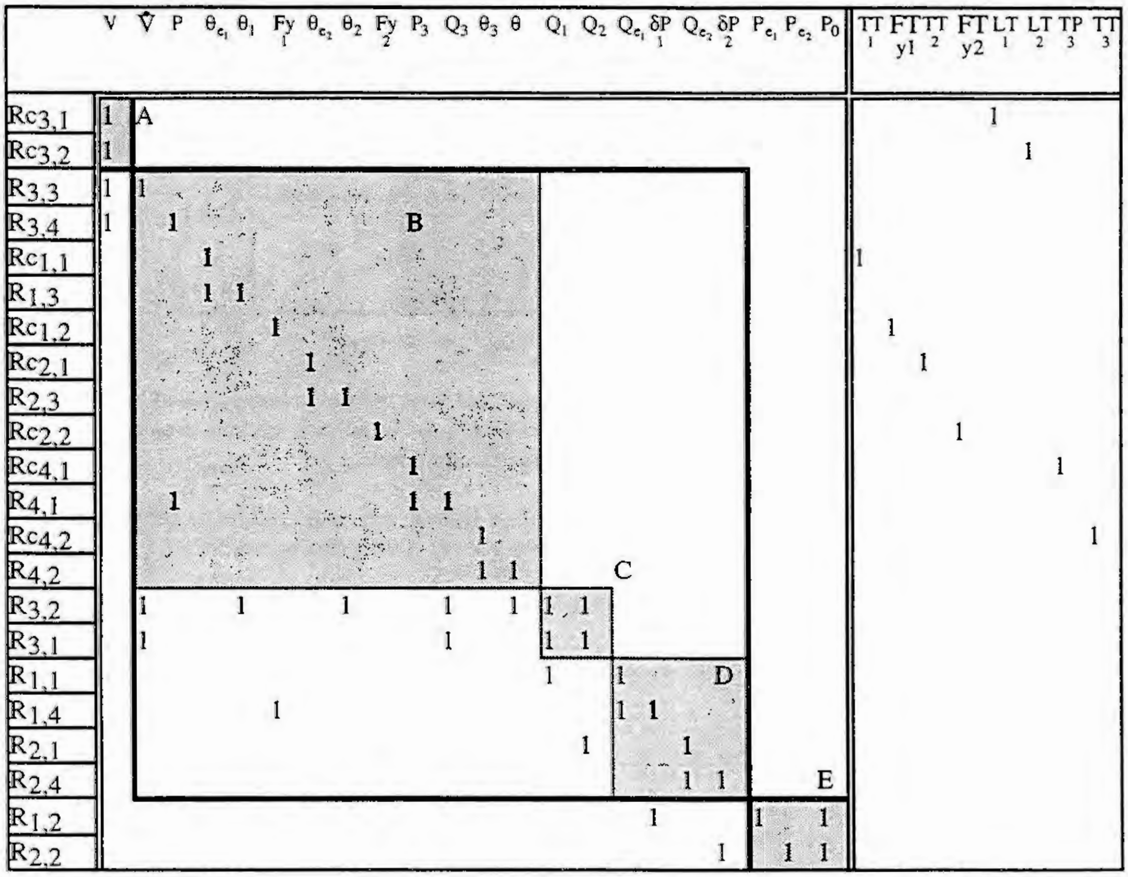
5.2. Exploitation of the structural model.

Canonical decomposition.

The canonical decomposition of a bipartite graph proposed by Dulmage-Mendelson[4], points out three subsystems : over, just and under-determined, this decomposition is unique.

Ph. Declerk[3] demonstrates that this decomposition can be reached from a maximal matching on the calculable structure of the system.

Applying this decomposition to the structural model of the example, we get (figure 7)



	V	$\hat{V}$	P	θ_{c_1}	θ_1	Fy_1	θ_{c_2}	θ_2	Fy_2	P_3	Q_3	θ_3	θ	Q_1	Q_2	Q_{c_1}	δP_1	Q_{c_2}	δP_2	P_{c_1}	P_{c_2}	P_0	TT_1	$FTy1$	TT_2	$FTy2$	LT_1	LT_2	TP_3	TT_3
Rc3,1	1																										1			
Rc3,2	1																											1		
R3,3	1	1																												
R3,4	1		1																											
Rc1,1				1																			1							
R1,3				1	1																									
Rc1,2						1																		1						
Rc2,1							1																		1					
R2,3							1	1																						
Rc2,2									1																	1				
Rc4,1										1																			1	
R4,1			1							1	1																			
Rc4,2												1																		1
R4,2												1	1																	
R3,2		1			1			1			1		1	1	1															
R3,1		1									1			1	1															
R1,1														1		1														
R1,4						1										1	1													
R2,1															1			1												
R2,4																		1	1											
R1,2																	1			1		1								
R2,2																			1		1	1								

figure 7 : Canonical decomposition of the structural model of the example.

Over-determined subsystem.

It can be shown that over-determined subsystems constitute the monitorable part of the overall system. In fact, over-determined subsystems which allow a complete matching on their associated digraph provide Analytical Redundancy Relations (ARR) which are used by the FDI algorithms.

A matching gives a causal orientation to the set of relations which leads to the construction of altemed chains [1]. These ones are helpful guides to generate computation sequences in order to get the numerical value or the analytical form of the ARR. The existence of an altemed chain between two knowledge relations is a neccessary and a sufficient condition for the existence of an ARR[3]. In the example we have a one level over-determination, so one ARR can be generated which is a physical redundancy (two sensors measure the same variable) :

ARR1 : LT1-LT2=0

Just-determined subsystem.

A variable x is said to be calculable if from a constraint F_i its unknown value or analytical form can be computed under the supposition that the values of all the other variables are known. Under the hypothesis of calculability we can find a matching from which we infer altemed chains in order to compute the values of the over and just-determined systems' variables.

All the variables of the subsystem $\{A \cup B \cup C \cup D\}$ are calculable. So we verify that the information system is able to provide an estimation of the variables required by the decision system.

Under-determined subsystem.

A under-determined subsystem can reveal an insufficiency of the behavioural model (not enough equations) or of the equipment (lack of sensors). It gives information about which hypothesis have to be made in order to obtain the knowledge of some variable, or which sensor much be introduced, in order to reduce or eliminate the under-determination.

For instance let us suppose that P_0 (the atmospheric pressure) is known, then the system is just-determined and all the variables are computable. If we suppose that P_{e_1} and P_{e_2} are measured too (pressure sensors have been placed) then the system is over-determined and we gain two other ARRs.

5.3. Exploitation of the coupling.

From the structural model we infer the computation chains of the ARRs and from the behavioural model we can perform the computation.

From the ARRs FDI algorithms are implemented, thanks to the connexion with the functional model we can provide the operators with information about the faults in terms of the system's functions.

6. Conclusion

The three models we propose to consider are complementary and easily connectable. They provide helpful tools for the design of the RTPOS of complex processes, and for their supervision.

We get in this way both a modelization tool and a methodological approach to complex systems.

Different applications on large scale systems [3][14] have shown the interest of such models.

7. REFERENCES

[1] C Berge , Graphes et hypergraphes, Dunod, Paris, 1973

[2] G. Brajnik, L.Chittaro, C.Tasso, E Toppano, "Epistemology, organisation and use of functionnal knowledge for reasoning about physical systems", Tenth International Workshop. Expert Systems and their application. General Conference. Second Generation Expert Systems. Nanterre, France : EC2 1990, p.53-66. Conference : Avignon, France, 28 May- 1 june 1990.

[3] P. Declerck, "Analyse Structurale et Fonctionnelle des Grands Systèmes: Application à une centrale PWR900", thèse de Doctorat, UST Lille, France , 20 decembre 1991.

[4] A.L Dulmage, N.S Mendelshon, "Covering of bi-partite graphs.", *Canadian Journal of Mathematic*, vol 10, 1959, pp 517-534.

[5] M. Galinier, "SADT un langage pour communiquer.", Eyrolle, France, 1989.

[6] D.C Karnopp, D.L Margolis, R.C Rosenberg, "Systems dynamics : a unified approach.", John Willey and son, 1990, 2nd ed.

[7] M.Lind, "Multiflow modelling of process plant for diagnosis and control.", Riso National Laboratory, DK 4000 Roskilde, Denmark, august 1989.

[8] E.Niel, A. Mille, J. Zaytoon, "A temporal SADT extension for modelling and discrete-event simulation of automated manufacturing systems.", *Journal Computer Integred Manufacturing Systems*, 10 mars 1993.

[9] R.J Patton, P.M Franck, R.N Clark, "Fault Diagnosis in Dynamic System, Theory and Application.", Prentice Hall, 1989.

[10] D.T Ross, "Structured Analysis : a langage for communicating ideas.", IEEE transactions on software engineering, January 1987, vol SE 3 n°1 pp 86-95.

[11] M.Staroswiecki, P.Declerck, " Analytical Redundancy in non linear interconnected systems based on a structural approach.", IFAC AIPAC 1989, Nancy, France, pp II 23-27.

[12] M. Staroswiecki, " Les Systèmes et le Concept d'Instrumentation Intelligente.", Ingénieurs et Scientifiques de France, Le Progrès Technique, février 1992, pp 35-42.

[13] J. Zaytoon, "Extention de l'Analyse Fonctionnelle à l'étude de la Sécurité Opérationnelle de Systèmes Automatisés de Production.", Thèse de Doctorat, INSA Lyon, février 1993.

[14] J.Ph Cassar, M. Staroswiecki, E. Herbault, C. Huynh, B. Cordier, " Supervision System Design for a Petroleum Production application.", SAFE PROCESS, Baden-Baden, RFA, september 1991, pp 289-294.

A PARAMETER IDENTIFICATION APPROACH USING OPTIMAL EXCITING TRAJECTORIES FOR A CLASS OF INDUSTRIAL ROBOT

J. SCHAEFERS†, S.J. XU†‡, and M. DAROUACH‡

† CRP-HT, 6, rue Coudenhove-Kalergi, L-1359 Luxembourg-Kirchberg.

‡ LARAL CRAN CNRS UA-821, 186 rue de Lorraine, 54400 Cosnes-et-Romain, France.

Abstract: In this paper, the problem of finding optimal exciting trajectories for parameter identification of industrial robots is investigated. A cost function of maximizing the minimum singular value of a recursive matrix is used in the optimization procedure. The optimal exciting trajectories obtained is insensitive with respect to the parameter perturbation. The identification accuracy and convergence speed of parameters is improved.

1.INTRODUCTION

The parameter identification problem is very important to various model-based controller design of robot. The implementation of identification experience needs a group of well excited trajectories. In order to find exciting trajectories, the approach based on trial is used in early references[1],[4],[5]. However, it is not easy to find a group of acceptable trajectories in a huge trajectory space by trial. Recently, Armstrong[2]-[3] proposed an approach to find the optimal exciting trajectories using an unconstrained non-linear path optimization procedure. A group of 5-th order polynomials is used as exciting trajectories. In practice, the position, velocity and acceleration provided by industrial robot are limited, Armstrong's approach results in excessive requirement on the velocity and acceleration of robot joints. Based on the above fact and considered that a great number of industrial robots work with trajectories of second order polynomials, Schaefers et al [6],[7] proposed an approach for finding constrained optimal exciting trajectories by using the nonlinear parameter optimization theory. A group of second order polynomials is used as exciting trajectories. In the pre-mentioned works[2],[3],[6],[7], the condition number of the recursive matrix is used as the cost function which is minimized in the procedure of optimization.

Unfortunately, the obtained optimal exciting trajectories can't be realised accurately because of the effects of perturbation. In this case, the optimal exciting trajectories are no longer optimal and even result in incorrect identification results. Therefore, to find a group of optimal exciting trajectories which are insensitive with respect to the parameter perturbation is necessary. Unlike the pre-mentioned works [2],[3],[6],[7], in the present paper, a cost function of maximizing the minimum singular value of recursive matrix is used. The identification accuracy and convergence speed of parameters is improved. The optimal exciting trajectories obtained are insensitive with respect to the parameter perturbation. The simulation results of an industrial robot with 2-DOF show the advantages of this approach.

2. THE PARAMETER IDENTIFICATION MODEL AND THE SENSITIVITY ANALYSIS

Consider the following model of robots:

$$M(\theta, p)\ddot{\theta} + D(\theta, \dot{\theta}, p)\dot{\theta}+G(\theta, p) = \tau \tag{1}$$

where θ, $\dot{\theta}$ and $\ddot{\theta}$ are the joint position, velocity and acceleration vector respectively, τ is the control torque (or force) vector, p is the unknown parameters vector, M , D and G are the inertia matrix, the damping matrix about Coriolis and centrifugal force, and the gravity vector, respectively.
After having reparametrized, (1) can be rewritten as

$$\tau=H(\theta, \dot{\theta},\ddot{\theta})p \tag{2}$$

a least square solution p is obtained :

$$p=H^{+}\tau \tag{3}$$

where H is so-called recursive matrix, $H^{+}=(H^{T}H)^{-1}H^{T}$ is the pseudo-inverse of H.
The accuracy of the solution p, generally, is related to the calculation accuracy of inverse of $H^{T}H$. The common method improving the estimation accuracy of parameters is to find a group of exciting trajectories such that $H^{T}H$ is good-conditioned (see[1]-[7]).

Let H and τ become $H+\delta H$ and $\tau+\delta\tau$ due to the parameter perturbations of the exciting trajectories, the corresponding perturbation δp of identified parameters are given by the following lemma:

Lemma1.[8] If $\sigma_{min}(H)>\sigma_{max}(\delta H)$, then

$$\|\delta p\|\leq\frac{\alpha}{1-\Delta}(\|\delta\tau\|+\sigma_{max}(\delta H)\,\|p\|) \qquad (4)$$

or

$$\|\delta p\|\leq\frac{\alpha}{1-\Delta}(\|\delta\tau\|+\Delta\,\|\tau\|) \qquad (5)$$

where $\alpha=\frac{1}{\sigma_{min}(H)}$, $\Delta=\frac{\sigma_{max}(\delta H)}{\sigma_{min}(H)}$

$\sigma_{max}(.)$, $\sigma_{min}(.)$ and $\|.\|$ represent the maximum, the minimum singular value and the 2-norm of (.) respectively.

Lemma2.[8] If $\sigma_{min}(H)>\sigma_{max}(\delta H)$, then

$$\delta cond\leq\beta\frac{\Delta}{1-\Delta} \qquad (6)$$

where $\delta cond=cond(H+\delta H)-cond(H)$, $\beta=1+cond(H)$ (7)

3. THE RELATIONSHIP BETWEEN TWO COST FUNCTIONS

Consider the following two types of cost functions:

$$f_1=cond(H) \qquad (8)$$

and

$$f_2=1/\sigma_{min}(H) \qquad (9)$$

Theorem1. If $\sigma_{min}(H)>\sigma_{max}(\delta H)$, then the variation of cost function f_2 due to the parameter variation of exciting trajectory satisfies the following inequality:

$$\delta f_2\leq\alpha\frac{\Delta}{1-\Delta} \qquad (10)$$

where

$$\delta f_2=\delta\alpha=\left|\frac{1}{\sigma_{min}(H+\delta H)}-\frac{1}{\sigma_{min}(H)}\right| \qquad (11)$$

Proof:

$$\delta f_2=\left|\frac{1}{\sigma_{min}(H+\delta H)}-\frac{1}{\sigma_{min}(H)}\right|\leq\frac{1}{\sigma_{min}(H)-\sigma_{max}(\delta H)}-\frac{1}{\sigma_{min}(H)}=\frac{1}{\sigma_{min}(H)}\left[\frac{1}{1-\Delta}-1\right]=\alpha\frac{\Delta}{1-\Delta}$$

Remarks: Comparing the theorem1 with the lemma2, we see that the two types of cost functions have similar perturbation bound. From the point of view of reducing the sensitivity of cost function with respect to the parameter variation, we desire to minimize Δ, which means maximizing $\sigma_{min}(H)$ if δH is given. This means that the cost function f_2 is an index of sensitivity. Moreover, in order to reduce the identification error, we have to minimize Δ, i.e., maximizing $\sigma_{min}(H)$. So we propose to use f_2 as the cost function for finding optimal exciting trajectories, such that the sensitivity and the identification error are minimized.

Theorem2. Let the partial derivative of cond(H) with respect to the maximum singular value and minimum singular value of H be η and ξ respectively, then

$$\xi=-cond(H)\eta \qquad (12)$$

Proof:

$$\xi=\frac{\partial(cond(H))}{\partial\sigma_{min}(H)}=\frac{-\sigma_{max}(H)}{\sigma_{min}^2(H)}=-\alpha cond(H)$$

$$\eta=\frac{\partial(cond(H))}{\partial\sigma_{max}(H)}=\frac{1}{\sigma_{min}(H)}=\alpha$$

then, (12) is evident.

Remark : (12) shows that, with the decrease of cond(H), $\sigma_{max}(H)$ and $\sigma_{min}(H)$ vary in opposite direction. $\sigma_{max}(H)$ decreases and $\sigma_{min}(H)$ increases. The change rate of cond(H) with the variation of minimum singular value is larger than that with the maximum singular value. In the procedure of minimizing the condition number of H, its maximum singular value has only a poor decrease and its minimum singular value has a great increase if H is poorly conditioned.

4. THE SIMULATION RESULTS

Consider the parameter identification problem of an industrial robot, assume that only the rotation of the fifth and the sixth joints of the robot are considered. The identification model is given by (2), where

$$\tau=[\tau_6\ \tau_5]^T,\ P=[p_1\ p_2\ p_3\ p_4\ p_5]^T=[I'_{6yy}\quad m_6s_6\quad I'_{6xx}-I_{6zz}\quad I_{6zz}+I_{5yy}\quad m_6]^T \qquad (13)$$

p_i is the combination of link parameters of robot and τ_i is the applied torque of i-th joint, and

$$H=\begin{bmatrix} \ddot{\theta}_6 & -gc_5c_6+r_6c_6\ddot{\theta}_5 & \frac{1}{2}s2\theta_6\dot{\theta}_5^2 & 0 & 0 \\ 0 & gs_5s_6-r_6(2s_6\dot{\theta}_5^2+s_6\dot{\theta}_6^2+c_6\ddot{\theta}_6) & -s2\theta_6\dot{\theta}_5\dot{\theta}_6+s_6^2\ddot{\theta}_5 & \ddot{\theta}_5 & gr_6c_5-r_6^2\ddot{\theta}_5 \end{bmatrix} \qquad (14)$$

with

$s_5=\sin(\theta_5)$, $c_5=\cos(\theta_5)$, $s_6=\sin(\theta_6)$, $c_6=\cos(\theta_6)$, $s2\theta_6=\sin(2\theta_6)$

The limits of velocity and acceleration are:

$$\dot{\theta}_5\leq 182°/\text{sec}, \dot{\theta}_6\leq 225°/\text{sec}, \ddot{\theta}_5\leq 445°/\text{sec}^2, \ddot{\theta}_6\leq 562.5°/\text{sec}^2 \qquad (15)$$

Let a second order polynomials be chosen as the exciting trajectories. 200-points are used for identification and the sampling interval is chosen as T=0.01 second, $g=9.81\text{m/sec}^2$, $r_6=1\text{m}$. Using the cost function f_1 and the constraint condition (15), the following optimal exciting trajectories are obtained:

$$\theta_5=1.6712-1.1018t+0.6145t^2 \qquad (16a)$$

$$\theta_6=-0.7677+1.0563t+0.6387t^2 \qquad (16b)$$

Using the cost function f_2 and the constraint condition (15), then we obtain

$$\theta_5=2.5027-3.1764t+1.5882t^2 \qquad (17a)$$

$$\theta_6=0.4334-1.3695t+3216t^2 \qquad (17b)$$

The fig.1 and fig.2 show the two groups of optimal exciting trajectories which are found by using same initial condition. The fig.3a-3e show the convergence procedure of parameters using the two group of exciting trajectories. Table1 gives the comparison of the results of sensitivity and identification error.

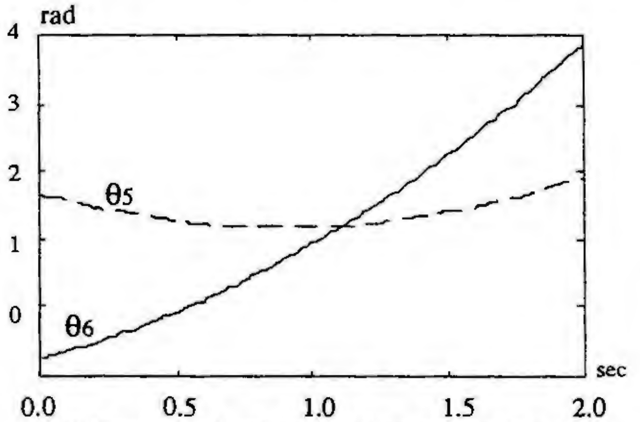


Fig.1 Constrained optimal exciting trajectories with f_1

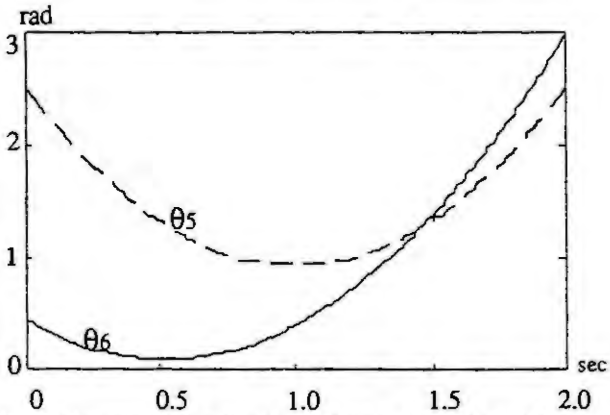


Fig.2 Constrained optimal exciting trajectories with f_2

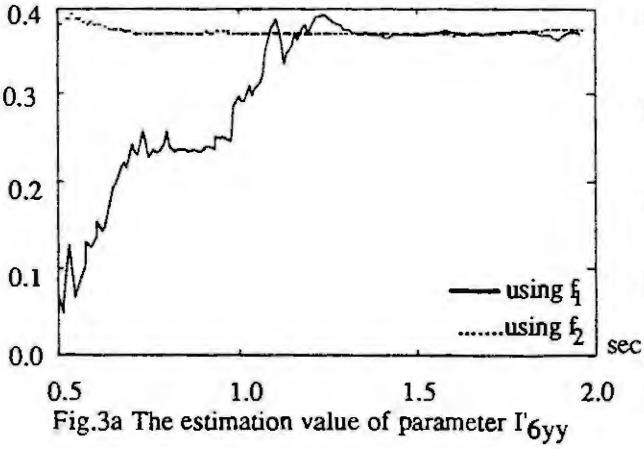


Fig.3a The estimation value of parameter I'_{6yy}

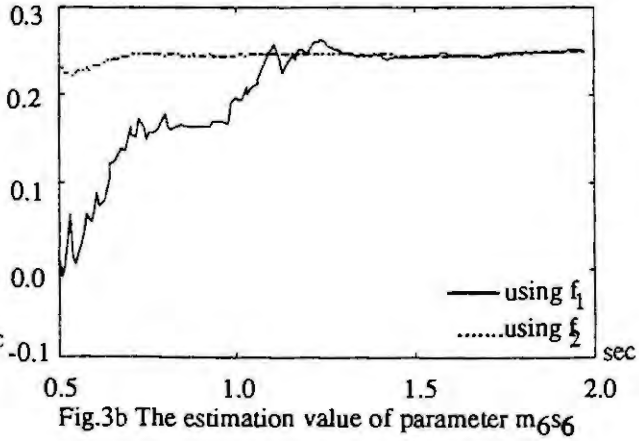


Fig.3b The estimation value of parameter m_6s_6

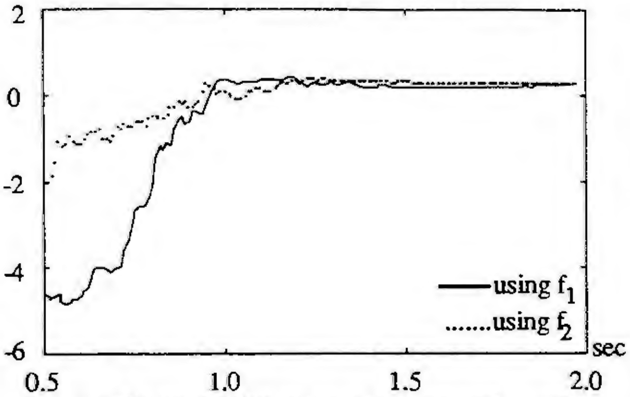


Fig.3c The estimation value of parameter $I'_{6xx}-I_{zz}$

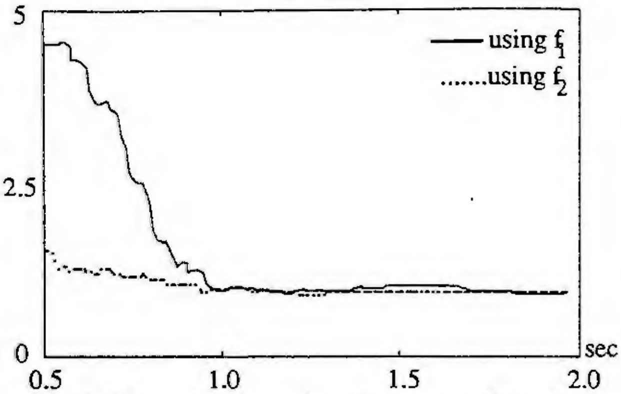


Fig.3d The estimation value of parameter $I_{6zz}+I_{5yy}$

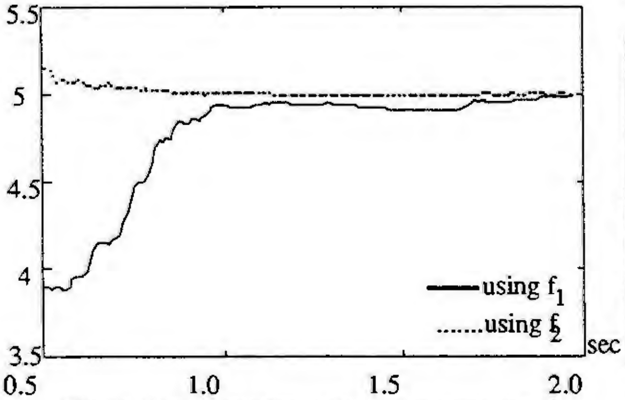


Fig.3e The estimation value of parameter m6

Table1.The comparison of calculation results

	f_1	f_2		f_1	f_2
cond(H)	2.6484	3.1197	δcond(H)	≤1.8494	≤1.1182
$\sigma_{max}(\delta H)$	5.5287	7.2965	$\delta\alpha$	≤0.0308	≤0.008
$\sigma_{min}(H)$	16.4355	34.1813	$\lVert \delta p \rVert_H$	≤2.5879	≤1.3856
Δ	0.3364	0.2135	$\lVert \delta p \rVert_\tau$	≤0.0061	≤0.0029

Remarks: The simulation results show that to use the trajectories (16) has the advantages of the high convergence velocity , the small parameter error , the insensitivity of trajectories with respect to the parameter perturbation, even if trajectories(16) has larger condition number than that of trajectories(17).

5. CONCLUSIONS

We have discussed the problem of finding the optimal exciting trajectories for parameter identification of industrial robots. We propose to use the maximizing minimum singular value of recursive matrix as the cost function. The optimal exciting trajectories obtained are insensitive with respect to its parameter perturbation and the estimation errors of parameters are minimized. The present approach can be applied to parameter identification of industrial robots.

6. REFERENCES

[1] An, C.H., C.G. Atkeson, and J.M. Hollerbach, Model-Based Control of a Robot Manipulator, MIT Press, 1988.

[2] Armstrong, B., "On finding exciting trajectories for identification experiments involving systems with non-linear dynamics," Int. J. Robot. Research, vol.8, no.6, pp. 28-48, 1989.

[3] Armstrong, B.,"on finding 'exciting' trajectories for identification experiments involving systems with non-linear dynamics," Proc. IEEE Conf. Robot and Autom., pp. 1131-1139, 1987.

[4] Atkeson, C.G., C.H. AN, and J.M. HOLLERBACH, "Estimation of inertial parameters for manipulators," In Proc. of 24th IEEE Conf. Dec. and Contr., Fort Lauderdale, FL, pp.990-995, 1985.

[5] Khosla, P., and T. Kanade, "Parameter identification of robot dynamics," In Proc. of 24th IEEE Conf. Dec. and Contr., Fort Lauderdale, FL, pp.1754-1760, 1985.

[6] Schaefers, J., Contribution à l'identification des parametres des robots:application à un robot KUKA, Thèse de Doctorat, Université de Nancy 1, France(1992).

[7] Schaefers, J., S.J. Xu, and M. Darouach, "A Parameter Identification Approach of Using Thé Decentralized Optimal Exciting Trajectories for A class of Industrial Robot," In Proc. IMACS/IFAC second international symposium on mathematical and intellident models in system simulation, vol.II, pp.65-68, 1993, Beldium.

[8] Schaefers, J., S.J. Xu, and M. Darouach, "Sensitivity and Error Analysis of Parameter Idantification for A class of Industrial Robots," In Proc. IEEE/SMC International Conference, 1993, France.

RECURSIVE IDENTIFICATION OF INTERCONNECTED SYSTEMS APPLICATION TO THE MODELISATION OF AN ANNEALING LINE

Mustapha OULADSINE *, Claude IUNG ** et José RAGOT *
* Centre de Recherche en Automatique de Nancy - CNRS UA 821
BP 40 - Rue du doyen Marcel Roubault - F 54 500 Vandoeuvre les Nancy
Tél. : (33) 83 50 30 80   Fax : (33) 83 50 30 96
** ENSEM - 2 Avenue de la Forêt de Haye - 54516 Vandoeuvre les Nancy

1. ABSTRACT

In the field of identification of multi-input, multi-output (MIMO) systems, the methods can be divided into two principal groups, according to the model structure : the state-space formulation and the input-output description in terms of transfer functions.

A MIMO system, described by the second representation can be decomposed into ny interconnected subsystems, where ny is the number of outputs. These subsystems can be decomposed into single-input, single-output subsubsystems which are the transfer functions of the model. In this paper, the transfer function output is called "partial output" ; the transfer function input may be either a general input of the system or a partial output of another transfer function or even a white noise. This representation has two main advantages for identification : a minimal number of parameters has to be identified the parameters have physical system meanings. A disadvantage is the unmeasurable partial output, which must be estimated. A procedure is suggested, which uses estimates of the partial output in a recursive way. This method identifies the input-output and noise-dynamics of the multivariable system contaminated by coloured noise, using a simple stage estimator.

2 - STATEMENT OF THE PROBLEM

We consider the discrete-time multi-input, multi-output system described by the following model :

$$y_j(k) = \sum_{i=1}^{i=nu} \frac{B_{ij}(q)}{F_{ij}(q)}\, u_i(k-k_i\Delta) + \sum_{i=1}^{i=nu} \sum_{\substack{l=1 \\ l\neq j}}^{l=ny} \frac{G_{ilj}(q)}{H_{ilj}(q)}\, s_{il}(k) + \frac{C_j(q)}{D_j(q)}\, \varepsilon_j(k) \qquad j=1 \text{ to } ny \tag{1}$$

where Δ is the sampling period, q is the shift operator : $q\, f(k) = f(k-\Delta)$, k_i is the i^{th} transfer function delay, nu is the number of inputs $u_i(k)$, and ny the number of outputs $y_j(k)$, $\varepsilon_j(k)$ is a white stochastic sequence, $s_{il}(k)$ is the unmeasured partial output of the transfer function $B_{il}(q)/F_{il}(q)$, $B_{ij}(q)$, $F_{ij}(q)$, $G_{ilj}(q)$, $H_{ilj}(q)$, $C_j(q)$ and $D_j(q)$ are constant polynomials, for this model, the interconnection relationships between subsystem j and subsystem l are represented by the transfer function $G_{ilj}(q)/H_{ilj}(q)$ (where $l\neq j$). This model is more general than those used by some authors [1], [2] [3] because it takes into account the interactions between all the subsystems.

Consider a general two-input, two-output system with noisy output. The interconnected model of this system, according to equations (1) is represented in figure 1.

Then, according to the notations presented in figure 1 the equation (1) becomes :

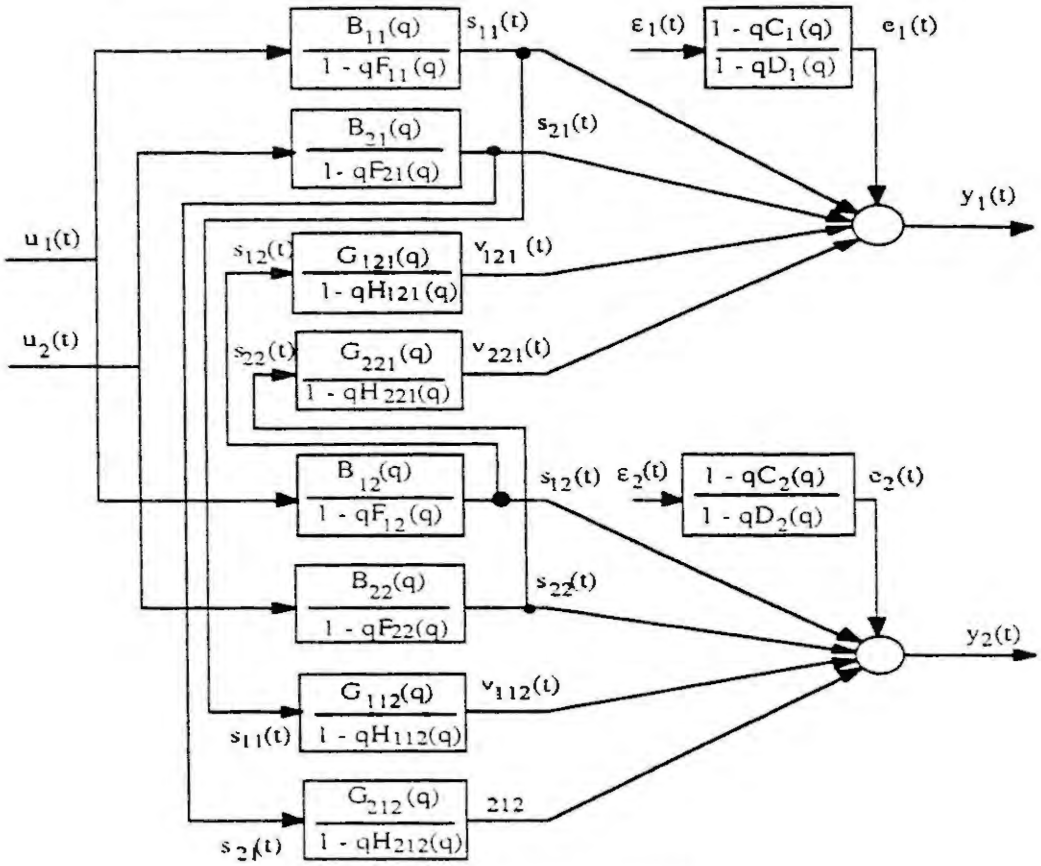
$$y_j(k) = \sum_{i=1}^{i=nu} s_{ij}(k) + \sum_{i=1}^{i=nu} \sum_{\substack{l=1 \\ l\neq j}}^{l=ny} v_{ilj}(k) + e_j(k) \qquad j=1 \text{ to } ny \tag{2}$$

where :

$$s_{ij}(k) = \frac{B_{ij}(q)}{1-qF_{ij}(q)}\, u_i(k-k_i\Delta) \tag{3a}$$

$$v_{ilj}(k) = \frac{G_{ilj}(q)}{1-qH_{ilj}(q)}\, s_{il}(k) \tag{3b}$$

$$e_j(k) = \frac{1+qC_j(q)}{1-qD_j(q)}\,\varepsilon_j(k) \tag{3c}$$



$\frac{B_{11}(q)}{1-qF_{11}(q)}$
$s_{11}(t)$
$\varepsilon_1(t)$
$\frac{1-qC_1(q)}{1-qD_1(q)}$
$e_1(t)$
$\frac{B_{21}(q)}{1-qF_{21}(q)}$
$s_{21}(t)$
$y_1(t)$
$u_1(t)$
$s_{12}(t)$
$\frac{G_{121}(q)}{1-qH_{121}(q)}$
$v_{121}(t)$
$u_2(t)$
$s_{22}(t)$
$\frac{G_{221}(q)}{1-qH_{221}(q)}$
$v_{221}(t)$
$\frac{B_{12}(q)}{1-qF_{12}(q)}$
$s_{12}(t)$
$\varepsilon_2(t)$
$\frac{1-qC_2(q)}{1-qD_2(q)}$
$e_2(t)$
$\frac{B_{22}(q)}{1-qF_{22}(q)}$
$s_{22}(t)$
$y_2(t)$
$\frac{G_{112}(q)}{1-qH_{112}(q)}$
$v_{112}(t)$
$s_{11}(t)$
$\frac{G_{212}(q)}{1-qH_{212}(q)}$
212
$s_{21}(t)$

Figure 1 : interconnected system

Assuming that $f(k\Delta)$ is noted $f(k)$, system (2) becomes at time $t = (k+1)\Delta$:

$$y_j(k+1) = \sum_{i=1}^{i=nu} s_{ij}(k+1) + \sum_{i=1}^{i=nu}\sum_{\substack{l=1\\ l\neq j}}^{l=ny} v_{ilj}(k+1) + e_j(k+1) \qquad j=1 \text{ to } ny \tag{4}$$

where :

$$s_{ij}(k+1) = F_{ij}(q)\, s_{ij}(k) + B_{ij}(q)\, u_i(k+1-k_i) \tag{5a}$$
$$v_{ilj}(k+1) = H_{ilj}(q)\, v_{ilj}(k) + G_{ilj}(q)\, s_{il}(k+1) \tag{5b}$$
$$e_j(k+1) = D_j(q)\, e_j(k) + C_j(q)\, \varepsilon_j(k) + \varepsilon_j(k+1) \tag{5c}$$

With the decomposition (4), the MIMO model is decomposed into ny MISO submodels [5]. The output of each submodel depends on unmeasured partial outputs : but these outputs can be estimated in a recursive manner.

3. IDENTIFICATION

These recursive forms (5) of the signals $s_{ij}(k+1)$, $v_{ilj}(k+1)$ and $e_j(k+1)$ are replaced in the equations of the system (4) :

$$y_j(k+1) = \sum_{i=1}^{i=nu} [F_{ij}(q)\, s_{ij}(k) + B_{ij}(q)\, u_i(k+1-k_i)] + \sum_{i=1}^{i=nu}\sum_{l=1,l\neq j}^{l=ny} [H_{ilj}(q)\, v_{ilj}(k) + G_{ilj}(q)\, s_{il}(k+1)]$$
$$+ D_j(q)\, e_j(k) + C_j(q)\, \varepsilon_j(k) + \varepsilon_j(k+1) \tag{6}$$

Estimates of the polynomials $F_{ij}(q)$, $B_{ij}(q)$, $H_{ilj}(q)$, $G_{ilj}(q)$, $D_j(q)$ and $C_j(q)$ of equation (6) are given by the recursive least squares algorithm [4], [7] et [8]. The quantities $y_j(k)$ and $u_j(k)$ are obtained from an experiment and the unknown signals $s_{ij}(k)$, $v_{ilj}(k)$, $e_j(k)$ and $\varepsilon_j(k)$ are replaced by their estimates :

$$\hat{s}_{ij}(k) = \hat{F}_{ij}(q)\,\hat{s}_{ij}(k-1) + \hat{B}_{ij}(q)\,u_i(k-k_i) \tag{7a}$$

$$\hat{v}_{ilj}(k) = \hat{H}_{ilj}(q)\,\hat{v}_{ilj}(k-1) + \hat{G}_{ilj}(q)\,\hat{s}_{il}(k) \tag{7b}$$

$$\hat{e}_j(k) = y_j(k) - \sum_{i=1}^{i=nu} \hat{s}_{ij}(k) - \sum_{i=1}^{i=nu} \sum_{\substack{l=1\\ l\neq j}}^{l=ny} \hat{v}_{ilj}(k) \tag{7c}$$

$$\hat{\varepsilon}_j(k) = \hat{e}_j(k) - \hat{D}_j(q)\,\hat{e}_j(k-1) - \hat{C}_j(q)\,\hat{\varepsilon}_j(k-1) \tag{7d}$$

Using matrix representation, system (5) becomes :

$$y_j(k+1) = \underline{X}_j(k+1)\,\underline{\Theta}_j + \varepsilon_j(k+1) \qquad j=1 \text{ to } ny \tag{8}$$

where $\underline{\Theta}_j$ contains all the parameter to be identified in subsystem j ; form (8) allows the estimation of the parameters with the help of a standard recursive algorithm.

4. EXAMPLE TO ILLUSTRATE THE PROPOSED REPRESENTATION

The proposed method is applied to the modelisation of a moving strip in a n-roller system in a continuous annealing line. Some authors have dealt with the web tension modelling problem [6].

The furnace composes the main device of the line. It gives the web a thermal cycle ; each stage of the cycle corresponds to a metal transformation. The process consists of nine heating and cooling zones. It is separated from the entry section and the exit section by two looping towers. Because of these two looping towers, the operations performed on the strip in the entry section and the exit section can occur without disturbing the constant flow of material in the center (figure 2).

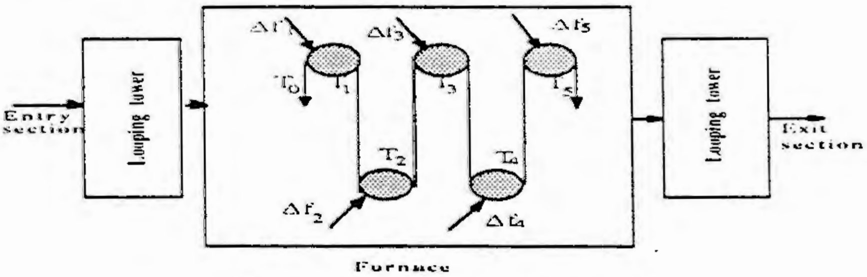


Figure 2 : continuous annealing line

In order to identify the model of the tension behaviour in the furnace, a decomposition of the model into smaller subsystems is chosen. This approach presents experimental and numerical advantages : only the inputs of the studied subsystem have to be stimulated and the number of parameters to be estimated is not too important.
In our case, a natural decomposition is a decomposition in which a subsystem corresponds to an output. In this paper, we only consider the subsystem of output T_2 (figure 3).

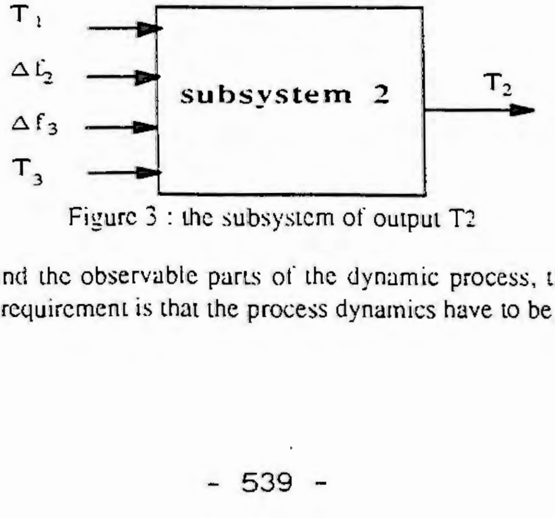


Figure 3 : the subsystem of output T2

To identify the controllable and the observable parts of the dynamic process, the input signals must satisfy certain conditions. A minimal requirement is that the process dynamics have to be persistently stimulated by the input signals over the measured period. This means that the input signals must be sufficiently rich in information to stimulate all the modes during the experiments. Satisfying these conditions, a Pseudo Random Binary Signal (PRBS) is therefore used. A PRBS is a series of positive and negative steps around the mean value. The sampling time of the successive steps is chosen according to the spectra band which corresponds to real dynamics of the system.

The wanted model is the one presenting a parametric discrete time form (for instance ARX model). Application of such a model to output T_2 yields the following equation :

$$T_2(t) = \frac{B_1(q-1)}{A_1(q^{-1})} T_1(t) + \frac{B_2(q^{-1})}{A_2(q^{-1})} T_3(t) + \frac{B_3(q^{-1})}{A_3(q^{-1})} \Delta f_1 + \frac{B_4(q^{-1})}{A_4(q^{-1})} \Delta f_2$$

To assess the validity of the method, the response of the model obtained by the new method is simulated. Figure 4 gives a comparison between simulated outputs T_2 and the measured outputs. We observe that the simulated output and the measured output are very close.

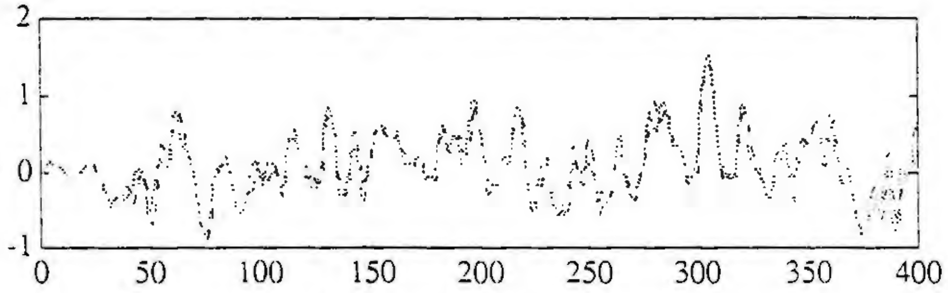


Figure 4 : the simulated output and the measured output

5. CONCLUSION

In this paper, we have used a special approach to identify a real multivariable system ; this system is decomposed into smaller subsystems. Then, the smaller system models are estimated and then connected together to build the complete system model. The simulation of the system using the model obtained by interconnection of the subsystems is in good agreement with the experiments shows the validity of the approach.

6. REFERENCES

[1] Dieckmann, K. and H. Unbehauen (1979). Recursive identification of multi-input, multi-output systems. *Proceedings of the 5th IFAC Symposium*, I, p. 423-428.

[2] Fkirin, M.A. (1989). Choice of models for on-line identification of MIMO stochastic systems. Int. J. Systems Sci., 20 (4), p. 609-618.

[3] Gauthier, A. and I.D. Landau (1978). On the recursive identification of multi-input, multi-output systems. *Automatica*, 14, p. 609-614.

[4] Gopinath, B. (1969). On the identification of linear time invariant systems from input-output data. *Bell System Techn.*. J., 48, p. 1101-1113

[5] Ouladsine, M (1993). Identification des systèmes dynamiques multi-variables. Thèse de l'université de Nancy I.

[6] Parant, F., Iung C., Bello P., (1989). Traction and speed Control of an iron strip in a continuous annealing line. Proceeding of the 3rd E.P.E conference, 9-11Octobre 1989, pp 1417-1419.

[7] Prata, A. and G.P. Rao (1986). A comparative study of certain recursive parameters estimation algorithms for linear discrete time dynamical systems. *IMACS-IFAC Symposium on modelling and simulation for control of lumped and distributed parameters systems*, 1986, p. 379-381.

[8] Talmon, J.L. and A.J.W. Van der Boom (1973). On the estimation of the transfer function parameters of process and noise dynamics using a single stage estimator. *3th Symposium IIFAC identification and system parameters estimation*. 2, p. 711-720

MODELLING AND IDENTIFICATION OF AN AIR CONDITIONING SYSTEM.

*L.FRIOT *, C.PETRAULT **, J.C.TRIGEASSOU **.*

* - GEC/ALSTHOM : Etablissement d'Aytré, La ROCHELLE.
** - Laboratoire d'Automatique et d'Informatique Industrielle
40, avenue du Recteur PINEAU, 86022 POITIERS Cedex.

****

ABSTRACT :

This paper presents the experimental results of the modelling and identification of an air conditioning system.
The process has been decomposed in two subsystems connected in series with a natural feedback. It has been identified by the Model Method using Marquardt's Algorithm.

KEYWORDS :

Heat Process Modelling, Parameter Estimation, Model Method.

INTRODUCTION :

This paper presents the results of modelling and identification of an air conditioning process.
The heating system is composed of two parts. The first one is used to heat, with an electrical radiator, a mixing of fresh and recycled air. The second one deals with the room heated by blowed air. The control input COM is the voltage applied to the radiator ; the recycled air temperature (TAR) is a secondary input. The outputs of this process are the temperatures of each part, i.e. the blowed air temperature (TAS) and TAR for the second one.

The structure of this system leads to a model composed of two transfer functions, connected in series, H1 (with inputs COM and TAR and output TAS) and H2 (with input TAS and output TAR). Notice that this air conditioning process is a natural feed-back system, due to the reinjection of recycled air.

The models H1 and H2 are differential equations, i.e we have chosen a continuous time representation. This unusual choice was motivated by the necessity of physical interpretation of systems parameters, which is impossible with difference equations. Nevertheless, this model presents a serious counterpart, i.e. its non-linearity in the parameters.

Consequently, they are estimated by an iterative procedure based on the minimization of a quadratic criterion by Non Linear Programming. The method used is the Marquardt's algorithm because of its robustness towards initialization. In order to implement it, it is necessary to compute the Gradient and the Hessian with a sensitivity model.

In the first part, we present the modelling of the air conditioning system ; the identification algorithm and its implementation compose the second part. Finally, experimental results exhibit the quality of this modelling.

I: MODELLING OF THE AIR CONDITIONING SYSTEM

Introduction

Previously, we have presented the structure of the process which is schematized by figure 1.

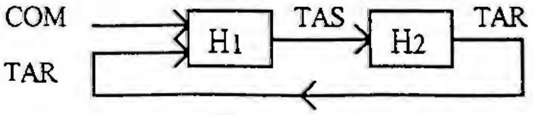


Figure 1.

Then, we are going to describe each subsystem.

H1 Model :

The physical output of H1 is the heated air flow injected in H2. Pratically, the only available measurement is the temperature of this air flow, TAS, which becomes H1 output. The air is heated by an electrical radiator controlled by COM input ; a secondary heat supply is provided by the recycled air flow.

A first order system gives a good approximation of the heating dynamics.

Thus we get the model of figure 2 :

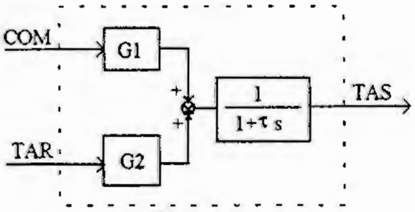


Figure 2.

H2 Model :

If we consider the step response of the output (TAR), we see clearly that it is composed of two dynamics with the time constants τ_1 and τ_2.

The analysis of this thermal process confirms this situation and leads to a second order system composed of two 1^{rst} order systems working in parallel (see figure 3) .

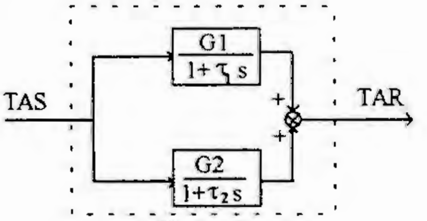


Figure 3.

So we get the following 2^{nd} order transfer function :

$$H_2(s) = \frac{G(1+Zs)}{(1+\tau_1 s)*(1+\tau_2 s)}$$

II : THE IDENTIFICATION METHOD

Introduction :

The choice of continuous time representation allows physical interpretation of system parameters, particularly for τ_1 and τ_2, which is an essential advantage for a collaboration with heat engineers.

Nevertheless, it involves a more difficult estimation problem needing Non Linear Programming. This identification technique is usually known as the Model Method (see figure 4).

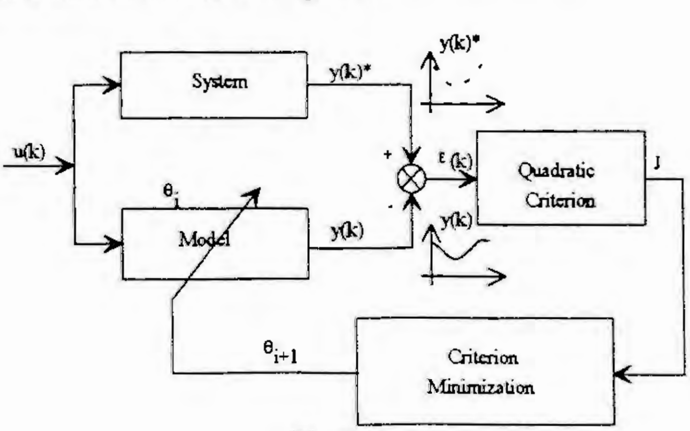


Figure 4 .

Marquardt's Method :

The quadratic criterion is given by the relation :

$$J = \sum_{k=0}^{K} (y_k^* - \hat{y}_k)^2$$

The goal is to minimize J while changing the model parameters $\underline{\theta}_i$ at each iteration.

$$\underline{\theta}_{i+1} = \underline{\theta}_i + \underline{\Delta\theta}$$

Where $\underline{\Delta\theta}$ is the parameters variation.

The technique used to minimize J is the Marquardt's Algorithm. It presents the advantage to be a combination of Gradient and Gauss-Newton Methods.
It realizes a good compromise between the Gradient Method, which is slow but numerically stable, and the Gauss-Newton one, which converges very quickly, but only in the vicinity of the optimum (it is unstable with a rough initialization).

So, we use the following algorithm :

$$\boxed{\left[\lambda * I + J''_{\theta\theta}\right] * \underline{\Delta\theta} = -J'_\theta}$$

* If λ tends to infinity, then :

$$\underline{\theta}_{i+1} \approx \underline{\theta}_i - \mu * J'_\theta \qquad \text{Where } \mu = \frac{1}{\lambda}$$

Which is the Gradient Method.

* If λ tends to zero, then :

$$\underline{\theta}_{i+1} \approx \underline{\theta}_i - \left[J''_{\theta\theta}\right]^{-1} * J'_\theta$$

Which is the Gauss-Newton Method.

This technique permits a continuous transition between the two methods by an adaptive modification of the λ coefficient during the iterative procedure.

Implementation :

This algorithm needs the calculation of the Gradient and Hessian of J. They are obtained by the sensitivity model ($\sigma_{\theta_{(k,n)}}$) of the output.
This model, and the system one, are computed by the Runge Kutta 4 method.

So we get the Gradient :

$$\frac{\partial J}{\partial \theta_i} = -2 * \sum_{k=1}^{K} (\varepsilon_k * \sigma_{\theta_{(k,i)}})$$

and the Hessian with the Gauss-Newton approximation :

$$\frac{\partial^2 J}{\partial \theta_i \partial \theta_j} \approx 2 * \sum_{k=1}^{k} \sigma_{\theta_{(k,i)}} * \sigma_{\theta_{(k,j)}}$$

III : EXPERMENTAL RESULTS

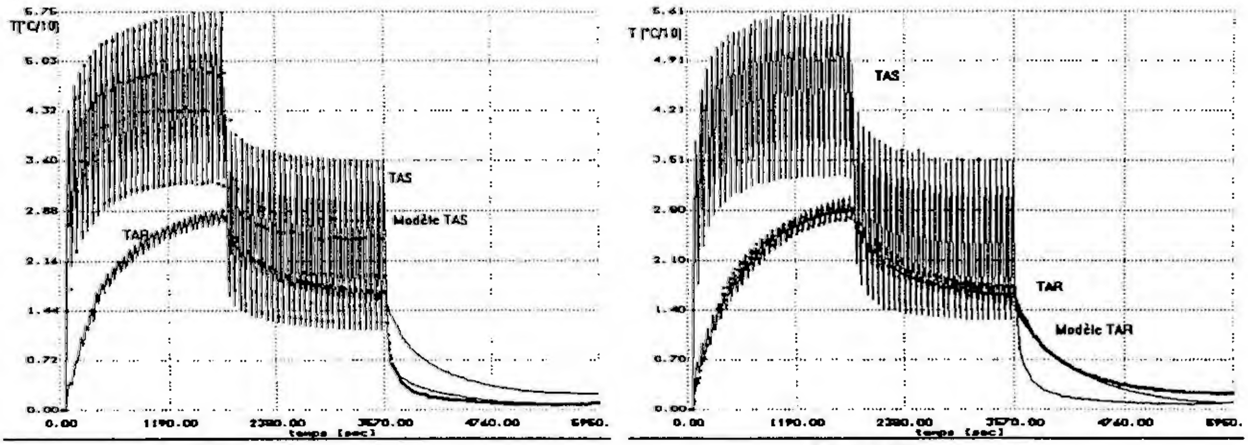
Implementation :

In fact, the heating process is based on a Pulsed Width Modulation technique, which must be took into account in the identification procedure (see the blowed air temperature oscillations).

The input COM is the excitation, it is composed of different steps allowing a large exploration of the operating domain.

Results :

In the proposed experiment, we have considered two heat steps followed by a decreasing phase. The sampling period was equal to 10 s, (figure 5).
Refer to the following table for H1 and H2 parameters.



Model H1

Model H2

Figure 5.

Model H1	*Model H2*
G1 = 0.481	**G = 0.66**
G2 = 0.401	**Z = 170**
τ = 10.98	**τ1 = 6.5**
	τ2 = 632

CONCLUSION

We have presented the modelling of a laboratory air conditioning system and the identification of its subsystems by the Model Method.

The estimated model has been validated by numerous experiments ; it has also been used succesfully to design a control loop (Cascade Internal Model Control).

It is important to notice the ability of this Model to give good results with large input variations.

ACKNOWLEDGEMENT

The support of this research has been given by GEC-ALSTHOM (Aytré - La ROCHELLE).

REFERENCES

[MAR 63] Donald W.MARQUARDT.
" An algorithm for last squares estimation of non linear parameters ".
J - SOC Indust. Appl. Math. , Vol 11 / USA.
N°2 , June 1963 Page 431 - 441.

[RIC 71] J.RICHALET, A.RAULT, R.POULIQUEN.
"Identification des processus par la Méthode du Modèle ".
Gordon & Breach.
Vol 4 , 1971.

[TRI 88] J.C.TRIGEASSOU.
"Recherche de modèles expérimentaux ".
Lavoisier TEC/DOC 1988.

Modelling a Motor Vehicle and its Braking System.

G.L. GISSINGER Y. CHAMAILLARD T. STEMMELEN
Laboratoire M.I.A.M. Modélisation et Identification en Automatique et en Mécanique.
Université de Haute Alsace
I.R.P. B.P. 2438, 34 rue Marc Seguin F 68067 Mulhouse Cedex

ABSTRACT

This article presents the modelling of a motor vehicle and its braking system (fig 1). The system to be modelled consists of the vehicle with all the suspension elements - wheel, tyre and wheel-road interface. The model must account for the behaviour of the whole system according to the longitudinal and vertical axes when the wheel is decelerated. The mass of the quarter vehicle on each wheel has to be adaptative during the braking phase in order to model the mass transfer. So, we have developed a 15th order knowledge model from the functioning equations for CAD and a 2nd order representation model for control which was obtained through identification. The knowledge model has been introduced using our "Prouesse" Bond Graph Tool. We are now developing the complete model of a semi-trailer.

1. INTRODUCTION

The advent of microprocessors and microcontrollers, their decreasing cost and concurrently increasing performance allow the application of a number of theories to the control of industrial processes. For many applications however, industries still prefer the PID regulator, even if it tends to be digital rather than analogic. Today, we may distinguish two types of controls : "conventional" controls and "modern" controls.

In industry, real systems are rarely available if the optimization of the regulation is to be tested. Therefore, even before conceiving a control system, a control-engineer will necessarily have to go through a modelling phase of the system. This phase offers, among others, the double advantage of improving the knowledge of the system and subsequently of allowing the validation of the developed control. Searching for a model requires a number of separate stages : a characterization stage which allows us to obtain the best adapted shape or structure of the model ; a quantitative stage which consists in quantifying the parameters of the previous model, generally followed by a reduction phase in the case of complex systems ; and finally a validation phase which compares the real operation and the simulation of the system and which will allows us to evaluate the modelling quality. This work, usually done iteratively in order to increase the performance of the models, requires great care and experience.

2. MODELLING - IDENTIFICATION

The models can be of different types according to how they were obtained - a knowledge model developed from physical laws that rule the system, or a representation model developed through identification from the overall behaviour between input and output. Each approach has its characteristics, advantages and drawbacks. It is obvious that, both methods do not lead to the same results during the first iteration, but it is no less desirable that both solutions converge on the same results, at least locally and microscopically, after a number of iterations and adjustments. The general processes in designing these models are shown in figure 2.

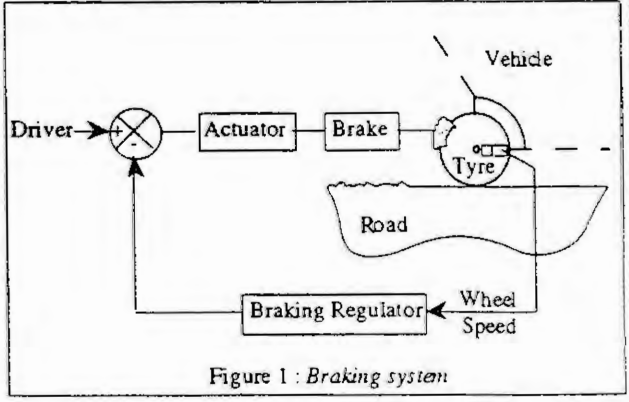


Figure 1 : *Braking system*

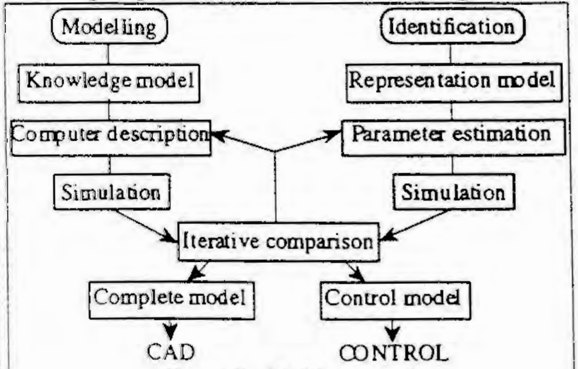


Figure 2 : *Model approach*

2.1 The knowledge model

This is a theoretical model obtained from the physical laws that rule the system; it is usually represented by differential equations, but it may also use the state representation, differences equations, or it may be represented by more general tools such as bond graphs which implicitly include the previous information. The knowledge model must be the most complete possible - general, precise and full of information ; it must integrate all the non-linearities of the system, in static as well as in dynamic conditions. Identity of internal behaviour is then achieved. For very complex systems, the model may reach a high degree of difficulty and refer to very different fields in physics, in the case of modelling an internal combustion engine, for example.

If such a model offers the advantage of representing the real system on a wide operation range, the fact remains that the necessary calculation powers, times and costs increase rapidly. As an example for our investigations, we shall give the mechanical model of a road vehicle. Known as "quarter vehicle model", it accounts for the vertical and horizontal behaviour of the bodywork and load suspended.

This model (fig. 3) allows us to take account of -among others- the road surface, the vertical behaviour of the tyre, the suspension (spring + shock absorber), the transfer of the static and dynamic load between the rear and front of the vehicle, Broulhiet's effect in the suspension, and to introduce the numerous non-linearities linked to the spring and shock absorber.

The model is developed from the differential equations that rule the system according to the following figure. It is to be noted that the vehicle model also refers to another knowledge model, namely the mechanical model of the longitudinal behaviour (fig. 4). Bond graphs are chosen because it is a unified and graphic method, and the representation of dynamic systems is simplified (fig. 5-6). Bond graphs rely on the fact that physical phenomena implied in the description of complex and industrial systems can be expressed in terms of four generalized variables, i.e., effort, flow, displacement and momentum [ROSE 86]. On our Bond Graph models, there are two special components not directly concerned with energy description - the first one (bottom of figure 5 - middle) is a module, represented as a square with a special glyph inside. Such a component represents the ability to introduce hierarchical bond graphs - the second one (figure 5 - bottom left hand corner) represents a connector for an external input or output, and is represente by a ring.

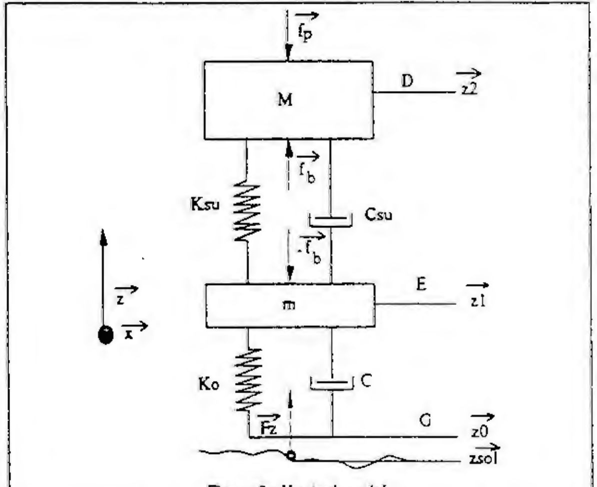


Figure 3 : *Vertical model*

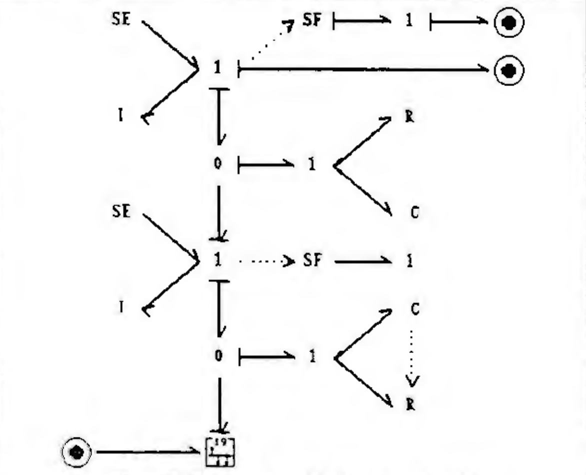


Figure 5 : *Bond graph model of the quarter vehicle*

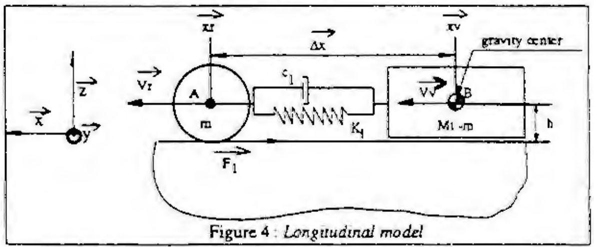


Figure 4 : *Longitudinal model*

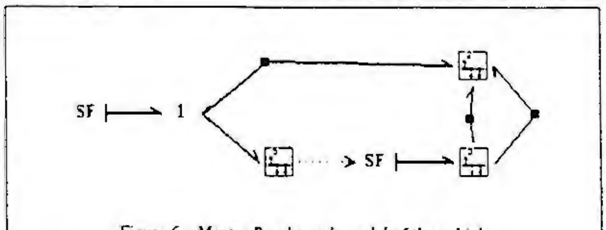


Figure 6 : *Master Bond graph model of the vehicle*

These different models must be quadrupled and a model of the linking elements must be added to account for the global behaviour of the vehicle.

2.2 The representation model

This model -also called "black box model"- accounts for the system in operation as seen from the outside, from its input and output variables. In this case, the model parameters have no physical significance and we obtain identity of external behaviour. The model is developed from experimental input/output data using methods of graphic, mathematical, statistic or other analyses. The advantages of the representation model lie in the simplicity of its structure and implementation and in its capacity to work in real time. However, it can only describe a limited working area and consequently it is not easy to use for simulation purposes in complex processes and/or in dynamics variable in time.

The braking circuit of a vehicle consists of an actuator (servo-valve) which is essentially a function of the hydraulic load (constant) and of the circuit configuration ; so, one good representation of the actuator is sufficient. For the actuator model, we shall use a transfer function identified from experimental measures obtained on Renault's test vehicle. The transfer (servo-valve) between the required pressure and the pressure in the circuit is shown in figure 7.

$$SV(p)=\frac{Pr(p)}{Pd(p)}=\frac{0.9858+0.01318\ p}{1+0.01748\ p+0.000166\ p^2}\ e^{-0.003\ p}$$

Figure 7 : *Actuator model.*

$$B(p)=\frac{Pr(p)}{Pd(p)}=21.42*\frac{1}{1+0.0019\ p}\ e^{-0.012\ p}$$

Figure 8 : *Brake model.*

The processes were the same for the brake model (pressure-torque transfer, shown in figure 8), but in this case, the variations of gain due to hysteresis phenomena and temperature variations appear as an input disturbance. It must be noted that the efforts transmitted by the tyre are given by Pacejka's mathematical model which is a representation model [PACE 87].

2.3 The "Grey Box Model"

In brief, the knowledge model is precise but complex, whereas the representation model is less precise but simple which shows that a compromise is necessary. Considering the specifications of the regulation, economic constraints, material and human means for the project, etc, the art of the control engineer will be to reconcile at best both models into one often called "Grey box" whose simplicity, cost and precision are "reasonable", i. e. a model that meets the aforesaid requirements of a manufacturer. The complete vehicle-model is typically of the grey box type, as it associates both knowledge and representation models.

In general, when any system is to be represented by a mathematical model and a representation model has been chosen, a "measuring campaign" is first carried out on site, from the output correlated with the input of the system. Subsequently, the system will be identified in laboratory using one of the numerous identification algorithms (Least square, Generalized least squares...). The quality of the identification clearly depends on that of the measurements as well as that of the excitation [WEBE 92] ; therefore, the measuring campaign must be prepared with a great deal of precision, so as not to occupy the system too often, too long, or in too repetitive a way. In the adaptive case, parameters become adjustable and then must be estimated at each sampling period. Determining these parameters requires a recursive adaptation algorithm that can be implemented in real time.

2.4 Real time identification algorithms

Such algorithms are numerous in literature, but the algorithms based on the recursive minimization of a "least squares" type criterion are the ones that can achieve the best performance and greater flexibility. There are different variants of these algorithms commonly called MCR; we chose five of them that are quite representative: the conventional recursive least squares, Kulhary's [KULH 84}, Fortescue's [FORT 81], Salgado's [SALG 88] and Bertin's [BERT 85] algorithms. After investigation, each of the algorithms presents the following advantages and drawbacks :

* Recursive least squares algorithm : it is simple to implement, but it cannot be used to estimate parameters whose dynamics is too important in time and it may sometimes be dangerous (no control of the adaptation matrix).

* Kulhary's algorithm : it is relatively simple to implement and introduces a forgetting factor regrettably a constant one -which, by reducing the operation window, accelerates and appreciably improves the performance compared with the previous one ; however, it accepts only slight variations of the system dynamics and is not suitable for our applications. Denomination : Directional Forgetting Method (DF).

* Fortescue's algorithm : its idea is to adjust the forgetting factor on the basis of the variations of the prediction error. Indeed, we admit that the estimator will be more sensitive to the parameter variations if we reduce (or increase) the working window when the prediction error increases (or decreases). This principle may be right in theory, but practically it is only verified in a deterministic environment. Denomination : Prediction-Error Forgetting Technique (PEF).

* Salgado's algorithm : This algorithm is a very good solution for problems of blow-up, important dynamics and noisy signals ; however, it requires many initial constants (whose adjustment soon becomes adaunting task) as well as great precision in calculation. Therefore, we could not use it for our application. Denomination : Exponential Forgetting and Resetting Algorithm (EFRA).

* Bertin's algorithm : The results of Kulhary's algorithm are markedly poorer than those of Fortescue's when high dynamics are followed through, but in many cases it avoids the blow-up phenomenon of the co-variance matrix. After observing this fact, Bertin's algorithm associates both concepts in one estimator. Denomination : PEDF.

For our application, we chose Bertin, Bittanti and Bolzern's algorithm (1985) which is the following :
Let the model :
$y(t) = \varphi(t)'\theta + e(t).$
where φ is the observation vector, θ the parameter vector and e(t) a white noise, the estimation of the parameters θ is then descibed by the following equations :

$\varepsilon(t) = y(t) - \varphi(t)'\hat{\theta}(t-1)$ where $\varepsilon(t)$ is the prediction error,

$K(t) = P(t-1)\varphi(t).[1+\varphi(t)'P(t-1)\varphi(t)]^{-1}$ where K(t) is the gain matrix and P(t) is the covariance matrix,

$\hat{\theta}(t) = \hat{\theta}(t-1)+K(t)\varepsilon(t),$

$P(t) = [I-H(t)\varphi(t)'].P(t-1),$

in which : $H(t) = P(t-1)\varphi(t).[\beta(t)^{-1}+\varphi(t)'P(t-1)\varphi(t)]^{-1},$

where $\beta(t)$ can be : - $\beta(t) = \mu(t)-(1-\mu(t)).[\varphi(t)'P(t-1)\varphi(t)]^{-1}$, if $\varphi(t)\neq 0$ in which $\mu(t)$ is the forgetting factor,
or : - $\beta(t) = 1$, if $\varphi(t)=0$,

$\mu(t) = \max\{\mu o\ ;\ \alpha(t)\}$

and : $\alpha(t) = 1-\left\{[1-K(t)'\varphi(t)].[\varepsilon(t)^2+\Gamma]^{-1}\varepsilon(t)^2\right\}$

This algorithm - which is simple to implement and adjust - has been used for various applications in our lanoratory (identification of a travelling crane, identification of the angular dynamics of a vehicle, etc) and has always offered a very good compromise between the calculation time and the precision of the results.

3. CONCLUSION

In this paper, we have described the principle of modelling a motor vehicle and its braking system. This set represents a system which is pseudo-stable (due to the wheel/road interface) and fast (it requires a sampling frequency in the region of a millisecond). Our study allowed us to obtain a complete, validated knowledge model, a representation model in progress thanks to measuring and identification campaigns on a Renault R-19 test-vehicle, and a grey box model which uses tje information of the former two and which allowed us to develop the algorithms for the braking regulation of a better performing RBS system than the existing ABS systems.

ACKNOWLEDGEMENTS

A good part of this research work was possible thanks to many, fruitful contacts and discussions with the study and Research Departments of Thomson CSF DOI, Renault and PSA-Peugeot-Citroën. We wish to thank every person who helped us to complete this work.

REFERENCES

[BERT 86] D. BERTIN, S. BITTANTI, P. BOLZERN. Politecnico di Milano (Italy)
Tracking of nonstationary systems by means of different prediction-error - directional forgetting techniques.

[FORT 81] T.R. FORTESCUE, L.S. KERSHENBAUM and B.E. YDSTIE.
Automatica, Vol 17, No 6, p 831 - 835, 1981.
Implementation of self-tuning regulators with variable forgetting factors.

[KULH 84] R. KULHAVY and M. KARNY. Proc 9th IFAC world congress, p 79 - 83,
Budapest, 1984. Tracking of slowly varying parameters by directional forgetting.

[PACE 85] H.B. PACEJKA. Delft University of Technology, Vehicle Research Laboratory.
Journal of the Franklin Institute, Vol. 319, No. 1/2, pp. 67-81, January/February 1985.
Modelling Complex Vehicle Systems Using Bond Graphs.

[PACE 87] H.B. PACEJKA, E. BAKKER and L. NYBORG.
SAE Paper No. 870421, 1987. Tyre modelling for use in vehicle dynamics studies.

[ROSE 86] R. ROSENBERG and D. KARNOPP. "Introduction to physical dynamics systems"
Mc Graw Hill 1986.

[SALG 88] M.E. SALGADO, G.C. GOODWIN and R.H. MIDDLETON.
Int. J. Control, vol 47, No 2, p 477 - 491, 1988.
Modified least square algorithm incorporating exponential resetting and forgetting.

[WEBE 92] Ph. WEBER. Thesis of Université de Haute Alsace, Mulhouse 1992.
Modélisation et identification d'un système de freinage pour véhicule automobile et conception de la commande.

A STUDY ON MODEL SENSITIVITY FOR SINGLE-LINK FLEXIBLE ARM CONTROL

Bruno SICILIANO

Dipartimento di Informatica e Sistemistica
Università degli Studi di Napoli Federico II
Via Claudio 21, 80125 Napoli, Italy
E-mail: siciliano@na.infn.it

Abstract. Design of high-performance control systems for a robot manipulator having flexible links heavily relies on an accurate dynamic model of the system. Discretization of an inherently distributed-parameter system into a finite-dimensional model plays a relevant role both for control design and simulation of the system. This work is aimed at studying the sensitivity of inverse dynamics control laws for a single-link flexible arm to the number of modes included into the analysis. Simulation results are presented.

1. INTRODUCTION

The potential for successful utilization of robot manipulators having lightweight flexible links reposes trust in the effectiveness of controllers which are capable to reduce the vibrations naturally induced along the motion of such systems. Currently used flexible manipulators move at very low speeds to limit the excitement of vibration; for instance, the structure of the telemanipulator used on the Space Shuttle has very low resonant frequencies (0.04÷0.35 Hz) and operates at an average speed of 0.5 deg/s [1].

Mechanical flexibility becomes important for relatively fast speed motions. The design of enhanced controllers must consider the effects of flexibility and then it relies upon the availability of an accurate dynamic model of the system. Due to the distributed nature of flexibility, the approach used to discretize the system as well as the order of approximation plays a relevant role for control design. Nonetheless, also for testing the performance of the controlled system it is important to model the system as accurately as possible in order to simulate a situation as close as to physical reality.

This work reports a study on sensitivity of control laws for a single-link flexible arm to the number of modes included in the model of the system. This is obtained through the usual Lagrangian approach combined with the assumed modes method for modelling distributed flexibility [2,3]. Inverse dynamics control laws [4] are considered with both types of collocated and non-collocated outputs [5]. The numerical results obtained in simulation tests are presented and discussed.

2. DYNAMIC MODEL

A single-link flexible arm can be modelled as an Euler-Bernoulli beam. Under the usual assumptions that only planar bending occurs and deflections are small, the dynamic model can be derived using the Lagrangian approach combined with the assumed modes method [2,3] leading to

$$\begin{pmatrix} J & \mu^{\mathrm{T}} \\ \mu & \rho\ell I \end{pmatrix} \begin{pmatrix} \ddot{\theta} \\ \ddot{\delta} \end{pmatrix} + \begin{pmatrix} 0 & 0^{\mathrm{T}} \\ 0 & D \end{pmatrix} \begin{pmatrix} \dot{\theta} \\ \dot{\delta} \end{pmatrix} + \begin{pmatrix} 0 & 0^{\mathrm{T}} \\ 0 & K \end{pmatrix} \begin{pmatrix} \theta \\ \delta \end{pmatrix} + \begin{pmatrix} g_\theta(\theta,\delta) \\ g_\delta(\theta) \end{pmatrix} = \begin{pmatrix} u \\ 0 \end{pmatrix} \tag{1}$$

where θ is the joint angle, $\delta = (\, \delta_1 \;\; \ldots \;\; \delta_N \,)$ is the vector of modal deflection variables, J is the total inertia at the joint, μ is a vector describing the inertial coupling between rigid body and flexible body motion and is a function of the mode shapes, ρ is the arm linear mass density, ℓ is the arm length, I is the identity matrix of proper dimensions, D and K are respectively the link damping and stiffness diagonal matrices, g_θ and g_δ are the gravity torques, and u is the joint driving torque; it has been assumed that the arm is clamped at the joint location and then the joint torque enters directly only in the rigid-body equation [6]. As can be easily seen the only nonlinearity in the model (1) is due to gravity, and in particular the gravity torque in the flexible dynamics equations is only a function of the joint angle [7].

3. INVERSE DYNAMICS CONTROL

The equations of model (1) can be rewritten as

$$J\ddot{\theta} + \mu^{\mathrm{T}}\ddot{\delta} + g_\theta(\theta,\delta) = u \tag{2}$$

$$\mu\ddot{\theta} + \rho\ell\ddot{\delta} + D\dot{\delta} + K\delta + g_\delta(\theta) = 0. \tag{3}$$

The flexible accelerations can be extracted from (3) as

$$\ddot{\delta} = -\frac{1}{\rho\ell}\left(\mu\ddot{\theta} + D\dot{\delta} + K\delta + g_\delta(\theta)\right) \tag{4}$$

and substituted in (2), yielding

$$\left(J - \frac{\mu^{\mathrm{T}}\mu}{\rho\ell}\right)\ddot{\theta} - \frac{1}{\rho\ell}\mu^{\mathrm{T}}\left(D\dot{\delta} + K\delta\right) + g_\theta(\theta,\delta) - \frac{\mu^{\mathrm{T}}g_\delta(\theta)}{\rho\ell} = u. \tag{5}$$

An inverse dynamics control law can be designed as

$$u = \left(J - \frac{\mu^{\mathrm{T}}\mu}{\rho\ell}\right)a - \frac{1}{\rho\ell}\mu^{\mathrm{T}}\left(D\dot{\delta} + K\delta\right) + g_\theta(\theta,\delta) - \frac{\mu^{\mathrm{T}}g_\delta(\theta)}{\rho\ell} \tag{6}$$

where a denotes a new input acceleration. Under control (6), the equations of the system become

$$\ddot{\theta} = a \tag{7}$$

$$\ddot{\delta} = -\frac{1}{\rho\ell}\left(\mu a + D\dot{\delta} + K\delta + g_\delta(\theta)\right) \tag{8}$$

where (7) is the linear system of a double integrator and (8) describes the internal zero dynamics left into the system. If $\theta_d(t)$ denotes a desired smooth joint trajectory, the well-known resolved acceleration linear control can be adopted

$$a = \ddot{\theta}_d + k_D(\dot{\theta}_d - \dot{\theta}) + k_P(\theta_d - \theta) \tag{9}$$

where k_D,k_P are suitable positive gains that shape the response of the system and guarantee that the joint trajectory is exactly reproduced. Regarding the internal dynamics, a simple Lyapunov argument can be used to show that the so-called zero dynamics ($\dot{\theta} \equiv 0$) is globally asymptotically stable [4] which is a sufficient condition for the overall system to be stable.

Notice that control (6) requires full state feedback. There is no problem to measure the joint position and velocity as well as to reconstruct the flexible variables from strain gauge measurements. In order to overcome the drawback of lack of flexible rates measurements, it is possible to implement the control law by resorting to a feedforward strategy. For the given joint trajectory $\theta_d(t)$, the dynamic equations (4) can be forward integrated over time with suitable initial conditions (usually null) to provide the time history of $\delta_d(t)$,$\dot{\delta}_d(t)$. As a consequence, control (6),(9) can be modified into

$$u = u_d + k'_D(\dot{\theta}_d - \dot{\theta}) + k'_P(\theta_d - \theta) \tag{10}$$

where

$$u_d = \left(J - \frac{\mu^{\mathrm{T}}\mu}{\rho\ell}\right)\ddot{\theta}_d - \frac{1}{\rho\ell}\mu^{\mathrm{T}}\left(D\dot{\delta}_d + K\delta_d\right) + g_\theta(\theta_d,\delta_d) - \frac{\mu^{\mathrm{T}}g_\delta(\theta_d)}{\rho\ell} \tag{11}$$

and

$$k'_P = \left(J - \frac{\mu^{\mathrm{T}}\mu}{\rho\ell}\right)k_P \qquad k'_D = \left(J - \frac{\mu^{\mathrm{T}}\mu}{\rho\ell}\right)k_D. \tag{12}$$

It can be shown that robustness of the system to imperfect model compensation as in (11) is satisfactory in most practical cases [4].

The above inverse dynamics control law (6) globally linearizes the system with respect to a collocated output taken at the joint level. A more challenging task is to track a non-collocated output taken at the arm level. In detail consider a point at location x along the arm; the angle $\alpha = \theta + \mathrm{arctg}(y(x)/x)$ can be considered as a parametrized output, where $y(x) = \phi^{\mathrm{T}}(x)\delta$ expresses the deflection as a function of the mode shapes at that point and the flexible variables. Proceeding as above, it is not difficult to show that the inverse dynamics control —in the case of zero gravity without loss of generality—

$$u = \left(J - \left(\mu^{\mathrm{T}} - \frac{J}{x}\phi^{\mathrm{T}}(x)\right)\left(\rho\ell I + \frac{1}{x}\mu\phi^{\mathrm{T}}(x)\right)^{-1}\mu\right)a - \left(\mu^{\mathrm{T}} - \frac{J}{x}\phi^{\mathrm{T}}(x)\right)\left(\rho\ell I + \frac{1}{x}\mu\phi^{\mathrm{T}}(x)\right)^{-1}\left(D\dot{\delta} + K\delta\right) \tag{13}$$

transforms the system into

$$\ddot{y} = a \tag{14}$$

$$\ddot{\delta} = -\left(\rho\ell I + \frac{1}{x}\mu\phi^{\mathrm{T}}(x)\right)^{-1}\left(\mu a + D\dot{\delta} + K\delta\right), \tag{15}$$

where, similarly to (9), a can be chosen as

$$a = \ddot{y}_d + k_D(\dot{y}_d - \dot{y}) + k_P(y_d - y). \tag{16}$$

The stability of the zero dynamics that can be obtained from (15) by setting $y \equiv 0$ depends on the sign of the function [5]

$$\Gamma(x) = 1 - \frac{\phi^{\mathrm{T}}(x)\mu}{\rho\ell x}. \tag{17}$$

When $x = 0$ (joint output) it is $\Gamma(0) = 1$ and the system is always stable. At an x where $\Gamma(x)$ becomes negative the system goes unstable (equivalent phenomenon to nonminimum-phase linear systems).

4. MODEL SENSITIVITY

The actual arm considered in this work has the following data: length of 0.5 m, linear mass density of 0.2 kg/m, flexural rigidity of 1 Nm^2, joint+actuator inertia of 0.1083 kg/m^2, tip payload mass of 0.1 kg and inertia of 0.0005 kgm^2; the arm inertia is negligible with respect to the joint+actuator inertia, so that constrained mode shapes can be used to model deflections [8]. With these data the first four natural frequencies are 2.1784, 15.9145, 40.1008, 92.6093 Hz which have been used in the simulated model of the flexible arm with a sampling time of 1 ms. The elements of the damping matrix D_i have been chosen as $0.1K_i$, $i = 1, 4$.

In the first case study, a joint trajectory of 90 deg has been assigned and three types of control laws have been simulated: • joint PD control + gravity compensation [$PD+$] with $k_P = 12$,$k_D = 1$, • full state inverse dynamics control [ID] with $k_P = 100$,$k_D = 20$, • inverse dynamics control with feedforward of flexible states [FID] and same gains. Notice that the first kind of control does not ensure exact tracking but at least guarantees globally asymptotic regulation of the final joint position [7]. The following table summarizes the results obtained with the above control laws and a variable number of modes in terms of the maximum joint and tip tracking errors.

control	# modes	max error [deg]	max error [mm]
PD+	0	3.0793	5.8811
ID	1	0.090461	5.7483
ID	2	0.089988	5.7452
ID	3	0.089953	5.7440
FID	1	0.092417	5.7484
FID	2	0.092057	5.7452
FID	3	0.092054	5.7440

In the second case study, a trajectory of 90 deg has been assigned to the angular output associated to an arm point. The full state inverse dynamics control has been simulated with the same gains as above. The following two tables summarize the results obtained with a variable distance from the joint location and a variable number of modes: in the left table the arm point x^* at which Γ changes sign is reported as a function of the number of modes, whereas in the right table the maximum joint and tip tracking errors are reported for stable outputs.

# modes	x^* [m]
1	0.480
2	0.352
3	0.212
4	0.148

x [m]	# modes	max error [deg]	max error [mm]
0.20	1	0.0922	0.6900
0.20	2	0.0910	0.6900
0.20	3	0.0902	0.6896
0.34	2	0.0900	0.6758
0.47	1	0.0901	0.6696

5. DISCUSSION

The foregoing numerical results indicate the following facts:

- As expected, the inverse dynamics control has better tracking performance than the PD control + gravity compensation.
- For the joint output tracking case, there is no appreciable reduction in the error tracking (both at joint and at tip level) as the number of modes used in the controller is increased.
- The inverse dynamics control with feedforward of flexible states behaves as well as the full state inverse dynamics control.
- The range of arm points for which stable tracking can be obtained decreases as the number of modes increases; if only one mode is used it is possible to control nearly all the points along the arm.
- Compared to the case of joint control, the angular tracking error is about the same but remarkably the tip tracking error is reduced by an order of magnitude.

ACKNOWLEDGEMENT

This work was supported by *Consiglio Nazionale delle Ricerche* under contract n. 93.00905.PF67.

REFERENCES

[1] Nguyen, P.K., Ravindran, R., Carr, R., Gossain, D.M., Doetsch, K.H., "Structural flexibility of the Shuttle remote manipulator system mechanical arm," AIAA paper no. 82-1586, New York, NY, Aug. 1982.

[2] Book, W.J., "Recursive Lagrangian dynamics of flexible manipulator arms," *International Journal of Robotics Research*, vol. 3, no. 3, 1984.

[3] De Luca, A., Siciliano, B., "Closed-form dynamic model of planar multilink lightweight robots," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 21, no. 4, 1991.

[4] De Luca, A., Siciliano, B., "Inversion-based nonlinear control of robot arms with flexible links,"*AIAA Journal of Guidance, Control, and Dynamics*, vol. 16, no. 6, 1993.

[5] De Luca, A., Siciliano, B., "Trajectory control of a non-linear one-link flexible arm," *International Journal of Control*, vol. 50, no. 5, 1989.

[6] Cetinkunt, S., Yu, W.L., "Closed-loop behavior of a feedback-controlled flexible arm: A comparative study," *International Journal of Robotics Research*, vol. 10, no. 3, 1991.

[7] De Luca, A., Siciliano, B., "Regulation of flexible arms under gravity," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 3, 1993.

[8] Barbieri, E., Özgüner, Ü., "Unconstrained and constrained mode expansions for a flexible slewing link," *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 110, no. 4, 1988.

EXPERIMENTAL IDENTIFICATION OF THE INERTIAL PARAMETERS OF A ROBOT WITH CLOSED LOOP*

W. KHALIL, P. P. RESTREPO

Ecole Centrale de Nantes/Université de Nantes
Laboratoire d'Automatique de Nantes URA C.N.R.S 823
1, Rue de la Noë, 44072, Nantes cedex, FRANCE
E-Mail: Khalil@lan.ec-nantes.fr

ABSTRACT

This paper presents the identification model and the hardware system for the experimental identification of the inertial parameters of the 6 degree of freedom robot SR400, which is characterised by having a parallelogram closed kinematic chain and a mechanical coupling between the joints of the hand. The SR4OO robot is an industrial robot from Renault Automation, but we have replaced the classical control system by a Digital Signal Processor system from dSPACE™ (based on a TMS 320C30 Texas Instruments™ processor), in order to get an open control system with a big computational capacity; the position, velocity and motor currents are now available to direct measure in our system.

1. GEOMETRIC DESCRIPTION OF THE ROBOT

1.1 Geometric Parameters

The SR400 robot has 6 degrees of freedom. The number of moving links (denoted n) is equal to 8 and the number of joints (denoted N_j) is equal to 9, thus it contains a closed loop which is of parallelogram type. Its motors are synchronous and the nominal charge is 10kg. The description of the geometry of the robot is carried out using the modified Denavit and Hartenberg notation [1,2]. The coordinate frame j is fixed on link j, the z_j axis is along the axis of joint j, the x_j axis is along the common perpendicular of z_j and one of the succeeding axis on the same link. The geometry of the robot is defined by the following parameters (for j=1,..., N_f):

γ_j, b_j, α_j, d_j, θ_j, r_j define frame j with respect to its antecedent frame a(j), with θ_j is the joint variable, for j rotational.

- σ_j defines the type of joint. $\sigma_j = 0$ for j rotational, $\sigma_j = 1$ for j translational.
- a(j) denotes the frame antecedent to frame j,
- μ_j indicates if the joint j is motorised (active, μ_j= 1) , or not (passive, μ_j= 0).

In the case of closed loop robot, the geometric parameters are determined for an equivalent tree structure by opening each closed loop. Then two frames are added on the opened joints [1,2], thus the number of frames is equal to: $N_f = n + 2B$, where B = number of closed loops = N_j - n.
The geometric parameters of the SR400 Robot are given in table 1.

j	a(j)	μ_j	σ_j	γ_j	b_j	α_j	d_j	θ_j	r_j
1	0	1	0	0	0	0	0	θ_1	0
2	1	1	0	0	0	-90	d_2	θ_2	0
3	2	0	0	0	0	0	d_3	θ_3	0
4	3	1	0	0	0	-90	d_4	θ_4	RL4
5	4	1	0	0	0	90	0	θ_5	0
6	5	1	0	0	0	-90	0	θ_6	0
7	1	1	0	0	0	-90	d2	θ_7	0
8	7	0	0	0	0	0	d_8	θ_8	0
9	8	0	0	0	0	0	$d_9=d_3$	θ_9	0
10	3	0	0	90	0	0	$d_{10}= -d_8$	0	0

Table 1: Geometric parameters of SR400 robot.

* This work is realised in collaboration with Renault Automation and the C.E.A., by the help of the French Ministry of Research and Space.

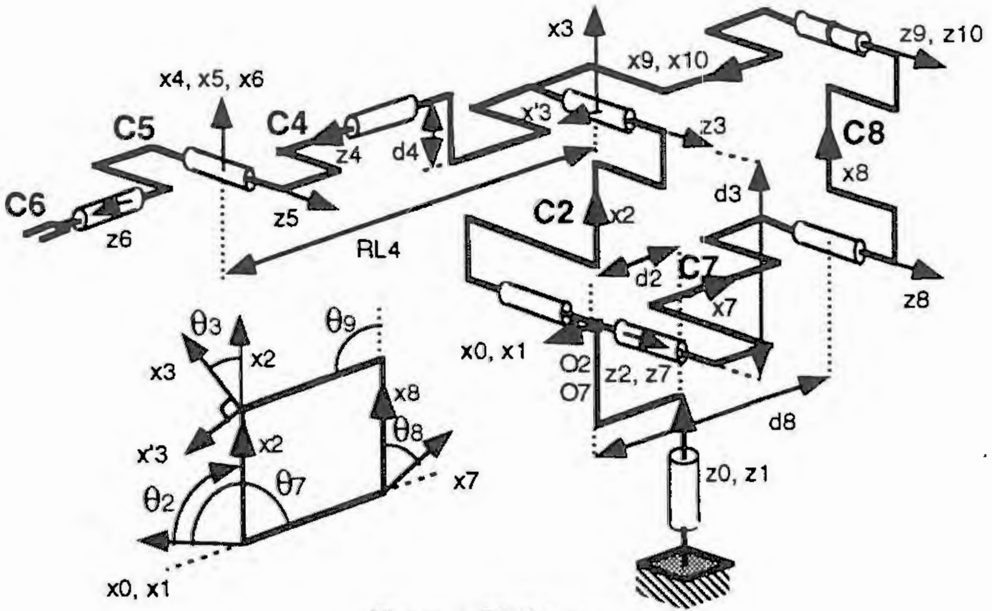


Figure 1. SR400 robot.

1.2 Closed loop geometric constraint equations

Owing to the parallelogram loop, links 3 and 7 are always parallel, and links 2 and 8 are also parallel so the following geometric constraint equations are verified :

$$^{7}\mathbf{A}_3 = {}^{7}\mathbf{A}_1\, {}^{1}\mathbf{A}_2\, {}^{2}\mathbf{A}_3 = \mathbf{A}(\mathbf{z}, -\theta_7+\theta_2+\theta_3) = \mathbf{A}(\mathbf{z}, \pi/2)$$

and

$$^{2}\mathbf{A}_8 = {}^{2}\mathbf{A}_1\, {}^{1}\mathbf{A}_7 {}^{7}\mathbf{A}_8 = \mathbf{A}(\mathbf{z}, -\theta_2)\, \mathbf{A}(\mathbf{z}, \theta_7)\, \mathbf{A}(\mathbf{z}, \theta_8) = \mathbf{A}(\mathbf{z}, 0)$$

where $\mathbf{A}(\mathbf{u},\theta)$ gives the 3x3 orientation matrix corresponding to a rotation θ around the axis $\mathbf{u}$.
Thus the values of the passive joints as functions of the active joints are given as:

$$\theta_3 = \pi/2 - \theta_2 + \theta_7 \quad , \quad \theta_8 = \theta_2 - \theta_7 \quad , \quad \theta_9 = \pi/2 + \theta_3 = \pi - \theta_2 + \theta_7 \qquad (1)$$

1.3 Geometric constraint equations due to mechanical coupling

The motors 5 and 6 are fixed on link 4, thus a mechanical coupling exists between the joints 5 and 6 such that:

$$\dot{\theta}_6 = \dot{\psi}_6 - \dot{\theta}_5 \qquad (2)$$

where: $\dot{\psi}_6$ is the velocity of motor 6 referred to the joint side.

2. THE DYNAMIC MODEL

The inverse dynamic model calculates the motor torques or forces as a function of the joint positions, velocities and accelerations.
The joint variables of the equivalent tree structure of the robot can be written as:

$$\mathbf{q}_{tr} = \begin{bmatrix} \mathbf{q}_a \\ \mathbf{q}_p \end{bmatrix}$$

where $\mathbf{q}_a$, $\mathbf{q}_p$ represent the variables of active and passive joints (of the equivalent tree structure) respectively.
The dynamic model of the closed-loop structured robot can be obtained as a function of the corresponding tree structure dynamic model using the following relation [2,4]:

$$\Gamma_m = \mathbf{G}^T \Gamma_{tr} = \Gamma_a + \mathbf{W}^T \Gamma_p \qquad (3)$$

where:
Γ_m is the (mx1) vector of the torques of motorised joints of the robot.
$\mathbf{G} = \frac{\partial \mathbf{q}_{ar}}{\partial \mathbf{q}_a}$ is the Jacobian matrix of $\mathbf{q}_{tr}$ with respect to $\mathbf{q}_a$,
$\mathbf{W} = \frac{\partial \mathbf{q}_p}{\partial \mathbf{q}_a}$ and Γ_{tr} is the (nx1) vector of the joint torques (or forces) of the corresponding tree structure.

Γ_a , Γp are the parts of Γ_{tr} corresponding to the active and passive joints respectively.
The dynamic model of the tree structure of the SR400 robot calculated using the base inertial parameters (see section 2.2), needs 347 multiplications and 300 additions. It is obtained using the SYMORO+ package which uses the customised Newton Euler algorithm [12].

In the case of the SR400 robot, we have :

$$q_a = [\,\theta_1\ \theta_2\ \theta_4\ \theta_5\ \theta_6\ \theta_7\,]^T \quad , \quad q_p = [\,\theta_3\ \theta_8\,]^T \tag{4}$$

using relations (1) and (3), taking into account the coupling between 5 and 6, and adding the effect of motor inertia and frictions, we get:

$$\Gamma_{m1} = \Gamma_{ar1} + I_{a1}\ \ddot{q}_1 + F_{s1}\ \mathrm{Sign}(\dot{q}_1) + F_{v1}\ \dot{q}_1$$

$$\Gamma_{m2} = \Gamma_{ar2} - \Gamma_{ar3} + \Gamma_{ar8} + I_{a2}\ \ddot{q}_2 + F_{s2}\ \mathrm{Sign}(\dot{q}_2) + F_{v2}\ \dot{q}_2$$

$$\Gamma_{m4} = \Gamma_{ar4} + I_{a4}\ \ddot{q}_4 + F_{s4}\ \mathrm{Sign}(\dot{q}_4) + F_{v4}\ \dot{q}_4$$

$$\Gamma_{m5} = \Gamma_{ar5} - \Gamma_{ar6} + I_{a5}\ \ddot{q}_5 + F_{s5}\ \mathrm{Sign}(\dot{q}_5) + F_{v5}\ \dot{q}_5 + F_{s56}\ \mathrm{Sign}(\dot{q}_5) + F_{v56}\ \dot{q}_5$$

$$\Gamma_{m6} = \Gamma_{ar6} + I_{a6}\ \ddot{\psi}_6 + F_{s6}\ \mathrm{Sign}(\dot{\psi}_6) + F_{v6}\ \dot{\psi}_6 + F_{s65}\ \mathrm{Sign}(\dot{q}_6) + F_{v65}\ \dot{q}_6$$

$$\Gamma_{m7} = \Gamma_{ar7} + \Gamma_{ar3} - \Gamma_{ar8} + I_{a7}\ \ddot{q}_7 + F_{s7}\ \mathrm{Sign}(\dot{q}_7) + F_{v7}\ \dot{q}_7 \tag{5}$$

Fv_j, Fs_j are the viscous and static friction coefficients of motor j and F_{v56}, F_{v65}, F_{s56} and F_{s65} are the viscous and static friction coefficients due to the coupling between joints 5 and 6, and I_{aj} is the inertia of motor j referred to the joint side.
To use this model in the robot control, we have to identify the links inertial parameters and friction coeficients appearing in the previous equations.

2.1 Standard Inertial parameters of the SR400 robot

These parameters are composed of the inertial parameters of links, and motors:
For each link the following inertial parameters are defined:

$$X^j = [XX_j, XY_j, XZ_j, YY_j, YZ_j, ZZ_j, MX_j, MY_j, MZ_j, Mj, Iaj],$$

where:

- $(XX_j, XY_j, XZ_j, YY_j, YZ_j, ZZ_j)$ are the elements of the inertia matrix ${}^jJ_j$, defining the inertia matrix of link j around the origin of frame j,
- (MX_j, MY_j, MZ_j) are the elements of ${}^jMS_j$ defining the first moments of link j, around the origin of frame j,
- M_j the mass of link j.

X^j and the parameters I_{aj}, F_{sj} and F_{vj} for j = 1,...,n, constitute the dynamic parameters of the robot.

2.2. The base inertial parameters

The base inertial parameters are defined as the minimum parameters which can be used to get the dynamic model. They represent the set of parameters which can be identified using the dynamic or energy model, thus its determination is essential for the identification of the inertial parameters of robots [5,6,7]. The use of the base parameters in Newton-Euler algorithm leads to reduce the computation complexity of the model [3].
These parameters can be obtained from the standard inertial parameters by eliminating those which have no effect on the dynamic model and by regrouping some others. In [7,8] we have presented a symbolic method to calculate these parameters for serial or closed loop robots. These algorithms have been programmed in SYMORO+. The final results of the minimum inertial parameters of the SR400 robot can be summarised as follows:
The following 11 parameters have no effect on the dynamic model :

XX_1, XY_1, XZ_1, YY_1, YZ_1, MX_1, MY_1, MZ_1, M_1, MZ_2, and MZ_7.

The number of base inertial parameters of the SR400 robot is 42, they are given in table 2.
The regrouped relations as :

$$MXR_7 = MX_7 - MX_8\,\frac{d8}{d3} + M_8\ d_8 \quad , \quad XXR_6 = XX_6 - YY_6$$

$$XXR_5 = XX_5 + YY_6 - YY_5 \quad , \quad ZZR_5 = ZZ_5 + YY_6$$

$$MYR_5 = MY_5 + MZ_6 \quad , \quad XXR_4 = XX_4 + YY_5 - YY_4$$

$$ZZR_4 = ZZ_4 + YY_5 \quad , \quad MYR_4 = MY_4 - MZ_5$$

$$XYR_3 = XY_3 - XY_7 - d4\ MZ_4 - d4\ RL4\ MR4 \quad , \quad XZR_3 = XZ_3 + YZ_7$$

$$YZR_3 = YZ_3 - XZ_7 + MZ_8{*}d8 \quad , \quad MXR_3 = MX_3 + d4\ MR_4$$

$$MYR_3 = MY_3 - MX_8 \frac{d8}{d3} + MZ_4 + RL_4\ MR_4 \quad , \quad XXR_2 = XX2 + XX8 - YY_2 - YY_8 - d3^2\ MR_3$$

$$XYR_2 = XY_2 + XY_8 \quad , \quad XZR_2 = XZ_2 + XZ_8 - d_3\ MZ_3$$

$$YZR_2 = YZ_2 + YZ_8 \quad , \quad ZZR_2 = ZZ2 + ZZ8 + d_3{}^2\ MR3 + I_{a2}$$

$$MXR_2 = MX_2 + MX_8 + d3\ MR_3$$

$$XXR_3 = XX3 + YY7 + M8{*}d_8{}^2 + YY_4 + 2\ RL4\ MZ_4 + RL4^2\ (MR4) - YY3 - XX7 - d_4{}^2\ MR4$$

$$ZZR_3 = ZZ_3 + ZZ_7 + YY_4 + 2\ RL4\ MZ_4 + (d_4{}^2 + RL4^2)\ MR4 + Ia_7 + d_8{}^2{*}M_8$$

$$ZZR_1 = ZZ1 + YY_2 + YY_8 + YY3 + XX7 + d_4{}^2\ MR4 + d3^2\ MR_3 + d_2{}^2\ MR2 + I_{a1} + (M_7 + M_8){*}\ d_2{}^2$$

where: $MR_4 = M_4 + M_5 + M_6$, $MR_2 = M_2 + M_3 + M_4 + M_5 + M_6$, $MR_3 = M_3 + M_4 + M_5 + M_6$

j	XX_j	XY_j	XZ_j	YY_j	YZ_j	ZZ_j	MX_j	MY_j	MZ_j	M_j	Iaj
1	0	0	0	0	0	ZZR_1	0	0	0	0	0
2	XXR_2	XYR_2	XZR_2	0	YZR_2	ZZR_2	MXR_2	MY_2	0	0	0
3	XXR_3	XYR_3	XZR_3	0	YZR_3	ZZR_3	MXR_3	MYR_3	0	0	0
4	XXR_4	XY_4	XZ_4	0	YZ_4	ZZR_4	MX_4	MYR_4	0	0	Ia_4
5	XXR_5	XY_5	XZ_5	0	YZ_5	ZZR_5	MX_5	MYR_5	0	0	Ia_5
6	XXR_6	XY_6	XZ_6	0	YZ_6	ZZ_6	MX_6	MY_6	0	0	Ia_6
7	0	0	0	0	0	0	MXR_7	MY_7	0	0	0
8	0	0	0	0	0	0	0	MY_8	0	0	0

Table 2: Base inertial parameters of the SR400 robot

Owing to the symmetry of some links the following 10 parameters are supposed equal to zero:

MY_7, MX_5, XY_5, YZ_5, XZ_5, MY_4R, MX_4, XY_4, YZ_4, XZ_4.

3. THE CONTROL SYSTEM [13,14]

Our control system is based on a DSP TMS 320C30 Texas Instruments™ from dSPACE™ (processor) and a 486 (33MHz) PC computer. There will be two programs running in parallel, one in the DSP (which makes the measuring and control) and one in the host computer, which can be any of the utility modules made by dSPACE or our programs, to realise the interface with the user.

The main advantages of this DSP configuration is the high execution speed, the real time tracing the possibility of pseudo parallel execution and the direct measure of several digital and analog signals. Thus, we have the possibility of measuring the velocity and current images of the motors and generating any desired trajectory.

We have developed some DSP and Host C language programs for the trajectory generation, using either classical bang-bang with velocity step or 5th degree polynomials

In addition, the TRACE™ module from dSPACE allows the real time tracing of any of the variables of the DSP program.

4. THE IDENTIFICATION MODEL OF DYNAMIC PARAMETERS

4.1 The dynamic identification model

Since the dynamic model is linear in the dynamic parameters, it can be used to identify the dynamic parameters [5,6], and can be written as follows:

$$\Gamma = \mathbf{D}\ (\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\ \mathbf{K} \tag{6}$$

where

K is the vector of dynamic parameters $\mathbf{K} = [\ \mathbf{X}^T\ \ \mathbf{FS}^T\ \ \mathbf{FV}^T\ ]^T$ (base inertial parameters and Coulomb and viscous friction coeficients vectors respectively).

D is (nxNp) matrix.

4.2 The energy identification model

Using the dynamic model in the identification need to estimate or measure the joint accelerations. To overcome this difficulty, a model based on the energy theorem has been proposed [9]. From the energy theorem we get:

$$y = \int_{t1}^{t2} \dot{q}_m{}^T \Gamma_m\, dt = H(t2) - H(t1) + \sum_{j=1}^{n} F_{sj}.\left[\int_{t1}^{t2} |\dot{q}_j|.dt\right] + \sum_{j=1}^{n} F_{vj}.\left[\int_{t1}^{t2} \dot{q}_j{}^2.dt\right] \tag{7}$$

where $H(t_i) = \mathbf{h}(\mathbf{q},\dot{\mathbf{q}}).\mathbf{X}$ is the total energy (kinetic and potential) at time t_i, and $\dot{q}_{mj}$ and $\dot{q}_j$ are the velocities of motor j and joint j, respectively.

Since H is linear in the inertial parameters then (7) can be written as a linear equation in the dynamic parameters of the robot:

$$y = \mathbf{w}(\mathbf{q},\dot{\mathbf{q}})\ \mathbf{K} \qquad (8)$$

where **w** denotes a row matrix.
Relation (8) is function of the joint positions, velocities and of the input joint torques.
To identify the dynamic parameters a sufficient number of equations can be obtained by calculating relation (8) between different intervals of time. These row matrices w will be the rows of matrix W, which defines an overdetermined linear system. The least squares solution is generally used in this identification.
In order to improve the identification process, an exciting trajectory using the following criterion will be used [11]:

$$f(\mathbf{W},\mathbf{Z}) = \mathrm{Cond}(\mathbf{W}.\mathbf{diag}(\mathbf{Z})) + \frac{1}{\Sigma_p(\mathbf{W}.\mathbf{diag}(\mathbf{Z}))} + \frac{1}{\mathrm{Y_{am}}} \qquad (9)$$

where: $\mathrm{Y_{am}} = \min_i|\mathbf{Y_{ap}}(i)|$ and $\mathbf{Y_{ap}}$ is the a priori values of dynamic parameter vector.
$\Sigma_p(\mathbf{A})$ denotes the smallest eigenvalue of matrix **A**
Z is the a priori dynamic parameters values vector.
diag(Z) is a square matrix with the elements of Z in its diagonal

The a priori information about the solution vector is available from C.A.D. system or can be obtained from special motion tests.

5. IDENTIFICATION PROCEDURE

5.1 Identification of motor constants

5.1.1. Drive Gain

To apply relation (5), we need to calculate the motor torques from the current image, which is the measured signal.
Thus: $\Gamma_j = G_{Tj}\ .\ V_{Tj}$, where V_{Tj} is the joint current image and G_{Tj} is the drive gain of motor j.
The experimental determination of G_T is done by fixing a disk to the terminal link of the robot and by applying a constant velocity steps signals. We do this experiment with the disk alone, first, and then, we repeat this experiment with different masses suspended to the disk as shown in the figure. 2 [15]
If the experiments, with or without the suspended mass are such that $\dot q$ is the same, we can say that the torque expressions are (the joint axis is positioned to be orthogonal to gravity):

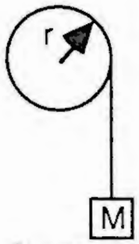


Figure 2

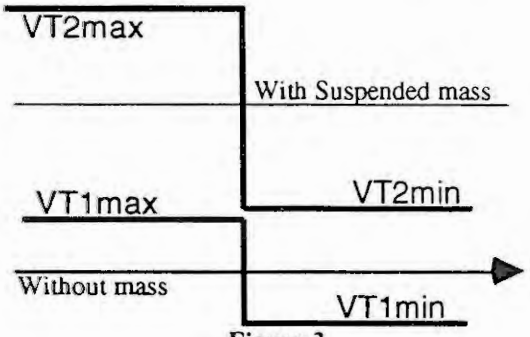


Figure 3

With suspended mass M (g is the acceleration of gravity):

$$\Gamma_2 = G_{T2}\ V_{T2} = M.g.r + F_v.\dot q + F_s.\mathrm{sign}(\dot q) \qquad (10)$$

Without suspended mass (Figure 3)

$$\Gamma_1 = G_{T1}\ V_{T1} = F_v.\dot q + F_s.\mathrm{sign}(\dot q)$$

We have then:

$$M.g.r = GT1.[\ V_{T2max} - V_{T1max}\ ] \qquad (11)$$

and

$$M.g.r = GT2.[\ V_{T2min} - V_{T1min}\ ] \qquad (12)$$

And G_T will be the mean of G_{T1} and G_{T2}.
This experiment is easily done for axes 4 and 6, but for the other axes it is difficult. In [16], it has been shown that with the knowledge of G_{Ti}, for one axis, the other ones can be determined.

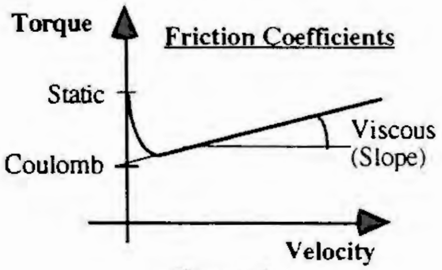


Figure 4

5.1.2. Friction parameters (Figure 4)
Equation (7) can be also used to identify the friction coefficients, but it is preferable to identify them separately

Static Friction:

For the determination of the static friction, we apply a triangular current signal with low frequency (0.1 Hz) and low peak to peak amplitude (0.3 A peak to peak) in order to evaluate, as precisely as possible, the motor torque value when the joint begins to move. To do that, we trace in real time both signals with the DSP system described above.

Coulomb and viscous coefficients:

For several velocity values, from about 3% to nominal speed, we apply classical bang-bang trajectories, with constant velocity steps. The torque signals corresponding to each velocity step, is filtered and the mean is taken to define the corresponding value of $F_{vj}.\dot{q}_j + F_s.\mathrm{sign}(q_j)$. F_{vj} and F_{sj} can be determined from the linear interpolation of all the velocity values

We have remarked that F_v and F_s must be defined once for positive velocities, and once for negative velocities, and that the mechanical coupling of joints 5 and 6 introduces an interaction friction F_{56} and F_{65} seen in equation (5).

5.2 Identification of link parameters

The link parameters will be identified using the exciting trajectories introduced in section 4.2 and developed in [13], and using the energy model (section 4.2). These trajectories will be, at first, executed without charge on the robot, then with a given charge.

When identifying the parameters without charge, the parameters MX6, MY6, XY6 XX6R, YZ6, XZ6 of link 6 are supposed to be equal to zero, owing to the symmetry of link 6.

To identify the parameters of the charge, we need to re-identify the following parameters: XX6R, XY6, XZ6, YZ6, ZZ6, MX6, MY6, XX5R, ZZ5R and MY5R.

6. CONCLUSION

This paper presents the identification model of the dynamic parameters and the experimental installation of the SR400 robot, the base inertial parameters are given, and the identification model using the energy theorem is presented.

The identification procedure takes into account the calculation of only one motor drive gain, and identifies the joint friction coeficients.

It is to be noted that this procedure needs access to motor reference currents and uses particular trajectories which cannot be obtained using a classical industrial control system. Therefore an open control system is needed.

Owing to the allowed number of pages, the identified numerical values and the curves of the experiments are not given.

REFERENCES

[1] Khalil W., Kleinfinger J.-F., "A new geometric notation for open and closed-loop robots", *Proc. IEEE Conf. on Robotics and Automation*, San Francisco, 1986, p. 1174-1180.

[2] Dombre E., Khalil W., "Modélisation et commande des robots", *Edition Hermès*, Paris, 1988.

[3] Khalil W., Kleinfinger J.-F., "Minimum operations and minimum parameters of the dynamic model of tree structure robots", *IEEE J. of Robotics and Automation*, Vol. RA-3(6),1987, p. 517-526.

[4] J.-F. Kleinfinger, W Khalil , " Dynamic modelling of closed-chain robots", Proc 16$^{Th.}$ Int. Symp. on Industrial Robots, Brussels, sept-oct. 1986, p. 401-412.

[5] An C. H., Atkeson C.G., Hollerbach J.M., " Estimation of inertial parameters of rigid body links of manipulators". Proc. 24th Conf. on Decision and control, 1985, p. 990-995.

[6] Khosla P.K.,Kanade T., "Parameter identification of robot dynamics " . Proc. 24th Conf. on Decision and Control. 1985, p. 1754-1760.

[7] Gautier M., Khalil W., "A direct determination of minimum inertial parameters of robots", *Proc. IEEE Conf. on Robotics and Automation*, Philadelphia,1988, p. 1682-1687.

[8] F. Bennis, W. Khalil, "Minimum inertial parameters of robots with parallelogram closed-loops", IEEE Conference of robotics and automation, Cincinnati,May 90,p.1026-1031.

[9] Gautier M., Khalil W., "On the identification of the inertial parameters of robots", Proc. 27 th CDC .1988.

[10] Khalil W., Gautier M. "Modélisation du Robot SR400", Rapport Intérmediare projet DEMMAR, Mars 1992.

[11] Pressé C.,Gautier M., "New Criteria of exciting trajectories for Robot Identification", ICRA. *Proc. IEEE Conf. on Robotics and Automation*, Atlanta, 1993, p. 907-912.

[12] Khalil W., Bennis F., Chevallereau C., Creusot D., SYMORO+, SYmbolic MOdelling of RObots, User's guide, Nantes, October 1993, E.C.N.-Nantes.

[13] dSAPCE GmbH, "DSP-CITpro Hardware Installation Guide", Document version 1.0, 1992

[14] dSAPCE GmbH, "DSP-CITpro TRACE30 User's Guide", Document version 2.0, 1992

[15] Gaudin H., "Contribution à l'identification in situ des constantes d'inertie et des lois de frottement articulaires d'un robot manipulateur en vue d'une application expérimentale au suivi de trajectoires optimales", Thèse de Doctorat, Poitiers, 1992

[16] Khalil W.,Gautier M., "Computed Current Control Of Robots", *IFAC 12th World Congress*, Sydney, 1993, p. 129-134.

SIMULATING DISCONTINUOUS PHENOMENA AFFECTING ROBOT MOTION

GIANNI FERRETTI, CLAUDIO MAFFEZZONI, GIANANTONIO MAGNANI, PAOLO ROCCO

Dipartimento di Elettronica e Informazione,
Politecnico di Milano, Piazza Leonardo da Vinci, 32, 20133 Milano - Italy

Abstract: The paper deals with simulation of mechanical systems affected by discontinuous phenomena. These phenomena involve impulsive events and/or models whose structure changes depending on the values of some system variables. The models of three kinds of these discontinuities (joint with static friction, collisions with rigid environment, bifurcation behavior near kinematic singularities) are given, and a simulation environment, based on the DAE solver DASSL is presented, that also allows efficient simulation of sample-data systems. Some simulation results achieved with the proposed environment are finally presented.

1. INTRODUCTION

Dynamic simulation of robot motion is currently the subject of considerable research effort, both for mechanical and control system design purposes. Unfortunately, detailed robotic simulation is hampered by the complex and non-linear nature of some phenomena that affect robot motion. A number of efficient packages (see [1], [2], [3] to name just a few) have been developed for computer-aided generation of dynamic models of manipulators, mainly in open loop configurations, which however do not take into account some typical events in real robot behavior, best modelled as discontinuities. Examples are transition from rest to motion, or viceversa, in joints affected by static friction [4], [5], collisions with stiff surfaces [6], bifurcations at kinematic singularities. Moreover, simulation of robots equipped with digital control systems requires a careful design of the interface between the computer executed algorithms and the numerical solver of the continuous time part of the system, particularly to avoid unnecessary limitations to the integration step due to the sampling times of the controllers. Special concepts are therefore required for reliable and efficient simulation of such systems.

On the other hand, standard numerical solver for ODE (Ordinary Differential Equations) systems prove to be inadequate for most of the simulations involving complex mechanical systems. In fact mechanical models often take the form of implicit DAE (Differential Algebraic Equations) systems, where the algebraic relationships typically arise from the presence of mechanical constraints (closed kinematic loops, interaction with rigid environment).

Based on the above arguments, a new robotics simulation environment is being built around one of the most efficient numerical solvers for DAE systems, DASSL. A special version of the solver, called DASSLRT, has been used, that detects user-defined events depending on state variables, allowing accurate simulation of discontinuous phenomena.

This paper first discusses the models of some discontinuous phenomena affecting robot motion and then how these models can be efficiently coded using the DASSL-based environment. The environment allows simulation of arbitrary order DAE systems possibly affected by discontinuous events (impulsive behaviour or sudden changes in the equations describing the system) and interacting with digital controllers.

In particular, Section 2 discusses the models of friction in the joints, interaction with stiff environments, and constrained mechanical systems near singular configurations, while Section 3 illustrates how to treat them by the solver DASSLRT. Section 4 describes the robotics simulation environment, Section 5 presents some simulations performed with the said environment and Section 6 concludes the paper with a few remarks on the results achieved.

2. DISCONTINUITIES IN THE DYNAMICS OF MECHANICAL SYSTEMS

2.1 Joint friction

Apart from the case of steady state analysis and design of rotating machines, where tribology has achieved considerable results in explaining the atomic details of friction [4], the theory of friction in dynamic conditions is still incomplete. In particular, joint friction models are often given the form of characteristics relating the joint velocity ω to the friction torque τ_F, derived by fitting experimental observations. A typical characteristic is

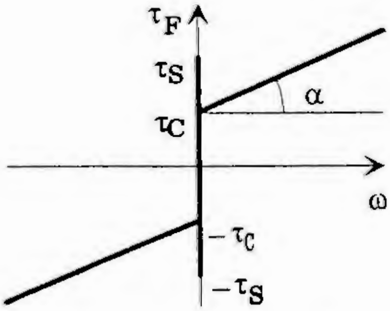


Fig. 1 Friction characteristic

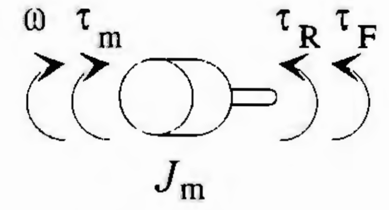


Fig.2 Motor

shown in Fig. 1, where three main parameters are pointed out: the kinetic friction torque τ_C, independent of velocity; the viscous friction coefficient $D_m = tg\ \alpha$ and the static friction torque τ_S, defined as the minimum torque needed to start motion.

The most severe problems connected to the model of Fig. 1 are encountered in the simulation of velocity reversals, and are illustrated by considering the simple motor model sketched in Fig. 2, where J_m, τ_R and τ_m are the rotor inertia, the externally applied torque and the motor torque, respectively.
Suppose that the motor is decelerating and at instant t the velocity vanishes: $\omega(t) = 0$ with $\dot{\omega}(t^-) < 0$. According to the model of Fig. 1, the velocity will reverse and the friction torque will instantaneously switch from $\tau_F(t^-) = \tau_C$ to $\tau_F(t^+) = -\tau_C$ only if $\tau_m - \tau_R < -\tau_C$; in that case $\dot{\omega}(t^+) = 1/J_m\ (\tau_m - \tau_R + \tau_C) < 0$, otherwise motion will stop, i.e. $\dot{\omega}(t^+) = 0$; motion will start again only when $|\tau_m - \tau_R| > \tau_S$.

2.2 Interaction with a stiff environment

The most straightforward method to model the interaction between a robot arm and the environment is to assign a suitable compliance to the interaction surfaces, so as to relate the reaction forces to the strain the surfaces undergo. However, in case of collisions with very stiff surfaces, huge contact forces are originated in very short time intervals, leading to numerical problems. Moreover, an accurate prediction of the motion after the collision is strictly related to the correct evaluation of the hysteresis cycles modelling energy dissipation in the collision [7], thus requiring the adoption of integration steps much smaller than the duration of the contact.
A more efficient and accurate simulation of collisions with stiff surfaces can be obtained by resorting to a rigid model of the interaction, worked out on the basis of the impulsive dynamics. As an example, consider the case of the point contact of the end effector of a robot with a frictionless stiff surface:

$$\mathbf{D}(\mathbf{q})\,\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} + \mathbf{J}(\mathbf{q})^T f_n\, \mathbf{n}$$

where q is the generalized coordinates vector, $\mathbf{D}(\mathbf{q})$ is the inertia matrix, $\mathbf{h}(\mathbf{q},\dot{\mathbf{q}})$ accounts for Coriolis and centrifugal terms and $\mathbf{g}(\mathbf{q})$ for gravitational terms, $\boldsymbol{\tau}$ is the joint torque vector, $\mathbf{J}(\mathbf{q})$ is the Jacobian matrix of the manipulator, f_n is the amplitude of the interacion force and $\mathbf{n}$ is the normal to the interaction surface. A rigid collision can be modelled through a finite variation $\Delta\dot{\mathbf{q}}$ imposed to robot motion by a finite impulse, acting on the end effector in the instant of collision t, detected by a null value of the relative distance of the end effector from the surface ($\mathbf{x}^-$ stands for $\mathbf{x}(t^-)$, being $\mathbf{x}$ a generic variable):

$$\Delta\dot{\mathbf{q}} = -\frac{(1+e)\,(\mathbf{n}^-)^T\,\mathbf{v}^-\ \mathbf{D}(\mathbf{q}^-)^{-1}\ \mathbf{J}(\mathbf{q}^-)^T\,\mathbf{n}^-}{(\mathbf{n}^-)^T\,\mathbf{J}(\mathbf{q}^-)\ \mathbf{D}(\mathbf{q}^-)^{-1}\ \mathbf{J}(\mathbf{q}^-)^T\,\mathbf{n}^-}$$

being e the Poisson's coefficient, accounting for energy dissipation, and $\mathbf{v}$ the Cartesian linear velocity of the end effector (in case of contact friction the computation of $\Delta\dot{\mathbf{q}}$ is much more involved, [7]).
It must be also pointed out that the capability of handling discontinuities allows a clear and rigorous modelling of the transitions among different motion regimes [7], namely motion in free space, collision and constrained motion, where, the bouncing phase being exhausted, the reaction forces are implicitly defined as those that maintain fulfilment of the kinematic constraints modelling the interaction with a rigid surface [8].

2.3 Constrained mechanical systems and singular configurations

Every constrained mechanical system can be modelled by a mixed DAE system [9]:

$$\mathbf{M}(\mathbf{q})\,\ddot{\mathbf{q}} + \mathbf{V}(\mathbf{q},\dot{\mathbf{q}}) + [\partial\Phi(\mathbf{q})/\partial\mathbf{q}]^T\,\boldsymbol{\lambda} = \boldsymbol{\tau} \qquad (1.a)$$

$$\Phi(\mathbf{q}) = 0 \qquad (1.b)$$

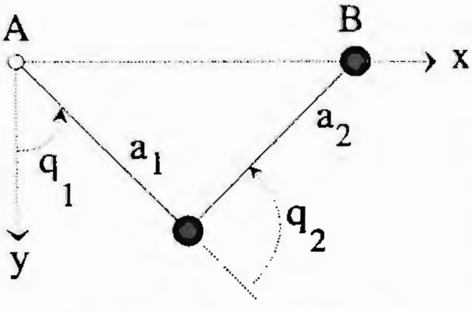


Fig. 3 The horizontal double pendulum

where $\mathbf{M}(\mathbf{q})\,\ddot{\mathbf{q}} + \mathbf{V}(\mathbf{q},\dot{\mathbf{q}}) = \boldsymbol{\tau}$ are the motion equations of the unconstrained system, being $\boldsymbol{\tau}$ the vector of external, position-independent applied forces, and $\boldsymbol{\lambda}$ are the Lagrange multipliers introduced to account for the effect of the kinematic constraints $\boldsymbol{\Phi}(\mathbf{q}) = 0$. A unique solution exists for system (1) if matrix $\partial\boldsymbol{\Phi}(\mathbf{q})/\partial\mathbf{q}$ (Jacobian of the constraints equations) has full row rank in every admissible configuration of the generalized coordinates $\mathbf{q}$ [9]; otherwise the motion of the system is not uniquely determined and bifurcations may arise [9]: some internal reaction forces may tend to infinity, yielding to abrupt discontinuities in the motion of the system.

To show how bifurcation may occur in singular configurations we will consider the same example treated by Ellis and Ricker [10]: the horizontal double pendulum, depicted in Fig. 3. Both links are massless and two point masses of 1 *Kg* are located at the distal end of each link. Link a_1 is connected to the ground by a rotary joint and is 1 *m* long, link a_2 is coupled to link a_1 by a rotary joint while its distal end is constrained to move along the x-axis; the length of this last link is assumed to be $(1 + \varepsilon)$ *m*. The only action applied to the system is the gravity force, directed along the y-axis: this implies that the total mechanical energy of the system must remain constant. This mechanical system can be modelled through the equivalent open loop chain, obtained by removing the constraint at the distal end of the second link and adding the equation of the removed constraint:

$$\phi(\mathbf{q}) = \phi([q_1 \quad q_2]^T) = \cos(q_1) + (1+\varepsilon)\cos(q_1 + q_2) = 0 \quad .$$

If $\varepsilon = 0$, the configuration $\mathbf{q} = [0 \quad \pi]$ is *singular*. However, starting, for instance, from the initial condition $\{\mathbf{q} = [\pi/2 \quad 0]^T,\ \dot{\mathbf{q}} = 0\}$, the motion of the system could be assumed continuous while passing through the singularity so that the abscissa of point B would oscillate from −2 to +2. On the other hand, this solution is actually singular (note that this is the only solution considered in [10]), in the sense that for any $\varepsilon \neq 0$ the only (non singular) solution of motion is definetely different from that solution. In fact, if $\varepsilon \neq 0$ (no matter how small $|\varepsilon|$ is) singular configurations do not exist; moreover, as $\varepsilon \to 0^-$ or $\varepsilon \to 0^+$ the motion of the system asymptotically undergoes a finite bump as point B approaches point A[1], switching between the two branches of the solution of the constraint equation with $\varepsilon = 0$ (S_1: $q_2 = -2q_1 + \pi$; S_2: $q_2 = \pi$). The finite bump can be computed by imposing the conservation of the mechanical energy, i.e. the kinetic energy, since the configuration of the system does not vary in the switch.

Therefore, the behavior of the system at the singular configuration can be classified as follows:

a) $\varepsilon \to 0^-$: S.1 → S.2, with $\dot{\mathbf{q}}^+ = [\,-\sqrt{5}\,\dot{q}_1^- \quad 0\,]^T$ and S.2 → S.1 with $\dot{\mathbf{q}}^+ = [\,-1/\sqrt{5}\,\dot{q}_1^- \quad +2/\sqrt{5}\,\dot{q}_1^-\,]^T$;

b) $\varepsilon \to 0^+$: S.1 → S.2, with $\dot{\mathbf{q}}^+ = [\,+\sqrt{5}\,\dot{q}_1^- \quad 0\,]^T$ and S.2 → S.1 with $\dot{\mathbf{q}}^+ = [\,+1/\sqrt{5}\,\dot{q}_1^- \quad -2/\sqrt{5}\,\dot{q}_1^-\,]^T$;

c) $\varepsilon = 0$: the motion has a bifurcation among three possible solutions: either no bumps, or the same as a) or the same as b).

3. THE DAE SOLVER DASSL

DASSL [11] is a numerical code widely used for the solution of DAE systems of the form:

$$F(t,y,\dot{y}) = 0 \tag{2.a}$$

$$y(t_0) = y_0 \tag{2.b}$$

$$\dot{y}(t_0) = \dot{y}_0 \tag{2.c}$$

where F, y, $\dot{y}$ are *N*-dimensional vectors. To solve the DAE system, DASSL approximates the derivatives by means of the *k*th Backward Differentiation Formula (BDF), with *k* ranging from one to five: in particular it implements a variable integration step and variable order version of the BDF method. DASSL adopts a predictor-corrector scheme: the predictor makes a first guess of the solution at the new integration point, while

[1] This fact can be intuitively justified by noting that, for $|\varepsilon| \neq 0$, the abscissa of point B is prevented from becoming negative; to respect this constraint, the smaller is $|\varepsilon|$, the faster are the velocity variations of point B when approaching point A.

the corrector obtains the final solution by numerically solving the implicit algebraic system which derives from the substitution of the derivative in (2.a) with the BDF formula, by means of an iterative process, initialized with the predictor formula.

DASSL requires a user subroutine to compute function F in (2.a): the subroutine receives as inputs the current time t and the vectors y and $\dot{y}$ and outputs the *residuals* of function F, i.e. the vector valued function F for the given inputs. The Jacobian of function F, used in the solution of the implicit system, can be numerically computed by DASSL itself, or it can be supplied by a user-defined function in order to speed up the iterating process. Among the distinctive features of DASSL it is worth remarking the use of a different tolerance for each equation, which turns to be useful for problems with solutions having differently scaled components. Moreover, each tolerance is split into two components: one (*absolute* tolerance) is constant while the other one is obtained by multiplying the current value of the appropriate component of vector y by the so called *relative* tolerance. The relative tolerance is useful in problems where variables change order of magnitude during the simulation.

The version of DASSL we actually used in the present work is DASSLRT, a version endowed with a root finding algorithm. This means that the program is able to return the time when a component of a user supplied function of the form g(t,y) vanishes. This feature is effective to deal with systems whose model changes in connection with some events.

4. A DASSL BASED ROBOTICS SIMULATION ENVIRONMENT

4.1 General concepts

The adoption of DASSLRT as the core of a robot simulation environment is motivated by at least three considerations: first, mechanical models often take the form of DAE systems and, moreover, models which derive from aggregation of submodels describing plant components are readily given as DAE systems; second, thanks to the variable integration step, DASSLRT is particularly accurate in the detection of input steps, which can typically come from digital controllers or simply from external events; finally, the root finding algorithm allows great accuracy in simulating discontinuous phenomena where the discontinuity is related to state variables, so that it is not possible to determine *a priori* the time when the event takes place.

The simulation environment is capable to deal with:

1) DAE systems of arbitrary order;
2) systems whose model can take different structures depending on the values of some state variables;
3) discontinuities in state variables (impulsive behavior);
4) interface of the numerical solver to digital control algorithms.

4.2 Steady state computation.

For the computation of a steady state condition for the given model, the user can choose a set of variables to be considered as known and let the program compute a number of variables equal to the number of system equations. It is worth remarking that there is no constraint on the type of the variables (inputs, states, outputs): for a manipulator model, typically the program computes the torques that keep the system in a certain steady state.

4.3 Interface to digital controllers.

For the simulation of hybrid (mixed continuous-time and discrete-time) systems, the user has to specify the sampling time and the connections to plant variables of each digital controller, as well as to write the control algorithm. The integration of the continuous time system is suspended at each sampling instant and control is transferred to the routines of the digital controllers, which modify the inputs of the continuous time system. The adoption of a variable integration step solver, as DASSL is, when simulating hybrid control systems, seems to require that the integration step be smaller than the smallest sampling interval of all the digital controllers. This proves to be frequently inefficient since much of the computation time savings achieved with the use of DASSL relies on the ability of the solver to use large integration steps when the system evolves slowly. As a matter of fact, this possible limitation has been overcome thanks to the adoption of the following solution: when the integration step becomes larger than the sampling interval of a controller, the value of its input variable is computed using a linear extrapolation based on the last computation of the variable and possibly of its derivative. This mechanism proved to be efficient and reliable: especially when the tolerances adopted are not too severe there is a dramatic reduction of the number of steps used to carry out a simulation.

4.4 Simulation of impulsive dynamics.

Impulsive behavior, with discontinuities in the state variables, is handled by means of the root function g(t,y) of DASSLRT, to accurately detect the instant when the discontinuous phenomenon takes place. When such an event occurs, DASSLRT returns the control to the main program, which modifies the value of the variable(s) involved in the event and restarts the simulation.

4.5 Simulation of other discontinuities

Models of systems undergoing structural changes at particular states, called *conditional* models, are formulated by means of two or more sets of equations, each one associated to a validity region in the state space. This means that a particular set of equations can only be used if some inequalities are verified. At the beginning of the simulation, the user specifies which set of equations should be used and consistency with the validity conditions is checked. As the passage from one validity region to another one is detected, i.e. when a component of g(t,y) changes its sign, a new set of equations is adopted: should the particular equation whose structure is to be changed assume more than two forms, a priority list can be supplied, which tells the program what form of the model must be taken into consideration first. Then a check is made on the consistency of the new model by evaluating the system with the new equations and by verifying that the conditions for the new equations to be applied hold true. If this occurs, the simulation is restarted, otherwise another set of equations is considered. This mechanism ensures both accuracy of the simulation and consistency, since the model equations are changed exactly at the instant when the conditions for the change to take place become true.

5. SIMULATION RESULTS

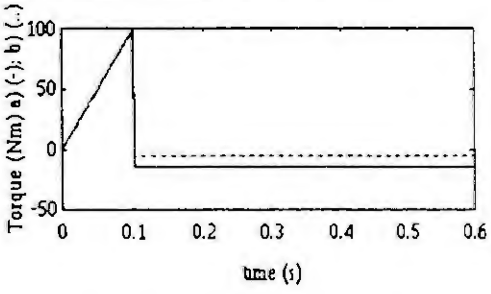


Fig. 4 Torque input in cases a) and b)

5.1. Joint with friction

The model of the joint affected by friction, described in Section 2.1, has been implemented in the simulation environment, exploiting the *conditional* model feature. While the first equation of the model always states that the derivative of the velocity is the acceleration of the motor, the second equation can actually take three different forms, depending on whether the motor is at rest, or in motion with positive velocity or in motion with negative velocity. For the simulation reported in this paper, the rotor inertia was set to 20 *Kgm*^{2}, no viscous friction was taken into account, while the values of the kinetic friction torque τ_C and of the static friction torque τ_S were set to 10 *Nm* and 20 *Nm*, respectively. To compare a case of velocity reversal to a case of transition from motion to rest, two simulations were performed, with the same physical data but with slightly different input torque τ_m (the load torque was zero). Both the simulations begin with the joint at rest and with a ramp of 100 *Nm* in 0.1 *s* on τ_m, which is followed by a 115 *Nm* step in the opposite direction, in case a), and by a 105 *Nm* step in case b) (see Fig. 4). After the initial transition from rest to motion (note, in Fig. 5, the small step in the acceleration due to the difference $\tau_S - \tau_C$), the velocity of the motor (Fig. 6) first increases, then decreases because of the decelerating effect of the torque step, until it vanishes. When the simulator detects this event by means of a DASSL root function, it changes the model, adopting the equations of motion with negative velocity. Then a set of values for the system with the new equations is computed and consistency is checked with the validity conditions associated to the new model. Since for the new equations the consistency is ensured if the acceleration does not change sign, see Section 2.1, the new model is accepted in case a) and the simulation is

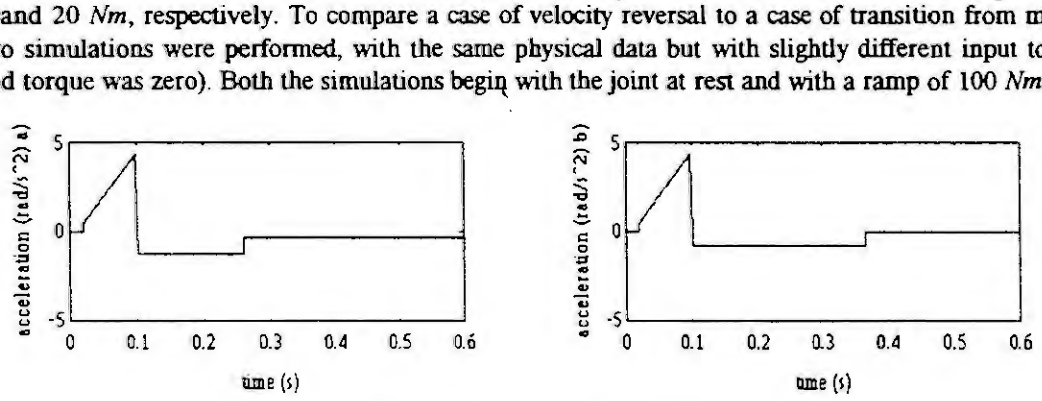


Fig. 5 Acceleration in cases a) and b)

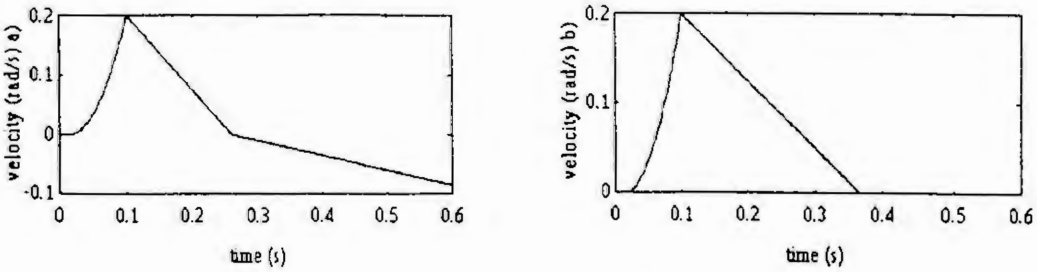


Fig. 6 Velocity in cases a) and b)

restarted, while in case b), where the acceleration would actually change sign, the model is refused. Then, in case b), the simulator checks the third model, namely the model of joint at rest, by imposing a null value for the acceleration. Since the condition for consistency of this new case (applied torque less than the static friction torque) is satisfied, the model of joint at rest is assumed, and the simulation is restarted.

5.2 Double pendulum

To show the behavior of the mechanical system of Fig. 3 for very small $|\varepsilon|$, the limit case a) has been compared with the case d) $\varepsilon = -10^{-6}$. Case a) has been simulated using DASSLRT, alternatively switching between S.1 and S.2 (starting from S.1 in the initial configuration) when the singular configuration is detected (see Fig. 7), while case d) has been simulated solving the constraint $\phi(\mathbf{q})$ for variable q_1:

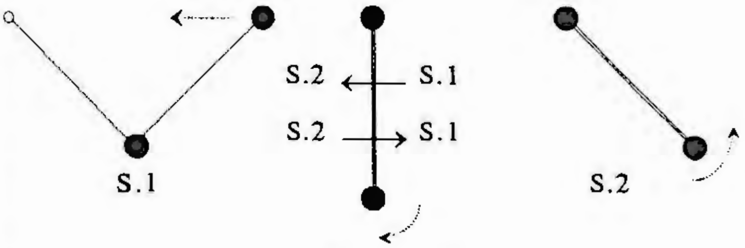


Fig. 7 Simulation case

$$q_1 = \tan^{-1}\left(\frac{1 + (1+\varepsilon)\cos(q_2)}{(1+\varepsilon)\sin(q_2)}\right),$$

and using the Adams integration routine provided by the package SIMULINK [12]. Defining with $e_{iad} = q_{ia} - q_{id}$ the difference between the joint angle q_i ($i = 1,2$) computed in the cases a) and d), Fig. 8 depicts e_{1ad} and e_{2ad}. The analysis of these figures confirms the asymptotic behavior of the system as $|\varepsilon| \to 0$: the differences between the joint trajectories, computed in the cases a) and d) are very small in the first 10 s and maintain still limited after 50 s. The reliability of the simulations, from the point of view of the mechanical energy conservation, is confirmed by Fig. 9 where the total mechanical energy, computed in the cases a) and d) respectively, is shown. Note that there are larger energy errors in the case d) near the singular configuration, due the fact that, since the motion of the system almost has a finite bump, the joint accelerations tend to infinity, causing numerical problems to the solver; the slow and limited trends can be considered negligible and due entirely to numerical round off. Finally it must be pointed out that the computation time in case a) took few tens of seconds while in case d) was more than 20 minutes (with integration routines different from Adams the simulation failed).

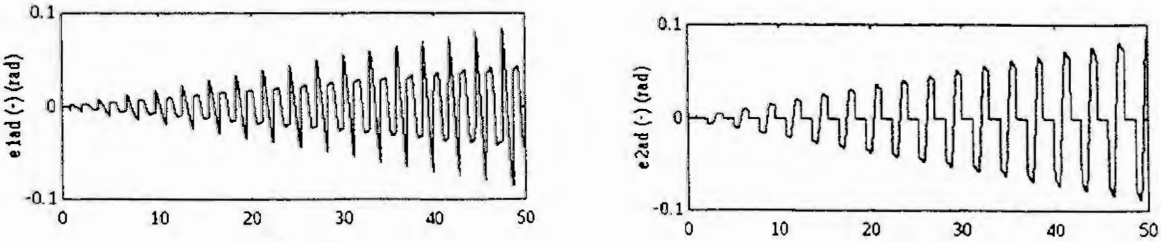


Fig. 8 Differences between joint coordinates in cases a) and d)

6. CONCLUDING REMARKS

Simulation techniques, based on the DAE solver DASSLRT, conceived for the simulation of discontinuous mechanical phenomena and hybrid systems, have been presented in this paper. Our approach allows accurate detection and handling of impulsive events affecting the motion of mechanical systems (collisions and kinematic singularities) as well as the implementation of dynamic models varying their structure at particular states (as in the case of joint friction). Interaction between the variable-step solver and sampled data systems is also tackled efficiently.

Based on these methods, an advanced object-oriented software environment is being developed, allowing automatic generation of efficient simulation code from declarative model description [13].

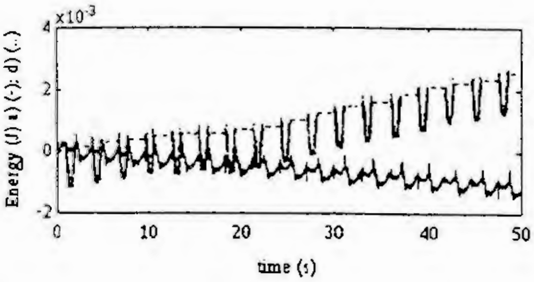


Fig. 9 Total mechanical energies in cases a) and d)

7. ACKNOWLEDGEMENTS

This work was done with support of "Progetto Finalizzato Robotica" of CNR (Italian National Research Board) under contracts N° 91.01933.67 and N° 91.01995.67.

8. REFERENCES

[1] Burdick , J.W., An Algorithm for Generation of of Efficient Manipulator Dynamic Equations. Proc. IEEE Int. Conf. on Robotics and Automation, San Francisco, 1986.

[2] Khalil, W., SYMORO: Systeme Pour la Modelisation des Robots. ENMS-LAN, Nantes, 1989.

[3] Vukobratovic, M., Kircanski, N., Timcenko, A., Kircanski, M., SYM - Program for Computer-Aided Generation of Optimal Symbolic Models of Robot Manipulators. In: W. Schiehlen (Ed.), Multibody Systems Handbook. Springer-Verlag, 1989.

[4] Armstrong-Hélouvry, B., Control of Machines with Friction. Kluwer Academic Publisher, 1991.

[5] Uran, S., Jezernik, K., Troch, I., Coulomb friction and simulation problems. Proc. SYROCO'91, Wien, (1991), 33-38.

[6] Zheng, Y. F., Hemami, H., Mathematical modeling of a robot in collision with its environment. Journal of Robotic Systems, 2, N°3, (1985), 289-307.

[7] Ferretti, G., Maffezzoni, C., Magnani, G., Dynamic Simulation of Robots Interacting with Stiff Contact Surfaces. Transactions of the Society for Computer Simulation, 9, N°1, (1992), 1-24.

[8] McClamroch, N. H., Wang, D., Feedback stabilization and tracking of constrained robots. IEEE Transaction on Automatic Controls, 33, (1988), 419-426.

[9] Haug, E. J., Computer-Aided Kinematics and Dynamics of Mechanical Systems. Allyn and Bacon, 1989.

[10] Ellis, R.E., Ricker, S.L., Two Numerical Issues in Simulating Constrained Dynamics. Proc. IEEE Int. Conf. on Robotics and Automation, Nice, France, (1992) 312-318.

[11] Brenan, K.E., Campbell, S.L., Petzold, L.R., Numerical solution of Initial-Value Problems in Differential-Algebraic Equations. Elsevier Science Publishing, 1989.

[12] The MathWorks, SIMULINK User's Guide, 1992.

[13] Bellasio, F., Benvenuti, A., Lluka, P., Groppelli, G., Maffezzoni, C., A modular simulation environment based on object-oriented database technology. Proc. European Control Conference, Groningen, (1993), 1568-1574.

Integrated Modeling, Simulation, and Animation of Rigid Arms and Vehicles through Object-Oriented Programming

Paul R. Schmitt   John E. Hogan   Jonathan M. Cameron   Wayne J. Book

Georgia Institute of Technology
The George W. Woodruff School
of Mechanical Engineering
Atlanta, GA 30332
USA

Abstract. This paper describes an object-oriented system, "MBSIM" (MultiBody SIMulator), that models, simulates, and animates the kinematics and dynamics of robotic arms and vehicles. This system creates a three-dimensional graphical environment which can be used as a tool in robotic design and control.

1. INTRODUCTION

A motivation for developing MBSIM arises from an inspection task in the nuclear industry. This task requires vehicles to move safely along narrow aisles to inspect drums of low-level radioactive waste. Safe motion requires sensor-based motion control to avoid collisions with walls and obstacles. Given the difficult nature of the environment involved, a simulation system is necessary for the development and testing of motion planning and control algorithms.

We desired MBSIM to be a convenient and flexible tool for studying the motions of various mechanisms including vehicles, robotic arms, and combinations of the two. Our simulation integrates the kinematics, dynamics, sensors, and graphics of mechanisms into a modular environment. Each mechanism consists of a series of links. The mechanism's kinematics and dynamics are determined from the relative kinematics and mass properties encapsulated in each link. Sensors can be attached onto each link object as desired. These sensors are useful for path generation and control routines. Mechanism graphics are constructed from graphics descriptions associated with each link object.

The types of systems that MBSIM can model involve physical objects such as robots, vehicles, and obstacles, objects that share many characteristics. An object-oriented approach is appropriate for modeling these systems. Common characteristics such as position and geometric shape can be developed once and reused appropriately via object inheritance and inclusion. The resulting software objects reflect the modularity of the world and can be dealt with intuitively. In fact, one of the important benefits of the object-oriented approach is the ability to think of software objects in the same way as we do real objects. MBSIM utilizes object-oriented inheritance to construct new types of links from existing links. To illustrate some benefits of the object-oriented approach, this paper includes a case study that outlines the steps taken as well as the time involved in modeling and simulating a mobile platform.

2. SOFTWARE DESIGN

Our system is implemented using the object-oriented C++ language [4]. C++ is an object-oriented programming language that directly supports inclusion (using user-created objects as data) and inheritance (designing objects which inherit previously developed data structures and code from more basic object definitions).

In MBSIM, each mechanism is constructed as a series of links. Each link incorporates the joint kinematics that connect it to the preceding link. In other words, each link "knows" how to move itself with respect to the previous link. This general joint paradigm allows holonomic and non-holonomic constraints to be treated similarly. This paradigm also facilitates modeling and simulation of the kinematics and dynamics of multibody mechanisms with a wide variety of link types. The links contain graphic objects which describe the physical shape of the link and sensors which provide feedback of the environment.

3. KINEMATICS

Following the joint paradigm, each link contains functions to compute its angular and linear velocity and acceleration relative to the preceding link. These functions are pure virtual functions in the generic link class and therefore must be defined in every specific link sub-class. This ensures a uniform interface to each specific link that can be used to calculate velocities and accelerations of any link. The velocities or accelerations are calculated by propagating the velocity out from the base link to the link of interest. This propagation from link to link requires the uniform interface so that the velocities or accelerations can be calculated for mechanisms composed from any number and types of links.

Each specific link also contains functions which enable velocity or acceleration joint variables to be integrated to yield the mechanism motion. These velocities or accelerations would be commanded from control schemes such as obstacle avoidance and path planning. These control schemes would generally be implemented in the simulation to make use of sensor feedback and dynamic characteristics of the system. The commanded velocity or acceleration may be output to an external file for further analysis or for future replays. If needed, this also enables data from external control schemes to be read and tested with the simulation. These integration functions in each link, one for velocities and one for accelerations, contain equations which relate joint velocity variables to joint position variables and joint acceleration variables to joint velocity variables, respectively. Inside these functions, constraint or auxiliary equations may be used to determine the specific behavior of the link, as in a non-holonomic link. The mechanism calls the integration function of each link in the same manner, whether the link is holonomic or non-holonomic.

4. INVERSE DYNAMICS

The inverse dynamics of the system is calculated using an iterative Newton-Euler dynamic formulation similar to the one found in [2]. The inverse dynamics yields the forces and torques that each link's actuator would need to generate motion based on each link's positions, velocities, and accelerations. This is useful to determine whether the desired velocity or acceleration commands are feasible for a particular actuator. Many control schemes such as those found on a Denning robot produce velocity or acceleration commands rather than forces or torques. The consistent velocity and acceleration interface contained within each link enables only one function at the mechanism level to calculate the dynamics for each link in any mechanism in any configuration. Once again, by embedding the specific characteristics of a link inside the specific link class and using consistent interfaces to each link, general functions can be used to yield desired quantities regardless of the mechanism being studied.

5. SENSOR MODELING

To model a mechanism in an unknown or partially known environment, we require sensor feedback of the surroundings. Our system currently incorporates an idealized rangefinder model. As previously mentioned the link sensor file holds the positions and rotations of each sensor on the link. This file is flexible, allowing the user to place an arbitrary number of sensors in any position or orientation on any link.

When activated, the sensor currently scans through all objects in the environment to determine whether the objects' enclosing spheres lie on the sensor's line of sight. Once this rough reading has been determined, the object's actual geometric characteristics are tested for intersection with the sensor's line of sight. This two-stage method reduces computational overhead.

6. IMPLEMENTATION

The MBSIM hierarchy, as developed above, is implemented into four main parent classes: Mechanism, Link, Sensor, and Graphics_Object. The partial class structure in Fig. 1 shows their inheritance and inclusion relationships. These classes are constructed from script files external to the program. These files are read at run-time so additions and modifications of a wide variety of mechanisms can be made without having to re-compile MBSIM.

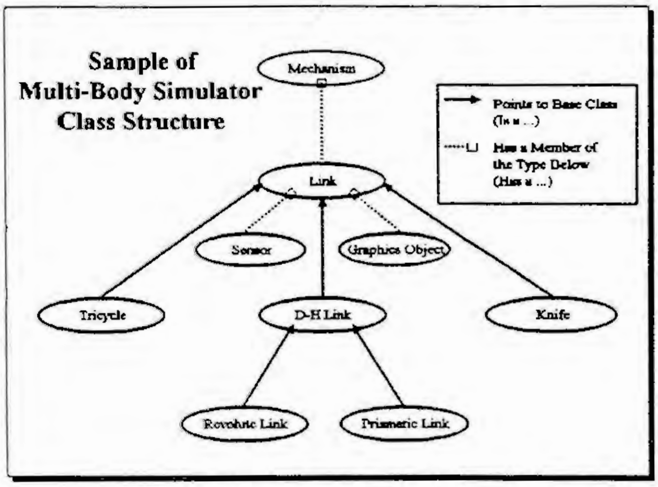


Fig. 1. MBSIM class structure sample.

The Graphics_Object class encapsulates all graphics library function calls. The system initially utilized SPHIGS, a public domain software library for 3-dimensional graphical display [3]. (SPHIGS is a simplified implementation of the well known 3-dimensional graphics protocol, PHIGS.) This encapsulation facilitates improvements to the graphics library.

7. CASE STUDY

A nuclear industry testbed vehicle is driven by two parallel wheels on a fixed axis with independent velocity control. The vehicle, as shown in Fig. 2 is equipped with twenty-four ultrasonic sensors.

The following steps were taken to simulate this vehicle. The manual derivation of the vehicle's kinematic and dynamic link equations required one hour. These equations were incorporated into a new class derived from the parent Link class, requiring a second hour. The vehicle took shape as its dimensions were set into a graphics description file requiring a half hour. Next, sensors were added to the vehicle through a sensor description file. Positioning and orienting the twenty-four sensors took an hour. Another hour was needed to develop a routine to test the accuracy of the above files and code. The total time required to develop this new link and tailor it to a specific vehicle was about four and one half hours.

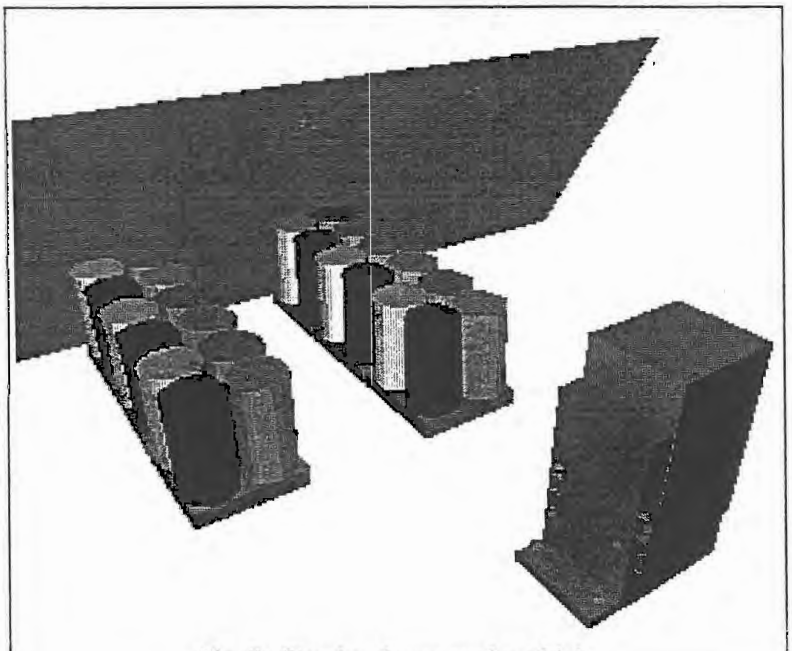


Fig. 2. Simulated case study vehicle

8. CURRENT WORK

Current work is focused on extending the flexibility of MBSIM. These areas include forward dynamics, sensor modeling, sensor data mapping, and graphics library improvements.

Forward dynamics can be determined by putting special values into the iterative Newton-Euler routine to determine the mass matrix and the non-linear Coriolis, centrifugal, and gravity terms. Once found, one can then use these matrices to solve for joint accelerations given joint torques. Velocity and position are then obtained by integrating acceleration.

An ultrasonic sensor is being modeled. Where appropriate, noise will be added to the returned data to more accurately simulate real sensor data.

An intelligent sensor data storage array map is being developed to handle sensor input. Histogramic in-motion mapping will be used mainly to record levels of obstacle existence evidence [1]. The map updates data from new readings and will be used to create and evaluate obstacle avoidance paths.

Currently, the graphics routines limit the performance of our system. This is due to our hardware and the simple nature of the SPHIGS graphics library. While excellent for our initial learning phase, SPHIGS does not have the performance of commercial graphics libraries. We are upgrading our hardware and implementing a more advanced and efficient graphics library.

9. CONCLUSION

MBSIM, an object-oriented three-dimensional robotic simulator, has been introduced. This simulation integrates the kinematics, dynamics, sensors, and graphics of mechanisms into a modular environment. A case study that outlines the steps taken and time involved in modeling and simulating a particular mobile was presented to demonstrate the system's flexibility. Current work will enhance the system's performance, flexibility, and sensing the environment.

10. ACKNOWLEDGMENTS

This work was partially supported by subcontract 92006 to the Educational Research and Development Association (ERDA) of Georgia Universities under contract with Westinghouse Savannah River Plant.

11. REFERENCES

[1] Borenstein, Johann and Koren, Yoram, Histogramic In-Motion Mapping for Mobile Robot Obstacle Avoidance, *IEEE Transactions on Robotics and Automation*, vol. 7, no. 4, August 1991, pp. 535-539.

[2] Craig, John, Introduction to Robotics: Mechanics and Control, 2nd Ed., Addison-Wesley Publishing, 1989.

[3] Foley, van Dam, Feiner, and Hughes, Computer Graphics - Principles and Practice, Addison-Wesley Publishing, 1990.

[4] Stroustrup, Bjarne, The C++ Programming Language, 2nd Edition, Addison-Wesley Publishing, 1991.

Modelling Impacts with Friction Phenomena within the Simulation of Multibody Structures

P. Baiardi
University of Genova
kurgan@dist.dist.unige.it

G. Cannata *
I.A.N. - C.N.R.
cannata@dist.dist.unige.it

G. Casalino
University of Pisa
pino@dist.dist.unige.it

P. Pagano
University of Genova
pagano@dist.dist.unige.it

Abstract

In this paper, suitable models capable of representing impacts with friction phenomena possibly occuring among the bodies of a kinematic chain and the environment, are presented and discussed in some details, with a particular emphasis directed toward their possible use within dynamic simulation environments for advanced robotic structures.

1 Introduction

The present paper introduces and analyses some modelling techniques suitable for representing and simulating impact phenomena in presence of different friction conditions, i.e. static or dynamic, which may occur during the evolution of a complete robotic simulation experiment.

The approach is partially based on previous authors results, presented in [1],[2], concerning the more general problem of modelling the whole set of possible "interactions between bodies and structures", where such denomination has been used for indicating all situations where bodies, belonging to the same or different structures, generally non in contact among them (if not due to the constraint imposed by the fact of being part of a kinematic structure) fall instead into the conditions of necessarily satisfying some additional constraints, as a consequence of "unusual" contact situations that can be occasionally established among them.

As already mentioned, within this paper only collision phenomena will be taken into consideration, with the goal of detailing all a class of models suitable to be efficiently used within more general dynamic simulation environments, for robotic applications [5].

For sake of simplicity, but without loss of generality, the analysis will be however restricted to the case of a robotic structure colliding with a motionless surface. As described in [4], extensions to the more general case of collisions between moving structures can be easily obtained on the basis of the here developed theory.

2 Impacts with friction

Let us start by considering a rigid body, possibly belonging to a kinematic chain, that at time $t = 0$ is colliding with a motionless and rigid surface S, as depicted in figure 1, with impact velocity $\mathbf{v}(0) \triangleq \mathbf{v}^-$

[*] Naval Automation Institute - Italian National Research Council

at the contact point (from here on, index "−" will denote the evaluation of a generic quantity at time $t = 0$).

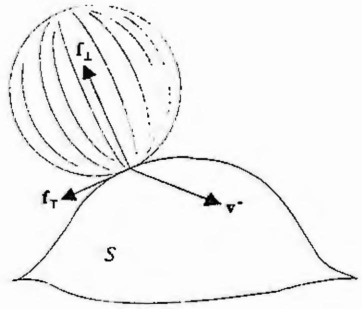


Figure 1

Naturally enough, we assume that immediately before the instant $t = 0$ the so called non contact condition (see [1])

$$\mathbf{n}^T(\mathbf{X}_1 - \mathbf{X}_2) < 0$$

holds, turning into the well known contact one

$$\mathbf{n}^T(\mathbf{X}_1 - \mathbf{X}_2) = 0$$

in correspondence of $t = 0$ ($\mathbf{X}_1$, $\mathbf{X}_2$, $\mathbf{n}$ are the minimum distance points and the common normal between the colliding object and surface, computed as described in [6]).

Moreover we assume that interactions between the surface and the object will be characterized by friction coefficients μ (static) and ρ (kinetic), with $\rho \leq \mu$.

By defining *normal* and *tangential* components of a generic vector $\mathbf{u}$ as

$$\mathbf{u}_\perp \triangleq (\mathbf{n}^T\mathbf{u})\mathbf{n} \; ; \; \mathbf{u}_T \triangleq \mathbf{u} - \mathbf{u}_\perp \tag{1}$$

we have

$$\mathbf{v}_\perp^- \triangleq (\mathbf{n}^T\mathbf{v}^-)\mathbf{n} \; ; \; \mathbf{v}_T^- \triangleq \mathbf{v}^- - \mathbf{v}_\perp^- \tag{2}$$

Then assume that such vectors satisfy the following inequalities

$$|\mathbf{v}_\perp^-| > 0 \; ; \; |\mathbf{v}_T^-| > 0 \tag{3}$$

where the former represents the necessary condition for the impact to take place, while the latter only constitutes a possible one (actually the impact could also occur with null tangential velocity as we shall later discuss). Throughout the following, an *infinitesimal* but *not* instantaneous duration is assumed for the impact, and the corresponding analysis will be carried out within such "microscopic" time interval.

Since the first condition in (3) holds, from a qualitative point of view we can interpret the action of the body as beginning to compress the surface, which in turn *slightly deforms itself* and reacts toward the body with a force $\mathbf{f}_\perp$ acting in the opposite direction of the unit vector $\mathbf{n}$. Meanwhile, due to the second condition in (3), the body starts to *creep* also subjected to the friction force

$$\mathbf{f}_T \triangleq -\rho|\mathbf{f}_\perp|\mathbf{s} \; ; \; \mathbf{s} \triangleq \frac{\mathbf{v}_T}{|\mathbf{v}_T|} \tag{4}$$

Note that whenever a tangential stop ($|\mathbf{v}_T| = 0$) occurs within the "microscopic" impact duration, then $\mathbf{f}_T$, as given by (4), must be instantaneously changed into another suitable force (still tangential) capable to preserve the condition $|\mathbf{v}_T| = 0$, and this until such new force $\mathbf{f}_T$ will maintain its modulus within the bound established by the well known static friction constraint

$$|\mathbf{f}_T| \leq \mu|\mathbf{f}_\perp| \tag{5}$$

Moreover, in case of a successive invalidation of (5), $\mathbf{f}_T$ must be replaced by its original form (4); and so on in correspondence of any further stopping situation or invalidation of static friction condition.

Let us now observe that, by virtue of the slight deformation hypothesis, we can assume that $\mathbf{f}_\perp$ consists of two contributions: the first one, $\mathbf{f}_\perp^{'}$, of elastic nature, while the other, $\mathbf{f}_\perp^{''}$, is of dissipative viscous type.

As a consequence, impact starts with a first phase of *compression*, existing until $\mathbf{v}_\perp$ vanishes at a certain instant t^*, with $\mathbf{f}_\perp^{'}$ acting toward the body. In this phase the modulus of $\mathbf{f}_\perp^{'}$ grows in a monotonic way from its zero value while increasing the elastic potential energy of the surface, that will actually reach its maximum value in correspondence of $t = t^*$, where the modulus of the normal velocity attains its minimum value $|\mathbf{v}_\perp| = 0$. Meanwhile, within the whole compression phase, $\mathbf{f}_\perp^{''}$ continuously dissipates energy. The modulus of $\mathbf{f}_\perp^{''}$ has its maximum value in correspondence of the beginning of the impact (when $|\mathbf{v}_\perp|$ is it also maximum), and gradually vanishes toward zero at $t = t^*$ (where $|\mathbf{v}_\perp| = 0$).

Throughout the same compression phase, another source of energy dissipation is represented by the kinetic friction force $\mathbf{f}_\mathrm{T}$, as given by (4), acting whenever $|\mathbf{v}_\mathrm{T}| > 0$.

In other words, within the collision duration time, a "microscopic" time interval $[0, t^*]$ exists (ranging from the impact beginning to the vanishing of $|\mathbf{v}_\perp|$), which identifies the so called *compression* phase, where the energy is in part transferred from the structure to the surface (stored as elastic energy) and in part lost due to the combined action of the dissipative forces.

A successive phase, the so called *relaxation* phase, ranging within the time interval $[t^*, t_f]$, with t_f denoting the instant at which the end of the impact occurs, will obviously follow the compression one, where the stored elastic energy will be in part returned to the structure and in part lost again due to the same combined action of the dissipative forces. This phase begins when $|\mathbf{v}_\perp|$ vanishes at $t = t^*$, with the force $\mathbf{f}_\perp^{'}$, having the same direction owned within compression (i.e. toward the body), which imposes to the structure a normal velocity, $\mathbf{v}_\perp$ growing in modulus from the zero value, directed away from the surface. For this reason the viscous component of the reaction force $\mathbf{f}_\perp^{''}$, always acting opposite to motion, reverts its direction by assuming now the same orientation of the unit vector $\mathbf{n}$, that is toward the surface. Furthermore, as it happens for the compression phase, dissipative effects exist related with the tangential friction force, when $|\mathbf{v}_\mathrm{T}| > 0$.

2.1 Compression phase analysis

Let us now restart by considering the dynamical equations of the structure within the time interval $[0, t^*]$, generally given by

$$\mathbf{A}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{M} + \mathbf{J}^T\mathbf{f}_\perp + \mathbf{J}^T\mathbf{f}_\mathrm{T} \tag{6}$$

Then consider the sub-interval $[0, \bar{t}) \subset [0, t^*]$ where $|\mathbf{v}_\mathrm{T}| > 0$: it can be shown (see [1], [2]) that defining the variable α (*impulse of the normal reaction force*) as

$$\begin{cases} \alpha \triangleq -\displaystyle\int_0^t \mathbf{n}^T\mathbf{f}_\perp \, dt \\ \\ d\alpha = -\mathbf{n}^T\mathbf{f}_\perp \, dt \end{cases} \tag{7}$$

to be used as an independent one in place of time, within such sub-interval the generalized velocities n-ple $\dot{\mathbf{q}}$ can be estimated as

$$\dot{\mathbf{q}}(t) \approx \hat{\dot{\mathbf{q}}}(t) \quad t \in [0, \bar{t}) \tag{8}$$

where $\hat{\dot{\mathbf{q}}}(t)$ satisfies the following differential equation

$$\frac{d\hat{\dot{\mathbf{q}}}}{d\alpha} = [\mathbf{A}\mathbf{J}^T\mathbf{n}]^- - [\mathbf{A}\mathbf{J}^T]^- \rho\hat{\mathbf{s}} \tag{9}$$

with

$$\hat{\mathbf{s}} \triangleq \frac{\hat{\mathbf{v}}}{|\hat{\mathbf{v}}|} \triangleq \frac{[(\mathbf{I} - \mathbf{n}\mathbf{n}^T)\mathbf{J}]^- \hat{\dot{\mathbf{q}}}}{|[(\mathbf{I} - \mathbf{n}\mathbf{n}^T)\mathbf{J}]^- \hat{\dot{\mathbf{q}}}|} \tag{10}$$

and initial condition $\hat{\dot{\mathbf{q}}}(0) = \dot{\mathbf{q}}^-$

Integration of (9) for $\alpha \geq 0$ directly gives $\hat{\dot{\mathbf{q}}}$ as a function of the increasing values taken by the impulse of the normal reaction force.

The above equation must be integrated (see [1]) until a threshold value $\bar{\alpha}$ below which both the following inequalities hold true

$$|\widehat{\mathbf{v}}_{\perp}| = |(\mathbf{n}^T\mathbf{J})^{-}\dot{\widehat{\mathbf{q}}}| \quad > \quad 0 \tag{11}$$

$$|\widehat{\mathbf{v}}_{\mathsf{T}}| = |[(\mathbf{I}-\mathbf{n}\mathbf{n}^T)\mathbf{J}]^{-}\dot{\widehat{\mathbf{q}}}| \quad > \quad 0 \tag{12}$$

In particular, three possibilities may actually occur

- condition (11) becomes zero while (12) still holds
- condition (12) becomes zero while (11) still holds
- both condition become zero at the same time

The first and third condition simply imply the end of compression, after which the relaxation phase starts under kinetic or static friction condition respectively, as described in sub-section (2.2). In this cases we set $\alpha^{*} = \bar{\alpha}$ denoting the value of α at the end of compression.

The second possibility may instead take place whenever, still in compression, an instantaneous tangential stop occurs, which requires a switch toward static friction conditions. Such second possibility will be briefly analyzed in the following of the present subsection.

To this end let us denote with $\bar{t}$ the true time instant, even unknown, at which we assume $|\mathbf{v}_{\mathsf{T}}|$ vanishes, and with $\bar{t}'$, it also unknown, the one in correspondence of which $\alpha = \bar{\alpha}$. Due to the approximation $\mathbf{v} \approx \widehat{\mathbf{v}}$ we can actually also assume $\bar{t} \approx \bar{t}'$, then, from $\bar{t}'$ onward, the dynamic equations governing the colliding structure become (see [1], [2])

$$\left\{ \begin{array}{l} \mathbf{A}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{b}(\mathbf{q},\dot{\mathbf{q}}) = \mathbf{M} - \mathbf{J}^T\mathbf{n}|\mathbf{f}_{\perp}| - \mathbf{J}^T\mathbf{H}^T\lambda \\ \mathbf{H}\mathbf{J}\dot{\mathbf{q}} = \mathbf{0} \end{array} \right. \tag{13}$$

with $\mathbf{H}\mathbf{J}\dot{\mathbf{q}} = \mathbf{0}$ representing the null tangential velocity (rolling-friction) constraint and the two dimensional vector λ representing the tangential static friction force required for the fulfillment of the previous constraint.

Moreover, by defining the new variable

$$\left\{ \begin{array}{l} \beta \triangleq \displaystyle\int_{\bar{t}'}^{t} \lambda \, dt \\ \\ d\beta = \lambda \, dt \end{array} \right. \tag{14}$$

and using again the variable α as previously done, we can finally get, as shown in [1] and [2], the following relationships

$$\frac{d\dot{\widehat{\mathbf{q}}}}{d\alpha} \quad = \quad -[\mathbf{I} + \mathbf{A}^{-1}\mathbf{J}^T\mathbf{H}^T(\mathbf{H}\mathbf{J}\mathbf{A}^{-1}\mathbf{J}^T\mathbf{H}^T)^{-1}\mathbf{H}\mathbf{J}\mathbf{A}^{-1}\mathbf{J}^T\mathbf{n}]^{-} \tag{15}$$

$$\frac{d\beta}{d\alpha} \quad = \quad [(\mathbf{H}\mathbf{J}\mathbf{A}^{-1}\mathbf{J}^T\mathbf{H}^T)^{-1}]^{-}[\mathbf{H}\mathbf{J}\mathbf{A}^{-1}\mathbf{J}^T\mathbf{n}]^{-} \tag{16}$$

where $\dot{\widehat{\mathbf{q}}}(\alpha)$ immediately follows from (15) as given by

$$\dot{\widehat{\mathbf{q}}}(\alpha) = \dot{\widehat{\mathbf{q}}}(\bar{\alpha}) - [\mathbf{I} + \mathbf{A}^{-1}\mathbf{J}^T\mathbf{H}^T(\mathbf{H}\mathbf{J}\mathbf{A}^{-1}\mathbf{J}^T\mathbf{H}^T)^{-1}\mathbf{H}\mathbf{J}\mathbf{A}^{-1}\mathbf{J}^T\mathbf{n}]^{-}(\alpha - \bar{\alpha}) \tag{17}$$

Naturally enough, expression (17) maintains its validity, for increasing α, only within the fulfillment of the condition for the existence of a static friction situation.

As it is known, such condition is represented by the inequality

$$|\mathbf{f}_{\mathsf{T}}| = |\lambda| \leq \mu|\mathbf{f}_{\perp}|$$

which can be rewritten (keeping into account the definitions (7). (14) for variables α, β) as

$$\left|\frac{d\beta}{dt}\right| \leq \mu\frac{d\alpha}{dt} \tag{18}$$

or also

$$|d\beta| \leq \mu d\alpha$$

which immediately leads to the equivalent condition

$$\left|\frac{d\beta}{d\alpha}\right| \leq \mu \tag{19}$$

Then, by noting from (16) that $\frac{d\beta}{d\alpha}$ actually results in a constant value, we can immediately conclude that, provided (19) is true at $\alpha = \bar{\alpha}$, it will obviously remain true also for all successive values of α, thus implying the validity of (17) for any $\alpha \geq \bar{\alpha}$. In the opposite case, since (19) will be also invalid for $\alpha \geq \bar{\alpha}$, the possibility of occurrence of any static friction situation will be instead prevented from $\bar{\alpha}$ onward. This obviously implies that in such second case we must necessarily renounce to relationship (17) and maintain differential equation (9) (valid for kinematic friction situation) active also for all successive $\alpha \geq \bar{\alpha}$.

In both cases however, (condition (19) valid or not), the independent variable α must be increased (from the value $\bar{\alpha}$) until the attainment of the appropriate value α^* in correspondence of which the equality

$$|\mathbf{v}_\perp| = |(\mathbf{n}^T\mathbf{J})^-\dot{\hat{\mathbf{q}}}| = 0$$

becomes true, thus indicating the approximate ending of the compression phase.

2.2 Relaxation phase analysis

When the modulus $|\mathbf{v}_\perp|$ of the normal velocity vanishes in correspondence of the time instant $t = t^*$, the compression phase ends, while the relaxation one begins.

In our approximate analysis, we can estimate such ending of compression, and beginning of relaxation, with the attainment of the (approximate) condition $|\hat{\mathbf{v}}_\perp| = 0$, occurring in correspondence of the relevant value α^* assumed by the independent variable α. Then by still using the variable α as an independent one ranging from α^* onward (to be used in place of time), and adopting a reasoning line strictly analogous to that proposed within the compression phase analysis, one can easily show that the same kind of approximate equations actually hold also for the relaxation phase.

More specifically, in case the compression phase ends under static friction conditions (validity of (19) at $\alpha = \alpha^*$), we shall continue to use relationship (17) for evaluating $\dot{\hat{\mathbf{q}}}(\alpha)$ also within the relaxation phase, that is also for $\alpha \geq \alpha^*$.

On the contrary, if compression terminates under kinetic friction conditions, we shall extend the validity of differential equation (9) also to the set of values $\alpha \geq \alpha^*$, meanwhile monitoring the possible attainment of condition $|\mathbf{v}_T| = 0$ that, provided (19) also holds, drives the transition toward the eventual static friction situation, still represented by relationship (9).

Naturally enough, while behaving in the above sketched way, the independent variable α must be however increased from the initial value α^* until a suitable one, say α_f, in correspondence of which the value $-\mathbf{n}^T\mathbf{f}_\perp$ of the total normal force is estimated to vanish, thus concluding the persistency of the whole collision phenomenon.

Unfortunately enough, however, due to the fact that throughout the whole analysis the value of the normal force only appears within its time integral α (see definition (7)), assumed as the independent variable, the evaluation of its zero-crossing apparently becomes a problem which is impossible to solve on the sole basis of the here developed approximated theory.

In order to overcome such drawback, it then appear convenient to refer to the widely accepted phenomenological criterion based on the use of the so called "restitution coefficient", which states that the collision phenomenon is assumed to end (i.e. $-\mathbf{n}^T\mathbf{f}_\perp = 0$) in correspondence of a value α_f given by the expression

$$\begin{cases} \alpha_f = (1+\gamma)\alpha^* \\ 0 \leq \gamma \leq 1 \end{cases} \tag{20}$$

where γ just represents the above mentioned restitution coefficient, classifying the collision phenomenon on the basis of the percentage $\gamma\alpha^*$ of the normal impulse α^*, evaluated at the end of compression that the nature of impact can return at the end of relaxation phase.

As it is well known, for $\gamma = 0$ we have the so called anelastic impacts, while for $\gamma = 1$ we have the completely elastic ones. For $0 \leq \gamma \leq 1$ we have instead all the intermediate situations.

At this point, by concluding the present section, we finally observe that, once the collision phenomenon is "solved", the eventually available "output velocities" informations $\hat{\dot{\mathbf{q}}}(\alpha_f)$, $\mathbf{v}_\perp(\alpha_f)$ (with $-\mathbf{n}^T\mathbf{v}_\perp(\alpha_f) \geq 0$, see [1], [2]) and $\mathbf{v}_T(\alpha_f)$ can be used as the new initial conditions for prosecuting the dynamic simulation.

More precisely note that for $\mathbf{v}_\perp(\alpha_f) = 0$ we shall have the simulation prosecution evolving under contact constraints (see [1], [2]), for $\mathbf{v}_\perp(\alpha_f) = 0$ and $\mathbf{v}_T(\alpha_f) = 0$ the constraints for simulation prosecution must instead become those corresponding to static friction contact phenomenon (see again [1], [2]), while for $-\mathbf{n}^T\mathbf{v}_\perp(\alpha_f) > 0$ the simulation must prosecute without any contact constraint.

3 Simulation experiments

Extensive simulations have been performed in order to evaluate and to confirm the correctness of the proposed interaction models, concerning both the collision phenomena and also "permanent contact" conditions, as well as the transitions among different situations.

Satisfactory results have been actually obtained and collected within the work [7], which can be made available to the intersted reader. Notwithstanding this fact, it is however authors opinion that, to the scopes of the present section, an even simple but significant experiment concerning impact phenomena, should be sufficient for pointing out the validity of the proposed models.

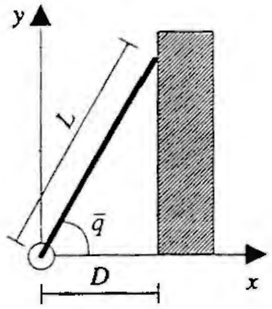


Figure 2

To this aim, let us consider the very simple mechanical structure depicted in figure 2, where a 1-d.o.f. link of mass M and lenght L is made to impact toward a rigid surface located at a distance $D < L$ from the single rotational joint. Depending on the value assumed by the angle $\bar{q} \triangleq \arctan\left(\frac{\sqrt{L^2 - D^2}}{D}\right)$, which determines the "impact configuration", different friction conditions turn out to persist during the "microscopic" impact duration, as a consequence of the specific value assigned to the term $\left|\frac{d\beta}{d\alpha}\right|$ by the impact configuration itself.

More specifically, first note that, due to the planar motion conditions, matrix $\mathbf{H}$ and normal $\mathbf{n}$ at the impact point take on the simple forms

$$\mathbf{H} = \begin{bmatrix} 0 & 1 \end{bmatrix} \; ; \; \mathbf{n} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{21}$$

respectively, while the jacobian and the inertia matrix at the impact configuration simply become

$$\mathbf{J} = L \begin{bmatrix} -\sin\bar{q} \\ \cos\bar{q} \end{bmatrix} \; ; \; \mathbf{A} = \frac{ML^2}{6} \tag{22}$$

thus leading to the following expression for the term $\left|\frac{d\beta}{d\alpha}\right|$

$$\left|\frac{d\beta}{d\alpha}\right| = \left|\frac{ML^2}{6}\tan\bar{q}\right| = \left|\frac{ML^2}{6D}\sqrt{L^2 - D^2}\right| \tag{23}$$

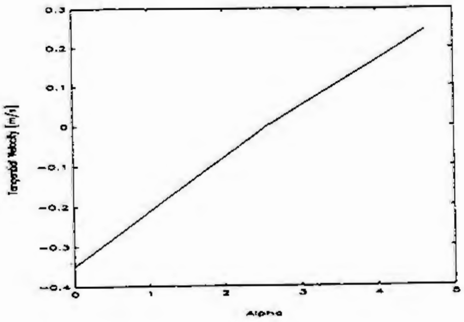


Figure 3

It then follows that the necessary and sufficient condition (19) for the existence of static friction during impact, i.e.

$$\left|\frac{d\beta}{d\alpha}\right| \leq \mu \tag{24}$$

may or not be satisfied, depending on the value of the angle $\overline{q}$ or, equivalently, on the parameters L and D.

As a first example, if the surface is located sufficiently close to the joint, so that the impact angle $\overline{q}$ takes on a large value, not allowing the fulfillment of (24), the collision fully evolves under dynamic friction conditions, where the tangential velocity $\mathbf{v}_T(\alpha)$ at the contact point becomes zero only "instantaneously" within the impact (see figure 3). Also note that for the chosen dynamic friction coefficient $\rho < \mu$, the link is also allowed to bounce back at the end of the impact, as it clearly also apperas from the diagram of the corresponding normal velocity $\mathbf{v}_\perp(\alpha)$ shown in figure 4.

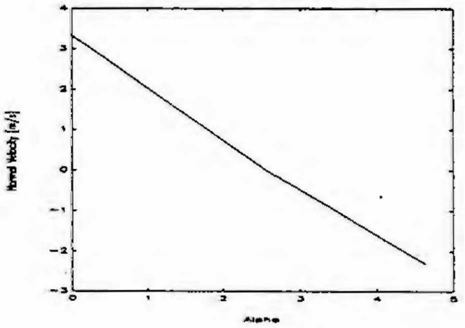


Figure 4

In the opposite case, that is when the surface is moved sufficiently far such that the impact angle $\overline{q}$ takes on a small value. When the collision ends, the link remains stopped at the contact point, thus evidencing what can be interpreted as a "seizing" effect for the specific case.

This is a direct consequence of the fulfillment of condition (24) for the considered impact configuration. The fulfillment of (24) also joined with the fact that the system is a single d.o.f. planar one, naturally implies that, from the end of compression onward, the total velocity must remain permanently zero (figures 5 and 6).

Such obtained simulation results are actually in complete accordance with every day common experience, as well as with real system experiments. However, as it has been pointed out also in [3], similar effects could not be experienced on the basis of previously existing, more simplified, impact models

This also better clarifies the importance attributed by the authors to the presented examples that, notwithstanding their very simple nature, are however able to bring into evidence the contribution given by the proposed impact models.

Conclusions

In this paper, suitable models capable of representing impacts with friction phenomena possibly occuring among the bodies of a kinematic chain and the environment, were presented and discussed in some details.

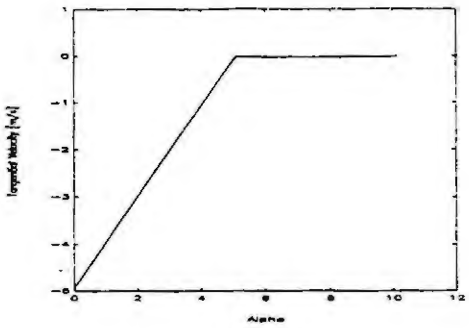


Figure 5

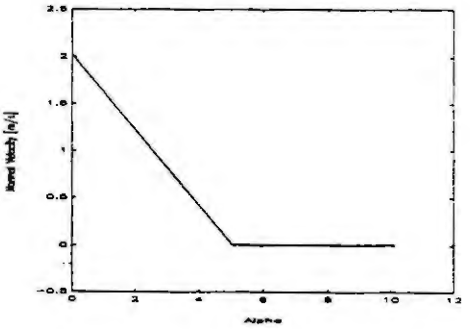


Figure 6

with a particular emphasis directed toward their possible use within dynamic simulation environments for advanced robotic structures.

Acknowledgements

This work has been supported by ESPRIT BRA project "SECOND" and C.N.R. Special Project on Robotics

References

[1] P. Baiardi G. Cannata G. Casalino P. Pagano *Modelling Contact Phenomena within the Simulation of Advanced Robotic Structures*, I.E.E.E. Int. Conf. on Robotics & Automation, Atlanta 1993.

[2] P. Baiardi G. Cannata G. Casalino P. Pagano *On the Modelling of Contact Phenomena*, *DIST* Internal Report, University of Genova, 1992.

[3] J. B. Keller *Impact with friction*, Jour. of Applied Mechanics, Vol. 53, March 1986.

[4] P. Baiardi, G. Cannata, G. Casalino, P. Pagano *Modelling Simultaneous Collisions between Moving Kinematic Chains*. *DIST* Internal Report, University of Genova, 1993.

[5] P. Baiardi, G. Cannata, G. Casalino *Methodologies and Tools for the Dynamic Simulation of Complex Mechanical Structures*. Automazione e Strumentazione, May 1992 (In Italian).

[6] P. Baiardi P. Pagano *On the Modelling of Solids with Extended Superquadrics: Collision Control*, LIRA-Lab Report, University of Genova, 1993

[7] U. Affaticati S. Bonati *Analysis and Modelling of Contact Phenomena within Dynamic Simulation of Robotic Structures*, Doctorate Thesis, 1993

INTEGRATING PLANNING AND KNOWLEDGE REVISION

Aldo Franco Dragoni
Università di Ancona, Istituto di Informatica
via Brecce Bianche 60131 Ancona, Italy
e-mail dragon@anvax2.cineca.it

Abstract We have focused our research on the possible interactions between two classical paradigms of Artificial Intelligence, Planning and Knowledge Revision, both in a single and in a multi-planner domain. We've examined various thinkable ways in which plan formation, execution, monitoring and replanning could take advantage from Truth Maintenance techniques. The creation of a plan is based on a set of assumptions about the external environment and task goals. The inconsistency of these sets or the falsehood of some among their elements can cause failures both in plan generation or execution. On failure it becomes important to understand which assumptions were responsible for the flaw in order to get a more updated knowledge base or a consistent goal. Consistency restoration, multisensor information integration and updating knowledge bases, all need knowledge revision techniques.

1. INTRODUCTION

Given a first order language *L*, we describe the state of the world by means of a set *S* of ground sentences of *L*. Let *R* be a definite set of sentences of *L* representing the causal theory of the world. We define $D=S \cup R$ a description of the world. We refer to the following characterization of Planner and of Reviser.

The Reviser (Fig 1) is typically a knowledge revision system [4]. It is an automaton that accepts as input an eventually inconsistent description *D* of the world and gives as output a maximally consistent subset of it. We give to the term "Inconsistency" a broad and indefinite meaning. It could mean strictly "logical inconsistency" (i.e. the description has no models), but in this case we would have troubles to check for it in a full first order description. It could also mean simply that, for whatever arbitrary reasons, the sentences in the set ("nogood sets" in [2]) can't stand together. What is important to use an ATMS is that if a set of sentences is inconsistent, then every superset of it has to be inconsistent too. A *context* is a subset of *D* that is *consistent* (i.e. it has no inconsistent subsets) and *maximal* (it becomes inconsistent if augmented with whatsoever else assumption in *D*). The ATMS takes as input *D* and gives as output the set of all the contexts of *D*. The task of the Chooser is that of selecting one of the contexts supplied by the ATMS as the preferred context. The Chooser is a plausibility/preferability meta-function; it is an opportune domain dependent algorithm that adopts some arbitrary domain dependent selecting criteria. So, the overall automaton takes as input an eventually inconsistent description of the world and gives as output a consistent and anyhow preferable subset of that description.

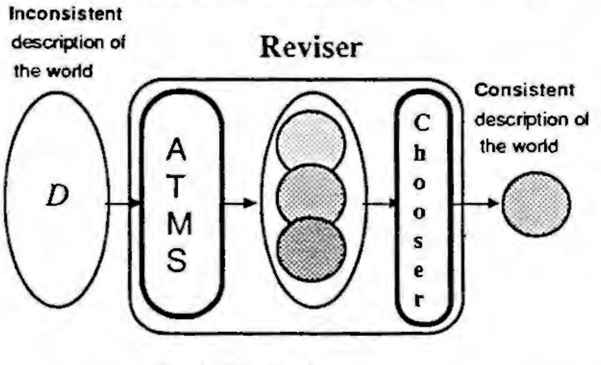


Fig 1. The Reviser

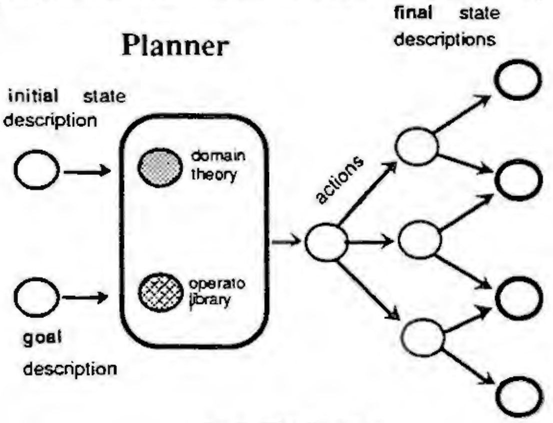


Fig 2. The Planner

We adopt the simple STRIPS-like definition for actions [5]. They are atomic means to modify the state of the world. An action A changes the state of the world from S to A(S). A goal statement G is a set of sentences of L. A planner (Fig 2) is an automaton that takes as input a description *D* of the world and a goal statement G, and gives as output a plan, that is an acyclic labelled graph of states. The arcs are labelled with action instances. Each leaves of the graph is a final description SF_i of the world in which the goal statement is satisfied. That is, for each complete ordered action sequence in the plan $A_1..A_n$ it holds

$$A_n(A_{n-1}(..(A_1(D))..))\models G$$

2. INTEGRATING A REVISER WITH A SINGLE PLANNER

We begin to study the integration of an ATMS with a single Planner. We distinguish the use of an ATMS during the plan creation from its use during the plan execution.

2.1 Use of a Reviser during the plan creation

If the initial situation is inconsistent, then the final situations may be inconsistent too. If the situation is *logically* inconsistent the problems are more serious; in this case every sentence of L is a logical consequence of the initial situation. A classical planner, before planning an action, checks for the action's preconditions to be verified in the current situation. Suppose that the method used by the planner to verify a sentence is complete, that is, if a sentence is a logical consequence of a situation *D* then the planner is able to prove it. If the situation is logically inconsistent then such a planner will verify every precondition of every operator. It will be justified the planning of every action and the whole planning process will be vanished. In such a scenario it is justified the use of a Reviser to check and eventually resolve in input to the planner the inconsistencies of the initial situation. The resulting initial situation will be a consistent maximal subset of the previous one. In practice, the occurrence of an inconsistent initial situation is not unlikely, especially in robot planning domain, where the information about the environment come from a multisensor apparatus. In a multi-agent domain, the initial situation is also built upon information coming from the various other agents in the world. This information sources' multiplicity is being considered as the main cause of inconsistency.

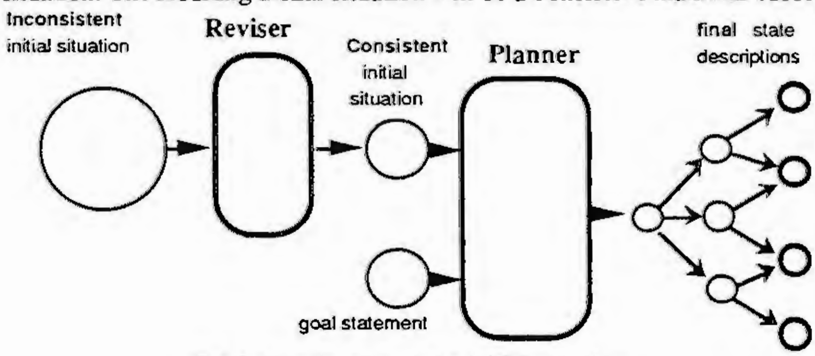


Fig 3. Consistency of the initial situation

Another cause for the inconsistency of the initial situation is the dinamicity of the world. If the world changes during the time needed to the system to sinthesize the plan, then the new information coming from the sensors may be inconsistent with the previous ones. Before starting the execution of the plan it will be necessary to recheck for and eventually to restore the consistency. It will mean to retain as much of the previously held knowledge is consistent with the new information about the world. Once detected the contradictions and calculated the set of contexts, we need a good Chooser to select the preferred context. In [3] we've examined a Chooser for knowledge revision in a multi-agent environment, presenting specific algorithms and criteria. Here I report simple criteria to judge the force of the individual assumptions.

1. Assumptions derived from the sensor's perception are stronger then the others.
2. The multiplicity of the sources confirms the assumption.
3. The more the conflicts with other assumptions, the weaker the assumption.
4. The less reliable the agent/sensor who gave an information, the weaker the assumption derived from it. We could estimate the agent/sensor's reliability by:

- Self-Inconsistency (he gave information mutually inconsistent)
- Average of Inconsistency of the assumptions derived from information received from that agent/sensor with respect of all the other sensor observations.

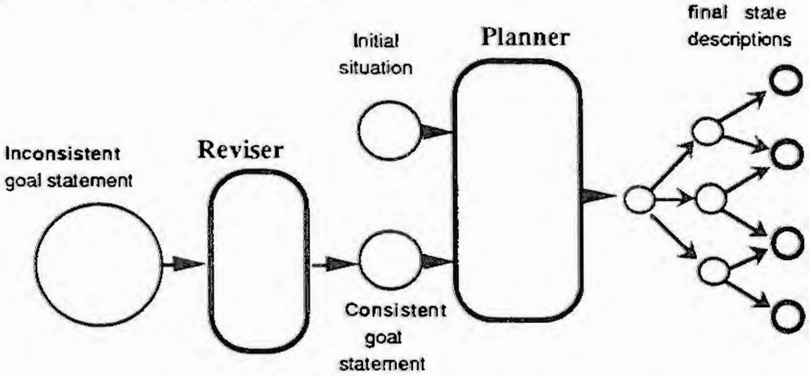


Fig 4. Consistency of the goal.

Now, let G be a goal statement. If G is inconsistent then every final situation verifying it shall be inconsistent too. If the initial situation is consistent and all the operator available to the planner are sound then we can't obtain an inconsistent final situation, therefore there not exists a plan to reach that goal statement. However, this seems to be an acceptable behaviour because it would be strange the planner's being able to sinthesize plans for goals that do not have a model. Goal inconsistency can appear, for instance, in multi-agent cooperative domain, when an agent asked for collaboration adopts the goals of more then one other agent at the same time.

2.2 Use of a Reviser during the plan execution

During the execution of a plan, a robot could find the state of world different from the one expected. May be the expected initial situation was different from the real world's one. May be the state of the world is changed because of casual or unforeseen events. Most of the differences and of the changes cause inconsistencies with the previous expected or unchanged situation. The general *frame problem* is that of specifying what doesn't change when an event occurs [6]. In our implementations we have addressed this problem simply retaining from the previous situation as much knowledge as possible that is consistent with the new information. This task has been accomplished by a Reviser. We've added a special important criterion to those presented in the previous section, that is simply: new coming information from an agent/sensor are always the strongest assumptions. In this way the preferred context will always contain the latest information about the world and as much knowledge, from the previously held one, is consistent with them and preferable according to the other criteria presented.

May be the changes don't affect the plan execution. The "interesting" case is that in which the removing of inconsistencies causes the invalidity of the preconditions of a subsequent action in the plan. First of all, we can't be sure that the action will not be performable because of the fact that intermediate actions can restore the validity of its preconditions. However, there are various possible strategies. At least two extremely positions are worth notice: 1) the planner could stop the execution and replan from the actual situation (the situation in which the robot has perceived the change), 2) the planner could continue the execution until the unperformable action is encountered and then replan from that new situation (the situation it has reached).)Both strategies have their rationales and pitfalls.If it is impossible to sinthetize any other plan, then it could be considered the following idea. Although often events are irrevocables, sometime it could be useful to start a diagnostic process to search for an abductive explanation for the occurrence of the impeding event. If successful, this process may start a planning to remove the causes of the obstacle. The ATMS can be used as the basis for a diagnostic procedure.

3 INTEGRATING A REVISER WITH MORE THAN ONE PLANNER

In distributed planning [1] a single plan is produced and executed by the cooperation of several planners. Each planner produces a subplan, but there may be conflicts among subplans that need to be reconciled. In general, both during creation and during execution Planners work in parallel. To tell the truth, many arguments in this area are still not well understood. Again, we distinguish the use of a Reviser during the plan creation from its use during the plan execution.

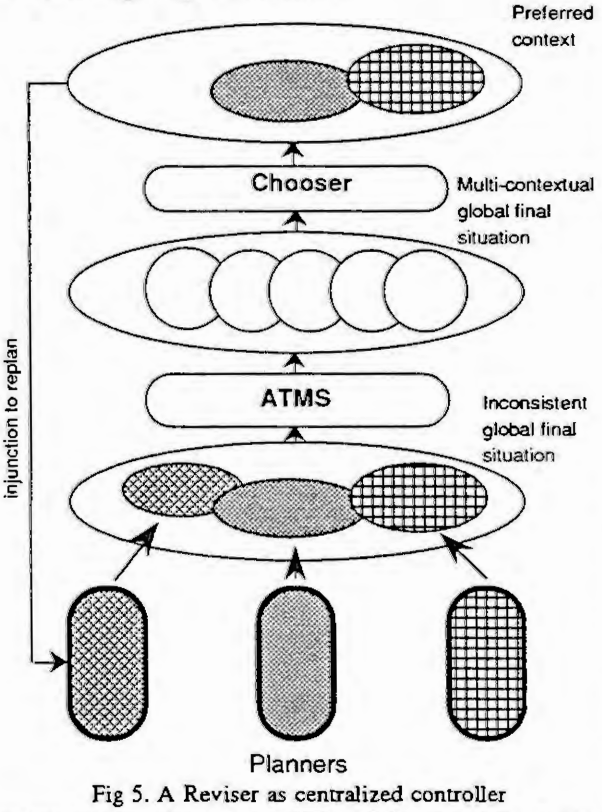

Preferred context
Chooser
Multi-contextual global final situation
ATMS
Inconsistent global final situation
injunction to replan
Planners

Fig 5. A Reviser as centralized controller

3.1 A Reviser during the subplans creation

Given a goals conjunction, each planner develops a plan for a subgoal. The distribution of the subgoals is accomplished out of the system. May be each planner is specialized on a class of problems, may be they work at different levels of abstraction. As usual, we must check for interactions between the various subplans. These interactions may be "positive" (a subplan is part of another one), but here we are interested in checking and, possibly, resolving "negative" interactions (a subplan is incompatible with another one). We are going to study the use of a Reviser to check and eventually resolve these incompatibilities. Consider the following scheme. There are *n* planners working on separate goals. Each one produces its plan and then gives the final situation in which the world would be if its plan were the only to be executed. We define *global final situation* the union of each final situation. We make the fundamental assumption that there aren't synergies among the plans. It means that there is a new ground sentence in the global final situation if and only if one planner has introduced it; moreover, a ground sentence disappears from the global final situation if and only if one planner has removed it. Again, the global final situation may be inconsistent. It can happen even if the goals conjunction is consistent. In such a scenario, the Reviser could select a preferable context CFin of the global final situation. At this point, the idea is that of accepting only the plans whose final situation belongs to CFin. The planners whose plan is not accepted must replan taking as initial situation the union of the final situations of the accepted plans, i.e. CFin minus the set of the sentences introduced only by the rejected plans. Obviously, the plans that will substitute those rejected must be executed after the end of the execution of the plans accepted initially. This procedure has to be iterated until all the planners produce an accepted plan. Alternatively, a planner could replan using a dependency directed backtracking strategy. If there are mechanisms to record the dependencies between actions and effects the planner whose plan is rejected may be notified with the earliest action in its plan which introduced the problematic effects, then it could replan from that action forward. These strategies are not necessarily alternative. Dependency directed backtracking may be the main tactic and total replanning may be the drastic strategy. If there is at least one planner that in unable to sinthesize a plan to reach its goal, the only solution is that of backtracking over the choice of the rejected plans. It means that we now accept a plan previously rejected (may be the plan whose planner has been unable to sinthesize another one) and reject some plans (in a minimal number) among them previously accepted.

3.2 A Reviser during the subplans execution

Suppose that all the planners were able to produce their plan and all the final incompatibilities were solved. Suppose that one (or more) of the planners fails during execution because of the fact the world is different from what expected. May be the global initial situation was not correspondent to the real world situation, may be something is changed due to casual or unforeseen events, may be the plans were final compatibles (compatibles from the final situation point of view) but not compatibles in their intermediate steps. Because of the fact that we can't know a priori the actions' duration we can't resolve the intermediate step's incompatibility simply by synchronising the actions. The drastic solution is that of making a preliminary cross compatibility test among every action in every plan and all the other actions of all the other plans. On detecting a cross incompatibility one of the two planners will have to replan (may be only the subsequence of the plan starting from the incriminate action). When the cross compatibility test is successful the execution starts. As said before, the critical point is what happens when one of the planners fails the execution. Having resolved the eventual intermediate incompatibilities, the only cause of the flaw can be an unexpected external event. This situation is similar to that of the single planner. The planner will have to resolve its inconsistencies and then to replan in order to accomplish its task goal. However, replanning in this case is not so simple. The problem is that the global initial situation now is changed because of the fact that the other planners have already started the execution of their plans. This problem doesn't exist if the planners work on separate domains. May be that the different initial situation doesn't affect the new plan, but how can we be sure of this? However, a first solution may be simply to ignore this problem. The planner starts to replan the unexecuted subsequence of actions without taking in count the changes already produced in the world by the other planners. When the planner produces a new subsequence of its plan, it will be necessary to recheck that subsequence for cross compatibility as done before for the total plan. A finer solution may be making the planners able to exchange information about their current description of the world. Having the global plan passed the cross compatibility test, the global current description of the world should be consistent. Once obtained, this global description may become the new initial situation in input to the planner whose plan has been unsuccessful. A third solution tries to eliminate the time consuming cross compatibility test. After the elaboration of a new plan, the planner could ask again the other planners for their current description of the world. Then the planner could start to simulate the execution on the global description obtained as initial situation. If the simulation is successful it has lower risk to fall in incompatibilities with the other planners. This method is not as safe as the cross compatibility test. In all of these cases the main risk is that of falling in a long sequence of replanning stages when the cross incompatibilities are frequent. However, it seems that the use of an ATMS during the execution in a distributed architecture is not much useful!

4. CONCLUSIONS

We've studied four cases of possible integration between the classical Planner's and the ATMS's paradigms.

1. Use of an ATMS during a single planner planning stage; we've justified the employment of a Reviser to preserve the consistency of the initial description of the world and of the task goal.
2. Use of an ATMS during a single planner execution stage; we've found useful a Reviser to restore the consistency of the description of the world, lost because of unexpected changes of the world.
3. Use of an ATMS during a distributed planning stage; we've justified the employment of a Reviser to preserve the final consistency of the distributed plan.
4. Use of an ATMS during a distributed execution stage; we've found the employment of a Reviser not more useful than in the second case.

5. REFERENCES

[1]. Readings in Distributed Artificial Intelligence, A. H. Bond and L. Gasser eds, Morgan Kaufmann Publishers, San Mateo, CA, 1988.

[2]. Johan de Kleer, An Assumption-based TMS, *Artificial Intelligence*, Elsevier Science Publisher B.V., North Holland, 28, 127-162, 1986,

[3]. Aldo Franco Dragoni, A Model for Belief Revision in a Multi-Agent Environment. In E. Werner & Y. Demazeau (Eds.), *Decentralized A. I. 3*. North Holland Elsevier Science Publisher.(1992)

[4]. Joao P. Martins, Stuart C. Shapiro, A Model for Belief Revision, Artificial Intelligence, Elsevier Science Publisher B.V., North Holland, 35, 25-79, 1988.

[5]. N. J. Nilsson: "Principles of Artificial Intelligence" Springer-Verlag 1981

[6]. John McCarthy and P. Hayes, Some Philosophical Problems from the stand point of Artificial Intelligence, in Machine Intelligence 4, pp. 463-502 (1969).

A MODEL ANALYSIS OF CAR DRIVING ON TWO-LANE ROADS

P.H. WEWERINKE
Department of Applied Mathematics
University of Twente
P.O. Box 217, 7500 AE Enschede
The Netherlands

Abstract

In this paper car driving is considered at the level of human tracking and maneuvering in the context of other traffic. A model analysis revealed the most salient features determining driving performance and safety.

Also learning car driving is modelled based on a system theoretical approach and based on a neural network approach.

1 Introduction

Road traffic performance and safety is determined by several aspects, one of which is the car driving behavior of the human operator. This is the subject of this paper.

Car driving is considered in terms of lane keeping and car following or overtaking slower vehicles, avoiding collisions with oncoming cars. These tasks are analysed and modeled in Chapter 2. The result is a relationship between a variety of task and human operator related parameters and measures of safety and average driving speed (traffic performance).

Several model aspects can be related to the driver's experience level. Learning the driving task is discussed in Chapter 3. Two approaches are followed to model learning. The first one is based on system theory. Learning is modeled as an adaptive estimation process of unknown model parameters. The second approach utilizes a neural network to describe adaptively the input-output behavior of learning the driving task (by adjusting the neural network weights).

Apart from the interest in car driving itself, the study is motivated to compare the two approaches and their relative benefit to describe and predict human learning behavior. For this purpose a simulation program is planned.

2 Model analysis of car driving

The overall goal of car driving is to go from A to B in a certain way (safely, in a given time, etc.). The principal tasks derived from this are lane keeping and overtaking slower vehicles, avoiding a collision with oncoming cars, based on visual cues of the outside world.

Lane keeping is based on two primary visual cues: the inclination of (or distance to) the road side and the direction of an aimpoint. The first visual cue provides information about the lateral position y (see Figure 1). The direction of an aimpoint ψ_a at distance d ahead is given by

$$\psi_a = \psi + y/d \tag{1}$$

Equation (1) shows that for large d, $\psi_a \approx \psi$ and for small d $\psi_a \approx y/d$; in other words: depending on the 'looking' distance ahead d, the driving task resembles more a (relatively easy) heading control task or a (relatively difficult) position control task. This can be illustrated by a simple root locus analysis.

Assuming that the driver is generating a steering wheel deflection δ proportional to the system output o (i.e., ψ, y or ψ_a), the closed loop system dynamics can be visualized by the poles of the root loci shown in Figure 2, containing also the coresponding transfer functions $\frac{0}{\delta}(s)$.

The figure shows that good heading control performance can be obtained. Position feedback results in unstable behavior. Thus driver compensation is required (e.g. by means of a heading inner loop). The difference between aimpoint control and position control is the additional lead (zero at $-u/d$, with u the forward driving speed) corresponding to the implicit heading feedback. For large d, the zero effectively cancels one of the free poles, and the task approaches the heading control task. In the following the value of d will be related to the driver's experience level.

The decision to pass a slower preceding car is based on the estimated distances to the preceding and oncoming cars. The passing maneuver requires a given distance between oncoming vehicles X_k, which depends on the distance X_j from the preceding car at the moment of accelerating (determining the driving speed on the opposing lane). This relationship is derived in [1] and given by

$$X_k \doteq S_k + \frac{u_m + u_k}{u_m - u_j} \cdot S_j + T \left(u_j \frac{S_j}{X_j + S_j} + u_k \cdot e^{-\sqrt{\frac{2X_j}{T(u_m - u_j)}}} \right) \tag{2}$$

In Figure 3, most of the parameters of Equation (2) are clarified. S_k is the minimum distance to car k, S_j is the distance which car i is overtaking in the left lane with respect to car j, by accelerating from $u_{i_0} = u_j$ to u_m with a first order time constant T.

At distance X_j from j car i is starting to accelerate. Because of the increased speed it takes a shorter time to overtake car j (to cover the distance S_j). For that reason the resulting required distance between oncoming vehicles X_k is decreasing with increasing X_j. This tradeoff is shown in Figure 4.

Slower cars require (of course) a larger X_k, especially when also their maximum speed u_m is smaller (yielding a larger $X_{k_{min}}$). The latter is not assumed in the Figure. The Figure also reveals that slower cars can obtain a larger reduction in X_k by increasing X_j.

This possibility to tradeoff X_k and X_j allows a car to optimize its overtaking strategy depending on the momentaneous traffic situation. This situation can statistically be specified in terms of two probability density functions of the actual distances between the right-lane and left-lane cars, denoted by p_{a_j} and p_{a_k}, respectively. These determine the available spacing between the cars. This is summarized in Figure 5. X_{k_0} and X_{j_0} represent the minimum car distances and X_{k_1} corresponds with X_{j_0}. Assuming that the optimal overtaking strategy implies that the distribution of X_k coincides with p_{a_k}, an expression for the average of the optimal distance, $\bar{X}_{j_{opt}}$ can be derived (in [1])

$$\bar{X}_{j_{opt}} = \int_{X_{k_{min}}}^{X_{k_1}} g(X_k) \left[\int_{g(X_k)}^{\infty} p_{a_j}(\alpha) d\alpha \right] p_{a_k}(X_k) dX_k \tag{3}$$

Similarly the corresponding average $X_{k_{opt}}$ can be determined, as well as the overall average driving speed, etc. [1]. These measures can be used to assess the effect of a variety of task variables of interest on traffic performance. In addition these measures can be compared with the corresponding measurements of a simulation experiment (discussed in the following).

3 Learning

For many practical questions it is important to operationalize the experience level of the driving task and to have insight in the learning process involved.

In this chapter learning involved in car driving is discussed following two approaches: a system theoretic approach and a neural network approach. The overall objective is to assess the relative benefit of both methods to describe the adaptive characteristics of human control tasks.

3.1 System theoretic approach

The system theoretic approach is based on a model of the system and the task but only partly known. Learning is described as an adaptive estimation process of the task based on new data (experience).

More specifically, it is assumed that for the naive driver the system behavior and the system outputs (visual cues) are partly known. In addition, adaptive control is assumed in terms of varying weightings (tradeoffs) in the performance index which the human operator is assumed to optimize, or directly in terms of feedback control gains. Learning the overtaking maneuver is described as an adaptive estimation process of unknown parameters.

The partly known system model is given by

$$x_{k+1} = A(\theta)x_k + B(\theta)u_k + Ew_k \tag{4}$$

$$y_k = C(\theta)x_k + v_k \tag{5}$$

with x, u, w and y the state, control, disturbance and output vector, respectively. It is assumed that uncertainty about the system can be related to unknown parameters (θ) in the system model. Learning is then modelled as a parameter estimation problem. The procedure to solve this is by adding the unknown parameters to the state vector (using the parameter model $\theta_{k+1} = \theta_k$) yielding an augmented nonlinear system

$$\bar{x}_{k+1} = f(\bar{x}_k, u_k) + \bar{E}w_k \tag{6}$$

$$y_k = h(\bar{x}_k) + v_k \tag{7}$$

with

$$f = \begin{bmatrix} A(\theta)x + B(\theta)u \\ \theta_k \end{bmatrix}; \quad \bar{E} = \begin{bmatrix} E \\ 0 \end{bmatrix}$$

and

$$h = C(\theta)x$$

This can be solved by means of an extended Kalman filter to estimate $\bar{x}$ and thus x and θ.

In addition to estimating the partly unknown system, learning can be related to adaptive control behavior.

Lateral car control amounts to feedback control of heading ψ and lateral position y. Thus

$$\delta = \ell_1\psi + \ell_2 y \tag{8}$$

Therefore, learning the optimal control strategy can be related to learning the optimal values of ℓ_1 and ℓ_2. This can be modelled by adjoining the control to the state vector $\bar{x}$ (of equation (4)) and consider ℓ_1 and ℓ_2 as unknown model parameters. These can be treated as the unknown parameters θ in the system model of equation (4), yielding estimates of ℓ_1 and ℓ_2 by means of the extended Kalman filter.

The optimal overtaking maneuver is based on the functional relationship between X_k and X_j i.e. $X_k = f(X_j)$ as shown in Figure 5. Learning the optimal maneuvering strategy involves the estimation of f. This, again can be considered as an estimation problem of unknown model parameters in f as discussed before.

The adaptive estimation process starts with an initial estimate $\hat{X}_0$. It is a nontrivial question how the prior knowledge of naive car drivers can be translated into $\hat{X}_0$. Experience, in terms of new data y_k of equation (5) results in improved knowledge of the system and a better task performance.

3.2 Neural network approach

Human operator behavior can be described as the relationship between task inputs y and control outputs u (inputs to the system). Learning this functional relationship between y and u can be described by a neural network (NN).

A NN consists of a number of processing elements with weighted connections. The weights represent the memory of the network and reflect the input-output relationship. The NN can have a given structure (e.g. feedforward) and a given learning strategy (e.g. back-propagation) as discussed in [3].

Human operator learning is described in terms of adjusted weights of the NN based on input-output data of real life tasks. The NN assumes no specific structure of the input-output relationship but requires data to be trained. Only when these are available a NN model can be 'built' and used for further analysis of learning, etc.

For the car driving task (see References [1] and [4]) the human operator inputs y consist, for the lateral task, of heading and lateral deviation and, for the overtaking task, of speed and relative distances (to preceding and oncoming cars). The outputs u consist of steering wheel deflection, gass and brakes. In Reference [4] preliminary results are discussed to train a NN for the aforementioned car driving tasks.

4 Concluding remarks

The model analysis of car driving in Chapter 2 revealed the most interesting aspects of both the lane keeping task and the overtaking task. Several characteristics could be related to the driver's experience level.

Learning the driver's task is discussed and modelled in system theoretical terms. Basically, learning is modelled as an adaptive estimation proces of unknown system- and task parameters. A neural network is considered to model human learning of car driving by describing adaptively the input-output relationship. For this purpose input-output date must be available to train the NN by adjusting the NN parameters.

The next step is to simulate car driving learning and to compare both approaches and their capability to describe (predict) human learning behavior in car driving.

References

[1] P.H. Wewerinke, "Modelling and analysis of car driving", *Memorandum*, Dept. of Applied Mathematics, University of Twente, to appear.

[2] A. Bagchi, *Optimal Control of Stochastic Systems*, Prentice Hall, 1993.

[3] P.K. Simpson, "Artificial neural systems", Pergamon Press, 1990.

[4] K.-F. Kraiss and H. Küttelwesch, "Identification and application of neural operator models in a car driving situation", *Proc. 5th IFAC Conference on Man-Machine Systems*, The Hague, 1992.

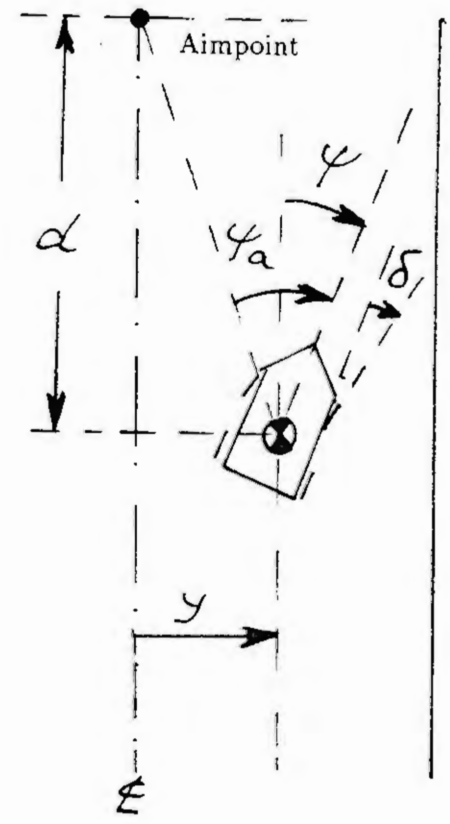


Fig. 1. Lateral driving task situation

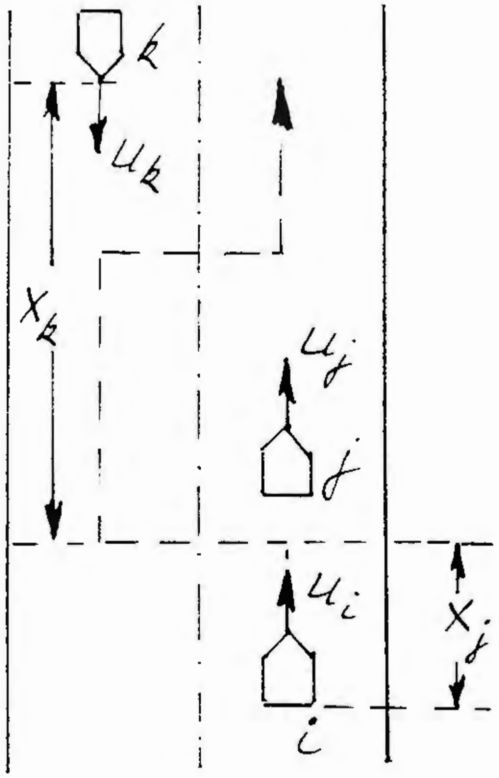
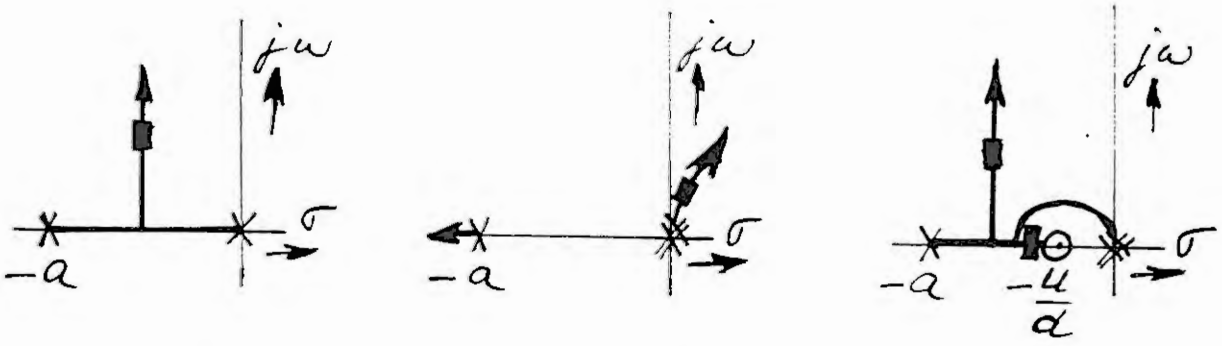


Fig. 3. Driving situation



$$\frac{\psi}{\delta}(s) = \frac{K}{s(s+a)}$$

$$\frac{y}{\delta}(s) = \frac{KU}{s^2(s+a)}$$

$$\frac{\psi a}{\delta}(s) = \frac{K(s+u/d)}{s^2(s+a)}$$

a) heading control

b) position control

c) aimpoint control

Fig. 2. Root loci of the lateral control modes

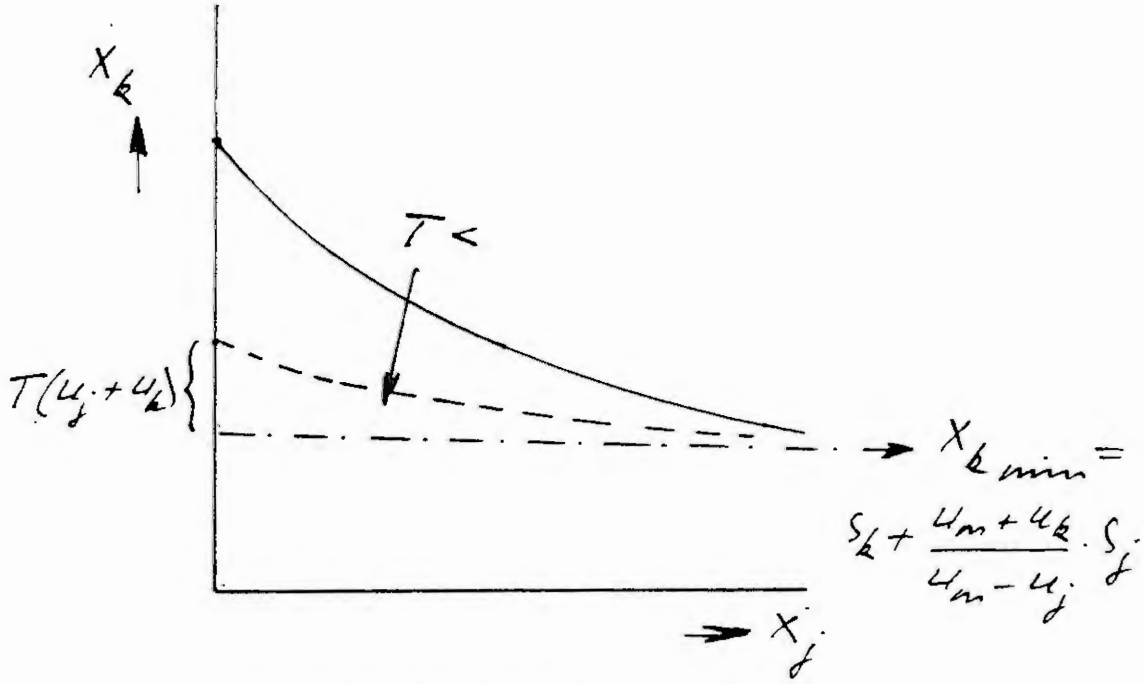


Fig. 4. X_k as a function of X_j.

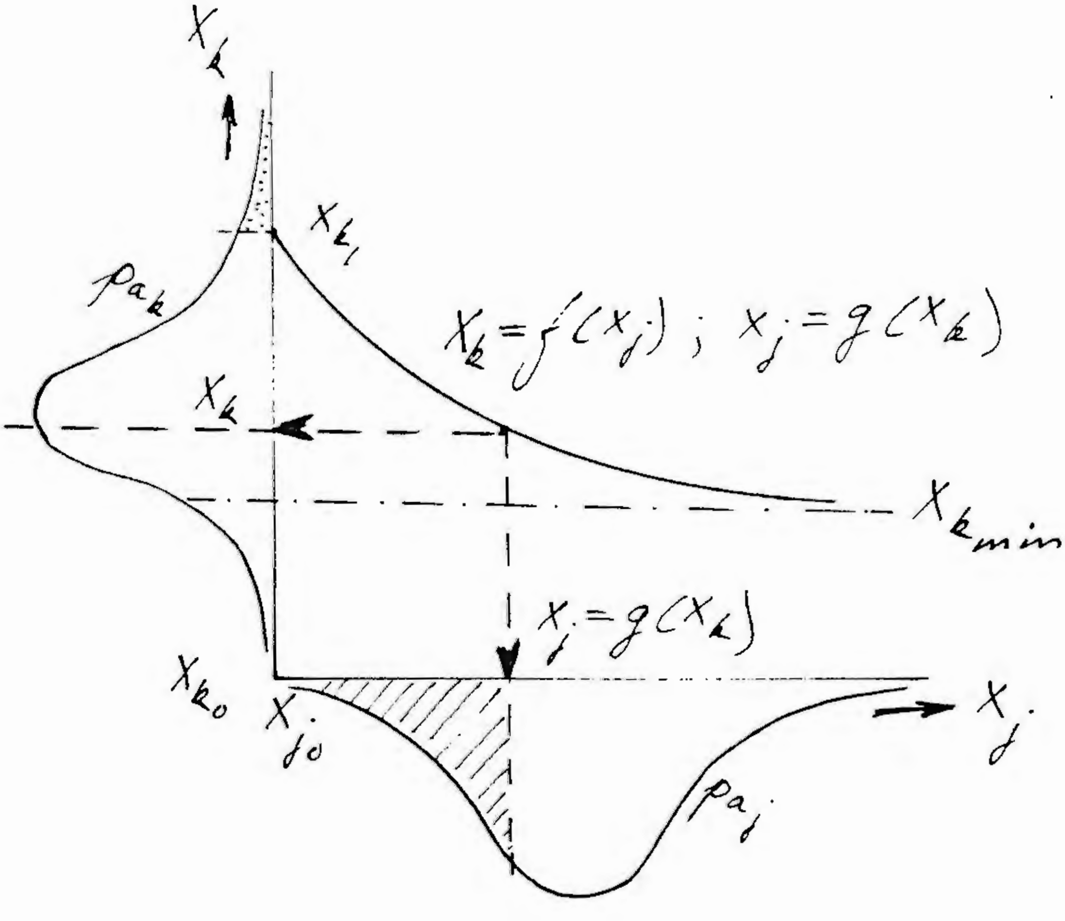


Fig. 5. The statistical relationship between X_j and X_k.

DYNAMIC MODELLING OF A FLEXIBLE-LINK MANIPULATOR

by

A. S. Morris and A. Madani

Robotics Research Group, Dept. of Automatic Control and Systems Engineering,
University of Sheffield, P O Box 600, Mappin Street, Sheffield S1 4DU, U.K.

ABSTRACT

The work to be described involves the modelling of a robot manipulator with two flexible links connected to an actuated joint. The aim of the model is to predict all static and dynamic link deflections so that a model-based robot controller can be synthesised. The dynamic equations are derived from a modified Bernoulli-Euler beam model with a shear deformation effect model superimposed. Particular attention is given to the techniques used for overcoming computation difficulties in the model. The performance of the model is illustrated by various simulations.

1. INTRODUCTION

The control problems due to the large inertia forces generated when the links of current-generation robots move at high speed has stimulated research into the development of robots with lightweight links. The use of low-mass links greatly reduces the magnitude of inertia forces during motion and so avoids this particular robot control problem. However, the design of low mass links means using small section components which inevitably involve a degree of flexure under both static (gravity) and dynamic forces. This creates a requirement for a controller which can calculate the magnitude of and compensate for this flexure. Thus, in changing from a large mass rigid-link manipulator to a small. mass flexible-link one, we have merely replaced one control problem for a different one, but one that is hopefully easier to solve.

This paper describes the development of a model of a flexible link manipulator system which will be suitable for use in a model-based controller. The theoretical basis of the model is a modified Bernoulli-Euler beam model on which a shear deformation effect model has been superimposed. This is described in section 2 following.

The model developed is for a single flexible link. When two links are joined together, the motion of the first link causes movement of the origin of the second link. Hence, to simulate motion of a two-link system, an iterative procedure is required, where the output of the link 1 model at the end of each iteration becomes the initial condition for the link 2 model. To simplify the development of this, a system consisting of one rigid link and one flexible link was simulated first, as described in section 3. The extension of this to a two-flexible-link system is currently in progress and results will be presented at the Symposium. .

2. THEORETICAL BASIS OF FLEXIBLE-LINK MODEL

Previous research on the dynamic modelling of flexible link manipulators has concentrated on the Bernoulli-Euler beam approach. A basic assumption in this approach is that the plane sections of the flexible beam remain plane and normal to the elastic axis before and after deformation. Conse-

The authors have addressed this problem by using the Timoshenko beam model[7] and introducing some elements of finite element theory[5]. The Timoshenko beam model is a modified Bernoulli-Euler model which takes account of both shear deformation and rotational inertia effects and thus leads to a more accurate model, but at the expense of greater computational complexity. The following assumptions have been made in the development of this model:

- Axial elongation, Coriolis and torsional effects are negligable
- Frictional forces and motor backlash can be ignored
- Motion is in the vertical plane only
- The arm is uniform and prismatic

3. SYSTEM WITH ONE RIGID LINK AND ONE FLEXIBLE LINK

The initial two-link system simulated consisted of one rigid link connected via an actuated joint to one flexible link. With parameters as defined in figure 1, the following equations express the kinematic relationship of any point on the flexible link at a distance x from the actuation point, in terms of the inertia frame:

$$X = x cos\theta + w sin\theta \qquad\qquad Y = x sin\theta - w cos\theta$$

where w is the deflection calculated using the beam model developed.

For the rigid link, the coordinates of any point after a rotation of θ degrees are obtained by setting w to zero[4].

The model was simulated using an iterative procedure in which the outputs of the first (rigid) link become the input parameters for the second (flexible) link. The simulation was carried out for a set of different payloads and for a range of speeds of the actuator at the joint between the two links. Figure 2 shows the effect of variation in actuator speed on the deflection of the end point of the flexible link. Figure 3 similarly shows the effect of varying the payload. For purposes of comparison, figures 4 and 5 are included to show the error in the simulated deflections if the shear force term is omitted from the model.

4. SYSTEM WITH TWO FLEXIBLE LINKS

Extension to a two-flexible-link system is relatively simple and only requires the substitution of the flexible link model in place of the rigid one for link 1 in the simulation. Results will be presented at the symposium.

5. SUMMARY

A computationally efficient model of a flexible manipulator system has been developed and used to demonstrate the effect of variation in the joint rotation speed and payload. The necessity of modifying the basic Bernoulli-Euler model by adding the shear deformation term has been demonstrated. These results will be incorporated in future work within a model-based, flexible-manipulator controller.

REFERENCES

[1] Book, W.J., Maizza-Neto, O. and Whitney, D.E., Feedback Control of a Two-beam, two-joint system with distributed flexibility, J. Dynamic System, Measurement and Control, 97 (1975), 424-431.

[2] Book, W.J., Recoursive Lagrangian Dynamics of flexible Manipulator Arms, Int. J. Robotics Research, 3 (1984), 87-101

[3] Canon, R.H. and Schmitz, E., Initial Experiments on the End-point Control of a Flexible One-link Robot, Int. J. Robotics Research, 3 (1984), 62-75

[4] Fu, K.S., Gonzales, R.C. and Lee, C.S.G., Robotics: Control, Sensing, Vision and Intelligence. McGraw-Hill, New York, 1988.

[5] Naganathan, G. and Soni, A.H., Coupling Effects of Kinematics and Flexibility in Manipulators, Mechatronics, 2 (1992), 129-148.

[6] Sakawa, Y., Matsume, F. and Fukushima, S., Modelling and Feedback Control of a Flexible Arm, J. Robotics Systems, 4 (1985), 453-472.

[7] Timoshenko, S., Vibration Problems in Engineering. Van Nostrand, Princeton, N.J., 1955.

[8] Yang, G.B. and Donath, M., Dynamic Model of a One-link Robot Manipulator with both Structural and Joint Flexibility, Proc. of IEEE Int. Conf. on Robotics and Automation (1988), Vol 1, 476-481.

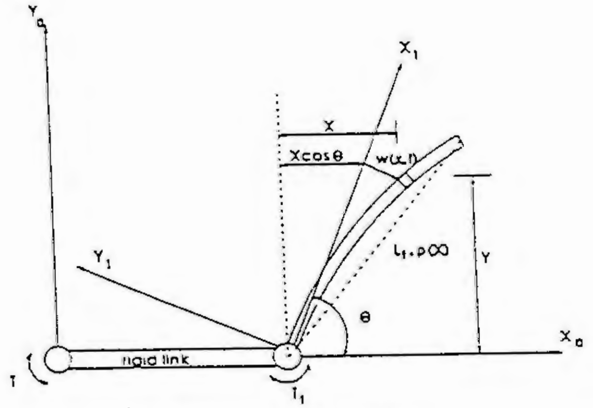


Figure 1. Configuration of the manipulator.

Figure 2. Shear force and speed effects on the position of the end-point.

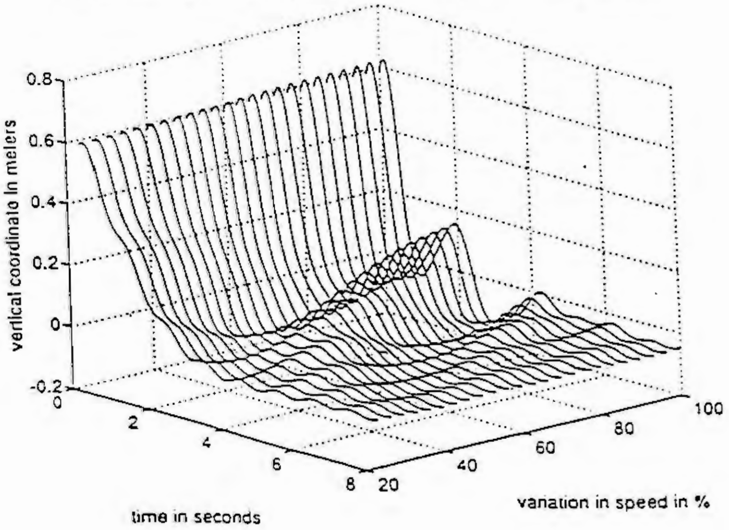


Figure 3. Shear force and pay-load effects on the position of the end -point

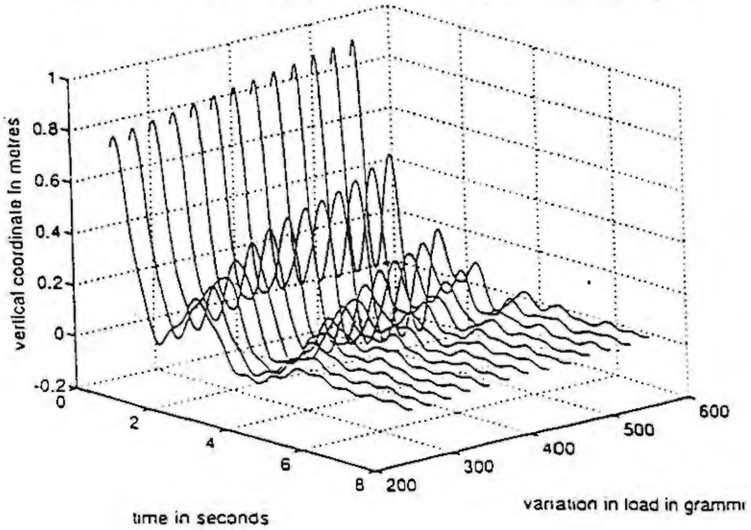


Figure 4. Speed effect on the position of the end-point.

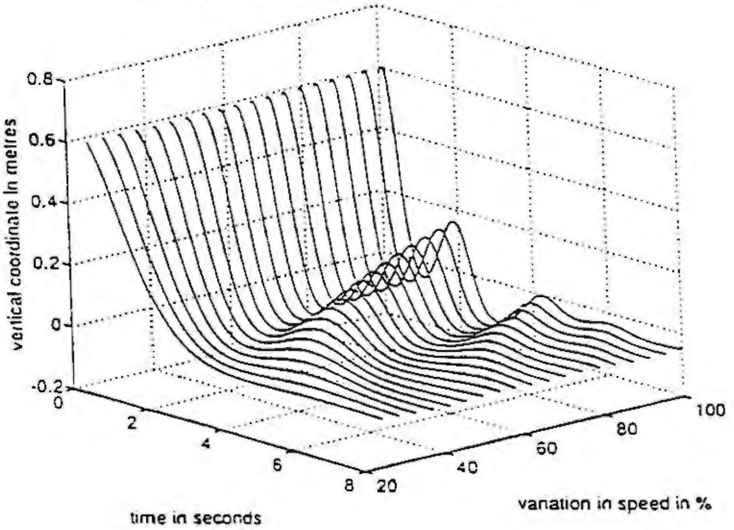
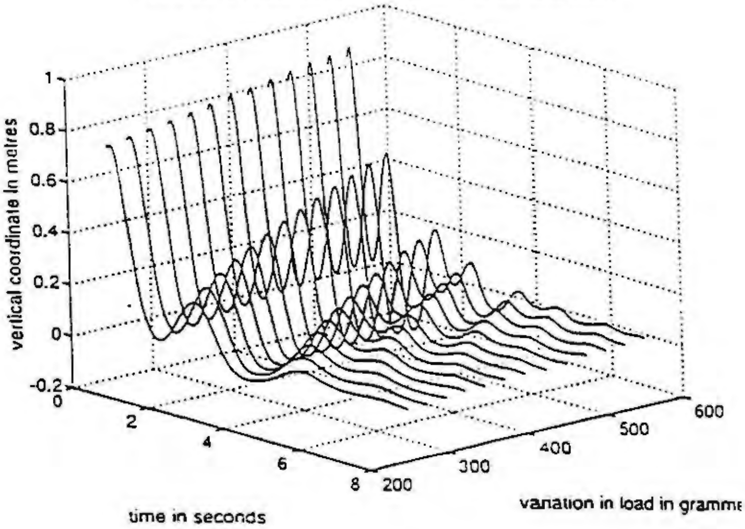


Figure 5. Pay-load effect on the position of the end-point



SIMULATION OF A ROBOTIC CONTROL SYSTEM: PID AND ATG SCHEMES

Spyros Tzafestas
National Technical
University of Athens
15773, Athens, Greece

Gerasimos Frangakis
Institute of Informatics and Telecommunications
NCSR "Demokritos"
15310 Aghia Paraskevi, Athens, Greece

Vincent Prival (*)
National Technical
University of Athens
15773, Athens, Greece

Abstract. In this paper some simulation results are presented concerning the position / trajectory control of a robotic manipulator by two types of controllers: PID controller and autonomous trajectory generating controller. First the PID control scheme utilized is studied followed by a brief presentation of the autonomous trajectory generating (ATG) scheme. Then the simulation results are provided and discussed.

1. INTRODUCTION

The control of robots involves many challenging problems and possesses certain peculiarities in comparison to other controlled technological systems. Among the control schemes utilized in robotics a high position is occupied by local PID control, computed torque control, adaptive control (several types) and robust control [1-3]. Our purpose here is to give a short account of the kind of performance obtained by local PID control and autonomous trajectory generating (ATG) control [4-5]. The analysis and modelling of robots has reached a very mature state with important results available. Due to space limitation we skip the discussion on robot modelling. The models employed here are the Euler-Lagrange model and the Newton Euler model [6].

2. PID CONTROL SCHEME

The PID robot control scheme has the organization shown in Fig.1 where B_0 is a zero-order holding block, and $K_{\tau i}$ is tachometer constant. The motor torque τ_i of the ith motor is equal to

$$\tau = K_m(u - K_\tau \dot{\theta}) \tag{1}$$

where the index i was suppressed for notational convenience, and the control signal (output of the PID controller) is given by

$$u(t) = K_{pc}\, e(t) + K_{ic} \int_0^t e(\sigma) d\sigma + K_{dc} \frac{d e(t)}{d t} \tag{2}$$

where $e(t) = \theta_{ref}-\theta$ is the position error. In discrete time the above PID control signal takes the form

$$u_k = K_{pd}\, e_k + K_{id} \sum_{s=1}^{k} e_s + K_{dd}\,(e_k - e_{k-1}) \tag{3}$$

where the new coefficients depend on the sampling period T as

$$K_{pd} = K_{pc}, \qquad K_{id} = K_{ic}\, T, \qquad K_{dd} = K_{dc}\,/T \tag{4}$$

The incremental form of (3) is

$$\Delta u_k = u_k - u_{k-1} = K_{pd}\,(e_k - e_{k-1}) + K_{id}\, e_k + K_{dd}\,(e_k - 2e_{k-1} + e_{k-2}) \tag{5}$$

The incremental form (5) allows to correct the control signal at each sampling period. The design problem is to select suitable values of the control gains K_{pd}, K_{id} and K_{dd} which result in desirable (or at least acceptable) step response. The selection of these coefficients can be performed by several techniques [7]. However, in robot control a given set of values of K_{pd}, K_{id} and K_{dd} does not lead to the same performance for the various robot configurations. A first way to face this problem is to divide the configuration space in a set of zones and assign a suitable set of gain values in each one of them. A table of gain triples is then constructed and applied to the control system with the aid of suitable logic.

(*) Under an exchange scheme between IDN (France) and NTUA within the EEC Erasmus programme.

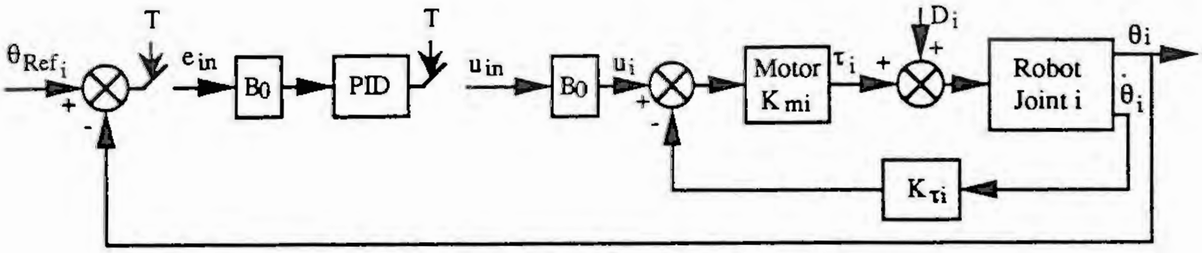


Fig.1. Block diagram of discrete-time PID robot control

3. ATG CONTROL SCHEME

The PID control is mostly suitable for the problem of driving the robot end effector to a final state (point B, zero velocity) starting from an initial state (point A, zero velocity) without any other consideration. Consider now the more general problem where the end effector is required, starting from the same initial state A, to follow a given path in cartesian coordinates, with continuous specified velocity along the path. To use the PID control in this case the continuous path must be transformed to a discrete path, i.e. to a sequence of equidistant (or not) points. The total number of these points depend on the required accuracy. To each one of these points one must associate a corresponding velocity. We thus obtain a sequence of states $E_0, E_1, ..., E_N$ from which the end effector must pass at the times $t_0, t_1, ..., t_N$. The PID control scheme can then be applied for each time interval $[t_{i-1}, t_i]$, but one needs a different gain tuning to each interval.

An alternative scheme is the so called Autonomous Trajectory Generating (ATG) control scheme which assures the independence between the path and velocity control. This technique is now briefly described.

3.1. Homogeneous Transformations

Consider Fig.2 and define the following homogeneous transformations:

i) Transformation for which the trajectory segment is defined by f(x',y')=0, z'=0.

$$B_s = \begin{bmatrix} L & \vdots & p_s \\ \dots\dots & \vdots & \dots \\ 0\ 0\ 0 & \vdots & 1 \end{bmatrix}, \ L = [s_n\ s_o\ s_a],\ L^{-1} = L^T$$

ii) Transformation directly attached at a current point Q of the trajectory

$$S_c = \begin{bmatrix} H & \vdots & p'_c \\ \dots\dots & \vdots & \dots \\ 0\ 0\ 0 & \vdots & 1 \end{bmatrix}, \ H = [c'_n\ c'_o\ c'_a],\ H^{-1} = H^T$$

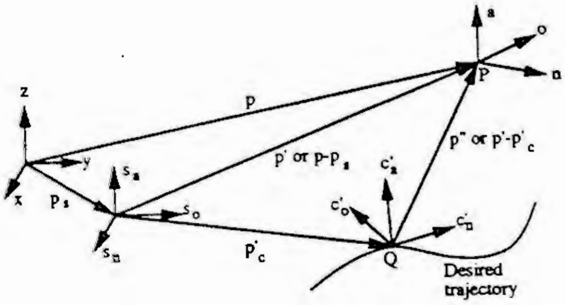


Fig.2. Definition of homogeneous transformations (All vectors with (') are defined in S, all vectors with (") are defined in C, and all the others in B)

where c'_n is tangential to the trajectory at Q, and c'_o is perpendicular. If $\nabla f = [f_x, f_y]^T$ is the gradient of f, then

$$c'_n = \begin{bmatrix} f_y/|\nabla f| \\ -f_x/|\nabla f| \\ 0 \end{bmatrix}, \ c'_o = \begin{bmatrix} f_x/|\nabla f| \\ f_y/|\nabla f| \\ 0 \end{bmatrix}, \ c'_a = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

iii) Transformation directly attached to the end effector

$$B_R = \begin{bmatrix} n\,o\,a & \vdots & p \\ \dots\dots & \vdots & \dots \\ 0\,0\,0 & \vdots & 1 \end{bmatrix}$$

Now, define

$$ {}^{S}R = \begin{bmatrix} n'\, o'\, a' \vdots p' \\ \dots\ \dots \vdots \dots \\ 0\ 0\ 0 \vdots 1 \end{bmatrix}, \qquad {}^{C}R = \begin{bmatrix} n''\, o''\, a'' \vdots p'' \\ \dots\ \dots \vdots \dots \\ 0\ 0\ 0 \vdots 1 \end{bmatrix} $$

Then
${}^{B}R = S.{}^{S}R$, [n, o, a] = L [n', o', a'], $p\text{-}p_s = Lp'$; ${}^{S}R = C.{}^{C}R$, [n', o', a'] = H [n", o", a"], $p'\text{-}p_c = Hp''$

3.2. Control Principle

The control is performed in cartesian coordinates. For a position P of the end effector, Q is defined to be the point of the desired trajectory nearest to P (see Fig.3). The control equation is the following

$$ \ddot{p}' - \ddot{p}'_r + K_1(\dot{p}' - \dot{p}'_r) + K_2 \ddot{+} (p' - p'_c) = 0 \quad (6) $$

where $K_1>0$ and $K_2>0$ are suitable constants, i.e. such that the characteristic polynomial $\lambda^2+K_1\lambda+K_2$ has roots with desired negative real values. Obviously, the error is then convergent to zero. Equation (6) can be written in the following form which is nearer to the given data:

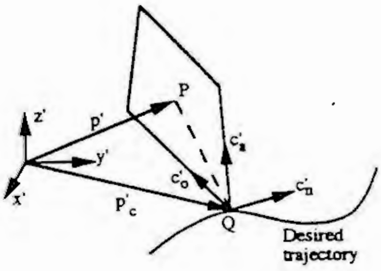


Fig.3. Definition of point Q

$$ H^T\ddot{p}' + K_1 H^T\dot{p}' + K_2 H^T(p' - p'_c) - H^T\ddot{p}'_r - K_1 H^T\dot{p}'_r = 0 \qquad (7) $$

However, all parameters are known in B and not on S. We therefore have:

$$ H^T\dot{p}'_r = \begin{bmatrix} v(t) \\ 0 \\ 0 \end{bmatrix}, \qquad H^T\ddot{p}'_r = \begin{bmatrix} \dot{v}(t) \\ 0 \\ 0 \end{bmatrix} - \dot{H}^T H \begin{bmatrix} v(t) \\ 0 \\ 0 \end{bmatrix} \text{ with } \dot{H}^T H = \begin{bmatrix} 0 & \Omega_z & 0 \\ -\Omega_z & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} $$

and $\Omega_z = [f_y(f_{xx}\dot{x} + f_{xy}\dot{y}) - f_x(f_{xy}\dot{x} + f_{yy}\dot{y})]/|\nabla f|^2$.

Now, noting that $p'' = [o, f/|\nabla f|, z'']^T$ (see Fig.2) and that $p'\text{-}p'_c = Hp''$, eq.(7) reduces to

$$ \ddot{p} = \ddot{p}_s - K_1(\dot{p} - \dot{p}_s) + r\,,\ r = LH\left[K_1 v(t) + \dot{v}(t),\ -\left(K_2 f/|\nabla f|\right) + \Omega_z v(t),\ -K_2 z''\right]^T \qquad (8) $$

In the majority of cases ps is constant, and so eq.(8) reduces to: $\ddot{p} = -K_1\dot{p} + r$.

Finally using the Jacobian relation $\dot{p} = J\dot{q}$, this equation can be written in terms of the joint coordinates as:

$$ \ddot{q} = -K_1\dot{q} + J^{-1}(r - \dot{J}\dot{q}) \qquad (9) $$

This is the required ATG equation. The overall scheme used in our simulation is shown in Fig.4.

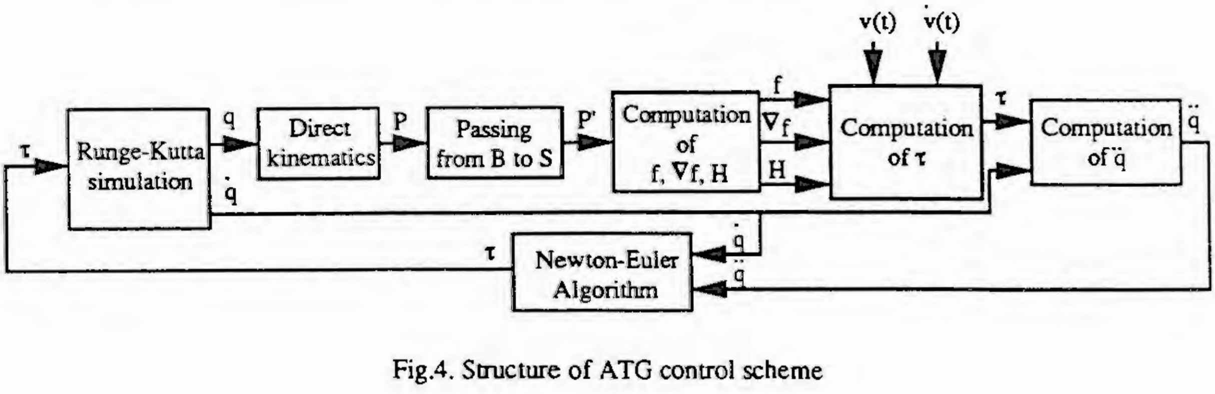
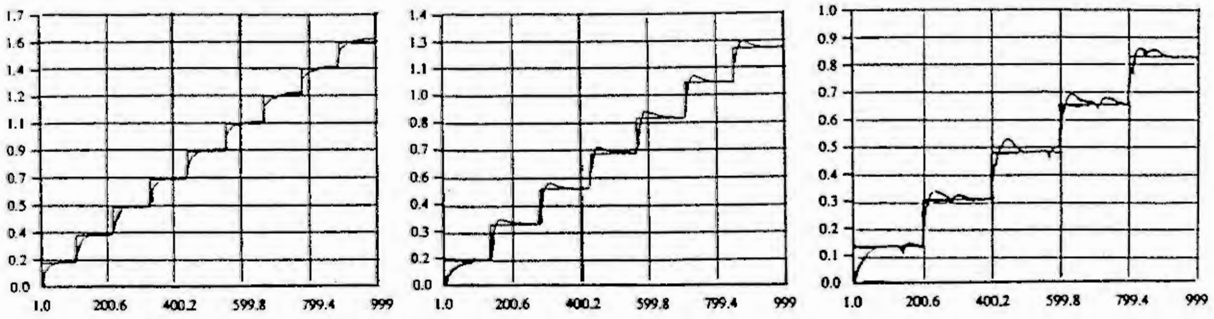


Fig.4. Structure of ATG control scheme

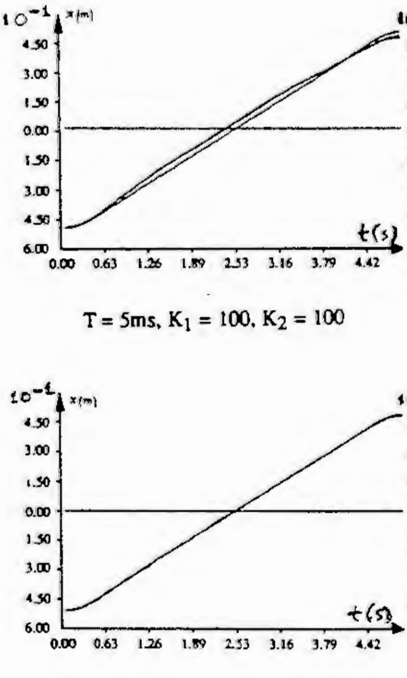


(a) θ_1(rad)

(b) θ_2(rad)

(c) θ_3(rad)

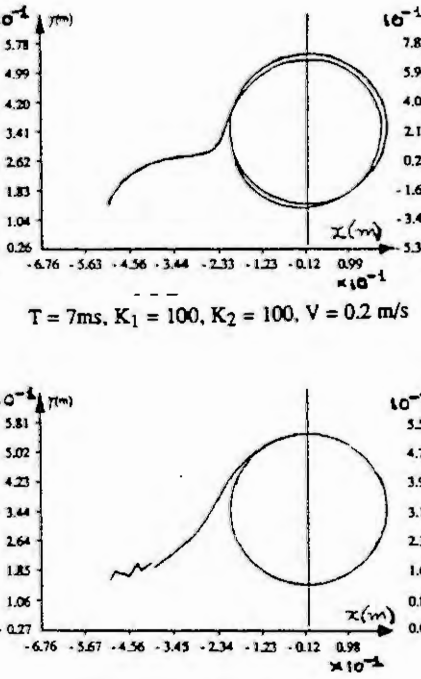
Fig.5. Robot responses obtained with PID control (T=0.01sec)



T = 5ms, K_1 = 100, K_2 = 100

T = 5ms, K_1 = 1000, K_2 = 1000

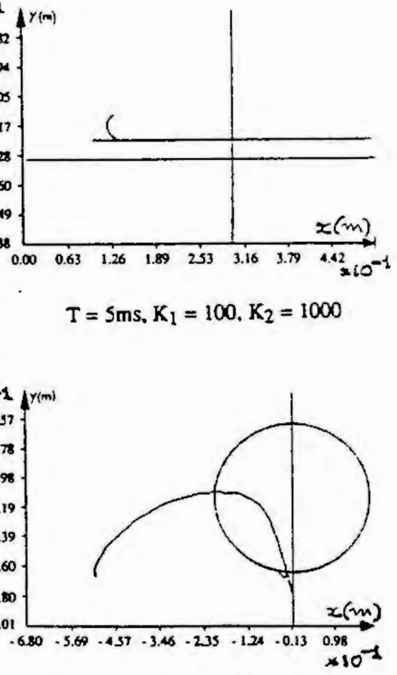
Fig.6. ATG Control



T = 7ms, K_1 = 100, K_2 = 100, V = 0.2 m/s

T = 7ms, K_1 = 1000, K_2 = 1000, V = 0.2 m/s

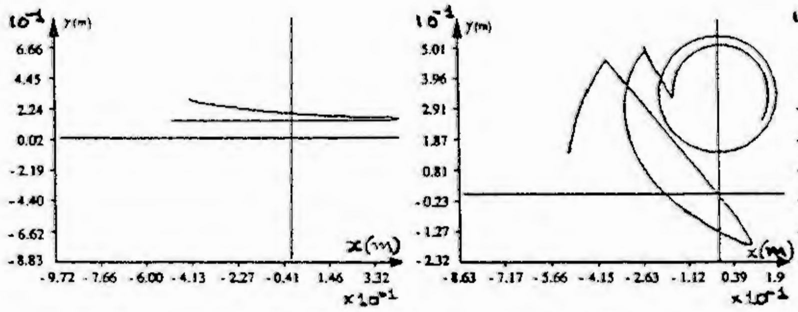
Fig.7. ATG Control



T = 5ms, K_1 = 100, K_2 = 1000

T = 7ms, K_1 = 100, K_2 = 1000, V = 0.2 m/s

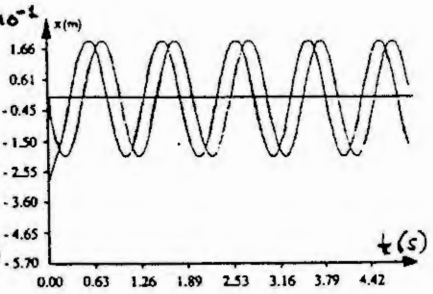
Fig.8. ATG Control



T = 5ms, K_1 = 1000, K_2 = 500

T = 1ms, K_1 = 1000, K_2 = 100, V = 0.2 m/s

Fig.9. ATG Control



T = 1ms, K_1 = 1000, K_2 = 6000, V = 1.25 m/s

Fig.10. ATG Control

4. SIMULATION STUDY

A large number of simulation results were obtained using the two control techniques described above. Some of them corresponding to a 3-link robot will be described below. The robot parameters are:

$m_1 = 1Kg$,	$L_1 = 0.15m$,	$I_{a1} = 0.05Kg.m^2$,	$K_{m1} = 1Nm/V$
$m_2 = 1Kg$,	$L_2 = 0.35m$,	$I_{a2} = 0.0004Kg.m^2$,	$K_{m2} = 10Nm/V$
$m_3 = 0.25Kg$,	$L_3 = 0.25m$,	$I_{a3} = 0.0004Kg.m^2$,	$K_{m3} = 2Nm/V$

where the actuators (motors) are represented by the constants K_{mi}. Defining the state variables as $x_1 = \theta_1$, $x_2 = \theta_2$, $x_3 = \theta_3$; $x_4 = \dot{\theta}_1$, $x_5 = \dot{\theta}_2$, $x_6 = \dot{\theta}_3$, the robot equations can be written as:

$$\dot{x}_1 = x_4, \ \dot{x}_2 = x_5, \ \dot{x}_3 = x_6, \ \dot{x}_4 = \tau_1/D_{11}$$

$$\dot{x}_5 = (D_{33}\,\tau_2 - D_{23}\,\tau_3 + H_{211}\,x_4^2 + H_{222}\,x_5^2 + H_{222}\,x_5\,x_6 - H_2)\,/\,H_{222}$$

$$\dot{x}_6 = (D_{22}\,\tau_3 - D_{23}\,\tau_2 + H_{311}\,x_4^2 + H_{322}\,x_5^2 + H_{322}\,x_5\,x_6 - H_3)\,/\,H_{222}$$

This set of equations was solved via a 4th-order Runge Kutta technique (see Fig.4).

4.1. PID Simulation Results

Since a very small value of the sampling period T requires a very high computing power, T cannot be selected extremely small. With T = 0.1sec, it was found that θ_3 diverges after some tenths of second. Thus we have selected T = 0.01sec and adapted the PID gains to this value. The results are shown in Fig.5. It is useful to observe the joints' interactions. Joint 1 does not have any influence upon the other two joints, but joint 3 has a strong influence on joint 2. At low speeds the term D211 $\dot{\theta}_1^2$ (or H211 $\dot{\theta}_1^2$) can be neglected.

4.2. ATG Simulation Results

Two types of trajectories were considered, i.e. a straight line segment of length 1 m, and a circle of radius 20 cm. For a given ratio K_1 / K_2, the accuracy of the trajectory tracking is better if K_1 and K_2 are large (see Figures 6 and 7). Figure 8 shows the kind of results obtained when $K_2 >> K_1$, and Fig.9 those obtained when $K_1 >> K_2$.

5. CONCLUSIONS

The local PID control scheme, although very simple, is not sufficiently robust. This means that when the robot parameters are known with some uncertainty the errors in trajectory tracking increase prohibitively. The ATG control scheme is robust. At constant motion speed, the external perturbations and/or motor saturations do not have a measurable effect on the control performance (Fig.10). However, we have observed that this is not so if the motion is not uniform.

REFERENCES

[1] S.G. TZAFESTAS, Dynamic modelling and adaptive control of industrial robots, SAMS: Syst. Anal. Modelling Simul., Vol.6, No.4, pp.243-266 (1981).

[2] S.G. TZAFESTAS and L. DRITSAS, Combined computed torque and model reference adaptive control of robot systems, J. Franklin Inst., Vol.327, pp.273-294 (1990).

[3] S.G. TZAFESTAS, Adaptive, robust and rule-based control of robotic manipulators, In: Intelligent Robotic Systems (S.G.Tzafestas, ed.), Marcel Dekker, pp.313-419 (1991).

[4] K. HASEGAWA and T. MIZUTANI, Manipulator control using autonomous trajectory generating servomechanism, Proc. Intl. Symp. of Robotics Research, Kyoto (1984).

[5] S. TZAFESTAS, B. KALOBATSOS, G. STAVRAKAKIS and A. ZAGORIANOS, Some results concerning the autonomous trajectory generation and adaptive control of industrial robots, J. Franklin Inst., Vol.329, No.1, pp.1-14 (1992).

[6] J.Y.S. LUH, M.W. WALKER and R.P.C. PAUL, Resolved-acceleration control of mechanical manipulators, IEEE Trans. Auto. Control, Vol.AC-25, No.3, pp.468-474 (1980).

[7] S.G. TZAFESTAS and J.K. PAL, Digital control algorithms, In: Real Time Microcomputer Control of Industrial Processes (S.G.Tzafestas and J.K.Pal eds.), Kluwer, pp.81-138 (1990).

Cooperative Neural Field for the Path Planning of a Robot Arm

P. Muraca
D.E.I.S. Università della Calabria
87030 Rende Italy
pietro@elena.deis.unical.it

G. Raiconi
D.I.A. Università di Salerno
V. Chirico, 84081 Baronissi (Sa) Italy
gianni@udsab.dia.unisa.it

T. Varone
SIPI s.r.l.
Pomigliano d'Arco (NA) Italy

Abstract. This paper deals with the problem of finding a good trajectory, from an initial position to a prescribed target point, for the end effector of a robot arm moving on a two dimensional work field and avoiding obstacles lying on the work field. Two algorithms based on cooperative neural fields are proposed: the former is suited for the case where the location of obstacles is known, the latter doesn't require any a priori knowledge and is based on a very crude collision detector.

1 INTRODUCTION AND PROBLEM STATEMENT

In this paper the path planning problem for a robot arm is considerd. The motion of the end effector is constrained on a plane and the arm itself consists of coplanar links connected by rotational joints in such a way that the configuration of the arm is uniquely defined by the angles θ_i, $i = 1, 2, \ldots, n$ and the workspace is delimited by the full extension of the arm. A point of the workspace can be occupied (i.e. it lies on an obstacle) or free (i.e. it lies on the so called free space) : the problem is to find a trajectory of the arm, belonging entirely to the free space, that joins the initial point of the end effector to the target point with a line of minimal length. This problem is related to the path planning problem for a mobile robot (navigation problem), largely treated in the literature [1–4], but it is much more complex. This is because in the navigation problem only the trajectory of a point, representing the robot itself, supposed of small dimensions, must avoid obstacles, where as in the present problem, the trajectories of all the points lying on the links must be taken into account. It is also relevant for the solution of the problem what type of knowledge is used in the construction of a trajectory. In a global navigation system a complete map of the locations of obstacles on the whole work field is assumed to be known: such an assumption is not realistic in real time applications. In the local path-planning approach data obtained by a sensory system are used, both in a merely reactive fashion, either as means for constructing a partial map of the workspace, following the strategy called anticipatory planning. In this paper some modifications are introduced to the algorithm given by Lemmon [5] for the navigation problem, that enable to solve the path planning of the arm. First of all the problem is discretized: a regular, square shaped mesh is superimposed to the workspace and only the knots of the mesh are considered as candidates to be occupied by the moving robot. The mesh points are labeled by ordered pairs (i, j), all the discretized workspace is represented by the set $\{(i, j), 1 \leq i \leq N, 1 \leq j \leq N\}$, the free workspace by the set of all (i, j) that the robot may occupy, the collection of obstacles is represented by the free workspace complement. A point (r, l) is near to (i, j) if : $r \in \{i-1, i, i+1\}$ and $l \in \{j-1, j, j+1\}$, and N_{ij} denotes the neighbors of (i, j):

$$N_{ij} = \{(r, s) \mid \max(| r - i |, | s - j |) \leq 1\}.$$

Let (i_s, j_s) the label of the starting point and (i_G, j_G) the target label: a trajectory (for the navigation problem) is a sequence of locations (i_k, j_k), $k = 0, 1, \ldots, F$ starting at the assigned initial point $(i_0, j_0) = (i_s, j_s)$ and terminating to the target: $(i_F, j_F) = (i_G, j_G)$, which has the property that (i_{k+1}, j_{k+1}) is near to $(i_k, j_k) \forall k$. A trajectory is admissible if (i_k, j_k) lies on the free space $\forall k$. A trajectory is optimal if F is minimal in the set of all admissible trajectories. An admissible trajectory for the robot arm is an admissible trajectory of the end effector such that:

- The trajectory of the end effector is compatible with the kinematics of the arm.
- During the trajectory no link passes over a point lying on an obstacle

Since obstacle points are discrete whereas link positions are continuous, in the simulation of a collision test the crossing of a link with segments (vertical horizontal or diagonal) joining obstacle points must be considered.

2 COOPERATIVE NEURAL FIELD

In the solution of the problem we use the neural network proposed in [5]. It consists of N^2neurons arranged as a $2D$ sheet called *neural field*. The neurons are put in one to one correspondence with the locations of the workspace and labeled by the ordered pairs (i,j). Any neuron (i,j) is characterized by:

- a short-term activity state STA denoted by x_{ij}
- a long-term activity state LTA denoted by w_{ij}
- an external disturbance y_{ij} : $y_{ij} = \begin{cases} 1 & : \ (i,j) = (i_G, j_G) \\ 0 & : \ \text{elsewere} \end{cases}$, used to start the evolution of the net.
- a rate constant λ_{ij} : $\lambda_{ij} = \begin{cases} 1 & : \ (i,j) \text{ is on free space} \\ 0 & : \ (i,j) \text{is on obstacle} \end{cases}$

The STA and LTA dynamics is governed by the state equations:

$$x_{ij}^{+} = G\left(x_{ij}^{-} + \lambda_{ij}\, y_{ij} + \lambda_{ij} \sum_{(r,s) \in N_{ij}} D_{r,s} \delta_{-1}(x_{ij}^{-}) \right) \quad . \tag{1}$$

$$w_{ij}^{+} = w_{ij}^{-} + \lambda_{ij} \mid \delta_{-1}(x_{ij}^{+}) - \delta_{-1}(x_{ij}^{-}) \mid \tag{2}$$

where the plus and minus notation denote new and old values of the state, respectively. The function G is defined by:

$$G(\xi) = \begin{cases} 0 & : \ \xi \leq 0 \\ \xi & : \ \xi > 0 \end{cases}$$

The constant D_{rs} are weights which characterize the behavior of the network: if they are assigned as :

$$D_{rs} = \begin{cases} -A & : \ (r,s) = (i,j) \\ > 0 & : \ (r,s) \neq (i,j) \end{cases} \quad A \geq 2\,\mathrm{card}(N_{ij}) + 1$$

the dynamics of the net is cooperative in the sense that a given neuron turning active increases the STA state of its neighbors. The LTA state equation shows clearly that w_{ij}counts the numbers of changes of the STA state of (i,j) neuron. The algorithm, shown in [5], consists of the iteration of equations $(1,2)$ starting from $w_{ij} = 0$, $x_{ij} = 0 \, \forall (i,j)$ until the neuron representing the initial position changes STA state the first time. It can be proved [5] that:

- the algorithm stops in at most $N+1$ iterations.
- Let (i,j) a point of an optimal trajectory and (r,s) the next point of the same trajectory ; (r,s) is such that

  $$(r,s) = \arg \max_{(l,h) \in N_{ij}} (w_{lh})$$

  Due to this property an optimal trajectory can be easily constructed from any initial position to the target by joining, starting from the initial position, any neuron with its neighbor which has the maximum LTA state.

3 ROBOT ARM PATH PLANNING

In the following we consider a 2 degree of freedom, mechanical arm consisting of two coplanar links of the same length with two rotational joints (fig.1); the approach can be extended easily to *n–joints* coplanar arm. Firstly the Lemmon's algorithm is used to define a tentative trajectory for the end effector, then a full trajectory of the arm is obtained by solving the inverse kinematic problem. If the trajectory is not admissible the tentative trajectory is such that a link collides with an obstacle. Our approach consists in modifying the map of obstacles so that the new trajectory is constrained to evolve farther from the obstacle. This procedure will be iterated until the solution is found or the problem is classified as unfeasible. There is an indetermination in the definition of the trajectory due to the non uniqueness of the inverse kinematics. With the simple robot considered in this paper, the indetermination occurs when the arm has full or null extension ($\theta_2 = 0$ or $\theta_2 = \pi$). The proposed algorithms use a decision tree and a new iteration is started only when all the tree is recurred to the root. Two different algorithms depending from the information that we can use, have been developped. In the first algorithm a complete map of the workspace is given, and a knowledge if a collision occurred is needed; at any iteration we artificially enlarge the obstacles in order to take away the next trajectory from the obstacle. The set G, used in the description of algorithms, denotes the union of the set of points occupied by the arm in the rest position and of the target point.

Algorithm A1

```
step 1:  Set all states $x_{ij}$ and $w_{ij}$ to 0; set $\lambda_{ij} = 0$ on all the points lying on obstacles $\lambda_{ij} = 1$ elsewhere
step 2:  Iterate equations (1) and (2) until:
             the number of iterations is $N + 1$ or $x_{i_s j_s} = 1$
         if the number of iterations is $N + 1$ (the problem is unfeasible) stop else go to step 3
step 3:  Construct the trajectory of the end effector by choosing:
             $(i_{k+1}, j_{k+1}) = (r, s), \quad (r, s) \in N_{i_k j_k}, \quad w_{rs} = \max_{(l,h) \in N_{i_k, j_k}} \{w_{lh}\}$
step 4:  Solve the inverse kinematics of the arm, at any step verify if there are interference with the obstacles,
         when necessary explore the decision tree.
         If an admissible solution has been found stop else go to step 5
step 5:  Reset all states to 0, change $\lambda$ according to the rule:
             $\lambda^+_{ij} = 0$ if $(i, j) \notin G$ AND $\exists (r, s) \mid (r, s) \in N_{ij}$ AND $\lambda^-_{rs} = 0$
         go to step 2
```

It can be easily proved that the algorithm terminates in at most N external iterations. The most expensive task of the algorithm is the step 4, requiring the solution of the inverse kinematic problem in particular when the extension to the n-link arm is concerned. In field application this task can be overcome by using at any step of the trajectory the coordinates of the next point of the end effector as set point for the existing controllers.

In the second algorithm no a priori information is available about the location of obstacles but a device detects if a collision has occurred on the end effector or on a link. A partial map is constructed using the detector data: if a collision occurs with the end effector, then the point must be added to the obstacle set, if a collision occurs on a link, instead, any of the points occupied by the arm can be on the obstacle. Our approach is to discourage the algorithm to follow the path of the failed attempt. For this purpose we introduce real values for the rates λ_{ij}, interpreted now as probability that the point can belong to an optimal trajectory.

Algorithm A2

```
step 1:  Set $x_{ij} = 0$, $w_{ij} = 0$, $\lambda_{ij} = 1 \, \forall (i, j)$
step 2:  As in Algorithm A1
step 3:  As in Algorithm A1
step 4:  Solve the inverse kinematics of the arm, at any step verify if a collision has occurred.
         If an admissible solution is found stop else go to step 5
step 5:  If a collision has occurred on the end effector at location $(r, s)$ then let $\lambda^+_{rs} = 0, x_{ij} = 0, w_{ij} = 0 \, \forall (i, j)$
step 6:  If a collision has occurred on a link, let $(i, j)$ the location that the end effector had to reach,
         decrease the rate according to the rule $\lambda^+_{ij} = \alpha \cdot \lambda^-_{ij}$ ($\alpha$ fixed $0 < \alpha < 1$) ;
step 7:  If $\exists (r, s) \mid \lambda_{rs} \leq \beta$ then let $\lambda^+_{rs} = 0$ ( $\beta$ small positive constant)
step 8:  Reset $x_{ij} = 0, w_{ij} = 0 \forall (i, j)$, change rates $\lambda$ according to the rule:
             $\lambda^+_{ij} = \lambda^-_{ij} \cdot \alpha$ if $(i, j) \notin G$ AND $\exists (r, s) \mid (r, s) \in N_{ij}$ AND $\lambda^-_{rs} \leq 1$
         go to step 2
```

Note that if step 7 is omitted the algorithm can generate an infinite sequence of unfeasible trajectories. With the introduction of this expedient the algorithm has finite termination, the actual maximum number of iterations $\overline{N}$ depends on the choice of both α and β and it can be showed that even in this case $\overline{N} = O(N)$.

4 EXAMPLES

The efficiency of the proposed algorithms has been tested on several examples, here we report results obtained from the solution of two different examples which are particularly suited to show the characteristic behaviors of the proposed approach. In all cases the solution has been founded with a number of operations much lesser than the theoretical maximum, which is of order $O(N^2)$ for both algorithms.

Example 1

There are two obstacles scattered on the work field obstructing the direct path from the initial position of the hand to the target point; in this example N has been chosen equal to 10. *Algorithm A1* finds a good solution in two iteration (figs.2,3). The *algorithm A2*, which is completely blind, has spent five iterations in order to learn the obstacles configuration and then has found a solution which is 18% worse than the solution given by the first algorithm (not showed).

Example 2

The work field is limited by a circular shell, with two appendices where are located the start position and the target position of the end-effector; in this example N has been chosen equal to 20. Fig.4 shows the solution found in only one iteration by the *algorithm A1*. In this example even the *algorithm A2* performs very good in fact only two trial has been necessary to find a trajectory of the same length of that obtained from *algorithm A1* (figs. 5, 6).

5 CONCLUSIONS

A novel approach to the path planning of a robot arm is proposed: it uses cooperative neural fields to implement a highly parallel search strategy. The method can be adopted both when the configuration of the free space is completely known and when the free space must be learned by the robot itself. The results obtained applying the proposed algorithms to several examples are encouraging; the approach seems to be very attractive for real time applications. At least from a theoretical point of view a drawback of the algorithms proposed is the possibility that a soluble problem can be classified as unfeasible. We are looking for an improvement that can exclude this possibility without increasing the computational effort. At the same time we are working in order to generalize the approach to the case of multi-link and non-planar robot arms.

References

[1] J. C. Latombe *Robot Motion Planning*, Kluver Academic Publisher, Boston, 1991.

[2] R. C. Arkin *Motor Schema based mobile robot navigation*, Int. Jour. of Robotics Research, 8, (4), 99-112, 1989.

[3] R. B.Tilove *Local obstacle avoidance for mobile robot based on method of artificial potential*. IEEE Int. Conf. Robotics and Automation, Cincinnati 1990.

[4] J. Connel *Minimalist Mobile Robots*, Academic Press, Boston, 1990.

[5] M. Lemmon *2-Degree-of-freedom Robot Path Planning using Cooperative Neural Fields*. Neural Computation 3, 350-362, 1991.

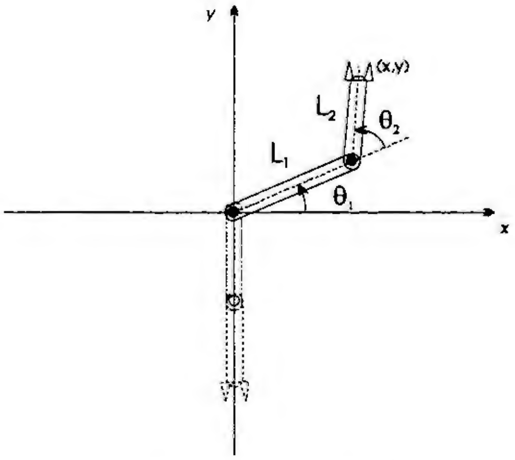


Fig. 1. The parameters of the arm
and its starting position

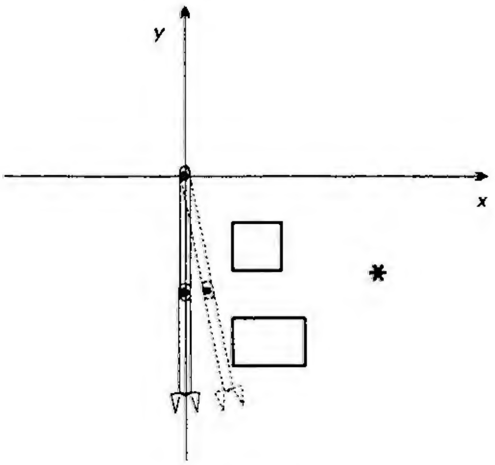


Fig. 2.*Algorithm 1.*
Example 1: the first trial.

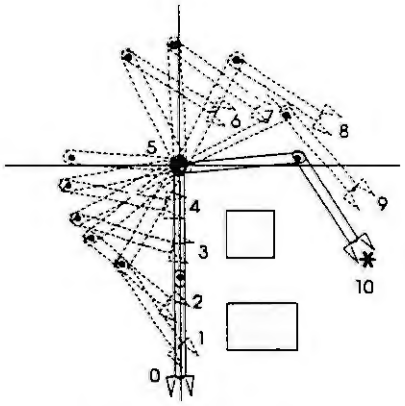


Fig. 3. *Algorithm 1.*
Examlpe 1: the second trial.

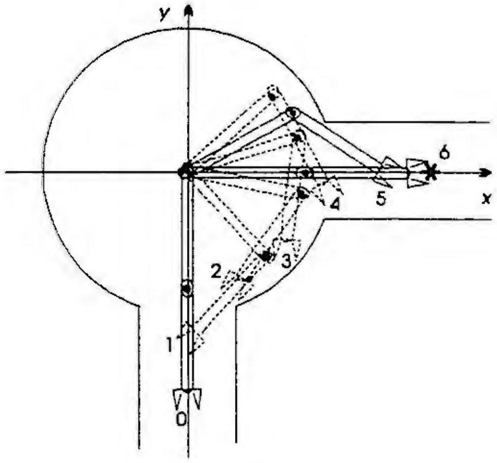


Fig. 4. *Algorithm 1.*
Example 2: the first and only trial.

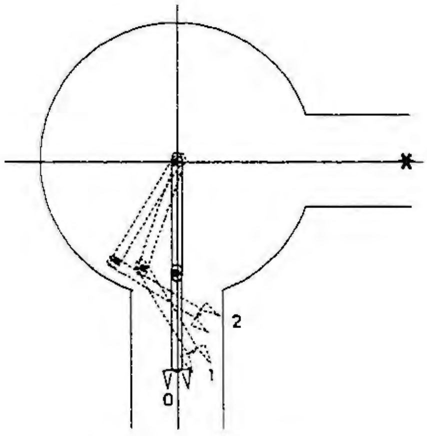


Fig. 5. *Algorithm 2.*
Example 2: the first trial.

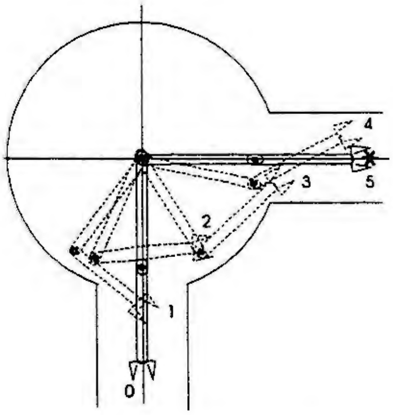


Fig. 6. *Algorithm 2.*
Example 2: the second trial.

On the order of observable monovariable systems obtained by sliding mode control strategies

Pierre LOPEZ - Ahmed Saïd NOURI - Hebert SIRA-RAMIREZ
GARI/LESIA/INSA
Av. de Rangueil, 31077 TOULOUSE cedex, France

Abstract

A systematic procedure is proposed for the determination of the system relative degree using a growing family of sliding surface candidates in the phase space of the system. A sliding regime exists on the zero level set on a candidate surface if, and only if, it is relative degree one. The procedure yields the order if, and only if, the system is differentially flat.

1. Introduction:

An important problem in the automatic regulation of systems is the determination of the orders of the associated system model (system dimension n, relative degree n* and dimension of zero dynamics α=n-n*) whose performance is to be improved by means of state feedback. This usually considers two possible operating modes: stabilisation and tracking, for repetitive or non repetitive tasks. In this article, some new concepts introduced by Fliess [1] [2] [3], Sira-Ramirez [4] and Messager [5] (Generalized Controller Canonical Form, linearizing dynamical state feedback and generalized variable structure system dynamics) will be used to propose a discontinuous feedback approaches for the determination of system orders. In particular *sliding modes* will be used [7].

A *sliding surface* may be defined in terms of state or phase variables depending on observability properties of the system. Nonlinear sliding surface may be proposed in general. However the sliding manifold is usually defined as an hyperplane due to simplicity of synthesis and the possibility of inducing closed loop linear dynamics.

The objective of this paper is to show that, in the case of a free system fedback with a discontinuous control, the analysis of the sliding surface coordinate allows to validate the local dimensions of the presumed model associated to the system.

2. Mathematic formulation of the problem:

Consider an n-dimensional monovariable dynamic system of the form:

$$\begin{cases} \dfrac{dx}{dt} = f(x,u) \\ y = h(x) \end{cases} \qquad (1)$$

where $u \in R$, $x \in R^n$, $y \in R$ are respectively the input, the state and the output. Under very mild conditions ([12] Conte, Moog & Perdon) there exists an input-dependent state coordinate transformation:

$$z = \Phi(x, u, \dot{u}, \ldots, u^{(\alpha-1)}) \qquad (2)$$

that places system (1) in generalized observability canonical form (GOCF)

$$\begin{cases} \dot{z}_i = z_{i+1} \quad i = 1,\dots,n-1 \\ \dot{z}_n = C(z,u,\dot{u},\dots,u^{(\alpha)}) \\ y = z_1 \end{cases} \tag{3}$$

where α=n-r is the dimension of the *zero dynamics* ($C(0,u,\dot{u},\dots,u^{(\alpha)})=0$). In case α=0, the system is said to be *differentially flat* [13] with linearizing output given by y=h(x).

The relations of the GOCF with Isidori [14] normal canonical form are clear from the fact that only state dependent coordinate transformations are allowed $(\eta,\xi)=\psi(x)$. Such that $\frac{\partial}{\partial u}\left[\dot{\xi}\right]=0$, which take the system into:

$$\begin{cases} \dot{\eta}_i = \eta_{i+1} \quad i = 1,\dots,r-1 \\ \dot{\eta}_r = \Theta(\eta,\xi,u) \\ \dot{\xi} = \sigma(\xi,\eta) \\ y = \eta_1 \end{cases} \tag{4}$$

The autonomous differential equation $\dot{\xi}=\sigma(\xi,0)$ is also the zero dynamics of dimension α=n-r.

A stabilizing sliding surface can be readily proposed for the above system as:

$$s = \lambda_1\eta_1 + \lambda_2\eta_2 + \dots + \lambda_{r-1}\eta_{r-1} + \eta_r \tag{5}$$

such that the set $\{\lambda_1,\lambda_2,\dots,\lambda_{r-1},1\}$ constitute coefficients of an r-th order Hurwitz polynomial.

If s is forced to zero in finite time, the closed loop dynamics is given by:

$$\begin{cases} \dot{\eta}_i = \eta_{i+1} \quad i = 1,\dots,r-1 \\ \dot{\eta}_{r-1} = -\lambda_1\eta_1 - \lambda_2\eta_2 - \dots - \lambda_{r-1}\eta_{r-1} \\ \dot{\xi} = \sigma(\xi,\eta_1,\eta_2,\dots,\eta_{r-1},-\lambda_1\eta_1 - \lambda_2\eta_2 - \dots - \lambda_{r-1}\eta_{r-1}) \\ y = \eta_1 \end{cases} \tag{6}$$

it is assumed that the system is *minimum phase* i.e. the dynamics $\dot{\xi}=\sigma(\xi,0)$ is asymptotically stable to an equilibrium point.

The creation of a sliding motion on s entitles forcing the coordinate s to satisfy the reaching condition: $s\dot{s}<0$. This, may be guaranteed by adopting the following dynamics with discontinuous right hand side for s:

$$\dot{s} = -W\operatorname{sign}(s) \tag{7}$$

Evidently a sliding regime can always be locally created on an open set of s=0 if and only if $\dot{s}$ depends explicitly on u i.e. if the s is relative degree one with respect to u.

The basic result to be used in the simulation and experimental procedure which is here proposed is based in the fact a sliding regime exists on a given representative of a family of sliding surfaces if, and only if, the chosen coordinate has relative degree equal to one with respect to u.

So, consider a growing sequence of sliding surfaces:

$$s_\beta = \eta_\beta + \sum_{i=1}^{\beta-1}\lambda_{i,\beta}\eta_i \qquad \beta=1,2,\dots$$

with $\{\lambda_{1,\beta},\lambda_{2,\beta},\dots,\lambda_{\beta-1,\beta},1\}$being the coefficient of an β-th order Hurwitz polynomial, but otherwise arbitrary. It is clear that a sliding regime exists on $s_\beta=0$ if and only if β=r.

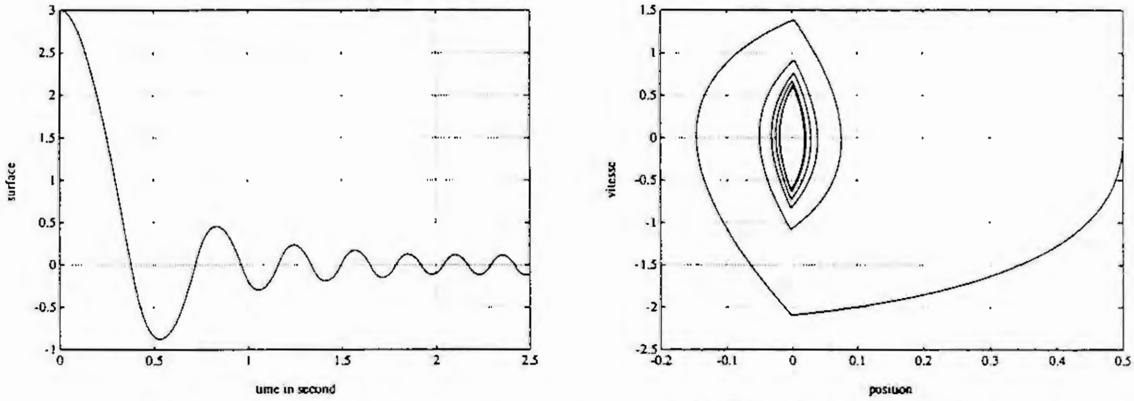
The experimental procedure is based on sequentially testing the sliding surface coordinate s_β for β=1,2,... until a sliding regime locally appears on $s_\beta = 0$. Note that if the system is differentially flat a sliding regime appears, for the first time, on $s_n = 0$ and there for the order of the systems is determined.

3. Simulation results:

Consider for simulation purposes a second order system already in normal form:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -3x_1 - 4x_2 + u \\ y = x_1 \end{cases}$$

Choosing $s_0 = \lambda x_1$, for, say λ=6, the condition $s_0 \dot{s}_0 < 0$ is only satisfied in the second and fourth quadrants of the phase space $\{y, \dot{y}\}$ and a sliding regime does not exist on $s_0 = 0$ as demonstrated on figure 1.



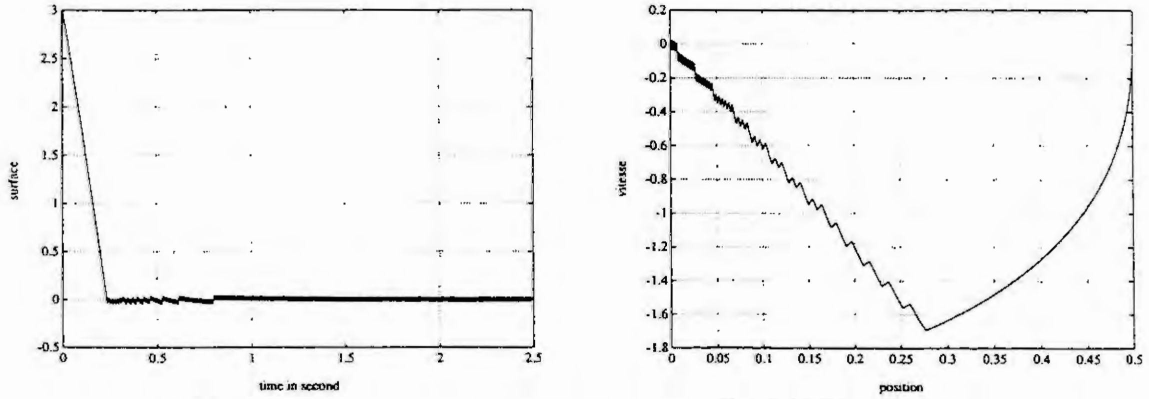
a) s(t) b) Plan de phase x_2=f(x_1)

Figure 1: Second order system - Surface: s_0=6 x_1

Choose now $s_1 = 6x_1 + x_2$ which is relative degree one. The condition

$$s_1 \dot{s}_1 = s_1(-3x_1 + 2x_2 + u) < 0$$

is satisfied, for sufficient high k when the control u=-k sign(s_1) is used. The result, which drastically contrasts with that of figure 1, is shown in figure 2.



a) s(t) b) Plan de phase x_2=f(x_1)

Figure 2: Second order system - surface: $s_1 = 6x_1 + x_2$

The above procedure has been successfully used in an experimental set up for the validation of a third order model of a system consisting of a robotic manipulator with artificial muscles [16]. The computation of the required phase variables was made by numerical differentiation.

References:

[1] M. FLIESS, Generalized controller canonical forms for linear and non linear dynamics, *IEEE Trans. AC, 35, 1990, p. 994-1001.*

[2] M. FLIESS, Generalisation non linéaire de la forme canonique de commande et linéarisation par bouclage, *C.R. Acad. Sci. Paris, t. 308, série I, p. 377-379, 1989.*

[3] M. FLIESS et F. MESSAGER, Sur la commande en régime glissant. *C.R. Acad. Sci. Paris, t. 313, Série I, p. 951-956, 1991.*

[4] H. SIRA-RAMIREZ, S. AHMAD & M. ZRIBI, Dynamical feedback control of robotic manipulators with joint Flexibility, *IEEE Transactions on systems man and cybernetics. Vol. 22, N° 4, pp. 736-747,1992.*

[5] F. MESSAGER, Sur la stabilisation discontinue des systémes. *Thesis, Orsay, n° 2034, Paris, 1992.*

[6] M. HAMERLAIN, Commande hiérarchisée à modèle de référence et à structure variable d'un robot manipulateur à muscles artificiels. *Thesis, INSA, n° 223, Toulouse, 1993.*

[7] A.S. NOURI et P. LOPEZ, Generalized variable structure control of two one-link manipulators in two operating modes, *IEEE-CDC, 1993.*

[8] V.I. UTKIN, Sliding mode in control optimisation, *Springer Verlag, Berlin, 1992.*

[9] J.J.E. SLOTINE et V. LI, Applied non linear control, *Prentice Hall, New Jersey, 1991.*

[10] C. MIRA, J.P. VERNHES, J. ERSCHLER, Extension du principe des systèmes à structure variable au cas où l'hypersurface de glissement est non linéaire, *RAIRO, mai, J2, 1972, p. 59-72.*

[11] A.S. NOURI, M. HAMERLAIN, C. MIRA, P. LOPEZ, Variable structure model reference adaptive control using only input and output measurements for two real one-link manipulators, *IEEE-SMC, Le Touquet, 1993.*

[12] G. CONTE, C. H. MOOG & A. PERDON, Un thèoreme sur la représentation entrée-sortie d'un systéme non linèaire, *C. R. Acad. Sci, Paris, t. 307, Série 1, p. 363-366, 1988.*

[13] M. FLIESS, J. LEVINE, P. MARTIN & P. ROUCHON, Sur les systèmes non linèaires diffèrentiellement plats, *C. R. Acad. Sci., Paris, série I, p. 619-624, 1992.*

[14] A.ISIDORI Nonlinear control systems, *Springer-Verlag, New York 1990.*

[15] H. SIRA-RAMIREZ, On sliding mode control of nonlineat systems, *Systems and control letters, Vol. 19, N°. 4, pp. 302-312, 1992*

[16] A.S. NOURI, *Thesis INSA, Toulouse, 1993 (sub-mitted).*

A VSS Theory Based Regulator for Robotic Systems with Fuzzy Parameter Adaptation

E. Condello, P. Muraca, C. Picardi, P. Pugliese
Dip. di Elettronica, Informatica e Sistemistica
Università della Calabria
87036 Rende - Italy

Abstract. Starting from a regulator for robotic systems recently proposed by two of the authors, and based on the theory of Variable Structure Systems, this paper proposes a fuzzy parameter adapter, which carries out the tuning of some parameters of the regulator according to the operating conditions. The proposed adapter is very simple, being characterized by a small number of decision rules; however, as shown by simulation results, it allows an improvement of the tracking performances and a substantial reduction of the chattering phenomenon.

1. INTRODUCTION

The sliding mode technique, derived from the theory of Variable Structure Systems (VSS), has been used by many researchers to control robotic arms. Young [8] and Slotine and Sastry [7] first introduced the sliding mode method in this field; however, their methods require the inversion of the inertia matrix.

Recently, Muraca and Pugliese [5] have proposed a new control law, still based on the sliding mode technique, which does not require either the inversion of the inertia matrix and the evaluation of its derivative, and is able to track the desired trajectory in a generalized coordinate space. This regulator compensates both system parameter uncertainties and external disturbances if some of its parameters have enough high values. On the other hand, this choice may involve excessive amplitudes of the control signals and accentuation of the "chattering" phenomenon.

In this paper we propose a self-tuning algorithm, which adapts those parameters of the regulator according to the operating conditions; the self-tuning algorithm is based on the theory of fuzzy sets. The use of this theory, as evident from the large number of recently published papers in this field, is strongly growing up to synthesize regulators for plants with partially or totally unknown mathematical models [1, 4]; recently, also adaptive controllers using fuzzy logic have been proposed [2, 3].

Presentation of the results of computer simulations carried out to verify the performance of the proposed regulator in controlling a robotic system concludes the work.

2. CONTROL LAW

The control law presented here is suitable to be applied to the class of second-order dynamical systems characterized by the vectorial differential equation

$$M(q)\ddot{q} + B(q,\dot{q})\dot{q} = w + u. \tag{1}$$

The robotic systems with rigid links fall in this class of models, where $q = q(t)$ and $\dot{q} = \dot{q}(t)$ are the generalized position and velocity vectors, respectively, belonging to $\mathbf{R}^n$, $M(q)$ is the (symmetric and positive definite) inertia matrix, $B(q,\dot{q})\dot{q}$ takes into account the Coriolis, centrifugal, gravity forces and the dynamic friction, $w(t)$ is a (deterministic) disturbances vector (such as static friction), and $u(t)$ is a generalized force vector, that is the control input.

Let $q_d(t)$ be the vector of reference trajectories in the generalized coordinates, for which it is assumed that $\| \ddot{q}_d(t) \|^2 \leq \gamma$: it describes the desired dynamic behaviour and represents the motion control specification. Moreover we assume that the state variables $q_i, \dot{q}_i$, $i = 1, \ldots, n$, are accessible for measurement; the tracking error is defined, componentwise, by $e_i(t) = q_i(t) - q_{d_i}(t)$.

The control problem consists in finding $u(t)$ such that $q(t)$ tracks $q_d(t)$, assuming bounded disturbances and bounded model misfits.

In a previous paper by Muraca and Pugliese [5], a control law based on VSS theory is proposed for the above system; this uses a switching surface $s_i(t) = e_i(t) + c_i\dot{e}_i(t)$ for each phase plain of the error variables $(e_i, \dot{e}_i)$, and defines the input function as

$$u(t) = B(q, \dot{q})\,\dot{q} + M(q)\,[\,\ddot{q}_d - C\,\dot{e} - \Lambda\, S(t) - f(S)\,]\,, \tag{2}$$

where

$$S(t) = e(t) + C\,\dot{e}(t)\,, \tag{3}$$

$$f(S) = \begin{bmatrix} \sigma_1\, sgn(s_1) \\ \sigma_2\, sgn(s_2) \\ \vdots \\ \sigma_n\, sgn(s_n) \end{bmatrix} \qquad sgn(\zeta) = \begin{cases} 1 & \zeta > 0 \\ 0 & \zeta = 0 \\ -1 & \zeta < 0 \end{cases},$$

and $\Lambda = diag(\lambda_i), C = diag(c_i), i = 1, \ldots, n$; under the hypotheses that $\lambda_i > 0$. $\sigma_i > 0$, this regulator leads the system to track the desired trajectory; in fact, by substitution of the control law into the (undisturbed) model equation we have

$$\dot{S(t)} + \Lambda\, S(t) = -f(S(t)); \tag{4}$$

a Lyapunoff function for the above (nonlinear) system is given by $V(t) = \frac{1}{2}||\,S(t)\,||^2$ and, under the hypotheses on λ_i and σ_i, it results

$$\frac{d}{dt}\,V(S(t)) = \sum_{i=1}^{n} \dot{s}_i\, s_i = -\sum_{i=1}^{n} (\lambda_i s_i^2 + \sigma_i \mid s_i \mid) < 0.$$

Moreover, the proposed regulator doesn't require the inversion of the inertia matrix $M(q)$ and succeeds in compensating parameter uncertainties and external disturbances; as this point is concerned, in [5] Muraca and Pugliese have given a way to select parameters σ_i, based on the maximum values assumed for the uncertainties on the plant parameters and the amplitudes of external disturbances: the higher are the values of σ_i, the larger are plant parameter variations and external disturbances that can be compensated. However, high values of σ_i lead to excessive amplitudes of the control signals and to the well-know phenomenon of chattering, *i.e.* a high frequency motion about the desired trajectory.

3. SELF-TUNING ALGORITHM

This section describes, for the above shown regulator, a self-tuning algorithm based on the fuzzy set theory, which adapts parameters σ_i at the operating conditions. Without going into detailed descriptions about fuzzy sets and fuzzy control, which can be found in [4], we only point out that the essential part of a fuzzy controller is a set of linguistic control rules related by the dual concepts of fuzzy implication and the compositional rule of inference. In essence, a fuzzy controller provides an algorithm which converts a linguistic control strategy based on expert knowledge into an automatic decision strategy.

First of all, the proposed "fuzzy adapter" can be considered as constituted by n fuzzy adapters, where the *i-th* of them provides the tuned value of the parameter σ_i. From the assumption that the state variables are accessible for measurement, the input variables of the *i-th* fuzzy adapter are chosen equal to the magnitude of the *i-th* position tracking error and the *i-th* velocity tracking error. A block diagram of the fuzzy adapter for each parameter σ_i is shown in Figure 1, where $\alpha_i = |\,e_i\,|$ and $\beta_i = \dot{e}_i$.

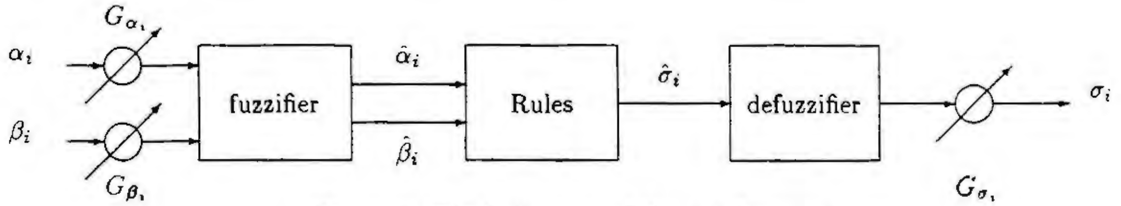


Figure 1. Block diagram of the *i-th* fuzzy adapter.

The first block, denoted as "fuzzifier", takes as inputs variables α_i and β_i scaled by the gains G_{α_i} and G_{β_i}, respectively. It realizes the fuzzification process that provides the degrees of membership to define the linguistic variables in the decision rules. The fuzzification algorithm selected for the inputs is represented by the membership functions $\mu(\alpha_i)$ and $\mu(\beta_i)$ reported in Figure 2.

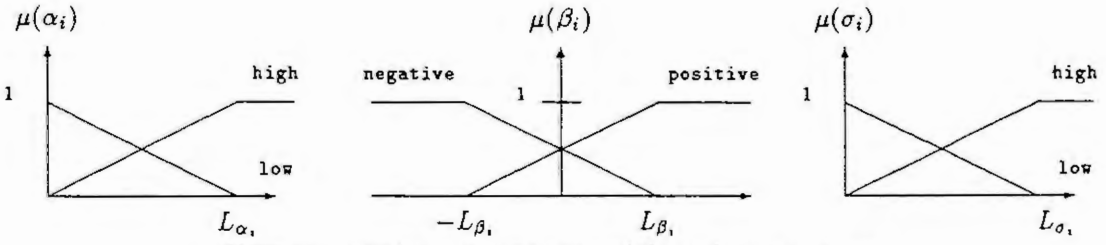


Figure 2. Membership functions of the *i-th* fuzzy adapter.

In particular, we have two membership functions $\mu(\alpha_i)$ associated to the fuzzy sets "high α_i" and "low α_i", with a same universe of discorse equal to the interval $[0, L_{\alpha_i}]$; analogously there are two membership functions $\mu(\beta_i)$ associated to the fuzzy sets "positive β_i" and "negative β_i", with an universe of discorse equal to the interval $[-L_{\beta_i}, L_{\beta_i}]$.

Figure 2 also reports the membership functions $\mu(\sigma_i)$ of the (output) fuzzy sets "high σ_i" and "low σ_i", with an universe of discorse equal to $[0, L_{\sigma_i}]$. Actually, since the degrees of membership of the output fuzzy sets are computed from the decision rules, no fuzzification of adapter output σ_i would be necessary; however, the definition of the functions $\mu(\sigma_i)$ needs for the defuzzification.

From the previous considerations, the fuzzified variable $\hat{\alpha}_i$ consists of two elements representing the degrees of membership of α_i in the fuzzy sets "high α_i" and "low α_i" and the fuzzified variable $\hat{\beta}_i$ also consists of two elements representing the degrees of membership of β_i in the fuzzy sets "positive β_i" and "negative β_i".

In consequence of the input signals α_i and β_i and their associated linguistic values "high" and "low" and "positive" and "negative" respectively, we have the set of the decision rules reported in Table 1.

	$\alpha_i =$ high	$\alpha_i =$ low
$\beta_i =$ positive	$\sigma_i =$ high	$\sigma_i =$ low
$\beta_i =$ negative	$\sigma_i =$ high	$\sigma_i =$ low

Table 1. Decision rules for the fuzzy adapter.

By using the classic logic about the combination of fuzzy sets, these rules provide the fuzzy variable $\hat{\sigma}_i$ which consists of the elements $\mathtt{h}_i$ and $\mathtt{l}_i$. The former element is the maximum value of the degree of membership of σ_i in the fuzzy set "high σ_i"; the second one is the maximum value of the degree of membership of σ_i in the fuzzy set "low σ_i".

Finally, the defuzzification process provides the actual value of parameter σ_i to be used in the control law. Taking into account the membership functions $\mu(\sigma_i)$, this value is given by the simple expression:

$$\sigma_i = \frac{L_{\sigma_i}\,\mathtt{h}_i}{\mathtt{l}_i + \mathtt{h}_i}\,G_{\sigma_i}, \tag{5}$$

where G_{σ_i} is a suitable gain for parameter σ_i.

4. SIMULATION RESULTS

In order to evaluate the performances of the proposed VSS regulator using the fuzzy adapter described in the previous section, we have considered the same example already used in [5, 7], that is the control of the planar two-link robot represented in Figure 3; the robot has rigid links of lengths l_1 and l_2 and masses m_1 and m_2, and a payload of mass ρ is present at the end of the second link.

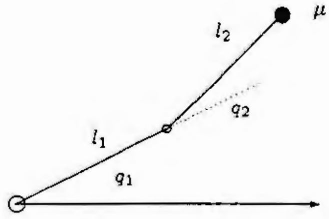


Figure 3. Two-links robotic system.

The differential equation describing the dynamic of this system can be arranged according to equation (1) with $n = 2$. The control input $u(t)$ is constituted by torques T_1 and T_2 applied at the joints of the links. For the numerical simulations we have used the following parameters of the robot: $l_1 = l_2 = 1$ m, $m_1 = m_2 = 1$ Kg. Moreover, a payload $\rho = 0$ Kg has been regarded as the nominal payload, while a value $\rho = 0.25$ Kg has been considered as a perturbed condition. The analytic expressions of the desired trajectories are the following, and their plots are given in Figure 6:

$$q_{1d} = \begin{cases} -90^\circ + 52.5^\circ\,(1-\cos(1.26\,t)) & t \leq 2.5\,sec. \\ 15^\circ & t > 2.5\,sec. \end{cases}$$

$$q_{2d} = \begin{cases} 170^\circ - 60^\circ\,(1-\cos(1.26\,t)) & t \leq 2.5\,sec. \\ 50^\circ & t > 2.5\,sec. \end{cases}$$

For the regulator described by equations (2,3) we have chosen $c_1 = c_2 = 5$, $\lambda_1 = 7$, $\lambda_2 = 10$, while parameters σ_1 and σ_2 have been obtained by two fuzzy adapters of the type described above, with $L_{\alpha_1} = L_{\alpha_2} = 0.05$, $L_{\beta_1} = L_{\beta_2} = 0.1$, $L_{\sigma_1} = L_{\sigma_2} = 3$, $G_{\alpha_1} = G_{\alpha_2} = 10$, $G_{\beta_1} = G_{\beta_2} = 2$, $G_{\sigma_1} = G_{\sigma_2} = 1.4$. Finally, initial values equal to zero for the two parameters σ_i, a final time $T_f = 5$ sec. and a discretization time equal to 0.02 sec. have been selected for computer simulations.

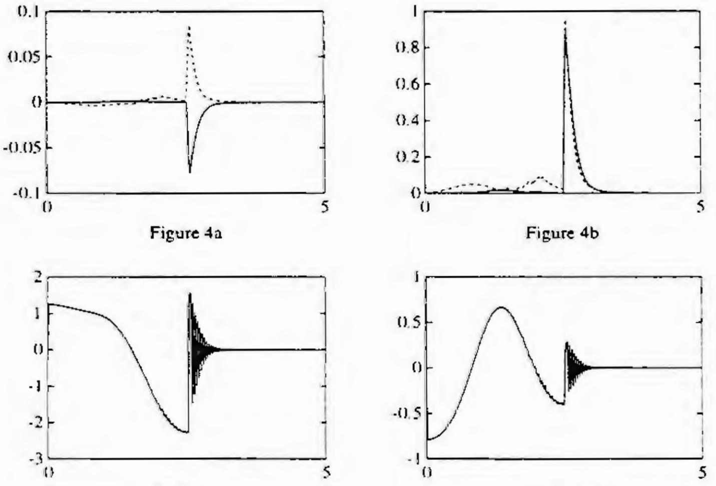


Figure 4a

Figure 4b

Figure 4c

Figure 4d

Figure 4. Solid lines are relative to variables with index 1, dashed lines to those with index 2.

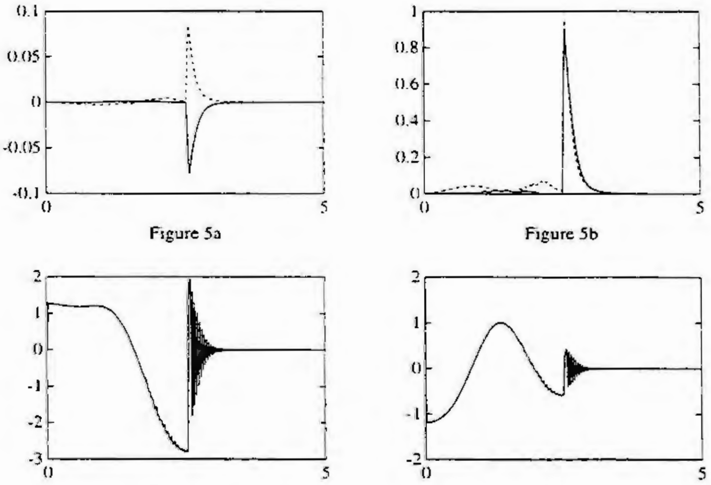


Figure 5a

Figure 5b

Figure 5c

Figure 5d

Figure 5. Solid lines are relative to variables with index 1, dashed lines to those with index 2.

Figure 4 reports, for the nominal case, the time plots of the most significant variables; in particular Figure 4a shows the behaviour of σ_1 and σ_2 and Figure 4b reports the tracking errors e_1 and e_2; Figures

4c and 4d show torques T_1 and T_2, respectively. Figure 5 reports the time plots of the same variables referred to the perturbate case. In both figures, angles are given in degrees and torques are given in Newton per meters.

From these results we observe the substantial reduction of the chattering phenomenon on the torques with respect to the use of constant values of parameters σ_i (comparative analyses are shown in [6]). Moreover, the similar behaviour of the tracking errors both in the nominal case and in the perturbate one, confirms the robustness of the proposed regulator; finally from Figures 4a and 5a we point out how high values of parameters σ_1 and σ_2 are required only when the tracking errors are high.

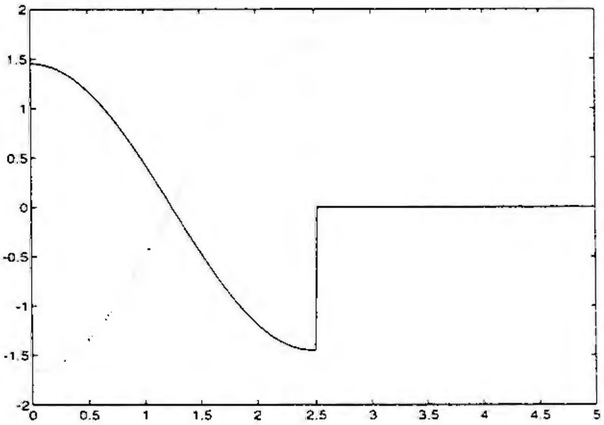


Figure 6. Time plot of q_{d_1} (solid line) and q_{d_2} (dotted line).

5. CONCLUSIONS

The paper has presented an application of the fuzzy set theory to the real-time adaptation of some of the parameters of a VSS regulator for robotic systems, recently presented in the literature by two of the authors. The introduction of a self-tuning algorithm for those parameters overcomes the need to select for them high constant values at the operating conditions, which would lead to an amplification of the chattering.

Moreover, the proposed fuzzy adapter is a very simple device, being based on a very small number of decision rules. The next aim of the authors is towards the study of the stability of the whole closed-loop system so as to theoretically justify the results given by simulations.

6. REFERENCES

[1] Carotenuto, L., P. Muraca, C. Picardi, N. Rogano, A fuzzy rule-based controller for the dynamic control of a two-link robot arm. *Proc. of the Int. Symposium on Intelligent Robotics*, Bangalore, India, January 1993.

[2] Hong, H. P. *et. al.*, A design of auto-tuning PID controller using Fuzzy logic. *Proc. of the IECON '92 Conference*, San Diego, USA, November 1992.

[3] Kyriakides, K. and A. Tzes, Adaptive Fuzzy dominant-pole placement control. *Proceedings of the 31st IEEE Conference on Decision and Control*, pp. 2517–2522, Tucson, Arizona, December 1992.

[4] Lee, C. C., Fuzzy logic in control systems: Fuzzy logic controller – part I and II. *IEEE Trans. on SMC*, vol. SMC-20, n. 2, 1990, pp. 404–435.

[5] Muraca, P. and P. Pugliese, Application of variable structure system theory to a mechanical system. *Proceedings of the 31st IEEE Conference on Decision and Control*, pp. 3550–3553, Tucson, Arizona, December 1992.

[6] Muraca, P., C. Picardi, P. Pugliese, A modified Variable Structure Regulator for a Class of Mechanical Systems. Technical Report, D.E.I.S. - Università della Calabria, August 1993.

[7] Slotine, J. J. and S. S. Sastry, Tracking control of nonlinear systems using sliding surface with applications to robot manipulators. *Int. Journal of Control*, vol. 33, n. 2, 1983, pp. 465–492.

[8] Young, K. K. D., Controller design for a manipulator using theory of variable structure systems. *IEEE Trans. on SMC* , vol. SMC-8, n. 2, 1978, pp. 251–259.

General mathematical modelling of kinematic behaviour of robots having supple bodies

P. ANDRE professor *, **J-P. TAILLARD** professor *
P. MITROUCHEV PHDoctor Lecturer**

* ENSMM La Bouloie. Route de Gray. 25030 BESANÇON Cedex. FRANCE, tél. + 81 82 26 07
L.A.B. Centre " Microsystèmes et Robotique " Laboratoire d'Automatique de Besançon. 15, Impasse des Saint Martin, 25000 BESANÇON Cedex. FRANCE. tél. + 81 88 65 44, Fax + 81 88 65 02
** Université "Joseph Fourier" I.U.T.-1 de Grenoble, Département Génie Mécanique et Productique, B.P. 67, SAINT-MARTIN-D'HERES Cedex. FRANCE. tél. + 76 82 53 94, FAX + 76 82 53 26

Abstract :

Within the framework of the study of robot-control efficiency, we have studied the geometric and kinematic models of supple robots. We have carried out the modelling, a set of programs allowing the automatic generation of the kinematic symbolic model of open-chain robots having supple bodies, and their integration into the EUCLID-IS CAD system. In this paper we present a mathematical method for the kinematic modelling of open-loop chain articulated robots, taking into account the displacements due to the elastic deformations of the links constituting the robot and the programs allowing the automatic generation of the geometric and kinematic symbolic models.

The elastic deformations of the mechanical structure lead to errors that must be taken into account when defining the control law.

The suppleness of the robot can have two different origins :

- the links can exhibit deformations due to the applied efforts having either a dynamic (inertial, centrifugal...) or a mechanical (forces, torques...) origin, or both,
- the joints can exhibit elasticity due to transmissions.

The aim of the this paper is to describe how we propose to take into account the elastic displacements of the robot's structure in the kinematic modelling of the robot. This kinematic modelling can be considered from four view points :

- the speed of the nodes
- the end-effector's speed
- the speed of any intermediate point
- the various instantaneous rotation speeds

We will assume that : (hypotheses)

1) the joints are perfect ;
2) the bodies are suppleand and long enough to be modelized by making use of the Euler-Bernoulli bidimensional beams ;
3) the deformations are small and they stay within the limits of the material's elastic behaviour ;
4) the elastic behaviour of the material is linear ;
5) the rigid displacements of the bodies can have any amplitude.

Method :

The structure of the robot is described by using the Denavit-Hartenberg parameters. Unlike existing methods, our approach is based on developing matrices and vectors along Taylor-McLaurin. For kinematic modelling we limit the developments to the first order ; so that each matrix (resp. vector) is written as the sum of two terms:

- one main term representing the mouvement of rigid bodies,
- another term representing the small displacements due to suppleness, which is a corrective term as compared with the previous one.

The string development of the $[T_i^{i+1}]$ transformation matrix along Taylor-McLaurin gives :

$$[T_i^{i+1}]= [T_i^{i+1}r] + [T_i^{i+1}s] +.....$$

with :

$[T_i^{i+1}r]$ - being the transformation matrix in the case of rigid movement.

$[T_i^{i+1}s]$ -being the supplementary transformation matrix due to the elastic (supple) displacements ; it can be viewed as a corrective term as compared with the previous one.

The nstantaneous rotation vector is represented as the sum of the two terms :

$$\underline{\Omega}_{i+1}=\underline{\theta}_{i+1}+\underline{\phi}_i$$

with :

$\underline{\theta}_{i+1}$ the generalized rotary coordinate for rigid movement ;

$\underline{\phi}_i$ the angular elastic displacement of S_i , it represents the rotation between the two extremities of the beam.

As a result, the kinematic model is :

$$\underline{T}_c=J_c(q)\underline{q}$$

with :

$\underline{T}_c$ -kinematic torsor, of the S_n body taking into account R_n frame and expressed in R_o frame.

Jc(q) - kinematic Jacobiane matrix in dimension (mxn)

$\underline{q}=[\underline{q}_r^T \quad \underline{q}_s^T]^T$

$\underline{q}_r$ - the generalized vector speed for rigid movement,

$\underline{q}_s$ - the generalized vector speed for elastic movement,

m degree of freedom of the task,

$n=n_r+n_s$ - total number of the mobility freedom for rigid movement (solid) and elastic displacement.

☞ We give the analytical expressions that we have obtained. The results are applied to various planar -R, RR (SCARA- like) and PPR-architectures.

☞ We have realized a set of programs allowing the automatic generation of the kinematic symbolic model of open-chain robots having supple segments in VAX-C language.

☞ The software modelling tools that we have developed are under integration into the "EUCLID-IS" CAD system running under VAX-VMS and under SGI-UNIX.

Keywords :

Robotics ; Mathematical kinematic modelling ; Supple robots ; Automatic generation ; CAD system

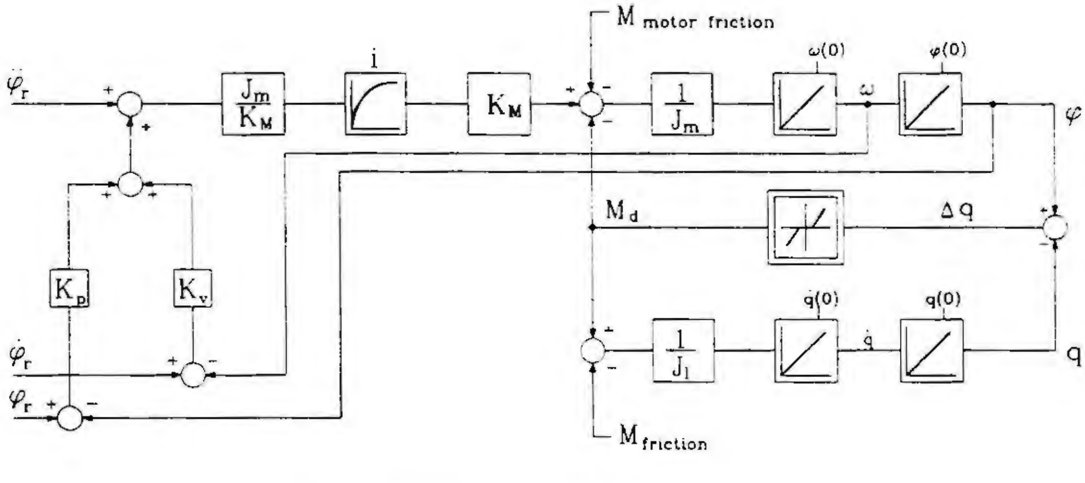
NONLINEAR STATIC CHARACTERISTIC OF THE ROBOT JOINT GEAR

Suzana URAN, Karel JEZERNIK
Faculty of Technical Sciences Maribor
University of Maribor
Smetanova 17, 62000 Maribor, Slovenia

Abstract. In the case of robots with high gear ratios in the robot joints the problem appears when the actual values of the elasticity, backlash and friction of the robot joint are needed. In the paper an approach is presented, which exploits a known part of the robot joint mathematical model for the estimation of torques. The validation of the implemented approach by simulations was succesfull and is represented.

1. INTRODUCTION

A joint of a geared electrical robot consists of an electrical motor and a gear. Gear ratios used are high, it is in the range from 50 up to 100 and more. The most frequently used gears for robots are harmonic drives. Therefore the model of a robot joint with harmonic drive is the best known [1,2]. The gears used in our robotic laboratory machanism are a special type excenter gears called AKIM. Previous works of [1,2,3] have shown that the geared robot joint is satisfactorly modelled by a two mass model with a spring, backlash and friction. Therefore for our modelling approach the same model was used. The mathematical model used for the controlled robot joint is represented with a block scheme in Fig.1.



Slika 1: Mathematical model of the robot joint

The nonlinear static characteristic $M_d = f(\varphi - q)$ in the model represents a model of a joint backlash and elasticity. The upper - controlled part of the model (for the first inertia J_m) represents the model of the permanent magnet DC motor with current control and a well known position and speed controller. The lower part of the model (for the second inertia J_l) represents the model of the robot link, where only the actual robot joint could be moved, while the rest of them are blocked. All the values of the mathematical model should be considered on the motor side of the gear.

The motor inertia J_m and the torque constant K_m of the robot joint model are known from the motor manufacturer's data. Therefore on the basis of the motor current and speed measurements the torque M_d with motor friction could be estimated from the motor part of the robot joint model.

2. ESTIMATION OF TORQUE

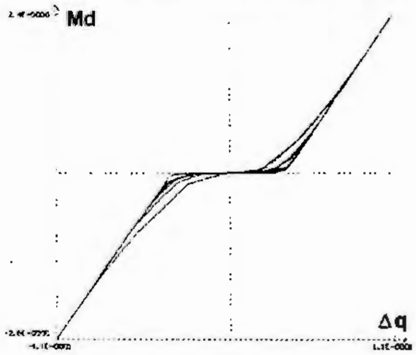
The estimation of torque was performed on the basis of the so called disturbance observer. It is the torque M_d and the motor friction are considered as the disturbance in the motor part of the robot joint model. The disturbance observer was derived and designed in the discrete form. In the derivation of the disturbance observer was adopted that the dynamics of the motor current control loop is so high that the actual motor current values could be replaced by the reference motor current values. Therefore the simplified state space model of the current controlled motor with the disturbance model was transformed into discrete form by the method of the step invariance. The disturbance M_d was modelled as a constant. The part of the state space vector ω is measured, therefore for the disturbance estimation only a reduced order observer should be derived. It was derived by a standard procedure for reduced order observers [4] and is given by equation 2.

$$\begin{gathered} \mu(K+1) = (1 + h.\tfrac{T}{J_m}).\mu(K) - h.\tfrac{K_M.T}{J_m}.i_{ar}(K) + h^2.\tfrac{T}{J_m}.\dot{q}(K) \\ \widehat{M_d}(K) = \mu(K) + h.\dot{q}(K) \end{gathered} \tag{1}$$

The nonlinear static characteristic of the robot joint gear is obtained through the estimated disturbance M_d and the two measured positions (the motor position φ and the joint position q) according to the robot joint model. The result of the approach implemented in the laboratory would be a hysteresis due to frictions existing in the robot joint instead of the characteristic from Fig.1. A verification of the approach in the laboratory is impossible due to unknown elasticity and backlash of the actual robot joint and while no suitable testbed exists. Therefore a verification of the approach by simulations is the most suitable.

3. VERIFICATION OF THE APPROACH BY SIMULATIONS

The model from Fig.1 without friction and with the disturbance observer were build in the simulation program PADSIM. Simulations were performed for a case of PTP joint movement with smoothed (sin^2) accelerations, chosen values of the joint gear elasticity and backlash and chosen parameter of the disturbance observer. In order to verify the approach two nonlinear static characteristics of the robot joint gear were compared. The first was a result of the actual disturbance M_d obtained for the chosen elasticity and backlash and the second was a result of the disturbance estimated by the observer. Verification of the approach has shown that in order to get proper characteristic measured positions (φ and q) should be filtered by the first order filter. The parameter of the filter should be equal to the disturbance observer parameter. With additional filtering of measured positions and the estimated disturbance a gear characteristics represented in Fig.2 was obtained.



Slika 2: Nonlinear static characteristics of the robot joint gear

4. RESULTS

Verification of the approach (Fig.2) shows that with additional filtering of measured positions an accurate nonlinear static characteristics of the robot joint gear could be obtained.

REFERENCES

[1] Sweet, L.M., Redefinition of the Robot Motion-Control Problem, IEEE Contr. Syst. Mag.(1985),Aug. 85,18-25.

[2] Legnani, G., Harmonic drive transmissions: the effects of elast., clear. and irregu. etc., Robotica (1992), 368-375.

[3] Schäfer, U., Posit. contr. for elast. pointing and track. syst. with gear play and Coul. frict. and applic. to Robots, SYROCO Vienna '91.

[4] Föllinger, O., Regelungstechnik, Huthig Verlag 1985.

Trajectory Planning, Dynamic Modelling and Robust Control of Articulated Robot Arm

Vesna Laci, Zdenko Kovačić
Faculty of Electrical Engineering, University of Zagreb
Unska 3, 41000 Zagreb, Croatia

Abstract. The paper presents an object-oriented software package developed for educational purposes in the field of robotics. An emphasis has been given to an improved method of trajectory planning and usage of object-oriented programming technique in dynamic modelling of robots.

1. Trajectory Planning

A problem of planning trajectories for different tools has to be solved precisely in order to perform meaningful manipulation tasks. This problem is connected with a mathematical model of robot arm kinematics. If knot points along the path such as the end points and intermediate via-points are specified (these points must be reached by the tool of the robot), segments of the path between them have to be interpolated to produce a smooth trajectory which can be executed using continuous-path motion control technique. The path between the knot points in tool-configuration space should be smooth, i.e. the spline functions used for interpolation of the path should have at least two continuous derivations in order to avoid infinite acceleration. Therefore, the third order polynomials are used to interpolate intermediate segments of the path and the fourth order polynomials are used for interpolation of the end segments. Adjacent spline segments can be connected by following the requirement that the acceleration should be continuous at the segment boundaries. An accurate value of the traverse time for each segment can be calculated by multiplying the presumed traverse time and factor S which is defined in [3]. The described algorithm has to be iteratively repeated until maximum of any acceleration or velocity is reached.

This algorithm was verified by computer simulation and the results given in Figs. 1 and 2 show that the improved iterative method for trajectory planning in tool-configuration space is more accurate than the method proposed in [3].

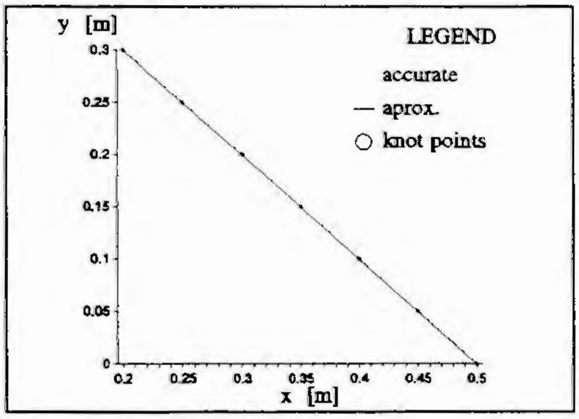


Fig. 1. Trajectory planning in tool-configuration space.

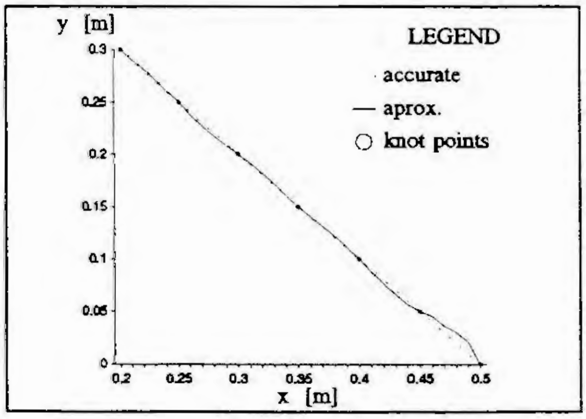


Fig. 2. Trajectory planning in joint space.

2. Dynamic Modelling

Precise control of robot motion requires a realistic dynamic model of the arm which can be derived by means of the Lagrange-Euler method and the recursive Newton-Euler technique.

The Lagrange-Euler approach is based on the concept of generalized coordinates, energy and generalized force. It contains a frictional force model which includes viscous, dynamic and static friction.

An alternative approach, called the recursive Newton-Euler formulation, that includes the forward and the backward equations is presented in [4]. If a joint-space trajectory is given, the velocities and accelerations of each link are recursively calculated, starting at the base and computing forward to the tool (forward equations). These informations are used to compute the forces and moments acting on each link, starting at the tool and calculating backward to the base (backward equations). This method is much more suitable for computer implementation.

Both mentioned methods give the same dynamic model which is derived for the five-axis articulated robot arm Rhino XR-3 and the three-axis articulated planar robot. The dynamic models were tested by computer simulation and the results indicate that the joint angle and speed of each robot joint can be controlled independently. Namely, joints hardly interfere to each other thus making the robot arms suitable for controlled continuous-path motion.

3. Application of object-oriented programming technique

The proposed method for trajectory planning, together with kinematic and dynamic models of a three-axis and a five-axis articulated robotic manipulators and several control techniques (PID, fuzzy, adaptive and robust) are included into an object-oriented software package. This software is made owing to the main features of object-oriented programming and the C++ language presented in [1], [2] and [5], so this program is very easy to use and modify.

4. Conclusions

The described method for trajectory planning ensures accurate approximation of the desired path and includes limits of the velocities and accelerations of each robot joint. The results obtained by computer simulation using realistic dynamic models of the three-axis and five-axis articulated robotic manipulators prove that in both cases each robot joint can be controlled independently. The developed object-oriented software package can be used for educational purposes in the field of robotics.

5. References

[1] Jordan, D., Implementation Benefits of C++ Language Mechanisms, Communications of the ACM, vol. 33, no. 9, 1990, 61-64.

[2] Korson, T., and McGregor, J. D., Understanding Object-Oriented: A Unifying Paradigm, Communications of the ACM, vol. 33, no. 9, 1990, 40-60.

[3] Ránky, P. G., and Ho, C. Y., Robot Modelling: Control and Applications with Software, IFS (Publications) Ltd. & Springer Verlag, 1985.

[4] Schilling, R. J., Fundamentals of Robotics: Analysis and Control, Prentice-Hall, Englewood Cliffs, New Jersey, 1990.

[5] Stroustrup, B., The C++ Programming Language, Addison-Wesley Publishing Company, Reading, 1987.

SENSORIAL INTEGRATION FOR NAVIGATION AND CONTROL OF AN AUTONOMOUS MOBILE ROBOT

A.Traça de Almeida, Helder Araújo, Jorge Dias, and Urbano Nunes
Electrical Engineering Department, University of Coimbra,
3000 Coimbra, PORTUGAL

Abstract. Available sensors for robot navigation are unreliable and noisy. Therefore there is a need to simultaneously employ different types of sensors to acquire the information required for navigation. In this paper different problems associated with the integration of several sensors in a mobile platform are discussed. The full exploitation of the data provided by the sensors is difficult, requiring good sensor models. In this work we dealt with integration of data given by inertial sensors, odometry, sonars and an active vision system.

1. SENSOR-BASED NAVIGATION

A mobile robot is supposed to move between positions in an environment. To accomplish such a task it must be able to locate itself in the environment and must also be able to follow a path. These two functions, localization and path following, are the basic building blocks of what is usually called navigation. For that purpose, sensors are essential for extracting information of some structures of the environment. The sensing process is essentially the acquisition of some physical measures. From that data the interesting features are extracted, constituting the perception of the environment. The role of extracting data from the environment is played by sensors. Therefore it is crucial that sensors be chosen according to the expected types of measures. In the case of navigation, sensors should provide information that enables the localization of the vehicle, obstacle detection and/or avoidance. Navigation by itself is well understood. Difficulties arise in getting the suitable data since there is a lack of adequate sensors and the existing sensors are unreliable and inaccurate. For navigation we need sensors that measure distances, velocities, accelerations and orientations. These physical quantities determine the type of sensors that have to be employed.

Two broad classes of navigation sensors can be considered, one of them used with *dead-reckoning* navigation (usually called internal sensors) and the other used with *reference guidance navigation* (including all the sensors whose measures require external references). *Dead-reckoning* refers to navigation with respect to a coordinate frame that is an integral part of the guidance equipment. Dead-reckoning has the advantage that it is totally self-contained. However, dead reckoning suffers from several sources of inaccuracy. One of the main sources of inaccuracies is the cumulative nature of the errors whose sources include wheel slippage, and terrain irregularities. To reduce the problems with dead reckoning, we can use landmarks or beacons along the trajectory. The localization of the mobile robot can be corrected by sensing these landmarks and beacons. *Reference guidance* has the advantage that the position errors are bounded but the detection of external references or landmarks and real-time position fixing, may not always be possible. However reference guidance has the disadvantage of reducing the degree of autonomy of the vehicle by making it dependent of a certain environment. As a matter of fact in this type of navigation the vehicle must rely on a map to navigate from one beacon to the next and also to determine its location. To achieve a good accuracy and flexibility different types of sensors must be used. These sensors include vision, sonar, laser range finders and other sensors.

2. EMPLOYED SENSORS

Dead reckoning is usually performed by odometry and/or inertial guidance sensors. Odometry consists on the measurement of distance by means of the integration of the number of turns of a wheel. For *reference guidance* it is necessary to measure distances. For that purpose a wide variety of range sensors can be used. These sensors can be divided into two classes: active and passive ones. Active sensing computes the distance between the sensor and an object by observing the reflection of a reference signal produced by the object. Generally active sensing methods provide range data in a direct way without a significant computational cost (unlike what happens in most of the passive methods, e.g., stereo vision). In the first class ultrasonic and optical range sensors are the most commonly used.

The displacements of the robot can be classified into two classes: *intrinsic* or *extrinsic* to the movement commands. The intrinsic displacement is due to the action of robot actuators and the consequent displacement is partially predictable from the commands. The displacements of the robot not directly due to its actuators are referred to as *extrinsic* displacements. In this case, the inertial sensors, in cooperation with other sensors, can be used to estimate this type of displacements. The inertial sensors' information could be combined with odometric information to compute a better estimate of the actual displacement of the robot. This estimation process can also be useful on the calibration of the odometric and inertial devices. This combination is performed by using the redundancy of the different sources of information.

The information obtained from inertial system can be used for different tasks such as navigation, trajectory generation or stabilization of the robot. The available inertial sensors are of two kinds: linear accelerometers and gyrometers or gyroscopes. The angular velocity can be measured by gyrometers. The gyrometers give an output with the instantaneous angular velocity, and the gyroscope gives an output with the integration of the angular velocity during a period of time. Angular position information can be computed from gyrometers' output by integration, while angular velocity information can be computed from gyroscope output by derivation.

Considering our goal of implementing the navigation system for a mobile platform (Robuter) [9], we will concentrate on the issues related with sensor integration. Taking into account the basic features, availability, and cost of the sensors generally employed on robotics navigation systems, we decided to base our navigation system on odometry, sonar, vision and inertial sensors (accelerometers and gyrometers).

3. SOME METHODOLOGIES FOR SENSOR INTEGRATION

The data obtained by a sensor are always corrupted by noise, sometimes with spurious or completely incorrect values. In practice, it is very important to explicitly manipulate the uncertainty of the measurements in order to effectively use the information provided by a sensor. The correct modeling of the sensors' data is only part of the problem of sensor integration in mobile robots navigation. Another aspect is the multisensor data fusion, which seeks to combine data from multiple sensors to perform inferences that cannot be possible from a single sensor alone. This includes the improvement of the accuracy of inferences such as the position of the mobile robot by using multiple sensors. In this point we propose an architecture where the data provided by odometers, inertial system, sonar and an active vision system are combined to be used by a mobile robot navigation system.

We will consider sensor integration within the framework of navigation in a partially known structured environment where new obstacles can show up and some obstacles can be removed. Therefore all data integration processes will be based on the previous knowledge of a partial environment map, even though this map may undergo a restricted set of changes. The methodologies for data integration will rely on the sensor models.

3.1.Data integration - Intrinsic sensor level (Dead-Reckoning level)

In this level we propose the integration of the data from the odometer, the accelerometers and gyrometer to estimate localization of the mobile platform. From the accelerometers and gyrometer we extract estimates of the vehicle linear and angular velocities. These two sensors are the basic components of the inertial system. The inertial system forms a rigid body made up of two accelerometers and one gyrometer with each sensor mechanically solidary with an axis of an orthogonal system.

Each accelerometer measures the linear acceleration along its axis and the gyrometer measures the angular velocity about z-axis (vertical axis). Since we consider that our vehicle moves on a plane, its trajectory can be defined two-dimensionally. In this case the linear acceleration of the motion has only two components defined by the vector $\vec{a} = a_x\hat{\mathrm{i}} + a_y\hat{\mathrm{j}}$ and the angular velocity is defined by the one-dimensional vector $\vec{\omega}$. Moreover, the sensor model has to take into account that the exact location and orientation of each one of the sensors is not precisely known. The linear velocity is obtained by integrating the outputs of the accelerometers, and the orientation is obtained by integrating the angular velocity. By combining the odometry data and inertial system data we obtain a first estimate of vehicle's position and orientation. We are also considering the integration of velocity and orientation estimates obtained via vision. For that purpose we will use an active vision head mounted on the platform [3]. With such a system it is possible in many circumstances to compute velocities and/or orientations. By actively controlling the 3-D relative position of two cameras it is possible in some circumstances, to have estimates of vehicle's egomotion. Even with a single camera it is possible to have estimates of the translational and rotational velocities provided that a certain number of assumptions are verified [10]. Even though a general solution for determining the 3-D egomotion of a robot does not exist as yet, our case is a much simpler one; indeed our robot will be navigating in 2-D. In the case where we wish to compute the rotational component of the robot egomotion the availability of an active stereo head enables such a computation to be performed in a relatively simple way. For that purpose a landmark on the environment is fixated. Fixation is the process by which the image of a 3D feature is kept unchanged in spite of the vehicle's motion [1], [Ballard 89]. A critical element in this process is feature determination. Features to be used are line segments, corners and intersections. These features can be characterized both two-dimensionally and three-dimensionally. Even though fixation on a 3D feature does not provide a significant improvement on the estimates of the rotational angle, it allows a more precise determination of the distance. For feature determination we will not use all the available resolution of the image. Images will be subsampled in the regions off the center [5], [8]. This permits feature tracking on almost real-time .

3.2 Data integration - Extrinsic sensor level (Reference guidance)

The data provided by the sonar must be combined with the estimate of the vehicle's position and orientation obtained by the integration process explained above. Different approaches for vehicle's location estimation are reported in the literature but in our problem we will consider two approaches. Both have proved to be applicable on actual platforms navigating in a structured environment. One of the approaches was presented by John Leonard and Durrant-Whyte [6], [7] and uses EKF (Extended Kalman Filter) to match beacon observations to a navigation map to maintain an estimate of vehicle's location. The other approach was presented by Ingemar Cox [2] and is used on the autonomous robot Blanche developed at the AT&T Bell Laboratories.

The J.Leonard and Durrant-Whyte approach denotes the position and orientation of the vehicle at any step k by a state vector $\vec{x}(k)$ containing the position and orientation of the vehicle on a Cartesian reference with respect to a global coordinate frame. Additionally J.Leonard and Durrant-Whyte define a system state vector composed by not only the previous location state vector but also an environment state vector containing the set of geometric beacon locations. The robot starts at a known location, and it has an a priori map of the locations of the geometric beacons. The map used in this approach is just a set of beacon's locations and not an exhaustively detailed environment map. Each beacon is assumed to be precisely known. At each time stop, observations of these beacons are taken, and the goal of the system is to associate measurements taken from the beacons with the corresponding map beacon to compute an updated estimate of the vehicle's position. The use of the EKF on this approach relies on two models: a plant model that describes how the vehicle's position changes with time in response to a control input, and a measurement model that expresses a sensor observation in terms of the vehicle location and the geometry of the beacon.

In the other approach, presented by Ingmar Cox [2], for the mobile robot Blanche, the location estimation was developed to work in structured environments. A combination of odometry and optical range finder to sense the environment is used and a matching algorithm for the sensory data and a map was developed. The a priori map of the vehicle environment consists of a 2-D representation based on a collection of line segments. In addition an algorithm is used to estimate the precision of the corresponding match/correction which allows the correction to be optimally combined with the odometric location to provide an improved estimate of the robot's location. Since the robot's location is to be constantly updated as the vehicle moves, there is a need to remove any motion distortion that may arise. This can be done by reading the odometric position at each range sample. The current position and range data are then used to convert the data point into to world coordinates for matching to the map.

For a structured environment the algorithm proposed by I.J.Cox has proven to be effective. Indeed the entire autonomous vehicle is self-contained, all the processing being performed on-board. The algorithms enable the range data to be collected while the vehicle is moving. For those reasons the mobile platform Blanche represents a high-level of performance at low-cost.

On the other hand the J. Leonard and Durrant-Whyte algorithm requires much more computational effort and stops of the vehicle to acquire the range data.

Yet in both cases a dense range map is acquired by servo-mounted range sensors. Even though we have decided to employ an approach similar to I.J.Cox the different sensor arrangement in our vehicle prevents the immediate application of the algorithm. Indeed a ring of sensors disposed around a rectangularly shaped vehicle prevents many configurations of the vehicle from being detected by the sonar. The ring of sensors around the vehicle Robuter (set of 24 fixed sonars), will be used essentially for obstacle detection.

To integrate the estimate of the vehicle's position and orientation obtained from odometry and inertial system and matched position from sonar processing, we need to have estimates of standard deviations in the measurements of x, y and θ, for both the matcher and odometry. Given the locations and standard deviations from the matcher and the estimate provided by the first level of integration, the values of the location and its corresponding standard deviation can be updated. This updated value is fed back to the odometry where it is used as the new value from which a new location is estimated. Since the odometry is corrected after each matching, failure of the sonar or matching subsystem does not lead to immediate failure of the robot navigation system.

4. DISCUSSION

Mobile robot navigation depends heavily on the quality of sensor information. Therefore, when designing a navigation system one has to carefully consider the types of sensors to be used as well as the related issue of combining and integrating the information. Even for the simple case of robot localization in a known and structured environment there still are open issues when considering that the robot navigates in a real environment.

These considerations led us to restrict ourselves to the problem of navigation in a structured environment with a priori partial knowledge of its map. For that problem we decided to build our navigation system around a set of sensors that includes an inertial system, odometry, a sonar network, and one active vision system.

Data integration and fusion itselves have many issues that require deeper consideration and study. From our point of view some of these issues are sensor modeling, data association and estimation process. We are currently tackling these subjects within the framework of the development of a sensor system for a mobile platform.

5. REFERENCES

[1] Aloimonos, J., Bandopdhay, A. and Weiss, I., Active vision, *Proc. 1st Int. Conf. Computer Vision*, pp. 35-54, 1987

[2] Cox, I. J., Blanche: Position Estimation For an Autonomous Robot Vehicle. In: I.J.Cox and G.T.Hilfong (Eds.), *Autonomous Robot Vehicles*, Springer-Verlag, 1990.

[3] Dias, J., Batista, J., Simplicio, C., Araújo, H., Almeida, A., Implementation of an Active Vision System, *Mechatronical Computer Systems for Perception and Action*, Halmstad, June 1993.

[4] Durrant-Whyte, H., Uncertainty Geometry in Robotics. *IEEE Journal Robotics and Automation* - 4 (1), (1988), 23-31.

[5] Grosso, E., Tistarelli, M., Sandini, G., Active-Dynamic Stereo for Navigation, *2nd European Conf. Computer Vision, ECCV'92*, Italy, May 1992.

[6] Leonard, J.J., Durrant-Whyte, H., Mobile Robot Localization by Tracking Geometric Beacons, *IEEE Trans.Rob.Automation*, 7 (3), (June 1991), pp. 376-382.

[7] Leonard, J.J., Durrant-Whyte, H., Directed Sonar Sensing for Mobile Robot Navigation. Kluwer Academic Publishers, 1992.

[8] Pahlavan,K., Uhlin,T., Eklundh, J.-O., Integrating Primary Ocular Processes, *2nd European Conf. Computer Vision*, ECCV'92, Italy, May 1992.

[9] Robosoft SA, *Robuter^{TM} User's Manual*. Robosoft, August 1991.

[10] Tomasi, C., Shape and Motion From Image Streams: a Factorization Method. *PhD Dissertation*, Carnegie Mellon University, CMU-CS-91-172, Sept 1991.

ROBOT'S MOVEMENTS MODELLING AT CONTROL LEVEL : CONSEQUENCES ON ROBOT'S BEHAVIOUR AND CONTROL EFFICIENCY

P.J. ANDRÉ and E. JAMHOUR

Laboratoire d'Automatique de Besançon
Institut de Productique - 15 Impasse des Saint Martin
F- 25000 BESANÇON

Abstract : We want to point out the consequences of the choice of the model used by the controller, on the mechanical behaviour of the system to be controlled as well as on the efficiency of the control. After specification of the context we show how it is possible to choose an unifying control algorithm, based upon Cubic B-Spline curves representation, and allowing both efficiency and Soft Control, as we call it.

1. INTRODUCTION

In this paper we first briefly analyse the mechanical context of our work and the models commonly used to represent usual trajectories (linear, circular ....) and show the problems that arise. We establish the requirements for a model able to represent all currently used contours (linear, circular, parabolic, free form shapes) and efficiently achieve the control of a mechanical structure, avoiding shocks (thus vibrations), minimizing the energy spent by the actuators and leading to calculations compatible with the real time requirements.We show how a B-Spline-based model is a geometrically unifying one and meets all the requirements previously stated. In addition the speed law along a trajectory can, in most cases, be simply obtained, assimilating the parameter to time, by adequately choosing the distribution and the nature (multiplicity order) of the vertices defining the contour. This property is illustrated by a simple example dealing with a linear trajectory.

2. GENERALITIES

Our work deals with the control of mechanical systems : robots or machine-tools. Usually the movements of such systems are defined in terms of trajectory of the tool, specified in a 2 to 6 dimensions space. It has to be noticed that machining of a surface is achieved by successive sweeping along one main direction (not necessarily that of one of the machine's axis) and an incremental displacement along another main direction ; the tool follows a succession of trajectories. Thus all along the paper we will only consider trajectories.

In the machine-tool or robots control domain, the user's requirements increase in terms of speed, accuracy and variety of shapes. That leads to high speed computation abilties and trajectory modelling algorithm(s) efficiency. In addition any mechanical structure exhibit own vibration modes that can be excited by acceleration discontinuities inducing shocks. Our problem is to define a trajectory model meeting all the above mentioned requirements and eliminating kinematic discontinuities thus mechanical disturbances.

3. CHOOSING A MODEL

In current control systems only a few interpolation algorithms are available : linear and circular ones always exist, in some systems free form curves can be defined by pass-points but either it is very difficult to control the speed along the trajectory (the feed rate) or they are micro-segmented into straight or circular segments for on line control (that needs a large memory). Each interpolation is related to a specific algorithm, usually based upon a parametric definition of curves. The multiplicity of these algorithms can lead to discontinuities, with sad mechanical results, when linking two different interpolations : succession of linear and circular segments for instance.

3.1. Choice criteria

The trajectory model has to satisfy three groups of needs :

- those of the user : variety of shapes and speed laws, combination of high speed and accuracy, programming facilities either by the user or through a CAD system ;

- those of the computer : simple, and if possible unique, algorithm for all interpolations in order to speed up the calculations,lowest possible complexity of the computer in order to minimize its cost ;

- those of the mechanical system : elimination of shocks, minimization of the energy needed to follow the trajectories in order to reduce mechanical sollicitations, thus wear and tear.

3.2. Towards a solution

The successive points of the trajectory will be calculated on line ; thus we will choose a parametric definition of the curves. It allows us to build a calculator in which the trajectoy calculations are distributed, using one simple processor for each axis, all those processors being synchronized by a host one distributing the successive values of the parameter simultaneously to all axes processors.

The calculations at axis processor level have to be as simple as possible. A polynomial model leads to simple calculations and the lower the degree is, the faster the processor will work. Obviously closed curves like circles can only approximately represented by a polynomial model. However it has to be noticed that sine and cosine functions can only be approximately calculated. The only important thing is that the error can be known thus bounded and made smaller than the resolution of the machine.

The speed along the trajectory is $V(t) = ds/dt = ds(\mu)/d\mu \bullet d\mu(t)/dt$, μ being the parameter. The simpler the expression of $ds(\mu)/d\mu$ will be, the simpler will be that of $d\mu(t)/dt$, thus the easier the control of $V(t)$ will be. The ideal solution being a linear relationship between μ and t ($\mu=\lambda t$), an interesting model would be the one allowing, by an adequate choice of some parameters, the following property : $V(t) = V(\mu/\lambda) = \lambda\ ds(\mu)/d\mu$ for the desired speed law.

Whichever the model is, it must exhibit at least a C^2 continuity avoiding any acceleration discontinuity : that is a necessary condition for shock avoidance and deformations limitation. A polynomial model should be, at least, of the third degree.

If we now consider the compatibility with CAD systems, there are two polynomial models currently used in such systems : the B-Spline-based one and its subset the Bézier's one.

Only considering free form curves the Béziers's model leads to polynomials whose degree is equal to the number of pass-points minus one. It is obvious that if a large number of points is specified, then the degree becomes very high, leading to very long computations ; in addition modelling of circles is almost impossible.

The B-Spline-based models lead to a piecewise representation of curves by a polynomial whose degree is constant and independent of the number of pass-points. According to the C^2 continuity requirement a cubic model can be chosen. With such a model it is very easy to represent any currently used interpolation, either exactly (linear and parabolic ones) or with an error that can be bounded using a sufficient number of pass-points [BOU-86]. It has to be noticed that in most CAD systems Non Uniform Rational B Splines (NURBS) are currently used to exactly represent any shape. Unfortunately when many successive interpolations must be linked, the degree of both numerator and denominator dramatically increase making this model inadequate for control applications. But it is easy, thanks to exactly controlled approximations, to convert NURBS representation into B-Spline representation [PAT-89]. So a Cubic B-Spline-based model allows a very easy compatibility with CAD systems, thus, through such systems, an efficient man-machine interface.

Last but not least one can easily demonstrate that the minimization of the overall curvature, which is one of the properties of Cubic B-Splines is equivalent to that of the energy needed by a moving body to follow the trajectory with a constant speed.

The use of Cubic B-Splines for trajectory modelling simultaneously meets the requirement of shock avoidance (C^2 continuity) and energy minimization. This modelling leads us to introduce the concept of **Soft Control**.

4. CUBIC B-SPLINES MODELLING : KINEMATIC CONSIDERATIONS

We briefly recall that a Cubic B-Spline is defined by the following vectorial expression [BOO-78, BAR-82, BOU -86] in which $C(\mu)$ is the current point of the curve, μ the global parameter and the P_i s constitute an ordered set of points called vertices of the curve ; $C(\mu)$ and the P_i s have a number of coordinates equal to the dimension of the control space :

$$C(\mu) = \sum_{i=1}^{n} P_i \bullet N_{i,4}(\mu) \;; \quad \text{with } \mu \in [\mu_4, \mu_{4+n}]$$

Basic functions $N_{i,4}(\mu)$ are polynomials having a local support and recursively defined by [RIE 73] :

$$N_{i,1}(\mu) = \begin{cases} 1 & \text{if } \quad \mu \in [\mu_i, \mu_{i+1}] \\ 0 & \text{elsewhere} \end{cases}$$

μ_j are positive integers so that

and, for k > 1

$$\mu_j \leq \mu_{j+1}$$

$$N_{i,k}(\mu) = \frac{\mu - \mu_i}{\mu_{i+k-1} - \mu_i} \bullet N_{i,k-1}(\mu) + \frac{\mu_{i+k} - \mu}{\mu_{i+k} - \mu_{i+1}} \bullet N_{i+1,k-1}(\mu) \qquad \forall\, j \in [4, \ldots, 4+n-1]$$

the μ_j are the components of the so-called nodal vector : they are the values of the parameter at the junction points between two successive segments of the curve, these junction points being the M_j pass-points.

4.1. Nodal vector effects

If the μ_j distribution is regular, usually μ_j=j, the Splines are called Uniform Cubic B-Splines ; if the distribution is not regular they are called Non Uniform Cubic B-Splines.

From a kinematic viewpoint the choice between UCBS and NUCBS is quite significant. Assuming a linear relationship between the parameter and the time, $\mu=\lambda t$, the mean value $\overline{V_i}$ of the speed along the i^{th} segment of the curve is equal to : $\overline{V_i} = \dfrac{M_iM_{i+1}}{t_{i+1} - t_i} = \lambda \dfrac{M_iM_{i+1}}{\mu_{i+1} - \mu_i}$, where $M_i\, M_{i+1}$ is the length of the arc.

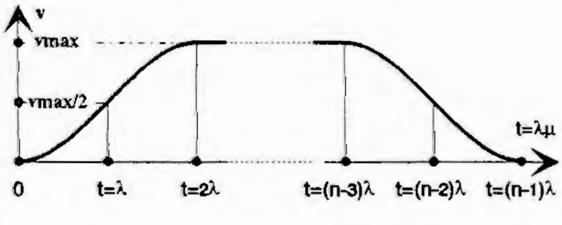
So the speed law along the curve will depend on the relative distribution of the nodal vector and of the positions of the pass-points ; conversely, if the pass-points are known and for a given speed law, it is possible to find a nodal vector leading to a satisfying approximate solution (the approximation is due to the integer nature of the μ_j).

In the case of constant curvature curves (circles, straight lines), the UCBS are of great interest since ds/dμ is constant (with a third order error for a circle) ; thus the speed law is identical to the temporal evolution of the parameter, leading to a very easy control of the speed.

4.2. Vertices effects

All above considerations are valid if vertices are simple ones. Vertices can be considered as attracting points whose level of attraction is defined by the $N_{i,4}$ functions. The attraction level of a given vertex can be increased by increasing its multiplicity order m : $P_i = P_{i+1} = \cdots = P_{i+m-1}$. Such a vertex will have a weight equal to $\sum_{l=1}^{m} N_{i+l-1,4}$.

As an example let us consider an UCBS defined by n vertices ($n \geq 7$), so that $P_0 = P_1 = P_2$ and $P_{n-3} = P_{n-2} = P_{n-1}$; in addition all vertices are aligned and equidistant. Assuming a linear relationship $\mu=\lambda t$, the speed evolution is represented below :



Such a speed law is particularly interesting since it means that the moving body starts from P_0 at rest (V=0, Γ=0) ; its speed increases until a maximum value V_{max}, according to two continuously linked parabolic laws ; the speed remains constant and then decreases according to symmetrical parabolic laws and the moving body reaches P_{n-1} at rest. The acceleration and deceleration speed laws are very close to the pseudo-sine law currently used in machine-tool control in order to avoid shocks during the acceleration and deceleration phases.

The result is that the movement is achieved in a near-minimum time but without the shocks induced by the classic trapezoidal law.

By means of this example we just wanted to point out how easy the speed law control is, using only the multiplicity order of the vertices. Equivalent examples can be given for other shapes of trajectories [HAD-91].

4.3. Minimum time control

We will give a third example of the efficiency of B-Spline-based Soft Control. Let us consider a trajectory defined by its initial and final points, given initial and final kinematic conditions and eventually a given number of pass-points (for instance in order to avoid any obstacle) ; in addition the movement is constrained by the maximum values of both the speed and acceleration. B-Spline modelling allows us to very easily design the trajectory leading to the minimum run time according to the various constraints, shocks avoidance included. The originality lies in that the trajectory's design takes into account *simultaneously* geometric and kinematic considerations instead of solving *first* a geometric problem *then* a kinematic one.

5. CONCLUSIONS

Starting from functionnal considerations we have shown that it is possible to design a control computer using a B-Spline-based unifying control model. Such a computer meets all the requirements of the user from the three viewpoints of the accuracy, the speed and the variety of interpolations and speed laws. The control model takes into account the behaviour of any mechanical system, avoiding undesirable effects and allowing a Soft Control. In addition, since the inputs of the control loops associated to each actuator exhibit a C^2 continuity, the regulators included in these loops have not to be as sophisticated as currently used ones. The prototype under developpement is realized using 320Cxx signal processors. It is able to renew the input of the control loop every 10 µs and thus is able to drive a robot or a machine-tool with a 10 µm accuracy combined with a 1 m/s speed.

REFERENCES

AND-91: P. ANDRÉ - M.C. HADDAD - C. MORLEC-BLONDEAU
Study and realization of a B-Spline-based trajectography module
"Engineering systems with intelligence/Concepts, tools and applications" - Volume "Microprocessor-based and intelligent system engineering" - pp 629-636 - Kluwer Academic publishers - 1991.

BAR-82: B.A. BARSKY
End conditions and boundary conditions for uniform B-Spline curve and surface representations
Computers in industry - N° 3 - pp 17-29 - 1982.

BEZ-86 : P. BÉZIER
Courbes et surfaces
Mathématiques et CAO - Volume 4 - Hermes - 1987.

BOO-78: C. de BOOR
A practical guide to Splines
Springer Verlag - 1978

BOU-86: Ph. BOUJON
Étude et réalisation d'un calculateur de commande numérique pour machines à hautes performances
Thèse de Doctorat de l'Université de Franche-Comté - Besançon - 1986.

HAD-91: M.C. HADDAD - P. ANDRÉ - C. MORLEC-BLONDEAU
Application of Uniform cubic B-Splines to machine-tools and robots numerical control : Determination of vertices under various constraints
Proceedings of the 15th ICAR - pp 1602-1605 - Pisa Italia - 1991.

LIN-83 : C.S. LI - P.R. CHANG - J.Y.S. LUH
Formulation and optimization of cubic polynomial joint trajectories for industrial robots
IEE Transactions on Automatic Control - Vol. AC-28 - N° 12 - pp 1066-1074 - 1983.

PAT-89 : N.M. PATRIKALAKIS
Approximation conversion of rational B-Splines
CAGD - Vol. 6 - pp 155-165 - 1989.

RIE-73 : R. RIESENFELD
Application of B-Spline approximation to geometric problems of computer-aided design
Ph. D Thesis - Syracuse university - 1973.

Parallel Computation of the Inertia Matrix of a Tree Type Robot Using one Directional Recursion of Newton-Euler Formulation

H. J. Pu; M. Müller; E. Abdalla*; L. Abdelatif *; E. Bakr*; H. A. Nour Eldin
Group of Automatic Control and Technical Cybernetics
University of Wuppertal, 42097 Wuppertal, Germany
e-mail: eldin@wrcd1.urz.uni-wuppertal.de
Fax: +49 202 4392953
* University of Helwan, Egypt

Abstract: This paper is devoted to the computation of the inertia matrix of a tree structured multi-arm robot system. Based on the PPO-Recursion proposed by the authors for the inertial, coupling, and gravitational dynamics of a robot [9], a parallel algorithm for computing the inertia matrix of chain structure robot has been achieved [10]. In the paper the PPO-Recursion is extended to be applied for the tree structured robot after. Appropriate interpretation of the dynamic properties of the branching link in the tree structure is introduced. The proposed algorithm offers high parallelism and is being under the realization in a transputer network.

1.INTRODUCTION

While the chain structure robot manipulator is the mostly used robot structure, the tree structure robot also finds its field of application in some especial, but promising areas of the industry and scientific research. It is, for instance, the case of multi-arm robot. In the Laboratory of Automatic Control and Technical Cybernetics at the University of Wuppertal a research project involving a camera guided robot motion is being carried out. The robot system for this project is a Manutec R2 manipulator with a camera-arm fixed on its 4th link. Fig. 1 illustrates the configuration of this flexible multi-arm robot system with 10 links and 10 joints. The dynamic model describing its motion comprises 10 second order differential equations.

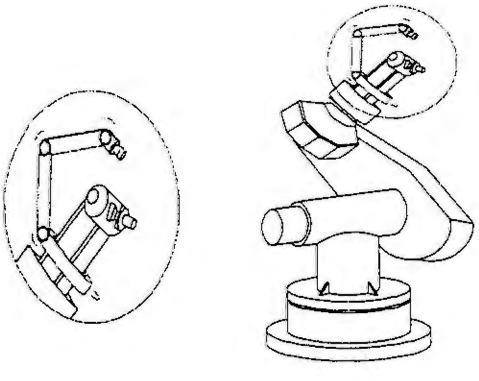


Fig. 1 Manutec R2 with a camera-arm

For the real time computation of the inverse dynamics of this tree type robot and for the robot motion simulation, a parallel executable scheme should be used such that a tolerable executing time is achieved. In both problems, the computation of the inertia matrix is the most expensive part. For this purpose several algorithms for computing the robot inertia matrix have been published [2, 5, 6]. Moreover, many efforts have also been devoted to the parallelization of the algorithms [7, 8]. In [10], the authors have suggested a novel algorithm for parallel computation of the inertia matrix of the chain structure robot. For 6 degrees of freedom, using 6 processors for parallel computing, this algorithm leads to the lowest number of additions and multiplications. In this paper the algorithm proposed in [10] will be extended for the tree structure robot.

2. INVERSE DYNAMICS OF A TREE STRUCTURE ROBOT

The tree structured robot system consists of a group of chain structure subsystems connected at some branching links. All links except the branching links can be treated equally as links in an open chain structure. For the open chain structur robot, as well as for the tree structure, extensive research results concerning its inverse dynamics have been published in the past years [1, 3, 4]. Recently the authors have presented a novel one directional recursive algorithm for the individual computation of the driving torque component (inertial, coupling and gravitational) according to their physical sources [9]. This PPO-Algorithm is based on the recursive Newton-Euler

Formulation and offers a high potential of parallelism for distributed computing. In [10], the computation of the inertia matrix of a chain structure robot using the PPO-Algorithm was achieved. For the purpose of extending this algorithm to compute the inertia matrix of the tree structured robot, the first two parts of the PPO-Algorithm, namely the recursions for the angular velocity, the geometrical parameters and the inertial torque are listed [9]:

A. *Recursion for Angular Velocity and Geometrical Parameters:*

$$ {}^{i}Z_k = R^{i}_{i-1}\,{}^{i-1}Z_k, \qquad \text{for } 0<k<i. \tag{2.1}$$

$$ {}^{i}b_k = R^{i}_{i-1}\left[{}^{i-1}b_k + ({}^{i-1}Z_k \times {}^{i-1}p_i)\right] \tag{2.2}$$

for 0<k<i; with ${}^{k}\mathrm{b}_{\mathrm{k}} \equiv 0$.

$$ {}^{\mathrm{i}}\beta_{\mathrm{k}} = {}^{i}Z_k \times {}^{i}p_{c_i} \qquad \text{for } 0<k<i \tag{2.3}$$

$$ {}^{\mathrm{i}}\omega_{\mathrm{i}} = \sum_{k=1}^{i} {}^{i}Z_k\dot{\theta}_k \tag{2.4}$$

B. *Inertial Torque Recursion:*

<u>*Acceleration*</u>

$$ {}^{\mathrm{i}}\varepsilon_{\mathrm{i}} = \sum_{k=1}^{i} {}^{i}Z_k\ddot{\theta}_k \tag{2.5}$$

$$ \begin{aligned} {}^{i}a_i &= R^{i}_{i-1}\left[{}^{i-1}a_{i-1} + {}^{i-1}\varepsilon_{i-1} \times {}^{i-1}p_i\right] \\ &= \sum_{k=1}^{i-1} {}^{i}b_k\ddot{\theta}_k \end{aligned} \tag{2,6}$$

$$ \begin{aligned} {}^{i}a_i^{*} &= {}^{i}a_i + {}^{i}\varepsilon_i \times \ {}^{i}pc_i \\ &= \sum_{\mathrm{k}=1}^{\mathrm{i}} \left({}^{\mathrm{i}}\mathrm{b}_{\mathrm{k}} + {}^{\mathrm{i}}\beta_{\mathrm{k}}\right)\ddot{\theta}_{\mathrm{k}} \end{aligned} \tag{2.7}$$

<u>*Centre of gravity force/torque*</u>

$$ {}^{i}F_i = m_i\,{}^{i}a_i^{*} \tag{2.8}$$

$$ {}^{i}N_i = {}^{c_i}I_i\,{}^{i}\varepsilon_i \tag{2.9}$$

<u>*Driving torque*</u>

for 0<k<i:

$$ \tau_{\mathrm{k}} = \tau_{\mathrm{k}} + {}^{i}N_i \bullet {}^{i}Z_k + {}^{i}F_i \bullet \left({}^{i}\beta_k + {}^{i}b_k\right) \tag{2.10}$$

$$ \tau_{\mathrm{i}} = {}^{i}N_i \bullet {}^{i}Z_i + {}^{i}F_i \bullet {}^{i}\beta_i \tag{2.11}$$

Here τ_{i} is the component of the inertial torque τ_M which is defined as [9]

$$ \tau_M = \left[\tau_1 \ \ \ldots \ \ \tau_1\right]^T = \mathrm{M}(\theta)\ddot{\theta} \tag{2.12}$$

The tree structure illustrated in Fig. 2 can be decomposed into three chains. The first chain is from the base link A_0 to the branching link A_m. Then from the two links B_{11} and B_{21} which are connected with the branching link two chains are starting to B_{1p} and B_{2r} respectively. These three chains will be called herewith: chain A, chain B_1 and chain B_2. The formulation for the three chains A, B_1 and B_2 is reducable to new consideration for the triple links A_m, B_{11} and B_{21}, while the other links can be treated as links in an open chain structure.

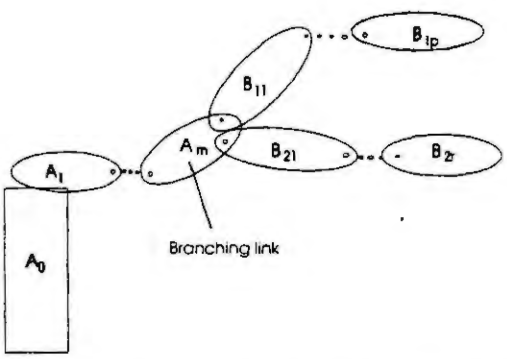


Fig. 2 Tree structure with one branching link

For the forward propagation of the kinematic variables such as the angular velocity ω, angular acceleration ε and linear acceleration *a* of the centre of gravity of the links the usual recursions can be executed from A_0 to A_m and continued to B_{1p} and B_{2r} respectively.

Concerning the backward propagation of the force and torque, one should consider the fact that the motion of the branching link A_m is influenced by the motion of both B_{11} and B_{21}. The following equations are thus valid for the branching link A_m:

$$ {}^{a_m}f_{a_m} = {}^{a_m}F_{a_m} + R^{a_m}_{b_{11}}\,{}^{b_{11}}f_{b_{11}} + R^{a_m}_{b_{21}}\,{}^{b_{21}}f_{b_{21}} \tag{2.13}$$

$$ \begin{aligned} {}^{a_m}n_{a_m} = {}^{a_m}N_{a_m} + {}^{a_m}p_{c_{a_m}} \times {}^{a_m}F_{a_m} + R^{a_m}_{b_{11}}\,{}^{b_{11}}n_{b_{11}} + \\ R^{a_m}_{b_{21}}\,{}^{b_{21}}n_{b_{21}} + {}^{a_m}p_{b_{11}} \times R^{a_m}_{b_{11}1}\,{}^{b_{11}}f_{b_{11}} + \\ {}^{a_m}p_{b_{21}} \times R^{a_m}_{b_{21}}\,{}^{b_{21}}f_{b_{21}} \end{aligned} \tag{2.14}$$

$$\tau_{a_m} = {}^{a_m}n_{a_m} \bullet {}^{a_m}Z_{a_m} + {}^{a_m}f_{a_m} \bullet {}^{a_m}Z_{a_m} \qquad (2.15)$$

3. INERTIA MATRIX OF THE TREE STRUCTURE ROBOT

According to the equation (2.12), the inertia matrix of the robot can be simply written as:

$$M(\theta) = \frac{\partial \tau_M}{\partial \ddot{\theta}} \qquad (3.1)$$

As shown in (2.5) to (2.9), all the kinematic and kinetic variables ε_i, a_i, a_i^*, F_i, N_i of the Inertial Torque Recursion are linear to the joint accelerations $\ddot{\theta}$.

Introducing the two new parameters

$${}^iF_i^j := \frac{\partial\, {}^iF_i}{\partial \ddot{\theta}_j} = m_i\left({}^ib_j + {}^i\beta_j\right) \qquad (3.2)$$

$${}^iN_i^j := \frac{\partial\, {}^iN_i}{\partial \ddot{\theta}_j} = {}^{Ci}I_j\, {}^iZ_j \qquad (3.3)$$

one obtains the individual element $M_{ij}(\theta)$ of the inertial matrix

$$M_{ij}(\theta) = \frac{\partial \tau_i}{\partial \ddot{\theta}_j} = \sum_{k \geq i}\left[{}^kN_k^j \bullet {}^kZ_i + {}^kF_k^j \bullet \left({}^kb_i + {}^k\beta_i\right)\right] \qquad (3.4)$$

The summation should be carried out over all the links succeeding the link i, for each branch of the tree.

Only the recursion for the geometrical parameters in (2.1)-(2.3) has to be executed. Then the partial forces and torques defined in (3.2) and (3.3) can be calculated. By considering the special situation of the branching link described in section 2, one obtains the following procedure for computing the inertial matrix of tree structured robot:

A. Recursions for the Geometric Parameters

a.1 Executing (2.1)-(2.3) for i=1 to m by setting the index 0 for the base A_0, 1 to m for the links A_1 to A_m.

a.2 Continuing the recursion **a.1** until i=m+p by setting the index m+1 for link B_{11}, and subseqently to the other links and up to setting m+p for link B_{1p}.

a.3 Continuing the recursion **a.1** until i=m+r by setting the index m+1 for link B_{21}, and the subsequent links till setting index m+p for link B_{2r}

The recursions a.2 and a.3 are independent on each other and can be executed in parallel.

B. Recursions for Inertia Matrix

b.1 Executing (3.2) and (3.3) for each link.

b.2 Executing (3.4) for i=1 to m and k=1 to m.

b.3 Executing (3.4) for i=1 to m+p and k=m+1 to m+p by setting the index as in **a.2**.

b.4 Executing (3.4) for i=1 to m+r and k=m+1 to m+r by setting the index as in **a.3**.

b.5 For i=1 to m (each link in chain A)

$$M_{ij} = M_{ij}^a + M_{ij}^{b_1} + M_{ij}^{b_2}$$

where

M_{ij}^a: is the value for M_{ij} obtained by **b.2**.

$M_{ij}^{b_1}$: is the value for M_{ij} obtained by **b.3**.

$M_{ij}^{b_2}$: is the value for M_{ij} obtained by **b.4**.

b.2, b.3 and b.4 are independent on each other and can be executed parallel

As is shown in [10], the computation of the geometric parameters and the elements of the inertia matrix can be organized in a manner that rends a high parallelism to be achieved.

4. CONCLUSION

The computation of the inertia matrix of a tree structure robot is discussed in this paper. The PPO-Algorithm [9] for robot inverse dynamics has been extended to the multi-arm robot manipulator. Based on this extension, a parallel, one directional and recursive algorithm in Newton-Euler Formulation for the computation of the inertia matrix of the robot system with tree structure has been achieved. This algorithm is now being under realization on a transputer-network and implemented in the DFG-Project 135/13-2.

5. ACKNOWLEDGEMENT

The results shown in this paper have been partially supported by the Deutsche Forschungsgemeinschaft (DFG), Project No. 133/13-1.

6. LIST OF SYMBOLS

a•b: Scalar product of the vector a and vector b.

a×b: Vector product of the vector a and vector b.

m_i: Mass of the i-th link.

${}^{ci}I_i$: The inertia of the i-th link with I_i the component in the direction of its own rotational axis.

${}^{j}p_i$: Position vector of the i-th frame origin represented in j-th frame.

${}^{j}p_{c_i}$: Position vector of the c.o.g. of the i-th link presented in j-th frame.

${}^{i}Z_i$: The i-th z-axis fixed on the i-th link. (rotational axis of the i th link).

$\dot{\theta}_i$: Joint angular velocity of the i-th link.

$\ddot{\theta}_i$: Joint angular acceleration of the i-th link.

${}^{i}\varepsilon_i$: Angular acceleration of the i-th link.

${}^{i}a_i$: Linear acceleration of the origin of the i-th frame.

${}^{i}a_i^*$: Linear acceleration of the centre of gravity (c.o.g.) of the i-th link

R_i^j: Coordinate rotation matrix from i-th frame to j-th frame.

${}^{i}F_i$: Force acting on the c.o.g. of the i-th link.

${}^{i}N_i$: Torque acting on the i-th link.

${}^{i}Z_k$: The k-th z-axis represented in i-th frame.

${}^{i}b_k$: Its magnitude is the distance between the k-th frame origin and the i-th z-axis.

${}^{i}b_k + {}^{i}\beta_k$: Its magnitude is the distance between thecentre of gravity (c.o.g.) of the k-th link and the i-th z-axis.
Its direction is the effective direction of ${}^{k}F_k$ for driving the rotation of the i-th link referring to the i-th z-axis.

7. REFERENCES

1. Luh, J. Y. S.; M. W. Walker and R. P. C. Paul (1980): On-line computational scheme for mechanical manipulators. Tran. ASME J. Dynamic Syst., Meas., Contr., Vol. 102, pp. 69-79.

2. Walke, M. W. and D. E. Orin (1982): Efficient Dynamic Computer Simulation of Robotic Mechanisms. Tran. ASME J. Dynamic Syst., Meas., Contr., Vol. 104, pp. 205-211.

3. Desoyer, K.; P. Kopacek and I. Troch (1985): Industrieroboter und Handhabungsgeräte: Aufbau, Einsatz, Dynamik, Modellbildung und Regelung. R. Oldenbourg Verlag München Wien.

4. Khalil, W. and J. F. Kleifinger (1987): Minimum Operation and Minimum Parameters of the Dynamic Models of Tree Structure Robots. IEEE J. Robot., Autom. Vol. RA-3, No. 6. pp.

5. Lee, C. S. G. and B. H. Lee (1987): Development of Generalized d´Alembert Equations of Motion for Robot Manipulators. IEEE Tran. Syst., Man, Cybn., Vol. SMC-17, No. 2, pp. 311-325.

6. Li, C. J. (1988): A New Method of Dynamics for Robot Manipulators. IEEE Tran. Syst., Man, Cybn., Vol. SMC-18, No. 1. pp. 105-114.

7. Lee, C. S. G. and P. R. Chang (1988): Efficient Parallel Algorithms for Robot Forward Dynamics Computation. IEEE Tran. Syst., Man, Cybn., Vol. 8, No. 2.

8. Fijany, A. and A. K. Bejczy (1988): A Class of Parallel Algorithms for Computation of the Manipulator Inertia Matrix. IEEE Tran. Robot., Autom. Vol. 5, No. 5, pp. 600-615.

9. Pu, H. J.; E. Abdalla; L. Abdelatif; H. A. Nour Eldin (1993): A Novel Physically Parallel One Directional Recursive Algorithm (PPO-Recursion) for Robot Inverse Dynamics. IEEE/SMC'93 Conference: System Engineering in Service of Humans, Le Touquet, France, Oct. 17-20 1993.

10. Pu, H. J.; E. Abdalla; M. Müller; H. A. Nour Eldin (1993): Parallel Comutation of the Manipulator Inertia Matrx through One Directional Recurtion in the Newton-Euler Formulation. VDI/VDE Fachtagung: Intelligente Steuerung und Regelung von Robotern, Langen, Germany, Nov. 9-10 1993.

GOALS RECOGNITION AS ABDUCTION

Aldo Franco Dragoni, Paolo Puliti
Istituto di Informatica, Università di Ancona
via Brecce Bianche, 60131, Ancona (Italy)
e-mail dragon@anvax2.cineca.it

Abstract. In the forthcoming distributed autonomous robotic systems it will be useful for a robot to recognize others robots' goals and plans from visual information. This paper is about goals' recognition. Let A and O be two robots both guided by a STRIPS-like planner. Let A be the robot which is doing an action and let O be the robot which is observing A's doing the action. We show that, under simple hypotheses on the nature of the planner that guides A's behaviour, O can recognize A's goal by means of abduction.

1. INTRODUCTION

Many people in the Artificial Intelligence community and, especially, in the Distributed A. I. community [1] have shown the importance of *plan recognition* as inferring the other agents' plans from their partially performed portions [3][4]. There is no biunique correspondence between plans and goals because, in general, the same plan can be performed to accomplish different goals and the same goal can be accomplished by different plans. This paper is about *goals recognition*: having recognised a plan (may be after the entire plan has been performed), try to recognise which were the reasons for the plan to be performed. It should be obvious that the same sequence of actions (plan) produces different effects on the world depending on the particular situation in which it is performed. Furthermore, if the planner possesses sufficient inference ability, then the plan's goal could be not simply that of adding and/or removing the facts explicitly listed in the actions' definitions, but it can be some state of affairs which will be implied by these changes; in other words, the goal(s) of the plan could be some "logical consequence(s)" of the changes made in the world by the actions in the plan (particularly the last one). If this is the case, then goal recognition is, in general, not a trivial problem and we show that it can be considered as an abduction problem. To simplify the matter (without loss of generality) we'll limit the discourse to (plans made of single) actions. We address this problem in a formal way adopting the classical and very simple characterisation of actions given in [6]; actions are means to modify the state of the world by adding and/or removing facts. Finally, we'd like to note that a goal of a partially performed plan is that of making performable the rest of the plan, so goals recognition can be regarded as a step in plan recognition.

2. ABDUCTION

2.1 A general model of abduction

Abduction is generally presented as an abstract hypothetical inferential schema that, given a causal theory of the world (a set of formal rules or links between causes and effects) and a set of observations (facts which don't follow simply from the causal theory), tries to find an explanation of the observed facts, that is a set of hypothetical facts which, along with the causal theory, justifies the presence of the observed facts. We've found very general the account of abduction given in [2]. *d* stands for a datum, e.g., a symptom. *h* stands for an individual hypothesis, e.g., a hypothesized disease. *H* can be treated as a composite hypothesis, i.e., each $h \in H$ is hypothesized to be present, and each $h \notin H$ is hypothesized to be absent or irrelevant.

Definition. An *abduction problem* is a tuple $<D, H_{all}, e>$, where:

- D *is* a finite set of all the data to be explained,
- H_{all} is a finite set of all the individual hypotheses,
- e is a map from subsets of H_{all} to subsets of D (H explains $e(H)$),

Furthermore:
H is *complete* if $e(H)=D$
H is *parsimonious* if $\neg\exists H' \subset H (e(H) \subseteq e(H'))$,
H is an *explanation* if it is complete and parsimonious.

Definition. An abduction problem is *monotonic* if:

$$\forall H, H' \subseteq H_{all} \, (H \subseteq H' \rightarrow e(H) \subseteq e(H'))$$

that is, a composite hypothesis does not "lose" any data explained by any of its subsets and might explain additional data.

Definition. An *incompatibility abduction problem* is an abduction problem in which some collection of hypotheses are incompatible.

In [2] it is proved that, for an incompatibility abduction problem, even finding an explanation is NP-hard.

2.2 A logic-based model of abduction.

Let L be a propositional language. A literal is an atomic sentence of L or its negation. A clause is the disjunction of distinct literals of L. Let both the data and the hypotheses be expressed as clauses of L,[1] so that D and H_{all} are *given* sets of clauses[2]. Let the *domain theory* Σ be a distinct set of clauses of L. Let e be the mapping so defined:

$$e(H)=\{d\in D \mid H\cup\Sigma\models d\}$$

that is, $e(H)$ is the deductive closure of $H\cup\Sigma$ restricted to D. So we can define a logic-based explanation of a set of clauses D to be a finite set of clauses $H\subseteq H_{\text{all}}$ such that

- $\Sigma\cup H\models D$, that is H is complete,
- H is parsimonious.

Monotonicity of the first order logic assures that this is a monotonic abduction problem. We specify the notion of "incompatibility abduction problem" by excluding all the composite hypotheses and only them which logically follows from $\Sigma\cup\sigma$, where the *consistency set* σ is a given set of clauses[3]. These characterisations bring to the following definition:

Definition. A *logic-based abduction problem* is that of finding an explanation $H\subseteq H_{\text{all}}$ of a set of observations D such that, given a prefixed consistency set σ,

- $\Sigma\cup\sigma\not\models\neg H$, that is H is consistent with $\Sigma\cup\sigma$
- $\Sigma\cup H\models D$
- H is parsimonious

If there are a finite number of explanations for D then the cautious explanation is their disjunction $\vee_i H_i$. If H_{all} is limited to atomic sentences and if $\sigma=\emptyset$, then this logic-based abduction problem collapses to that defined in [5] and in [7]. However, we'll stay in the more general framework in which individual hypotheses are disjunction of literals.[4] There is no intrinsic relation between the cardinalities of H and D. Depending on the domain theory Σ, an explanation H can be a set of more than one clause even if D is made of a single clause and, vice-versa, a single clause can be the explanation of a set of clauses. Sometimes may be useful to consider only explanations of a given cardinality. We introduce the following definition.

Definition. An *$n\times m$ abduction problem* is a logic-based abduction problem in which the cardinality of D is n and the cardinality of the explanations H is forced to be m.

3. ACTIONS

We adopt the simple STRIPS-like definition for actions. Actions are atomic means to modify the state of the world by adding and/or removing facts. We say that an action $\mathfrak{A}$ changes the state of the world from S to $\mathfrak{A}(S)$. We adopt the extensional view of actions, in the sense that two different instances of the same action are considered as different actions. In so far, both the preconditions and the effects of an action will be expressed as sets of propositions of L.

Definition. A *state theory* T is a prefixed set of sentences of L. A *state* S is a set of atomic sentences of L consistent with the state theory T.

A state represents the world where the robot is working in at a given instant. The state theory represents constraints that must be verified in every instant. To simplify the matter we ignore relations which must be verified between facts at different instants.

[1] It is a useful generalization of the case in which they are simple literals.

[2] Every set of clauses can be mapped in a wff and vice-versa, so D_{all} and H_{all} can be regarded as wffs of L.

[3] A set of sentences is logically equivalent to their conjunction and the negation of a conjunction of sentences is logically equivalent to the disjunction of their negations.

[4] This generalization, although problematic from a pragmatical computational point of view, is conceptually stimulating because it provides for the case in which, in order to explain some observations, it is not sufficient to hypothesize mere facts but it is necessary to hypothesize the presence of other rules in the domain theory (other links in the mapping e). It is this kind of abduction that (along with induction) is at the base of the formulation of scientific theories.

Definition. An *action* $\mathfrak{A}$ is a tuple $<P, AL, DL>$ in which P, AL and DL are sets of atomic sentences of L respectively called *preconditions*, *add list* and *delete list*.

The preconditions are facts that must hold in the current state for the action to be planned. The add list contains facts that are added to the world by the action and the delete list contains facts that are removed from the world by the action. It can be given a functional definition of action.

Definition. An action $\mathfrak{A}=<P, AL, DL>$ is a partial function over the space of the states defined as follows:
$S \cup T \nvDash P$: $\mathfrak{A}(S)$ undefined.
$S \cup T \vDash P$: $\mathfrak{A}(S)=(S \setminus DL)\cup AL$

Definition. A goal G is a conjunction of literals of L.

We suppose that the planner has complete and correct knowledge about the environment and the effects of each action in the world. So, if the robot has already performed the action then the intent of the action cannot be a disjunction of literals but a literal or a conjunction of literals.

4. GOALS RECOGNITION

The results of this paper hold upon the following fundamental assumption:

Assumption. The planner is *correct* and *complete*, that is, if the planner verifies a sentence p in a state S then $S\cup T \vDash p$, and if $S\cup T \vDash p$ then the planner is able to prove it. Furthermore, if the planner plans an action $\mathfrak{A}$ to pursue a goal G in a state S then:
1. $S\cup T \nvDash G$, that is G is not already satisfied in S
2. $\mathfrak{A}(S)\cup T \vDash G$, G will be satisfied in $\mathfrak{A}(S)$

Theorem. If a planner plans an action $\mathfrak{A}=<P, AL, DL>$ in a state S, under a state theory T, to pursue a goal G, then the problem to recognise the planner's goal in planning the action is a 1×1 abduction problem which has $T\cup(S \setminus DL)$ as "domain theory", $\neg AL$ as "observation", DL as consistency set and the negation of the goal as explanation.

Proof. Consider a generic action $\mathfrak{A}=<P, AL, DL>$. Let S be $s\cup DL$ and $\mathfrak{A}(S)$ be $s\cup AL$. From the 1. and 2. it follows

1a. $T\cup s\cup DL \nvDash G$
2a. $T\cup s\cup AL \vDash G$

From the 2a. it follows:

2b. $T\cup s\cup\neg G \ \vDash \neg AL$

If we take H for $\neg G$, D for $\neg AL$, σ for DL, and Σ for $T\cup s$ we obtain:
- $\Sigma\cup\sigma \nvDash \neg H$
- $\Sigma\cup H \vDash D$

Evidently, both D and H are made of a single clause because they are the negations of a conjunction, so this is a 1×1 abduction problem. ❑

A plan P is a sequence of actions $A_1,..,A_n$. We can represent a plan as the ordered composition of the partial functions corresponding to the actions in the plan $\mathfrak{P}=\mathfrak{A}_n(..(\mathfrak{A}_1(S)..)=<PP, PAL, PDL>$, in which PP is the set of facts that must hold in the initial state for the plan to be performed, PAL contains facts that are added to the world by the plan as a whole and PDL contains facts that are removed from the world by the plan. It is straightforward to extend the previous result to the partial function $\mathfrak{P}=<PP, PAL, PDL>$.

Example. In a classical block-world domain there are a table, three blocks and a gripper. Consider the following instanced action for a robot (from [6])

putdown_b1
DL: holding_b1
AL: ontable_b1,clear_b1,handempty

along with the following (piece of) current state and domain theory:

S: holding_b1,ontable_b2,ontable_b3
T: holding_b1 ∨ holding_b2 ∨ holding_b3 → ¬handempty
on_b2_b1 ∨ on_b3_b1 → ¬clear_b1
holding_b1 → ¬on_b2_b1 ∧ ¬on_b3_b1 ∧ ¬on_b1_b2 ∧ ¬on_b1_b3 ∧ ¬ontable_b1
holding_b1 → ¬holding_b2 ∧ ¬holding_b3
ontable_b1 ∧ ontable_b2 ∧ ontable_b3 → filled_table

handempty → ontable_b1 ∨ on_b1_b2 ∨ on_b1_b3

If the robot plans **putdown_b1**, then *S\DL*={ontable_b2,ontable_b3}, *D*= {¬ontable_b1 ∨ ¬clear_b1 ∨ ¬handempty}. We obtain the following 1×1 abduction problem:

- *T*∪{ontable_b2,ontable_b3}∪{holding_b1} ⊭ ¬*H*
- *T*∪{ontable_b2,ontable_b3}∪*H* ⊨ ¬ontable_b1 ∨ ¬clear_b1 ∨ ¬handempty

from which we infer abductively the following singleton explanations:

	H	*G*
1	¬ontable_b1	ontable_b1
2	¬clear_b1	clear_b1
3	¬handempty	handempty
4	holding_b1	¬holding_b1
5	holding_b1 ∨ holding_b2	¬holding_b1 ∧ ¬holding_b2
6	holding_b1 ∨ holding_b3	¬holding_b1 ∧ ¬holding_b3
7	holding_b1 ∨ holding_b2 ∨ holding_b3	¬holding_b1 ∧ ¬holding_b2 ∧ ¬holding_b3
8	¬filled_table	filled_table

Goals 1+3 are trivial, the others depends on the state theory and on the current state. The following clauses are not explanations because they satisfy the second condition but not the first one (they are inconsistent with *T*∪*DL*, that is the same to say that the correspondent goals were already verified).

	H	*G*
1	holding_b2	¬holding_b2
2	holding_b3	¬holding_b3
3	holding_b2 ∨ holding_b3	¬holding_b2 ∧ ¬holding_b3
4	on_b2_b1	¬on_b2_b1
5	on_b3_b1	¬on_b3_b1
6	on_b2_b1 ∨ on_b3_b1	¬on_b2_b1 ∧ ¬on_b3_b1

The explanation ¬ontable_b1 ∧ ¬on_b1_b2 ∧ ¬on_b1_b3 is not singleton (the disjunctive goal ontable_b1 ∨ on_b1_b2 ∨ on_b1_b3 is not acceptable).

5. RESULTS

If a planner possesses sufficient inference ability, then the goal(s) of a plan can be generalized to be some "logical consequence(s)" of the changes made in the world by the plan. We've shown that, in this case, under the hypothesis that the planner is correct and complete, goal recognition can be regarded as an abduction problem with a single datum to explain (a clause) making a single hypothesis (an other clause). Furthermore we've charachterized this logic-based abduction problem as an incompatibility one.

6. REFERENCES

[1] A. H. Bond and L. Gasser eds., Readings in Distributed Artificial Intelligence, Morgan Kaufmann Publishers, San Mateo, CA, 1988.

[2] Tom Bylander, D. Allemang, M. C. Tanner and J. R. Josephson, The computational complexity of abduction, *Artificial Intelligence*, 49,25-60, 1991.

[3] Sandra Carberry, Incorporating Default Inferences into Plan Recognition, Proceedings of 1990 Conference of the American Association for Artificial Intelligence, 471-478, 1990,

[4] Henry Kautz, A Circumsriptive Theory of Plan Recognition, in *Intentions in Communication*, P. R. Cohen & J. Morgan & M. E. Pollack eds., MIT Press, Cambridge, Massachusetts,1990.

[5] K. Konolige, Abduction versus closure in causal theories, *Artificial Intelligence*, 53, pp 255-272, 1992.

[6] N. J. Nilsson: "Principles of Artificial Intelligence" Springer-Verlag 1981

[7] Raymond Reiter, Johan De Kleer, Foundations of Assumption-Based Truth Maintenance Systems, Proceedings of 1987 Conference of the American Association for Artificial Intelligence,, 183-187, 1987.

APPLICATION OF NEURAL NETWORKS IN ROBOT DYNAMIC CONTROL

by A.S.Morris and S.Khemaissia

Robotics Research Group, Department of Automatic Control and Systems Engineering
University of Sheffield, P O Box 600, Mappin Street, Sheffield S1 4DU, U.K.

Abstract. Neural network based adaptive controllers have been shown to achieve much improved accuracy compared with traditional adaptive controllers when applied to trajectory tracking in robot manipulators. This paper describes a new Recursive Prediction Error technique for estimating network parameters which is more computationally efficient. Results show that the neural controller suppress disturbances accurately and achieves very small errors between commanded and measured trajectories.

1. INTRODUCTION

The control of a robot such that it adheres accurately to some pre-planned trajectory in space is a problem of acknowledged difficulty, especially when high-speed motion is demanded. The use of conventional controllers demands the availability of an accurate dynamic robot model. However, this is rarely achievable because of unmodelled dynamics (neglected time delays, non-linear friction etc.) and parameter uncertainties (deviation of link lengths etc. from nominal values).

The emergence of neural networks has provided an alternative means of controlling high-speed robot motion, and investigations into their use are the subject of this paper. Neural networks can perform a combined system identification and adaptive control function, and they yield a good manipulator trajectory tracking performance without requiring an analytical dynamic robot model.

The form of learning algorithm which has most commonly been tried in neural-network-based adaptive controllers is back propagation. Recently, a new way of estimating neural network parameters has been developed, called the recursive prediction error technique. The main purpose of this paper is to compare the relative merits of these alternative learning algorithms.

2. NEURAL ADAPTIVE CONTROLLER

Early work on neural adaptive controllers by Kawato [4] used a neural network in the feedforward loop with a conventional PD controller in the feedback loop. However, this was computationally expensive, requiring extensive pre-processing of non-linear transformations of the input signals. Later work by Tomochika [7] used a nonlinear compensator using neural networks, which incorporates the idea of the computed torque method. The neural networks are used to compensate for nonlinearities in the robotic manipulators, rather than to learn the inverse dynamics.

The authors have recently proposed a new approach to neural-adaptive, trajectory-tracking control in which the neural network architecture is combined with a servo feedback controller [5]. The scheme resembles a standard feedforward control structure except that the manipulator's inverse dynamic model is replaced by a generic neural network model for each joint which adaptively approximates the joint inverse dynamics using a back propagation algorithm. The performance of this scheme is good in terms of trajectory tracking.

Billings [2] has recently developed a scheme where the recursive prediction error (RPE) identification method is used to estimate the network parameters rather than using back propagation to adjust the network weights. This has been shown to greatly improve the speed of learning and hence it seemed appropriate to investigate the application of this in neural adaptive control of robots.

3. DYNAMIC EQUATIONS AND NEURAL CONTROLLER ALGORITHM

Given a desired trajectory defined in terms of joint variables $(q_d, \dot{q}_d, \ddot{q}_d)$, the control problem is to compute the necessary torques to apply to the joint actuators such that the manipulator follows the desired trajectory, figure 1. This requires computation of the inverse dynamic equation, which, for an n-link rigid robot arm, is given in vectorial form as:

$$T = M(q)\ddot{q} + V(q,\dot{q}) + F(\dot{q}) + G(q) = M(q)\ddot{q} + Q(q,\dot{q}) \tag{1}$$

The manipulator's forward dynamic equation can be readily obtained by manipulating equation(1):

$$\ddot{q} = M^{-1}(q)\, T - M^{-1}(q)\, Q(q,\dot{q}) = R(\, q,\dot{q},T\,) \tag{2}$$

Equation (2) represents a nonlinear mapping from the robot input (*joint torque* T) to the robot output (*joint motion*).

The robot inverse dynamics can now be written as

$$T = R^{-1}(\, q,\dot{q},\ddot{q}\,) \tag{3}$$

where the transformation R^{-1} is a nonlinear mapping from the joint coordinate space to the joint torque space.

In practice, robot dynamics cannot be modeled exactly. An estimated model $\hat{R}^{-1}$ is used to predict the feedforward torques and a servo-feedback is usually included to bring robustness to the overall control scheme.

The system dynamics are not time invariant and undergo changes such as variations in payloads, changes in the friction coefficients of the joints etc. Hence, the model estimate $\hat{R}^{-1}$ has to be modified accordingly in order to accommodate for these changes. To achieve this, an adaptive or a learning control element is usually associated with the control structure.We propose a novel control architecture where $\hat{R}_p^{-1}$ is modeled by artificial neural networks.

The nonlinear inverse transformation (3) can be decoupled into n less complex transformations:

$$T = R^{-1}(\, q,\dot{q},\ddot{q}\,) = \left[\, r_1^{-1}(q,\dot{q},\ddot{q}) \quad ..... \quad r_n^{-1}(q,\dot{q},\ddot{q})\right]^T \tag{4}$$

where

$r_i^{-1}(\, q,\dot{q},\ddot{q} \;\; ; \;\; i = 1, ... ,n\,)$ defines the inverse dynamics transformation of the corresponding joint. Each r_i^{-1} can be modeled by an ANN such that:

$$T = R^{-1}(\, q,\dot{q},\ddot{q}\,) = \left[\, \hat{r}_1^{-1}(q,\dot{q},\ddot{q}) \quad ..... \quad \hat{r}_n^{-1}(q,\dot{q},\ddot{q})\right]^T = \left[\, N_1(q,\dot{q},\ddot{q},\Theta_1) \quad ..... \quad N_n(q,\dot{q},\ddot{q},\Theta_n)\right]^T \tag{5}$$

where (ˆ) denotes an estimate, $N_i(\, .\,)$ represents the output of each ANN model used to realize the nonlinear mapping $r_i^{-1}(\, .\,)$ and Θ terms denote the set of adjustable weights of the corresponding ANN model.The steps in the neural controller algorithm can be summarised as follows:

1. Present the desired inputs $q_d(t)$, $\dot{q}_d(t)$, $\ddot{q}_d(t)$.
2. Execute forward phase of back propagation algorithm to give torques N at the output of ANN model
3. Add error vector to torques N to obtain required joint actuator torques
4. Measure real robot system outputs $q(t)$, $\dot{q}(t)$, $\ddot{q}(t)$ with calculated torques applied to joint actuators, and compute the error vector between desired and actual vector of positions and velocities
5. Apply this error vector in the learning algorithm (BP, RPE, PRPE)
6. Repeat steps 1-5 until convergence is achieved

4. RECURSIVE PREDICTION ERROR (RPE) PARAMETER ESTIMATION

Let $\Theta = [\theta_1, \theta_2, ...\theta_n]^T$ represents all the unknown weights and thresholds of the network. A network with a single layer of hidden units can then be represented by the model [3]:

$$\hat{y}_i(k,\theta) = \sum_{j=1}^{j=n_1} \omega_{ij}^2 x_j^1(k) = \sum_{j=1}^{j=n_1} \omega_{ij}^2 g\left[\sum_{m=1}^{j=n_0} \omega_{jm}^1 x_m(k) + b_j^1\right] \qquad with \qquad 1 \le i \le n_1 \tag{6}$$

where $x(k) = [\, x_1(k), \ldots\ldots, x_{no}(k)\,]^T$ is the input vector to the network, ω_{jm} are the weights , b_j are the thresholds and $g\,[.]$ is the node activation function.

The equations for the RPE algorithm are [3]:

$$e(k) = y(k) - \hat{y}(k) \tag{7}$$

$$P(k) = \frac{1}{\lambda(k)}\left(P(k-1) - P(k-1)\Psi(k)\, (\lambda(k)I + \Psi(k)^T P(k-1)\Psi(k)\,)^{-1}\Psi(k)^T P(k-1)\, \right) \tag{8}$$

$$\hat{\Theta}(k) = \hat{\Theta}(k-1) + P(k)\, \Psi(k)\, e(k) \tag{9}$$

$\lambda\,(k)$ is the forgetting factor in the form

$$\lambda(k) = \lambda_0\lambda(k-1) + (1 - \lambda_0) \tag{10}$$

The number of parameters to be estimated with a single hidden layer is given by:

$$n_\theta = (n_0 + 1)\, n_1 + n_1\, n_2 \tag{11}$$

The elements of the gradient $\Psi(k,\theta)$ can be obtained by differentiating (6) with respect to θ_i [1]:

$$\Psi_{ij} = \frac{d\hat{y}_j}{d\theta_i} = \begin{cases} x_m^1 & \text{if } \theta_i = \omega_{jm}^2 \quad 1 \le m \le n_1 \quad \textit{weights in hidden-output layers} \\ x_m^1(1-x_m^1)\omega_{jm}^2 & \text{if } \theta_i = b_m^1 \quad 1 \le m \le n_1 \quad \textit{thresholds in hidden layers} \\ x_m^1(1-x_m^1)\omega_{jm}^2 x_m & \text{if } \theta_i = \omega_{ml}^1 \quad 1 \le m \le n_1 \quad \textit{weights in input-hidden layers} \\ & \qquad\qquad\quad 1 \le l \le n_0 \\ 0 & \qquad \textit{otherwise} \end{cases} \tag{12}$$

A parallel recursive prediction error (PRPE) algorithm is derived from the conventional RPE algorithm by choosing the Hessian $\bar{H}(\theta)$ to be a near-diagonal matrix [3]. The algorithm can be viewed as applying the conventional RPE algorithm to each neuron in the network. The implementation of the PRPE algorithm is summarised in figure 2.

5. COMPUTATIONAL CONSIDERATIONS

In order to make reasonable timing comparisons between algorithms for use on a sequential machine, it is not enough to compare only the number of iterations required by each algorithm. The number of arithmetic operations per iteration must also be included in the comparison. While the RPE algorithm will be shown to converge in less iterations than the backpropagation algorithm, figure 3 and figure 4, it is more computationally burden. This implies that computational complexity is not a good method of judging the speed of an algorithm. To compare the speed of different algorithms a "normalized" complexity can be used, the complexity of the algorithm multiplied by a factor determined from the simulation studies. The lower the normalised complexity, the faster the algorithm.

In order to improve convergence, the RPE (PRPE) algorithm can be implemented using the QR decomposition [1,8].This algorithm has distinct advantages over conventional recursive ones. The backpropagation algorithm and the RPE (PRPE) algorithm suffer from initialization problems, figure 5, while the Q.R. decomposition algorithm does not suffer from these problems, and it is also robust numerically, as no matrix inversion is done. In addition, to improved numerical properties, Q.R. decomposition also permits the inclusion of several very flexible *forgetting* strategies. Furthermore, at each iteration the number of training patterns used can be reduced. The goal is to be able to determine which training patterns will have a very small effect on the resulting weights and avoid the rotation of these into the system of equations, thereby reducing the number of computations at this iteration and making training more efficient.

6. CONCLUSION

This work presented has confirmed the value of Neural adaptive controllers in robot manipulators, figure 6, and has demonstrated the superior performance of the recursive prediction error algorithm for estimating network parameters. Computational performance has been compared using a normalised complexity measurement.

7. REFERENCES

[1] AZIMI-SADJADI, M. et al., Supervised Learning Process of Multi-Layer Perceptron Neural Networks Using Fast Least Squares. Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing, (1990), 1381-1384.

[2] BILLINGS, S. A., JAMALLUDDIN, H. B., and CHEN, S., A Comparison of the Backpropagation and Recursive Prediction Error Algorithms for Training Neural Networks. Mechanical Systems and Signal Processing, 5 ,3(1991a), 233-255.

[3] CHEN, S., BILLINGS, S.A. and GRANT, P.M.,. Non-linear Systems Identification Using Neural Networks. Int. J. Control, 51,6(1990a), 1191-1214.

[4] KAWATO, M., et al., Hierarchical Neural Network Model for Voluntary Movement with Application to Robotics. IEEE Control Systems Magazine, (1988), 8-16.

[5] KHEMAISSIA, S. and MORRIS, A. S., Neuro-adaptive Control of Robotic Manipulators. Robotica, 11(1993), 465-473.

[6] LJUNG, L. and SODERSTROM, T., Theory and Practice of Recursive Identification. MIT Press, Cambridge. (1983).

[7] TOMOCHIKA, O., et al., Trajectory Control of Robotic Manipulators Using Neural Networks. IEEE Transactions on Industrial Electronics, 38, 3(1991), 195-202.

[8] SUN, J., GROSKY, W. and HASSOUN, M., A Fast Algorithm for Finding Global Minima of Error Functions in Layered Neural Networks. Proc. of the IEEE Int. Joint Conf. on Neural Networks, 1(1990), 7155-720.

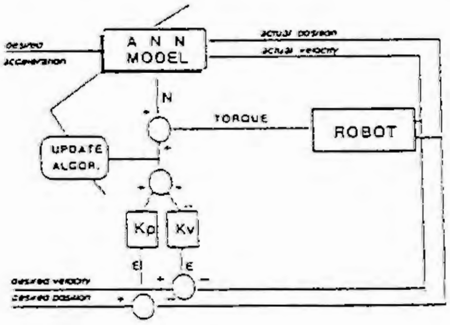


figure1: Neuro-controller

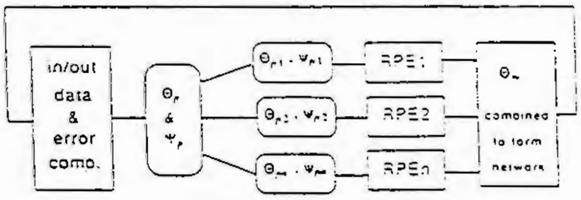

in/out
data
&
error
comp.
Θ_r
&
Ψ_r
$\Theta_{r1} \cdot \Psi_{r1}$
$\Theta_{r2} \cdot \Psi_{r2}$
$\Theta_{rn} \cdot \Psi_{rn}$
RPE1
RPE2
RPEn
Θ_w
combined
to form
network

figure2: Parallel Recursive Prediction Error Algorithm

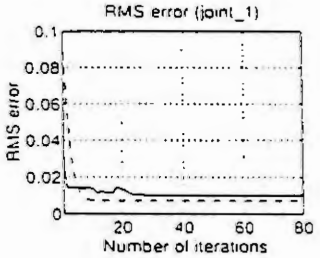


figure3: joint_1 RMS error. (solid):BP. (dashed):PRPE

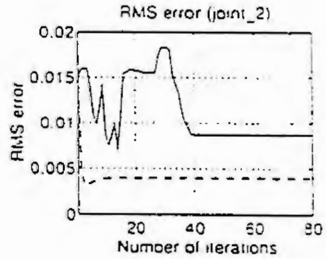


figure4: joint_2 RMS error. (solid):BP. (dashed):PRPE

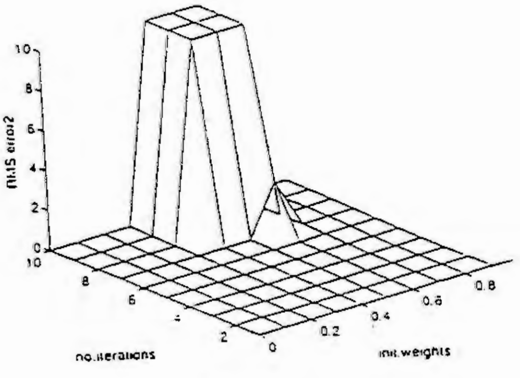


figure5: joint_2 RMS error vs initial weights

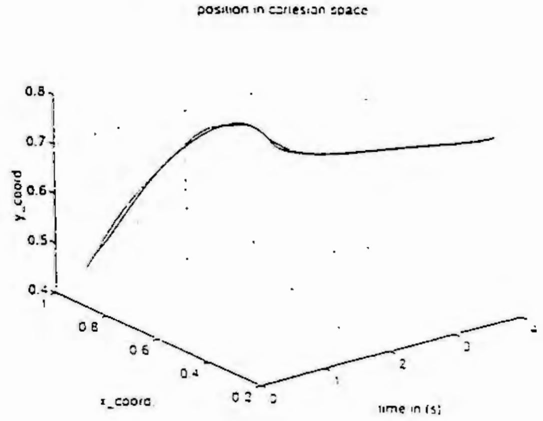


figure6: desired and actual trajectory in cartesian space

ROBUSTNESS SIMULATION STUDY OF CLASSICAL AND ADAPTIVE CONTROL APPLIED TO THE AUTOMELEC ACR ROBOTIC MANIPULATOR

Irena Jaworska, Tomasz Laski
Inst. of Industrial Electronics and Control,
Warsaw Univ. of Technology,
Koszykowa 75, 00662, Warsaw, Poland.

Spyros Tzafestas
Intelligent Robotics and Control Unit
National Technical University of Athens
15773 Zografou, Athens, Greece.

Abstract. The aim of this paper is to provide a comparative study, through simulation, of the robustness features of the classical PID controller and an adaptive control algorithm (based on a particular adaptation scheme) when applied to the Automelec ACR robotic manipulator. The robustness quality of these controllers is studied considering the manipulator motion trajectory and load variations. The results show that the adaptive controller possesses much better performance than the pure PID controller, a fact of course that was expected theoretically.

1. INTRODUCTION: THE SYSTEM ROBUSTNESS PROBLEM

Any model of a real system can only be regarded as an approximation of reality. Model errors due to such factors as unmodelled (neglected) dynamics and nonlinearities, errors of parameter identification, parameter variations during operation caused by aging and environmental conditions are collectively called *modelling uncertainty*. One can say that the uncertainty is connected with our ignorance or with an intentional simplification of phenomena. The control system uncertainty is distinguished in internal uncertainty (or model uncertainty) and external uncertainty (or signal uncertainty). The subject of robust control theory is the control system design that takes into consideration the modelling uncertainty given in a deterministic fashion. The main robust control problems are concerned with closed-loop stability and system performance (tracking, noise reduction, parametric sensitivity) in the face of uncertainty usually limited to the plant. If the plant uncertainty does not increase above the admissible bound, the desired properties of the system are robust and then the controller is called robust too. The system ability to maintain stability performance in the face of plant perturbation is called stability robustness. The plant uncertainty may sometimes be too large (structure-parameter-signal uncertainty) to allow the design of a robust controller. When the robust design techniques fall, then the literature proposes the adaptive approach for the plant structured uncertainty or the neural networks approach for the structured and unstructured uncertainty. In the last decade the development of robust control theory [1-4] has led to the formulation of the so called *robustness measures* which provide the information on the magnitude of the admissible model uncertainty which preserves closed-loop stability. Using the robustness measures, one can design the controller as "the best" one that considers the system robustness feature.

The aim of the present work is to test the practical usefulness of the robustness measures. Actually, the computer results are useful for verifying the robustness measurements of the control systems considered [2,5-6]. The paper deals with the system robustness problems and is based on the manipulator-Automelec ACR control case where classical PID and adaptive control algorithms are used.

2. THE CONTROL OBJECT: MANIPULATOR

Figure 1 presents the manipulator which usually has three degrees of freedom: φ - azimuth rotation, r - radial translation, z - vertical translation. Neglecting the resistance that depends on the velocities, one obtains two linear equations describing the vertical translation z. The equations are independent of the coordinates φ and r. Introducing state variables as follows:

$$x_1(t)=r(t),\ x_2(t)=\dot{r}(t),\ x_3(t)=\varphi(t),\ x_4(t)=\dot{\varphi}(t)$$

the dynamic behavior of the manipulator is described by the equations:

$$\dot{x}_1(t)=x_2(t),\qquad \dot{x}_2(t)=(u_1(t)+m_a[x_1(t)-r_0]x_4(t)^2+m_n x_1(t)x_4(t)^2)/(m_a+m_n)$$

$$\dot{x}_3(t)=x_4(t),\qquad \dot{x}_4(t)=\frac{u_2(t)-2m_a[x_1(t)-r_0]x_2(t)x_4(t)-2m_n x_1(t)x_2(t)x_4(t)}{\theta_t+\theta_a+m_a[x_1(t)-r_0^2]+m_n x_1^2(t)}$$

where:
$u_1(t)$=F(t): force for radial movement r [N],
$u_2(t)$=M(t): torque for azimuth rotation φ [Nm],
m_a: mass of the arm [Kg] (without hand and load),
m_n: mass of the hand and the load [Kg],
r_0: distance between the lumped mass m_n and center of gravity [m],
θ_t: inertia moment of manipulator (without arm and hand) with regard to the axis φ [m^2Kg],
θ_a: inertia moment of arm (without hand and load) with regard to the center of its mass [m^2Kg].

The mechanical parameters of the Automelec ACR manipulator are [7]:

$$m_a = 3.7\text{Kg},\ m_n = 4.6\text{Kg},\ r_0 = 0.37\text{m},$$
$$\theta_t = 0.29\text{m}^2\text{Kg},\ \theta_a = 0.09\ \text{m}^2\text{Kg}.$$

The manipulator inputs satisfy the constraints:

$$|u_1(t)| \le 15\text{N},\quad |u_2(t)| \le 5\ \text{Nm}.$$

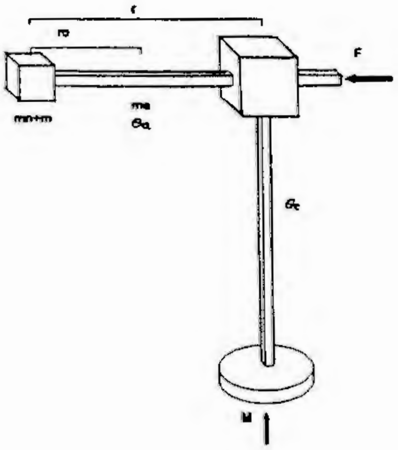


Fig.1 Automelec Robotic ACR Manipulator

3. THE CONTROLLERS

The main task of the controllers is to assure the manipulator movement from the initial to the desired points according to the reference trajectory. The complicated trajectory is approximated by a number of sections.

Figure 2 shows the structure of the manipulator control system consisting of the Automelec ACR manipulator and two separate controllers for the coordinates. The controllers are described by the following parameters (the indexes 1 and 2 refer to the coordinates r and φ respectively):

K_1, K_2 = gains of proportional action; KI_1, KI_2 = integral action; T_{d_1}, T_{d_2} = differential action.

Figure 3 shows the block diagram of the adaptive control algorithm adopted here for the manipulator. The detailed description of the adaptive controller is given in [9]. The controller is designed so as to track the reference trajectory (r,φ) of the system, and consists of two parts; the first part is to compensate the manipulator dynamics on the basis of the information on the motion equations and on the estimated parameters, the second part is a PD block where the tracking error is the input signal. The controller action is dependent on the following parameter matrices:
Γ: estimation level of manipulator model parameters (masses, distances, inertia moments, etc);
Λ: tracking trajectory exactess; K_d: gain of the differential action.

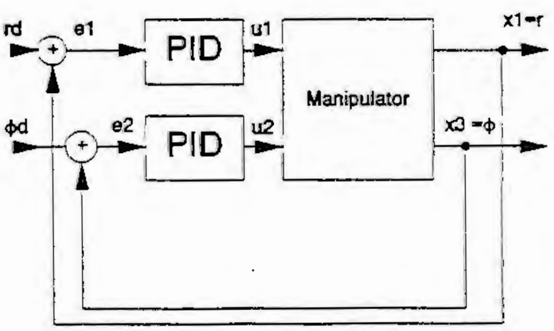


Fig.2 Block diagram of the manipulator control system with PID control

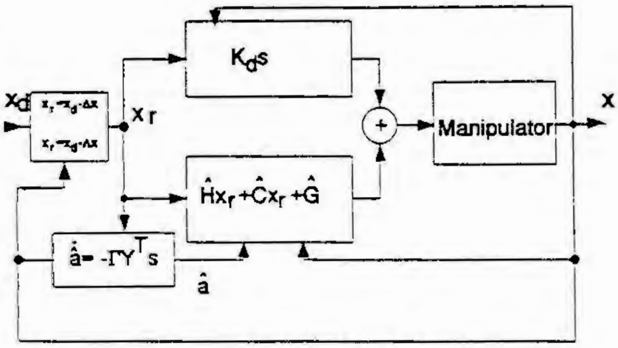


Fig.3 Block diagram of the manipulator control system with adaptive control

To determine the adaptive algorithm the generalized dynamic equation of the manipulator motion is needed, i.e.

$$H(x)\ddot{x} + C(x,\dot{x})\dot{x} + G(x) = u$$

where: x = vector of coordinates representing the manipulator position (r,φ); H(x): inertia matrix;

$C(x, \dot{x})\dot{x}$ = Coriolis and centrifugal forces vector; G(x)= gravity forces vector; u: generalized input vector (F,M).

The above equation is to be transformed as follows: Y a = u where:
a = vector of coefficients that are independent on x (masses, distances, inertia moments, etc.), and
Y = matrix dependent only on $x, \dot{x}, \ddot{x}$. The control law is given by:

$$u = \hat{H}(x)\ddot{x}_r + \hat{C}(x, \dot{x})\dot{x}_r + \hat{G}(x) - K_d(s)$$

where $\hat{H}, \hat{C}, \hat{G}$ are matrices of the manipulator motion equation after replacing the parameters $a = \hat{a}$ (where $\hat{a}$ means the estimated value for the vector a). The vector $\hat{a}$ is given by:

$$d\hat{a}/dt = \Gamma Y^T s$$

where Γ = estimation level matrix-valued parameter for the manipulator parameters, and s = vector of tracking trajectory error:

$$s = \dot{x} - \dot{x}_r = (\dot{x} - \dot{x}_d) + \Lambda \tilde{x}$$

where: $x_d, \dot{x}_d, \ddot{x}_d$ = desired trajectory. To determine the tracking error s use is made of the modified velocity trajectory $\dot{x}_r$ and acceleration $\ddot{x}_r$ taking into account their deviation from the desired trajectories $(\dot{x}_d)$ and $(\ddot{x}_d)$. The deviations are defined as:

$$\dot{x}_r = \dot{x}_d - \Lambda \tilde{x}, \qquad \ddot{x}_r = \ddot{x}_d - \Lambda \tilde{x}$$

where $\tilde{x}, \dot{\tilde{x}}$ = tracking errors of position and speed:, i.e. $\tilde{x} = x - x_d$ and $\dot{\tilde{x}} = \dot{x} - \dot{x}_d$

The adaptive algorithm is presented in [8] with all details. The matrix-valued parameters K_d, Λ, and Γ are optimal and must satisfy the matrix positiveness condition. In practice the matrices are diagonal with elements greater than zero.

4. COMPUTER SIMULATION

All simulations have been performed with linear reference trajectory for the coordinates (r, φ) from the start point X_0 to the end point X_d in the T_{ref} time.

Start point X_0: φ=0 rad, r=0.2 m; End point X_d: φ=π/4 rad, r=0.5 m.

The plots presented in Figs.4-7 are observed for Tref=2sec. The parameters for the adaptive controller used in the simulation are as follows: $\gamma 1 = \gamma 2 = \gamma 3 = 0.3$, $\lambda 1 = 0.05$, $\lambda 2 = 1$, $K_{d_1} = K_{d_2} = 240$.

The best control quality was acheived for the above parameter set. The simulations have been carried out using an IBM PC software package developed by the authors. The package has been constructed for didactic goals; two different manipulators, some actuators and sensor models are possible to be used with the PID and adaptive control algorithms (currently research is carried out to accomodate a neural network controller).

5. SYSTEM ROBUSTNESS ANALYSIS: SOME REMARKS

The influence of the PID regulator parameters to the trajectory of the system coordinates was studied. Respecting the limits of the control signals, the best parameter set for the PID controller obtained is the following: K_1=30, K_2=35, KI_1=KI_2=0, T_{d_1}=T_{d_2}=0.8. The PID controller with the above parameters is compared to the adaptive control results; the system's robustness is compared. Actually, the system is under perturbation, since there is different load of the manipulator arm.

Figures 4 and 5 illustrate the load influence to the movement trajectory for PD control algorithm (a,c) and for the case when the adaptive algorithm is used (b,d); the system robustness of the adaptive control is better. Figures 6 and 7 allow one to note how the preciseness of the trajectory tracking depends on the control algorithms: the PD controller does not give such good results as the adaptive one.

The authors have been extensively worked on the robustness measurements. Using the empirical description of the robustness, together with suitable mathematical relations, we assigned numerical values to the particular robustness characteristics. The verification of the measurement data (measurement scale) needs the computer simulation of the system measured. The computer simulations presented in this paper aimed at satisfying this task. A future paper will deal with robustness measurements and their verification procedure based on the computer experiments for the manipulator control systems presented.

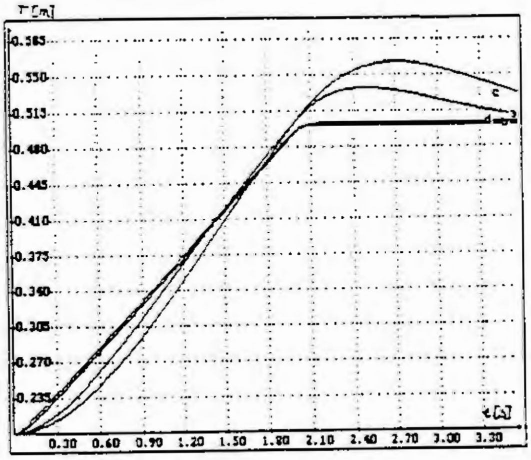


Fig.4 Change of manipulator load: Radial motion of arm for m_n=3.6Kg (a,b) and m_n=8.6Kg (c,d)
a,c - controller PID, b,d - adaptive controller

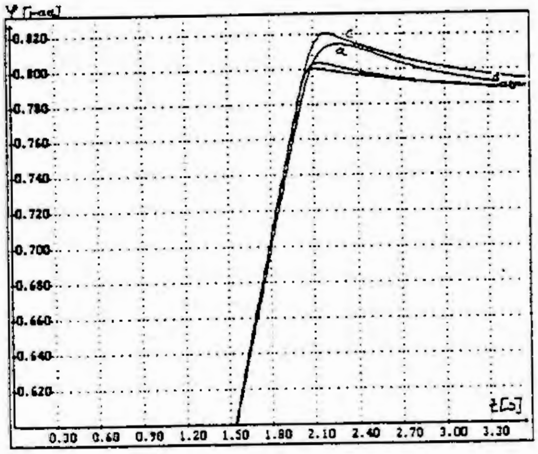


Fig.5 Change of manipulator load: Azimuth rotation of arm for m_n=3.6Kg (a,b) and m_n=8.6Kg (c,d)
a,c - controller PID, b,d - adaptive controller

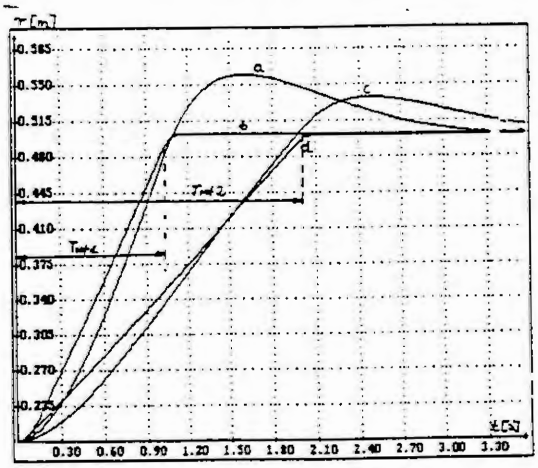


Fig.6 Reference trajectory tracking preciseness of radial motion for T_{ref1}=2s (a,b) and T_{ref2}=Is (c,d):
a,c - PID control, b,d - adaptive control

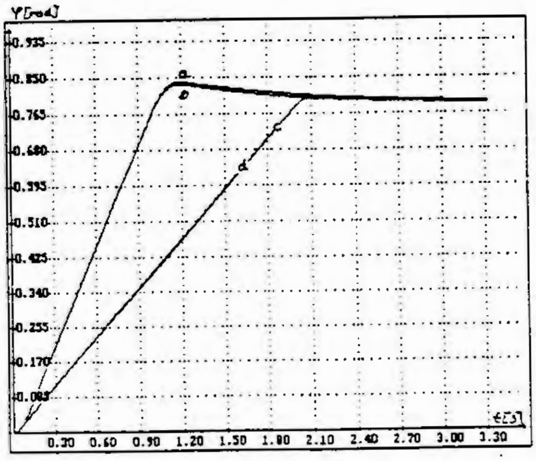


Fig.7 Reference trajectory tracking of azimuth rotation for T_{ref1}=2s (a,b) and T_{ref2}=1s (c,d):
a,c - PID control, b,d - adaptive control

6. REFERENCES

[1] Dorato P., A historical review of robust control. IEEE Contr. Syst. Magaz., Vol.7, No.2 (1987) 44-47.

[2] Jaworska I. and Kurylowicz A., Modelling uncertainty estimation to maintain control systems stability. IFAC Symp. on Estimation and Parameter Identification, Budapest, July, (1991).

[3] Maciejowski J., Multivariable feedback design. Addison-Wesley Publishing Company, (1983).

[4] Kurylowicz A, Jaworska I. and Tzafestas S., Robust Stabilizing Control: An Overview. In: (S. Tzafestas Ed.) Applied Control: Current Trends and Modern Methodologies, Marcel Dekker (1993) 289-324.

[5] Tzafestas S., Dritsas L. and Kanellakopoulos J., Robust Robot Control:A Comparison of Three Techniques Through Simulation. In:(P.Breedveld Eds.) Modelling and Simulation, J.C.Baltzer (1989) 255-260.

[6] Jaworska I. and Tzafestas S., Robust Stability Analysis of Robot Control Systems. Robotics and Autonomous Systems, Vol.7 (1991) 285-290.

[7] Geering H., et al., Time-optimal motions of robots in assembly tasks. IEEE Trans. on Auto. Contr., Vol.AC-31 No.6 (1986) 512-518.

[8] Słotine J. and Li W., Adaptive manipulator control: Acase study. IEEE Trans. on Auto. Contr., Vol.33, No.11 (1988), 995-1003.

Software Engineering and Data Models of Mechanical Systems

Michael Hocke, Roland Rühle, Jochen Seybold
Computer Center of the University of Stuttgart (RUS), Allmandring 30, D-70550 Stuttgart

Abstract

An object-oriented data model is defined to describe parametrized multibody systems. Class descriptions for an object-oriented database and a file format for data exchange files are directly derived from the data model. The development of an open, modular and extendable software package for the synthesis and design of mechanical systems is based on the data model.

1 Introduction

It was the goal of the nationwide German research project "Dynamics of Multibody Systems" to develop a powerful software package for the analysis and design of multibody systems. The software package provides a modern system architecture and advanced data management concepts. It is designed as an open and modular system which is extendable by the application engineer. The central data storage facility is an object-oriented database which provides standardized interfaces for the data exchange between the modular components. Therefore existing modules may be replaced and new modules may be added without affecting other modules.

A multibody system is defined and stored on database according to an object-oriented data model which is particularly described in [3]. The data model is independent from any specific multibody formalism and allows the description of rigid bodies, deformable bodies (see [6]) and geometric properties as needed for high speed animation (see [1]). Since the data model supports the parametrization of a multibody system, synthesis and design methods may be applied.

2 A Neutral Object-Oriented Data Model

A data model describes the conceptual database which is an abstraction of the real world, independent from the implementation on physical devices. The discussion of different data models in [5] shows, that engineering applications are described by an *object-oriented* data model in a natural and efficient way. Hence, a neutral object-oriented data model due to Ullman [5] is selected to describe multibody systems. It is characterized by the following properties:

1. *Encapsulation:* The internal data structure of the object is completely hidden. Generic operations are provided by the database system to access the data objects.

2. *Object Identity:* Different objects are distinguished by a unique identifier. Modification of the internal state and the data values does not affect the identity of the object.

3. *Complex Objects:* Complex objects are explicitly defined, using data types and object types as components. A complex object is an aggregation of several other objects.

4. *Inheritance:* Inheritance is used to construct type hierarchies. A derived class is called subclass, since it inherits the components and methods of the superclass. New components and methods may be defined for the derived class, existing methods may be redefined.

An object-oriented data model describes the structure of the objects and their behaviour by classes. A class description consists of the scheme description of the *object type* and the specification of the *methods*, also called operations. Both aspects are discussed next.

At a basic level, the data model supports *elementary data types* which are implicitly defined by the database system. These are: integer value (*int*), single and double precision floating point value (*real,double*), character string of variable or fixed length (*char,name*), identifier of a component (*sname*), parameter which either stores a name or a double precision value (*dparam*), and name of a time dependent input signal (*input*).

Furthermore, *implicit object types* are supported. These are multi-dimensional arrays with fixed or variable length of the elementary data types. *Explicit object types* are recursively defined by applying the following rules according to Ullman [5]:

1. Let $T_1, \ldots, T_n$ be *implicit* or *explicit object types*, then **recordof** $(T_1, \ldots, T_n)$ defines a *composed* or a *complex object type*. Each component of the record is assigned a unique name, a data type and a short description text. Composed object types additionaly support lower and upper bounds, a default value and a physical unit.

2. Let T be an *object type*, then **setof** (T) also defines a valid *object type*. A set is an unordered collection of any number of objects of class T. Each object in the set is assigned a unique name and a short description text. A set of objects does not necessarily define a new class, since a set may also be a valid component of a complex object.

3. Let T_{sup} be an *object type* defined according to rule 1 and let $T_1, \ldots, T_n$ be *implicit* or *explicit object types*, then **recordof** (**subtypeof**$(T_{sup}), T_1, \ldots, T_n)$ also defines a valid *object type*. The definition **subtypeof**(T_{sup}) states that a new object type is derived by inheritance from the supertype T_{sup}. The derived object type is called subtype.

Inheritance is an important aspect of the data model as it enables the extension and modification of classes without modifying the source code of existing methods. It also enables the introduction of new classes for specific problems, e.g. for frames, joints, forces and sensors.

A class description also includes the specification of the available methods. *Generic methods* are used to manage and manipulate data objects and to access data values, i.e. access to objects in main memory (initialize and delete objects; read and write components and data values; query attributes), transfer of objects from database to main memory and vice versa, and manipulation of objects on database (delete, copy, rename, and browse objects). These methods are independent of the application and are introduced by the database system at the top of the inheritance hierarchy. They are applicable to objects of all classes. *Class specific methods* are applicable to objects of a specific class only. These methods perform the computations which depend on the application, and must therefore be provided by the application engineer. Class specific methods are inherited from the superclass and may be redefined within the subclass.

3 Data Model for Multibody Systems

A data model for multibody systems is derived from the neutral data model. It describes a multibody system by predefined classes and methods. Due to the openess and extendability of the system, the user may introduce new classes and methods for special multibody applications.

Description of Dynamic Systems. A multibody system is treated as a special dynamic system which is described as a parametrized input/output block with input signals $u(t)$, output signals $y(t)$, parameters $p = const$, and internal signals, depending on the mathematical description. This allows the utilization of general purpose methods for dynamic systems, e.g. synthesis and design methods.

Class *dsblock* describes the input/output properties of a block. The complex object type is defined according to rule 1 of the neutral data model: $dsblock =$ **recordof** $(input, output, param)$. A time signal is described by a name, a unit, and a short description text. A parameter additionaly has a default value and lower/upper bounds. Independent parameters are assigned actual values while dependent parameters are calculated by evaluation of a mathematical expression.

Description of Multibody Systems. A multibody system consists of material bodies, connected by constraint elements (joints) and coupling elements (forces). It is treated as a special input/output block, i.e. it has all the properties of class *dsblock* and additional ones. Since a multibody system may be utilized whenever a dsblock is required, a broad range of methods becomes available for multibody systems, e.g. simulation and parameter optimization.

Class *mbs* describes the elements of a multibody system. The complex object type is derived by inheritance from class *dsblock* according to rule 3 of the neutral data model and is defined as: mbs = **recordof** (**subtypeof**($dsblock$), $global$, **setof**($part$), **setof**($interact$)). Class *mbs* inherits the input/output signals and the system parameters from superclass *dsblock*. It additionaly defines the basic elements of a multibody system, i.e. a set of parts and a set of interaction elements, and the global data of the multibody system, at present only the gravitational acceleration. A detailed description is given in [3].

Class *part* defines a rigid body as a collection of coordinate systems, called frames. A frame is described with respect to a reference frame on the same part and provides operations to calculate the positional vector and the rotation matrix from the reference frame to the frame. A rigid body with mass and inertia properties is described by class *rigid*, a subclass of class *part*. Class *rigid* inherits the definition of the frames and additionaly defines the mass, the center of mass and the inertia tensor of the part. In [6], Wallrapp describes deformable bodies in modal representation by class *modal*. Class *modal* is a subclass of class *part*.

Interactions between two frames on different parts are described by class *interact*. Class *interact* defines the names of the two frames and parts and the type of the interaction. Between two parts, only one joint but several force elements and sensors is allowed. Classes *joint*, *force*, *sensor* are used as superclasses of more specific elements. These classes do not have components but define virtual methods which have to be redefined in the subclasses. Predefined subclasses are provided by the data model, e.g. to describe a revolute joint by class *revolute*, a spherical joint by class *sphere*, and a translational spring force element by class *springt*.

Description of Geometric Properties. Visualization methods require powerful graphic workstations and geometry data to achieve realistic images of the multibody model. The geometry data are either extracted from CAD systems, e.g. via standard data exchange files (Iges, Step/Express), or edited by special geometry editors. A standardized interface to store the geometry data of a multibody system on database is provided by an extension of the data model which is particularly described in [1].

The geometric properties of a multibody system are described by class *g3mbs* which is derived by inheritance from class *mbs* according to rule 3 of the neutral data model and is defined as: $g3mbs$ = **recordof** (**subtypeof**(mbs), $g3global$, **setof**($g3part$), **setof**($g3inter$)). Class *g3mbs* inherits the components of class *mbs* and therefore describes a multibody system as well as the geometry data of the elements. No modification of existing methods is required, since an object of class *g3mbs* can be utilized whenever an object of class *mbs* is demanded.

Class *g3mbs* consists of three components: the global graphic information, the geometry data of the parts and frames, and the geometry data of the joints and forces. The global graphic information provides default values for visualization modules, at present the viewing and projection parameters, light model and light sources.

The geometry of a part is described by a *planar face model* in terms of vertices, edges, and normal vectors for lighting. The planar face model defines a standardized interface between CAD systems and the graphics hardware: CAD systems internaly use different geometry models, e.g. Constructive Solid Geometry or Boundary Representation. These CAD models are easy to convert into a planar face model. Basic graphic packages such as Iris GL and Phigs provide programming interfaces for planar faces because it is well suited for high speed visualization.

The geometry of frames, joints and forces is described by *parametrized shapes*. These are predefined geometries which are scaleable by parameters to fit into the actual multibody model. The geometries visualize the mechanical characteristics of the elements, e.g. orientation of the frames, degree of freedom and kinematic movement of the joints. If new classes for specific elements are introduced by the application engineer, the according shape for the visualization of the elements must also be provided.

Integrity Constraints. The data objects of a multibody system stored on database must satisfy several integrity constraints on different levels.

Elementary integrity constraints relate to a single object and are defined in the class description. These constraints are guaranteed by the database system, e.g. correctness and completeness of the data types and object types, check of lower and upper bounds for numerical values.

Further integrity constraints exist for the multibody model which cannot be expressed by class descriptions. These constraints are checked by special methods, e.g. the inertia tensors must be positive semidefinite, the kinematic connection structure must be complete and a consistent position of the multibody system is computed if closed kinematic loops are defined.

4 Implementation of the Data Model

The object-oriented data model for multibody systems has been implemented at the RUS using RSYST, which is an open and modular software system for the development of large scientific application programs, see [4] for details. It provides several components, such as an object-oriented database, a window-oriented user interface, dynamic memory management, error handling and output system. All components are accessible through a monitor program by the user and through a complete set of programming interfaces by the application engineer.

The data model is realized in the following way: first of all, a scheme description for the RSYST database is defined according to the class descriptions in [3]. Furthermore, methods for multibody systems are defined and realized. In particular, a window-oriented module to edit the data objects of a multibody system on database is implemented. This module provides an interface to other multibody packages, since it parses data exchange files and generates the corresponding data objects on database. The implementation of the multibody software package is described in [2].

5 Summary

An object-oriented data model for multibody systems was presented which is the core of a multibody system program package. The program package has three unique features:

1. The data model is independent from a specific multibody algorithm and therefore defines a *neutral format* for the exchange of multibody data between different multibody programs.

2. Multibody systems are described as *input/output blocks* to allow easy incorporation in modeling, analysis, and design packages for e.g. connecting multibody systems with control units. The multibody system no longer is the central part of the modeling process but just one block among others.

3. The data model is *parametrized,* i.e. each constant input data (e.g. mass, spring constant) can be given either a *numeric* or a *symbolic* value.This allows the utilization of synthesis methods to determine the actual values of symbols, e.g. by parameter optimization.

References

[1] Hocke, M.; J. Seybold; U. Wagner: Visualisierung und Animation von Mehrkörpersystemen unter Verwendung eines objektorientierten Geometrie-Datenmodells, RUS-18, Oktober 1993.

[2] Hocke, M.; R. Rühle; M. Otter: An Open Software Environment for the Analysis and Design of Multibody Systems. In: Schiehlen, W. (ed.): Advanced Multibody System Dynamics – Simulation and Software Tools. Kluwer Academic Publisher, 1993.

[3] Otter, M.; M. Hocke; A. Daberkow; G. Leister: An Object-Oriented Data Model for Multibody Systems. In: Schiehlen, W. (ed.): Advanced Multibody System Dynamics – Simulation and Software Tools. Kluwer Academic Publisher, 1993.

[4] Rühle, R.: RSYST – Ein Softwaresystem zur Integration von Daten und Programmen zur Simulation wissenschaftlich-technischer Systeme. Rechenzentrum der Universität Stuttgart, RUS-5, März 1990.

[5] Ullman, J. D.: Principles of Database and Knowledge-Base Systems, Volume 1. Computer Science Press, 1988.

[6] Wallrapp, O.: Standard Input Data of Flexible Members in Multibody Systems. In: Schiehlen, W. (ed.): Advanced Multibody System Dynamics – Simulation and Software Tools. Kluwer Academic Publisher, 1993.

A Systematics of Modelling Mechatronic Systems

Frank Junker, Joachim Lückel
University of Paderborn
MLaP - Mechatronics Laboratory Paderborn
Prof. Dr.-Ing. Joachim Lückel
Pohlweg 55, D - 33098 Paderborn, Germany

Abstract

Mechatronic systems consist of components from different technical disciplines. They require an integrated design of all components. This leads to systems of high complexity, with the mechanical and informational components at the centre. The modelling systematics presented here allows an easy and flexible exchange of any desired subsystems on the basis of their dynamical equations. This way of modelling considerably facilitates analysis and synthesis of complex systems and supports distributed digital simulation oriented according to the physical structure.

Introduction

On closed inspection, mechatronic systems consist of independent functional groups which reflect the functionalities provided by the construction engineer. On the one hand, these systems consist more and more of components from different technical disciplines, such as mechanics, electrical engineering, and hydraulics; on the other hand, computer science and information processing can hardly be dispensed with if high efficiency is to be assured. These decentralized functional groups can be called **MFMs** (Mechatronic Function Modules). Their development and modelling require interdisciplinary treatment and appropriate computer-integrated design tools suitable for a specific methodical work.

Therefore the Department of Automatic Control in Mechanical Engineering at the University of Paderborn has developed a Computer-Aided Mechatronic Laboratory (**CAMeL**) [5] which allows access to the computer-integrated development tools for the design cycle of mechatronic systems. **CAMeL** requires the models to be represented in the symbolic model description language **DSL** (**D**ynamic **S**ystem **L**anguage) which is based on the explicit nonlinear state-space representation and allows description of complex, nonlinear, hierarchically organized systems.

Differential equations of 1st order and/or algebraic equations can easily be formulated in block diagrams. Description of mechanical subsystems requires special treatment. For this purpose, a modelling method based on **MBS** (multibody systems) is presented here. The recursive Newton-Euler algorithm developed by Bae and Haug [1, 3] is applied. This efficient (O)N algorithm (it requires only very low computation costs) makes use of the principle of virtual work (variational equations of motion) and is based on a systematic transformation of the equations of motion described in Cartesian coordinates into some form of relative coordinates.

For open-loop systems, this yields the explicit state-space form with a minimum number of degrees of freedom. In the case of closed-loop systems, the equations of motion are combined with a minimum set of algebraic constraint equations [2, 3]. The resulting differential-algebraic description can be transformed with the help of an index reduction and a stabilization of constraints [4] into the explicit state-space form. This approach makes possible multi-level and hierarchical couplings, from elementary multibody components (e. g. rigid bodies) through aggregates up to the entire system.

The Recursive Formulation Method

The recursive formulation of the kinematics can be described by means of two neighbouring rigid bodies *i* and *j*. A detailed presentation can be found, among others, in [1, 2, 3]. The algorithm requires information on the absolute interrelations between position, velocity, and acceleration of each rigid body in relation to the inertial coordinate system. These interrelations are then transformed into a relative coordinate system. The resulting kinematic relationships for the rigid body *j* of a chain of joints are the following:

$$\begin{bmatrix} \dot{r}_j \\ \omega_j \end{bmatrix} = \begin{bmatrix} E & -\tilde{r}_{ij} \\ O & E \end{bmatrix} \begin{bmatrix} \dot{r}_i \\ \omega_i \end{bmatrix} + \begin{bmatrix} -\tilde{h}_{ij}\, s_{ji} & d_{ij} \\ h_{ij} & O \end{bmatrix} \begin{bmatrix} \dot{\Theta}_{ij} \\ \dot{\alpha}_{ij} \end{bmatrix} , \tag{1}$$

$$Y_j = B_{ij1}\, Y_i + B_{ij2}\, \dot{q}_{ij} \; .$$

With the vector of the absolute velocity Y_i of the rigid body i and the vector of the relative velocity $\dot{q}_{ij}$ between the rigid bodies i and j. The derivative of the velocities with regard to time yields the acceleration:

$$\dot{Y}_j = B_{ij1}\, \dot{Y}_i + B_{ij2}\, \ddot{q}_{ij} + \dot{B}_{ij1}\, Y_i + \dot{B}_{ij2}\, \dot{q}_{ij} \quad . \tag{2}$$

The variational form of the Newton-Euler equation of motion [1, 3] holds for all kinematically admissible variations δZ_i and for a general constrained multi-body system which consists of n rigid bodies and can be written as:

$$\sum_{i=1}^{n} \left\{ \delta Z_i^T (M_i \dot{Y}_i - Q_i) + \sum_{(j,k) \in I_1} \delta Z_i^T \Phi_{Z_i}^{(j,k)T} \lambda_{jk} \right\} = 0 \quad . \tag{3}$$

The variation vector $\delta Z_i = [\delta r_i^T \quad \delta\pi_i^T]$ consists of a virtual translation δr_i and a virtual rotation $\delta\pi_i$. The acceleration vector $\dot{Y}_i = [\ddot{r}_i^{\,T} \quad \dot{\omega}_i^{\,T}]$ represents both of them. Matrix M_i and vector Q_i respectively represent the mass matrix and generalized forces. Closed loops in a system are opened when a joint is imagined to be cut and its effect replaced by a set of cut-joint constraint equations which can be expressed as $\Phi^{(n,n+1)} = 0$. These constraint equations are then differentiated twice with respect to time to be included in the equations of motion by means of Lagrange multipliers $\lambda_{n,n+1}$:

$$\ddot{\Phi} = \Phi_{Z_n} \dot{Y}_n + \Phi_{Z_{n+1}} \dot{Y}_{n+1} - \gamma = 0 \, . \tag{4}$$

Lagrange multipliers can be eliminated at the junction body where the loops are closed, in order to eliminate the dependence on the multiplier in further inboard reduction of the variational equation.

Block-Oriented Representation of Elementary Mechanical Elements

The basic equations of the recursive algorithm employed here can be attributed to three calculation steps: firstly, a forward recursion for calculating the kinematic relations, a backward recursion for mass and constraint forces, and another forward recursion for calculating kinetics. With the model description language DSL, these three computational steps can be represented in a block-oriented way [6]. By means of appropriate couplings of algorithmic basic blocks presented - couplings which reflect the variable-dependence of the algorithm -, one obtains so-called elementary mechanical components, such as a rigid body with revolute joint (Fig. 1). These elements are available in a block-oriented form with uniform interfaces and can be further coupled in any desired configuration, thus making it possible to build up more complex mechanical structures; this will be dealt with in the following examples.

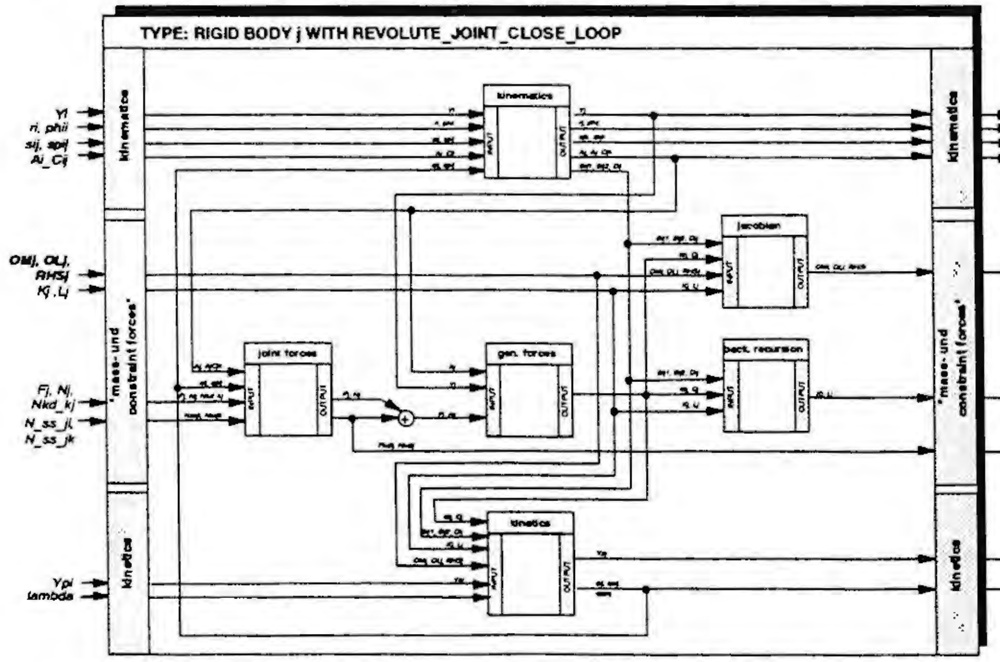


Fig. 1 : Block-Oriented Representation of the Recursive Algorithm

A Crank Slider as a Multi-loop Mechanism

As a simple example, a planar crank slider mechanism is used to demonstrate how to treat multi-loop mechanisms with this kind of modelling method. On the left-hand side of Fig. 2, the structure and the topological graph are displayed. Here, every node represents a body, while an edge represents a joint between a pair of bodies. In the graph, *R* and *T* denote revolute and translational joints respectively. A loop is a path where the beginning and the ending nodes are identical. The structure presented includes two coupled loops, loop I (I-1-2-3-4-I) and loop II (I-1-2-3-5-I). The right-hand side gives an idea of the block-oriented description.

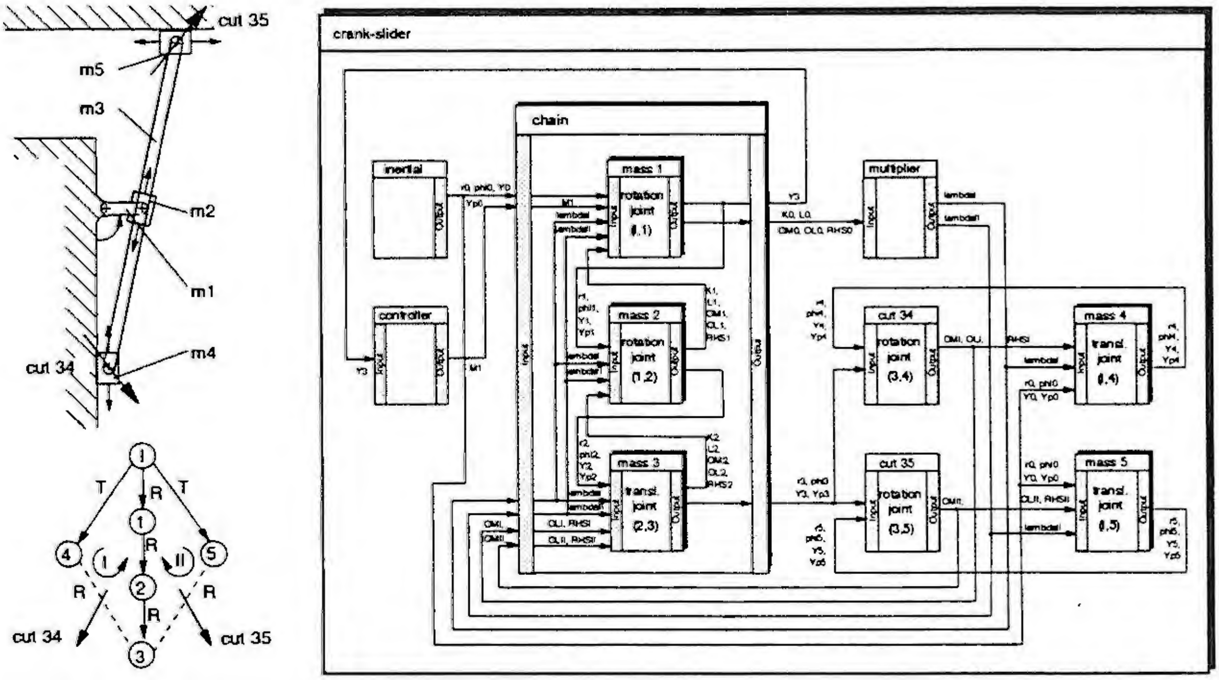


Fig. 2 : Block-Oriented Representation of a Multi-loop Mechanism

When the hierarchical system is built up, the elementary blocks described in the way detailed above can be coupled via their system inputs resp. outputs. The management of closed-loop systems require further, algorithmic basic blocks which will be presented in short in the following:

Each mass (*m1, m2, ...*) - with one joint respectively - is represented by a corresponding elementary mechanical block *"mass i"*, the inertial system by the block *"inertial"*. The *"controller"* assures that *m1* keeps to a predetermined reference velocity. By cutting an edge into each loop a closed-loop system can be opened to form a tree structure (spanning tree). In this example, cuts can be made at joints (3, 4) and (3, 5), to form a spanning tree with three chains, chain I-4, chain I-1-2-3, and chain I-5. The corresponding cut-joint constraint equations (4) are put into the algorithmic blocks *"cut34"* and *"cut35"*. The Lagrange multipliers are calculated in the algorithmic block *"multiplier"* corresponding to the cut-joint constraints of each loop.

The latter are evaluated numerically at every discrete time step, e. g. during digital simulation. The sequence of these evaluations will be recognized by the DSL compiler in the light of the variable-dependencies. Thus, the entire mechanical system is coupled anew at each main step of the integration.

The recursive algorithm applied is especially suitable for complex systems because the numeric expense rises only in linear accord with the system order, even on a serial computer. In order to save further computational expenses, parallel computation is employed. In the present approach, the equations of each chain of the spanning tree are independent of one another, so they can be solved simultaneously on the basis of the augmented system equation of motion.

The Modular Structure of a Six-Axis Robot

The characteristic features of the mechanical robot construction, as shown in Fig. 3, are aggregates and functional groups (e. g. DC motor, gear, ...), marked by modular and hierarchical structuring. The mechanical aggregates formulated in this way can be complemented by components of electrical and information processing (e. g. controller, observer, ...) and thus become decentralized functional groups, called MFMs (Mechatronic Function Modules). Modularity of this kind is ideally suited to exchange individual components within the construction, but also within the corresponding dynamic model [7]. E. g. the mechatronic function module *"robot joint"* contains components (sensors, controllers, observer, ...) designed to compensate for unwelcome features, such as backlash, friction, and elasticity in the gear, already within the decentral module.

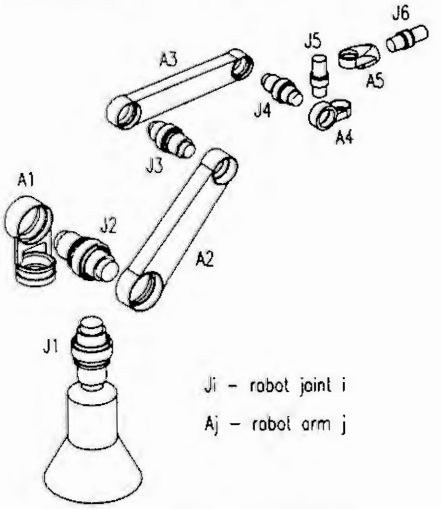


Fig. 3 : Modular Structure of a Robot

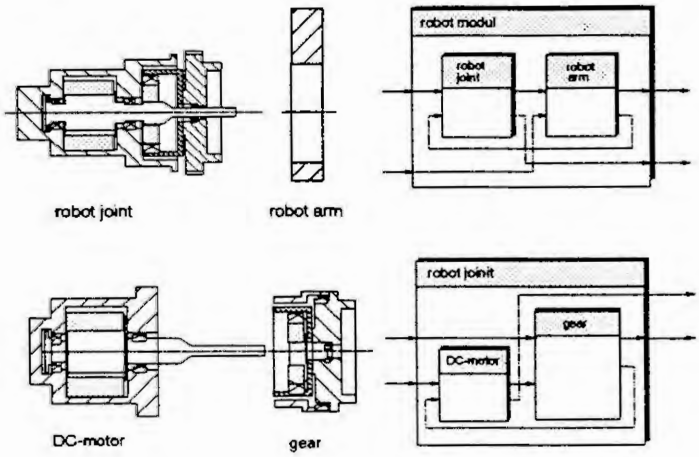


Fig. 4 : Block-Oriented Representation of some Robot Subsystems

The six-axis robot presented here consists of four main structural components, the *"vertical"*, *"shoulder"*, *"elbow"*, and *"hand" axes*, which are designed in a way as to form one construction series. Each main structural component forms a mechatronic function module, consisting of a *"robot joint"* with a *"DC motor"*, a *"Harmonic Drive gear"* and the corresponding electric and electronic components, such as sensors, controllers, etc.

Results

Development and modelling of mechatronic systems and functional modules require interdisciplinary treatment and appropriate computer-integrated design tools (e. g. for nonlinear simulation, linearization, and linear analysis as well as controller optimization and generation of controller code) and suitable for a specific methodical work supported by CAMeL. These tools require the uniform description language DSL which is based on the explicit nonlinear state-space representation.

The systematics of modelling mechatronic systems presented here allows the mathematical model to closely follow the structure of the technical system (Fig 2, 4). It also allows, especially for the mechanical components, an independent generation of aggregates and functional groups of a system, management and then recoupling with computer support in a modular und hierarchical way.

The function-oriented approach to the problem and the successive build-up of the model on the basis of reusable subsystems makes possible a simplified and reasonable (with regard to costs) treatment of complex tasks.

References

1 Bae, Dae-Sung; Haug, Edward J.: A Recursive Formulation for Constrained Mechanical System Dynamics, Part I: Open Loop Systems, Mech. Struct. & Mach. 15, 3 (1987), pp. 359-382.

2 Bae, Dae-Sung; Haug, Edward J.: A Recursive Formulation for Constrained Mechanical System Dynamics, Part II: Closed Loop Systems Mech. Struct. & Mach. 15, 4 (1987-88), pp. 481-506.

3 Bae, Dae-Sung: A recursive Formulation for Constrained Mechanical System Dynamics, Ph.D. Thesis, The University of Iowa, 1986.

4 Baumgarte, J.: Statilization of constraints and integrals of motion in dynamical systems, Computer Methods in Applied Mechanics and Engineering 1 (1972), pp. 1-16.

5 Jäker, K. P.; Klingebiel P.; Lefarth, U.; Lückel, J.; Richert, J.; Rutz, R.: Tool Integration with a Computer-Aided Mechatronic Laboratory (CAMeL), Preprints, 5th IFAC/IMACS Symposium on Computer Aided Design in Control Systems CADCS '91, Swansea, Wales, July 1991.

6 Junker, F.: Modular-hierarchisch strukturierte Modellbildung mechanischer Systeme, Archive of Applied Mechanics (to appear).

7 Lückel, J.; Moritz, W.; Neumann, R.; Schütte, H.; Wittler, G.: Development of a Modular Mechatronic Robot System, Second Conference on Mechatronic and Robotics, Proceedings edited by M. Hiller and B. Fink, Duisburg/Moers, Germany, Sep. 27-29 1993, pp. 485-500.

Modelling of Mechatronic Systems by an Object–Oriented Data Model

U. Neerpasch, W. Schiehlen

Institute B of Mechanics, University of Stuttgart, Pfaffenwaldring 9, D–70550 Stuttgart

Abstract

An object oriented data model is defined to describe multibody systems. Extensions for modelling mechatronic elements like sensors and actuators within the multibody system as well as interfaces to other dynamic systems have been developed. An implementation in a neutral modelling kernel and a format to store the description of a multibody system on a data exchange file are directly derived from this data model. Data converters transfer these data to several multibody formalisms.

1 Introduction

In this paper an object–oriented data model for multibody systems with extensions to mechatronic elements is described. A multibody system defined by this data model consists of rigid bodies connected by ideal joints and force elements. Measurement elements called sensors deliver internal, time dependent quantities like distances, accelerations as well as forces. Position actuators are defined to model drives within the multibody system.
The data model is independent of a specific multibody program and can therefore be used as a neutral format for the exchange of multibody system descriptions. The datamodel is given as a block with input/output interfaces for the connection to control units or other elements.

2 Object–Oriented Data Model

Engineering applications, such as multibody systems may be described by an object oriented data model in a natural and efficient way, following the discussion of a data model by Otter, Hocke, Daberkow, Leister [1]. The data model for the description of multibody systems is based on the simple, neutral, object–oriented data model due to Ullman [2].

In an object–oriented data model, the structure of the objects and their behaviour are described by classes. A class description consists of two parts: the scheme description of the object type and the specification of the available methods. Both aspects of a class are discussed in more detail in the following sections.

Object types:
The first part of a class description consists of the definition of the object type. At a basic level the data model supports a set of elementary data types, like integer values, real values or character strings. Furthermore, multi–dimensional arrays of the elementary data types are supported. New object types are defined by building composed or complex object types out of already defined object types (**recordof**) or by building collections of a number of objects of the same class (**setof**) and by deriving class descriptions by inheritance from superclasses, according to Ullman [2]. Applying these rules, arbitrary complex object types can be defined based on a small set of elementary data types.

Methods:
The second part of a class description consists of the specification of the available methods.

The data model distinguishes between administrative methods like creating, deleting or manipulating objects and class specific methods which can only be applied to objects of a specific class.

3 Description of Multibody Systems with Mechatronic Elements

Multibody Systems consist of material bodies (parts) connected by constraint elements (joints) and coupling elements (forces, torques), see Schiehlen [3]. They are well qualified for the dynamical analysis of machines, mechanisms, robots, and vehicles.

A multibody system is defined by an object of class *mbs* (multibody system), which is derived form the class *block*. Class *block* describes a general dynamical system and is characterized by input and output–signals, parameters, and internal signals which depend on the mathmatical model of the block.

A multibody system is essentially composed of the two basic elements: class *part* and class *interact*, see figure 1.

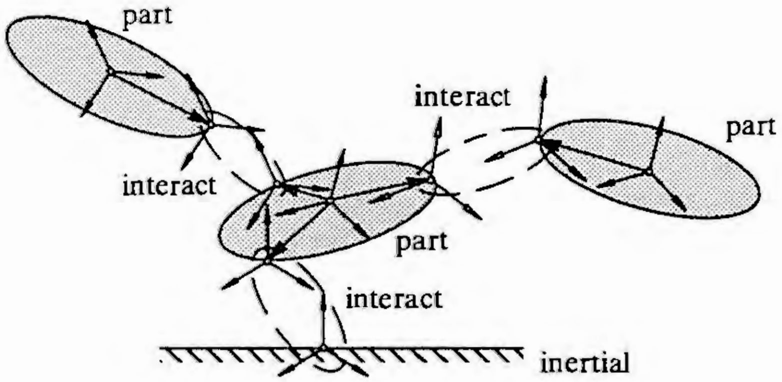


Figure 1: Elements of a multibody system

Class *part* defines a rigid body as a collection of coordinate systems, or frames, respectively. An object of class *frame* is described with respect to a reference frame on the same part and provides operations to evaluate the position vector and the rotation matrix from the reference frame to the frame. Class *rigid* is a subclass of class *part*. It has all the characteristics of the superclass and additionally the component **body** of class *body*. Class *body* is used to characterize the mass and the inertia tensor of the rigid body.

An object of class *interact* describes the interaction between one frame on a first part and one frame on a second part. Class *interact* has the components **connect** and **member** which form the class *connect* and *member*, respectively. The names of the two parts and two frames of the interaction element are stored in the object of class *connect*. The object of class *member* consists of the components **joint**, **force** and **sensor**.The object of class *joint* defines the restrictions of the relative motion between the two frames imposed by an interaction element. Component **force** is a set of objects of class *force* and defines the forces and torques exerted by the interaction element. Finally component **sensor** is a set of objects of class *sensor* which serves as a superclass. The derivations of this class will be discussed later.

Due to inheritance, this class description of a multibody system represents a decomposition into basic elements. Therefore the class *mbs* consists of the components of class *block* and additionally of the components **global**, **part** and **interact**. Component **global** contains all the data needed for the overall multibody system like the definition of gravity. Component **part** is a set of objects of class *part*. Similary, component **interact** is a set of objects of class interact. The class hierarchy of this decomposition is show in figure 2.

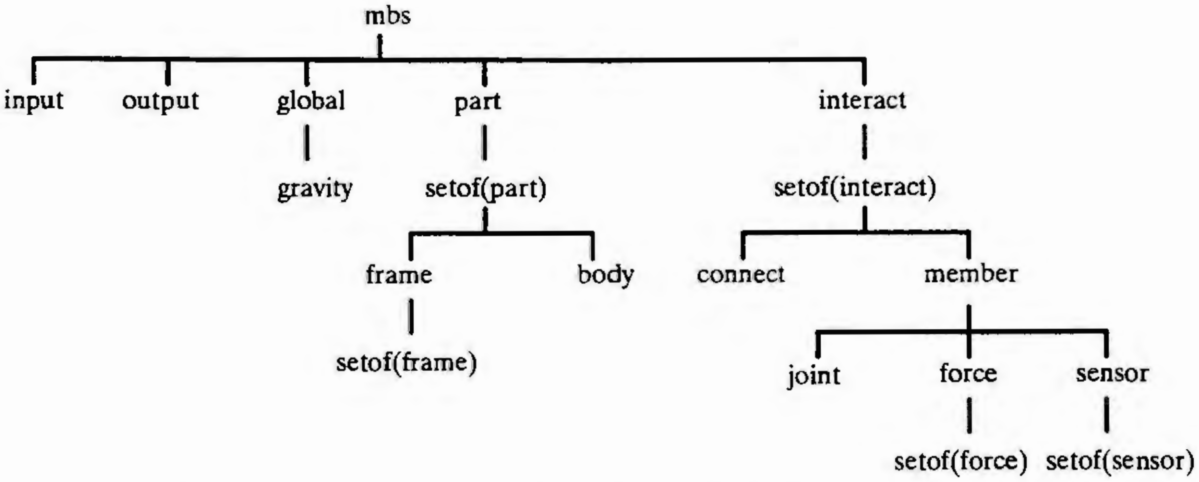


Figure 2: Object hierarchy of the multibody system data model

The data model for multibody systems has been extended in the direction to mechatronics. Classes are added to describe elements of mechatronic systems like sensors and actuators within the multibody system. Sensor elements are used to determine quantities that occur between two frames, e.g. kinematic quantities, applied forces, and reaction forces. These quantities can be used as input signals for other dynamical systems. Class descriptions for rheonomic joints are available for the modelling of position actuators. The connection of these elements which are derived from the description of class *joint* with other dynamical systems like controllers is realized via strictly defined interfaces.

Description of Sensor Elements

An object of class *sensor* defines quantities that are not explicitly defined in the datamodel to be computed and resolved in a desired frame.

Three classes of sensor elements *srel*, *sab* and *slin* are derived from the basic class *sensor* to specify a frame to which the results have to be transformed. An object of class *srel* consists of the components **inpart** and **inframe** to specify an arbitrary frame. Class *sab* is developed to refer to the inertial frame or one of the two frames specified in the object *connect* for the output of the desired quantity. A class *slin* is defined to compute the amount of a certain quantity. To specify the observed quantity between two frames, several classes are derived by inheritance from these three basic classes. Figure 3 shows the hierarchy of the objects of class *sensor*.

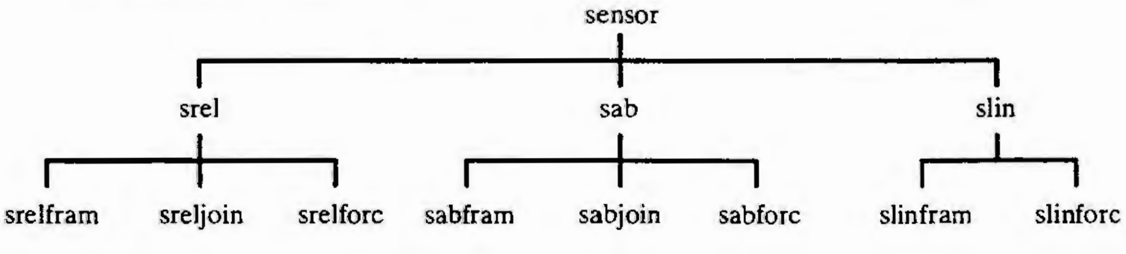


Figure 3: Object hierarchy of class *sensor*

The classes *srelfram*, *sabfram* and *slinfram* are defined to observe kinematic quantitites between two frames. Objects of class *sreljoin* and *sabjoin* are used to analyze reaction forces or joint coordinates. In a similar way, the classes *srelforc*, *sabforc* and *slinforc* are defined to obtain actual exerting forces of coupling elements. For a detailled description refer to Seybold and Neerpasch [4].

Description of Actuators

Classes to describe position actuators are derived by inheritance from objects of class *joint* for modelling the behaviour of multibody systems containing rheonomic constraints. Based on the description of class *joint*, new components **pos**, **vel** and **acc** are added to enable the definition of relative position, velocity, and acceleration between the two connected frames. If more than one component is used, the integrity will be checked. Furthermore, components are added to define the initial conditions of the actuator with respect to position and velocity. Using this scheme of inheritance the classes *revrh* and *transrh* describing a revolute and translational rheonomic joint, respectively, are defined. The class *jgenrh* (joint general rheonomic) allows the description of complex actuators with any combination of free, blocked or driven directions of movement.
A class to describe force actuators is to be defined. All actuators are driven by external signals e.g. they may be functions of time. The signals are transmitted via the object of class *input* which realizes the input interface for signals of other dynamical systems and the connection to the elements defined within the class *mbs*.

4 Implementation and Data Exchange

The object oriented data model for multibody systems and its extensions has been implemented using RSYST, a software environment for scientific and engineering applications. Another implementation, called DAMOS–C has been realized by Daberkow [5]. DAMOS–C represents a spezialized modelling kernel for multibody systems consisting of a data base and methods acting on this data base. Using this methods to access the data means a complete data encapsulation, since the structure of the data on the data base is hidden by the method. The user only has to know the interface of the method but not the structure of the data base.
DAMOS–C is suitable to be used as a neutral data exchange interface between several programm packages for the analysis of mechanical systems.

5 Summary

New classes with respect to mechatronic systems have been derived by inheritance from an existing object–oriented data model for multibody systems.
Class *sensor* and all of its subclasses enables to measure internal, time dependent quantities of the multibody system. The development of class descriptions for rheonomic constraints enables the integration of position actuators in multibody systems.

6 References

[1] Otter, M.: Hocke, M.; Daberkow, A.; Leister, G.: Ein objektorientiertes Datenmodell zur Beschreibung von Mehrkörpersystemen unter Verwendung von RSYST. Stuttgart: Universität, Institut B für Mechanik, Institutsbericht IB–16

[2] Ullman, J. D.: Principles of Database and Knowledge–Base Systems, Volume 1. Computer Science Press, 1988.

[3] Schiehlen, W: Technische Dynamik. Stuttgart: Teubner, 1986.

[4] Seybold, J.; Neerpasch, U.: Erweiterung des objektorientierten Datenmodells zur Beschreibung von Mehrkörpersystemen. Stuttgart: Universität, Institut B für Mechanik, Institutsbericht IB–24, Mai 1993.

[5] Daberkow, A.: DAMOS–C, Beschreibung der Programmschnittstelle der Klassen– und Methodenbibliothek für die Modellierung von Mehrkörpersystemen. Stuttgart: Universität, Institut B für Mechanik, Institutsbericht IB–23, 1992.

INTERDISCIPLINARY MODELLING LANGUAGE FOR MULTI BODY SYSTEMS

O.I. Sivertsen, G. Moholdt, T. Rølvåg[1] and H. P. Hildre
The University of Trondheim
The Norwegian Institute of Technology
N - 7034 Trondheim

Abstract. This paper describes a modelling concept for multi body systems developed in the ESPRIT II #5524 project called "High performance computing for multidiscipline dynamic simulation of mechanisms" (MDS)[2] [4]. The concept is based on a simulation tool combining multi body simulation, the finite element method and control engineering (FEDEM)[3]. Also a STEP toolkit is developed to standardize data communication between different simulation programs.

1. INTRODUCTION

As mentioned above the MDS project was based on the general purpose program system FEDEM for multidiscipline dynamic simulation of mechanism motion combining a non-linear finite element formulation with mechanism and control analysis.

Modelling the system model for a mechanism is usually not a functionality of a finite element method (FEM) preprocessor, and a new module is developed within the project for this purpose. This mechanism modeler may import the FEM models for the different bodies of the mechanism from the FEM preprocessor, in this case FEMGEN[4]. It is also of interest to export coordinates of points on each link relative the link coordinate system, for instance for joints, springs, dampers, loads etc., to the FEM preprocessor as external points for the FEM mesh. The mechanism modeler will display simplified drawings for the different bodies and present joints, springs, dampers, loads etc., as graphic symbols on the display. Control models integrated in a mechanism are generated through numerical input, however, these data will also be entered graphically in the next version of the mechanism modeler.

The original input format for the interdisciplinary simulation program was quite complex and with redundancies that made modelling a quite tedious and cumbersome job. Early in the MDS project a decision was made to develop a new and consistent modelling language for multi body systems including interdisciplinary model entities.

2. MODELLING LANGUAGE

2.1. General description

The MDS interdisciplinary modelling language [1,2] is meant as a unified approach for modelling at system level of multi body systems. The FEM modelling part is referred to by references to separate models in order to reduce the complexity of the modelling language.

The language is compost of 16 entities including the header entity (MECHANISM) for modelling of global parameters and the termination entity (ENDDATA), see Table 1. Each entity occurrence is starting with the entity keyword followed by a number of attributes and ended by a termination character. Many of the

[1] SINTEF Production Engineering, N - 7034 Trondheim, NORWAY

[2] Running from Nov. 1990 to April 1993 with partners from Germany, Italy, England, Sweden, Norway and coordinated by the Danish Technological Institute in Denmark.

[3] FEDEM - Finite Element Dynamics of Elastic Mechanisms, A multidiscipline simulation tool for mechanisms developed by The Norwegian Intitute of Technology and SINTEF Production Engineering, NORWAY

[4] A commercial finite element mesh generating program developed by FEMSYS Limited, United Kingdom

attributes are optional and with a default value. The language is requiring a free format interpretation both regarding the sequence of the entities and the sequence of the attributes within each entity. Of course also consistency of joints, geometry and topology etc. are checked for by the interpreting program.

For a completed model the occurrences of entities should be exactly one for some entities as for the MECHANISM, ANALYSIS and ENDDATA entities, see Table 1. For the LINK and TRIAD entities one or more entities are required. No or one entity occurrence is required for the SENSITIVITY entity. However, the interpreter will for most of the entity types expect to find no or a number of occurrences for the entities. This is the case for the JOINT, SPRING, DAMPER, MOTION, HIGER_PAIR, LOAD, FRICTION, FUNCTION, CONTROL_MOD and CONTROL_IO entities. All entity occurrences have an identification number and a short descriptive text of 30 characters as attributes. The identification number is required while the descriptive text is optional.

Table 1 *Mechanism entities*

NAME	# ENTITIES	DESCRIPTION
MECHANISM	1	Model name, gravity constants,etc.
ANALYSIS	1	Program Control parameters, etc.
LINK	1 : #	Link position/orientation etc.
TRIAD	1 : #	Positions for coordinate systems used in joints, springs, dampers, loads etc.
JOINT	0 : #	Joint definitions
SPRING	0 : #	Axial and joint springs.
DAMPER	0 : #	Axial and joint dampers
MOTION	0 : #	Motion input
HIGHER_PAIR	0 : #	Modelling of transmissions
LOAD	0 : #	Modelling of external loading
FRICTION	0 : #	Modelling of joint or gear friction
FUNCTION	0 : #	Functions defining input motions, forces, spring stiffness's etc.
CONTROL_MOD	0 : #	Input data for control module.
CONTROL_IO	0 : #	Definition of control inputs/outputs
SENSITIVITY	0 : 1	Definition of sensitivity output for optimization
ENDDATA	1	Indicates end of model.

2.2. Description for the different entities

The MECHANISM entity is the header entity for an interdisciplinary multi body simulation model. The descriptive text attribute for this entity may be of three lines in opposite to the 30 character text for the rest of the entities. The gravitational vector and the control data for the FEM processing module are optional attributes for the MECHANISM entity.

The ANALYSIS entity includes all attributes for controlling the analysis. Start time, end time and time stepping for the time integration are required attributes while a large number of integration parameters, tolerances and options are optional and have a default value.

A LINK entity occurrence is used for positioning each body of a multi body system by specifying the position of the link coordinate system. Mass and damping properties of the actual body may also be specified. The flexibility model for the body is specified through an external FEM model reference.

A TRIAD entity occurrence will have an OWNER_LINK reference and options for specifying the TRIADs position and orientation. Initial velocities and accelerations in local TRIAD or global directions may be specified and lumped masses may be added to the TRIAD's degrees of freedom (DOFs). Additional boundary conditions for one or more of the TRIAD's DOFs may be included for static or eigenvalue analysis.

A JOINT entity occurrence will have a reference to a so called slave TRIAD and may also have references to one or more master TRIADs depending on joint type. More master TRIADs are used when distributed prismatic or cylindric joints are specified. When a so called free joint is specified a verity of joint types may be modelled by defining the joint constraints as linear or non-linear springs. Different constraining effects may be modelled by varying the spring lengths as a function of time or implicitly as a function of another variable in the model.

A HIGHER_PAIR entity is used to model gear, rack and pinion and screw transmission types. This entity will refer to one or more JOINT entities. The transmission ratio is an attribute of this entity

SPRING entities may be of type AXIAL or JOINT, that is linear or non-linear springs may be connected between points on different links or in a joint. As mentioned above, joint springs may be used to model elastic constraints or joint motion.

Similar to springs, *DAMPER entities* may be of type AXIAL or JOINT, that is dashpots between points on different links or within a joint. The damping effects may be modelled linear or non-linear.

The MOTION entity is an alternative to using spring length to model motion for the degrees of freedom for a TRIAD or within a joint. The motion may be time dependent or an implicit function of a model variable.

LOAD entities are used to model external loading with reference to a TRIAD. The loading may be specified directly along the TRIAD degrees of freedom as forces or torques, or as forces or torques with general direction in three dimensional space. The general load direction is specified by two points referring to the global coordinate system or to coordinate systems on one or more links. The magnitude of the load may be time dependent or an implicit function of a model variable.

A FRICTION entity is referring to a JOINT or HIGHER_PAIR entity, that is joint or gear friction. For calculation of the different friction effects a so called friction function is referenced.

The FUNCTION entity is a very general facility for the user to select between a number of predefined algorithms or to algorithms generated by himself for time dependent or implicit function evaluations for the simulation. Function evaluations may be specified and referenced from the MECHANISM, SPRING, DAMPER, MOTION, LOAD, CONTROL_IO, FRICTION and SENSITIVITY entities. Specific function algorithms are referred from the FRICTION and SENSITIVITY entities. Also spring stiffness reference from the SPRING entity are using a specific function algorithm. A list of parameters are entered through this entity where both the number of parameters and the meaning of the parameters depend on function type. User written algorithms are easily referred to within the same framework.

Each *CONTROL_MOD entity* occurrence is referring to a control element type and a corresponding control element parameter list. A specific control element will have a predefined number of input, output and internal terminals with a unique local numbering sequence. A control system will consist of one or more control elements coupled through a number of control variables with global numbering. Each entity of this type will also have a topology sequence to connect the control element terminals to the control variables. A number of predefined control element algorithms are available from a library and the user may program his own control algorithms to be referred in the same way.

The CONTROL_IO entities are used to couple the mechanical and the control system in the simulation. This entity distinguishes between input to the control system from the mechanical system and visa versa. Control input entities may be regarded as sensors on the mechanical system for positions, velocities, accelerations, distances etc. Control output entities may be regarded as actuators on the mechanical system introducing forces and torques. Each entity will have a control in/out flag, a type code and up to three indexes to specify the actual coupling element.

A SENSITIVITY entity is used to specify sensitivity calculations for simulation response variables with respect to model design parameters, to be used for manual or automatic design optimization. The entity will contain a list with initial values for design parameters and a list of references to the sensitivity function that specifies which sensitivities to be calculated.

The ENDDATA entity is used as a termination indicator for the interdisciplinary multi body model.

3. DATA COMMUNICATION

The entities of a simulation model are entered into a symbolic text file containing the language keywords for entities and attributes with numerical or text values connected to the attributes. The file is input for the simulation program data interpreter. Within the MDS project a STEP EXPRESS SCHEMA[5] was implemented for the interdisciplinary modelling language presented here. Through the STEP interface the data model may be communicated in a standard way between different simulation programs and graphic modelers. The STEP EXPRESS SCHEMA and a general STEP toolkit [3] developed in the MDS project are used to control a STEP database and to interpret and generate STEP data files.

4. CONCLUSIONS

The modelling language presented here have been used and tested for more than a year and the feedback from the users have been very positive. The language itself is simple and easy to understand especially when none of the advanced features are used, that is it is easy to get started. Compared to the old FEM inspired input data format for the simulation program in question all redundancies in the data model are avoided. Using graphic modelers, the user will primarily be working with graphic symbols. However for advanced features and user defined algorithms the file should be easily readable for manual editing, and from the users feedback we think that this is obtained by this language definition.

5. REFERENCES

[1] Iversen T., Sivertsen O.I.,"Multidiscipline software specification document 2", MDS deliverable D2104 (restricted), 1993

[2] Rølvåg T., Hildre H.P. and Sivertsen O.I.,"Updated Multidiscipline Simulation;User's Guide", MDS deliverable D2106, 1993.

[3] Korwaser H.,"STEP toolkit documentation", MDS deliverable D2301 (restricted), 1993

[4] Sivertsen O.I., Rølvåg T. and Hildre H.P.,"The Multidiscipline Design Concept for Mechanisms", NATO ASI conference: Computer Aided Analysis of Rigid and Flexible Mechanical Systems. Troia-Portugal, 27 June - 9 July, 1993.

[5] STEP is an international standard for data modelling and EXPRESS is the data modelling language used by STEP.

submitted to IMACS/MATHMOD Conference, Vienna, February 2–4, 1994

MODELLING AND IDENTIFICATION OF FLEXIBLE ROTORS IN MAGNETIC BEARINGS

R. Herzog and C. Gähler

Mechatronics Lab, CH 8092 ETH Zürich
Switzerland

Abstract. The modelling of flexible rotors in active magnetic bearings (AMB) is a manifold and demanding task which is often combined with *frequency domain identification* methods. We consider identification methods which need *no* a-priori knowledge, and methods which make use of existing finite element modelling (FEM) of the flexible rotor. A Matlab-based signal processor environment which includes tools for frequency response measurements and AMB identification was developed.

1. INTRODUCTION

Active magnetic bearings (AMB) are used to support a body by magnetic forces without any mechanical contact [4]. The main advantages of magnetic bearings are: absence of mechanical wear and friction, lubricant-free operation, and the possibility of very high rotational speeds. The basic principle of an AMB is the following: Electromagnets mounted around the rotor generate *attracting* bearing forces. These forces by themselves will *not* lead to a *stable* equilibrium position of the rotor. Therefore, stability must be achieved through a *feedback control loop*: contactless gap sensors are fed into a digital signal processor (DSP) which controls the current of power amplifiers driving the electromagnets.

Magnetic Bearings including its levitated mechanical body, the electronic circuitry, and its control software, naturally fall into the category of *mechatronic* systems. There is a strong need for precise modelling of such systems for two reasons. First, the modelling of AMB systems enables valuable prediction of their dynamic properties under various operating conditions. Second, precise model data is required for the controller design. Modelling of AMB systems is a manifold task consisting of the following items:

- Mechanical modelling of the rotor. With flexible rotors, many weakly damped natural frequencies may occur. Usually, this modelling is carried out using standard finite element (FEM) software packages.

- Mechanical modelling of housing and machinery environment. Even though this modelling part is frequently neglected, practice has shown that it often considerably influences systems dynamics. This raises the general question where the boundary of system modelling should be drawn.

- The modelling of magnetic fields. This can be achieved with FEM methods or with approximative assumptions on the magnetic field distribution.

- The modelling of sensor and actuator dynamics.
- Modelling the effects of computational delays and finite word-length precision of the signal processor.

It is necessary to combine the modelling of AMB systems with *identification methods* based on a finite number of input/output measurements. For measurement and identification purposes we developped a link between the signal processor and a Matlab environment on a PC [3]. *Frequency domain* input/output measurements are perfectly suitable for filtering out noise and non-linear phenomena.

2. AMB IDENTIFICATION USING THE ROTOR FEM MODEL

It is useful to determine the AMB characteristics as a separate block, see figure 1. When applied in a double acting configuration [4], the force f of an AMB is linear to both current i and displacement x, i.e. $f = k_x x + k_i i$, where the displacement coefficient k_x denotes the *negative* bearing stiffness and k_i is the current coefficient. Both k_x and k_i depend on the bearing geometry with its magnetic field distribution and the premagnetization bias current. Since FEM modelling of the flexible rotor is quite accurate in most cases, the AMB parameters k_x, k_i can be identified using the plant frequency response $P(i\omega)$ and the known rotor dynamics $P_R(i\omega)$.

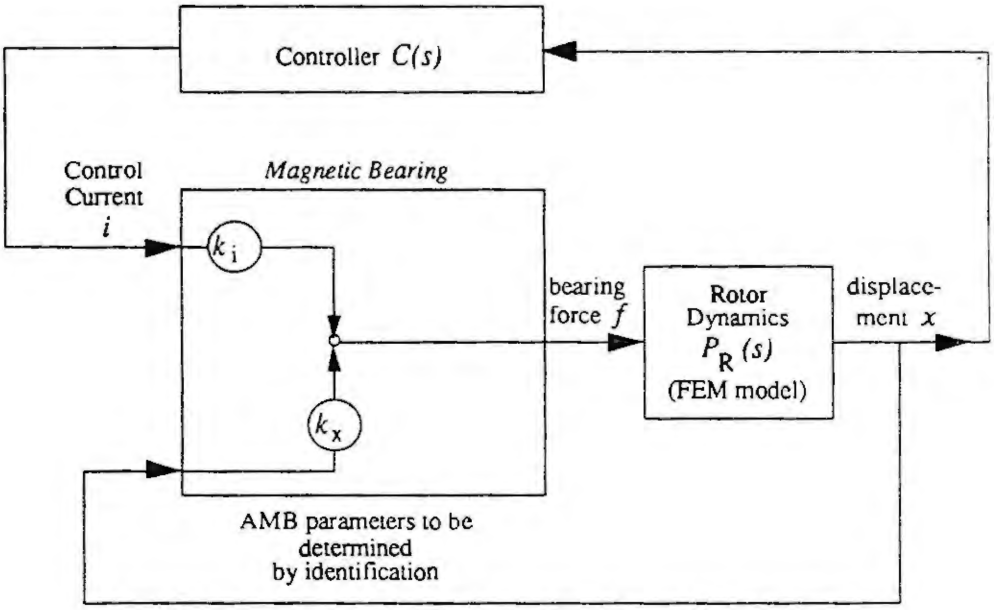


Figure 1: Identification of a voltage controlled magnetic bearing plant.

The plant frequency response equals $P(s) = k_i P_R(s) / (1 - k_x P_R(s))$. Equivalently, $P_R(i\omega_k) k_i + P(i\omega_k)P_R(i\omega_k) k_x = P(i\omega_k)$, $k = 1 \ldots N$. Since this latter equation is linear in the unknowns k_x and k_i, standard least square solvers can be applied.

The measured transfer function $P(s)$ in figure 2 illustrates the resulting fit in the case of a flexible rotor with one significant mode. The truncated FEM model $P_R(s)$ which was used in figure 2 has the form $P_R(s) = (s^2 + \nu^2) / (s^2(s^2 + \omega^2))$.

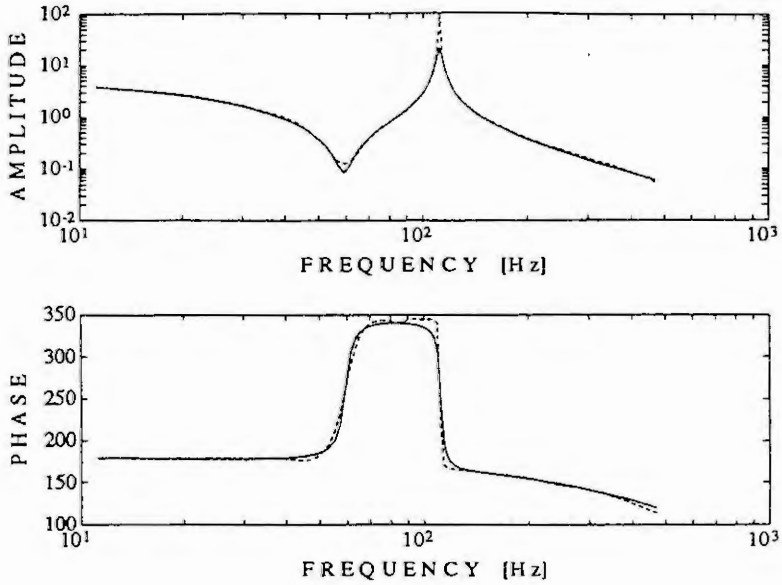


Figure 2: Solid: identification result, Dashed: measured frequency response $P(s)$.

3. AMB IDENTIFICATION USING LITTLE A-PRIORI KNOWLEDGE

It is generally desirable that an identification algorithm be "robust" under small measurement noise or small perturbations of the data. For example, *Lagrange interpolation* which consists of fitting a polynomial of minimal degree through points (x_k, y_k) with equidistant abscissa x_k is known *not* to have this property. Figure 3 shows this bad behaviour of Lagrange interpolation. The noisy measurement data (marked with o) causes large oscillations of the identification output (solid curve).

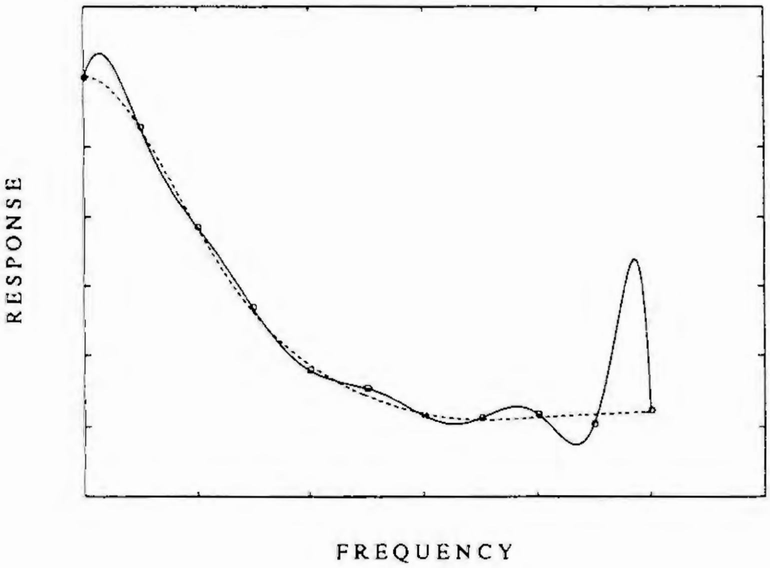


Figure 3: Robust identification seeks to prevent large sensitivities w.r.t. noisy data.

The mathematical framework of "robust identification" and various identification algorithms were proposed recently in literature [1], [2], [5]. Most of these algorithms consist of two steps. In the first step, the frequency points are transformed to an impulse response function $\sum_{-N}^{+N} h_k z^k$ using inverse FFT and windowing functions. In the second step, the anti-causal part of the impulse response is approximated by a stable function using Nehari extension. Both steps can be easily implemented in Matlab since they rely only on standard matrix computations. The following items report some of our experience with this identification approach.

- Usually, the sampling rate of AMB systems is quite fast, compared to the open-loop system dynamics. For identification in the z-plane a very high number N of terms in the impulse response function $\sum_{-N}^{+N} h_k z^k$ is needed for accurate results. We lessened this problem by applying a bilinear transform which maps the unit disc $\{|z| < 1\}$ onto a unit disc $\{|w| < 1\}$ in a new w-plane. An equally spaced w frequency grid $w_k = e^{i\varphi_k}$ corresponds to an irregularly spaced z frequency grid, which in turn is closely spaced at low frequencies and more loosely at high frequencies. After carrying out the identification in the w plane, the result is transformed back to the z-plane.

- The methods proposed in [1], [2] are only applicable to stable systems. Since magnetic bearings are unstable plants one could identify closed-loop functions, e.g. the sensitivity function $S(s) = (1 - P(s) \cdot C(s))^{-1}$, and determine plant $P(s)$ using the known controller $C(s)$. This approach ends up with a pole-zero cancellation which is a numerically delicate operation. Therefore, we preferred to identify the unstable plant and to replace the Nehari step in [1], [2] by a model reduction of the two-sided impulse response $\sum_{-N}^{+N} \tilde{h}_k w^k$.

4. RESULTS

The methods described above allowed us a successful first step in identification of AMB systems. Further investigations are required in order to compare different identification methods using different a-priori knowledge.

REFERENCES

[1] Gu G. and P. Khargonekar: A Class of Algorithms for Identification in $\mathcal{H}_\infty$, Automatica, Vol. 28, No. 2, pp. 299-312,1992.

[2] Helmicki A.J., Jacobson C.A. and C.N. Nett, Control Oriented System Identification, A Worst-Case Deterministic Approach in $\mathcal{H}_\infty$ IEEE Trans. AC, V. 36, No. 10, 1991.

[3] Herzog R. and R. Siegwart, High Performance Data Acquisition, Identification, and Monit. for Act. Magn. Bearings, 2nd Int. Symp. Magn. Susp., Nasa, Seattle, 1993.

[4] Schweitzer G., A. Traxler und H. Bleuler, Magnetlager, Springer Verlag, 1993.

[5] Tse D.N.E., Dahleh M.A. and J.N. Tsitsikilis, Optimal Asymptotic Identification under Bounded Disturbances, IEEE Trans. AC, Vol. 38, No. 8, Aug. 1993.

Travelling Wave Ultrasonic Motors -
Novel Mechatronic Drive Systems

Jörg Wallaschek
Heinz Nixdorf Institut
Universität-GH Paderborn
33095 Paderborn, Germany

Abstract. Travelling wave ultrasonic motors are novel mechatronic drive systems, which are characterized by high torque at low rotational speed, simple mechanical design and good controllability. They also provide a high holding torque even if no power is applied. Compared to electromagnetic motors the torque per volume ratio can be higher by an order of magnitude. In this paper the working principle of these motors is explained and the structure of a hierarchical block-oriented model is proposed.

1. INTRODUCTION

The travelling wave motor is an embodiment of a piezoelectric ultrasonic vibration motor. In these motors mechanical oscillations of high frequency and small amplitude are excited by piezoelectric elements in such a way that material points on the surface of the stator perform an elliptic motion. The rotor is pressed against the stator and is driven by frictional forces generated in the contact area. Usually the elliptic motion of the stator's surface is obtained by the proper superposition of two orthogonal vibration modes of the stator having the same resonance frequency and the motor is operated in resonance. Although this driving principle is well known for quite a while [1-6], only few types of piezoelectric ultrasonic motors have been developed until recently. This is mainly due to the fact that piezoelectric materials with high conversion efficiency and fast electronic power control of the mechanical oscillations have not been available until a few years ago. After the appearance of first prototypes of travelling wave ultrasonic motors, much research has been devoted to develop motors with better performance and many papers have been published in the last years. However, most of these contributions have been concerned with specific problems arising in the design and control of the travelling wave motor. The present paper is an attempt to combine these results in order to obtain a general mathematical model of the motor and to identify missing links in the model which still require further research. Due to space limitations only few equations will be presented. Whereever possible the reader is referred to the references at the end of the paper.

2. WORKING PRINCIPLE OF TRAVELLING WAVE MOTORS

The travelling wave motor is described in detail in [7 - 9]. As can be seen from Fig. 1, its stator is a cylindrical plate. In this plate a travelling bending wave is excited by a piezoceramic layer which is polarized according to the wavelength of the travelling wave. The rotor is pressed against the stator by a disc-spring.

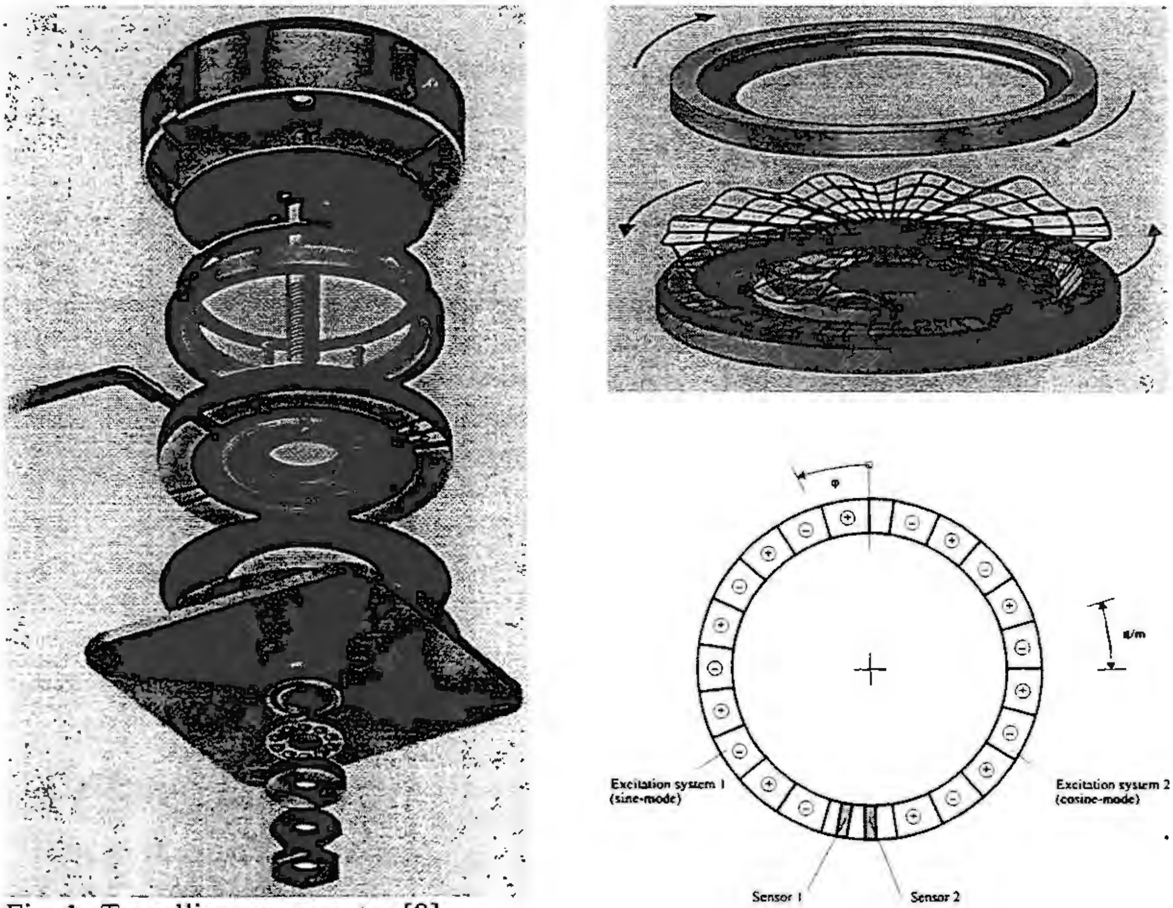

φ
λ/m
Excitation system 1 (sine-mode)
Excitation system 2 (cosine-mode)
Sensor 1
Sensor 2

Fig. 1 Travelling wave motor [8].

It is well known that due to the symmetry there exist double eigenvalues in the free vibration problem of circular plates: associated to each eigenfrequency ω there are two linearly independent eigenfunctions, which are called sine-mode and cosine-mode in the following. The piezoceramic layer is polarized in such a way that one group of segments (excitation system 1) excites the cosine-mode and a second group of segments excites the sine-mode (excitation system 2) of a vibration mode whose number of nodal diameters m corresponds to the polarization. The piezoceramic is polarized in its thickness direction and the bending vibrations of the stator are excited by the in-plane expansion and contraction of the ceramic. If the frequency of the excitation is tuned to the eigenfrequency of the free vibration, and if the relative phase shift of the excitation is chosen properly, a resonant travelling wave

$$w(r,\varphi,t) = R(r)\cos(m\varphi - \Omega t), \qquad (1)$$

is generated in the stator, Ω being the circular frequency of the excitation. In order to achieve the correct mode-superposition, the vibrations of the stator are controlled using sensors included in the piezoceramic layer, and an electrical control circuit.

If the stator vibrates according to (1), material points located on the surface of the stator perform an elliptical motion and have a tangential velocity. The contact mechanism which is responsible for the tangential stresses driving the rotor is very complicated. The contact area between stator and rotor moves with the travelling wave, stator and rotor are always in contact near the wave crests. In most travelling wave motors the rotor is coated with special lining material in order to obtain a good force transmission and good wear resistance. The lining material´s properties do strongly influence the motor characteristics. It has been shown that in the contact area between stator and rotor, regions of slip and regions of stick do occur due to the different tangential velocities of the contact points of stator and rotor [10,11]. If in a first approximation, zero slip and point contact are assumed for the no-load case, the rotor´s angular velocity is

$$\dot{\Phi}_{Rotor} = -\Omega \frac{a}{r_B} m \frac{R(r_B)}{r_B}, \tag{2}$$

with a being the distance between the stator´s surface and the middle plane, r_B the radius of contact and $R(r_B)$ the vibration amplitude of the middle plane at r_B. As can be seen from equation (2) there is a large frequency reduction between the vibration frequency of the stator Ω and the rotation frequency of the rotor $\dot{\Phi}_{Rotor}$. This is due to the fact that the stator thickness is small and the vibration amplitude is extremely small compared to the radius of contact. If actual design values are inserted, a frequency reduction in the order of 1 : 40.000 is obtained.

Most travelling wave motors have stators of variable thickness. A stiff inner ring clamped to the motor casing is followed by a thin "insulating" ring, and a thick outer annular ring containing radial notches. The piezoceramic layer is bonded to the lower surface of the outer ring, and the rotor is in contact with the "teeth" of the upper surface of the outer ring. This particular shape of the stator has the advantage that the distance of the contact surface from the middle surface can be made large without significant increase of the stiffness of the stator.

As mentionned before, the travelling wave motor is operated in the vicinity of an eigenfrequency of the stator and an electrical control is needed in order to achieve the correct mode superposition. The system´s behavior is dominated by the mechanical resonance of the stator. Fig. 2 shows the transfer function between the electrical excitation and the mechanical vibration. The vibrations of the stator can be controlled by

- the amplitude of the electrical excitation,
- the frequency of the electrical excitation (using resonance amplification),
- the phase shift between the two excitation systems (influencing the form of the travelling wave).

Depending on the type of motor application, different controls can be designed using the principles stated above or combinations thereof.

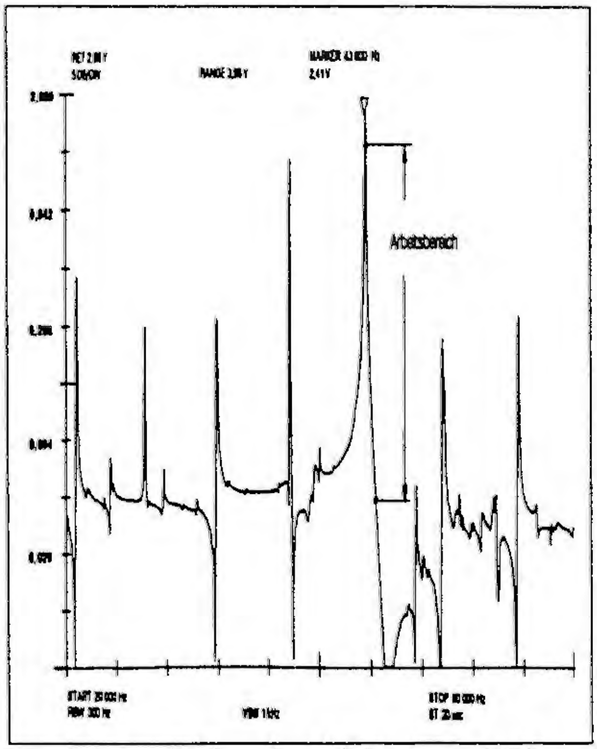


Fig. 2 Transfer function of the stator [12].

In the motor control some information about the vibration of the stator is required in order to stabilize the desired vibration in the vicinity of the resonance frequency which depends on the motor load, the temperature and other parameters. This information can be obtained most effectively using piezoelectric sensors, which can be integrated in the same piece of ceramics which is used for the excitation. However, also sensorless controls have been realized succesfully [13].

In order to excite a travelling wave in the stator two high voltage signals of proper frequency and well-defined phase shift must be generated and applied to the piezoceramic layer. To this end a two-phase high frequency resonant inverter can be used. The converter inductivity can be tuned to the capacitance of the piezoceramics to form an electric resonance circuit. This interdependence of mechanical and electrical quantities and the complex information processing of the motor control make the travelling wave motor indeed a true mechatronic system. Besides the many interesting technical properties of this motor it is also an excellent educational example in mechatronics.

3. STRUCTURE OF A BLOCK-ORIENTED MODEL OF THE MOTOR

Before a mathematical model of the travelling wave motor can be formulated, the system has to be divided into functional modules. Fig. 3 shows one possible choice of functional modules. It is based on the fact that travelling wave ultrasonic motors are usually operated in a small vicinity of the resonance frequency and that - in a first order approximation - all state variables perform harmonic oscillations. Then the piezoceramic actuaor can be modelled as a linear subsystem which transforms the voltage amplitudes A_1, A_2 of the electrical excitation to an induced strain excitation of the stator, represented by F_1, F_2, where the index 1 refers to an excitation of the sine-mode and index 2 refers to the cosine-mode. Although models for piezoceramic strain actuation of simple geometries are readily available in the literature [14] it is still a difficult task to take the complicated geometry of an actual travelling wave motor into account.

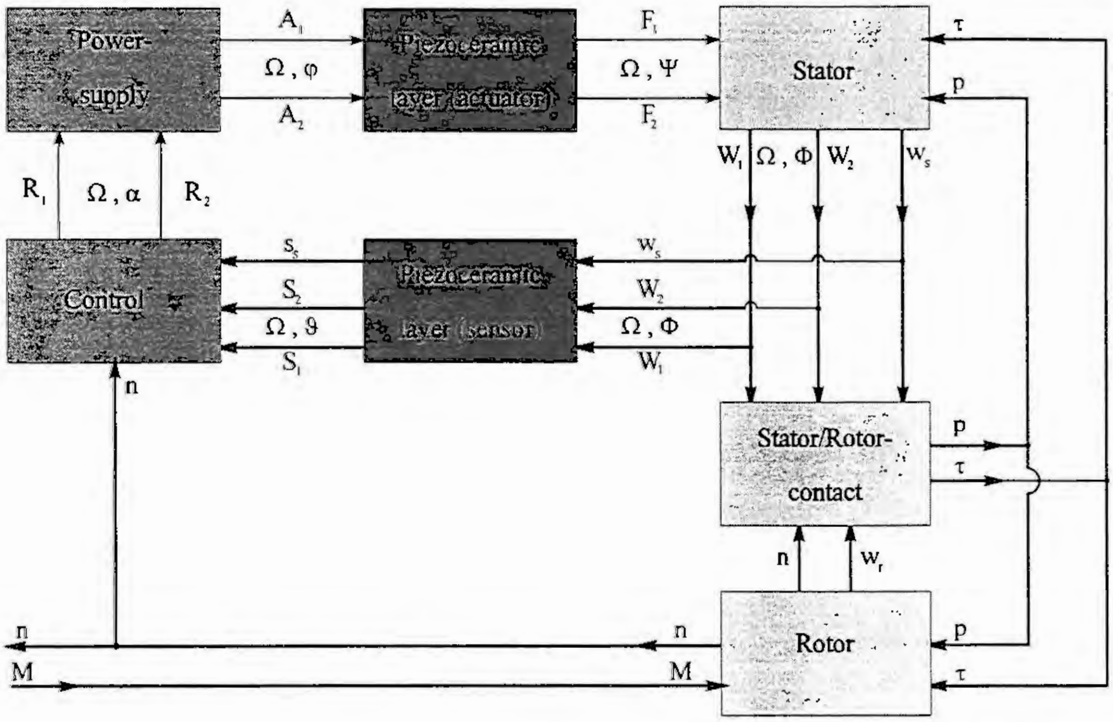


Fig. 3 Functional modules of the mathematical model of a travelling wave motor.

As long as the stator of the travelling wave motor is not in contact with the rotor, it merely acts as a linear subsystem, which transforms the induced strain excitation F_1, F_2 to the vibration amplitudes W_1, W_2 of the sine- and cosine-mode respectively. It can then be described by its (complex) transfer matrix which is in diagonal form if the stator is perfectly symmetric. However, in most cases due to manufacturing imperfections there is a small cross-coupling between the two modes. Interestingly enough, the notches of the stator do not always lead to a symmetry disturbance. If the number of notches and the number of nodal diameters are chosen properly, perfect symmetry of sine- and cosine-mode can be preserved [15].

The motion of the stator is observed via the piezoceramic sensor which basically measures the vibration amplitudes W_1, W_2 of the stator and transforms them to the sensor voltage amplitudes S_1 and S_2. Considering the piezoelectric layer (actuator and sensor) and the stator as a unit, it is possible to describe the electrical terminal behavior of the mechanical system as an electrical impedance which is determined by the mechanical design of the stator and the dielectric properties of the ceramic. Although often the simple electrical equivalent circuit of Fig. 4 can be employed to describe the behavior of the travelling wave motor, there are several limitations in the model which are as follows

- the cross-coupling of modes is not modelled
- the influence of the rotor/stator contact is not modelled
- the parameters can only be determined experimentally, up to now there are no analysis tools available that allow an a-priori estimation of the system parameters.

The cross coupling of the modes can easily be included in the model by considering two independent electrical equivalent circuits which are connected via a cross-capacitance. It is much more difficult to take the stator/rotor interaction into account. In a first approximation the stator/rotor contact can be considered as a nonlinear subsystem that transforms the motion of stator and rotor into a contact stress distribution represented by p (normal stress) and τ (tangential stress). Of course these contact stresses act on the stator and on the rotor and have also to be considered as force inputs to these modules. The corresponding outputs are denoted by w_s and w_r respectively. While stator and rotor itself can be modelled as linear subsystems, the stator/rotor contact is highly nonlinear and up to now, to the best of the author's knowledge, there are only few adequate models available, which can be used to describe the stator/rotor interaction.

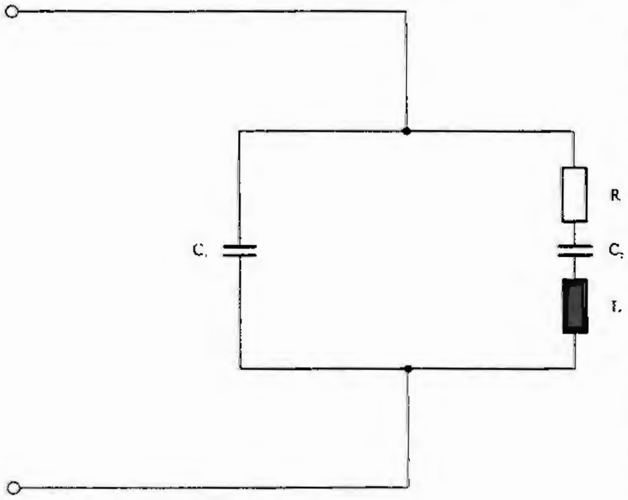


Fig. 4 Electrical equivalent circuit model of a travelling wave motor (one phase only).

4. SUMMARY AND OUTLOOK

Travelling wave ultrasonic motors have now been investigated for quite a while. There are, however, still many fundamental questions which need to be answered, before the full potential of this new actuator generation can be exploited. The most important and most difficult problems are related to the stator/rotor contact. There is no theory explaining all the important interdependences between the material parameters of the contact layer, the stator and rotor vibration, and the forces acting between stator and rotor. Another important subject is the detailed modelling of the piezoceramic actuation. If these problems with be solved, the block-oriented model described in this paper can be used to characterize the behavior of a travelling wave motor and it can be used to optimize the motor parameters in the early design phase as well as to design optimal motor controls.

5. ACKNOWLEDGEMENTS

The author gratefully acknowledges the support provided by the Deutsche Forschungsgemeinschaft under grant no. DFG-Wa 564/6-1 and 564/7-1.

6. REFERENCES

[1] Williams, A.; Brown, W.: Piezoelectric motor. US-Patent 2 439 499, August 1942.

[2] Barth, H. V.: Ultrasonic driven motor. IBM Technical Disclosure Bulletin, Vol. 16, No. 7, p. 2236, December 1973.

[3] Wischnewski, V. C. et. al.:Elektrischer Motor. Deutsches Patent, Offenlegungsschrift DE 2530045 C 2, July 1975.

[4] Lavrinenko, V. ;Kartaschew, I. A.; Wischnewski, V. C.:Piezoelectric motors (in russian), 1980, Energia, Moscow.

[5] Sashida, T.: Trial construction and operation of an ultrasonic vibration driven motor (in japanese), Oyo Buturi 51 (1982) 6, pp. 713 - 720.

[6] Ragulskis, K.; Bausevicius, R.; Barauskas, R.; Kulviets, G.: Vibromotors for precision microrobots. 1988, Hemisphere Publishing Corp.

[7] Akiyama, Y.: State of ultrasonic motors in Japan. Journal of Electrical Engineering 4 (1987), pp. 76 - 80.

[8] Schadebrodt, G.; Salomon, B.: Der Piezo-Wanderwellenmotor - ein neues Antriebselement in der Aktorik. 24. Technisches Presse-Kolloquium der AEG, 1989.

[9] Hagedorn, P.; Wallaschek, J.: Travelling wave ultrasonic motors, part I: Working principle and mathematical modelling of the stator. Journal of Sound and Vibration (1992), 155 (1), pp. 31 - 46.

[10]Hirata, H.; Ueha, S.: Revolution speed characteristics of an ultrasonic motor estimated from the pressure distribution of the rotor. Japanese Journal of Applied Physics, Vol. 31 (1992), Supplement 31-1, pp. 248 - 250.

[11]Maeno, T.; Tsukimoto, T.; Miyake, A.: Finite element analysis of the rotor/stator contact in a ring type ultrasonic motor. IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, Vol. 39, No. 6, November 1992, pp. 668 - 674.

[12]Anon.: Swingender Ring. m+w 7/1990 - Konstruktion und Entwicklung 3, pp. 40 - 47.

[13]Furuya, S.; Maruhashi, T.; Izuno, Y.; Nakaoka, M.: Load-adaptive frequency tracking control implementation of two-phase resonant inverter for ultrasonic motor. IEEE Transactions on Power Electronics, Vol. 7, No. 3, July 1992, pp. 542 - 550.

[14]Crawley, E.F.; Anderson, E.H.: Detailed models of piezoceramic actuation of beams. Journal of Intelligent Material Systems and Structures, Vol. 1, Jan. 1990, pp. 4 - 25.

[15]Pu, C.; Hagedorn, P.; Wallaschek, J.: Der Ultraschall-Wanderwellenmotor, Neue Ergebnisse. In. Isermann, R. (ed.): Integrierte Mechanisch-Elektronische Systeme, VDI-Verlag, Reihe 12, Nr. 179 (Fortschritt-Berichte), pp. 157 - 167.

OBJECT-ORIENTED MODELLING AND SIMULATION OF MECHATRONIC SYSTEMS AND THEIR APPLICATION TO AUTOMOTIVE INDUSTRY

Dr.-Ing. Roland Kasper
Dipl.-Math. Stefan Hagel
Robert Bosch GmbH
Dept. K/SWE,
POB 300240
70442 Stuttgart,
Germany

Abstract. The basic ideas of a modelling and simulation tool are presented which is optimized to support the development of mechatronic components in automotive industry. It is shown, how the advantages of object orientation can be used to handle very large models under realtime conditions. Special problems are discussed that are associated with the modelling of mechanical systems.

1. INTRODUCTION

During the last decades the automobile has evolved from a primarily mechanical product to a mechatronic one. This fact is quantitatively evident by the continuously growing number of electronic compononents. Much more important is a qualitative aspect; none of the latest improvements in automobile functionality could have been reached, without the optimized cooperation of mechanical and electronic components. In the past, most of these components have been developed independently. Now the time has come, to help them grow together and to cooperate across traditional system borders. Mathematical modelling and simulation has proven to be a an effective tool to support this process. For some steps of a modern automotive design cycle, it is even necessary to handle the union of all these functions in a model of the complete automobile. In terms of mathematical modelling, a complete automobile is a very large system consisting of a great number of algebraic, differential and discrete equations. For practical purposes however it is not possible to work on this level, and the resulting problem turns out to be unnecessarily complex. Fortunately there exists a natural way to avoid this level of complexity. Each automotive component has besides its primary function a well defined interface. This is guaranteed by construction! Our approach is to imitate this aspect of real components to create component models that are as easy to handle as their real world counterparts. This method is made available to the design engineer by a new modelling and simulation tool called ASCET (Advanced Simulation and Control Engineering Tool) [5 and 6]. This tool has been developped by Robert Bosch during the last five years and will now be available to customers and other users.

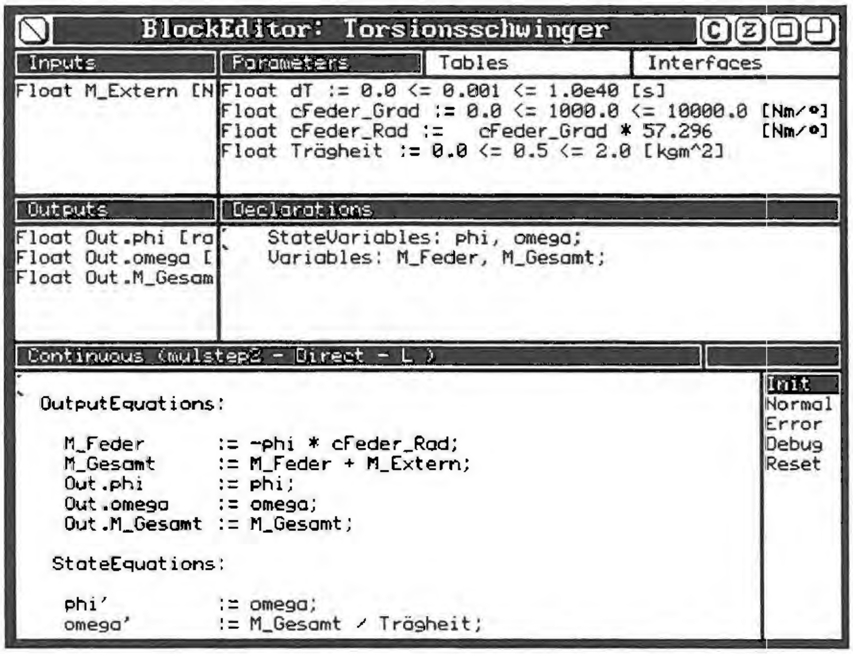
2. MODELLING

From a classical point of view the mathematical model of a component consists of its states, its inputs and outputs and its parameters. It should be evident that a difference must be made between *base-blocks* that are used to define a components behavior by means of a C like simulation language and *structure-blocks* that serve to build more complex blocks out of base-blocks or other structure-blocks.

2.1 Base-blocks

The description of the base-block´s behavior can be divided into several section (fig. 1). In the *parameter-equations* section the parameter dependencies are defined. This is the only place, where parameters appear on the left side of an expression. In the *state-equations* section the block´s states are defined. For discrete blocks by difference equations, for continous blocks by differential equations. In the *output-equations* section the block´s state is used to define the outputs. Base-blocks that use at least one of the inputs to calculate an output are called *direct blocks*. If only states and parameters are used in the output-equation section it is a *dynamic block*. During simulation, dynamic blocks can calculate their outputs and send them to the connected receivers independently from the sender blocks, even before one input is known for the actual time step. The inputs are needed only to calculate the state equations. For direct blocks in general all inputs have to be known, before the output equations can be solved.

Each block has its own integrator. By supporting only multistep integration methods like Euler, Adams-Bashford etc. we avoid problems arising when abstract mathematical data must be transferred between blocks. This garantuees that for each time step all inputs and outputs of blocks carry only physically defined data.



```
Float M_Extern [N
```

```
Float dT := 0.0 <= 0.001 <= 1.0e40 [s]
Float cFeder_Grad := 0.0 <= 1000.0 <= 10000.0 [Nm/°]
Float cFeder_Rad :=   cFeder_Grad * 57.296    [Nm/°]
Float Trägheit := 0.0 <= 0.5 <= 2.0 [kgm^2]
```

```
Float Out.phi [ra
Float Out.omega [
Float Out.M_Gesam
```

```
    StateVariables: phi, omega;
    Variables: M_Feder, M_Gesamt;
```

```
 OutputEquations:

   M_Feder       := -phi * cFeder_Rad;
   M_Gesamt      := M_Feder + M_Extern;
   Out.phi       := phi;
   Out.omega     := omega;
   Out.M_Gesamt  := M_Gesamt;

  StateEquations:

   phi'          := omega;
   omega'        := M_Gesamt / Trägheit;
```

Fig. 1: Block-Editor for a torsional oscillator base-block

From an object oriented point of view, the definition of a *base-block* can be used to define a *class*. The states, parameters and the integrator of a block can be viewed as the *instance variables*. Its inputs and outputs together with the state- and output-equation section are *methods*. This allows to use valuable advantages of object orientation. *Instantiation*: To use a base-block in a block diagram, one simply creates a new instance of the block´s class. Its instance variables (states, parameters, integrator) can be initialized with new instances of appropriate classes. A block´s class can be seen as an abstract template of a block´s behavior. The initialization procedure fixes the types of the states and parameters. So it is possible to create an integer or a float version or even a version supporting the complicated arithmetic used in automotive control units. *Inheritance*: A further advantage is the possibility to use inheritance and to create subclasses. This is a very nice feature for example, if a new base-block class is very similar to an existing class, but uses some extra states or parameters or implements different output-equations or extra inputs and outputs.

2.2 Structure-blocks

A structure-block is used to combine predefined behavior of other blocks. Fig. 2 shows how it is built up from *blocks* and *connections*. A connection defines a signal transmission from a sender block´s output to a receiver blocks´s input. Like base-blocks, structure-blocks supply a *parameter-section* which is used to define parameter dependencies as well as *inputs* and *outputs* to communicate with connected blocks.

From an object oriented point of view, a *structure-block* can also be used to define a *class*. Its *internal blocks* and *parameters* build up its *instance variables*. Its *inputs* and *outputs* define the associated *methods*. Now one can use some advantages of object oriented abstractions. *Instantiation*: To use a structure-block in a block diagram, one must merely create a new instance of the structure-block class. Its instance variables (base-blocks, structure-blocks and parameters) can be initialized with new instances of appropriate classes. A structure-block class can be seen as an abstract template of a block´s behavior. The initialization procedure fixes the instances of blocks and parameters that are used. This way, it is possible to define an abstract template of, let´s say, a vehicle power train that can be used to model arbitrary types of gears, motors, clutches and so on. Only the interfaces have to match. *Inheritance*: A further advantage is the possibility to use inheritance and to create subclasses. This is a very nice feature for example, if a new structure-block class is very similar to an existing class, but uses some extra blocks or parameters or implements extra inputs and outputs. A model constructed this way is scalable in complexity. Component models can vary from very coarse grain ones, which are sufficient for many standard applications, to very fine grain ones, that can be used if it is necessary to simulate very special effects. The flexibility and the clearness of the formulation garantuee that the desired model instance can be constructed with little effort.

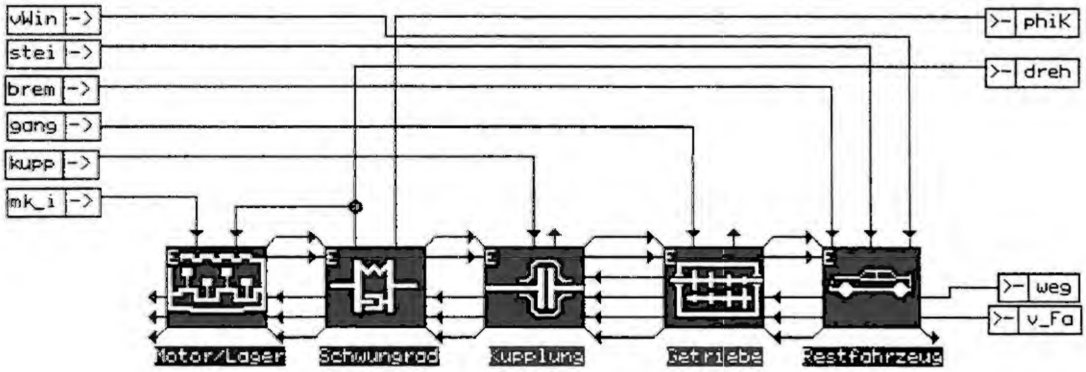
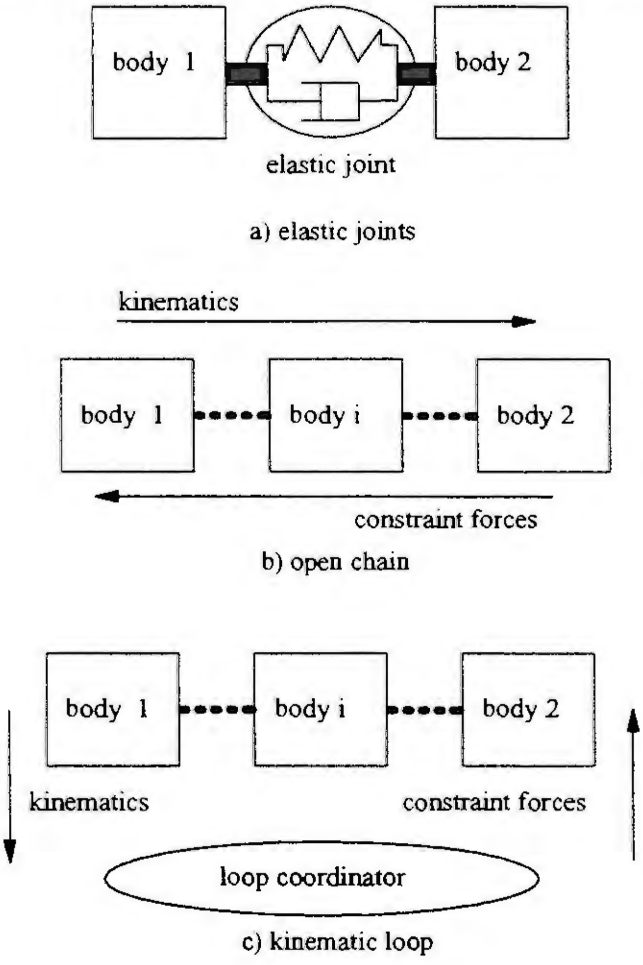


Fig. 2: Block-Editor of a structure-block defining a vehicle power train

3. COMPONENTS AND INTERFACES

The approach presented in this paper is based on the idea of independent components. These components are furnished with well defined interfaces that carry coupling information of mechanical, hydraulic and electrical systems as well as of typical controllers and other digital systems. Information transferred across these interfaces represents real physical data. In practice, problems arise with the idealisation of components that interact only via directed signals. For mechanical systems for example, the preconditions are fullfilled only for those cases for which all rigid bodies are coupled via *elastic joints* (fig. 3 a). All bodies sink forces and torques as inputs and source positions and velocities at their outputs. This type of joint is very frequent in automotive environment, because it is often necessary to use rubber dampers and similar types of joints that decrease noise and vibrations and reduce peaks in transmitted accelerations. The second type of mechanical rigid body system met in automotive applications is an *open chain* (fig. 3 b). In this case it is not enough to transmit forces and geometric data between rigid bodies and joints. Auxilliary data representing the kinematic state of each body is sent along the chain. At the end of the chain constraint forces for the last body can be calculated and played back along the chain. Reaching the beginning of the chain, all forces are known and the integration step can be executed. This procedure is similar to that described in [1 and 4]. For



a) elastic joints

b) open chain

c) kinematic loop

Fig. 3: Types of mechanical rigid body systems.

our purposes we prefer to solve the underlying mathematical problem symbolically, using tools like Mathematica [7]. Of course the interfaces resulting from this procedure are somewhat problem specific. Replacing an elastic joint by a rigid one, not only changes the joint´s model but even influences the model of the connected rigid bodies. For practical purposes the effects are not so dramatic, because the changes often are inside complex components (a complete gear box for example) while the joints, where it is connected to the engine and the vehicle remain unchanged. To handle the case of *closed kinematic loops*, as shown in fig. 3 c, we need an extra block that coordinates the kinematics and the calculation of the constraint forces. This coordinator receives the positions, velocities and accelerations of all joints involved in the kinematic loop. This data can be used to calculate constraint forces as well as stabilizing terms according to Baumgarte [2 and 3]. The stabilized constraint forces are fed back to the joints and the bodies and integration can be performed. The stabilization in this case is necessary, because the calculation of constraint forces is performed on the basis of joint accelerations. Numerical errors during the integration of these accelerations to velocities and positions would cause instabilities. Numerical examples showed that for moderate integration step sizes the stabilization method works quite well. In practical applications this case doesn´t occur as often as expected. Typical spatial models of complete passenger vehicles include approximately two or three kinematic loops that can be handled very easily using this method.

4. INTERACTIVE REALTIME SIMULATION

As all elements of the model and the simulation environment are in fact objects, they can be handled very naturally. This means selecting a parameter and changing its value is as easy as connecting or disconnecting several blocks or adding or removing a group of blocks. These features are not restricted to the windows and graphic objects as with many so called object oriented tools, but are also valid for the behavioral model of each block with all connections and communication elements. All actions can be performed while the simulation is running in realtime.

Another aspect is using physical data at the interfaces. This allows one to split the complete model at these points and to replace the model of a component by the real hardware. Supplying data in realtime allows for hardware in the loop applications.

5. RESULTS

Several years of industrial application of ASCET in the area of research, development, test, quality management etc. have proven the advantages of a tool combining the practical aspects of realtime based hardware-in-the-loop experiments and the conceptual clearness of object-oriented modelling. It could have been shown that complex automotive models, like complete power-train models for example, can be realized and integrated in the existing development and production cycle.

6. REFERENCES

[1] Bae, D. S., A recursive Formulation for constrained Mechanical System Dynamics. Ph.D. thesis, University of Iowa, 1986.

[2] Baumgarte, J., Stabilisierung von Bindungen über Zwangsimpulse. ZAMM 62, 1982, 447-454.

[3] Baumgarte, J., Ostermeyer, G. P., Zur numerisch exakten Berücksichtigung innerer und äußerer Bindungen. ZAMM 65, 1985, T28-T29.

[4] Brandl, H., Johanni, R., Otter, M., A very efficient Algorithm for the Simulation of Robots and Simular Multibody Systems without Inversion of the Mass Matrix. IFAC Proceedings Series, 1988, Number 3, 95-100.

[5] Eppinger, A., Kasper, R., and Heinkel, H.M., Hardware-in-the-loop Design Techniques with ASCET. Esprit CIM, CIM-Europe workshop on computer integrated design of controlled industrial systems. Paris 26.4.-27.4. 1990.

[6] Eppinger, A., Kasper, R., Schnelle Echtzeitsimulation mit dem Softwarewerkzeug ASCET. Bosch-Zünder, 1992, Ausgabe 5, Robert Bosch GmbH, Stuttgart

[7] Wolfram, S., Mathematica: A System for Doing Mathematics by Computer. Addison-Wesley Publishing Company, New York, 1988.

OBJECT-ORIENTED PROGRAMMING TECHNIQUES IN VEHICLE DYNAMICS SIMULATION

Andrés KECSKEMÉTHY and Manfred HILLER
Fachgebiet Mechatronik
Universität Duisburg
47048 Duisburg
Germany
email: kecs@mechatronik.uni-duisburg.de

Abstract. Described in this paper is a novel approach for the object-oriented computer modelling of vehicle dynamics. The key feature of the method is to represent the mechanical system as a collection of concatenated transmission elements carrying out component-related tasks autonomously and in a coordinate-free manner. This is accomplished by applying differential geometric concepts to the generation of the dynamical equations. The result is a computer library which can be used as a building-block system for generating vehicle dynamics simulation programs.

1. INTRODUCTION

The modelling and simulation of complex mechanical systems in conjunction with non-mechanical components is a recurrent task in the design process of modern mechatronic systems. Much time and money are spent in developing and purchasing, respectively, computer programs which accomplish these tasks in an efficient and comfortable manner. In the past, simulation packages where mainly developed under the aspects of universality and comprehensiveness, yielding huge monolythic programs which were easy to apply to standard problems but very difficult to hand-tailor and even more to extend to non-foreseen issues. Today, application engineers pursue a different paradigm, which can be characterized by the divide-and-conquer strategies of early computer algorithm design [1], or, in modern terminology, by the *object-oriented* programming paradigm. In this approach, the goal is to divide a given problem in clearly demarked sub-problems, which can be solved independently, and whose solutions can be combined without re-visiting their imlementations. The engineer can re-use in this case previously done work, following the idea of a building-block system and pursueing new goals by assembling standardized components. A very promising and widespread methodology in this direction is *object-oriented design*, which has been the basis of many successful re-designs in computer science and numerical analysis. In many modern applications, there is no alternative to this approach, because the overall system would be too broad to be tackled efficiently by one program. Examples are combinations of mechanics with nonlinear control elements, hydraulics, complex tire models or aerodynamic effects under the premise of real-time simulation.

There have been isolated attempts to use object-oriented programming for mechanic modelling [8, 9]. However, the object-oriented idea was applied there more as a programming technology, while one of the key features of object-oriented design, namely the possibility of data-independent realization, has been left as an open issue. In this paper, a recent concept for achieving a coordinate-independent formulation of multibody dynamics by "dissecting" the system into small, independently solvable pieces called *kinetostatic transmission elements* [7, 5] is further pursued. The idea is to include the transmission of mass-properties into the set of quantities handled by the transmission elements. This is accomplished by regarding operations arising in the setting of differential geometry on RIEMANNian manifolds. The result is an intuitive and efficient programming environment which helps to speed-up the design process in vehicle dynamics substantially and is well-suited also for other application fields in which many different disciplines of modern engineering concur.

The rest of the paper is structured as follows: Section 2 gives a short description of the *client/server* paradigm underlying the present approach. In Section 3, a differential-geometric, coordinate-free representation of dynamics is elaborated, which is the basis of the object-oriented implementation discussed in Section 4. Section 5 shows the application of the derived concepts to vehicle dynamics simulation.

2. THE CLIENT/SERVER CONCEPT OF OBJECT-ORIENTED PROGRAMMING

The client/server model originates from object-oriented program design [12], but has been also applied to many other fields of engineering and computer science. It is a *responsibility driven* approach, as opposed to the *data driven* approach used in traditional programming, in that it is not the internal data or algorithmic structure of the modules which determines the design, but the decomposition of the overall problem into autonomous pieces whose behaviour can be determined by abstract "*services*". These services are transactions which are supplied by particular objects acting as "*servers*", and which are requested by another set of objects acting (momentarily) as "*clients*". The key idea is to describe these services at a very high and abstract conceptual level in order to leave the details as implementation-specific issues, which can be realized in different manners, but do not affect the overall functionality. Examples for such high-level transactions are services such as "`break`" (for a breaking system of a car), "`step`" (for a stepper-motor) or "`get me an ice`" (for a family father). The client/server model features many substantial advantages, among which are re-usability, extensibility, easy maintainance, and the possibility of rapid prototyping. Also, a high degree of "isomorphism" can be achieved between entities in the "problem domain", i.e. the real world being modelled, and objects within the "solution domain", i.e. the space in which the modelling takes place [3].

The application of the client/server model to multibody dynamics makes it necessary to describe the relationships in a coordinate-independent way. As shown in [7, 5], it is possible to model mechanical components independently of coordinate representations by regarding them as nonlinear elements capable of transmitting motions and forces. These "*kinetostatic transmission elements*" can be easily assembled to form systems of arbitrary complexity. It is even possible to solve the motion and force closure relationships in multibody loops and also the inverse and direct dynamics problems of general multibody systems purely by applying these notions. As shown in the following sections, it is also possible to regard the generation of dynamical properties such as the generalized mass or the generalized forces as self-contained "services" which can be carried out automously by specialized objects acting on RIEMANNian manifolds.

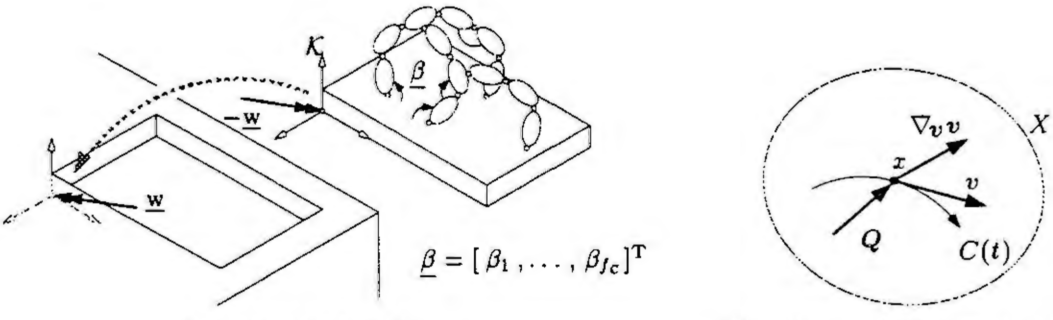
3. DIFFERENTIAL-GEOMETRIC MODELLING OF A MECHANICAL COMPONENT

3.1. Structure of a Mechanical Component

A typical mechanical component $\mathcal{C}$ consists of any number of articulated bodies (Fig. 1). The motion of the bodies can be described by the spatial motion of a "carrier body", represented by the reference frame $\mathcal{K} \in \mathrm{SE}(3)$, where SE(3) is the Euclidean group of rigid-body displacements, and f_C "inner" independent coordinates $\underline{\beta} = [\beta_1, \dots, \beta_{f_C}]^{\mathrm{T}}$, representing the relative motion of the bodies with respect to the carrier body. The coordinates $\underline{\beta}$ may themselves be rigid-body displacements or scalar joint coordinates $\beta_i \in \mathbb{R}$ or $\beta_i \in T^1$ (the one-dimensional torus), depending on whether the natural displacement is a translation or a rotation (as for e.g. for prismatic or revolute joints), respectively. Altogether, the state of the dynamical component is described as an n-dimensional smooth differentiable manifold X having the structure of a product space

$$X = \underbrace{\mathrm{SE}(3) \times \cdots \times \mathrm{SE}(3)}_{n_{\mathcal{K}} \text{ times}} \times \underbrace{\mathbb{R} \times \cdots \times \mathbb{R}}_{n_R \text{ times}} \times \underbrace{T^1 \times \cdots \times T^1}_{n_{T^1} \text{ times}} \quad . \tag{1}$$

The solution of the dynamical equations represents a trajectory $C(t)\colon \mathbb{R} \to X : t \mapsto x \in X$, x corresponding to the momentary position of the component. The component typically results from



(a) mechanical model

(b) differential-geometric model

Figure 1: Model of a dynamical component.

detaching a subsystem from an overall system. After analyzing it locally, it can be re-attached by providing a six-dimensional *wrench* $\underline{w} = [\boldsymbol{\tau}, \boldsymbol{f}]^{\mathrm{T}}$ containing the torque $\boldsymbol{\tau}$ and the force $\boldsymbol{f}$ necessary to constrain the movement of the carrier frame such that it is fixed to a predecessor body.

3.2. Basic Intrinsic Geometric Elements

For an intrinsic formulation of the dynamical equations, the notions of tangent and cotangent vectors, as well as the concept of the metric are introduced. *Tangent vectors* are viewed as derivations of real-valued functions on X, forming at each point $x \in X$ an n-dimensional linear vector space T_x termed the *tangent vector space* [2]. The basis of the tangent vector space is given by the partial derivatives

$$\widehat{e}_i = \frac{\partial}{\partial \pi^i} = a_i^j \frac{\partial}{\partial x^j} \quad , \quad i = 1 , \ldots , n \ , \ j = 1 , \ldots , m \ , \qquad (2)$$

where $\underline{x} = [\, x^1 , \ldots , x^m \,]^{\mathrm{T}}$ are a set of coordinates and $\underline{\pi} = [\, \pi^1 , \ldots , \pi^n \,]^{\mathrm{T}}$ may represent *pseudocoordinates*, i.e. non-integrable coordinates. The tangent vectors for rotation and translation of the Euclidean group, respectively, are given in a body-fixed reference system (ξ, η, ζ) by the formulas [7]:

$$\begin{array}{llllllll} \text{Rotation:} & \widehat{e}_1 & = & \zeta \dfrac{\partial}{\partial \eta} - \eta \dfrac{\partial}{\partial \zeta} \ , \ \widehat{e}_2 & = & \xi \dfrac{\partial}{\partial \zeta} - \zeta \dfrac{\partial}{\partial \xi} \ , \ \widehat{e}_3 & = & \eta \dfrac{\partial}{\partial \xi} - \xi \dfrac{\partial}{\partial \eta} \ , \\ \text{Translation:} & \widehat{e}_4 & = & \dfrac{\partial}{\partial \xi} \ , \ \widehat{e}_5 & = & \dfrac{\partial}{\partial \eta} \ , \ \widehat{e}_6 & = & \dfrac{\partial}{\partial \zeta} \ . \end{array} \qquad (3)$$

In particular, the six-dimensional spatial velocity vector (the "twist") $\underline{t} = [\omega, v]^{\mathrm{T}}$, $\omega = [\omega_\xi , \omega_\eta , \omega_\zeta]^{\mathrm{T}}$ being the angular velocity and $v = [\, v_\xi , v_\eta , v_\zeta \,]^{\mathrm{T}}$ being the linear velocity, has the representation

$$\underline{t} = \omega_\xi \, \widehat{e}_1 + \omega_\eta \, \widehat{e}_2 + \omega_\zeta \, \widehat{e}_3 + v_\xi \, \widehat{e}_4 + v_\eta \, \widehat{e}_5 + v_\zeta \, \widehat{e}_6 \ .$$

Cotangent vectors are linear functions mapping the tangent vector space T_x to the set of real numbers. At each point $x \in X$, the cotangent vectors form an n-dimensional linear vector space T_x^* denoted the *cotangent vector space*. The basis of the cotangent vector space is called the cobasis and consists of a set of vectors $\check{e}^i \in T_x^*$ having the property

$$\check{e}^i(\, \widehat{e}_j \,) = \delta_j^i \quad , \quad \delta_j^i = \begin{cases} 1 & \text{for } i = j \\ 0 & \text{else} \end{cases} \quad , \quad i, j = 1 , \ldots , n \ . \qquad (4)$$

The typical representatives of cotangent vectors in mechanics are the forces and moments. For the Euclidean group, the force vector is represented by the wrench and is decomposed in terms of the cobasis corresponding to the basis $\widehat{e}_j$ given above as

$$\underline{w} = \tau_\xi \, \check{e}^1 + \tau_\eta \, \check{e}^2 + \tau_\zeta \, \check{e}^3 + f_\xi \, \check{e}^4 + f_\eta \, \check{e}^5 + f_\zeta \, \check{e}^6 \ . \qquad (5)$$

By endowing the manifold X at each point with a nondegenerate metric, one obtains a RIEMANNian manifold. For such a metric, any bilinear, symmetric, nondegenerate form can be used. In the case of the dynamics of mechanical systems, the metric to be utilized is the *kinetic energy*

$$g_x(\, v , v \,) \doteq 2\, T(\, \underline{x} ; \dot{\underline{\pi}} \,) = \dot{\underline{\pi}}^{\mathrm{T}} M(\underline{x}) \, \dot{\underline{\pi}} \ , \qquad (6)$$

where $M(\underline{x})$ is the generalized mass matrix with respect to the coordinates $\underline{x}$ at the point x.

3.3. Local Dynamical Equations

Given a metric $g_{i\ell}$ and a cotangent vector $\underline{Q} = [Q_1, \ldots, Q_f]^{\mathrm{T}}$ representing the applied forces being exerted upon the dynamic system, the equations of motion can be represented compactly as [10]:

$$g_{i\ell} \, [\nabla_{\boldsymbol{v}} v]^\ell = g_{i\ell} \frac{\mathrm{d}v^\ell}{\mathrm{d}t} + \gamma_{ijk} \, v^j v^k = Q_i \ , \qquad (7)$$

where $[\nabla_{\boldsymbol{u}} v]^\ell$ is the ℓ-th component of the covariant derivative of a velocity field v in direction of another velocity field u, given in coordinates by

$$\nabla_{\boldsymbol{u}} v = u^k \left[\frac{\partial v^i}{\partial \pi^k} + \gamma_{kj}^i \, v^j \right] \widehat{e}_i \quad , \quad \text{with} \quad v = v^i \, \widehat{e}_i \quad , \quad u = u^i \, \widehat{e}_i \ . \qquad (8)$$

The coefficients γ_{ijk} result [2] as

$$\gamma_{ijk} = \frac{1}{2} \left(\frac{\partial g_{ij}}{\partial \pi^k} + \frac{\partial g_{ik}}{\partial \pi^j} - \frac{\partial g_{jk}}{\partial \pi^i} \right) - \frac{1}{2}(g_{kl} \, C_{ji}^l + g_{jl} \, C_{ki}^l + g_{il} \, C_{jk}^l) \quad , \quad \text{with} \quad \gamma_{ijk} = g_{i\ell} \, \gamma_{jk}^\ell \ , \qquad (9)$$

where the C_{jk}^i are the *structure coefficients* related to the basis of the tangent vector space. They are defined via the relationship

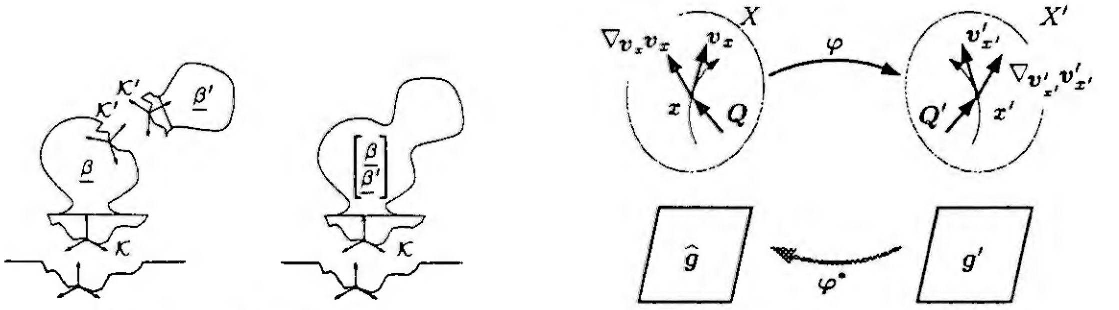
$$[\, \widehat{e}_j \, , \widehat{e}_k \,] = C_{jk}^i \, \widehat{e}_i \ , \qquad (10)$$

where $[\, \widehat{e}_j \, , \widehat{e}_k \,] \doteq \widehat{e}_j \, \widehat{e}_k - \widehat{e}_k \, \widehat{e}_j$ is the LIE bracket. It can be verified that in case of a "natural basis", $\widehat{e} \equiv e_i = \partial / \partial x^i$, the structure coefficients vanish, whereas in the case of the basis of the Euclidean group given in Eq. (3) one obtains

$$C_{jk}^i = \begin{cases} 1 & \text{if } \{i, j, k\} \text{ or } \{i-3, j, k-3\} \text{ is a cyclic permutation of } \{1, 2, 3\} \\ -1 & \text{if } \{i, j, k\} \text{ or } \{i-3, j, k-3\} \text{ is a noncyclic permutation of } \{1, 2, 3\} \\ 0 & \text{else} \end{cases} \ . \qquad (11)$$

3.4. Global Dynamical Equations

Global dynamical equations result from assembling mechanical components together. In order to be able to use the intrinsic properties of the components without resorting to coordinate-dependent expressions, one views the assembly of two mechanical components as a mapping φ from one manifold X , called the *input* manifold, representing the coordinates of the inboard mechanical component to another manifold X' , called the *output* manifold, representing the coordinates of the outboard mechanical component (Fig. 2).



(a) mechanical model

(b) differential-geometric model

Figure 2: Coupling of two mechanical components.

Let the dynamical equations be locally given with respect to the input and output manifolds, respectively, as

$$g_{i\ell}\frac{\mathrm{d}v^\ell}{\mathrm{d}t} + \gamma_{ijk}\, v^j v^k = Q_i \ , \tag{12}$$

$$g'_{i\ell}\frac{\mathrm{d}v'^\ell}{\mathrm{d}t} + \gamma'_{ijk}\, v'^j v'^k = Q'_i \ . \tag{13}$$

It is well known that, together with the mapping φ transmitting points $x \in X$ to points $x' \in X'$, there is an associated mappings φ_*: $v_x \in T_x \mapsto v'_{x'} \in T'_{x'}$, called the *differential mapping*, mapping tangent vectors defined with respect to X to tangent vectors defined with respect to X' , as well as a mapping φ^*: $Q' \in T^{*\prime}_{x'} \mapsto Q \in T^*_x$, called the *pull-back function*, transmitting cotangent vectors defined with respect to X' to corresponding cotangent vectors defined with respect to X such that $Q\,(v_x) \equiv Q'\,(v'_{x'})$. This last expression represents the condition of power-free transmission of velocities and forces, as the application of a cotangent vector (here: a force) to a tangent vector (here: a velocity) yields the mechanical quantity of power, which is set to be equal at the input and output of φ [7].

In analogy to the transmission of cotangent vectors, there exists also a pull-back function φ^* for the transmission of metrics. This function maps the metric g' at the output manifold X' to the corresponding *induced metrix* $\hat{g}$ at the input manifold X , such that the relation

$$\hat{g}_x\,(v_x^{(i)}, v_x^{(j)}) \doteq g'_{\varphi(x)}(\,\varphi_*(v_x^{(i)}), \varphi_*(v_x^{(j)})\,) \tag{14}$$

holds for any pair of vectors $v_x^{(i)}, v_x^{(j)}$ at the input. The coefficients of $\hat{g}$ are given as the numbers

$$\hat{g}_{ij} = \hat{g}_x\,(\hat{e}_i, \hat{e}_j) \ . \tag{15}$$

A simple procedure for obtaining these coefficients is as follows: For $i = 1, \dots, n$ do: [Step 1] Transmit the tangent vector $v_x = \hat{e}_i$ as $\hat{e}'_i = \varphi_*(v_x)$ to X' . [Step 2] Calculate a corresponding cotangent vector α'_i such that $\alpha'_i(v'_{x'}) \equiv g'(\hat{e}'_i, v'_{x'})$ for all $v'_{x'} \in T'_{x'}$ (e.g. by multiplication of $\hat{e}'_i$ with the generalized mass matrix). [Step 3] Transmit the cotangent vector α'_i back to X as $\hat{\alpha}^i$. [Step 4] Evaluate

$$\hat{g}_{ij} = \hat{\alpha}^i(\hat{e}_j) \tag{16}$$

for $j = i, \dots, n$ locally at X . This procedure is the generalization of an approach first proposed in [11] for generating the direct dynamics of serial manipulators by means of the inverse dynamics.

A further quantity which must be generated when coupling two manifolds repesenting mechanical components arises from traveling along the trajectory $C(t)$ with constant velocity v_x , and looking at the resulting acceleration at the output manifold:

$$\eta' = \left.\frac{\mathrm{d}v'_{x'}}{\mathrm{d}t}\right|_{v^\ell\, =\, \mathrm{const.}} \ . \tag{17}$$

Using the metric g' at X', one can evaluate again a related force $\widehat{Q}$ featuring the property $\widehat{Q}(v'_{x'}) \equiv g'(\eta', v'_{x'})$ for all $v'_{x'} \in T'_{x'}$. This quantity corresponds to the generalized gyroscopic forces induced by the coupling. Now all ingredients for the coupling of the two systems are at disposition. One obtains the dynamical equations:

$$\left[g_{i\ell} + [\varphi^*(g')]_{i\ell}\right] \dot{v}^\ell + \left[\gamma_{ij\ell}\, v^j v^\ell + (\varphi^* \widehat{Q}')_i\right] = Q_i + [\varphi^* Q']_i \quad . \tag{18}$$

4. OBJECT-ORIENTED REALIZATION OF THE DISCUSSED CONCEPTS

According to the previous exposition, a coordinate-free, object-oriented modelling of mechanical systems is possible by supplying at a minimum the following two categories of objects together with the corresponding "services":

(I) Input/Output Objects (termed "Metric")

(A)	doMetric:	calculate the coefficients of the metric g_{ij}
(B)	doConnection:	calculate the term $\gamma_{ijk}\, v^j v^k$
(C)	doForce:	calculate the applied forces Q_i
(D)	doInverse:	resolve the dynamical equations for $\dot{v}^i$

(II) Transmission Elements (termed "DynamicMap")

(A)	doMotion:	transmit position, velocity and/or acceleration vectors $x' = \varphi(x)$, $v'_{x'} = \varphi_*(v_x)$, $\dot{v}'_{x'} = \varphi_*(\dot{v}_x) + \eta'$
(B)	doForce:	transmit force (cotangent) vectors $Q = \varphi^*(Q')$
(C)	doPullBackMetric:	calculate the induced metric $\widehat{g} = \varphi^*(g')$
(D)	doConnection:	calculate the acceleration term η'

The different types of mechanical components represent merely particular ways of implementing these functions, even if the given objects possess additional functions which are necessary for their internal functioning. The algorithmic structure of the objects is considered to be an "inner" detail of the objects and not relevant for the global coupling. In this way, it is possible to incrementally extend an initial and rudimentary library to systems of virtually any complexity, as is indeed being carried out currently for a variety of systems. One of these application fields is described in the following section.

5. Application to Vehicle Modelling

By the concepts derived in the previous sections, it is posible to assemble a vehicle model from pre-modelled parts of varying complexity. Fig. 3 shows as an example the components chosen for the modelling of a BMW 535i. Further details are elaborated in [6].

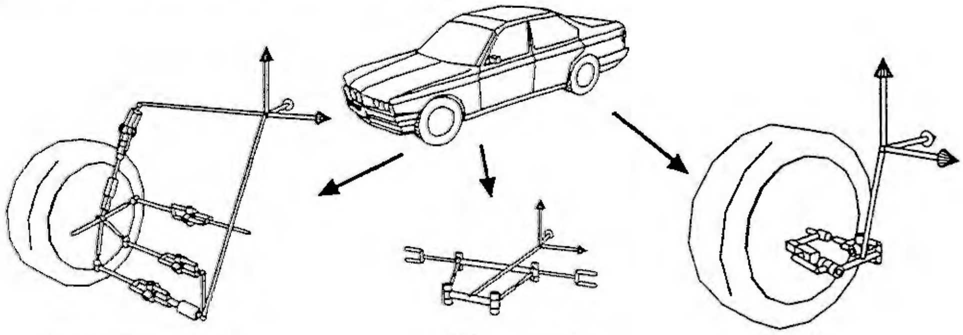


Figure 3: Components of the modeled vehicle BMW 535i.

For the overall simulation, the modelling of purely mechanical components is, though a relatively complex task, only a small portion of the problem. Fig. 4 exhibits a view of a minimum set of objects necessary for a functional vehicle simulation environment [4]. In this representation, the advantage of the object-oriented approach becomes evident, as topics from very different fields of engineering can be combined without having to re-visit the details of their representation or to prescribe a common data representation. Here, the building-block analogy becomes evident, which is indeed materialized by implementing the exhibited entities as autonomously operating objects which act as "servers" for the "clients" lying above in the figure.

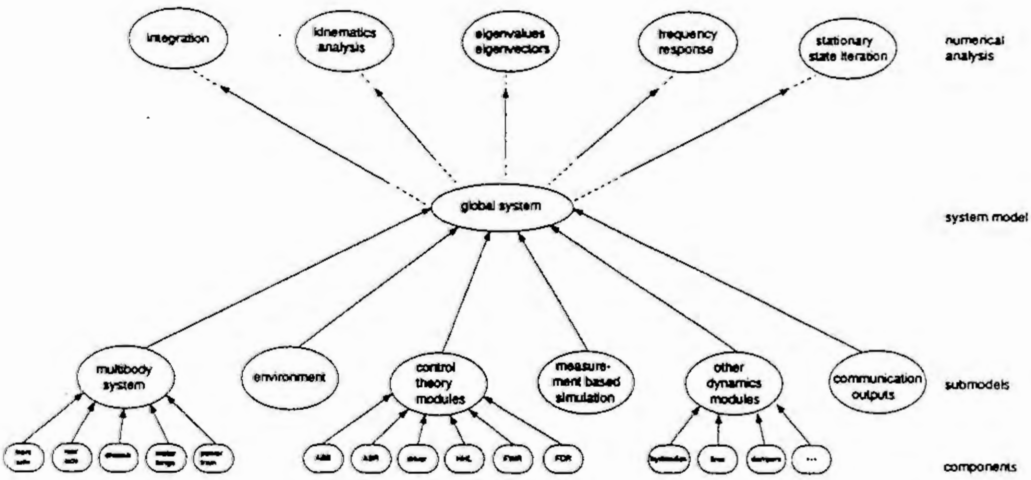


Figure 4: Components of a general vehicle dynamics simulation program.

6. Conclusions

The methods discussed in this paper are suited for realizing data-independent, object-oriented modellings of general multibody systems. Following the client/server approach of object-oriented programming, the overall relationships are described by two categories of objects, namely (a) transmission elements and (b) corresponding input and output manifolds. An intrinsic model, where the corresponding "services" are carried out in a coordinate-independent manner, is derived on the basis of a novel combination of well-known concepts of mechanism analysis and differential geometry. The concepts have been implemented with the object-oriented programming language C++. This implementation is the basis of a general package for the simulation of the dynamics of modern passenger vehicles, which is being currently completed. The object-oriented, coordinate-independent modelling has also proved very valuable for the design of flexible, intuitive and effective computer models of a variety of mechatronic systems.

References

[1] A.V. Aho, J.E. Hopcroft, and J.D. Ullman. *The Design and Analysis of Computer Algorithms.* Addison-Wesley Publishing Company, Reading, 1974.

[2] Yvonne Choquet-Bruhat and Cècile DeWitt-Morette. *Analysis, Manifolds and Physics. Part I: Basics.* North-Holland, Amsterdam, New York, 1989.

[3] James O. Coplien. *Advanced C++ Programming Styles and Idioms.* Addison-Wesley Publishing Company, Reading, Massachusetts, 1992.

[4] M. Hiller, K.-P. Schnelle, and A. van Zanten. Simulation of nonlinear vehicle dynamics with the modular simulation package FASIM. In *IAVSD Symposium*, Lyon, 1991.

[5] A. Kecskeméthy and M. Hiller. Object-oriented approach for an effective formulation of multibody dynamics. In *Second U.S. National Congress on Computational Mechanics, August 16–18*, Washington D.C., 1993. To appear in *Comupter Methods in Applied Mechanics and Engineering.*

[6] A. Kecskeméthy and M. Hiller. An object-oriented tool-set for the computer modeling of vehicle dynamics. In *Proceedings of the 26th International Symposium on Automotive Technology and Automation (ISATA), 13–17 September*, Aachen, Germany, 1993.

[7] Andrés Kecskeméthy. *Objektorientierte Modellierung der Dynamik von Mehrkörpersystemen mit Hilfe von Übertragungselementen.* PhD thesis, Universität - GH - Duisburg, 1993.

[8] M. Otter, H. Elmqvist, and F.E. Cellier. Modelling of multibody systems with the object-oriented modelling language Dymola. In M.S. Pereira and J.A.C. Ambrósio, editors, *Proceedings of the NATO-Advanced Study Institute on Computer Aided Analysis of Rigid and Flexible Mechanical Systems*, volume II, pages 91–110, Tróia, Portugal, 27 June – 9 July 1993.

[9] Kai Sorge. *Mehrkörpersysteme mit starr-elastischen Subsystemen.* Fortschritt-Berichte VDI, Reihe 11, N. 184. VDI-Verlag, Düsseldorf, 1993.

[10] Cornelius von Westenholz. *Differential Forms in Mathematical Physics*, volume 3 of *Studies in Mathematics and its Applications.* North-Holland, Amsterdam, New York, Oxford, revised edition, 1981.

[11] M.W. Walker and D.E. Orin. Efficient dynamic computer simulation of robotic mechanisms. *Journal of Dynamical Systems, Measurement and Control*, 104:205–211, September 1982.

[12] Rebecca Wirfs-Brock and Brian Wilkerson. Object-oriented design: A responsibility-driven approach. In *OOPSLA '89 Proceedings*, pages 71–75, October 1989.

Systematic Treatment of Complex Mechatronic Structures, Exemplified by Automotive Engineering - Proposal of a Standardized Mechatronic Wheel Suspension

Reinhard Vullhorst

University of Paderborn
MLaP - Mechatronics Laboratory Paderborn
Prof. Dr.-Ing. Joachim Lückel
Pohlweg 55, D - 33098 Paderborn, Germany

Abstract

The term mechatronics stands for a novel discipline integrating mechanical structures and efficient information processing and thus meeting the ever increasing demands on technical systems. Another far-reaching intention, especially in the context of a *redesign of technical systems*, is to shift tasks usually attributed to mechanical engineering into the computer. Exemplified by wheel suspension, as being the central aggregate of vehicle dynamics, a proposal for a standardized mechatronic construction unit will be developed; the latter, via consistent use of information processing, will allow simplification of the mechanical structure while ensuring optimal operativeness through the use of active components.

Basic Functions of a Wheel Suspension

The main functions of the chassis are roughly the following:

support
guidance
suspension
damping
steering
drive
brake
insulation (noise).

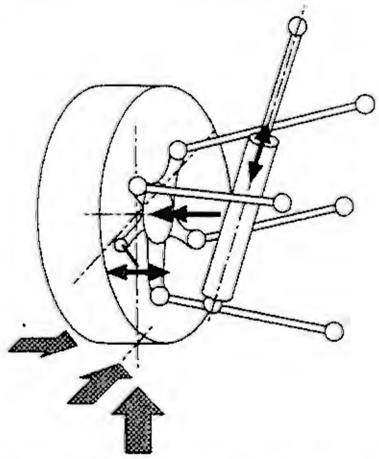


Fig. 1: Structure of an ordinary wheel suspension

At the point of contact of road surface and wheels, the forces become evident that, conducted along the wheel suspension, influence the entire vehicle dynamics. The **longitudinal forces** are mostly due to drive and brake forces diverted via the wheel. They are at the root of the longitudinal motion, but also of the pitching of the chassis. If the drive torques at the wheels can be altered independently of one another, the yaw of the vehicle can also be influenced.

The **lateral forces** arise in steering maneuvers. Generally speaking, lateral forces are related to the slip angle at the wheel; the effect of forces perpendicular to the lateral force, such as longitudinal slip and wheel load, on the lateral forces must not be neglected either. Steering, apart from its general task of making the vehicle follow a certain course, is able to alter lateral displacement and yawing of the car by means of an alteration of the lateral force.

The **vertical forces** result from the static and dynamic loads of the sprung and unsprung masses. They have a decisive effect on riding comfort and wheel loads. The degrees of freedom influenced in vertical direction are: lifting, rolling, and pitching of the chassis, but also the motion of the wheels. The passive mechanical transmission structure is the basis of wheel suspension. The following table will display the results of a comparison between some important conventional wheel suspension systems /Henker 93/, /Reimpell 88/:

	diagonal control arm (swing axle)	McPherson	parallelogram suspension	multi-link suspension
expenditure[1]	low +	higher -+	high -	high -
space required	small +	small +	large -	large -
kinematics[2] (safety)	insufficient -	satisfactory -+	good +	very good ++
comfort[3]	satisfactory -+	satisfactory -+	good +	high ++
elasto-kinematics	can be influenced -	satisfactory -+	good +	very good ++
propagation	-/-	about 90 % of all front-wheel suspensions	-/-	-/-

Table 1: Comparison between important conventional wheel suspensions

Comments on Table 1:

Further safety and comfort can only be reached through the appropriate amount of expenditure for construction and manufacturing and therefore with more space required.

The McPherson wheel suspension represents a good compromise between economy, safety, and comfort.

Sophisticated wheel suspensions assign different functions to many components which then yield more possibilities of variation (space-link suspension).

Simpler wheel suspensions, such as the McPherson strut, combine several functions within one component (integration of functions): in addition to its usual tasks, the strut provides wheel location; thus, the upper lateral control arms can be spared and space saved. Yet, the strut is subject to bending forces, and the resulting higher friction in the seals has an effect on the spring behaviour.

Although wheel suspension, in its conventional passive structure, has reached a high degree of perfection in view of functionality and price calculation, the standard build-in of active components has increasingly been called for following customers' wishes for higher safety standards (better traction in ABS) and more comfort (power steering).

[1] Here, "expenditure" means the expenses and efforts necessary for construction and manufacturing.

[2] Kinematics of a wheel suspension can be described by the change in steering angle, camber, and caster offset in dependence of the spring deflection. It has a decisive effect on the steering, rolling, and pitching behaviour of the car body.

[3] Comfort is noticeably influenced by suspension and noise insulation.

The demand for lower consumption compels to reduce the weight of vehicles. At the present level of research in conventional suspension systems this will lead to a loss in riding comfort. To avoid this, the installation of an active suspension is indispensable; yet, additional energy consumption and the weight of the new components must not even up the effect of weight reduction. Wheel suspension offers many possibilities for interference with which all degrees of freedom of a vehicle can be reached. Many active systems are already standard in the chassis. Examples of active systems:

active systems (chassis):	control variables:
power steering	drag link distance (rack force)
anti-blocking systems	braking force
anti-spin regulation	tractive force
active suspension	pressure in the cylinder

Such *mechatronic* systems consist generally of four functional groups. Exemplified by the active wheel suspension which is itself complemented by active components, the functional groups are the following (cf. Fig. 3):

Mechanical supporting and joint structures (passive conventional wheel suspension),

aktuators (power steering, ABS braking module, active strut, etc.),

sensors (pressure sensors, acceleration sensors, etc.),

compensators (digital information processing).

The dynamical behaviour of mechatronic systems has to satisfy very high requirements. The conclusions to be drawn include two basic tasks:

Appropriation of kinematic behaviour by means of the passive and the active components (**kinematics** of the mechanism).

Appropriation of a suitable behaviour under the influence of forces (**dynamics**) by means of actuators and sensors as signal converters as well as compensators and filters for information processing.

Contrary to traditional development processes that keep to the construction and the shape of the projected item, the design of mechatronic systems has to start out from a function-oriented formulation of the problem, as information processing cannot be determined by the construction. Proceeding in this way requires suitable design tools, in order that the problems can be described and treated.

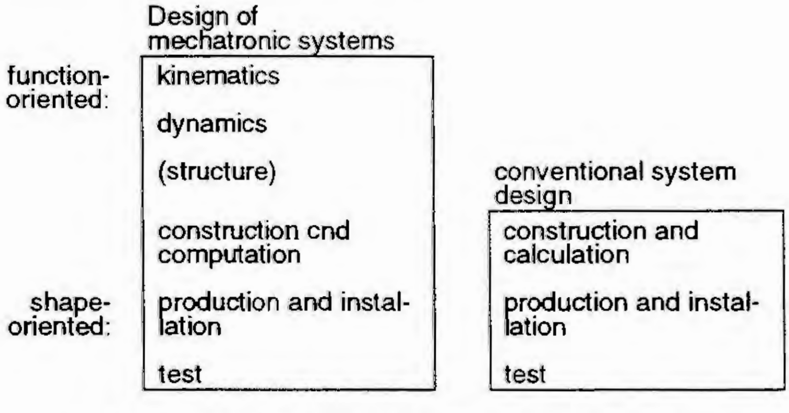

Design of mechatronic systems
function-oriented:
kinematics
dynamics
(structure)
construction cnd computation
shape-oriented:
production and instal-lation
test
conventional system design
construction and calculation
production and instal-lation
test

Fig. 2: The design of mechatronic and of conventional systems

In the future, problem solving will have to be fast and flexibly adapted to changing problems. It is quite conceivable that a wheel suspension module will have to be adapted to different variations of a construction series or even to different construction series. Thus, one and the same construction principle covers a whole range of applications, and scaling is done exclusively via information processing: **possibility to scale the system.**

Moreover, future interest will focus on the integration of the active components, such as steering, ABS, damping, and drive (etc.) and thus on the tuning of all systems influencing one another: **possibility to integrate the system**.

Concept of a Mechatronic Function Module "Compound Wheel Suspension"

Modern vehicles are more and more mechatronic systems. In order to treat the complexity and the largeness of such systems effectively and to make them manageable, structuring in view of modularity and hierarchy is indispensable. On every hierarchical level, mechatronic function modules (see above *examples*) can be found whose elements are a combination of mechanical supporting and joint structures, actuators, sensors, and compensators (digital information processing).

Functional groups of this kind can be defined as standardized MECHATRONIC FUNCTION MODULES (**MFM**) /Castiglioni 92/, /Lückel 92/. As independent blocks, they are marked by their function, its realization by means of the system-typical elements described above and their interface. In general, the interface comprises mechanical, energetical, and information-processing elements.

A MFM "active wheel suspension" consists e. g. of the mechanical structure (control arm, crossmember, etc.), the actuator (hydraulic cylinder, steering, ABS), sensors (pressure, displacement, and acceleration sensors) and compensators (controller, observer, measuring filter).

Mechatronic systems, such as ABS or power steering, are generally accepted today and employed as series products. A similar development can also be foreseen for the wheel suspension (independent suspension) as a central component with clearly defined functions. Fig. 3 represents such a MFM (exemplary with a space-link suspension as supporting and joint structure):

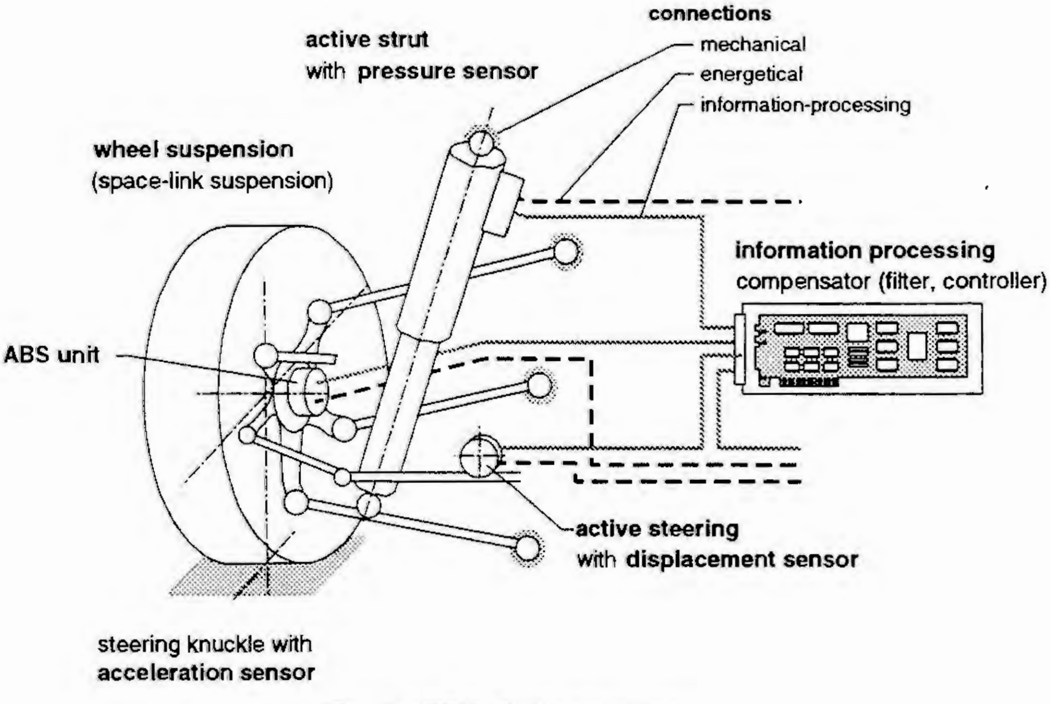


Fig. 3: MFM wheel suspension

The potential of mechatronic systems resides in an informational coupling of their elements and thus in the possibility to purposefully optimize the base functions and to achieve properties which surpass by far those of the passive structure.

For a novel mechatronic wheel suspension - *called MFM compound wheel suspension* - this would mean:

- **Redesign with highly improved properties by consistent use of information processing.**
- **With this, a simplification of the mechanical structure**, as active components compensate for the drawbacks of the mechanical structure.

Information processing allows to shift functions usually attributed to mechanics into the computer, thus reducing the complexity of the mechanical structure.

Simplified Mechanical Supporting and Joint Structure of the MFM Compound Wheel Suspension

The base functions of a wheel suspension are determined by the kinematic behaviour of the supporting and joint structure. The efficiency of the system depends on the possibilities of active intervention that allow to optimize properties of the wheel suspension by information processing. Distribution of tasks between mechanics and information processing is variable. Thus, it is most important to influence all degrees of freedom of the system independently of one another:

Degrees of freedom:

rotation:	active influence on braking and driving forces (ABS, anti-spin regulation)
steering:	active steering (to be provided as a standard for front axle and rear axle)
vertical motion:	active strut as actuator *without guidance functions*

For an optimal functioning, the active strut has to be relieved of the task of guiding the wheel. Thus, the complex task of guiding hub carrier and wheel rests with the passive mechanical structure. A comparison of the conventional wheel suspensions (see above) illustrates the conflict of goals for a wheel suspension between accomplishing the proper function (comfort and safety) and requiring space and effort.

The requirement the structure has to meet is the following: The wheel suspension has to require the smallest space possible (simple structure, few components), so that there is more liberty in arranging the remaining aggregates. Here, mechatronics offers the decisive advantage: The active actuator compensates for the deterioration of the kinematic properties which results from the simplification of the structure.

Fig. 4 shows a proposal for such a simplified mechanical structure as part of the MFM compound wheel suspension. Constructive details, such as the shape of the wheel-locating control arm (closed component or consisting of several individual control arms) or the shape of the rotational axis, are not dealt with in this context. Because of the clear distinction between wheel location, support, and steering, the guiding mechanism has to support the wheel and ensure the steerability. In order to waste as little space as possible, a swing axle in the position of a lower control arm (cf. Fig. 4) was chosen.

This suspension is going to be called "wheel-locating transverse link suspension". The advantage of a passive structure extended by information processing, sensors, and actuators, resides in the possibility to compensate for shortcomings by active interference and eventually to obtain satisfactory geometrical (space required), kinematic, and dynamical properties.

The MFM Compound Wheel Suspension

On the basis of the wheel-locating transverse link suspension, the MFM compound wheel suspension defines a standardized wheel suspension module as central mechatronic aggregate in vehicle construction. The combination of simplified mechanical structure, active components and sensors with an efficient information processing offers possibilities to adapt a wheel suspension very flexibly to changing conditions and fields of application (Fig. 4)

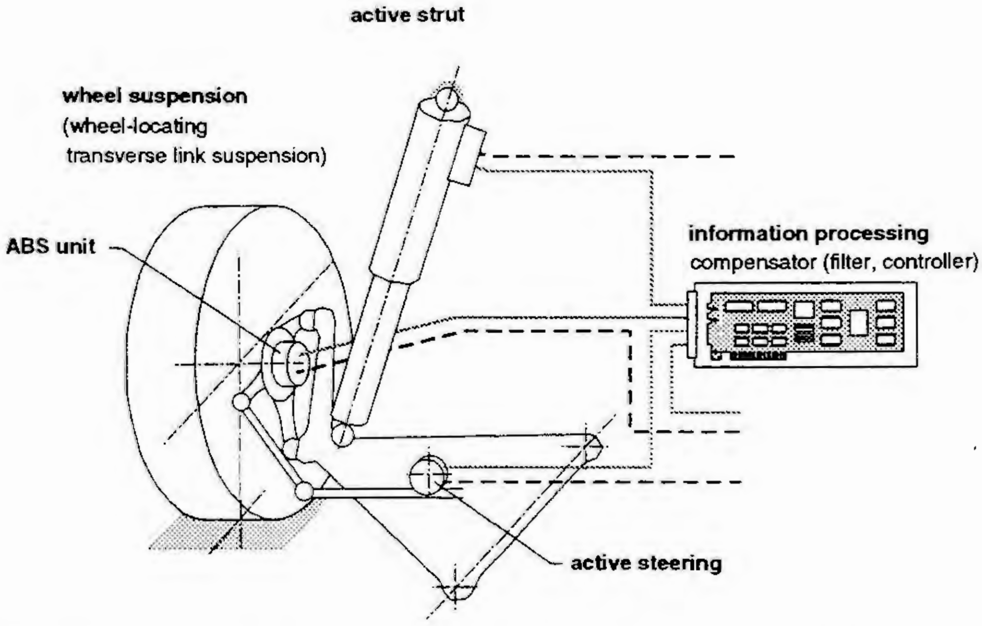


Fig. 4: MFM compound wheel suspension with wheel-locating transverse link suspension

Several variations of such a mechatronic structure "compound wheel suspension" can be thought of to reach a complexity matching that of the problem:

1 - MFM compound wheel suspension, with active steering only

2 - MFM with active steering and ABS

3 - MFM with active steering, ABS, and active damping

Now, the design of mechatronic systems requires a design environment allowing the computerized description of the problem and providing analysis and synthesis tools to meet the projected system properties on this basis. The software tools developed at the Department of Automatic Control (*MLaP*) and subsumed in the *C*omputer-*A*ided *Me*chatronic *L*aboratory (*CAMeL*) /Jäker 91/ offer such a design environment.

Example: Complete Vehicle

A vehicle model, based on the MFM compound wheel suspension presented above, serves as an example. It comprises complete longitudinal, transverse, and vertical dynamics and takes into account the properties of the nonlinear spatial wheel suspension kinematics modelled in analogy to the structure of the wheel-locating transverse link suspension. The wheel suspensions of front axle and rear axle are identical. It is of course possible to exchange the mechatronic wheel suspension for other suspension types, just as it is possible to extend the vehicle model by aggregates of motive power engineering, etc. Support for the user comes from large model libraries where the models of e. g. complete space-link or parallelogram suspensions are stored.

The example demonstrates the treatment of extremely complex structure in the shape of a model organized in a modular as well as hierarchical way. The physical aspect of the system remains intact in the model structure and allows approach to internal variables, such as the forces acting on the control arm or the deformations of a rubber mounting.

The MFM compound wheel suspension is based on a simplified mechanical structure whose drawbacks are to be compensated for by active intervention. The wheel-locating transverse link suspension, e. g., has the drawback of a vast change in camber and track width in dependence of the spring deflection. The lateral forces coming into effect with the spring deflection impair the tracking abilities as well as the car body motion and thus driving safety and ride comfort. It remains to be shown if the advantages of the simple axle (expenditure and space required) can be counterbalanced by the potential inherent in a mechatronic structure.

Modelling

For a representation of techno-physical systems in the computer, the models are available in symbolic form as nonlinear, dynamical equation systems in state-space form (block-oriented); they are organized in a modular and hierarchical way. The basis is the system description language *DSL* (*D*ynamic *S*ystem *L*anguage) /DSL 92/ developed at the *MLaP*.

Because of the special structure of vehicle systems in view of their couple elements, e. g. elasto-kinematic wheel suspensions, the couplings within the multibody system to be modelled are formulated as dynamical constraints. There is another method with which it is also possible to formulate kinematic_constraints and, in connexion with them, closed loops in a modular way which can eventually be structured hierarchically /Junker 94/.

Model structure:	
rigid bodies with mass:	car body, hub carrier, rim
couple elements without mass:	transverse link, rubber mounting, strut and tie rod as actuators, nonlinear tyre model, MFM compound wheel suspension as aggregate (couple system), formulated with basic systems
environment model:	street model for deterministic and stochastic excitation signals
controller model:	output vector feedback to influence the active tie rod (input: Lenkstangenweg) and the active strut (input: length of spring)
68 differential equations (DE) of 1st order for the mechanical structure 37 DE of 1st order for the environmental model (all systems described in *DSL* syntax)	

Tabelle 2: Model structure of the vehicle model

Analysis and Synthesis

The vehicle model is available as a nonlinear, parametrized, dynamical system in symbolic form. It is not nonlinear simulation (program package *SIMEX*: *SIM*ulation *EX*pert) /Jäker et al. 91/ which can be used for preliminary studies on the dynamical behaviour, but synthesis of the compensators which will be the main point of interest in the following.

As the behaviour of the passive structure is bound to produce disadvantageous effects, due to the simplification of the mechanical elements, the design ought to concentrate on the possibility to improve the behaviour of the system decisively by optimizing the controller parameters of the active struts and of the active steering rods (to the front wheels).

The program package *LINEX* (*LIN*ear *EX*pert) allows to optimize parametrized linear systems in view of formulated design objectives /Jäker et al. 91/. Important variables in the formulation of the optimization task are the lateral motion of the car body and its acceleration in vertical direction. In order to define appropriate objectives, a covariance analysis taking into account stochastic excitation of the linear system is performed to compute the covariance matrix of the output variables; the RMS values of the outputs in question are defined as variables that are to be minimized.

The optimization kernel of *LINEX* is *MOPO* /MOPO 89/. We have here a vector optimization method that, at each optimization step, generates a new parameter improving the objectives variables. In addition to computation of RMS values, calculation of frequency responses or pole areas can be performed by other tools. Generally, in addition to an optimization on the basis of the linear model, a computation of objectives variables from nonlinear simulation (*SIMEX*) is possible.

The above example represents just a small part of all possibilities to synthesize a mechatronic system. Fig. 5 and 6 illustrate the first, not yet optimal improvements with two time responses (linear simulation). Here only the left-hand side wheels of the vehicle, which makes a constant speed of 10 m/s, run across a manhole cover of 5 cm in height.

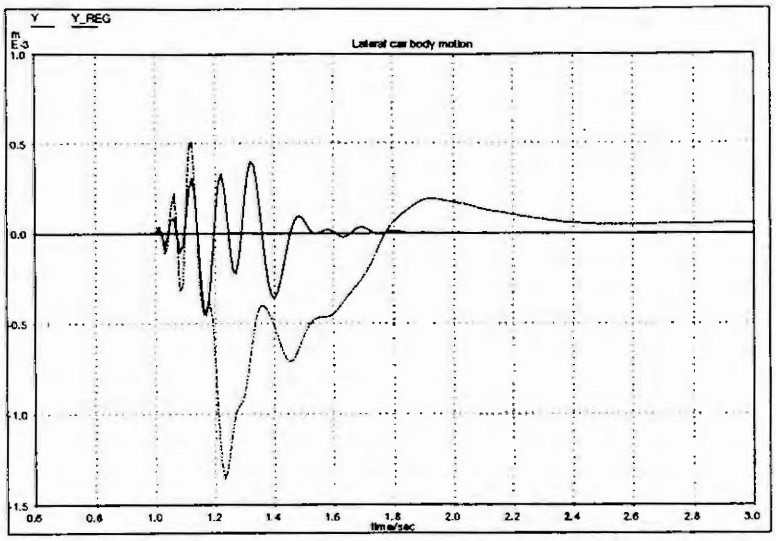


Fig. 5: Lateral car body motion (Y: uncontrolled, Y_REG: controlled)

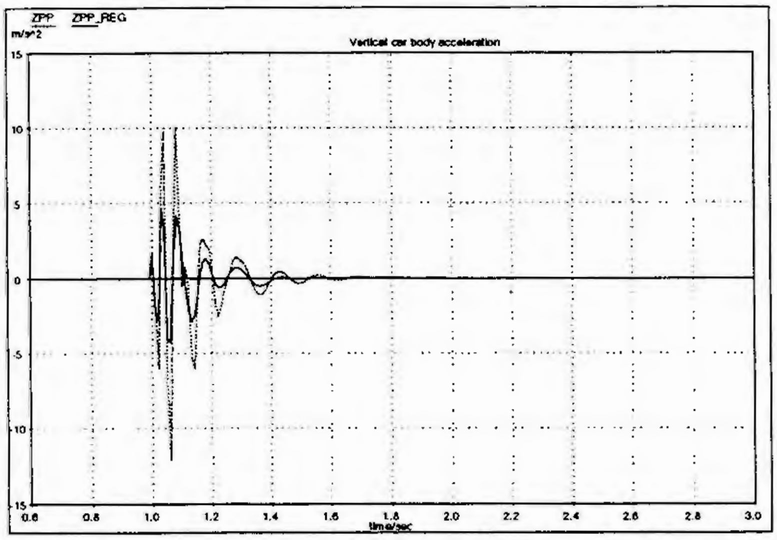


Fig. 6: Vertical car body acceleration (ZPP: uncontrolled, ZPP_REG: controlled)

Literaturverzeichnis

/Castiglioni 92/	Castiglioni, G.; Jäker, K.-P.; Lückel, J.; Rutz, R.: Active Vehicle Suspension with an Active Vibration Absorber, Proceedings of the International Symposium on Advanced Vehicle Control (AVEC' 92), Yokohama, Japan, 1992.
/DSL 92/	DSL-Sprachdokumentation, Universität-GH Paderborn, FB 10 - Automatisierungstechnik, 1992.
/Henker 93/	Henker, E.: Fahrwerktechnik, Vieweg Verlag, Braunschweig/Wiesbaden, 1993.
/Jäker 91/	Jäker, K.-P.; Klingebiel, P.; Lefarth, U.; Lückel, J.; Richert, J.; Rutz, R.: Tool Integration by way of a Computer-Aided Mechatronic Laboratory (*CAMeL*), CADCS '91, 5th IFAC/IMACS Symposium on Computer Aided Design in Control Systems, Swansea, 1991.
/Junker 94/	Junker, F.; Lückel, J.: A Systematics of Modelling Mechatronic Systems, 1. MATHMOD VIENNA, Vienna, 1994.
/Lückel 92/	Lückel, J.: The Concept of Mechatronic Function Modules (MFM), applied to Compound Active Suspension Systems, Research Issues in Automotive Integrated Chassis Control Systems, IAVSD, Herbertov, 1992.
/MOPO 89/	Kasper, R.; Lückel, J.; Jäker, K.-P.; Schröer, J.: MOPO - A CACE Tool for Multi-Input, Multi-Output Systems with the Aid of a New Vector Optimization Method, International Journal of Control, Vol. 55, 1990.
/Reimpell 88/	Reimpell, J.: Fahrwerktechnik: Radaufhängungen, Vogel-Buchverlag, Würzburg, 1988.

Physical Modelling of Mechatronic Systems According to Object-Oriented Principles

Martin Hahn

University of Paderborn
MLaP - Mechatronics Laboratory Paderborn
Prof. Dr.-Ing. Joachim Lückel
Pohlweg 55, D - 33098 Paderborn, Germany

1 Abstract

The modelling of mechatronic systems is a complex field in system design. The derivation of the mathematical representation is necessary for the calculation of the behaviour of the system. The process of the derivation can be divided into different levels with an encapsulated functionality. On the different levels, textual descriptions are used for system representation; they are logically interrelated by graph transformations. To manage the complexity resulting from many different part descriptions, object-orientation is employed.

2 Introduction

Modelling of mechatronic systems, due to their complexity, is a very time-consuming and error-bound process and requires systematic methods for work in a rapid and safe manner. Such a technique to manage complexity has long been known in computer science; it is named object-orientation. Object-orientation arranges different methods in a hierarchical way to manage complexity. The field of mechatronic systems yields a method to reduce complexity by means of physical models to encapsulate the mathematical models.

In this paper, an approach is presented which combines the advantages of physical modelling with those of object-oriented principles. First of all, the principles of object-orientation are explained, and it is shown how they can be used to accelerate the modelling as well as the software development process. Then the base classes and the class hierarchy of the mechatronic modelling approach are presented. For an internal representation, abstract data structures are used to reduce the complexity of the implementation. The topological and the hierarchical representation are explained; they are described by a special graph called "hierarchical graph". This leads to a network analysis of the system to calculate the static and the dynamic behaviour. This network model is used to generate the model equations of state in a modular hierarchical description.

3 Principles of Object-Orientation

The focus of object-orientation is the concept of objects and classes and the interrelation between them. In order to understand the nature of object-orientation, we need an idea of how an object can be modelled. The object-model defined in [BOOCH 91] has four major properties:

- abstraction
- encapsulation
- modularity
- hierarchy

Abstraction and encapsulation are complementary concepts. The important thing with abstraction is to reflect the way the user perceives the object. On the other hand, encapsulation means hiding all items that are implementation-dependent and of no importance to the user.

The terms modularization and hierarchy denote the structural decomposition of systems; they can be used to manage complexity. In this context, modularization denotes systems that are combined from a number of loosely coupled components.

The most important concept in object-orientation is the idea of hierarchy. Yet, there are different specifications of this term. Two of these specifications define a 'part of' hierarchy and a 'kind of' hierarchy. 'Part of' hierarchy means the aggregation of an object. In contrast, 'kind of' hierarchy denotes different hierarchical generalization-/specialization concepts. In general, the term 'kind of' hierarchy is associated with the term (single/multiple) inheritance.

The concepts of 'part of' hierarchy and 'kind of' hierarchy are used in object-orientation where the most important hierarchical concept is that of class hierarchy. Class hierarchy is a 'kind of' hierarchy in the sense of super-class-/subclass (generalization/specialization) relationships. Objects, however, are organized within a 'part of' hierarchy. They contain variables which in turn contain parts called instance variables ('part of' hierarchy). The cooperation of the instance variables is encapsulated within routines that are defined for the special class the object belongs to. These methods are the only means to communicate with the object. At least a relationship between classes and their objects is needed. To connect a class with the matching objects a relationship is needed that is called instance relationship.

The definition of the four major concepts "object", "class", "class hierarchy", and "method" are one possible approach to the object model from the point of view of object-oriented programming languages, especially (with regard to the terms employed) *SmallTalk* [GOLDBERG 76],[INGALLS 78].

4 Representation Levels of Mechatronic Systems and their Corresponding Object Levels

Mechatronic systems are compound systems with elements from different physical disciplines, e. g. mechanics, hydraulics, and electrical engineering, combined with controllers. In order to obtain expressive results, different views to the system have to be supported. These different views can be organized on three levels of abstraction; they build the representation levels of mechatronic systems (Fig. 1).

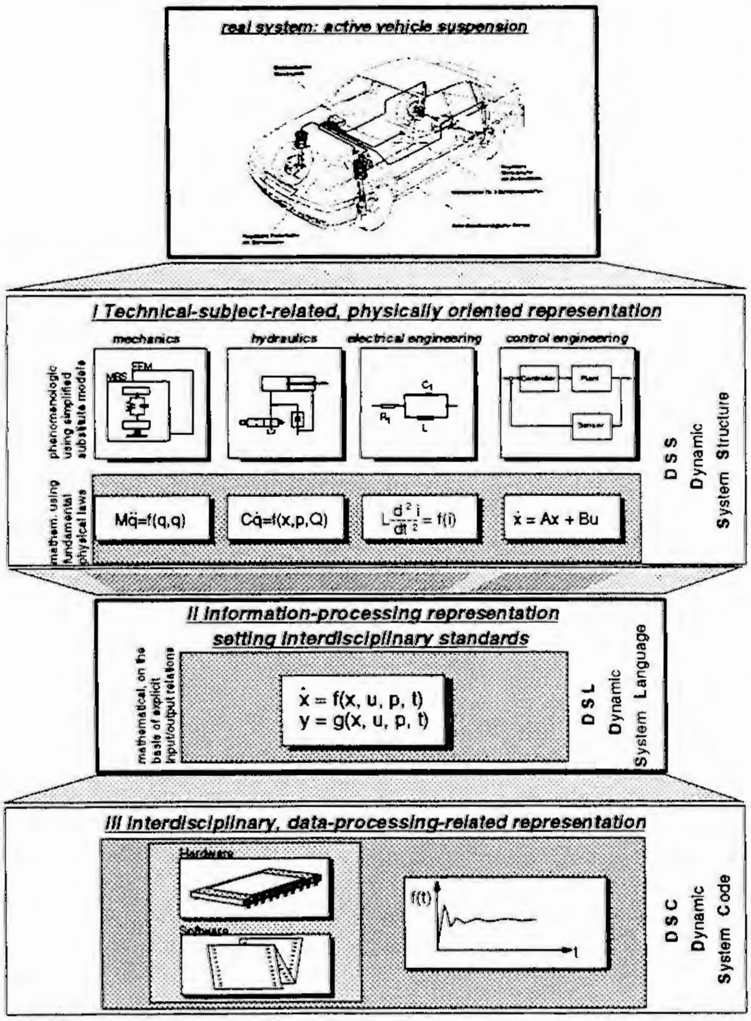

real system: active vehicle suspension
I Technical-subject-related, physically oriented representation
mechanics
hydraulics
electrical engineering
control engineering
phenomenologic using simplified substitute models
MBS
FEM
Controller
Plant
Sensor
mathem. using fundamental physical laws
Mq̈=f(q,q)
Cq̇=f(x,p,Q)
$L\frac{d^2i}{dt^2}=f(i)$
ẋ = Ax + Bu
D S S
Dynamic
System Structure
II Information-processing representation
setting interdisciplinary standards
mathematical, on the basis of explicit input/output relations
ẋ = f(x, u, p, t)
y = g(x, u, p, t)
D S L
Dynamic
System Language
III Interdisciplinary, data-processing-related representation
Hardware
Software
f(t)
t
D S C
Dynamic
System Code

Fig. 1
Representation levels of mechatronic systems

The top representation level is the subject-oriented one (level 1), comprising discipline-related descriptions of the different parts and their interconnections. In mechatronics, the subject matters are mechanics, hydraulics, electrical engineering, and control engineering. In each of these subjects, there is a special way of looking at the properties of parts and the coupling to subject-specific systems. It is necessary to integrate these specific descriptions into a method for the modelling of physical systems.

The middle representation level is the interdisciplinary information-technological representation (level 2). One possibility for an interdisciplinary representation to formulate systems- behaviour in the time domain is the state-space-representation. This mathematical description is used for the mathematical representation of the system.

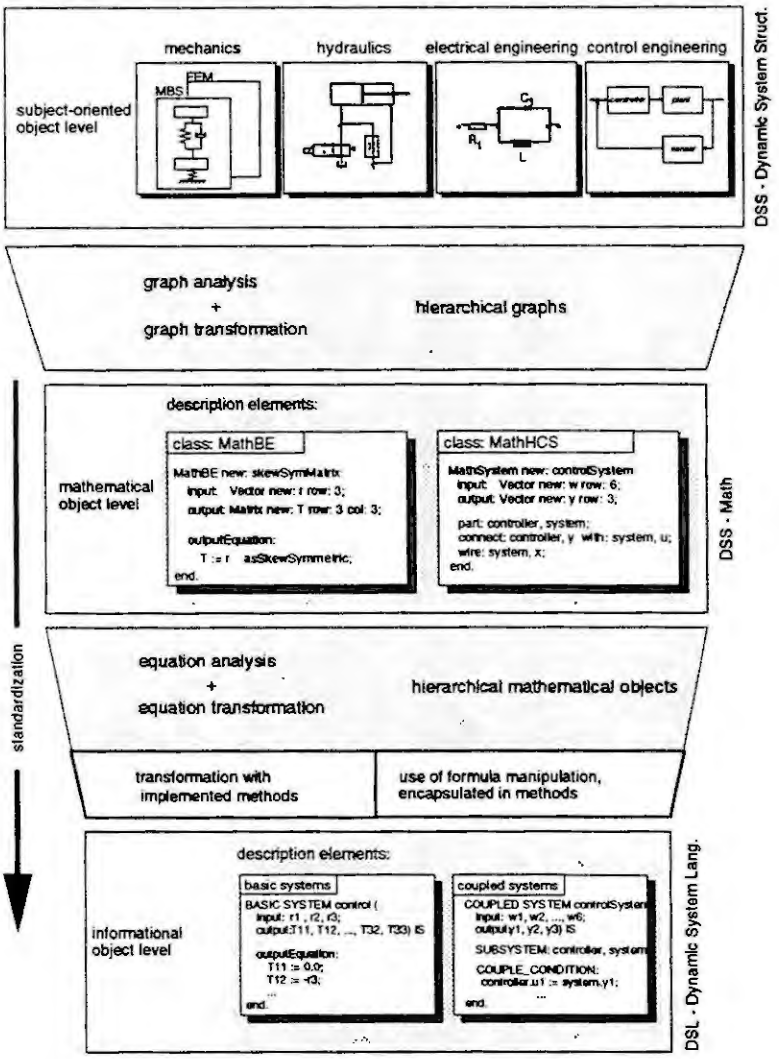
The third representation level is the data processing-related representation.

For each of the three representation levels, there is one specific textual description language. For the subject-oriented level (level 1), this is **Dynamic System Structure** (DSS).

Its textual representation consists of subject-specific components (called **basic elements**) on the one hand and joints (called **couple elements**) on the other. The third element of DSS is the description of aggregates, called **hierarchical systems**. These again may consist of basic elements, couple elements, and hierarchical systems (recursive formulation).

The interdisciplinary textual representation (level 2) is a description of hierarchically organized block diagrams based on scalar input/output relations. This is called **Dynamic System Language** (DSL) [SCHRÖER 91]. DSL comprises two main structural elements: **basic systems** and **coupled systems.** For both of them, inputs and outputs can be defined. The description of basic systems contains key words that define a system in state-space form. The coupled systems contain statements to declare input/output couplings and couplings to higher hierarchical levels (in order to provide visibility). Furthermore, a parametrization of the systems is possible.

The textual description on the hardware-oriented level (level 3) is done through the **Dynamic System Code** (DSC) and serves to provide efficient simulation and linearization code (DSC is not a topic of this paper; for further information, see [RICH 93]).



```
MathBE new: skewSymMatrix
   input  Vector new: r row: 3;
   output Matrix new: T row: 3 col: 3;

   outputEquation:
     T := r   asSkewSymmetric;
end.
```

```
MathSystem new: controlSystem
  input  Vector new: w row: 6;
  output Vector new: y row: 3;

  part: controller, system;
  connect: controller, y  with: system, u;
  wire: system, x;
end.
```

```
BASIC SYSTEM control (
   input: r1 , r2, r3;
   output:T11, T12, ..., T32, T33) IS

   outputEquation:
     T11 := 0.0;
     T12 := -r3;
     ...
end.
```

```
COUPLED SYSTEM controlSystem
   input: w1, w2, ..., w6;
   output:y1, y2, y3) IS

   SUBSYSTEM: controller, system

   COUPLE_CONDITION:
    controller.u1 := system.y1;
            ...
end.
```

Fig. 2
Object levels of mechatronic systems

This conceptual way (Fig. 1) leads from the subject-oriented to a standardized mathematical description in the DSL language. Because of this standardization (from the subject-specific to the state-space description) it's managed a representation which is suitable for the analysis and design of mechatronic systems.

In general, computers are not organized in an object-oriented way. Therefore, it is necessary to divide the transformation process into object-oriented and function-oriented parts. In the approach presented, the division is made on the intermediate level, the DSL-Description. The syntactical elements ofDSL are encapsulated in objects which connect the object-oriented to the function-oriented representation of DSL. On the top hierarchical level (Fig. 2), the subject-oriented description in DSS is fully object-oriented. It is useful to have as the next step a logical division of the transformation process, in order to make a clear distinction between the different system analyses. This transformation process can be subdivided into two major steps: the first step is the topological analysis of the system. It is necessary to identify the information flow in a mechatronic model and leads to a purely mathematical-oriented description of the system. The second step is the transformation of the vector-oriented description to DSL. In order to decouple these two steps, an intermediate level is introduced, called DSS-Math. This fully object-oriented textual description is based on a vectorial description of the state-space form.

All its elements can be modelled in a hierarchical and modular way. Since the equations in DSS-Math have an object-oriented form and since the syntax is closely related to that of the implementation language (*Smalltalk*), the equations can be analyzed syntactically and evaluated symbolically by the interpreter of the system.

The classes of the DSS-Math systems provide methods for transforming the *MathBE*[1] and *MathHCS* descriptions into DSL. If there are transformations of symbollically equations (e. g. the solution of large linear equation systems or the inversion of a matrix), it can be useful to integrate a formula manipulation program. In the methods which are represented by keywords in DSS-Math, the information is implemented whether a formula manipulation is

[1] In the following, class names are in italics

necessary or not. By analyzing the syntactical form of the DSS-Math elements defined (and perhaps altered) by the user and also the symbolical transformation of the vectorial into scalar equations, an object-oriented representation of DSL is obtained.

5 The Base Classes of Hierarchical-Modular Models

From the point of view of object-oriented design, a discipline concerned with structuring complex systems, the classes, the objects and their interrelations have to be organized within a class hierarchy. For a proper identification of the required classes, a problem analysis will have to be performed to find out the common design characteristics. The characteristic the three levels have in common, as shown in Fig. 2, is the organization of the systems as modular and hierarchical structures.

This characteristic, is employed to obtain the base classes of the design. A hierarchical system can be defined inductively [ZEIG 90, p. 29]. The elements of a hierarchical system are either basic elements or hierarchical elements. To define the relationship between two elements, couple elements are needed. This leads to the elementary class hierarchy, shown in Fig. 3. It can be shown that all three object levels of Fig. 2 and their building blocks can be implemented as specializations of these three classes.

```
Object
├BasicElement
│ ├<subclasses>
│ └...
├CoupleElement
│ ├<subclasses>
│ └...
└HierarchicalSystem
  ├<subclasses>
  └...
```

Fig. 3
Base class hierarchy of
modular-hierarchical systems

Every description of hierarchical systems needs a definition of the granular elements the description is based on. In this approach, granular elements have to be modelled as specializations (subclasses) of the class *BasicElement*.

The interconnection structure of a hierarchical system can also be classified. In the base-class hierarchy, interconnections between systems have to be modelled as specializations of the class *CoupleElement*. Every object level in Fig. 2 is independent of the other levels. They are not intended to be combined on the same abstraction level, so they are independent specializations of the class *HierarchicalSystem*. The transformation of the different object levels is performed by methods.

For the internal representation of the modular-hierarchical structure a special graph is used. Graphs consist of vertices and arcs and are a set-theoretical model of structures. The abstract elements vertex and arc are be specialized for the representation of modular-hierarchical systems, independent of the subject-specific application. This specific graph is called hierarchical graph.

In modular-hierarchical systems, there are two kinds of vertices: leaf vertices and inner vertices. The leaf vertices contain information on the parts of the system, whereas the inner vertices contain information about the hierarchical and the connection structure of the system on its respective hierarchical level.

The connection structure (the arcs) of modular-hierarchical systems can be divided into three kinds of arcs: couple-element arcs, wire arcs and link arcs. In this context, couple-elements represent subject-specific couplings ("topological coupling") and are only valid to connect systems on the same hierarchical level (they have the same "father"). The other kind of coupling is the wire coupling. Wires define if substructures of BasicElements, called Ports, are visible on a higher hierarchical level. The father/son relationship is represented by link arcs ("hierarchical coupling"). For this special kind of graphs, many methods from graph and tree theory can be implemented and specialized, independend from the specific use of the structure (abstract data types).

For every object level of Fig. 2, there is a textual description of the system. The "language" used for the formulation of the topological and hierarchical structure of the system on the different object levels (Fig. 2) is called DSS (Dynamic System Structure). In addition to the structure, the properties of the parts are formulated as atomic units of the system, and joints are modelled as couple elements. The functionality of hierarchical systems is completely encapsulated in the class *HierarchicalSystem*. Only additional properties have to be modelled in specializations of the class *HierarchicalSystem*. In the next two sections, aspects of specializations on the upper two object levels of Fig. 2 will be presented.

6 Multibody Systems: A Specialization of the Base Classes on the Subject-Oriented Level

In this chapter, the specialization of the base classes on the subject-oriented level is demonstrated on multibody systems. The granular elements of multibody systems are rigid bodies, springs, dampers, and bodies with predefined motions. These elements are interconnected by joints that define the relative motion between two granular elements. The granular elements are specialized elements of the class *BasicElement* in the sense of the base classes defined above, the joints, in the sense of the class *CoupleElement*. This leads to the extended class hierarchy shown in Fig. 4.

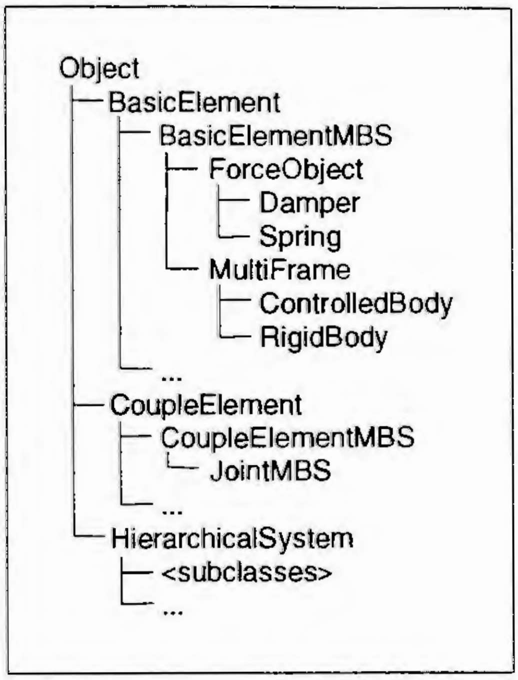


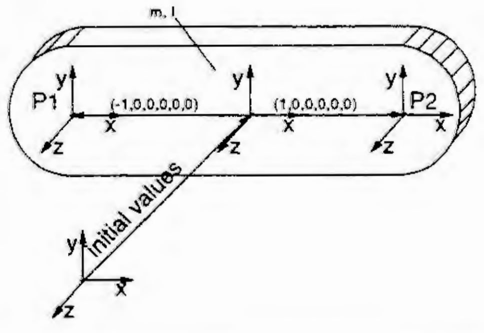
Fig. 4
Extended class hierarchy for multibody systems

All *BasicElement* subclasses inherit the properties of this superclass. The class *MultiFrame* is an abstract class and will never be instantiated. *MultiFrame* encapsulates the common properties of rigid bodies (class *RigidBody*) and of bodies with predefined motion (class *ControlledBody*). The features both classes have in common consist of a reference coordinate system (reference frame) and a set of coordinate systems relative to this reference frame. The difference between the two classes is that the class *RigidBody* describes models with a dynamical behaviour that depends on the structure of the system. Objects of the class *ControlledBody* describe models with a predefined motion within the time domain. This specialization of the base classes leads to additional instance variables. For the class *RigidBody*, the additional instance variables "mass" and "inertia tensor" are needed and for the class *ControlledBody* a vectorial function is necessary which defines the motion of the object in the time domain.

A spring in a multibody system is a part which transforms translatorial or rotatorial differences between two attachment points into forces (analogous for dampers in the (angular-)velocity). Both elements need a function which represents the spring (damper) constant or function according to the respective displacement.

After the parts, the connections that are admissible in a multibody system are defined. Connections in multibody systems are joints; they couple two bodies to one another (e. g. ball and socket joints). The main property of a joint is the information on the degrees of freedom admitted.

All elements can be represented in a textual way by the accompanying DSS part description. A description of a part of the class *RigidBody* is shown in Fig. 5 and a ball and socket joint in Fig. 6. A hierarchically assembled multibody system can be built up with the syntactical possibilities of DSS. Neither further information nor new syntactical elements added to the base classes are needed. An example is shown in Fig. 7. All other parts and systems can be represented in an analogous way.



DSS Description:

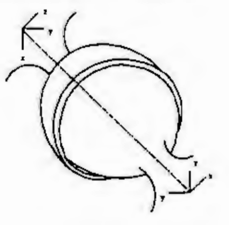
```
RigidBody new: controlArm.
    port: p1 on: MechanicPort value: (-1,0,0,0,0,0);
    port: p2 on: MechanicPort value: (1,0,0,0,0,0);

    mass:           10.0;
    inertia:        (1.0,2.0,4.0,0.0,0.0,0.0);
    {position:      (-1,0,0);}
    {orientation:   (0,0,0);}
    {velocity:      (0,0,0);}
    {angVelocity:   (0,0,0);}

end.
```

Fig. 5
DSS part description of a rigid body

To transform a model in DSS representation into a mathematical representation an analysis of the system graph is necessary. If the description of the system is correct, its internal representation (a hierarchical graph) is performed. This graph, representing the structure of the system, can be analyzed and transformed into the state-space representation.

In the case of multibody systems, different methods of transforming the structure into the mathematical description are possible. To keep the modular structure of the system within the mathematical description, two formulation methods can be used with multibody systems.

The first one, a method for the approximate description of rigid joints in multibody systems, is the method of dynamic suspensions [HENT 90]. To maintain the constraint equations in the locked directions, force laws can be formulated in order to approximately suppress these degrees of freedom.

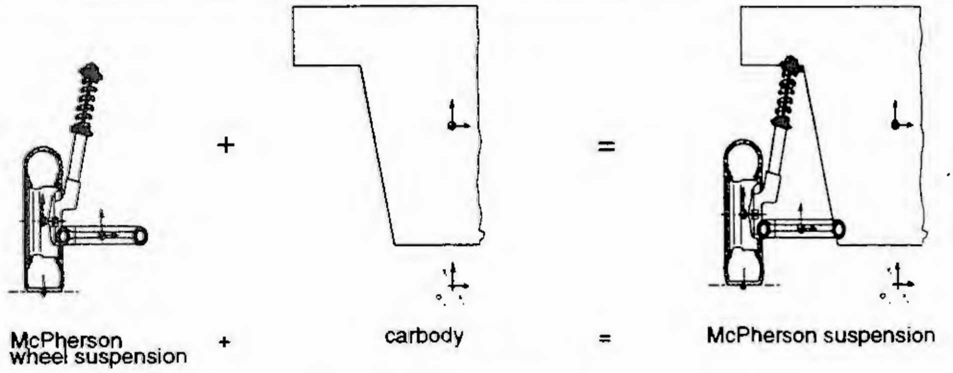


```
DSS description:

JointMBS new: ballAndSocket.
    degreesOfFreedom:           (PhiX, PhiY, PhiZ);
    {degreesOfFreedomValue:     (0,0,0);}
    {stiffnessOfDynamicJoint:   (1e6, 1e6, 1e6, 0, 0, 0);}
    {dampingOfDynamicJoint:     (1e3, 1e3, 1e3, 0, 0, 0);}

end.
```

Fig. 6
DSS couple element description of a ball-and-socket joint

This method holds for many fields of work, e. g. the modelling of car suspension systems, where elasticities and dampings can also be found in the joints of the real system. Due to the decoupling of the masses in the system, every mass and every joint can be formulated in the same way, and it is very easy to couple the components, because there are only couplings between neighbouring masses, independent of the system structure (tree or loop systems).

The second one is a recursive formulation of the system equations [JUNKER 93]. Base elements for the recursive calculation are the elements with predefined motion. In this approach, every joint is assigned to a body relative to its position in the structure (predecessor/successor relationships). This leads to an unambiguous assignment of the joint and therefore to an unambiguous calculation order.



```
DSS description:    HierarchicalSystem new: mcPhersonSuspension.
                        port: carBody3, wheel3 on: MechanicPort;

                        part:  carBody, mcPhersonWheelSuspension;
                        joint: carBodyMcPher, carBodyControlArm;

                        connect: carBodyMcPher       from: carBody, atp1   to: mcPhersonWheelSuspension, atp1;
                        connect: carBodyControlArm   from: carBody, atp2   to: mcPhersonWheelSuspension, atp2;

                        wire: carBody3    on: carBody, atp3;
                        wire: wheel3      on: mcPhersonWheelSuspension, atp3;

                    end.
```

Fig. 7
DSS description of a hierarchical system

For both methods the internal representation (as a hierarchical graph) and the external representation (as a DSS description) are identical; it is only the transformation method that varies. The topological and the hierarchical assembly are kept in the mathematical description.

7 DSS-Math: A Specialization on the Intermediate Level

```
MathBE new: unconstraintMass.
   parameter: mass                                  : ScalarVal;
              gravity                               : VectorVal(3);
              inertia                               : MatrixVal(3,3);
   input:     force, moment                         : VectorVal(3);
   output:    mcPos, mcVel, mcOri, mcOme            : VectorVal(3);
              mcQua                                 : VectorVal(4);

   state:     position, velocity, omega      : VectorVal(3);
              quater                         : VectorVal(4);

   auxiliar:  rightSide                      : VectorVal(3);

   auxiliarEquation:
      rightSide :=     ((inertia * omega crossProduct: omega)
                     + (quater quatAsTransfMatrix * moment));

   stateEquation:
      position'      := velocity;
      velocity'      := force / mass + gravity;
      quater'        := (quater quatAsKinemMatrix * omega) / 2.0;
      omega'         := inertia adjunctive * rightSide / inertia det;

   outputEquation:
      mcPos   :=     position;
      mcVel   :=     velocity;
      mcOri   :=     quater quatAsTransfMatrix;
      mcOme   :=     omega;
      mcQua   :=     quater;
end.
```

Fig. 8
An instance of the class *MathBE*

This chapter deals with the specialization of the base classes for the intermediate level in Fig. 2. The basic element in this description is a mathematical description which is based on the state-space form. These basic elements are called *MathBE* (**Math**ematical **B**asic **E**lements). For them, there are two types of ports: input ports and output ports. The inputs and outputs can be formulated in a vectorial form. For the description of the mathematical input/output behaviour of a *MathBE*, it is possible to declare internal variables and auxiliar calculations. All equations are formulated in an object-oriented way. This means a method is applied to a mathematical object, e.g. the inversion is applied to a matrix[1]. All methods which can be used in *MathBE* are implemented in two forms, to obtain different functionalities in different phases of the work with the elements:

syntactical check
symbolical evaluation

```
BASIC SYSTEM TYPE unconstraintMass(
   PARAMETER   :  mass,
                  gravity1,     gravity2,     gravity3,
                  inertia11,    inertia12,    inertia13,
                  inertia21,    inertia22,    inertia23,
                  inertia31,    inertia32,    inertia33;
   INPUT       :  force1,       force2,       force3,
                  moment1,      moment2,      moment3;
   OUTPUT      :  mcPos1,       mcPos2,       mcPos3,
                  mcVel1,       mcVel2,       mcVel3,
                  mcOri11,      mcOri12,      mcOri13,
                  mcOri21,      mcOri22,      mcOri23,
                  mcOri31,      mcOri32,      mcOri33,
                  mcOme1,       mcOme2,       mcOme3,
                  mcQua1,       mcQua2,       mcQua3, mcQua4) IS

   STATE       :  position1,    position2,    position3,
                  velocity1,    velocity2,    velocity3,
                  omega1,       omega2,       omega3,
                  quater1,      quater2,      quater3,       quater4;

   AUXILIAR : rightSide1, rightSide2, rightSide3;

   rightSide1 :=
      ((  ((inertia21*omega1 + inertia22*omega2 + inertia23*omega3) * omega3)
      -   ((inertia31*omega1 + inertia32*omega2 + inertia33*omega3) * omega2))
      +      (   quater1*quater1 - quater2*quater2
      -          quater3*quater3 + quater4*quater4     * moment1
      + 2.0*(    quater1*quater2 + quater3*quater4)    * moment2
      + 2.0*(    quater1*quater3 - quater2*quater4)    * moment3));
      ...
```

Fig. 9
Part of the *MathBE* (Fig. 8) in DSL code representation

In the phase of syntactical check, it is guaranteed that there are only formulations which are syntactically correct with regard to the combination of numerical objects or the sizes of matrices and vectors.
The other phase, that of symbolical evaluation, is used to generate the equations in a scalar format. This is necessary to obtain an executable code for standard simulation environments. In this approach the code is generated in DSL (Dynamic System Language) format. As an example of an element in the *MathBE* syntax, the calculation of the kinematic behaviour (position, orientation, and velocity) of a free rigid body (dependent on the initial states) is shown in Fig. 8 and a part of the scalar complement in Fig. 9.
The systematics at the basis of the transformation of the vectorial equations into the scalar format are encapsulated by methods (function calls). The latter "know" the objects involved in the method call; therefore, the method can be made to display their representation in a formula manipulation.

8 Results

The paper presents an approach which divides into logical parts the derivation process of the equations of motion and the transformation into state-space form. These parts and the interconnections between them are organized in an object-oriented way. The object-oriented organization makes possible a combination of different subject-oriented descriptions and allows a common protocol for equation generation.

The common characteristics of the different representation forms are used to obtain a common base-class hierarchy, the topological and the hierarchical structure of the different system descriptions. The consistent separation of different functionalities, such as the analysis of the structure and the transformation of the vectorial equations into scalar ones, allows a well-encapsulated integration of a formula manipulation.

The three levels support the subject-oriented view at the system by different groups of users, and the class hierarchy is the basis of the integration of other disciplines. The classification (superclass/subclass relationships) of the systems is an important requirement for the treatment of complex mechatronic systems.

9 References

[BOOCH 91] Booch, G.: Object-Oriented Design with Applications, Menlo Park, CA, 1991.

[ELMQVIST 78] Elmquist, H.: A Structured Model Language for Large Continuous Systems, Ph. D. thesis in Automatic Control, Lund, Sweden, 1978.

[GOLDBERG 76] Goldberg, A.; Kay, A.;Eds.: Smalltalk-72 Instruction Manual, Xerox PARC Technical Report SSL-76-6, 1976.

[HENT 90] Hentschel, M.; Engelke, A.: Decomposition of the Dynamic Equations of a Multibody System According to its Physical Structure for Parallel Computation on a Transputer Network, Proceedings of the First International Conference on Parallel Processing for Computational Mechanics, Southampton, September 4-6, 1990.

[INGALLS 78] Ingalls, D.H.H: The Smalltalk-76 Programming System: Design and Implementation, Conference Record, 5. Annual ACM Symposium on Principles of Programming Languages, 1978.

[JUNKER 93] Junker, F.; Lückel, J.: A Systematics of Modelling Mechatronic Systems, 1. MATHMOD VIENNA, Vienna, February 2-4, 1994.

[RICH 93] Richert, J.; Homburg, C.; Engelke, A.: DSC (Dynamic System Code) - Eine abstrakte blockorientierte Beschreibungssprache für mechatronische Systeme, 8. ASIM-Symposium, Berlin, 1993.

[SCHRÖER 91] Schröer, J.: A Short Description of a Model Compiler/Interpreter for Supporting Simulation and Optimization of Nonlinear and Linearized Dynamic Systems, in: CADCS 91, 5th IFAC/IMACS Symposium on Computer-Aided Design in Control Systems, Swansea, Wales, July 15-17, 1991.

[ZEIG 90] Zeigler, B. P.: Object-Oriented Simulation with Hierarchical, Modular Models: Intelligent Agents and Endomorphic Systems, Boston/San Diego, 1990.

a) In the syntax of the description language, the inversion of a matrix is described as "T_inv := T inverse;", with T and T_inv as matrices and inverse as a method call.

The DAE-Index in Electric Circuit Simulation

M. GÜNTHER*

Technische Universität München, Mathematisches Institut, D-80290 München, Germany

U. FELDMANN

SIEMENS AG, ZFE BT SE 43, Otto-Hahn-Ring 6, D-81739 München, Germany

The index of differential-algebraic equations is one measure for the numerical problems in electric circuit simulation. The index depends on the mathematical model in different ways: setup of equations, classical or charge-oriented formulation and modelling of basic elements and semiconductor devices. The modelling of MOS transistor circuits will show this in more detail.

AMS Subject Classification: 65L05
Key words: Electric circuit simulation, mathematical modelling, CAD, index of differential-algebraic equations, MOS transistor models

1 Introduction

Numerical simulation is an important tool in the design of electric circuits today. The underlying mathematical model has strong implications on the numerical behaviour during simulation. We will focus on the connections between models for time domain simulation and numerical properties of circuit simulation problems.

CAD (*C*omputer *A*ided *D*esign) oriented modelling of circuits generally leads to systems of differential-algebraic equations (DAEs). The DAE-index is a measure for the numerical problems to be expected when solving these equations. A short review of the different notions of index is given in the next chapter. In the following the basic modelling principles and the most commonly used modelling techniques are described. As an example a simple LC oscillator circuit will show their effects on the index.

The index depends also on various other items: the type of the circuit, the use of classical or charge-oriented formulation or the modelling of basic elements and semiconductor devices. The modelling of MOSFETs (*M*etal *O*xide *S*emiconductor *F*ield *E*ffect *T*ransistors) as an example will show this in more detail. Finally some remarks on regularization are given.

2 The index of differential-algebraic equations

A formulation of network equations in state space form with a minimal set of variables leads to a system of ordinary differential equations (ODEs). However, the automatic generation of the equations by CAD-methods involves a redundant set of variables and thus leads to a system of differential-algebraic equations (DAEs) of the general form

$$F(x, x', t) = 0, \tag{1}$$

which is generally nonlinear, stiff, sparse and sometimes weakly coupled.

Differential-algebraic equations are not ODEs, as stated by L. Petzold in her famous article (cf. [14]):

- Initial values have to be *consistent*, i. e. both $x(t_0)$ and $x'(t_0)$ have to fulfill (1).
- Some DAEs can be solved well by standard ODE-solvers. Other however require extensive modifications to the error estimates or forbid step size changes.

*The author is supported by FORTWIHR: Bavarian Consortium for High-Performance Scientific Computing

The index gives a classification of DAEs with respect to their numerical problems. It is – roughly spoken – a measure for the distance between the DAE and an ODE.

The notion of index has its origin in linear problems with constant coefficients

$$A \cdot x' + B \cdot x = f(t). \tag{2}$$

If $B + \lambda \cdot A$ is a regular matrix pencil (see [10]), regular matrices P and Q exist such that

$$PAQ = \begin{pmatrix} I & 0 \\ 0 & N \end{pmatrix} \quad , \quad PBQ = \begin{pmatrix} C & 0 \\ 0 & I \end{pmatrix} \tag{3}$$

where N is block diagonal with nilpotent blocks N_i as entries. Each N_i is of the form

$$N_i = \begin{pmatrix} 0 & 1 & & 0 \\ & \ddots & \ddots & \\ & & 0 & 1 \\ 0 & & & 0 \end{pmatrix} \quad , \quad \text{of dimension } m_i. \tag{4}$$

The *nilpotency* of $B + \lambda \cdot A$ is defined as the maximum of the dimensions m_i (cf. [6]).

We will focus on the so-called *differential index*, a generalization of the *nilpotency* to nonlinear problems (cf. [5], [10]):

The **differential index** of (1) is the minimum integer m such that the system (1) and

$$\begin{aligned} \frac{d}{dt}F(x, x', t) &= 0 \\ &\vdots \\ \frac{d^m}{dt^m}F(x, x', t) &= 0 \end{aligned}$$

can be solved for $x' = x'(x)$.

Another notion of index is given by Hairer, Lubich and Roche in [10]: The *perturbation index* is a measure for the sensitivity of the solutions to perturbations in the equations. A survey of the main notions of index is given in chapter 1 of [10]. New index concepts are investigated in recent works of Campbell and Gear (cf. [1]).

Remark: An ODE as a special case of (1) has index 0 – in all different notions.

In circuit simulation, the index depends on the applied modelling techniqne. This will be discussed in the following chapters.

3 Basic principles of mathematical modelling in circuit simulation

Electronic circuits are modelled by assembling idealized basic linear elements like resistors, capacitors,... and nonlinear ones like semiconductor devices. The electric behaviour of a circuit in the time domain is described by the waveforms of the potentials at each node between two or more adjacent elements and of the branch currents.

The circuit is characterized by the type of the elements, their electrical value (resistance, capacitance,...), and by the network topology. The nodes between elements are assumed to be electrically ideal. Therefore two kinds of relations are needed:

1. *Characteristic element equations*
   These equations relate the voltage drop between the nodes of an element to the current (cf. [12]). Nonlinear models for semiconductor devices like transistors are used to describe physical reality to a great extent. The main modelling principles are the following:

- A model is composed of the five basic elements resistor, capacitor, inductor, voltage source and current source.
- The nonlinear controlled voltage or current sources and resistors describe the static (i.e. time independent) behaviour.
- The nonlinear controlled capacitors or inductors describe the dynamic behaviour.

2. *Kirchhoff's laws*
   The voltages and branch currents of the circuit depend on the topology of the network and are determined by Kirchhoff's laws (cf. [2]):
   - *Kirchhoff's current law* (KCL)
     The algebraic sum of currents traversing each cutset of the network must be equal to zero at every instant of time.
   - *Kirchhoff's voltage law* (KVL)
     The algebraic sum of voltages around each loop of the network must be equal to zero at every instant of time.

Conservation of charge over time is another important fact, which holds true for every circuit (cf. [3], [16]). To guarantee this, charge-oriented modelling may be used (cf. [7]).

4 Automatic generation of network equations by CAD-methods

In network theory many different methods for setting up the equations have been discussed. In the following, we will describe two of them, which are universal and well suited for computer implementation:

1. *Sparse Tableau Approach* (STA, cf. [9])
   The vector x of unknowns contains all node potentials u, branch voltages U and branch currents I. Applying KCL to every node and KVL to every branch, one gets together with the characteristic equation for every element the sparse tableau formulation

$$\begin{aligned} A \cdot I &= 0 && \text{(KCL)} \\ U - A^t \cdot u &= 0 && \text{(KVL)} \\ \Phi(I, U, \dot{I}, \dot{U}) &= 0 && \text{(characteristic element equations)} \end{aligned} \tag{5}$$

   A denotes the incidence matrix, which describes the circuit topology. The structure of (5) implies that its index is greater or equal 1. Lötstedt and Petzold [13] have shown that the index is greater or equal 2 if there is a loop in the network with branches containing only voltage sources and capacitors (see also next chapter). STA is the most obvious scheme, but it generates a very large number of mainly short equations even for small circuits.

2. *Modified Nodal Analysis* (MNA, cf. [11])
   The vector x of unknowns contains all node potentials u and the currents I through all voltage-controlling elements (inductors and voltage sources). KCL is applied to every node. The element equations - as far as they define currents as a function of branch voltages - and the ones due to KVL are inserted directly. The characteristic equations for all voltage controlling elements are added explicitly, because they define voltages rather than currents. MNA shares the universality with STA, but has the advantage of a smaller number of unknowns and equations. It is therefore most commonly used in circuit simulation programs (cf. [4]).

   The application of MNA generally leads to a DAE of the form

$$C(x) \cdot \dot{x} + J(x) - S(t) = 0 \tag{6}$$

   with the "capacitance" matrix $C(x)$ describing the dynamic elements, $J(x)$ the static ones and $S(t)$ the independent sources. In the case of $C(x)$ singular, one gets an index greater than 0. The index is 0, if the circuit does not contain voltage sources and has capacitors at each node.

An example will show the different effects of STA and MNA on the index: Consider the linear LC oscillator circuit shown in fig. 1, containing only a capacitor with capacitance C and an inductor with inductance L. The potential of the only node (except ground) is u_1.

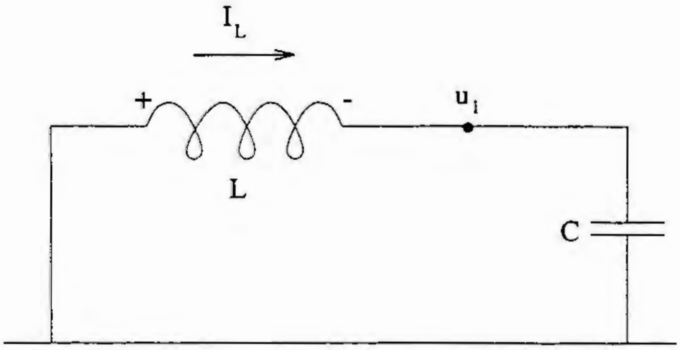


Fig. 1: Linear LC oscillator circuit

- Formulation in STA

$$
\begin{aligned}
\text{vector of unknowns:} \qquad x &= (I_L, I_C, U_L, U_C, u_1)^t \\
-I_L + I_C &= 0 && \text{(KCL)} \\
U_L + u_1 &= 0 && \text{(KVL 1)} \\
U_C - u_1 &= 0 && \text{(KVL 2)} \\
C \cdot \dot{U}_C - I_C &= 0 && \text{(char. equation for capacitor)} \\
L \cdot \dot{I}_L - U_L &= 0 && \text{(char. equation for inductor)}
\end{aligned}
$$

or written in matrix form:

$$
\begin{pmatrix} C & 0 & 0 & 0 & 0 \\ 0 & L & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} \dot{U}_C \\ \dot{I}_L \\ \dot{I}_C \\ \dot{U}_L \\ \dot{u}_1 \end{pmatrix} + \begin{pmatrix} 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} U_C \\ I_L \\ I_C \\ U_L \\ u_1 \end{pmatrix} = 0 \tag{7}
$$

One sees immediately that the differential index is not 0. One has

$$
\begin{pmatrix} \dot{U}_C \\ \dot{I}_L \end{pmatrix} = \begin{pmatrix} \frac{I_C}{C} \\ \frac{U_L}{L} \end{pmatrix} \qquad \text{and} \qquad \begin{pmatrix} I_C \\ U_L \\ u_1 \end{pmatrix} = \begin{pmatrix} I_L \\ -U_C \\ U_C \end{pmatrix} \tag{8}
$$

Differentiation of the right part of (8) yields

$$
\begin{pmatrix} \dot{I}_C \\ \dot{U}_L \\ \dot{u}_1 \end{pmatrix} = \begin{pmatrix} \dot{I}_L \\ -\dot{U}_C \\ \dot{U}_C \end{pmatrix} \overset{(8)}{=} \begin{pmatrix} \frac{U_L}{L} \\ -\frac{I_C}{C} \\ \frac{I_C}{C} \end{pmatrix} \tag{9}
$$

The differential index of (7) is thus 1.

- Formulation in MNA

$$
\begin{aligned}
\text{vector of unknowns:} \qquad x &= (u_1, I_L)^t \\
C \cdot \dot{u}_1 - I_C &= 0 && \text{(KCL)} \\
L \cdot \dot{I}_L + u_1 &= 0 && \text{(char. equation for inductor)}
\end{aligned} \tag{10}
$$

(10) has differential index 0.

5 MOSFET modelling and index

In this chapter we will discuss, how the level of accuracy for modelling circuit elements may affect the DAE index. We restrict ourselves to different models for MOSFETs, which play an important role in integrated circuit design.

An MOS transistor has four pins: Drain D, Gate G, Source S and Bulk B, see fig. 2. Current flow is from drain to source, if and only if the controlling voltage drop U_{GS} between gate and source is larger than a technology dependent threshold voltage v_{TH}.

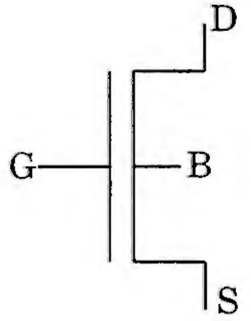


Fig. 2: MOSFET symbol

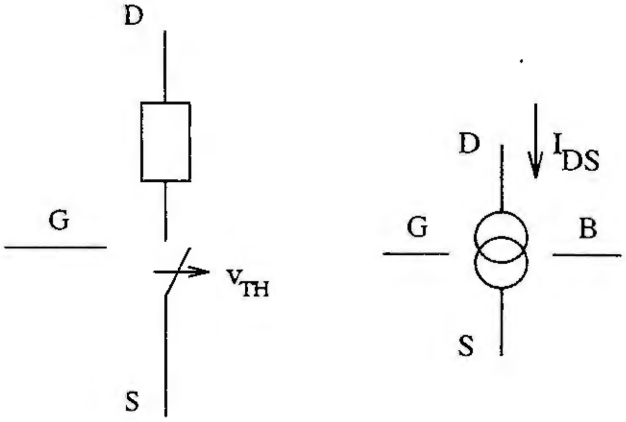


Fig. 3: Simple transistor models

So the simplest transistor model is an ideal switch in series with a resistor, which is controlled by the gate-source voltage U_{GS} (cf. fig. 3, left). This model is however seldom used in circuit simulation, because the discontinuity at the switching point requires extra efforts in the numerical algorithms, e.g. for the time step control.

A more popular MOS model is a nonlinear controlled current source, which describes the static current characteristics as a function of the applied gate-source voltage U_{GS} and drain-source voltage U_{DS}, see fig. 3, right. See Kampowsky et al. [12] for a description of the used element symbols.

First order equations for the current were derived by Shichman and Hodges ([17]), see fig. 4:

$$I_{DS} = \text{const} \cdot \frac{W}{L} \cdot \begin{cases} 0 & U_{GS} - v_{TH} \leq 0 \\ \frac{1}{2} \cdot (U_{GS} - v_{TH})^2 & U_{DS} > U_{GS} - v_{TH} > 0 \\ ((U_{GS} - v_{TH}) \cdot U_{DS} - \frac{1}{2} \cdot U_{DS}^2) & U_{DS} \leq U_{GS} - v_{TH} > 0 \end{cases} \tag{11}$$

The threshold voltage is dependent of the bulk-source voltage U_{BS}:

$$v_{TH} = v_{T0} + \gamma \cdot \left(\sqrt{\varphi - U_{BS}} - \sqrt{\varphi} \right) \tag{12}$$

W and L are the width resp. length of the transistor, and v_{T0}, γ, φ and *const* are technological parameters which can be computed from measurements by curve fitting (see fig. 4).

Note that both models of fig. 3 are purely static and generally do not affect the index of the problem.

Extensions of this model apply more accurate equations for the current and take the parasitics into account, which are associated with an integrated MOS transistor (see fig. 5 and 6): series resistors at drain and source, a leakage resistor, overlap capacitances between gate and drain resp. source, and finally a model for the pn-junction between bulk and drain resp. source. The latter consists of a static diode with exponential current characteristic and of a nonlinear capacitor.

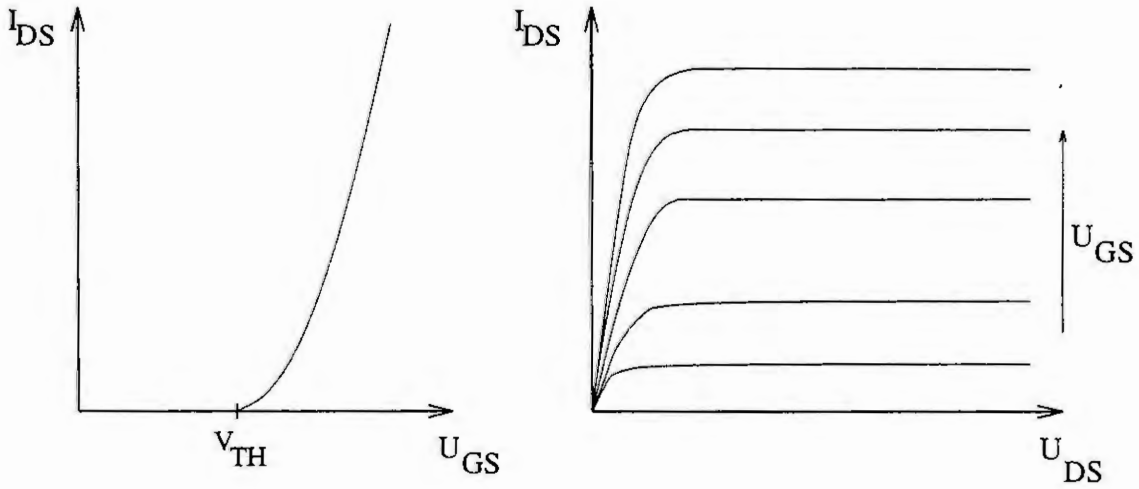


Fig. 4: Characteristics of an n-channel MOSFET with $v_{TH} > 0$

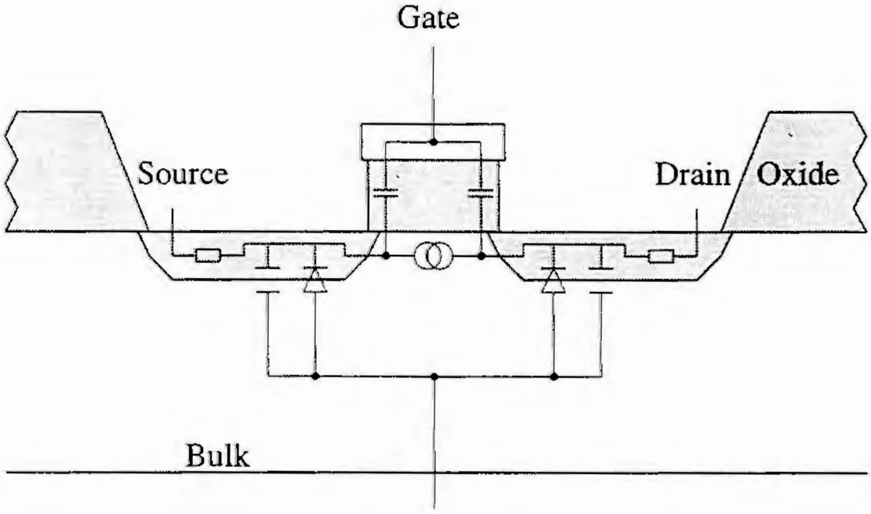


Fig. 5: Cross section of a MOSFET

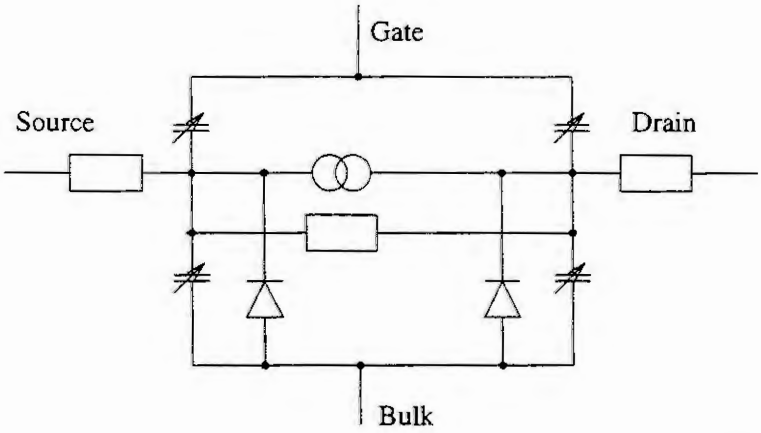


Fig. 6: Extended MOSFET-model

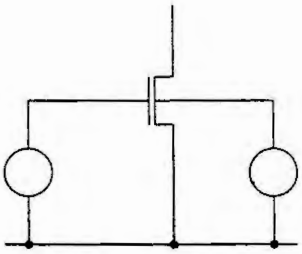


Fig. 7: Simple MOS circuit with loop of capacitors/voltage sources

Note that such a model has an inherent capacitor loop, which causes the problem to become of index 2 if the network equations are formulated with STA (see theorem of Lötstedt and Petzold in chapter 4). A more detailed analysis yields that the capacitor current is an index 2 variable, while other variables remain of index 1.

In MNA formulation however this current is not included within the vector of unknowns; hence the use of such a model must not necessarily lead to an index 2 problem. Nevertheless there is a hidden index 2 problem associated with this model even in MNA formulation. As an example may serve a frequently used part of an MOS circuit, as shown in fig. 7. In this case there exists a loop of voltage sources and capacitors, such that the currents through the voltage sources - which are included in the vector of unknowns - are of index 2. Fig. 8 contains the current waveform in such a situation. The dashed line is computed with the trapezoidal rule, which is not stable enough for these problems; the continuous line is computed with the more stable BDF method. Other examples are given by Wriedt in [18].

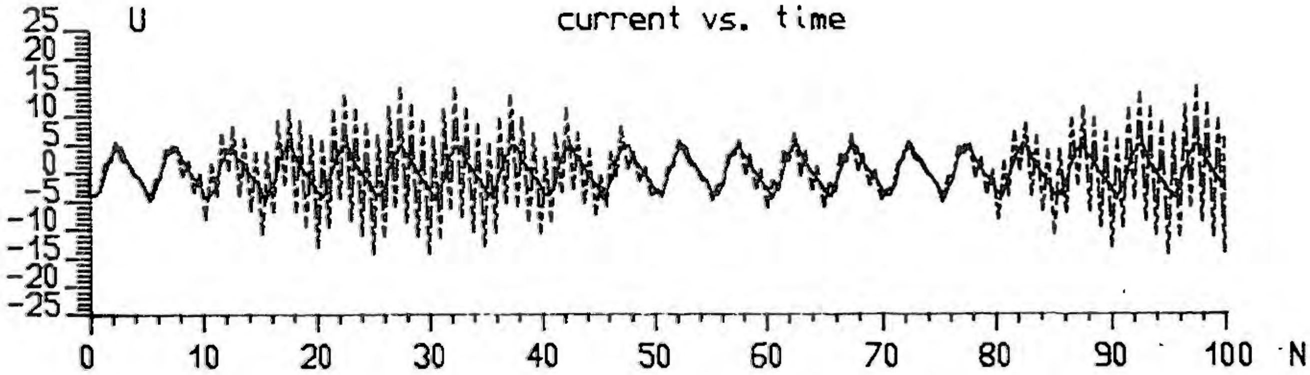


Fig. 8: MOS inverter with a circuit like fig. 7: Current through the voltage sources of index 2

The knowledge about the particular index 2 variables may be used in this case for a minor modification of the model in order to get a regularization (i.e. index reduction) of the problem (cf. [8], [15]):

If the capacitances of the pn-junction are linked to the outer drain resp. source node rather than to the intrinsic nodes, then the accuracy of the model is not seriously reduced. However the loop of voltage sources and capacitors is broken in every case, and the index 2 problem is reduced to index 1.

6 Conclusion

In this paper we have discussed the connections between models for time domain simulation and numerical properties of circuit simulation problems. We have seen that the mathematical modelling of electric circuits by CAD-methods leads to differential-algebraic systems. The numerical problems to be expected when solving these equations are mainly caused by a higher DAE-index. We have given a short review of the main notions of index.

On one hand, the index of a DAE appearing from circuit simulation depends on the used modelling technique. To show this, we discussed the modelling of a LC oscillator circuit by Sparse Tableau and Modified Nodal Analysis. On the other hand, among many other items, the modelling of basic elements and semiconductor devices controls the index, too. As an example, we investigated how the level of accuracy for modelling a MOSFET affects the index.

We saw that capacitor loops occur in more complex MOSFET models and cause DAEs of index 2. A regularization by a change of the modelling technique is not always possible. However, a deep knowledge about the index 2 variables may be used to modify the model to get an index 1 DAE.

To handle with the numerical problems in time domain simulation of electric circuits, it is necessary to investigate further the various influences of the mathematical model on the DAE-index: setup of equations, classical or charge-oriented formulation and modelling of basic elements and semiconductor devices. A classification of circuits with respect to the appearing index may be advantageous.

Acknowledgment

We are indebted to the Numerical Analysis group of R. Bulirsch and P. Rentrop from the TU München as well as to H. Wriedt from the University of Hamburg and Q. Zheng from Siemens AG for helpful discussions.

References

[1] S. L. Campbell and C. W. Gear: The index of general nonlinear DAEs, *Preprint, Dept. of Mathematics, North Carolina State University, Raleigh, 1993*

[2] G. Denk: An improved numerical integration method in the circuit simulator SPICE2-S. In R. Bank et al., Mathematical Modelling and Simulation of Electrical Circuits and Semiconductor Devices, *ISNM 93 (1990), pp. 85-99*

[3] D. E. Ward and R. W. Dutton: A charge-oriented model for MOS transistor capacitances, *IEEE J. Solid-State Circuits, vol. SC-13 (1978), pp. 703-708*

[4] U. Feldmann, U. A. Wever, Q. Zheng, R. Schultz and H. Wriedt: Algorithms for modern circuit simulation, *Archiv für Elektronik und Übertragungstechnik 46 (1992), pp. 274-285*

[5] C. W. Gear: Differential-algebraic equations, indices, and integral algebraic equations, *SIAM J. Numer. Anal. , vol. 27 (1990), pp. 1527-1534*

[6] C. W. Gear and L. Petzold: ODE methods for the solution of differential/algebraic equations, *SIAM J. Numer. Anal. , vol. 21, no. 4 (1984), pp. 716-728*

[7] M. Günther: Charge-oriented modelling of electric circuits and Rosenbrock-Wanner methods, *to appear in: J. of Computing and Information*

[8] M. Günther and P. Rentrop: Multirate ROW methods and latency of electric circuits, *Appl. Numer. Math.13 (1993), pp. 83-102*

[9] G. D. Hachtel, R. K. Brayton and F. G. Gustavson: The sparse tableau approach to network analysis and design, *IEEE Trans. Circuit Theory, CT-18 (1971), pp. 101-113*

[10] E. Hairer, Ch. Lubich and M. Roche: The numerical solution of differential-algebraic systems by Runge-Kutta methods. Lect. Notes Math. 1409, *Springer Verlag, Berlin, 1989*

[11] C. -W. Ho, A. E. Ruehli and P. A. Brennan: The modified nodal approach to network analysis, *IEEE Trans. Circuits and Systems, CAS-22 (1975), pp. 505-509*

[12] W. Kampowsky, P. Rentrop and W. Schmidt: Classification and numerical simulation of electric circuits, *Surveys Mathematics Industry 2 (1992), pp. 23-65*

[13] P. Lötstedt, L. Petzold: Numerical solution of nonlinear differential equations with algebraic constraints I: Convergence results for backward differentiation formulas, *Mathematics of computation, vol. 46, no. 174 (1986), pp. 491-516*

[14] L. Petzold: Differential/algebraic equations are not ode's, *SIAM J. Sci. Stat. Comput. , vol. 3, no. 3 (1982), pp. 367-384*

[15] W. Schmidt: Limit cycle computation of oscillating electric circuits, *to appear in: R. Bank et al. : Mathematical modelling and simulation of electric circuits and semiconductor devices. Proceedings of a conference held at the Mathematisches Forschungsinstitut Oberwolfach, 1992*

[16] E. R. Sieber, U. Feldmann, R. Schultz and H. Wriedt: Timestep control for charge conserving integration in circuit simulation, *to appear in: R. Bank et al. : Mathematical modelling and simulation of electric circuits and semiconductor devices. Proceedings of a conference held at the Mathematisches Forschungsinstitut Oberwolfach, 1992*

[17] H. Shichman and D. A. Hodges: Insulated-gate field-effect transistor switching circuits, *IEEE J. Solid State Circuits, SC-3 (1968), pp. 285-289*

[18] H. Wriedt: Transientensimulation elektrischer Netzwerke mit TR-BDF, *to appear in: R. Bank et al. : Mathematical modelling and simulation of electric circuits and semiconductor devices. Proceedings of a conference held at the Mathematisches Forschungsinstitut Oberwolfach, 1992*

Structured Modelling and Analysis of Saturated Synchronous Machines

N. Dourdoumas, M. Fette, C. Kröger, J. Voss

University of Paderborn
Department of Electrical Engineering
Pohlweg 47 – 49
33098 Paderborn, Germany

Abstract. In this paper a model of a synchronous machine with a special orthogonal structure is given. The model is extended to the phenomena of magnetic saturation, where the effects of cross–magnetization are included. Some comments on the necessity of including such effects to get accurate parameters for the model are stated out. With the special kind of mathematical modelling used for both models, with and without saturation, analytical expressions can be derived easily, for example to make statements about all possible equilibrium points. In the case of a model with magnetic saturation it is shown, that this extended model has an orthogonal structure, too. Furthermore, with this kind of analytical analysis and modelling of synchronous machines it is possible to do an improved analysis of the dynamic behavior of power systems.

1. Introduction

Most of the major electric power system breakdowns in recent years have been caused by the dynamic response of the system to disturbances. Economical and environmental pressures are causing power systems to be operated ever closer to their limits of stability. Additionally the dynamical behavior of power systems changes to be extremely nonlinear. Therefore it is even necessary to use improved nonlinear models for the analysis of the dynamic behavior of these systems.

An important model in power system engineering is that of a synchronous machine. Very often an idealized model is used, where all flux densities are assumed to be proportional to the currents. Therefore magnetic saturation or other nonlinear effects can't be treated. But, which is well known from practice, saturation has a great influence to the machine's behavior and much ingenuity has been used in devising methods of taking it into account; for example, the use of 'Potier reactance'. Such methods do not, however, introduce the nonlinear property into the basic theory. They are directed mainly to the determination of appropriate values of constants to suit the particular problem, the constants being defined in relation to a linear theory [1].

The accurate prediction of synchronous machine steady–state and transient performances also requires the precise determination of the machine parameters. In this context, some investigators suggested empirical approaches based on measurements to determine the synchronous machine reactances. Others proposed one or two saturation factors, obtained from the open–circuit saturation curves, for modifying the d– and q–axis reactances. A third group tried to calculate the machines parameters by analyzing the flux distribution insight the machine using methods of finite difference and the finite element. In most of these approaches, the effect of any magnetic coupling between the d– and q–axis windings, called cross–coupling or cross–magnetization, is totally ignored or is just presented partly. Many of these techniques do not give a clear physical insight of the saturation phenomenon in the d– and q–axis frame model and do not show the effect of various saturation factors on the machine parameters [2].

In this paper we present a nonlinear model of the machine, which allows to study the effects of the nonlinearities in a very structured manner, where usually a theory which allowed directly for nonlinear effects introduces much complication [1]. The models are in such a shape, that they can be handled mathematically in an efficient way. It is not the goal to fit all measured points of a saturation curve in detail, but we are interested in the typical behavior and the influence of important nonlinear effects on the dynamics of the system.

This paper is organized as follows. In section 2, a model of the synchronous machine with a special orthogonal structure is proposed. The model is extended in section 3 to the effect of magnetic saturation, where some comments are given on the necessity for including cross–magnetization effects of the core. Later on, in section 4, analytical expressions are derived with the special kind of a very compact mathematical representation of the model to make statements about all possible equilibrium points in dependence from the inputs. In section 5 the effect of a magnetic saturation on the equilibrium points is shown, because the extended model has also an orthogonal structure.

2. Model of the Machine

A special kind of Park transformation, which is based on an idea of Fouad and Anderson [3], is applied to the well known mathematical description of the synchronous machine [4, 5] connected to an infinite bus. This transformation matrix is orthogonal and therefore the transformed machine model preserves the symmetric structure from the time–variant model. Moreover, the derived model is provided with an essential orthogonal structure. The new model of a synchronous machine is then given by a system of *explicit* differential equations where the currents are divided into two groups according to the d– and q–axis of the Park system.

$$\mathbf{i}_1 := \left(i_d \;\; i_E \;\; i_D\right)^T \qquad\qquad \mathbf{i}_2 := \left(i_q \;\; i_Q\right)^T \tag{2.1}$$

Furthermore, with this and with the possibility of defining some important linearly independent weighting vectors g, the system–matrix is partitioned into submatrices $\mathbf{A}_1 ... \mathbf{A}_4$ compactly, where the g–vectors are multiplied by resistances respectively inductances of the machine.

$$\mathbf{A}_1 := -\left(r\mathbf{g}_1 \;\; r_E\mathbf{g}_2 \;\; r_D\mathbf{g}_3\right) \quad \mathbf{A}_2 := -\left(L_q\mathbf{g}_1 \;\; kM_Q\mathbf{g}_1\right) \quad \mathbf{A}_3 := \left(L_d\mathbf{g}_4 \;\; kM_E\mathbf{g}_4 \;\; kM_D\mathbf{g}_4\right) \quad \mathbf{A}_4 := -\left(r\mathbf{g}_4 \;\; r_Q\mathbf{g}_5\right) \tag{2.2}$$

As an infinite bus is assumed for this model, the system's only inputs are the mechanical torque T_m and the excitation voltage u_E. Due to the used transformation, the following mathematical model is in Park's coordinates:

$$\begin{aligned} \frac{d\mathbf{i}_1}{dt} &= \mathbf{A}_1\mathbf{i}_1 + \omega\mathbf{A}_2\mathbf{i}_2 + \mathbf{g}_1 U\sin\delta - \mathbf{g}_2 u_E & \qquad \frac{d\omega}{dt} &= \mathbf{i}_1^T\mathbf{M}\mathbf{i}_2 - \frac{c}{J}\omega + \frac{1}{J}T_m \\ \frac{d\mathbf{i}_2}{dt} &= \omega\mathbf{A}_3\mathbf{i}_1 + \mathbf{A}_4\mathbf{i}_2 - \mathbf{g}_4 U\cos\delta & \qquad \frac{d\delta}{dt} &= \omega - \omega_R \end{aligned} \tag{2.3}$$

An explanation of the model parameters is given in the appendix of this paper. To provide the machine with an excitation power, the excitation voltage u_E has to be negative.

The g–vectors can also be found in the feedback loops from the mechanical into the electrical part and in the path of the excitation voltage u_E. Although the components of the g–vectors are expressions of inductances, they don't have any effect on the equilibrium points, which will be shown later on. The mechanical part of the system is given by the two equations on the right hand side of (2.3), in which the electrical torque is represented by the term $\mathbf{i}_1^T\mathbf{M}\mathbf{i}_2$. $\mathbf{M}$ is called torque–matrix and contains the following entries:

$$\mathbf{M} := \frac{1}{J}\begin{pmatrix} L_d - L_q & kM_Q \\ -kM_E & 0 \\ -kM_D & 0 \end{pmatrix} \tag{2.4}$$

A mechanical damping torque $T_d = c\omega$ is modelled to be proportional to the angular velocity of the shaft. In spite of using a linear magnetization curve for this model, the transformed description has a nonlinear character, which is caused by the feedback loops from the mechanical into the electrical part.

3. Modelling of the Saturation

The modelling of the magnetic saturation is done in such a way, that in the case of small currents the derived model tends continuously to the model without saturation. With the help of the canonical unit vector $\mathbf{e}_1$ the electric circuit can be represented by the following formulae, which contain expressions of the magnetic fluxes:

$$\frac{d\Psi_1}{dt} = -\mathbf{R}_1\,\mathbf{i}_1 - \omega\Psi_q\,\mathbf{e}_1 - \mathbf{w}_1 \tag{3.1}$$

$$\frac{d\Psi_2}{dt} = \omega\Psi_d\,\mathbf{e}_1 - \mathbf{R}_2\,\mathbf{i}_2 - \mathbf{w}_2 \tag{3.2}$$

In contrast to common saturation models in this section an analytical model for synchronous machines is developed without any constraints on the range of the currents. Therefore, this model is destined to describe the dynamical reaction of the machine on big disturbances, which for example can be caused by immense load changes.

The vectors Ψ_1 and Ψ_2 consist of the flux–components of the d– and q–axis, where $\mathbf{w}_1$ and $\mathbf{w}_2$ contain all voltages:

$$\Psi_1 := \left(\Psi_d \;\; \Psi_E \;\; \Psi_D\right)^T \qquad \Psi_2 := \left(\Psi_q \;\; \Psi_Q\right)^T \qquad \mathbf{w}_1 := \left(u_d \;\; u_E \;\; u_D\right)^T \qquad \mathbf{w}_2 := \left(u_q \;\; u_Q\right)^T$$

$\mathbf{R}_1$ and $\mathbf{R}_2$ are diagonal matrices of resistances: $\mathbf{R}_1 := diag(r,\ r_E,\ r_D)$ $\mathbf{R}_2 := diag(r,\ r_Q)$

The magnetic fluxes in (3.1) and (3.2) must be obtained from the magnetization curve [6]. In view of the fact that the magnetic fluxes of one direction cause a cross–magnetization of the iron core in the other direction, the fluxes of one axis must be weighted additionally by a function of the other axis' currents. Therefore, the magnetic flux of one axis depends generally on the corresponding current vector and only on the norm of the remaining second current vector.

It has also been found experimentally that the effect of cross–magnetizing will reduce the magnetic flux linkages in both the d– and q–axis. Therefore, other authors define the per unit d– and q–axis mutual reactances as the per unit d– and q–axis mutual flux linkages divided by the corresponding per unit d– and q–axis ampere–turns respectively, the cross–magnetizing effect could be included [2] in these reactances. From this, the "concept of the saturated reactances" can be derived [2], saturation factors S_d and S_q can be determined experimentally, where the mutual reactances can be calculated. Previous results [2] show that:

- There are large discrepancies between the measured results and the unsaturated values of both the d– and q–axis mutual reactances.
- The accuracy of the machine reactances which include only the effect of the d– and q–axis saturation factors and do not consider the cross–magnetizing effect between the two axes is poor.
- The accuracy of the machine reactances which include the effect of both the cross–magnetizing phenomenon in addition to the d– and q–axis saturation factors is good. Moreover, the results of the machine active and reactive power outputs which are calculated using these reactances are in a good agreement with the measured results.
- The effect of saturation is found to reduce the values of both the d– and q–axis mutual reactances. The magnitude of this reduction is appreciable. Moreover, the reduction in the q–axis mutual reactance is larger than that in the case of the d–axis mutual reactance.
- The effect of the cross–magnetizing on the d– and q–axis mutual reactances is large and its negligence or improper representation may lead to inaccurate results for both reactances.

In these studies the mathematical advantages of nonlinear descriptions with differentiable functions to study the typical behavior and the influence of saturation is of interest and not the "nonlinearization" of classical synchronous machine models with reactances as parameters. For mathematical convenience an arctan–function is chosen to describe the magnetization curve and for the weighting function to include the cross–magnetizing effect an exponential mapping is utilized.

The modelling of the saturation is only shown for the d–axis, because the way for the q–axis is quite similar.

$$\Psi_1 = \Psi_1\left(\mathbf{i}_1,\ \|\ \mathbf{i}_2\ \|\right) = e^{-k_q\|\mathbf{i}_2\|^2}\cdot\Phi_1(\ \mathbf{i}_1\ ) \qquad \Phi_1(\mathbf{i}_1) := \frac{2\Psi_s}{\pi}\ \arctan\left(\frac{\pi}{2\Psi_s}\Psi_{0j}\right) \qquad j = d,\ E,\ D \tag{3.3}$$

The phenomenon of the cross–magnetizing is shown in the Figure 1, where the effect on the weighting function is demonstrated for a wide range of currents. In Figure 2 different saturation coefficient functions for both the d– and q–axis are given. The dotted curves are those given by polynomial functions from [2], the crossed curves are given by the proposed exponential approach with the coefficients k_d and k_q.

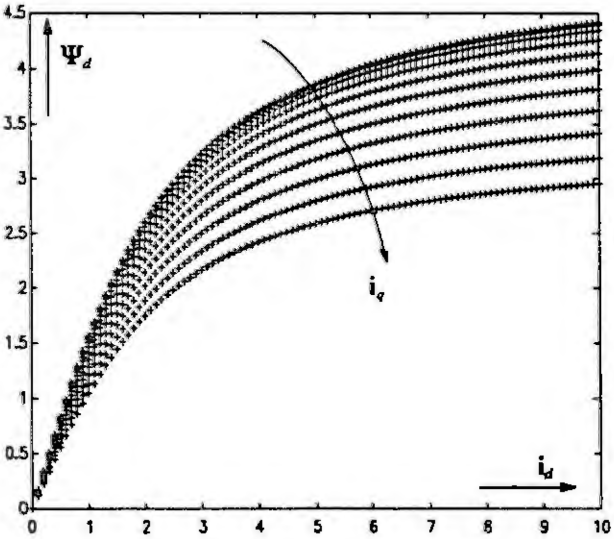


Fig. 1: Magnetization curve of the d–axis with respect to cross–magnetization by the q–axis current

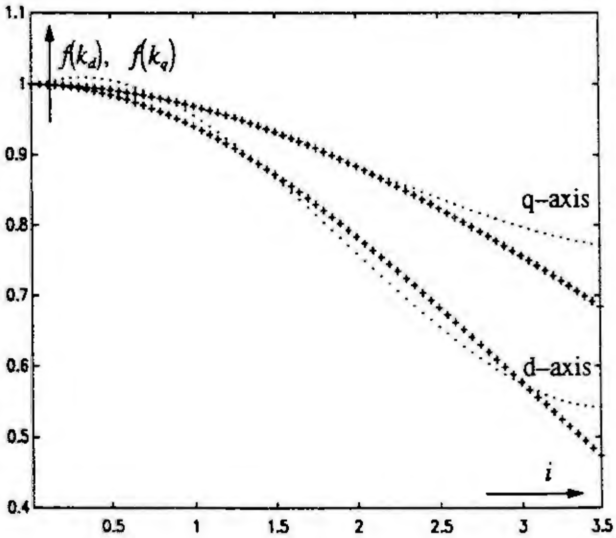


Fig. 2: Comparison of different saturation coefficients for both the d– and q–axis

One can see from Fig. 2, that in the q–axis case there is a good fit between the crossed and the dotted curve, the approximation is acceptable. In the case of d–axis, there is a discrepancy between the two curves, where the crossed curve form the exponential approach isn't a good approximation over the whole interval of currents. On the other hand, the dotted curves show an overshoot over the value one, which means an increase and not a reduction by the magnetic flux linkages. This is contradictionally to physics.

The fluxes are limited by the value Ψ_{s1} and the fluxes Ψ_{0i} in (3.3) are fluxes without saturation, which leads to a linear dependence on the corresponding current vector.

$$\Psi_{01} := \left(\Psi_{0d} \;\; \Psi_{0E} \;\; \Psi_{0D}\right)^T = \Lambda_1 \; \mathbf{i}_1 \qquad (3.4)$$

In (3.4) the rows of the nonsingular inductance matrix Λ_1 are denoted by λ_{11}^T , λ_{12}^T and λ_{13}^T . Moreover, the known weighting vectors $\mathbf{g}_1$, $\mathbf{g}_2$ and $\mathbf{g}_3$ are given as the columns of the inverse Λ_1^{-1} . The change in time of the magnetic flux is needed for the equation (3.1):

$$\frac{d}{dt}\Psi_1 = \left(\frac{d}{dt}e^{-k_q\|\mathbf{x}_2\|^2}\right) \Phi_1(\mathbf{i}_1) + e^{-k_q\|\mathbf{x}_2\|^2}\frac{d}{dt}\Phi_1 \qquad (3.5)$$

The first derivation in (3.5) results in: $\frac{d}{dt}e^{-k_q\|\mathbf{x}_2\|^2} = -\,2k_q \left(\mathbf{x}_2^T\frac{d\mathbf{x}_2}{dt}\right) e^{-k_q\|\mathbf{x}_2\|^2}$. A time derivation of the vector Φ_1 provides:

$$\frac{d\Phi_1}{dt} = \frac{\partial\Phi_1}{\partial i_1}\cdot\frac{d\mathbf{i}_1}{dt} = diag\left[\frac{1}{1+\left(\frac{\pi}{2\Psi_{s1}}\lambda_{1k}^T\mathbf{i}_1\right)^2}\right]\Lambda_1\frac{d\mathbf{i}_1}{dt} \qquad k = 1,2,3$$

The Jacobian matrix in above equation is represented as a product of the inductance matrix Λ_1 and a square matrix, which from now is called saturation matrix $\mathbf{S}_1$. The inverse $\mathbf{S}_1^{-1}$ exists for all states. Its main diagonal elements are denoted by s_{11}, s_{12} and s_{13} and are called saturation coefficients. From this, the equation (3.1) of the electrical system becomes:

$$\frac{d}{dt}\mathbf{i}_1 = 2k_q\Lambda_1^{-1}\mathbf{S}_1^{-1}\Phi_1\left(\mathbf{i}_2^T\frac{d}{dt}\mathbf{i}_2\right) + e^{k_q\|\mathbf{x}_2\|^2}\left(-\Lambda_1^{-1}\mathbf{R}_1\mathbf{S}_1^{-1}\mathbf{i}_1 - \omega\Psi_q s_{11}\Lambda_1^{-1}\mathbf{e}_1\right) - e^{k_q\|\mathbf{x}_2\|^2}\left(s_{11}\Lambda_1^{-1}\mathbf{e}_1 U\sin\delta + s_{12}\Lambda_1^{-1}\mathbf{e}_2 u_E\right)$$

where fact of the product's $\mathbf{S}_1^{-1}\mathbf{R}_1$ commuting is used; the product $-\,\Lambda_1^{-1}\mathbf{R}_1$ is already known as the system matrix $\mathbf{A}_1$.

One effect of the saturation is, that the explicit structure of the differential equations have been lost and some more couplings between the two axes have been introduced. After modelling the fluxes of the q–axis in an analogous way the new model is received:

$$\frac{d}{dt}\mathbf{i}_1 = 2k_q\Lambda_1^{-1}\mathbf{S}_1^{-1}\Phi_1\left(\mathbf{i}_2^T\frac{d}{dt}\mathbf{i}_2\right) + e^{k_q\|\mathbf{x}_2\|^2}\left(\mathbf{A}_1\mathbf{S}_1^{-1}\mathbf{x}_1 - \omega\Psi_q s_{11}\mathbf{g}_1\right) - e^{k_q\|\mathbf{x}_2\|^2}\left(s_{11}\mathbf{g}_1 U\sin\delta + s_{12}\mathbf{g}_2 u_E\right) \qquad (3.6)$$

$$\frac{d}{dt}\mathbf{i}_2 = 2k_d\Lambda_2^{-1}\mathbf{S}_2^{-1}\Phi_2\left(\mathbf{i}_1^T\frac{d}{dt}\mathbf{i}_1\right) + e^{k_d\|\mathbf{i}_1\|^2}\left(A_4 S_2^{-1}\mathbf{i}_2 + \omega\Psi_d s_{21}g_4 - s_{21}g_4 U\cos\delta\right) \qquad (3.7)$$

$$\frac{d}{dt}\omega = \frac{1}{J}\left[\left(\Psi_q i_d - \Psi_d i_q\right) - c\omega + T_m\right] \qquad (3.8)$$

$$\frac{d}{dt}\delta = \omega - \omega_R \qquad (3.9)$$

The electric torque can't be represented any longer by the matrix **M** and therefore the definitions of the fluxes must be used.

4. Equilibrium Points of the Model without Saturation

Since the synchronous machine is a nonlinear system, stability can only be computed for equilibrium points. Therefore it is necessary to determine the equilibrium points of the synchronous machine with respect to a fixed control vector. In this section some analytical expressions are derived, which can be used to make a decision about the existence and uniqueness of equilibrium points for a chosen input vector. It should be pointed out, that the steady–state of the original model corresponds to the equilibrium state of the Park model. In the usual way the time derivation of the state vector in (2.3) is set equal to zero for a calculation of the equilibrium points. By the last right hand equation the equilibrium value of the angular speed is given immediately. An equilibrium state for the entire system only exist if the angular speed of the rotor is identical to the angular frequency of the infinite bus. By using this result for the first two equations of (2.3), they become a linear system of equations, if the rotor angle δ is regarded to be known. These two equations of the electric circuit can be converted into two sums of the g –vectors, because the columns of the system–matrices are given by terms of the g –vectors.

$$0 = \left(ri_d + \omega_R L_q i_q + \omega_R k M_Q i_Q - U\sin\delta\right)g_1 + \left(r_E i_E + u_E\right)g_3 + r_D i_D g_3$$

$$0 = \left(\omega_R L_d i_d + \omega_R k M_E i_E + \omega_R k M_D i_D - ri_q - U\cos\delta\right)g_4 - r_Q i_Q g_5$$

In view of the fact that the vectors g_1, g_2, g_3 and the vectors g_4, g_5 are linearly independent, the vector sums can only vanish, if and only if all coefficients vanish. From these conditions the currents of an equilibrium state can be determined as functions of the rotor angle δ and of the excitation voltage u_E. Furthermore, the well–known synchronous reactances $x_d := \omega_R L_d$ and $x_q := \omega_R L_q$ are introduced for more convenience in writing the following formulae:

$$i_E = -\frac{u_E}{r_E} \qquad i_D = i_Q = 0 \tag{4.1}$$

$$i_d = \frac{\omega_R k M_E x_q u_E + U r_E\left(x_q\cos\delta - r\sin\delta\right)}{r_E\left(r^2 + x_d x_q\right)} \qquad i_q = \frac{-\omega_R k M_E r u_E + U r_E\left(x_d\sin\delta - r\cos\delta\right)}{r_E\left(r^2 + x_d x_q\right)} \tag{4.2}$$

No voltage is induced in the rotor windings in steady–state, because the rotor acts synchronously with the magnetic field of the stator. The last two functions (4.2) represent a closed curve in the (i_d, i_q)– plane by a parameter description in which the rotor angle δ is the parameter. This curve consists of all points (i_d, i_q), which satisfy the equations of the electric circuit for a chosen control vector. But not all points of this closed line are also solutions of the entire system. Only such points of the curve are equilibrium points, which supply the machine with an electric torque that is equal to the efficient mechanical torque. The efficient mechanical torque is equal to the forcing torque which is reduced by the damping torque in steady–state. The equivalence between the efficient mechanical torque and the electrical torque is represented by the differential equation of the angular speed, which is set to zero. By substituting the currents in this equation by the determined functions (4.1) till (4.2) of the torque angle δ, a nonlinear scalar equation for δ is obtained after a laborious calculation.

$$m_1\sin^2\delta + m_2\cos^2\delta + m_3\sin\delta\cos\delta + m_4\cos\delta + m_5\sin\delta + m_6 = 0 \tag{4.3}$$

The abbreviations m_1 till m_6 are used for:

$$m_1 = -U^2\, r_E^2\, r\, x_d\left(L_d - L_q\right) \qquad m_2 = -m_1\frac{x_q}{x_d}$$

$$m_3 = m_1\frac{x_q}{r} + m_2\frac{r}{x_d} \qquad m_4 = U\, r_E\, kM_E\, r\left[x_q\left(x_d - x_q\right) - \left(r^2 - x_q^2\right)\right] u_E$$

$$m_5 = U\, r_E\, kM_E\left[r^2\left(x_d - x_q\right) + x_d\left(r^2 + x_q^2\right)\right] u_E \qquad m_6 = r_E^2\left(r^2 + x_d x_q\right)^2\left(T_m - c\omega_R\right) - \omega_R\left(kM_E\right)^2 r\left(r^2 + x_q^2\right) u_E^2$$

The first three coefficients are constant. But the coefficients m_4 and m_5 depend linearly on the excitation voltage u_E and the coefficient m_6 has a linear dependence on the forcing torque T_m and furthermore a quadratic dependence on the excitation voltage u_E.

Without making any simplifications of the model, the rotor angle equation can't be solved analytically. For a calculation of δ a numerical method is required. Therefore it is appropriate to know how many solutions of the equation are possible for a fixed control vector. With the help of the following substitution (4.4) $\mu := \cos\delta$ and $\eta := \sin\delta$ the rotor angle equation (4.3) is converted into an equation of a conic section, which describes a hyperbola in the (μ, η)– plane for every control vector. The condition $\mu^2 + \eta^2 = 1$ must be satisfied additionally, because the new two variables μ and η aren't independent. Therefore all solutions of the equations (4.4) are on the unit circle and the hyperbola can intersect this unit circle at most four times. This means, that for a salient pole machine at most four equilibria can exist for a fixed control vector. A statement about the characteristics of these equilibria – whether they are stable or unstable – can't be made at this point.

Some more parameters are used to describe the hyperbola:

- The directional angle, the angle between a principal axis of the hyperbola and a parallel of μ –axis, doesn't depend on the control vector, but on the resistance r of the stator windings.
- The center of the hyperbola moves along a straight line through the origin of the coordinate system while the excitation voltage u_E is varied.

- The hyperbola can be represented in a normal form by the use of the principal values in such a coordinate system, whose origin is at the center of the hyperbola and whose axes are identical to the principal axes of the hyperbola.

Two special assumptions are made to find out the effects of some parameters.

4.1 Neglecting the stator–resistance r

If the stator resistance r is neglected in equation (4.3), the coefficients m_1 , m_2 and m_4 vanish. With the help of a trigonometric theorem the torque angle equation can be written as

$$U^2 \frac{L_d - L_q}{2\; x_d\; x_q} \sin 2\delta \;-\; U \frac{kM_E\; u_E}{r_E\; x_d} \sin\delta \;=\; T_m - c\omega_R$$

The right hand side of the equation above is given by the efficient mechanical torque. On the left hand side it can be seen that the anti–symmetry of the solutions is caused by the difference $L_d - L_q$. Generally, the dependence of the solutions on the parameters can be discussed, criteria for the existence and uniqueness can be given. [7]

4.2 Turbogenerator

As the rotor of a turbogenerator has no salient poles, the self– inductances of the stator windings are constant. This leads to the condition $L_d = L_q$, which allows to define a stator impedance $Z_s = \sqrt{r^2 + x_d^2}$.

Z_s is known from the classical theory of the synchronous machine, in which it is used to describe the drop of the induced voltage by the effect of a load. On condition that $L_d = L_q$, the first three coefficients of (4.3) get the value of zero and therefore all quadratic and mixed termes of the trigonometric functions in the rotor angle equation vanish. Thereby the rotor angle equation can be converted to:

$$\sin\left[\delta - \arctan\left(\frac{r}{x_d}\right)\right] \;=\; \frac{\left(T_m - c\omega_R\right) r_E^2\, Z_s^2 - \omega_R\, \left(kM_E\right)^2\, r\, u_E^2}{-\,U\; kM_E\; r_E\; Z_s\; u_E} \;=\; q\,\left(u_E,\; T_m\right) \qquad (4.5)$$

where q is a function of the input vector. A criterion for the existence of an equilibrium point is given immediately by the sin–function of (4.5): $\mid q\;(u_E, T_m) \mid \;\leq\; 1$.

The critical torque angle δ_c can also be determined from equation (4.5): $\delta_c \;=\; \frac{\pi}{2} \;+\; \arctan\left(\frac{r}{x_d}\right)$

The critical torque angle δ_c doesn't depend any longer on the excitation voltage u_E . But only if the stator resistance r is omitted, the classical result $\delta_c = \pi/2$ for the critical torque angle is received. If the resistance r grows up, which can also be caused by an additional transmission line, the critical torque angle δ_c is shifted toward higher values. Moreover, the equilibrium values of the torque angle δ are symmetric to the point δ_c .

5. Equilibrium Points of the Extended Model

By setting the derivation of the state vector equal to zero, the unchanged equilibrium value of the angular speed is obtained immediately from the equation (3.9). Because the exponential functions can't vanish, the electrical part of the system in the steady–state can be written as:

$$0 \;=\; s_{11}\left[r\, i_d + \omega_R \Psi_q \;-\; U\, \sin\delta\right] g_1 \;+\; s_{12}\left[r_E\, i_E + u_E\right] g_2 + s_{13}\, r_D\, i_D\, g_3$$

$$0 \;=\; s_{21}\left[r\, i_q - \omega_R \Psi_d \;+\; U\, \cos\delta\right] g_4 \;+\; s_{22}\, r_Q\, i_Q\, g_5$$

Because the g–vectors are linearly independent and the saturation coefficients are always different from zero, the remaining coefficients have to vanish. From this it is obvious that the equilibrium values of the currents i_E, i_D and i_Q haven't changed by the effect of saturation. For the remaining states i_d, i_q and δ a nonlinear equation system has to be solved for a determination of the equilibrium points.

$$0 \;=\; r\, i_d \;+\; \omega_R \Psi_q\!\left(i_d,\; i_q\right) \;-\; U\, \sin\delta \qquad (5.1)$$

$$0 = -r\,i_q + \omega_R \Psi_d(i_d,\ i_q) - U\cos\delta \tag{5.2}$$

$$T_m - c\omega_R = \Psi_d(i_d,\ i_q)\ i_q - \Psi_q(i_d,\ i_q)\ i_d \tag{5.3}$$

Only the functions of the fluxes determine what kind of magnetization curve is assumed. In the case of a nonlinear curve these equations tend to those of the first model, if the currents become small. In the case of the extended model it is impossible to get solutions $i_d=i_d(\delta)$ and $i_q=i_q(\delta)$ from this nonlinear system. But from (5.1) and (5.2) a new relation can be obtained, in which the torque angle δ is eliminated.

$$\begin{aligned} U^2 &= \left[\omega_R\Psi_q(i_d,\ i_q) + r\,i_d\right]^2 + \left[\omega_R\Psi_d(i_d,\ i_q) - r\,i_q\right]^2 \\ &= (\omega_R\Psi_d)^2 + (\omega_R\Psi_q)^2 + r^2\left(i_d^2 + i_q^2\right) - 2\omega_R\left(\Psi_d\,i_q - \Psi_q\,i_d\right) r \end{aligned} \tag{5.4}$$

On the right hand side of (5.4) the electric torque appears, which is known from equation (5.3). Thus a dependence on the forcing torque T_m is introduced.

$$U^2 + 2\omega_R\left(T_m - c\omega_R\right) r = \omega_R^2\left(\Psi_d^2 + \Psi_q^2\right) + r^2\left(i_d^2 + i_q^2\right) \tag{5.5}$$

Equation (5.5) is an implicit representation of the electric circuit's solutions, because the right hand side of this equation represents a function of the Park currents i_d and i_q.

The solutions of the electric circuit can be computed as the level line at the height $h = U^2 + 2\omega_R\left(T_m - c\omega_R\right) r$ in a numerical way. In the case of a linear magnetization curve the function (5.5) consists of two paraboloids.

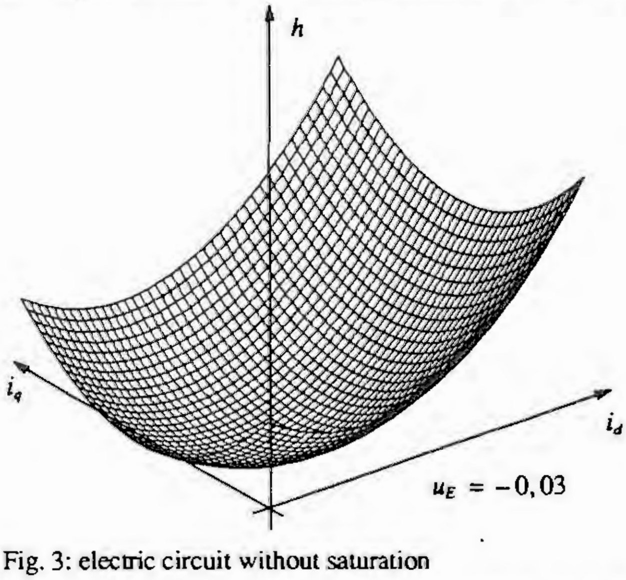


Fig. 3: electric circuit without saturation

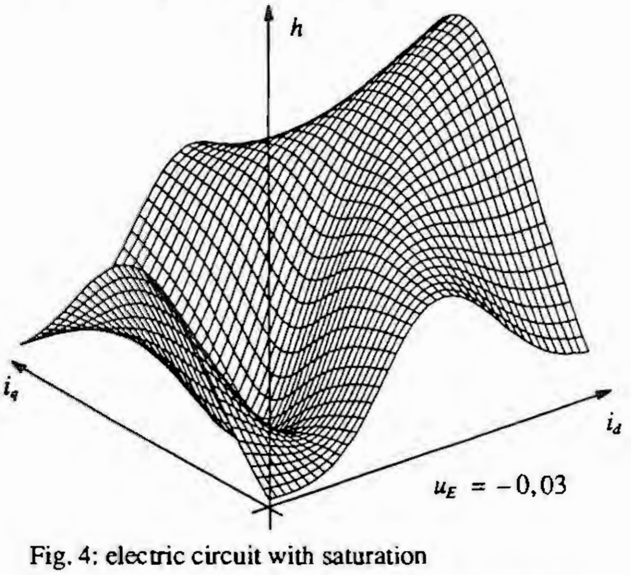


Fig. 4: electric circuit with saturation

The first one is given by a term of the magnetic fluxes and the second one in an expression of the stator circuit's heating loss. Therefore, if the stator resistance *r* is neglected or not, the function is always strongly monotonous for every direction, which causes only one closed curve as a level line for each height. Thus a saddle point of level lines can never occur.

If the nonlinear magnetization curve is assumed, the loss paraboloid provides only the function with unbounded values. The flux–term in the function is bounded for every direction. Furthermore, for a direction, which is not identical to the i_d– or i_q–axis, this part of the function tends towards zero for an increasing argument. Thereby additional level lines exist for one height, which depend on the parameter *r*. Because saddle points of level lines can occur in the dependence on *r*, the stator resistance *r* represents a bifurcation parameter [8] in the saturation model. If the stator resistance *r* is neglected in the extended model, level lines exist only up to certain height.

6. Conclusion

Most of the models of synchronous machines are based on the concept of linear theory, where they are "nonlinearized" to include for example saturation in the d– and q–axis frame model. Also, these models are simplified and for the analysis of the dynamic behavior of power systems, synchronous machines are represented only by their swing equations.

In this paper a detailed model of a synchronous machine with a special orthogonal structure is shown. The model is expanded by the effect of magnetic saturation with respect to analytical frame given by the unsaturated orthogonal model. The necessity of including cross-magnetization effects is reviewed. From the complete model of a saturated synchronous machine, which has also an orthogonal structure, analytical expressions can be derived easily to make a statement about all possible equilibrium points. The effect of magnetic saturation on the equilibrium points with respect to particular parameters is discussed with an extended model. For small currents this extended model with saturation converges steadily towards the model without saturation.

In general, this kind of modelling can be used for example for analytical parameter studies to compute the effect of changes for particular model parameters. It builds a framework for determining all kind of special machine parameters like critical excitation voltage or critical power angle. All results are interpretable physically.

Appendix

States:	i_d	direct axis synchronous current	**Inputs:**	u_E	excitation voltage
	i_q	quadrature axis synchronous current		T_m	mechanical forcing torque
	i_E	field current			
	i_D	damping current of the d-axis	**Infinite Bus:**	ω_R	angular frequency of the infinite bus
	i_Q	damping current of the q-axis		U	voltage of the infinite bus
	ω	angular velocity of the shaft			
	δ	rotor angle or torque angle			

Parameters of the Machine:

		k	coupling factor
J	inertia constant	r_E	resistance of the field winding
c	damping constant	r_D	resistance oft the damper winding of the d-axis
r	resistance of the stator windings	r_Q	resistance of the damper winding of the q-axis
L_d	d-axis synchronous self-inductance	L_q	q-axis synchronous self-inductance

M_D	mutual inductance between the damper winding of the d-axis and the stator windings
M_Q	mutual inductance between the damper winding of the q-axis and the stator windings
M_E	mutual inductance between the field winding a the stator windings
M_R	mutual inductance between the field winding and the damper winding of the d-axis

References

[1] Adkins, B; Harley, R.G.: "The General Theory of Alternating Current Machines: Application to Practical Problems", Chapman and Hall, London 1975

[2] El-Serafi, A.M.; Abdallah, A.S.: "Saturated Synchronous Reactances of Synchronous Machines", IEEE Trans. on Energy Conversion, Vol. 7, No. 3, September 1992, pp. 570 – 579

[3] Anderson, P.M.; Fouad, A.A.: "Power System Control and Stability", The Iowa State University Press, Ames, Iowa, U.S.A., Volume 1, Fourth Printing 1986

[4] Lerch, E.; Nour Eldin, H.A.; Wegmann, P.; Wehrli, P.: "Digitale Simulation der Synchronmaschine mit Zustandsraumdarstellung", etz-Archiv, Bd. 2 (1980), H. 12, S. 335 – 340

[5] Dourdoumas, N.; Fette, M.; Voß, J.: "Modelling and Simulation of Nonlinear Power Systems on Manifolds", Proceedings of the IMACS-Congress, Dublin, July 1991, pp. 1133 – 1134

[6] Ostovic, V.: "Dynamics of Saturated Electric Machines", Springer-Verlag, New York, Berlin, Heidelberg, 1989

[7] Fette, M.: "Strukturelle Analyse elektrischer Energieversorgungssysteme", Fortschrittberichte VDI, Reihe 21, Nr. 140, VDI-Verlag, 1993

[8] Abed, E.H.; Varaiya, P.P.: "Nonlinear oscillations in power systems", Electrical Power & Energy Systems, Vol. 6, No. 1, January 1984, pp. 37 – 43

MODELLING AND SIMULATION OF A SYSTEM FOR FAULT DETECTION AND DIAGNOSIS

E.Kraševec, M. Rak, Đ. Juričič
Department of Computer Automation and Control
Jozef Stefan Institute, Jamova 39, 61111 Ljubljana, Slovenia
E-mail: edvard.krasevec@ ijs.si

Abstract. The aim of the paper is to present an idea of using a model of a complete system for fault detection and diagnosis to support the engineering design process. For this purpose a simulation model of a system for fault detection and diagnosis of a DC motor is described.

1. INTRODUCTION

Modelling and simulation approach has already been recognized as a useful methodology which can be used efficiently when one deals with the problem of fault detection and diagnosis. An issue of using modelling and simulation as a research tool in the field of fault detection and diagnosis is usually hidden behind a common question: "What kind of a model of a treated system is needed in order to define and solve the problem of fault detection and diagnosis?" The researchers from almost all streams of the field use the term model. But the meaning of that term varies from a form of a mathematical model to the form of a qualitative model [3,8]. It depends on what approach to fault detection and diagnosis is used: model-based or knowledge-based. However the usefulness of modelling comes out also in the design phase of systems for fault detection and diagnosis. Of course, the concept of models, which support the design phase of a specific traditional engineering disciplines, is already well known. However the problem of transformation of the so called Fault Detection and Isolation schemes into a workable system for fault detection and diagnosis is still an open research issue [8]. According to our experience, modelling and simulation can play here an essential role as a problem solving methodology. However, general concepts should be first tailored to that particular problem domain.

System designer needs a model (models) which enables him an insight into the main aspects of the problem of fault detection and diagnosis for a treated process or object:

- perception of faulty behavior,
- knowledge about treated object or process,
- admissible experimentation procedures,
- decision making about faults - diagnostics.

A model developed with this purpose would enable rational engineering design decisions and thus acievement of systems objectives [5].

2. MODELLING A SYSTEM FOR FAULT DETECTION AND DIAGNOSIS - A CASE STUDY

Fault Detection and Diagnosis of a DC motor in a controlled DC motor drive has been the main object of our work. Basic goal is to build a "system" for fault detection and diagnosis, which consists of three main modules: perception module, decision (diagnostic) module and human operator. The latter can be treated as a part of the perception module which treats nonmeasurable sensual perceptions. But at this stage, only subsystems which treat measurable and numerically computable attributes are included into the perception module.

Basic function of the perception module is to generate symptoms from measured signals. For this purpose a method based on mathematical modelling and parameter estimation is used [4]. If a fault appears in the process this causes permanent change in a process coeficient if it is directly affected by the fault. Estimated parameters of the mathematical model are later used in a decision module as symptoms/attributes of the observed behavior.

Because our purpose was initially only to understand and to asses the limitations of the perception module, at that point we didn't make any final decisions about the structure of diagnostic module. However, in order to enable the analysis of relations between perception module and decision module a neural network was used as a diagnostic module. Of course, the presented view of the system for fault detection and diagnosis should only be treated as a conceptual model of the treated problem solving situation.

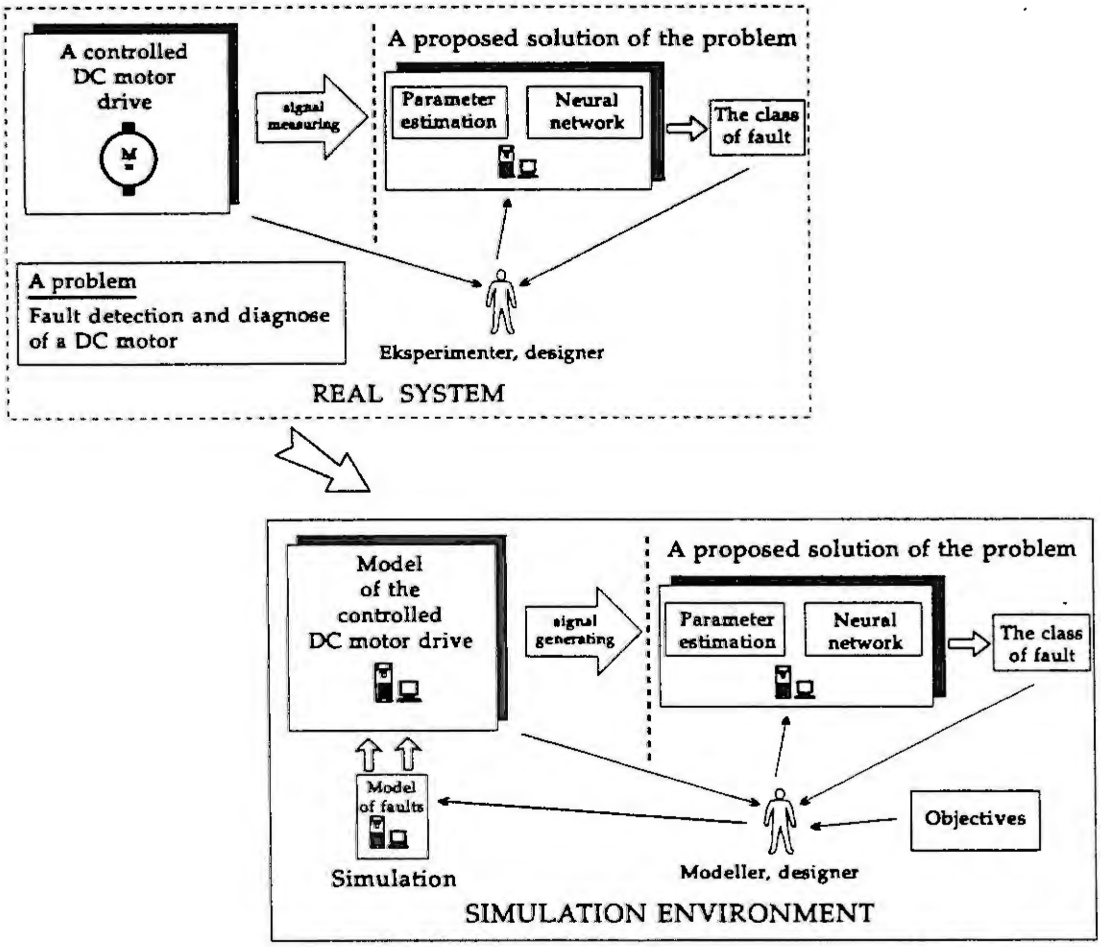


Figure 1: Modelling a system for fault detection and diagnosis of a DC motor.

As can be seen in Figure 1., in order to understand and estimate the approach to the problem of fault detection and diagnosis of a DC motor, a complete model of the problem solving situation (design) was built and transfered in a simulation environment. The basic assumption of the approach is expectation, that experimentation in simulation environment will enable a deeper insight into complex relations between an observed object (DC motor) and modules of the system for fault detection and diagnosis. Because these relations can become ambiguous and fuzzy, when the whole problem is treated in the real world of complex connections of hardware and software components.

Generally speaking, a modeller, who performes simulation experiments, should also be treated as a part of the simulation environment. In fact, he plays two roles at the same time. First, he is a a troubleshooting expert, who performs tests on the real DC motor. And second, he is a designer of a system for fault detection and diagnosis whose objective is to build a workable system according to specified requirements. Thus, all relations between objects in the real world (DC motor - system for fault detection and diagnosis - troubleshooting expert - system designer) are transfered in the simualtion environment where well known simulation concepts [1,6,9] can be used to analyse them in a controled way .

3. SIMULATION ENVIRONMENT

SIMULINK [7] was used for the implementation of the presented concepts. For this purpose several of its features were utilized to implement the principle of modularity and the principle of separation between model and experiment [1]. The complete model of a system consists of four connected submodels: model of a DC motor in a controlled DC drive, model of faults, model of a perception module and model of a decision module. As a kernel, a simulator of the DC motor in a controlled DC motor drive was developed. It represents an observed object - a DC motor in its environment. As a result of using the concept of experimental frame [1,9] a model of faults was defined. In fact, this model is based on knowledge analysis of troubleshooting charts for DC motors [2]. It represents a description of experiments for simulation of faulty behavior of the DC motor. The perception module was modelled with three connected units: state variable filters (SVF), least-square estimator and a unit which calculates physical parameters of the DC motor. Finally the diagnostic module was modelled with a neural network in the form of a three layered perceptron. The scheme of the simulation environment is presented in Figure 2.

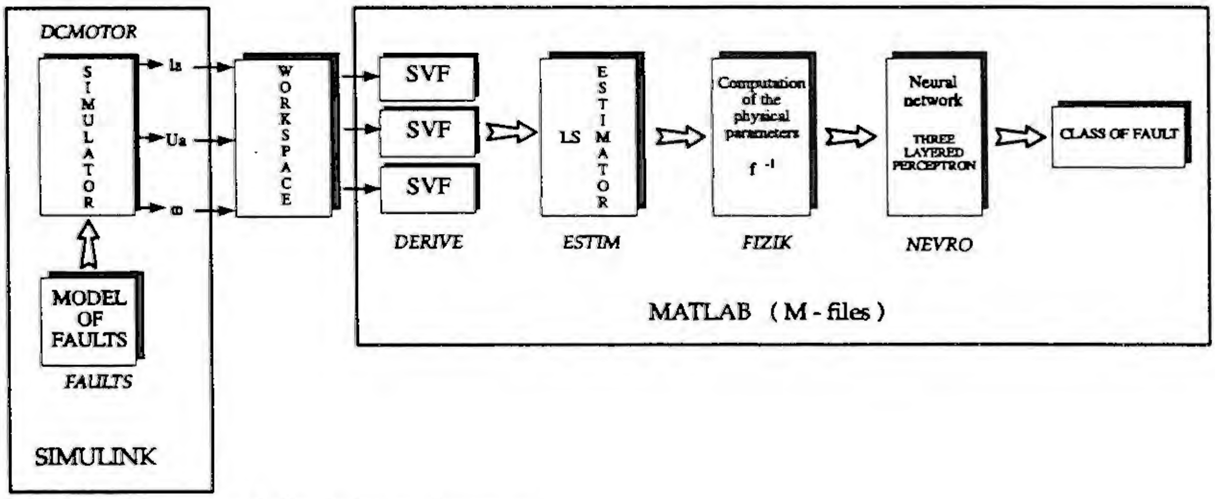


Figure 2: Structure of simulation environment.

4. EXPERIMENTATION

Process of engineering system design is usually an iterative procedure. During this process, a designer must take several decisions which have to be supported with rational arguments or based on practical experiences. However, a lot of questions conceived in an analysis of complex systems, which can not be just "calculated" from simple formulas. For example, the following set of questions, regarding the system for fault detection and diagnosis of a DC motor, can not be answered in a straightforward way:

a) Is it possible to use the proposed approach (parameter esimation) in all operating modes of the DC motor (constant load, variable load, start up, shut down) ?

b) How to estimate uncertainty of the final decision about a fault ?

c) How to asses influence of measuring accuracy, sampling frequency and signal filtering on behavior of the fault detection module and diagnosis module (the problem of specification of normal and faulty behavior regions [3]) ?

d) How to tune parameters of the applied algorithms (e.g. state variable filters) ?

Modeling and simulation approach, which is offered here can be expresed as follows:

"First, questions about the system are translated into experimental frames [1,9] for simulation experiments. Then the simulation experiments are planned and performed. In fact, such a simulation experiment can also include activities of construction of new models or modification of the available ones. Finally, an analysis and

representation of the simulation results reveals answers to the questions or new problems are identified."
At this stage of our work, only rather simple experiments are treated. A fault is introduced in the model of a DC motor as a parameter change of a DC motor during the simulation run. Of course, according to the knowledge analysis [2], the change of a parameter corresponds to a real fault. A partial result of a simulation experiment is shown in Figure 3. As can be seen, convergence of two physical parameters of a mathematical model of a DC motor is represented. A fault - "dirty comutator" - was simulated in this case as a step change of armature resistance R_{ac}. After the transient phenomena, a premanent change was established only for armature resistance R_{ac}, whilst the moment of inertia J_{ac} remained unchanged. However, such simulation experiments represent very rough model of real situations. But still they can give some insight into answers on the treated questions. For example, in case of "dirty comutator", simulation experiments can be used to study the following problem:
" Esimate the influence of a working point of a DC motor (temperature), sampling frequency and measuring accuracy on a range of normal and faulty behavior for parameter R_{ac} and a fault - "dirty comutator". "

5. CONCLUSION

Generally speaking, simulaton models can be used as conceptual problem solving tools in diverse scientific and engineering disciplines. However, if we want to use simulaton and modelling in an effective way, several problems connecetd with the problem domain and simulation methodology have to be answered and viewed from a proper aspect.
Our work can be put on the intersection of two research fields: Fault Detection and Diagnosis and Modelling and Simulation. We belive that a lot of new issues can be found in such an intersection. Of course, at present time we can not already speak about some kind of general modelling and simulation methodology for the field of fault detection and diagnosis. However, we want to express the need for such a methodology. Therefore our work can be treated only as an early attempt to open a new issue on the intersection of two research fields.

6. ACKNOWLEDGEMENTS

Support from the Forschungsyentrum KFA Jülich GmbH is gratefully acknowledged.

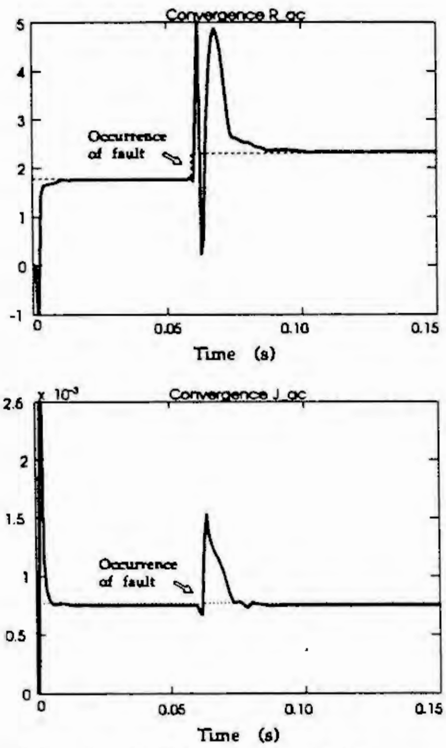


Figure 3: Simulation of a fault.

7. REFERENCES

[1] Cellier F.E, Continuous System Modelling, Springer-Verlag, New York, 1990.

[2] Electro Ccraft, DC Motors Speed Controls Servo Systems, Electro Craft Corporation, Minnesota, 1975.

[3] Himmelblau D.M., Fault Detection and Diagnosis in Chemical and Petrochemical Process, Elsevier, Amsterdam, 1978.

[4] Isermann R., Fault Diagnosis of Machines via Parameter Estimation, Proceedings of IFAC/IMACS Symposium "Safeprocess 91", Baden-Baden, 1991.

[5] Leitch R., Engineering Diagnosis: matching problems to solutions, International Conference on Fault Diagnosis, Tooldiag, Touluse, 1993, 837-844.

[6] Matko D., Karba R., Zupančič, Simulation and Modelling of Continuous Systems, A Case Study Approach, Prentice Hall, New York, 1992.

[7] Math Works, Simulink, User's Guide, The Math Works Inc., Massachsetts, 1992.

[8] Patton R., Frank P., Clark R., Fault Diagnosis in Dynamic Systems, Theory and Applications, Prentice HAll, New York, 1989.

[9] Ören T.I., Zeigler P.B., Concepts for advanced simulation methodologies, Simulation, 32(3), (1979), 69-82.

A COUPLED MAGNETIC NETWORK-FINITE ELEMENT MODEL FOR THE CALCULATION OF LOSSES IN STEEL LAMINATIONS USED IN ELECTRIC MACHINERY

L.R.Dupre* R.Van Keer J.A.A.Melkebeek***

* Department of Electrical Power Engeneering, University of Gent, Belgium
** Department of Mathematical Analysis, University of Gent, Belgium

Abstract. We deal with a mathematical model for the magnetic field induced in an electrical machine, to evaluate the magnetic iron losses. Herein the magnetic hysteresis and the eddy current effects are directly and simultaneously taken into account. The magnetic circuit is decomposed into magnetic and air gap network elements, the former showing a finite element structure. Although the model retains the essential features of a cumbersome 3D problem, a relatively simple algorithm may be developed.

1. INTRODUCTION

In the design of an electrical machine, an accurate computation of the magnetic field clearly is essential for a realistic prediction of the performance characteristics of this machine. Moreover, as the iron losses are a non-negligible portion of the total losses of a machine, the magnetic field computations should account for the precise features of the magnetic material used.

The iron losses consist, at the macroscopic level, of two components, viz. the magnetic hysteresis loss and the eddy current loss.

The hysteresis loss is connected with the fact that the relationship between the magnetic induction $\underline{B}$ and the magnetic field $\underline{H}$ in the material depends on the history of the magnetic field. The most popular hysteresis model, at the macroscopic level, used to describe this complex relationship is the Preisach model [1], and to a less extent, the Stoner-Wohlfarth model [3] and the Jiles-Atherton model [5]. In the Preisach model the basic properties of the magnetic materials are preserved.

The eddy current loss is caused by the currents induced in the magnetic material by the time varying magnetic induction. Indeed, the time varying flux ϕ, enclosed by each fictive loop in the magnetic material, induces an electromotive force from which, due to the electric conductivity σ of the magnetic material, eddy currents are generated. In the literature, different 3D descriptions of the eddy currents in electric machines may be found, however neglecting hysteresis effects [9],[10].

In our model, described below, the coupled hysteresis and eddy current effects have been taken into account in a more direct way than in the a posteriori technique, used in [2]. Although the model retains the essential features of a cumbersome 3D time dependent problem, it will lead to a straightforward algorithm, attractive for practical use. The effectivenes of the proposed model is illustrated by numerical results.

2. MATHEMATICAL FORMULATION

In the proposed dynamic model for the magnetisation of the steel laminations in electrical machines, the magnetic circuit is divided into individual *network elements*. In each network element the magnetic field is calculated taking the Preisach model into account. The network elements are interconnected by N_l *fundamental loops*. To each fundamental loop as well as to each network element an orientation is assigned similarly as in classical network analysis.

In [6], Ostovic presented a decomposition technique to calculate the static magnetic fields in saturated electrical machines, neglecting however the hysteresis effect. Inspired by this approach we propose a magnetic network model where the magnetic circuit is decomposed in N_m magnetic network elements and N_a air gap elements. The decomposition in the yz-plane (parallel to the laminations) corresponds to the uniformity of the magnetic field in each magnetic subregion with respect to y and z, while (in contrast with [6]) H varies there with x. In an air gap element, however, H is uniform with x.

To obtain an elegant dynamic model for the magnetic field, it is essential to directly include a discrete variational version of the local field problem of each magnetic network element. In this way we obtain a transparant system of first order nonlineair differential equations with respect to the time variable for the magnetic fields in all network elements and for all fundamental loop fluxes.

2.1 Field equations for the magnetic network elements

A single network element of length l, width w and tickness $2d$ is considered, the local axis being shown in Fig 1. Throughout the sheet, assumed to be isotropic, the time dependent total flux ϕ flows in the z-direction, and hence the magnetic field takes the form $\underline{H}=H\underline{e}_z$. As $d<<w$, by eliminating of edge effects, H may be taken to vary in the x-direction only. Recalling that the current density $\underline{J}=J\underline{e}_y$ is related to the electrical field $\underline{E}=E\underline{e}_y$ by $\underline{J}=\sigma\underline{E}$, where σ is the electric conductivity, and eliminating subsequently $\underline{E}$ and $\underline{J}$ from the relevant Maxwell-equations, the governing differential equation (DE) and boundary conditions (BC's) for H respectively are found to be:

$$\frac{\delta^2}{\delta x^2}H(x,t) = \sigma.\mu_d(x,H,H^{past}(x,t)).\frac{\delta}{\delta t}H(x,t), \qquad 0 < x < d,\ t > 0 \qquad (1)$$

$$\frac{\delta}{\delta x}H(0,t) = 0\,, \qquad \frac{\delta}{\delta x}H(d,t) = \frac{\sigma}{2w}\,\frac{d\phi}{dt}\,, \qquad t > 0 \qquad (2)$$

In the DE we also have taken into account the definition of the magnetic differential permeability μ_d of the material, i.e. $\mu_d = \delta B/\delta H$, with B the magnetic induction.The initial condition (I.C.), corresponding to the demagnetized state of the material, is:

$$H(x,0) = 0 \qquad\qquad 0 < x < d \qquad\qquad (3)$$

In (1), H^{past}represents the memory properties of the material, as described by the Preisach model. Here the material is assumed to consist of small dipoles, each of them being characterized by an hysteresis loop which is rectangular as shown in Fig 2. The characteristic parameters h_c and h_m of the dipoles are distributed statistically according to :

$$\gamma(h_c,h_m) = C_1.h_c^n.e^{-\alpha h_c}.e^{-\beta|h_m|} + \frac{C_2}{\cosh^2(C_3|h_m|)}.\delta(h_c)\,, \quad h_c \geq 0 \qquad (4)$$

The two terms at the right hand side correspond to the irreversible and the reversible feature of the material respectively. C_1, C_2, C_3, n, α and β are fitting parameters determined from experimental material identification.

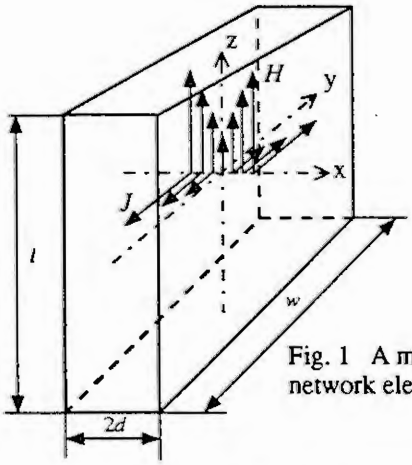


Fig. 1 A magnetic network element

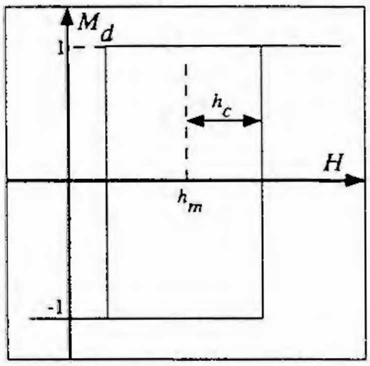


Fig. 2 One dipole

The irreversible part in the distribution function above is similar to the one proposed in [1]. However in [1], the reversible feature of the steel laminations is taken identical with the one of air, leading to a simplified reversible term independent of h_m.
Characteric for the material is the memory property, depictured schematically in Fig. 3-4 (for a fixed value of x);
- when, for t increasing, H increases [decreases], the differential permeability μ_d depends on the largest minimum [smallest maximum] of H in the past, kept in memory.
- when H becomes larger [smaller] than the last maximum [minimum], both the last maximum and minimum are eliminated from the memory (and hence the previous couple of largest minimum and smallest maximum substitute the eliminated couple).

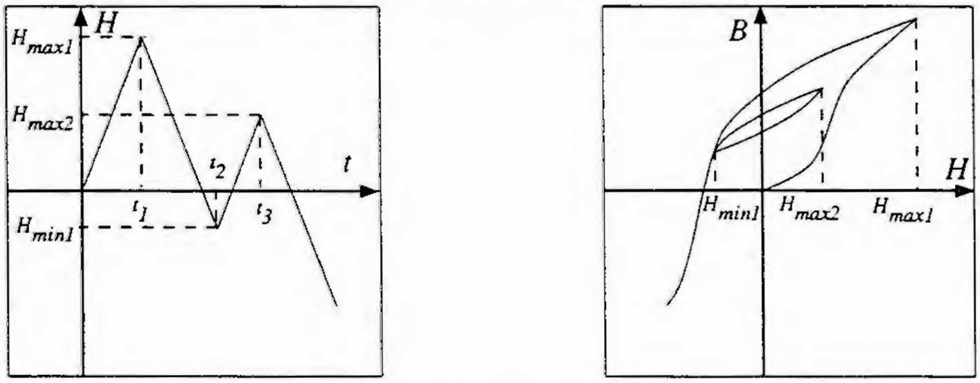


Fig. 3-4 The memory property of the magnetic material

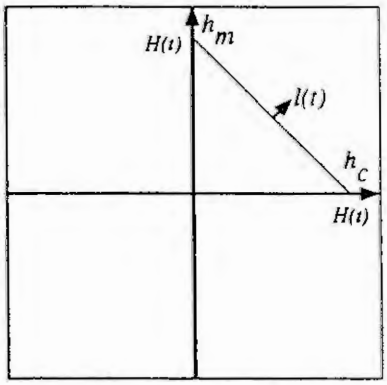


Fig. 5 The segment $l(t)$ when there are no extremes

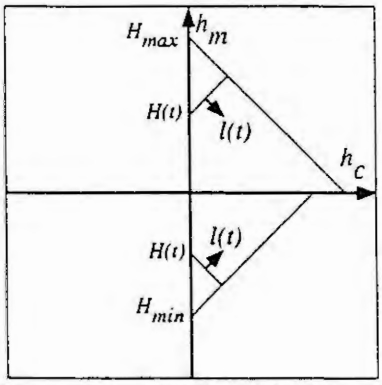


Fig. 6 The segment $l(t)$ when H increases [decreases] and there is a minimum [maximum] in the memory

The Preisach model essentially maintains this memory property:

- when there are no extremes, μ_d at time t (for a fixed value of x) is defined by the line integral of $\gamma(h_c,h_m)$ over the segment l(t) in Fig.5:

$$sign(\tfrac{\delta H}{\delta t}).h_c + h_m = H(t), \qquad 0 < h_c < H(t) \qquad (5)$$

- when there are extremes in the memory, μ_d is defined in a similar way, the segment in Fig. 6 now being:

$$sign(\tfrac{\delta H}{\delta t}).h_c + h_m = H(t), \qquad 0 < h_c < \tfrac{1}{2}|H(t) - H_{ext}| \qquad (6)$$

where H_{ext} stands for the last H_{min}, respectively H_{max} when H increases, resp. decreases. This complex memory behaviour of the material results in the discontinuity of the differential permeability μ_d with respect to t and in its strong nonlinear dependence on H. The discontinuity occurs when the magnetic field H reaches an extremum or when a maximum and the corresponding minimum are eliminated from the memory.

To discretize the boundary value problem (BVP) above with respect to the space variable (continuous t-dependence) we use a finite element method (FEM) with n elements and quadratic interpolation functions. We are led to a system of the type

$$[M]\tfrac{d}{dt}[H] + [K][H] = [F] \qquad (7)$$

Here $[H(t)]^T = [H_1(t),H_2(t),H_3(t),\ldots,H_{2n+1}(t)]^T$ is the vector of unknown vertex values of the magnetic field and K and F are the stiffness and force matrices respectively, the last one corresponding to the inhomogeneous Neumann B.C.. The mass matrix M is given by :

$$M_{ij} = \int_0^d \sigma\hat{\mu}_d(x,t).N_i(x)N_j(x)dx, \qquad i \;\; and \;\; j = 1,\ldots,2n+1 \qquad (7')$$

N_i is the piecewise quadratic FE-basis function corresponding to the node i and

$$\hat{\mu}_d(x,t) = \mu_d(x_{2k},H(x_{2k},t),H^{past}(x_{2k},t)), \qquad k = 1,\ldots,n; \qquad x_{2k-1} < x < x_{2k+1}$$

2.2 Field equation for the air gap element

For an air gap element the usual magnetic material property, corresponding to zero conductivity and (constant) permeability μ_0, is described by :

$$H(t) = \frac{\phi(t)}{2dw(t)}\frac{1}{\mu_0} \qquad (8)$$

Here H, ϕ, d and w have a similar meaning as in section 2.1. Eqn. (8) substitutes the system (7) for an air gap element.

2.3.Fundamental Loop Equations

The interaction between the network elements considered above is given by the N_l fundamental loop equations, [7],

$$\sum_{i\in\aleph_j} H^{(i)}(x = d,t).l^{(i)} = S^{(j)}(t), \qquad t > 0, \qquad j = 1,\ldots,N_l \qquad (9)$$

Here $H^{(i)}(x = d,t).l^{(i)}$ represents the drop of the magnetic motoric force (mmf) over the element i, $i \in \aleph_j = \left\{r = 1,\ldots,N_m + N_a \;\middle|\; element\ r\ belongs\ to\ loop\ j\right\}$.

The sum of these drops results from the mmf source $S^{(j)}(t)$ in the fundamental loop j.

2.4. The Connected Model

Crucial in our method is the assembling of the N_m magnetic element problems of type (7) and N_a air gap element equations (8) with the N_l loop equations (9) to a system of first order DE of the form:

$$\begin{bmatrix} \hat{K} & 0 & 0 \\ 0 & I & U \\ L_m & L_a & 0 \end{bmatrix} \begin{bmatrix} \hat{H}_m \\ \hat{H}_a \\ \hat{\Phi} \end{bmatrix} + \begin{bmatrix} \hat{M} & 0 & V \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \frac{\mathrm{d}}{\mathrm{dt}} \begin{bmatrix} \hat{H}_m \\ \hat{H}_a \\ \hat{\Phi} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \hat{S} \end{bmatrix} \tag{10}$$

Here $\hat{H}_m = [H_1^{(1)},\ldots,H_{2n+1}^{(1)};\ldots;H_1^{(N_m)},\ldots,H_{2n+1}^{(N_m)}]^T$, $\hat{H}_a = [H^{(N_m+1)},\ldots,H^{(N_m+N_a)}]^T$ and $\hat{\Phi} = [\Phi_1,\ldots,\Phi_{N_l}]^T$ represent the vector of the $(2n+1)N_m$ unknown vertex values of the magnetic fields in the N_m magnetic network elements, of the N_a unknown (uniform) magnetic fields in the N_a air gap elements and of the N_l fundamental loop fluxes respectively. We recall that the flux ϕ through a network element is the algebraic sum of the respective fluxes Φ through the fundamental loops sharing this element.

The square matrices $\hat{K}$ and $\hat{M}$ have a diagonal block structure, the constituting N_m uncoupled matrices being of the type K and M, entering (7).

L_m [V] has an horizontal [vertical] block structure, each of the N_m constituting N_l x $(2n+1)$, [$(2n+1)$ x N_l] submatrices showing nonzero elements only at one or two entries of the $(2n+1)$th column [row]. On the column [row] of L_m [V] corresponding to the nonhomogeneous Neumann BC for the magnetic network element k a nonzero entry is found at the row [column] j, $1 \leq j \leq N_l$,when $k \in \aleph_j$ .This value is l_k [$-\sigma/2dw_k$] when the orientations of the element k and the loop j coincide and is $-l_k$ [$\sigma/2dw_k$] in the opposite case. Here l_k and w_k are the length and the width respectively of the k-th magnetic network element.

L_a is an N_l x N_a matrix, the p-th column of which, corresponding to the p-th air gap element, $1 \leq p \leq N_a$, is constructed similarly as the nonzero columns of L_m. U is a N_a x N_l matrix, the p-th row of which, corresponding to the p-th air gap element, $1 \leq p \leq N_a$,is constructed similarly as the nonzero rows of V, however with σ replaced by $1/\mu_0$.

The column matrix $\hat{S}$ has the components $S^{(j)}(t)$, $1 \leq j \leq N_l$,entering (9). Finally I is the unit matrix of dimension N_a while 0 stands for the appropriate zero matrix. The I.C.'s for the system (10) are:

$$\hat{H}_m(0) = \hat{H}_a(0) = \hat{\Phi}(0) = 0 \tag{11}$$

according to the demagnetized state of the magnetic network elements (including (3) and zero fluxes for these elements), the network topology (interconnection of the air gap and magnetic network elements resulting in zero fluxes through the former elements) and (8).

It must be emphasized that the number of equations in the continuous time model (10) of the magnetic circuit is only piecewise constant in time, while also the width $w(t)$ of the air gap elements entering (8) may vary with time. Indeed, the number and the width of the air gap elements between the rotor and the stator in the electrical machine depend on the time varying rotor position.

An air gap element between a moving rotor tooth R and a fixed stator tooth S, see Fig. 7, occurs when the angle θ enclosed by their axes is in the range $[-\theta_c,\theta_c]$, while the variation with θ of the width *w* ,entering (8), typically has the form shown in Fig. 8 . The dependence of *w* on θ and, in particular, the critical value θ_c above which *w* is negligibly small, is determined from (8) by preliminary local magnetic field calculations, see e.g. [6] and [8]. In the former reference an analytical expression for *w* versus θ is proposed. In principle, the angle θ of Fig. 7, dealing with a single couple R-S, determines the configuration of all N_r rotor teeth relative to the N_s stator teeth. However, in practice, the total number of the air gap elements between stator and rotor as well as their respective width is found at any time t by relating, for each of the N_rxN_s couples of a rotor tooth and a stator tooth, the angle θ with respect to the critical interval $[-\theta_c, \theta_c]$.

This feature of the dynamic model must be properly taken into account in the system (10). In particular, when a new air gap element is created at a time t, the original fundamental loop is splitted into two new ones, i and j, say. According to (8), w=0 implies $\phi=\Phi_i-\Phi_j=0$. Hence, the fluxes at time t through the new fundamental loops have to be equal. Moreover this common value must be the loopflux through the original fundamental loop in order to maintain unchanged the value of the fluxes at time t in the surrounding network elements. A dual situation occurs when an air gap element is annihilated.

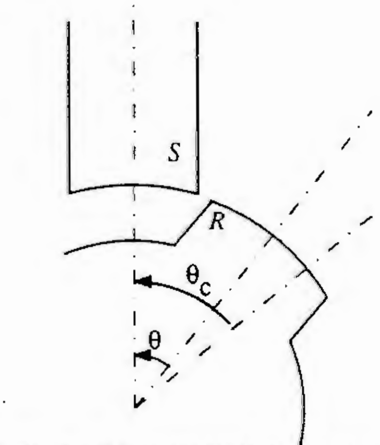


Fig. 7 The couple stator tooth S and rotor tooth R

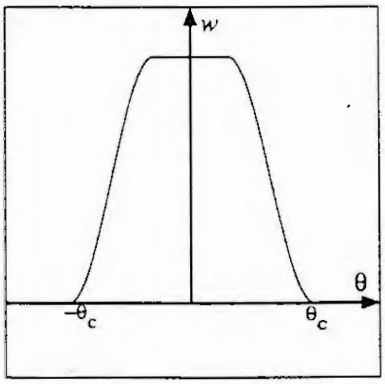


Fig. 8 The dependence of w on θ

3.THE ELECTROMAGNETIC LOSSES

The magnetic field in the magnetic network elements leads to the eddy current losses and the hysteresis losses in one laminate during a time interval $[T_1,T_2]$ according to the well known formula, see e.g. [4].

$$P_{ec} = \sum_{N_m} \int_{T_1}^{T_2} \left(\int_{\Omega} \frac{J^2}{\sigma}\, dv \right) dt \quad ; \quad J = \frac{\delta}{\delta x} H(x,t) \tag{12}$$

$$P_h = \sum_{N_m} \int_{\Omega} \left(\int_{H(x,T_1)}^{H(x,T_2)} \mu_d\, H dH \right) dv \tag{13}$$

Here Ω stands for the domain of a specific magnetic network element and the summation runs over all such network elements.

4. CONCLUDING REMARKS - RESULTS

In the mathematical model for the electromagnetic losses, described above, the magnetic circuit of the electrical machine is decomposed in (magnetic and air gap) network elements, interconnected through fundamental loop fluxes. The model takes into account both eddy current and hysteresis effects in the magnetic network elements, corresponding respectivily to the nonzero value of the conductivity σ and to the specific form of the differential permeability μ_d, the latter showing a nonlinear dependence on H as well as discontinuities in time. Moreover the time varying position of the rotor relative to the stator is also included in the model by means of the interconnecting air gap elements, the number and width of which may depend on time, as discussed at the end of section 2.4.

The mathematical model has been reformulated in (10)-(11) as an initial value problem for a system of first order nonlinear differential equations with respect to t. Clearly, a suitable numerical model has to be constructed to solve this much involved problem. We developed a modified Crank-Nicholson algorithm. Over each time step, we use for the magnetic field in the magnetic network elements a weighted average of the values of H at the end points of the time interval. Hereby the weight coefficients depend on the magnetic network element under consideration and its local space variable in the x-direction, as well as on the time level itself. Thus we could properly account for the hysteresis phenomena, in particular for the discontinuity of μ_d in time.

The algebraic system at each time point may be rearranged so as to led to a subsystem containing the unknown fundamental loop fluxes only. Correspondingly, N_m decoupled algebraic systems for the magnetic fields in the N_m individual magnetic network elements may be identified. At the other hand, the magnetic fields in the air gap elements directly follow from (8).

The validity of the mathematical model has been tested by the evaluation of the switched reluctance motor with 6 stator and 4 rotor teeth, shown in Fig. 9.

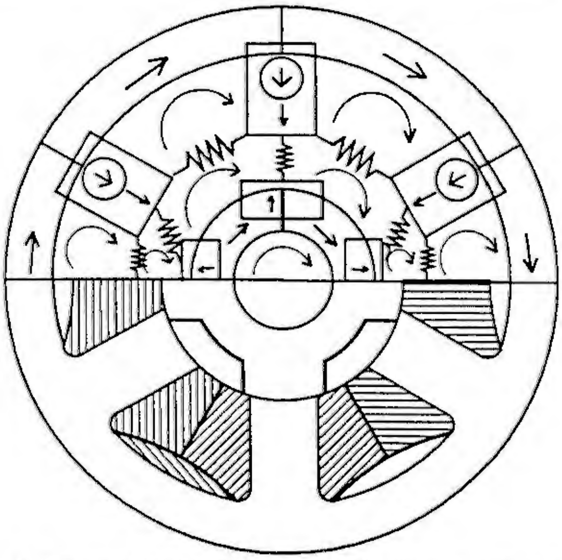


Fig. 9 The model of the 6-4 switched reluctance motor

(←)	mmf-source	[←]	magnetic network element
↷	fundamental loop	▥	winding
		⁓	air gap element

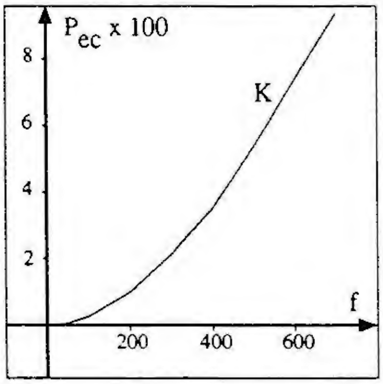


Fig. 10 The eddy current losses versus the frequency f

The material is characterised by the value σ=5,56.10^6[S/m^2] for the electric conductivity and by the Preisach function (4) with the following parameters: C_1=8,8.10^{-16}; C_2=1,350.10^{-3}; C_3=1,227.10^{-2}; n=8; α=0,104 and β=0,022.

The geometric characteristics measured in meter are: width of a stator [rotor] tooth: 0,017 [0,020]; width [diameter] of the air gap: 0,0003[0,06]; external [internal] diameter stator yoke: 0,135 [0,113]; external [internal] diameter of the rotor yoke: 0,045 [0,025].

Restricting ourselves to one relevant numerical result, the curve K in Fig. 10, shows the eddy current losses in one laminate versus the frequency f of the electric current in the windings (f=1/T, T=period). The parabolic part of K found in the range 0 < f < 250 Hrz confirms the well known physical behaviour of this type of magnetic losses [2], while the monotonously decreasing curvature of K for higher frequencies agrees with the skin effects in the laminations, becoming more and more important for f increasing.

5. ACKNOWLEDGEMENTS

We thank Dr. J. Cnops and Ir. D. Philips of the Laboratory of Industrial Electricity of the University of Gent for fruitful discussions. Partly, the work was sponsored by SIDMAR N.V., partly by the IUAP-project nr. 51. Finally the Belgian National Science Foundation is gratefully acknowledged by one of us (R.V.K.).

6. REFERENCES

[1]Akpinar, A.S., Computation of the quasi-stationary magnetic field in steel laminations in presence of hysteresis. Electric machines and power systems, 18, No. 4-5 (1990), 429-447.

[2]Bertotti, G. et al., An Improved Estimation of Iron Losses in Rotating Electrical Machines. IEEE transactions on magnetics, 27, No 6 (1991), 5007-5009.

[3]Friedman, G., Mayergoyz, I.D.,Stoner Wohlfarth Hysteresis Model with Stochastic Input as a Model of Viscosity in Magnetic Materials. IEEE transacations on magnetics, 28, No 5 (1992), 2262-2264.

[4]Grellet, G., Pertes dans les machines tournantes, Technique de l'ingenieur, Vol. D3 II. Schiltigheim, Strasbourg, 1989.

[5]Jiles, D.C. , Atherton, D.L.,Ferromagnetic Hysteresis. IEEE transactions on magnetics, 19, No 5 (1983), 2183-2185.

[6]Ostovic, V., Dynamics of Satured Electric machines. Springer Verslag, New York, 1989.

[7]Philips, D.,Dupre, L.,Van Keer, R., Computation of Magnetic Fields in Electric Machinery using a Coupled Finite Element-Network Element Model. In: H. Heiliö (Ed.), Proceedings of the 5th European Conference in Mathematics in Industry. Kluwer Academic Publishers, Dordrecht, 1991.

[8]Philips, D.,Loukipoudis, E.,Dupre, L.,Melkebeek, J., A Parametric Design Method for Computer - aided Design of Electric Machinery. J. of Engineering Design, 3 ,No. 3 (1992), 255-267.

[9]Pillsbury, R.D. Jr., A Three Dimensional Eddy Current Formulation using two Potentials: the Magnetic Vector Potential and Total Magnetic Scalar Potential. IEEE transactions on magnetics, 19, No 6 (1983), 2284-2287.

[10]Van Welij, J.S.,Calculation of Eddy Currents in Terms of H on Hexahedra. IEEE transactions on magnetics, 21, No. 6 (1985), 2239-2241.

USE OF CATEGORIAL THEORY FOR THE FORMATION OF ALGORITHM REGARDING SYNTHESIS OF MATHEMATICAL MODELS

H.P. ADRONATIY, V.D. DMITRIENKO, N.I. KORSOUNOV

KHARKOV POLITECH. INSTITUTE
Frunze Str. 21, UKR-310002 KHARKOV, UKRAINE

The method of self-organisation of mathematical models evolution modelling and genetic algorithms help in completing synthesis of mathematical models of different objects under unconditional unknown conditions. Algorithms shound on the basis of each method and uses the theory of biological evolution and adaptation up to this moment they had been developing independently.

In the present times in the well known algorithms polynomials, rational, trigonometric and exponential functions are used. One of my well known algorithm is MGUA in almost every case we work with MGUA to concrete universal algebra.

An attempt had been made to use modern algebra in evolutional modelling. Then had been presented A(M,D) algebra, where M - SET possible interchange $S_k^{i\alpha} S_l^{i\beta}$ resulting automatic $B_i = < X^i, Y^i, S^i, \varphi_y^i, \varphi_s^i >$ from the condition $S_k^i \in S^i$ to $S_l^i \in S^i$ when given final Entry X^i and Exit Y^i alfabits $(\alpha \in X^i, \beta \in Y^i)$; i, $(i = \overline{1,n})$ - index, pointing out undendependerec to i^{-th} automat; S^i - final set condition; $y_j^i = \varphi_y^i (X_j^i, S_j^i)$ - Exit functions; $S_{j+1}^i = \varphi_s^i (x_j^i, S_j^i)$ - interchange functions; y_j^i, x_j^i, S_j^i - relative exit symbol, entry symbol and i^{-th} condition of find automat B_i at moment t_j, where j=1,2,... . Set D contains "+", "⊗", "." operations.

Operation "+" helps with the help of function of the form $S_k^{i\alpha} S_l^{i\beta}$ in analyzing series of interchanging automats from one condition to another. In the set M transitional final automat $B = < X, Y, S, \varphi_y, \varphi_s >$ with the given starting condition and given first series of symbols give the characteristics

$$S_q^\alpha S_j^\mu + S_k^\beta S_l^\eta = S_k^\beta S_l^\eta + S_q^\alpha S_j^\mu, \qquad (1)$$

$$S_q^\alpha\ S_j^\eta + S_q^\alpha\ S_j^\eta = S_q^\alpha\ S_j^\eta, \qquad (2)$$

$$(\ S_q^\alpha\ S_j^\eta + S_k^\beta\ S_l^\mu\ ) + S_m^\gamma\ S_n^\sigma = S_q^\alpha\ S_j^\eta + (\ S_k^\beta\ S_l^\mu\ + S_m^\gamma\ S_n^\sigma) \qquad (3)$$

Characteristic (1) makes sure commutation operation and characteristics (2), (3) its potentness and associativeness.

Operation "⊗" allows multiplication of one number form like $S_q^\alpha S_j^\mu$ and move that one "·" forms. Operation "·" allows rescamehing forms like $S_q^\alpha S_j^\mu$ with coefficients from the set of whole numbers.

Operation "+" and " ⊗" can be understood like operations of addition and multiplication of algebraic forms and operation "." like multiplication of algebraic forms on some numbers. A straight relation is seen between algebra A(M,D) and algebra of polynoms on algebra of unperiodical functions, which are used in MGUA. To exclude this relation - ship while theory of evolutionary modelling is developing, with a definite destination the results of MGUA must be used.

Use of theory of category allows forming process of transit of algorithms from algebra to algebra and is capable of giving new algorithms in concrete algebra.

Definition 1. Pair (M,D), where M - set of elements, which is not empty, D - set (may be empty) operation on M, is called UNIVERSAL ALGEBRA or simply ALGEBRA.

Set D of operations can be analyzed exclusively of algebra. More definitely we can talk about operational symbols, which are called SIGNATORS. If to even operational symbol from D is put a respective operation on set M, then algebra SIGNATOR comes up.

Definition 2. If A and B - algebra SIGNATORS D, then reflection of algebra f: A → B is called GOMOMORFIZM, is for each zero operation $d_o \in D$, $f\ (\ d_o(A)) = d_o(B)$ and for any n^{th} operation $d_n \in D$ and any $a_1, \ldots, a_n \in A$

$$f\ (\ d_n(\ a_1, \ \ldots\ , a_n)) = d_n\ (\ f\ (a_1),\ \ldots\ ,\ f\ (a_n)).$$

As the product of gomomorfizm algebra is called gomomorfizm, then this capability helps in analysing category K_o universal algebra of one and the same signator. Objects of this category are - morfizm - gomomorfizm algebra, and products of morfizm - products of gomomorfizm algebra.

Example. Analysing algebra $R_1(X_1, D_1)$ polynomials and algebra $A(M,D)$, where $X_1 = \{ 0, 1, x_1, x_2, \dots , x_n\}$, $M = \{ 0, 1, S_1^{\circ\alpha_1} S_1^{\beta_1}, \dots , S_m^{\alpha_k} S_m^{\beta_n} \}$. Signators $D_1 = \{ d_o, d_1, \cdot , +, / \}$, $D = \{ d_o, d_1, \cdot, +, \circledast \}$ consists of zero operations d_o, d_1, d_o, d_1, aslracting zero and one,respectively in sets X_1, and M, singular operations "$\cdot$", "$\cdot$", multiplication of elements of sets in real number and binary operations of addition and division of elements of sets X_1 and addition and multiplication of elements of sets M. For sure, there is isomorphism of algebra $f: R_1 \to A$, because of this it is not difficult to have algorithms from algebra R_1 transferred to algebra A and vis versa. For example, in algebra R_1 it is known algorithm MGUA with not linear private writings, in which in the first place of selection build private writings

$$y_k^1 = a_o^1 + a_1^1 x_i + a_2^1 x_j + a_3^1 x_i / x_j , \qquad (4)$$

where $k = \overline{1, C_n^2}$; $i = \overline{1, (n-1)}$; $j = \overline{2, n}$.

In the later places of selection private writing happen to be in the form

$$y_m^r = a_o^r + a_1^r y_i^{r-1} + a_2^r y_j^{r-1} + a_3^r y_i^{r-1} / \, y_j^{r-1} , \qquad (5)$$

where $m = \overline{1, C_q^2}$;$i = \overline{1, (q-1)}$; $j = \overline{2, q}$; q - number got from r^{-th} place of selection of better private writing $(r-1)^{-th}$ place.

Putting in formula (4) in place of elements x_i, x_j of set X_1 elements of set M and replacing operations of algebra R_1 by operations of algebra A we get respective algorithms in algebra A. In the first place of selection under common conditions we have private writing in the form of

$$y_k^1 = S_1^{o\alpha_p} S_j^{\beta_q} + S_j^{\alpha_w} S_l^{\beta_g} + S_1^{o\alpha_p} S_j^{\beta_q} \quad S_j^{\alpha_w} S_l^{\beta_g} \qquad (6)$$

where $j, l = \overline{1, m}$; $p, w = \overline{1, k}$; $q, g = \overline{1, n}$; m - max. possible number having synthesizing automats; k, n- number of symbols irrespective of entry and exit alfabits.

In the same way we get private writings, respective to private writings of (5).

This way, if we have some sets of one type algebra, then with the help of gomomorphizm algebra algorithms MGUA on evolutionary modelling from one algebra may formaly transfer to another algebra.

Polynomials, which is used while solving a definite problem which is related to algorithm of self-organization, can be analysed like objects of some category of polynomials.

We can see the category KDP - category of discrete polynomials with given accuracy and category RKA which have known results of automatics, isomorfh category of KDP.

It is not difficult to make functures between analysed categories, in exclusion to univalent functures from category of polynomials in category PKA. With the help of functures any algorithm MGUA in the set of polynomials can be put in relation to synthesis of algorithm in the set of resulting automats. It is known, that under definite conditions we can analyze not exclusive functures but their categories.

Use of the theory of category helped in theoretical basing and combining intetive transfers of algorithms in evolutionary methods of modelling from one algebra to another and presenting the whole sector of new algorithms in the way of algebra. Some of there known algorithm are used while projective instruments of measuring technique, in exclusion to resulting automats, querning temporary series, for knowing and classifying signals.

K-CHARAKTER DIFFERENTIAL CALCULUS AND ITS USE FOR ELECTRONIC CIRCUIT RESEARCH

Korsounov N.I., Dmitrienko V.D., Leonov S.Yu.

Kharkov Politekchnikal Institute

Frunze str., 21, 310002, Kharkov, Ukraine

Modern evolution level of architecture of calculating and controling system, the increase of their fast-acting made it necessary to use gualitativly methods for solving time and construction coordination problems in order to pass the information between separate blocks of digital systems, which elementary basis is VLSI.

There are various methods, which allow to model elektronic equipment and they have various accuracy and complexity. This is, for example, the method of modeling, based on the theory of digital automators, the method of elements description using ordinary differential equations, different methods of modeling, also based on multiciphered functions, as well as the description of digital systems with the help of boolean differential calculus. Defects of this methods when they apply for modeling real digital equipment are either excessive expence of calculation, which don't give wide possibility of their practical aplication, or low accurasy and impossibility to describe dynamic processes in moments of switching logical elements, that is necessary at up-to-date level of modeling VLSI and equipment, which consists of VLSI for getting necessary accurasy of results.

In connection with this the question of creating the methods of modeling became of present interest. These methods are to be rather simple in use and don't need excessive calculations, but at the same time they are to have high accuracy when receiving signal meanings in every real moment of time.

One of such methods is modeling, which is based on the method of K-charakter differential calculus.

In frame of K-charakter differential calculus analogies of differential and integral operators of classic differential calculus are introduced, which allow to describe modifications K-charakter variables. By that the relation of function ΔF increase to the single increase of the independent variable Δt, that calls the function increase, is called the derivative of K-charakter function $F(t)$ by independent variable t, when $t = t_i$, $t \in [0,T]$ and it can be determined by the following expression:

$$\dot{F}(t_i) = \frac{d\iota(F(t_i))}{d\iota(t_i)} = \frac{F(t_i + \Delta t) \ominus F(t_i)}{\Delta t}, \qquad (1)$$

where $d\iota(F(t_i)) /\!\!/ d\iota(t_i)$ – differential-K-charakter operator for function $F(t_i)$, $\ominus$ and $/\!\!/$ – symbols of subtraction and division operations by module K, $\Delta t = 1$.

In case $F(t_i)$ is prototype function, then the integral of K-charakter function defined with precision to arbitrary constant and appears the indefinite integral K-charakter function:

$$F(t_i) = \int f(t_i)\, d\iota t_i ,$$

where arbitrary K-charakter constant is not obvious.

With the help of the K-charakter differential calculus it is possible to describe the dinamic of digital systems both on the "macroapproach" level when system dinamic is entirely analysed and on the "microapproach" level, when knowing the structure of the system, we can explore the dinamic on the level of the events in system elements. That modeling allows considerably full to model the dinamic of transitional processes of switching logical elements of numerical equipment, and also to take into account the influence of crossroads connections, which were called by corinductive and mutual capacity of connectoins both inside the logical elements and between separate bloks of digital systems.

Using the K-charakter differential calculus in order to research the work of VLSI and electronic equipment, which was done on VLSI on "microapproach" level, their functioning is described by the system of K-charakter differential equations of the following kind:

$$\frac{d_i U_{out_j}(t_i)}{d_i t_i} = F_j(U_{in_p}(t_i)) \ominus F_{out_j}(t_{i-1}),\ \text{by}\ F_{out}(t_0)=Const, \qquad (2)$$

where $d_i U_{out_j}(t_i) / d_i t_i$ - the meaning of derivative of the exit signal for j's element at the moment of time t_i; $F_j(U_{in_p}(t_i))$- K-character function of the j's logical element at the moment of time t_i; $U_{in_p}(t_i)$, $(p = \overline{1,m})$ - the meaning of the entrance signals on entrance of the logical element; $F_{out_j}(t_{i-1})$ - the meaning of the K-charakter exit signal of the j's logical element at the moment of time t_{i-1}; i=0,1,2,... - natural number, which defines the meaning of the moment of time; $j = \overline{1,n}$ - the number of the logical element in modeling equipment; $p = \overline{1,m}$ - the quantity of entrances of the logical element.

The "macroapproach" is used when it is necessary to get more detail information about the characteristics of the numerical system. In this case it is necessary to make its modeling with registration the meanings of hindrances, which pointed by the active conductors in the passive ones. This hindrances are called by the presence of crossroads connections, in particular, mutual capacity and corinductivity of conductors. In this case it is necessary to solve the system of K-charakter differential equations in the time interval $[0, t_f]$:

$$\frac{d_i U_m}{d_i t} = \frac{U_m}{C} \sum_i \sum_j \sum_n \frac{C_{ij}}{\tau} \frac{d_i U_n}{d_i t} + \frac{1}{C} \sum_i \sum_j \sum_n \frac{M_{ij}}{R_{out_n}} \frac{d_i I_n}{d_i t} \pm$$

$$\pm \frac{1}{C} \sum_i \sum_j \sum_n M_{ij} C_{out_n} \frac{d_i^2 I_n}{d_i t^2}, \qquad (2)$$

where U_m - the quantity of the pointed hindrance in the m-th conductor, which is called by the voltage change in the n-th conductor from zero to digit or from digit to zero; $m = \overline{1,p}$, p - the number of passive conductors where the hindrance is analysed; U_n - the quantity of voltage change in the n-th active conductor, which appears to be the sourse of hindrance; $n = \overline{1,g}$, g - the numbers of active conductors, which influence are taken into account; i - the number of section in the n-th conductor, which exerted influence on m-th conductor sections; j - the number of m-th passive conductor sections, which is a receiver of points; C - the summary capacity of exit of the logical element - the source of signal and of entrance of thelogical element - the receiver of hindrance; $\tau = \tau(t)$ - the constant of time of transitional process for the time $[0,t_f]$, $\tau=(R_{in}R_{out})/(R_{in}+ R_{out})$; I_n - the quantity of the current change in the n-th active conductor, which is a source of point; C_{ij} и M_{ij} - the capacity of mutual connection and the corinductivity between i-th section of the n-th conductor and j-th section of the m-th conductor correspondingly; t_f- the long term of the impulse signal.

With the help of the K-charakter differential equations of kind (2) all basic elements of digital technique, basic logical elements AND, OR, NOT, the impulses formers, triggers, arithmetical equipment and ets. are described. Universal numerical methods of solving the systems of K-charakter differential equation are elaborated. Examples of equipment calculations descibed by the systems of K-charakter differential equations kind (2) and (3) shows, that such modeling allows to get more full quantitative and qualitative charakteristics, which reflects the physics of process with quick change of signals and, therefore, to appraise more reliable the working possibility of projected VLSI and also, elektronic equipment done on their basis.

DESIGNING CERMET HEATING ELEMENTS ON CONDITION OF THEIR THERMAL RELIABILITY

A.O.Kostikov, O.S.Tsakanyan

Institute for the Problems in Machinery of the Academy of Sciences of Ukraine
2/10 Pozharsky 310046 Kharkov Ukraine

The problem of developing reliable heaters on the base of cermet plate is raised. The principles of developing design systems for ceramic heaters and the thermal and mathematical models of the processes taking place therein are considered. The possibility of transition from a 3-D to 2-D mathematical model is substantiated, and its application limits under different constrictions are determined. The heating element design procedure is described.

INTRODUCTION

At the present time the heaters are made in the form of a ceramic (aluminium oxide) plate in which resistance elements which are a wolfram-molybdenum conductor are placed. Such a structure of the cermet heater possesses a number of valuable properties. It works at high temperatures (above 800°C). However it is broken to pieces at relatively small temperature gradients. Since the latter restrict the use of the heaters it is necessary to take measures to flatten temperature field of the heater so that the temperature gradient should not exceed a permissible value.

One of the uses of the heaters may be thermostating. As a rule in the thermostating systems the heater surface temperature should be the constant T_r.

Main factors having influence on the temperature distribution in the heater body are external effects (heat exchange with surroundings) and internal ones (resistance elements - internal heat sources). Influence of surroundings and system structure elements are considered as known ones when the heater is designed. Therefore the only way to influence the heater temperature field is to change the resistance element location.

Thus heater designing for thermostating systems can be reduced to solving the problem of creating a constant temperature in the heater body. Such a formulation of the problem is also good for the heaters used in heat devices when minimum temperature gradients in the heaters should be obtained. In the latter the required heater temperature T_r is determined from the heat balance equation

$$P = \int_S h\left(T_r - T_S\right) ds \tag{1}$$

where h - heat transfer coefficient, S - heater surface area, T_S - surroundings temperature, P - heater power.

1. The principles of developing design system for cermet heaters

When the heater is designed his construction (shape, dimensions, reinforcing, shape and dimensions of electrical leads), thermal properties (boundary conditions at the heater surface, heat conduction coefficient, required temperature) and electrical properties (voltage, specific resistance, maximum permissible current) are given. A resistance element is of a thin rectangular track form (Fig. 1). A heater electrical resistance modification is obtained by changing the tracks width and distance between them, these being changed discretely because of technological restrictions.

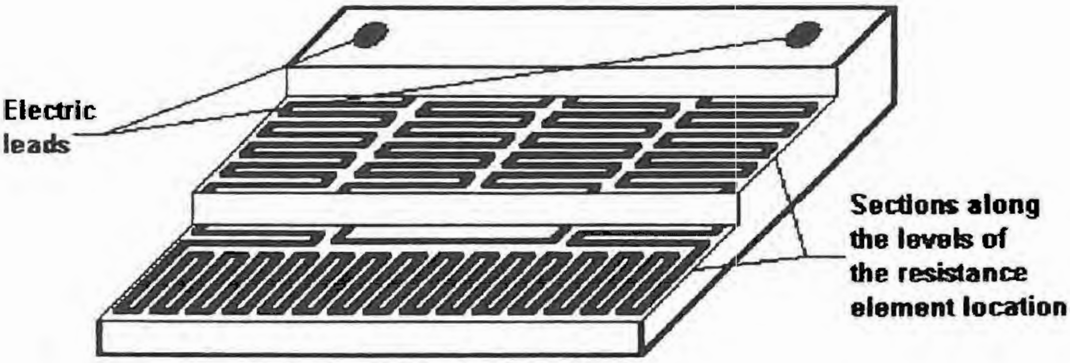

Electric leads
Sections along the levels of the resistance element location

Fig. 1. The heater.

A cermet heaters automatic designing procedure consists of four stages.

At the first stage a heater plate considered as 2-D region $\Omega=\{(x,y)\}$ is divided by N_1 subregions. In each subregion thermal properties (including heat source specific power) are considered as constants. Specific power of N_1 heat sources is determined by solving the inverse heat conduction problem. After that the heater temperature field $T_d(x,y)$ is determined by solving the heat conduction problem with the heat generation value already found.

Then the condition of agreement of this temperature with the required one

$$d(T_d, T_r) \le \Delta T_{perm} \tag{2}$$

is checked. In (2) ΔT_{perm} is a permissible difference between the temperature which was found and the required one, d is metric in one of the spaces of functions given in Ω.

If condition (2) is not satisfied the region Ω is divided by N_2 $(N_2>N_1)$ subregions and inverse heat transfer problem solving and agreement condition checking are repeated. The first stage is repeated until condition (2) is satisfied for some N_S.

At the second stage for each subregion the dimensions and shape of the resistance elements are determined so that their resistances correspond to the heat source specific power found at the first stage. Since in one subregion the heat source specific power is a constant the resistance elements are located with constant width and step in the subregion.

If resistance elements cannot be located at one level because of the technological restrictions they are placed at several levels along heater thickness. Maximum level number is defined by a minimum distance between levels and by heater thickness.

After locating the resistance elements condition (2) is checked for temperature T_d which is determined from the heat conduction problem solved for the heat source specific power corresponding to the found resistance elements. If this condition is not satisfied one may increase either the level number or that of the subregions thus returning to the first stage.

The third stage lies in determining the location of the resistance element levels. Since in a real heat device or a thermostate the heater surfaces are subjected to different thermal effects the temperature gradients resulting in plate destruction can increase along heater thickness. Therefore when locating the resistance element levels it is necessary to watch that maximum temperature gradient should not exceed permissible value.

The fourth stage consists in simulating the various situations associated with an inexactitude of a technical realization of the heater designed. For this purpose a deflection of the electrical and geometrical characteristics is set and its influence on the heater temperature distribution is determined.

Thus the cermet heaters designing procedure includes determining the characteristics of the optimum thermal regime of the heater and selecting the optimum location of the resistance elements in the heater body taking into consideration the restrictions of the technological and electrical characteristics.

2. Mathematical model of a thermal process taking place in the heater

It is generally necessary to solve the steady-state heat conduction problem for a cermet plate in the 3-D statement. One may reduce dimensionality of the problem neglecting a temperature variation along the heater thickness. The investigation of suiting the 2-D model is given below.

As an evaluation criterion we take relative variation of a temperature

$$\Theta = \frac{T(x,y,\delta/2) - T(x,y,0)}{T(x,y,0) - T_s} \leq \Theta_{perm} \tag{3}$$

where δ is the plate thickness.

The results of the investigation carried out for the plate 2 mm thick are given in Fig. 2. It shows a dependence of Θ on the heat transfer coefficient h at the heater surface and on the heat flux q flowing through it. The straight lines going out from the coordinate beginning are isotherms.

A family of isolines Θ=*const* is plotted at Fig. 2a for cases of location of resistance elements at one level. Having chosen some Θ_{perm} one can determine from the nomogram whether 2-D thermal model is suitable for given h and q.

The resistance elements placing at several levels, the region $\{(h,q): \Theta(h,q) < \Theta_{perm}\}$ is extended (see Fig. 2b).

The 2-D mathematical model of the thermal process taking place in the heater is given by the equation

$$\frac{\partial}{\partial x}\left(k_x \frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(k_y \frac{\partial T}{\partial y}\right) + \frac{h_1}{\delta}(T - T_1) + \frac{h_2}{\delta}(T - T_2) = -q_v \tag{4}$$

where $T(x,y)$ is a heater temperature, k_x and k_y are the thermal conductivities in x and y directions respectively, h_1 and h_2 are the heat transfer coefficients at the surfaces $z=0$ and $z=\delta$ respectively, T_1 and T_2 are the surroundings temperatures at these surfaces, q_v is a heat source specific power.

On the heater edges the boundary conditions of the third kind are given:

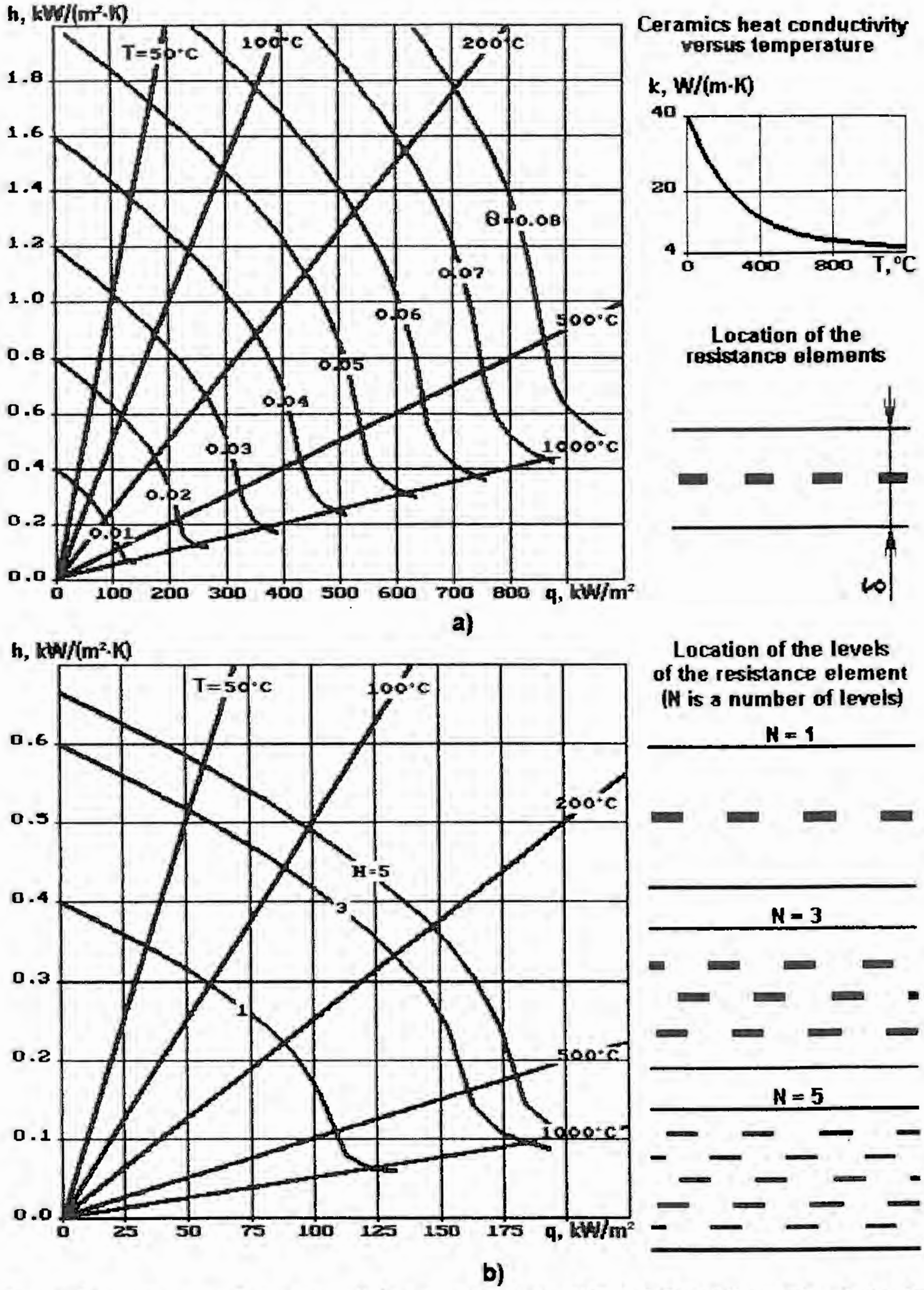
$$\left[k_n \frac{\partial T}{\partial n} + h(T - T_s)\right]_{Bound} = 0 \tag{5}$$

where n is a normal to an edge, k_n is the thermal conductivity in n-direction, h is the heat transfer coefficient, T_s is the surroundings temperature.

It should be noted that coefficients k_x, k_y, h_1, h_2, q_v, k_n, h, q do not depend on temperature since the heater temperature is almost a constant.

RESULTS

The suggested procedure of designing the cermet heaters allows determination of the heat source specific power and the dimensions, the shape and the location of the resistance elements and to carry out modelling of the possible thermal regimes of the heater work.



a)

b)

Fig. 2. Nomograms for determining a suitability of the 2-D mathematical model.

The procedure is suitable for designing heaters with inconstant temperature distribution. It is realized as a computer-aided design for IBM PC AT 386/387 or 486/487.

The procedure considered yields good results for a debugged technology of the heater manufacture and known conditions of their work.

MODELLING THE THERMAL STATE OF AN ELECTRONIC MODULE WITH A MICROCHANNAL SYSTEM OF FLUID COOLING

S.F. Lushpenko, Yu.M. Matsevity, O.S. Tsakanyan
Kharkov, Institute for Problems in Machinery of the Ukrainian Academy of Sciences

Abstract. A mathematical model and the method of modelling the thermal and hydraulic processes taking place in an electronic module equipped with a microchannel system of fluid cooling is considered. The results of calculated determining of the thermal and hydraulic parameters at different pressure drops between the distributor and collector of the cooling system are given.

INTRODUCTION

The development of mainframe high-speed computers goes along the lines of increasing the speed and working power of logical LSI. The maximal heat flow densities anticipated on the surface of an LSI chip shall exceed 80 W/cm². Therefore, to use such electronic elements it is necessary to employ methods of intensive heat removal. This paper deals with the constructive variants of conductive-fluid cooling of microassemblies and functional cells of electronic modules whose supporting structure is a ceramic board which should simultaneously accomodate the electrical connections and the system of hydraulic microchannels. Besides ceramic boards and variants of electronic modules on aluminium boards are considered. Their bodies have a system of parallel microchannels, and glass-clothbase laminate printed-circuit boards are bonded to both sides of the board. Such a construction of an electronic module has a high heat transfer capacity mainly due to the small resistance in the path of the heat flow from the base of the chip to the fluid.

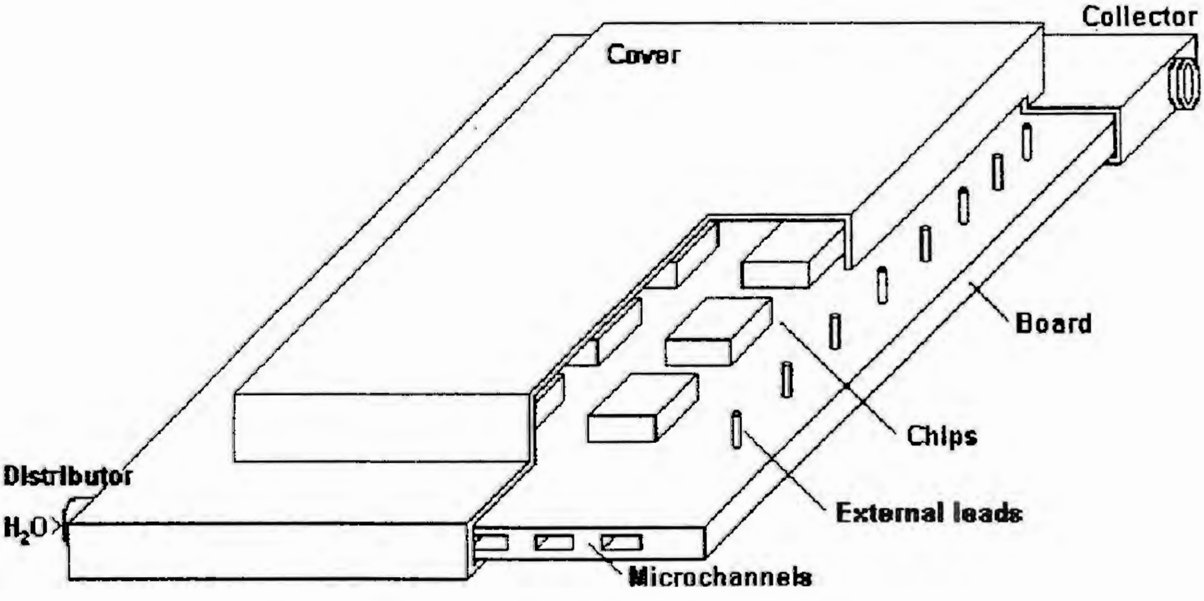


Fig. 1. Electronic module with a fluid cooling system

The construction shown in Fig.1 can be considered as a feasible construction of an electronic module both on a ceramic board and on a metal one. Such a construction of an electronic module possesses high potential capacities of heat removal. Thus, a channel 90 mm long with the cross-section area 0.6 mm^2 allows to remove the power 70 W at the pressure drop between the inlet of 20 kPa. Unfortunately, to use the potential capacities of the microchannel cooling system proves impossible due to technological constraints during manufacture of both ceramic boards and aluminium ones. In this case, when searching for a rational construction before designing an electronic module, there arises the problem of determining the constructive characteristics of an electronic module together with the hydraulic and thermal characteristics of its fluid cooling system. This problem may be solved by mathematical modelling of the hydraulics of the cooling channels and the thermal state of the electronic module at the established regularities of heat exchange on the surfaces of the microchannels and module housing.

Problem statement and calculation method

The thermal model of an electronic module is represented by a rectangular plate whose body has a system of regularly arranged microchannels located parallel to its end surfaces. According to the layout of components, on both sides of the plate there may be specified local surface heat sources occupying areas of rectangular shape. Heat exchange of the plate with the ambient medium is taken into account in the form of boundary conditions of the third kind.

The plate is assumed to be homogeneous, and its heat transfer is calculated by the formulas determining the effective heat transfer. The convective heat transfer coefficients in the microchannels are considered to be averaged over the whole surface, whereas the cooling agent temperature changes along the channel. To obtain the boundary conditions, a hydraulic calculation is made in the channels for the whole system of channels, as a result of which for each section of the hydraulic network the pressure drop, velocity and flow rate of the cooling agent become known. The geometric and hydraulic characteristics of the network sections are used as addresses which are accessed in the data base constructed according to the established heat exchange regularities for retreival therefrom of the convective heat transfer coefficients.

From the above-mentioned it follows that the determination of the thermal state of an electronic module is reduced to the solution of the heat transfer problem in the 3-dimensional statement, and to take into account the relationships between the hydraulic and thermal processes on the surfaces of the microchannels it is necessary to carry out a combined calculation of the thermal state of the boards and the hydraulic characteristics of the cooling network. This calculation is based on the solution of FED equations with the help of an iterative computational procedure allowing to specify during one iteration not only the board temperature field, but also all the quantities directly or indirectly depending on the surfase temperature of the microchannels.

Checking the mathematical model adequacy

To check the adequacy of the mathematical model describing the thermal state of an electronic module with a microchannel system of fluid cooling, an experimental investigation on a model of a multilayer ceramic board with the dimensions 118.6×101.3×2.9 mm carried out. The board has 18 channels with the length 118.6 mm, width 0.8 - 0.034 mm and height 0.5 - 0.035 mm. The distributor and collector are made of copper tubes brazed to the board ends. The area of the flow section of the collector and distributor was 7 mm^2. The surface of the board was fitted with immitators of chips having the area of contact with the board equal to 10×10 mm^2.

In the course of the experiment the temperature on the board surface along the 14th channel was measured, as well as the water temperature at the distributor inlet and collector outlet, and the pressure in the following points of the hydraulic network: at the distributor inlet, at the inlets and outlets of the 2nd and 14th cannels.

Comparison of the results of mathematical modelling with experimetally found temperatures was perfomed for different temperature regimes of the electronic module board.

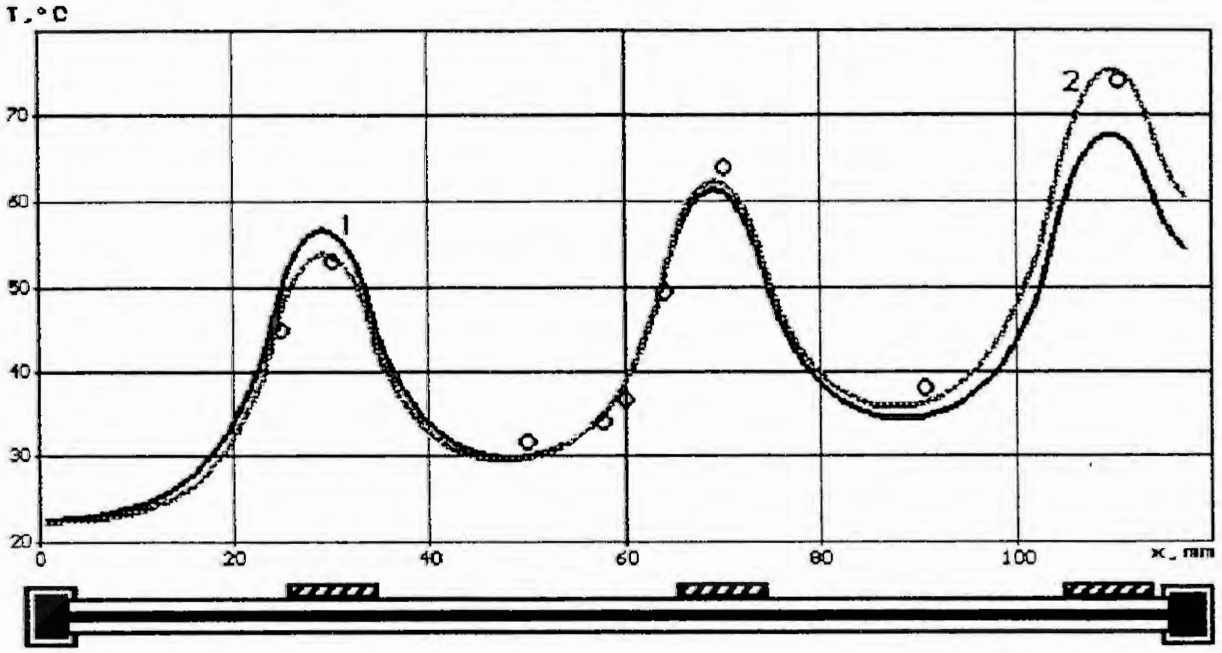


Fig. 2. Temperature of board external surface along the axis of the 14th channels at cooling fluid flow rate 6.245 g/s and total power 438 W

Fig. 2 shows the temperature change along a line which is the intersection of the board surface and a perpendicular there to plane symmetrically cutting the 14th channel along its length. The deviation of the calculated temperature values (curve 1, Fig. 2) from the experimental values shown by circles indicates that the mathematical model accounting for the heat exchange in the channels in the form of boundary conditions of the third kind with changing fluid temperature over the channel length $T_f = T(x)$ and constant convective heat transfer coefficient $h_f = h_{aver}$ over the channel length needs to be corrected.

The following regularity in the deviation of experimental temperature values from the calculated ones is observed: at the initial section of the channel the calculated temperature values are higher, at the end section they are lower, and at the middle section the temperature is within the measurement error limits. The temperature profile (curve 2) has been obtained by calculation at convective heat transfer coefficients (CHTC) changing according to

$$h = h_{aver}(a - bx)$$

where x is the current coordinate along the microchannel; h_{aver} is the CHTC average value (determined from the data base). Coefficiens a and b are determined as a result of solving the inverse problem for the experimental temperatures (for the specified above regime and channel dimensions a=1.32; b=5.38).

In this case the deviation of the calculated temperature values from the experimental ones is within the measurement error limits.

To construct a more exact mathematical model, it is necessary to carry out additional analytical and exerimental investigations.

The available data base containing information on the convective heat transfer coefficients and hydraulic resistances allows to model the thermal state of an electronic module and the hydraulics of its cooling system.

We shall give some results of modelling the temperature regimes of an elctronic module being designed. It has the following characteristics: the ceramic board has the dimensions 46×46×2 mm; the number of regularly arranged chips with the maximum power 5 W is 16 pcs; the characteristic dimensions of the chip are 3.2×3.2 mm. The results of investigating the influence of the number of microchannels on the maximal temperatures of the chips and on the fluid overheating in the cooling system are shown in Figs. 3 and 4.

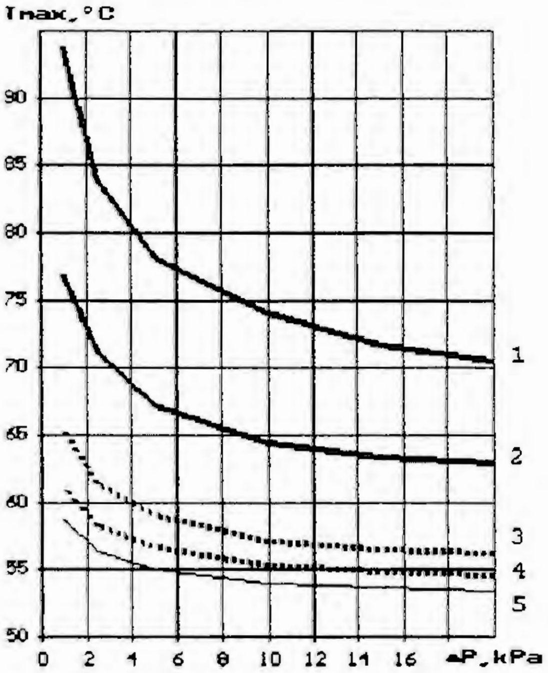


Fig.3. Maximum temperatures on the chips

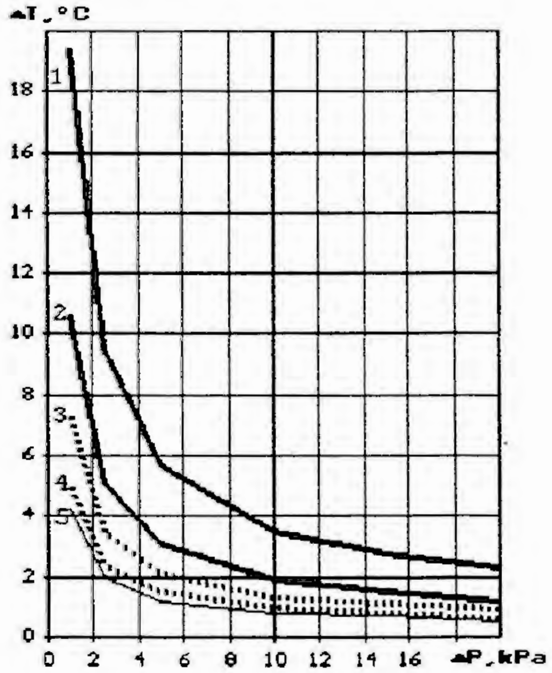


Fig.4. Temperature difference of fluid in the cooling system

1: 2 channels; 2: 4 channels; 3: 6 channels; 4: 9 channels; 5: 11 channels

RESULTS

A mathematical model of the thermal and hydraulic processes taking place in an electronic module with a microchannel system of fluid cooling is suggested. The adequancy of the mathematical model has been checked. It has shown that it is necessary to correct the convective heat transfer coefficients on the surfaces of the microchannals. The available data base containing information on the convective heat transfer coefficients and hydraulic resistances allows to model the thermal state of an electronic module and facilitate the process of its design.

ANALYSIS MODELS OF FUNCTIONING LARGE-SCALE DISTRIBUTED COMPUTER SYSTEMS

V.G.KHOROSHEVSKY

Computer Systems Department
Institute of Semiconductor Physics
Lavrent'ev avenue, 13, Novosibirsk, 630090
Russia
E-mail: Khor@isph.nsk.su

Abstract. Canonical Model of Distributed Computer Systems (DCS) is formulated. Results of Structural Robustness and Catalogue of optimal structures of DCS are presented. Stochastic Model of functioning large-scale DCS and unlaboured Approach to analysis of DCS Potential Robustness are described.

1. INTRODUCTION

Architecture of any Computing System (CS, hardware-software means for calculations) is manifested via its totality of properties and indices, which expose an ability of one to process information. The following architectural properties of CS should be mentioned: Performance, Dependability, Robustness, Fault-Tolerance, Adaptability, Realizability of solving problems, Technical-Economical Effectiveness. To satisfy the requirements to modern CS it is necessary to use quite a new basis - the Calculators Collective Model [1].

2. CS CANONICAL MODEL

As a common Model of CS we can use a couple $\langle H, A \rangle$ where H and A are, correspondingly, a hardware description and an algorithm of functioning a collective of calculators. The collective hardware is circumscribed by $H = \langle C, G \rangle$ where $C = \{c_1, c_2, ..., c_i, ..., c_N\}$ is the set of the interconnected calculators c_i; G is a structure description of the collective network.

The hardware H and the algorithm A are based on the following principals:

1) mass parallelism in data processing (parallel executing operations by set C and parallel interactions of calculators c_i through the network structure G);

2) programmability of the structure - adjustability of the network between calculators achieved by programming means);

3) constructive homogeneity.

As a description of the collective structure we may take a graph G whose vertices are associated with the calculators and edges - with the channels (connection lines) between them.

The algorithm A admits the representation in the form of the superposition A(P(D)) where $D = \cup D_i$, D_i is the initial individual array of data for the calculator c_i, besides in the general case $\cap D_i \neq \emptyset$; P is the parallel program of the computation, $P = \cup P_i$, $\cap P_i = \emptyset$, P_i is the i-th branch of the program, $i = \overline{1, N}$.

In the equivalent form the algorithm A of functioning the calculators collective is defined as the composition $(A_1 * A_1') * ... * (A_i * A_i') * ... * (A_N * A_N')$ where $(A_i * A_i')$ determines the behaviour of the calculator c_i among other calculators of the set C; A_i and A_i' are, respectively, the algorithm of the individual functioning of c_i and the algorithm of the execution of the c_i interactions with the calculators $c_j \in C/c_i$. The latter algorithm is represented as the superposition $A_i'(P_i'(G))$ in which P_i' is the program to establish connections and to realize interactions between the calculator c_i and other calculators of the subset C/c_i.

Variety in realizations of the above mentioned model is due to different ways of accomplishing the set $\{A_i'\}$, $i = \overline{1, N}$, describing the set $\{P_i'\}$ and choosing the structure G. It is also due to diversity in composition rules of algorithms. Hardware means which help to realize the set $\{A_i'\}$, $i = \overline{1, N}$, are Commutators. Efficiency of parallel computing means mainly depends on the way of commutators construction.

In case, when the homogeneity principle is fully held, we have the equivalence relations: $A_i \equiv A_j$, $A_i^{'} \equiv A_j^{'}$, $i,j \in \{1,2,...,N\}$, $i \neq j$. In particular, these relations provide high adaptability for designing and manufacturing, they lead to the Distributed Commutator joining N Local Commutators of calculators. They cause no troubles in creating the network between calculators.

Distributed CS with programmable structure is an architecturally flexible concept of realizing the calculators collective model [1]. A calculator of a DCS is usually called an Elementary Machine (EM). So, every EM of any DCS is intended to perform not only processing and storaging data but system instructions as well, i.e. to perform functions of organizing the collective of machines as a single whole. Let‘s consider here an elementary machine in a wide sense; every EM is a multipole cell containing a computer and local commutator. As a basis to construct EM we can take computing mediums, systolic structures, associative and array processors, transputers, von Neumann‘s computers, etc.

3. STRUCTURAL ROBUSTNESS

As one of DCS indices we can use Vector-Function of Structural Robustness:

$$L(G,S,s) = \{L_r(G,S,s)\},\ r = \overline{1,N},$$

where the component $L_r(G,S,s)$ is the probability of forming a subsystem of r rank in a system with G structure for assigned availability factors S and s of the local commutator and connection channel of elementary machine correspondingly. We call a subsystem of r rank a totality of r faultless elementary machines interconnected by channels in such a way that information transfer is possible from any EM into the rest of them.

The synthesis problem of the optimal structures is as follows: for the assigned S, s, N and the vertex degree v one should obtain structure G which provides the maximum of coordinates of L(G,S,s).

If systems are supposed to solve complicated problems (presented by parallel programs), then nowadays it is sufficient to restrict ourselves to structures which are D_n-graphs. These graphs have the parametric description $\{N, w\}$, where $w = \{w_0, w_1, ..., w_k, ..., w_{n-1}\}$; N,n are the order and dimensionality of D_n-graph respectively; number w_k is such that two vertices i and j are connected by the edge if

$$i - j \equiv w_k(modN),\ k = \overline{0, n-1},$$

holds. The catalogue of the optimal D_n-graphs for n=2 and $N \leq 256, n = 3,4,5$ and $N \leq 64$ has been compiled by means of Monte-Carlo simulation.

The investigation of L(N,v,g)-graphs has also been made. They are nonoriented and homogeneous graphs of N order, having degree v of vertices and girth g (the length of the shortest cycle in the graph).

4. STOCHASTIC MODEL FOR ANALYSIS OF FUNCTIONING DCS

Modern industrial CS are large-scale collectives of EM (e.g. the number of machines in MICROS systems [2] is not limited). These systems look as computing mediums and are intended for serving stochastic flows of jobs in the general case. Besides all components of CS are not absolutely dependable, their failures and restorations occur in stochastic time moments. There are such initial preconditions.

Thus, there is the situation which is in good agreement with the conception of mechanics of continua. Hence, we may consider a stochastic continuous model (or computing medium) instead of a discrete one to analyze the effectiveness of functioning CS. To calculate indices of performance, robustnees, realizability of solving problems and so on we can use the mean value of a number of serviceable elementary machines and assume that this value is a continuous function.

Let N be the number of EM constituting a DCS; m is the number of restoring devices (RD); λ and μ are the failure rate and restoration one. Evidently λ^{-1} and μ^{-1} are the mean values of time between failures of an EM and of time to restore a nonserviceable EM by one device respectively.

Further, let $\mathcal{N}$ (i,t) be the mean value of a number of serviceable EM at a time moment $t \geq 0$; i is the number of serviceable machines at the moment t=0 or i is said to be the initial state of DCS, $i \in \{0,1,...,N\}$. So, $\mathcal{N}$ (i,t) elementary machines constitute a computing "kernel" at the moment $t \geq 0$, $\mathcal{N}$ (i,0)=i. This kernel is intended to realize an adapting parallel programs. While realizing the adapting programs the number of parallel branches is fixed automatically, it corresponds to the amount of serviceable EM at any time moment. $\mathcal{M}$ (i,t) is used to designate the mean value of a number of restoring devices which realize a restoration of nonserviceable machines at a moment $t \geq 0$.

In the distributed CS there are hardware-software reconfiguration means. These means take into account nonserviceable machines of the computing kernel and EM restored by RD. They are also applied to exclude nonserviceable EM from the kernel, to make up the connected configuration of all serviceable EM of computing kernel, to reduce a number of branches in the adapting program, to organize its execution in the computing kernel with the new configuration.

Let $\mathcal{L}^{'}$ (i,t) be the mean value of a number of nonserviceable EM taken into account by the reconfiguration means at the moment $t \geq 0$, $i \in \{0, 1, ..., N\}$; $\nu^{'}$ is a rate of "eliminating" nonserviceable machines from the kernel. In addition, let $\mathcal{L}^{''}$ (i,t) be the mean value of a number of restored EM taken into consideration by the reconfiguration means; $\nu^{''}$ is the rate of including restored machines into the computing kernel. A mean value of including time will depend on the time of constituting the configuration of serviceable machines of the existing kernel and restorated ones, readjusting adapting program into one with a larger number of branchers, restarting this program at the newly created kernel.

The following equation takes place:

$$\mathcal{L}(i,t) + \mathcal{K}(i,t) + \mathcal{N}(i,t) = N, \ \mathcal{L}(i,0) = j, \ \mathcal{K}(i,0) = N - i - j,$$

where $\mathcal{L}$(i,t)=$\mathcal{L}^{'}$(i,t)+$\mathcal{L}^{''}$(i,t), $\mathcal{K}$(i,t) is the mean value of a number of nonserviceable EM taken into acount by RD, $t \geq 0$, $i \in \{0, 1, ..., N\}$. Thus, we have the stochastic model for analysis of functioning distributed CS.

TABLE. Optimal D_n-graphs

N	w_0	w_1	N	w_0	w_1	N	w_0	w_1	N	w_0	w_1	N	w_0	w_1
10	1	4	60	5	6	110	1	46	160	1	72	210	1	20
11	1	3	61	1	11	111	1	13	161	1	17	211	1	20
12	2	3	62	1	26	112	7	8	162	1	17	212	1	78
13	1	5	63	1	24	113	1	15	163	1	17	213	1	59
14	1	4	64	1	14	114	1	50	164	1	50	214	1	38
15	1	4	65	1	10	115	1	34	165	3	40	215	1	40
16	1	6	66	1	10	116	1	32	166	1	22	216	1	58
17	1	4	67	1	12	117	1	43	167	1	36	217	1	64
18	1	4	68	1	26	118	1	18	168	1	16	218	1	100
19	1	4	69	1	15	119	1	14	169	1	16	219	1	19
20	1	8	70	2	25	120	1	14	170	1	20	220	10	11
21	1	6	71	1	11	121	1	16	171	1	18	221	1	21
22	1	6	72	1	11	122	1	36	172	1	18	222	1	100
23	1	5	73	1	11	123	1	34	173	1	51	223	1	68
24	3	4	74	1	22	124	1	52	174	2	33	224	1	22
25	1	7	75	1	10	125	1	19	175	1	40	225	1	40
26	1	10	76	1	10	126	2	49	176	1	38	226	1	42
27	1	6	77	1	14	127	1	15	177	1	54	227	1	61
28	1	6	78	1	12	128	1	15	178	1	78	228	1	82
29	1	8	79	1	12	129	1	15	179	1	17	229	1	24
30	2	9	80	1	22	130	1	28	180	9	10	230	1	20
31	1	7	81	1	24	131	1	20	181	1	19	231	1	20
32	1	7	82	1	36	132	3	32	182	1	82	232	1	22
33	1	6	83	1	11	133	1	14	183	1	54	233	1	71
34	1	6	84	6	7	134	1	14	184	1	42	234	1	86
35	1	10	85	1	13	135	1	18	185	1	40	235	1	65
36	1	8	86	1	36	136	1	16	186	2	39	236	1	42
37	1	8	87	1	24	137	1	16	187	1	57	237	1	44
38	1	16	88	1	26	138	2	15	188	1	22	238	1	64
39	1	7	89	1	16	139	1	30	189	1	18	239	1	25
40	4	5	90	1	12	140	1	32	190	1	18	240	1	110
41	1	9	91	1	12	141	1	54	191	1	20	241	1	21
42	1	16	92	1	14	142	1	64	192	2	21	242	1	21
43	1	12	93	1	39	143	1	15	193	1	44	243	1	21
44	1	8	94	1	26	144	8	9	194	1	42	244	1	72
45	1	8	95	1	17	145	1	17	195	1	54	245	1	88
46	1	10	96	1	42	146	1	64	196	1	86	246	1	66
47	1	13	97	1	13	147	1	54	197	1	23	247	1	46
48	1	20	98	1	13	148	1	32	198	2	81	248	1	26
49	1	9	99	1	13	149	1	34	199	1	19	249	1	69
50	1	9	100	1	18	150	2	21	200	1	19	250	1	20
51	1	9	101	1	30	151	1	20	201	1	19	251	1	20
52	1	8	102	1	12	152	1	16	202	1	56	252	1	24
53	1	8	103	1	12	153	1	16	203	1	36	253	1	22
54	1	12	104	1	16	154	1	18	204	1	38	254	1	22
55	1	10	105	1	14	155	1	57	205	1	24	255	1	117
56	1	10	106	1	14	156	1	46	206	1	74	256	1	92
57	1	24	107	1	41	157	1	34	207	1	18			
58	1	22	108	1	32	158	1	36	208	1	18			
59	1	9	109	1	30	159	1	21	209	1	22			

5. POTENTIAL ROBUSTNESS OF DCS

Theoretically (on the basis of parallel algorithms analysis) and experimentally (by means of realizing adapting parallel programs of complicated problems by DCS) it is shown that the values of system performance is directly proportional to the number of serviceable EM. Hence, the mean values of CS power (performance, capacity of memory, etc.) and throughput of all restoring devices at the moment $t \geq 0$ are

$$\Omega(i,t) = \mathcal{N}(i,t)\omega, \quad \mathcal{M}(i,t)\mu$$

respectively, where ω is power of one EM. So, in our situation it is enough to use the Function of Potential Robustness:

$$N(i,t) = \Omega(i,t)/N\omega = \mathcal{N}(i,t)/N.$$

Unlabored calculations based on the stochastic model of functioning DCS give us:

$$\begin{aligned}\mathcal{N}(i,t) = \frac{N\mu\nu}{\alpha_1\alpha_2} &+ \left[\frac{N\mu\nu}{\alpha_1} + (i+j)\nu + i(\alpha_1+\mu)\right]\frac{e^{\alpha_1 t}}{\alpha_1-\alpha_2} + \\ &+ \left[\frac{N\mu\nu}{\alpha_2} + (i+j)\nu + i(\alpha_2+\mu)\right]\frac{e^{\alpha_2 t}}{\alpha_2-\alpha_1},\end{aligned}$$

$$\begin{aligned}\mathcal{M}(i,t) = \frac{N\lambda\nu}{\alpha_1\alpha_2} &+ \left[\frac{N(\alpha_1+\lambda)(\alpha_1+\nu)}{\alpha_1} - (i+j)(\alpha_1+\lambda+\nu) + i\lambda\right]\frac{e^{\alpha_1 t}}{\alpha_1-\alpha_2} + \\ &+ \left[\frac{N(\alpha_2+\lambda)(\alpha_2+\nu)}{\alpha_2} - (i+j)(\alpha_2+\lambda+\nu) + i\lambda\right]\frac{e^{\alpha_2 t}}{\alpha_2-\alpha_1},\end{aligned}$$

$$\mathcal{L}(i,t) = N - \mathcal{N}(i,t) - \mathcal{M}(i,t), \ \mathcal{K}(i,t) = \mathcal{M}(i,t),$$

$$i \in \{N-m,\ N-m+1, \ldots, N\},\ \nu = \nu' + \nu'',$$

$$\alpha_{1,2} = -\frac{1}{2}\left[\lambda + \mu + \nu \mp \sqrt{\lambda^2 + \mu^2 + \nu^2 - 2(\lambda\mu + \lambda\nu + \mu\nu)}\right],$$

if $N\lambda \leq\ m(\lambda + \mu + \lambda\mu/\nu)$ is valid. For modern computing means the inequality $\lambda < \mu \ll \nu$ is known to hold always, therefore

$$\mathcal{N}(i,t) = \frac{N\mu}{\lambda+\mu} + \frac{i\lambda - (N-i)\mu}{\lambda+\mu}e^{-(\lambda+\mu)t},$$

will describe DCS behaviour. Evidently the mean number of serviceable machines does not depend on the DCS initial state in the stationary regime:

$$\mathcal{N} = \lim_{t\to\infty}\mathcal{N}(i,t) = N\mu(\lambda+\mu)^{-1}.$$

So, there are no difficulties for express-analysis of Potential Robustness of DCS with arbitrary number of elementary machines.

6. RESULTS

The results obtained in the structure robustness are the basis for fast choosing the topologies of industrial distributed homogeneous multicomputer systems. Analysis of potential effectiveness of DCS shows that in order to choose the redundansy and the number of restoring devices it is sufficiently to use the formulae:

$$N - n <]0,1N[,\ m <]0,1N[,$$

where $]x[$ is the closest to x integer such that $]x[> x$.

7. REFERENCES

[1] Khoroshevsky V.G., Engineering analysis of functioning computing machines and systems. Moscow, Radio i Svjaz, 1987.

[2] Khoroshevsky V.G., MICROS-2: Distributed computer systems with programmable structure. To appear in: K.Boyanov (Ed.), IFIP TCG'93, International Simposium "Networks Information Processing System", Sofia, 1993.

The extension theory and solvable models in diffraction theory, nanoelectronics, hydrodynamics and biophysics

I.Yu. Popov
Department of Higher Mathematics,
Leningrad Institute of Fine Mechanics and Optics,
14 Sablinskaya ul., 197101 Leningrad, Russia, USSR.

November 8, 1993

Abstract

Short review of the models based on the theory of self-adjoint extension of symmetric operator for the problem of diffraction of acoustical and electromagnetic waves, nanoelectronics, the theory of Stokes and stratified flows, the problem of hydrodynamic stability of boundary layer and biophysics is given.

A class of solvable models based on the operator extensions theory shall be described. The first model of such sort was put forward ten years ago by B.S. Pavlov [1] for the description of the Helmholtz resonator. The of the approach is the followivg. Let Ω^{in} be the dowain in R^3 with smooth boundary $\partial\Omega$, $\Omega^{ex} = R^3\backslash\Omega^{in}$, $x_0 \in \partial\Omega$. Let us consider the Laplace operators with the Neumann boundary condition in $L_2(\Omega^{in})$ and $L_2(\Omega^{ex})$ correspondingly. Let us restrict these operators onto the sets of smooth functions vanishing near the point $x_0 \in \partial\Omega$. The closures Δ_0^{in}, Δ_0^{ex} of the obtained operators are the symmetric operators with the deficiency indices (1.1). Consequently, the operator $\Delta_0^{in} \oplus \Delta_0^{ex}$ has the deficiency indices (2.2). The equality of the deficiency indices causes the existence of self-adjoint extensions. These extensions are the model operators. The Green's function, the solution of the scattering problem, and the scattering matrix can be found for the model operator in an explicit form.

To make clear whether the model is realistic or not, it is necessary to compare the solutions of the model and "realistic" problems. Let $\Gamma_\delta = \partial\Omega\cap\{x : |x-x_0| < \delta\}$ be the opening of radius δ. Let us consider the Laplace operator with a Neumann boundary condition on $\partial\Omega$ Γ_d and the condition of the absence of the radiation from the edges. We shall call this operator the "realistic" one. The comparison of the realistic Green's function with the model one shows [3], [2] that one can choose the extension parameters in such a way that the model Green's function coincides with the main term of the asymptotics (in d) of the realistic Green's function. To obtain the other terms of this asymptotic expansion, it is necessary to take into account multipoles, but the corresponding derivatives of the Green's function do not belong to the space L_2. To include these derivatives into the domain of the model operator it is necessary to extend the original space and to construct the model in this extended space [6]. The analogous situation is for the Dirichlet boundary condition.

This program is realized in [4]. We obtain the model in the space with an indefinite metric. It is important that the model is "explicitly solvable". That is, if the solutions of the problems in Ω^{in} and Ω^{ex} (without opening) are known then the model solution is found in an explicit form. It should be mentioned that the model and realistic solutions are alike in the operator sense. The relation between the resolvents of the model and realistic operators allows one to obtain desired information about different characteristics of the realistic operator. For example, if one fixes a semicircle $|k| < R$, $Imk > 0$, in the upper half-plane of spectral parameter k, then it is possible to choose the extension parameters in such a way that in this semicircle the spectral characteristics (for instance, resonances, i.e. zeros of the S-matrix usually called scattering frequencies) of the model operator is the approximation with high precision of the corresponding spectral characteristics of the realistic operator. Thus the model gives a low-frequency approximation in the problem of scattering of waves by the resonator with a small opening. The resonator with narrow slit can be described in the framework of this approach too [5].

The model is useful not only in acoustics and electrodynamics but also in other fields. The model has an interesting application in nanoelectronics. One can use the approach for the description of transport properties of the electron in mesoscopic structures analogous to the system of waveguides and resonators. The advantage of the model is that it allows one to obtain the transmission coefficient and the conductivity in an analytic form. The dependence of the transmission coefficient on the electron energy has a resonance character [7]. This effect may be used for the construction of nanoelectronic devices, for example, a transistor based on the quantum interference phenomena. The extension theory can be used for the simulation of double-barrier mesoscopic structure. The quantum dot of this type is modelled by the resonator with semitransparent boundary [8].

The described zero-width slits model can be applied for the hydrodynamics problem. Let us consider the periodic system of Helmholtz resonators (in R^2) connected with the upper half-plane through small openings. The corresponding model problem can be solved explicitly. We obtain the dispersion equation.

$$(G_0^{in}(x,x_0,k)-G_0^{in}(x,x_0,k_0))\Big|_{x=x_0} = (G^{ex}(x,x_0,k)-G^{ex}(x,x_0,k_0))\Big|_{x=x_0} + \sum_{n\neq 0} e^{ipan} G^{ex}(x_n,x_0,k)$$

One can see that there is quasi-stationary bands and the corresponding surface wave. We shall consider the following problem. Let there is a flow of viscid compressible fluid in upper half plane. The problem of influence of acoustic effects on the hydrodynamic stability of boundary layer may be studied with using of the described model. It is shown that for some parameters of the system the flow becomes more stable due to the influence of the acoustic field.

The same model system can be used for the description of the another (biological) system - the lateral line organ of a fish. Our model allows one the compute the spectral characteristics of the acoustic system of the lateral line organ and the parameters of surface waves and to understand the possible mechanism of sound detection by this organ based on the resonance of the incoming wave with the surface wave of the lateral line.

The another application of the operator extension approach is to the stokeslet method in the fluid mechanics. Stokeslet is the singular solution of the Stokes equations describing the creeping flow (low-Reynolds-number flow). The review of the stokeslet applications is in [10]. It is possible to construct the operator version of the stokeslet approach based on the extension theory. In two-dimensional case the Stokes equations reduce to the biharmonic equation for the stream function. The first idea concerning to the operator description of zero-range interaction in this situation is in [11]. Here we start from the square of the Laplace operator restricted on the set of smooth functions vanishing simultaneously with its derivatives of first and second order. Its self-adjoint extensions gives us the model. The full description of the set of the extensions is given in [12]. The obtained model is one for the stokeslet of the first order. It is necessary to extend the initial space L_2 by adding the corresponding solutions with strong singularities to include higher stokeslets into the operator scheme.

One can obtain the operator version also for the problem for a bounded domain when the singularity is at the boundary. In this case the initial restricted operator in the space L_2 has the dificiency indices (2,2). The self-adjoint extension is here the model of the creeping flow in the domains, connected through the small opening. Using this approach one can describe the flows in different complicated domains. The results may be useful for the construction of various mechanical devices, such as the lubrication systems and fine servo systems.

The another interesting application of the model is in the theory of stratified flows. It is known that the two-dimensional stratified flow of inviscid incompressible fluid in gravitational field is described by the non-linear Dubreil-Jacotin equation. We suppose that the media is a dielectric one and consider this flow in an electric field. In this case the modified Dubreil-Jacotin equation can be derived. It can be reduced to the linear equation which is analogous to the Schrödinger one with an attractive potential. Hence, the eigenstates can exist. It means that there exist local eddies in the corresponding flow caused by the electric field. This result may be useful in the theory of lightning and the related topics. In the case when the electric field is absent one obtains the Helmholtz equation. Of course, we can consider the Dirichlet problem in the bounded domain. It is clear that zero-width slits model described above can be constructed in this case. Here it gives the description of stratified flow in the domains connected through small apertures.

Acknowledgements

I am grateful to the organizers for the possibility to make the report and to Prof. B.S.Pavlov and V.A.Geyler for the discussions.

References

[1] B.S.Pavlov. The theory of extensions and explicitly-solvable models. Uspechi Mat. Nauk **42** (1987), No 6, 99- 131.

[2] V.Yu.Gottlieb. The scattering by circle resonator with small slit as a problem of perturbation theory. Dokl. AN SSSR **287** (1987), 1109- 1113.

[3] I.Yu.Popov. The extension theory and localization of resonances for domains of trap type. Mat. Sbornik **181** (1990), No 10, 1366- 1390.

[4] I.Yu.Popov. Helmholtz resonator and the operator extension theory in the space with indefinite metrics. Mat. Sbornik **183** (1992), 3- 37.

[5] I.Yu.Popov. The resonator with narrow slit and the model based on the operator extensions theory. J. Math. Phys. **33** (1992), 3794- 3801.

[6] Yu.G.Shondin. Quantum-mechanical models in R_n based on the extensions of the energy operator in Pontryagin's space. Teor. Mat. Fiz. **74** (1988), No 3, 331- 344.

[7] I.Yu.Popov, S.L.Popova. The extension theory and resonances for a quantum waveguide. Phys. Lett. A. **173** (1993), 484- 488.

[8] I.Yu.Popov. The extension theory and the opening in semitransparent surface. J. Math. Phys. **33** (1992), 1685- 1689.

[9] B.S.Pavlov, I.Yu.Popov. The acoustical model of zero-width slits and hydrodynamic stability of boundary layer. Teor. Mat.Fiz. **86** (1991), 391- 401.

[10] H.Hasimoto, O.Sano. Stokeslets and eddies in creeping flow. Ann. Rev. Fluid Mech. **12** (1980), 335- 363.

[11] Yu.E.Karpeshina, B.S.Pavlov. Zero-range interaction for biharmonic and poliharmonic equations. Mat. Zametki. **40** (1986), No 1, 49- 59.

[12] I.Yu.Popov. Operator extensions theory and eddies in creeping flow. Physica Scripta. **47** (1993), 682- 686.

Stochastic simulation of mobile radio protocols based on their formal specification

T. Hellmich
FernUniversity of Hagen
Frauenstuhlweg31, D-58644 Iserlohn

Abstract

Protocol design nowadays is performed using formal description techniques getting rid of implementation details and aiming at clear and concise specifications. Based on the mathematical model of (extended) finite state maschines the specification languages SDL and Estelle have been standardized internationally.

In this contribution a method is presented bridging the gap between formally specified protocols of interest and their performance evaluation. Simulators are generated automatically and directly out of formal Estelle system specifications including the protocols to be evaluated using an Estelle-C compiler. The method is verified by comparing performance measures obtained out of generated simulators with those obtained out of a hand-coded simulator.

1 Introduction

Protocol specifications using SDL or Estelle remove the well known drawbacks of informal human language specifications like ambiguity, uncompleteness and unproveability of desired protocol properties. Formal specifications therefore result in correct specifications. On the other hand such formal specifications are bad suited to evaluate protocols in the time domain due to the level of abstraction. Moreover, implementations of formally specified protocols have to be shown conform to the specification, which is generally a hard problem.

In this paper media access control protocols for short-range mobile radio applications are used to demonstrate the method of generating simulators, which in turn can be executed to evaluate performance measures of interest. Thereby, the scenario in which the protocols are assumed to work is highway traffic. The intention is not to give a complete formal specification of a number of competitive protocols or to describe in detail the mobility or channel models used for simulation, therefore cf.[4], but to explain and verify the applicability of the method. The main idea is to integrate the simulation time into the operational semantic of the formal system specification in such a way, that the time ordering of the simulated events remains uneffected by the interpretation of the semantic of the system specification.

2 System concept

Protocols under investigation have to be specified completely using the formal description language Estelle [2], [8]. The services used by the specified protocols, like the physical layer service, have also to be specified in Estelle by using appropriate abstractions. The concept allows to replace a service specification in a modular way by a protocol specification, which itself again uses e.g. service specifications of a lower layer, cf. 2. Therefore, the concept supports the successive specification of a complete protocol stack. Embedding time progress into the specification, cf. sec. 7, the complete system specification becomes executable automatically, by translating it to C using an existent Estelle-C compiler, cf. [3]. The C-code subsequently is compiled and linked together with a number of assisting procedures and functions defined externally and the Estelle run-time library routines, realizing the operational semantics of the system specification cf. [2]. Fig. 1 shows the process of generating a simulator out of an Estelle system specification. The generated simulator's two major features are

1. dynamically monitoring the
   - internal behaviour of the specified protocols by observing the protocol data flow
   - observing the activation of service primitives to achive this flow
   - execution of transitions and the related specified actions
2. specification of dedicated event counters as additional variables, yielding the basis of statistical evaluation of interesting performance measures like throughput or handover rates.

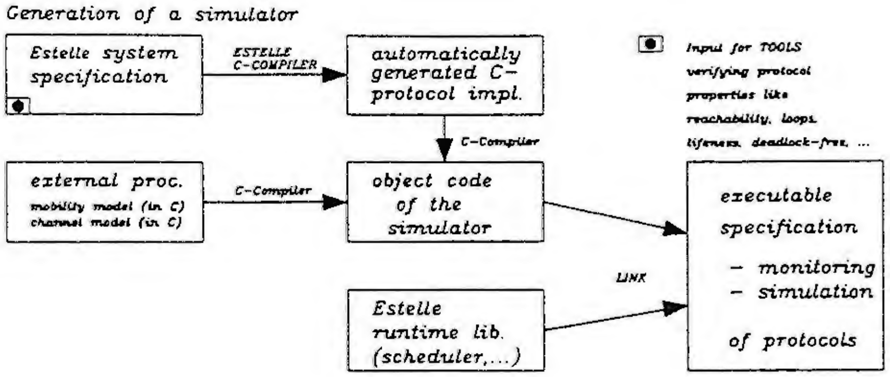

Generation of a simulator
Estelle system specification
ESTELLE C-COMPILER
automatically generated C-protocol impl.
C-Compiler
external proc.
mobility model (in C)
channel model (in C)
C-Compiler
object code of the simulator
Estelle runtime lib. (scheduler,...)
LINK
executable specification
- monitoring
- simulation
of protocols
Input for TOOLS verifying protocol properties like reachability, loops, lifeness, deadlock-free, ...

Figure 1: Generation of a simulator

Both, 1. and 2. are to specify, since the specifier is free to decide what events are to observe or evaluate.

3 Modelling protocols

Aiming at formal protocol specifications of media access control protocols using Estelle or SDL, the first task is to identify states, state transitions, protocol data units (PDUs), and appropriate service primitives (SPs), to be able to specify protocol behaviour in terms of extended finite state maschines, cf.[12]. As a part of this state-decomposition process the specifier has e.g. to decide, whether information is modelled using states or using associated variables. The guideline in this process should be readability and understandability [6] of the specification. The concept of communicating finite state maschines [12] leads to modular extensible protocol stacks as shown in fig. 2 above the physical layer service. Communication of each vehicle is modelled by an instance of the specified protocol stack, which here comprises only layer 2 (MAC) and layer 7 (user). To enable the running applications to communicate using the specified protocols, a physical layer service is necessary as well as a timer instance, synchronizing the protocol instances involved.

The physical layer service itself of course is specified as a finite state automata, cf.5. Under modelling aspects the embedding of a mobility model, indicated in fig. 2 by dotted lines within the physical layer service, can be thought of as a large structured variable with actual values for location, speed, ... of each vehicle. Similary the physical channel model represents a complex assignment operator to assign to each tuple (transmitter, receiver, interfererl, interferer2, ...) the probability, that the receiver gets the datablock. Sec. 5 explains the specification more detailed. It is worth noting that both, the mobility as well as the channel model, can be replaced by better models without changes to the specification.

Executing such a system specification means to interpret the specification according to the standardized operational semantics of the specification language used, e.g. Estelle [2],[8]. To do this, a compilation of the system specification to executable code is necessary. Sec. 2 has shown, how such a translation can be generated automatically.

4 Modelling of radio transmission and mobility

For radio transmission of datablocks a TDMA scheme with a frame and slot structure is assumed. Thereby, perfect synchronization is presumed. Identical slotnumbers in consecutive frames yield logical TDM-channels. Each station in the system is assumed to transmit exactly one datablock per frame, using a channel.

To decide, whether a receiver in a mobile radio network is able to receive a transmitted datablock during a slot, a channel model is necessary. The decision is a function of the actual transmitter-receiver distance and all distances of the receiver to each simultaneously transmitting station. Since the distances change continously, due to vehicles mobility, an appropriate mobility model is required too.

4.1 Channel model

The physical channel model is taken from [1] and considers the propagation and attenuation conditions at 60 GHz thereby including possible (multi) co-channel interferences at a receiver. The model is parameterized by transmission power, weather conditions, error detection/correction coding and antenna characteristics. Signal-to-noise and signal-to-interference ratios are derived and under the assumption of DPSK modulation the model

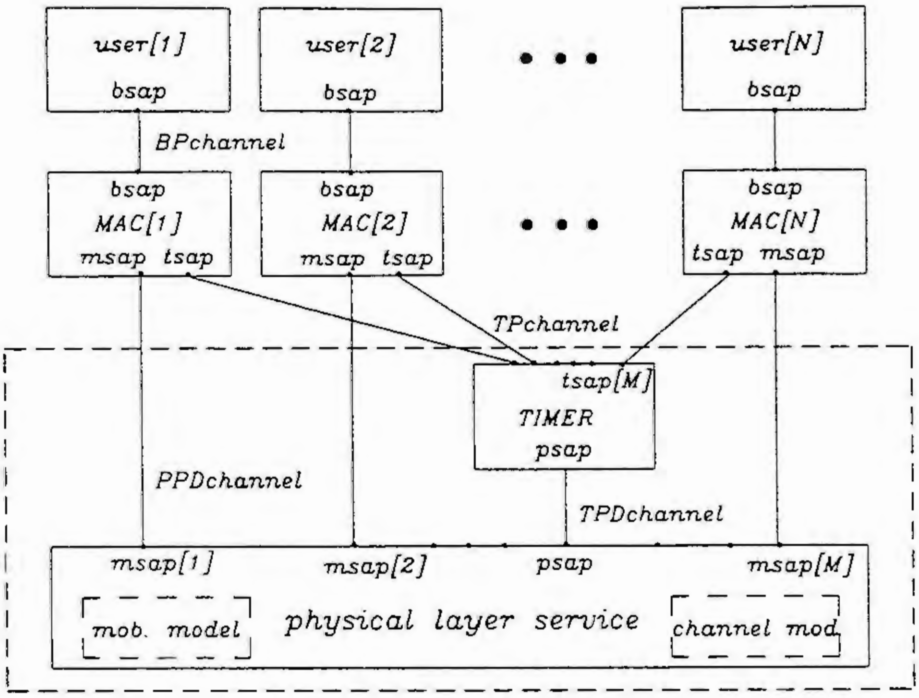


Figure 2: System specification

calculates the probability for reception of a datablock. The model provides the information that signal energy is detectable but reception is not possible whenever a receiver is too far distant from the transmitter or co-channel interference is present. Transmission ranges are up to 500m. Dependend on the actual transmitter-receiver distance, the same channel at the same time can be used successfully some distance apart (space multiplexing).

4.2 Mobility model

Mobility is modelled using the scenario of a n-lane highway crossing. Vehicles are generated from four given density parameters, each for one direction of movement. By setting three of them to zero, uni-directional (D1) mobility with all vehicles moving into the same direction can be simulated, and by setting two of them to zero, bi-directional (D2) mobility with one half moving into one and the other half of vehicles moving into the opposite direction can be investigated. The movement of vehicles is modelled correlated taking a large number of parameters like weight, length, accelleration, decelleration, ... into account. The speed of vehicles is normally distributed and constant, as long as no slower moving vehicle in the same lane in front forces a safe speed adaption by decelleration in case that no overtaking is possible. As soon as overtaking becomes possible it is performed thereby accellerating to the former speed. In [4] it is shown, that mobility is modelled realistically, by comparing mobility measures out of simulations with empirical data.

5 Physical layer service

The physical layer service is specified using only two states, namely IDLE and TRANSMIT, cf. fig.3. In state IDLE all SPs of all MAC-instances (C_SENDreq or C_NOSENDreq) interacted during a slot are gathered. Thereby the SP C_NOSENDreq, interacted by a MAC instance if it has no datablock to transmit during the actual slot, is only necessary for time synchronization. Using a counter variable, the physical layer service modnl transits into state TRANSMIT when the last MAC instance has interacted its SP (ready:=TRUE). The physical layer service is activated only once per slot for transmisson, because it must be known, which MAC instances try to transmit simultaneously. Using the mobility model variables and the channel model function mentioned, the physical layer modul proves for each MAC instance whether it receives a datablock by interacting the SP M_RECind with the appropriate datablock as a parameter or it detects signal energy (M_SIGNALind) or nothing (M_NOTHINGind). Again, the SP M_NOTHINGind is only necessary for time synchronization of the MAC instances, similary to C_NOSENDreq.

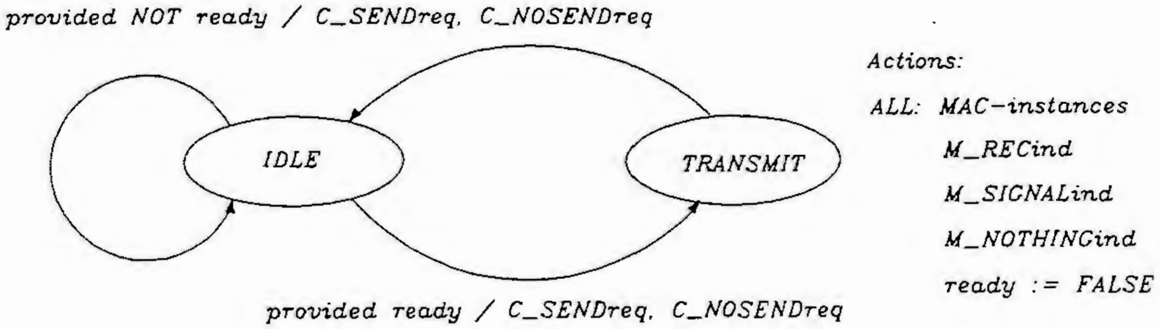


Figure 3: Finite state model for the physical layer service

6 Media access control protocols and higher layers

Based on the service of the physical layer, to interact per slot either a M_RECind or a M_SIGNALind or a M_NOTHINGind to each MAC instance, each receiver is able to record the observed event for each of the past N slots, where N ist the number of slots per frame. Presuming frame synchronization each station is able to code its locally observed channel occupancy information into a N bit comprising bitmap. Based upon such bitmaps, which moreover can be exchanged between neighboured stations, a number of multiple access protocols for TDMA radio networks have been defined cf. [10], [13], [7], [5]. The protocols differ in their

bitmap
data

Figure 4: Datablockformat of CSAP

- assumptions on detection information obtainable
- mechanisms to initiate and perform a handover
- protection of channels due to differences in gathering local channel occupancy information

The most simple protocol is CSAP (Concurrent Slot Assignment Protocol). It assumes no detection ability of a transceiver. The bitmap is coded '1' for reception of a datablock during a slot, '0' otherwise. The bitmaps are transmitted piggy backed as part of each datablock, cf. fig. 4. A station performs a handover, whenever it receives a datablock with a bitmap where its own transmission slot is marked '0', due to the assumption of interference. To reduce the resulting handover rate, the original CSAP was changed and a handover performed only, if in two consecutive frames a '0' is encountered. A station changes its transmission slot (channel) excluding all channels being marked occupied ('1') in at least one of the bitmaps received during the last frame. Estelle specifications of some MAC protocols are given in [4] and [11].

The service CSAP provides for a higher layer is a continous datablock stream with one block per frame from each station being within the receive range. Vice versa, a station transmits under the control of the MAC protocols exactly one datablock per frame.

ISO/OSI layers 3 to 6 of the investigated system specification are actually empty, but can be extended using the concepts mentioned. The specified user application only consumes the received datablocks without executing any application. Vice versa the assumed user application generates one empty datablock each frame.

7 Specification of time

To get executable specifications it is necessary to specify the progress of time explicitly as part of the system specification, cf. fig. 2. Time is modelled and specified according to the discrete periodic simulation approach. The atomic discrete time unit is the duration of a slot. Within the time unit all events are assumed to occur quasi simultaneously. The Estelle module TIMER is specified as a time supervisor of all (MAC) protocols instances generated during system instantiation in a server client manner. It controls the beginning and the number of slots of a simulation. The TIMER instance acts as a server and the MAC instances as clients. The TIMER synchronizes the system by simultaneously interacting T_SLOTind SPs (T_FRAMEind after each N times, respectively) to all

MAC instances. Each MAC instance starts asynchronously its individual protocol actions according to the valid system time right after receiving the SP, thereby initiating the actions of the physical layer service as described in 5. The subsequent synchronization of time is achived by the module TIMER by waiting until the last MAC instance has indicated asynchronously, that it is ready for the next slot by interacting the SP C_TACTreq to the TIMER module.

Every Estelle modularization of the system specification has to regard, that interpreting the system specification following the standardized semantic rules of Estelle [2] must not change the time ordering of the simulated events. Each specification satisfying this condition can automatically be translated to executable code, implementing a simulator of the whole system.

The method allows the specification of time only by using discrete atomic time units (slots), e.g. wait 5 slots for a time-out. Other measures like seconds must not be applied, because no semantic is standardized.

8 Results

The comparison of results gained by executing generated simulators out of Estelle system specificatipons (AC_PTS, Automotive Communications Protocol Test and Simulation tool) with simulation results obtained out of a hand coded MODULA_2 simulator (RAVS), cf. [9], show definitely their correspondences. For a detailed description of the simulation parameters cf. [4]. Exemplary fig. 5, 6,7 show the handover rates over the density of vehicles, the distribution of collided datablocks over the receive range R of a station and the throughput as the number of successfully received datablocks per frame within R. This result verifies impressively, on a simulation basis, the applicability of the method. Summerizing, the main benevits of the discussed method are:

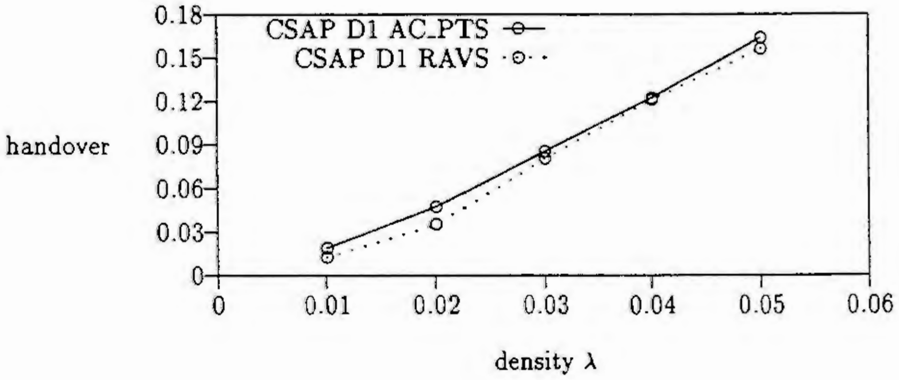


Figure 5: Handover rates of CSAP

- formal correctness proofs are possible immediately, using existing tools
- performance evaluation is possible immediately, using generated simulators
- no conformance testing between formally specified protocols under investigation and their respective implementation is necessary.

References

[1] H. Bischl, W. Schäfer, and E. Lutz. Modell für die Berechnung der Paketfehlerrate unter Berücksichtigung der Antennencharakteristik. Technical report, DLR Oberpfaffenhofen, 1990.

[2] S. Budkowski and P. Dembinski. An Introduction to Estelle. *Computer Networks*, 14(1), 1987.

[3] R.I.M.-H. Chan. An Estelle-C Compiler for Automatic Protocol Implementation. Technical report, University of British Columbia, 1987.

[4] T. Hellmich. *Formale Spezifikation und Leistungsbewertung von Vielfachzugriffsprotokollen in Mobilfunknetzen.* PhD thesis, FernUniversität -GHS- Hagen, LIT Verlag Münster-Hamburg, 1993. ISBN-Nr. 3-8258-2001-7.

[5] T. Hellmich and B. Walke. DMAR-A Decentral Multiple Access protocol with reservation. *Proc. of 4th PROMETHEUS Workshop*, 1990.

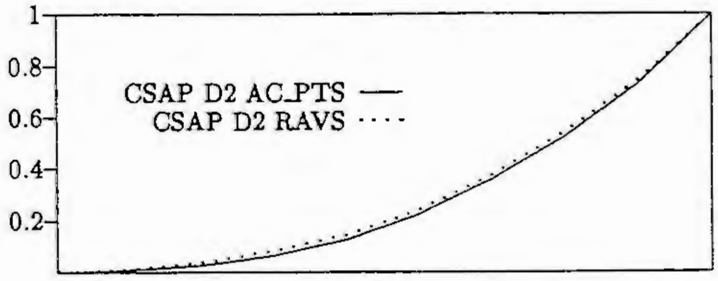


Figure 6: Distribution of collisions over R

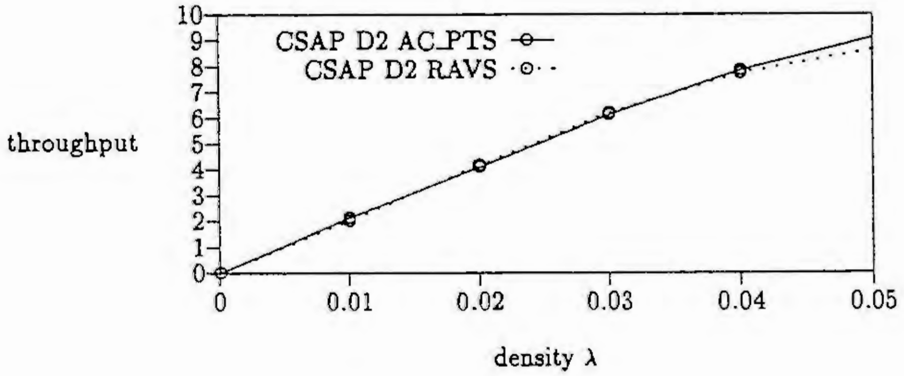


Figure 7: Throughput over stations density

[6] D. Hogrefe. *Estelle, LOTOS und SDL: Standardspezifikationssprachen für verteilte Systeme*. Springer Verlag, 1989.

[7] D. Hübner, K. Jakobs, and F. Reichert. Taking Advantage of the Disadvantage: Interference Detection for Improved Decentral Radio Channel Access. *Vehicular Technologie Conference*, 41(No. 41):pp. 374–379, 1991.

[8] ISO. Estelle: A formal description technique based on an extended state transition model, ISO/IS 9074. Technical report, International Standards Organisation, 1987.

[9] M. Kalle. Vergleich von Zugriffsprotokollen für slotsynchronisierte mobile Paketfunknetze. Technischer Bericht 3-92, FernUniversität -GHS- Hagen, Datenverarbeitungstechnik, 1992.

[10] J. Rückert and A. Mann. CSAP for packet-radio networks: A new concurrent slot assignment protocol. *EUROCON*, 1988. Stockholm Sweden.

[11] E. Tsimopuln. Formale Spezifikation und simulative Bewertung von Schicht-2 Protokollen für teilvermaschte Paketfunknetze mit mobilen Teilnehmern. Technischer Bericht 3-91, FernUniversität -GHS- Hagen, Datenverarbeitungstechnik, 1991.

[12] G. von Bochmann. Finite State Description of Communication Protocols. *Computer Networks*, 2, 1978.

[13] W. Zhn, T. Hellmich, and B. Walke. DCAP, a decentral channel assignment protocol: Performance analysis. In *Vehicular Technologie Conference*, volume 41, 1991. St. Louis, pp. 463-468.

Generative Radio Channel Models for Analysis and Simulation

Herbert Steffan
Communication Network Aachen University of Technology
52074 Aachen Kopernikusstr.16
e-mail: hst@dfv.rwth-aachen.de

Abstract

Generative channel models based on finite state models are introduced in this paper. The main method applied is separated quantisation of the various fading processes and their superposition to the channel process. A homogeneous finite state Markov model is proposed with constant transition probabilities to characterize the process. In addition a Semi-Markov model with transition probabilities depending on the specific sojourn-time is introduced. In both models the fading processes are modelled separately. Thus, the transition probabilities are independent from parameters like e.g. the mean of the process. Due to this type of independence, these models are very useful for stochastical simulation and mathematical analysis of mobile radio networks. The models defined are compared with well known fading processes derived from discrete lowpass filters.

In mobile radio communications the quality of transmission is influenced significantly by fading of the signal envelope. Traditionally burst error behaviour of radio channels is modelled by stochastic processes. A common assumption used is the Wide Sense Stationary Uncorrelated Scattering (WSSUS) channel, making the channel model mathematically more tractable. A well known method for simulating the characteristics of a radio channel is to filter white Gaussian processes with the shape of the Doppler spectrum of the fading process. Although this model is known to represent the correlation characteristics of the mobile radio channel quite well, it is not very suitable for investigations by stochastical simulation because of low numerical efficiency of digital filtering. Considering mathematical analysis of the radio channel there are only few methods proposed with a limited range of application. Probability functions are conventionally used describing the process to calculate symbol error [10]pp. 266.

In this paper three generative channel models are presented which are in a certain sense self-parameterizing. They can be derived from state continuous WSSUS stochastic processes. The main principle is discretizing the continuous state space of the process to a finite number of states. By this way a generalized Gilbert-Elliot model is parameterized. Algorithms are described how to get a mapping to finite state models. Correctness of mapping is shown by means of simulation considering propability of duration of fading and non-fading intervals. Fading processes described by a Rayleigh and a Lognormal distribution an be modelled separately. In the same way as these state continuous processes are superimposed independently, they might also be combined as state discrete models to substantially ease the calculation of the parameters characterizing the fading process.

These models are useful representations of the radio channel for mathematical analysis and to calculate its entropy. In addition they can easily be used to generate streams of correlated disturbed symbols for stochastical simulation.

The paper is organised as follows: Classical fading models are reviewed in chapter 1. In chapter 2 Markov models for the fading radio channel are introduced. Then the transition probabilities are derived. Two methods are proposed namely as an approximation, where level crossing rates are used to calculate the transition probabilities and as a more realistic model. For this model estimators are proposed to calculate the transition probabilities from a pattern processes gained by digital filters. In chapter 3 similar methods are derived for Semi-Markov models. In chapter 4 processes from these models are compared to random paths gained from discrete digital filters.

1 Classical Modelling of Fading Radio Channels

For the purpose of stochastic simulation the fading processes can be generated by quadratic addition of filtered independent white Gaussian noise sequences. In Fig. 1 the basic setup to generate the processes is described. After filtering, the two zero mean processes are associated with the inphase and quadrature components of the fading signal. Due to [8] the noise equivalent bandwidth of the filters is

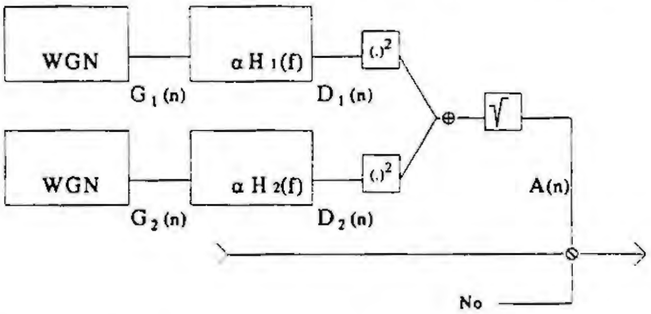


Figure 1: Model to generate Rayleigh fading processes

$$B_x = \frac{1}{H_x(0)} \int_0^\infty H_x(f)df \qquad x = 1, 2$$

Assuming for the transfer function $H_x(0) = 1$ and for the deviation of $\{G_x(n)\}$ $\sigma_{G_x} = 1$ the autocorrelation functions of the processes $\{D_x(n)\}$ are described by

$$C_{D_x D_x}(\tau) \circ\!\!-\!\!\bullet \frac{1}{2B_x}|H_x(f)|^2 \qquad x = 1, 2$$

For non-frequency selective fading the transmitted signal is multiplied by the fading amplitude. With a modified structure [9] [7] it is possible to generate asymmetrical spectra which are observed if the angle of the radio waves reaching the receiver is not uniformly distributed.

Non-Gaussian density functions are usually generated by algebraic operations on Gaussian processes. Rician fading is obtained by adding a constant factor to one of the branches [8]. It should be mentioned that usually the correlation is modified by these algebraic operations. A second method to generate correlated sequences is described in [5], pp 67. Algebraic algorithms are used to build a quasi deterministic channel with the desired correlation properties. Because several branches are needed, this method seems not to be numerical efficient.

Often channel state models are used for investigations in error control algorithms. Each state is related to a certain symbol error [1]. Calculating the transition probabilities of the

state model requires some effort, but the main problem is that with changing the scenario, e.g. the velocity of the station, the probabilities must be calculated again. In chapter 5 algorithms are proposed to avoid this problem and reduce the numerical effort.
Fading processes generated by the model outlined in Fig. 1 are usually denoted as ARMA processes [3]. They can be approximated by state models. Using a Markov chain needs lower numerical effort then using digital filters. In the following a brief review of Markov models is given.

2 Markov Models

Let $S = \{S_0, S_1, S_2 \cdots, S_K\}, K \in I\!N$ define a finite set of states and $\{S_n\}, n \in I\!N_0$ be a homogeneous discrete-time Markov process. The transition probabilities

$$p_{i,j} = \Pr\{S_{n+1} = j | S_n = i\}$$

are independent from index n. The equilibrium state probabilities can be described in

$$\nu = \nu \mathrm{p}$$

with $\nu = (\nu_i)_{i=1..K}$ and $\mathrm{p} = (p_{i,j})_{i,j=1..K}$. The transition probabilities can be determined by observing the original fading process $\{A(n)\}$. Let $A = \{A_1, A_2, A_3 \cdots, A_K\}$ define a set of thresholds with $A_i < A_{i+1}, i = 1..K$. Thus, $\{S_n\}$ is said to be in state S_i at time instant k if

$$A_i < A(n) < A_{i+1} \quad \text{for } n \in N_0,\, i = 1..K$$

Due to this algorithm a state S_i is a mapping of a set of values of the original process.
A transition $i \rightarrow j$ of the process $\{S_n\}$ means crossing the levels

$$A_l \quad \text{for all } l = i..j+1 \text{ if } i > j \qquad \text{and} \qquad A_m \quad \text{for all } m = j..i+1 \text{ if } i < j.$$

Thus, an approximation can be made.

2.1 Approximation of the Transition Probabilities

Extending [6] to a model consisting of more than only two states the transition probabilities can be approximated by level crossing rates (LCR) which can be calculated from second order statistics. In [14] a similar approach is presented. The following assumption is made: The original fading process $\{A(n)\}$ is assumed to be slow enough or the discrete-time parameter of the Markov chain is short enough so that the process only crosses one threshold. Thus, consecutive values are assumed to be neighbouring states. That means $p_{i,j} = 0$ for all $|i-j| > 1$. Then the transition probabilities of the Markov chain can be derived approximately by using the level crossing rate [5].

$$\begin{aligned}
p_{i-1,i} &= \frac{N_l(A_i)}{R_t(F(A_i) - F(A_{i-1}))} && i = 2..K \\
p_{i,i-1} &= \frac{N_l(A_i)}{R_t(F(A_{i+1}) - F(A_i))} && i = 1..(K-1) && p_{0,1} = \frac{N_l(A_1)}{R_t F(A_1)} \\
p_{i,i} &= 1 - p_{i,i-1} - p_{i,i-2} && i = 2..(K-1) && p_{0,0} = 1 - p_{0,1} \\
p_{K,K-1} &= \frac{N_l(A_K)}{R_t(1 - F(A_K))} \\
p_{K,K} &= 1 - p_{K,K-1} \\
p_{K-1,K-1} &= 1 - p_{K-1,K} - p_{K-1,K-2}
\end{aligned} \tag{1}$$

where

$$N_l(A_i) = \sqrt{\frac{2\pi A_i}{\rho}} f_d \exp\left\{-\frac{A_i}{\rho}\right\}$$

denotes the level crossing rate and R_t the sample rate. $F(A_i)$ denotes the stationary probability function of the process $\{A(n)\}$ at the threshold A_i, f_d represents the Doppler frequency which is $f_d = \frac{v}{\lambda}$. The velocity of the station is v, the wavelength is λ. Due to this algorithm **P** is a band diagonal matrix.

2.2 Estimation of the Transition Probabilities

In [3] the maximum likelihood estimator for the transition probabilities is derived. If $n_{i,j}$ denotes the number of observed transitions of the original process from state i to state j and n_i denotes the total number of transitions leaving state i, then the estimator is

$$\hat{p_{i,j}} = \frac{n_{i,j}}{n_i} \quad i,j = 1..K.$$

3 Semi-Markov Models

Using a discrete-state discrete-time Markov model implies a geometrical distribution of time the process remains in a state without changing to another one. Let this time be the sojourn-time. The probability of remaining in state i for k time intervals and changing then to j is

$$p_{i,j}(k) = {p_{i,i}}^k \, p_{i,j}.$$

The assumption of a geometrical sojourn-time distribution function is not valid in general. Processes with a general distribution are more favourable. Due to [3] a stochastic process is denoted as a homogeneous discrete-time Semi-Markov process if the sojourn-time distribution densities

$$q_{i,j}(k) = \Pr\{S_{n+1} = j, T_{n+1} - T_n = k | S_n = i\}$$

are independent of n. With the sojourn-time distribution functions $Q_{i,j}(m) = \sum_{k=1}^{m} q_{i,j}(k)$ the transition probabilities of the embedded Markov chain are $q_{i,j} = Q_{i,j}(\infty)$. The Semi-Markov kernel is the matrix $\mathbf{q}(k) = (q_{ij}(k))_{i,j=1..K}$. Estimators to calculate $q_{i,j}(k)$ can be derived in a similar way as described in chapter 2.2.

4 Realisation of the Models and Results

A digital filter for simulating fading channels is usually realized in the time domain, see [8] [15]. The generation of pattern sequences for setting up a Markov model is also possible by filtering in the frequency domain because only sequences of a certain length are needed. For our models Gaussian noise has been filtered with a second order lowpass. Rayleigh fading is obtained by the model outlined in Fig. 1.

Comparing the three models is possible by measuring the correlation of the fading processes, or the power spectra, respectively. This type of correlation is denoted in [2] as *global* correlation and the *local* correlation is introduced. It can be shown, that the first order local correlation of models derived from level crossing rates equals the level crossing rate itself related to a time unit [11].

The model derived from level crossing rates (LCR-Approximation) is easy to parameterize, but there is no possibility to influence the exact shape of the spectrum of the process, or autocorrelation respectively.

The other two models (Markov-Model, Semi-Markov-Model) parameterized according to the filter process can be influenced in their correlation properties by the original filter process itself.

In this paper two additional measures to characterize the correlation properties of the processes shall be proposed. Let $Fo(a,k)$ be a measure for the interval's length a process $\{S_n\}$ is above a certain level a. The density $fo(a,k)$ is given by the probability that the values of the process are greater than a for a sequence of k samples

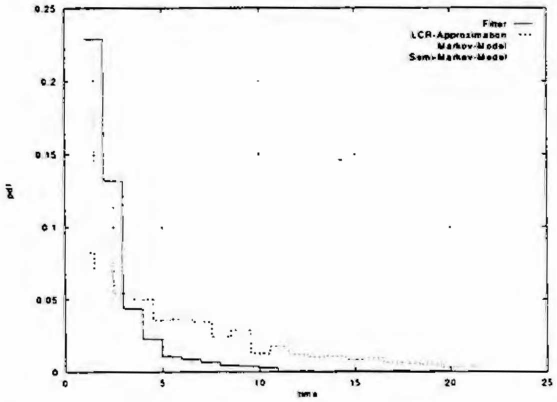
$$fo\,(a,k) := \Pr\left\{S_n > a,\ S_{n+1} > a,\ \cdots\ S_{n+k} > a,\ S_{n+k+1} \leq a\,|S_{n-1} \leq a\right\}.$$

The probability density of a process $\{S_n\}$ is below a level a can be defined in a similar way. These measures are interpreted as the probability density function (pdf) of fading and non-fading intervals. Let $So(t_o,t_u)$ be a measure for the length of time a process $\{S_n\}$ moves in a state space limited by two levels t_o and t_u and changes then to a value larger than t_o. The respective density is

$$\begin{aligned} so\,(t_o,t_u,k) := \Pr\{&t_u \leq S_n < t_o,\ t_u \leq S_{n+1} < t_o,\ t_u \leq S_{n+2} < t_o,\ \cdots t_u \leq S_{n+k} < t_o, \\ &S_{n+k+1} > t_o\,|\,(S_{n-1} < t_u \ \vee\ S_{n-1} \geq t_o)\}\,. \end{aligned}$$

A similar measure with respect to a change to lower values can be defined also.

In Figs. 2 to 5 the measured probability density functions are depicted. For the sojourn-time pdf the two thresholds are 0.8 and 1.0. For the fading pdf the threshold is 1.8. According to the measures sojourn-time pdf the Semi-Markov model fits best the original process represented by the curve denoted 'Filter'. The mean square distances of the measured sojourn-pdf of the three models related to a process generated by filters ($f_g/f_{sample} = 0.006$) are listed in the following table.



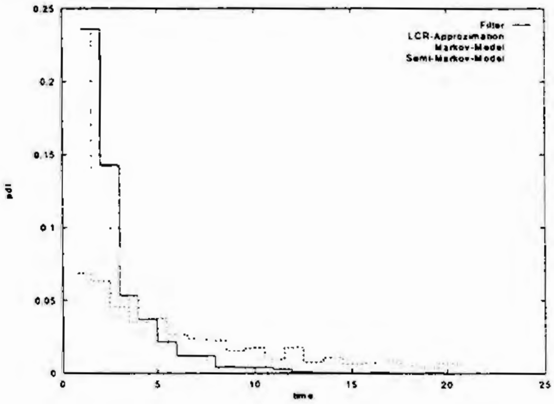


Figure 2: Pdf of Sojourn-Time moving down

Figure 3: Pdf of Sojourn-Time moving up

thres-hold		M-Model		SM-Model		LCR-Model	
		up	down	up	down	up	down
0.6	0.8	36.67	54.81	12.78	6.08	1210.76	1434.40
0.8	1.0	16.77	83.24	7.95	0.86	1133.19	1762.70
1.0	1.2	11.67	21.98	18.20	4.84	1382.88	1422.83
1.2	1.4	25.01	59.74	16.38	32.13	1481.95	1686.01

Considering a slowly varying process (which means a low Doppler frequency related to the sample frequency) the LCR-Model approximates the Markov-Model well but there is a significant sojourn-time in any state. Thus, in this case the Semi-Markov-Model is better

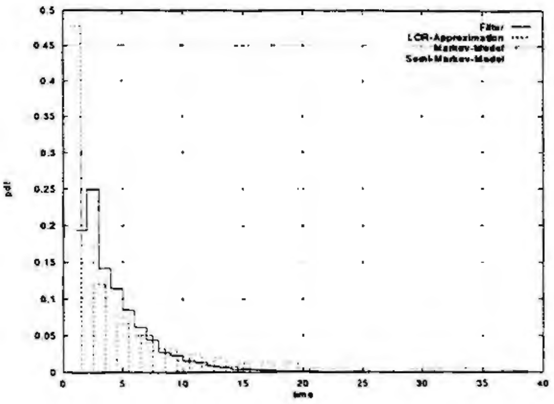


Figure 4: Pdf of Fading Intervals

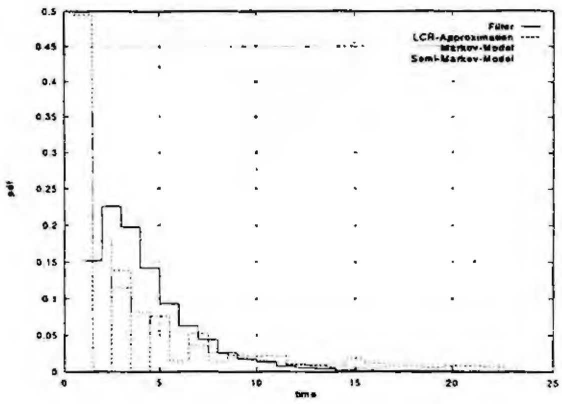


Figure 5: Pdf of Non-Fading Intervals

than the two Markov models. Rapid fading processes do not have a significant sojourn-time in any state. Thus, the Semi-Markov-Model is approximated well by the Markov-Model. The LCR-Model does not fit the original very well, because only transitions to neighbouring states are allowed.

The measures fading/non-fading pdf show that no one of the three model processes fit the original process represented by the curve denoted 'Filter' well, but the LCR-Model is the worst one, see Fig. 4 and 5.

5 Hybrid Models and Network Simulation

In mobile radio scenarios the fading processes are usually not modelled by a single fading process. Often a superposition of several processes is necessary. Two examples shall be given. The signal of each interferer is considered to be uncorrelated with the receivers signal and it's signal power is added at the receiver together with Gaussian noise. The resulting process of the signal-to-interference-ratio is determined by all components especially by all the interferers, Eq. 2.

$$S_{sir} = \frac{S_{P_t r}}{\sum_{i=1}^{M} S_i + S_N} \tag{2}$$

Another scenario for a hybrid process is the superposition in a Suzuki process. The fast Rayleigh fading process and the Lognormal process are superimposed there [7]. With the proposed state models a comfortable simulation structure can be derived. Each process is modelled by it's parameter-discrete approximation. After mapping the state to it's signal value these values are combined as in the original parameter-continuous process.

For network simulation, the influence of each station with a relevant amount of disturbance in respect to a certain receiver can be modelled by a finite state model. The mean value of the state model is normalized to one. The actual mean of the process, determined from the distance of the station to the receiver, is adjusted during the simulation run. By that independence of the processes from variations of the topology is gained. The transition matrixes of the models need not be calculated scenario specific.

Independence from the velocity of a certain station (which determines the width of the fading spectrum) is realized during a simulation run by sampling the original fading process according to the highest velocity. Then the actual velocity can be realized by interpolation. The goal should be to avoid calculating new transition probabilities during a simulation run.

6 Conclusion

Three finite state models for fading processes have been discussed and compared with fading processes generated from a digital filter model.

Two measures have been defined and used to compare the results of the models. According to these measures the Semi-Markov model fits best the original process. Suggestions have been made how to use the derived models for simulation including the mobile radio channel. With this proposals there is a possibility to consider a whole network (ISO/OSI-layer 1 to 3) including the effects of the radio channel. Further investigations are necessary to define classes of relevant fading spectra due to characteristic error patterns.

References

[1] J.P. A. Adoul, et al. *A Critical Statistic for Channels with Memory.* IEEE Trans. Inform. Theory, Vol. IT-18, No. 1, pp. 133–141, January 1972.

[2] W. Ding. *Korrelierte Zufallsprozesse in Wartesystemen von Kommunikationsnetzen.* Dissertation, RWTH Aachen, 1991.

[3] L. Fahrmeir, H. Kaufmann, O. Friedemann. *Stochastische Prozesse.* Hanser Verlag München Wien, 1981.

[4] B. Fleury. *Charakterisierung von Mobil- und Richtfunkkanälen mit schwach stationären Fluktuationen und unkorrelierter Streuung.* Dissertation, ETH Zürich, 1990.

[5] W.C Jakes. *Microwave Mobile Communications.* Wiley, New York, 1974.

[6] L. Kittel. *Analoge und diskrete Kanalmodelle für die Signalübertragung beim beweglichen Funk.* Frequenz, Vol. 36, pp. 153–160, 1982.

[7] A. Krantzik, D. Wolf. *Statistische Eigenschaften von Fadingprozessen zur Beschreibung eines Landmobilfunkkanals.* Frequenz, Vol. 44, No. 6, June 1990.

[8] E. Lutz, E. Plöchinger. *Generating Rice Processes with Given Spectral Properties.* In *IEEE Transactions on Vehicular Technology,* Vol. VT-34, pp. pp. 178–181, April 1985.

[9] H. W. Schüssler, et al. *A Digital Frequency-Seletive Fading Simulator.* Frequenz, Vol. 43, No. 2, February 1989.

[10] B. Sklar. *Digital Communications.* Prentice-Hall, New Jersey, 1988.

[11] H. Steffan. *Finite-state Radio Channel Models and their Properties.* submitted to Aachener Kolloquium Mobile Kommunikationssysteme, 1994.

[12] S. Takuro, et al. *Simulation of Burst Error Models and an Adaptive Error Control Scheme for High Speed Data Transmission Over Analog Cellular Systems.* IEEE Vehicular Technology Conference, Vol. 40, No. 2, pp. 443–451, May 1991.

[13] R. Tetzlaff, D. Wolf. *On the Distribution of Level-Crossing Time-Intervals for Gaussian Random Processes.* Archiv der Elektrischen Übertragung, Vol. AEÜ-45, No. 4, pp. 203–209, July 1991.

[14] H. S. Wang, N. Moayeri. *Modeling, Capacity, and Joint Source/Channel Coding for Rayleigh Fading Channels.* IEEE Vehicular Technology Conference, Vol. 42, pp. 473–479, 1993.

[15] M. Werner. *Bit Error Correlation in Rayleigh-Fading Channels.* Archiv der Elektrischen Übertragung, Vol. AEÜ-45, No. 4, pp. 245–253, July 1991.

Optimal Channel Allocation in Quasi–Linear and Quasi–Cyclic Cellular Radio Networks

Rudolf Mathar and Jürgen Mattfeldt
Institut für Statistik, RWTH Aachen
Wüllnerstraße 3, D-52056 Aachen, Germany

Abstract — We investigate channel assignment problems for two general classes of cellular radio networks. The blocking probability of a network is chosen as optimization criterion. Two divide–and–conquer algorithms are presented which for both types of networks determine an optimal allocation of one channel. By iteratively applying these algorithms an abitrary number of channels may be allocated efficently. This method yields a blocking probability close to the optimum.

1. Introduction

A growing demand for mobile radio communication is expected in the near future, particularly for urban areas with high traffic density. Thus, carefully designing channel allocation to accommodate a maximum of calls becomes one of the most challenging tasks for cellular radio networks. The available channels have to be allocated to base stations in such a way that channels in use at neighbouring stations do not interfere. The system can be described by an interference graph in which the nodes represent cells, and linked nodes indicate neighbouring cells, which must not use the same channel simultaneously.

Fixed channel assignment (FCA) and dynamic channel assignment (DCA) are two basic methods for allocation. At low traffic intensities DCA is more flexible and has smaller blocking probability, while for heavy load conditions FCA seems to be superior. This is a result of Kelly [7] for simple linear networks and maximum packing, introduced by Everitt and Macfadyen [3]. A mixed form of both strategies, hybrid channel assignment (HCA), tries to sidestep the disadvantages by allocating a certain subset of channels fixed and the others dynamically. In general, the analysis of DCA strategies seems to be very difficult. Analytic results are known only for special cases, e.g., constant traffic intensity over all cells and linear interference patterns with direct neighbours. Investigations of the performance of certain DCA algorithms are often based on simulation and comparison with the corresponding optimal fixed channel assignment.

This shows that optimal FCA is an important question for cellular radio networks. Unfortunately, even fixed channel assignment is NP–hard in general. For the subclass of linear networks there are efficient algorithms, compare [8]. In this paper we cover wider classes of networks which allow the efficient calculation of an optimal channel design. The purpose of this approach is threefold: (1) for many real world problems the class of investigated networks may serve at least as an approximation, (2) the performance of different DCA algorithms may be compared to an achievable bound, and (3) we obtain a class of realistic scenarios to test heuristic procedures developed for general FCA (e.g., based on stochastic optimization [2], [8]).

Section 2 illustrates in detail our model to tackle the channel assignment problem, where in particular traffic intensity may vary over different cells. In section 3 we introduce the class of quasi–linear networks where vertices are linearly ordered such that for each node the largest adjacent node is less than or equal to the largest adjacent node of its next largest neighbour. A slightly adapted definition applies for quasi–cyclic networks which are investigated in section 4.

This work was supported by the Deutsche Forschungsgemeinschaft under grant no. Ma 1184/2-4.

2. Model Assumptions

For ease of reference we start with some basic definitions, which will be important later on.

Definition 1. *The pair (V, E) with a finite set V of nodes (vertices) and a set of edges $E \subset V^{(2)} = \{e \subset V;\, |e| = 2\}$ is called a (simple) graph. A subset $W \subset V$ is called independent, if $W^{(2)} \cap V = \emptyset$ holds.*

In the following we investigate the problem of assigning N channels, numbered $1, \ldots, N$, to a network $\mathcal{Z} = \{Z_1, \ldots, Z_z\}$ of z cells such that a benefit function is minimized under certain technical constraints. The notion 'channel' has different meanings, depending on the particular technology. In FDMA–systems a 'channel' may be identified with its carrier signal, in TDMA–systems a group of logical time slots using the same frequency is combined to a 'channel'.

Because of interference, neighbouring cells may not use the same channel. In general, these constraints are represented by a so called interference graph $G_{\mathcal{Z}} = (V, E)$ with

$$V = \{Z_1, \ldots, Z_z\}, \quad \text{and} \quad E = \big\{\{Z_i, Z_j\}\,;\, \text{interference between } Z_i \text{ and } Z_j \text{ may happen}\big\}.$$

In this paper we restrict our attention to co–channel–interference, i.e., interference may only occur between channels of the same frequency.

The assignment of channels to cells is described by binary matrices in the following way.

Definition 2. *A matrix $M_N = (m_{ij})_{i,j=1}^{z,N} \in \{0,1\}^{z \times N}$ is called an N–channel design for an interference graph $G_{\mathcal{Z}}$, if each column of M_N induces an independent set in $G_{\mathcal{Z}}$, i.e., the sets $W_j = \{Z_i\,;\, m_{ij} = 1, i = 1, \ldots, z\}$ are independent in $G_{\mathcal{Z}}$ for all $j = 1, \ldots, N$.*

$m_{ij} = 1$ means that the jth channel is allocated to cell Z_i. The number a_i of channels available for cell Z_i under design M_N is given by

$$a_i = \sum_{j=1}^{N} m_{ij}, \; i = 1, \ldots, z. \tag{1}$$

Definition 3. *An N–channel design $M_N = (m_{ij})_{i,j=1}^{z,N}$ is called admissible, if there is no N–channel design $M'_N = (m'_{ij})_{i,j=1}^{z,N}$ such that $m'_{ij} \geq m_{ij}$ for all i, j and $M'_N \neq M_N$. Otherwise M_N is called inadmissible.*

The performance of channel designs is ranked by their overall blocking probability for the whole network. For this purpose we assign to each cell Z_i a discrete probability measure f_i with support N. The blocking probability $B_{Z_i}(a_i)$ for cell Z_i with a_i allocated channels is calculated according to

$$B_{Z_i}(a_i) = 1 - \sum_{k=1}^{a_i} f_i(k). \tag{2}$$

This representation allows for modeling unbalanced traffic intensity in different cells, which is usually encountered. Extreme cases are $a_i = 0$, when the blocking probability is 1, and $a_i \to \infty$ which entails $B_{Z_i}(a_i) \to 0$, since $\sum_{k=1}^{\infty} f_i(k) = 1$. $f_i(k)$, $k \in \mathbb{N}$, describes the decrease of blocking probability in Z_i if an extra channel is added to $k - 1$ present. This notion will turn out to be very convenient w.r.t. the subsequent algorithmic approaches.

The overall blocking probability $\Im(M_N)$ of $\mathcal{Z}$ using M_N is thus calculated as

$$\Im(M_N) = \sum_{i=1}^{z} w_i B_{Z_i}(a_i) = 1 - \sum_{i=1}^{z} w_i \sum_{j=1}^{a_i} f_i(j), \tag{3}$$

where each cell Z_i is assigned a certain weight $w_i \geq 0$, expressing the relative importance of satisfying calls in this cell. In our model we neglect roaming and handover between neighbouring cells. Investigations of Everitt und Macfadyen give reasons for this widely accepted assumption; these effects have nearly no influence on the blocking probability [4].

A further aspect is mobile–to–mobile–communication. If this type of communication makes up less than 20% of load it may be ignored [4]. Blocking probabilities of magnitude 2–4% are tolerable for a smooth operation of the network [12].

In the following we will look for an N–channel design which minimizes the benefit function $\Im$, or equivalently maximizes $\sum_{i=1}^{z} w_i \sum_{j=1}^{a_i} f_i(j)$. Obviously, we may restrict our attention to the class of admissable designs. The following example deals with an important instance of blocking probabilies and weights.

Example. Traffic load λ_i (in erlangs) is assumed in each cell Z_i. If arriving calls in each cell are described by independent Poisson processes with arrival rate ν_i, and the corresponding service times by independent random variables with distribution G_i having finite expectation α_i, then the stationary blocking probability $B_{Z_i}(a_i)$ in cell Z_i is obtained by the Erlang–B–formula [13]

$$f_i(k) = E_{k-1}(\lambda_i) - E_k(\lambda_i), \quad E_k(\lambda_i) = \frac{\lambda_i^k/k!}{\sum_{j=0}^{k} \lambda_i^j/j!}, \quad \lambda_i = \nu_i \alpha_i,\ k \in \mathbf{N}. \tag{4}$$

Without priorities among calls, weights proportional to λ_i seem to be reasonable. After normalization we have $w_i = \lambda_i / \sum_{k=1}^{z} \lambda_k$, $i = 1, \ldots, z$. These weights have been suggested in [14], [15].

We are now prepared to introduce the **co**–**C**hannel **A**ssignment **P**roblem in a complexity theoretical way.

coCAP(N), $N \in \mathbf{N}$:

Instance: A network $\mathcal{Z} = \{Z_1, \ldots, Z_z\}$, rational cell parameters $(f_i, w_i)_{i=1}^{z}$, and an interference graph $G_{\mathcal{Z}}$.

Problem: Determine an N–channel design M_N^* with $\Im(M_N^*) \leq \Im(M_N)$ for all N–channel designs M_N.

coCAP(N) is a generalization of graph colouring, and thus NP–hard, see [8]. In particular, no efficient algorithm is known yielding an adequat approximate solution of coCAP(N) in polynomial time.

3. Quasi–Linear Networks

The next two sections deal with two general classes of cellular networks for which coCAP(1) is efficiently solvable. Iterative application of the corresponding algorithm leads to heuristics well suited for coCAP(N), $N > 1$.

Definition 4. *Given an interference graph $G_{\mathcal{Z}} = (\mathcal{Z}, E)$, the set of right and left neighbours of $Z_i \in \mathcal{Z}$ is defined as*

$$\mathcal{R}(Z_i) = \{Z_j \in \mathcal{Z}\,;\, i < j,\ Z_i \text{ and } Z_j \text{ are adjacent}\},$$
$$\mathcal{L}(Z_i) = \{Z_j \in \mathcal{Z}\,;\, j < i,\ Z_i \text{ and } Z_j \text{ are adjacent}\}.$$

With $D(Z_i) = |\mathcal{R}(Z_i)| + |\mathcal{L}(Z_i)|$ we define the maximum bandwidth of the network $\mathcal{Z}$ as $\Delta(\mathcal{Z}) = \max_{1 \leq i \leq z} D(Z_i)$.

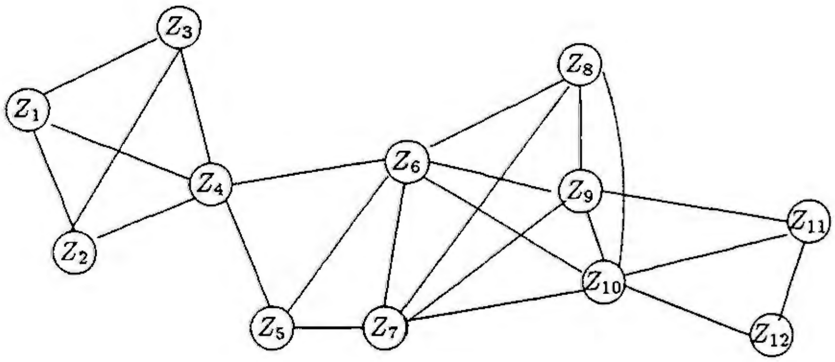


Fig. 1. Interference graph of a quasi-linear network ($k = 4$)

Definition 5. a) *A network $\mathcal{Z}$ is called quasi-linear with reuse distance $k+1$, if for all $Z_i \in \mathcal{Z}$, $1 \le i \le z-1$, there are integers $r_i \ge i$ satisfying*

- $\mathcal{R}(Z_i) = \{Z_j \in \mathcal{Z}\,;\, i < j \le r_i\}$,
- $r_i \le r_{i+1},\ i = 1, \ldots, z-2$,
- $\max_{1 \le i \le z-1}\{r_i - i\} = k$.

b) *A quasi-linear network $\mathcal{Z}$ is called linear with reuse distance $k+1$, if $z > k$ and if the numbers r_i from* a) *satisfy $r_i = k+i,\ i = 1, \ldots, z-k$, and $r_i = z,\ i = z-k+1, \ldots, z-1$.*

Remark. a) Quasi-linear networks may be characterized analogously via left neighbours and integers l_i, $2 \le i \le z$, satisfying

- $\mathcal{L}(Z_i) = \{Z_j \in \mathcal{Z}\,;\, \ell_i \le j < i\}$,
- $\ell_i \le \ell_{i+1},\ i = 2, \ldots, z-1$,
- $\max_{2 \le i \le z}\{i - \ell_i\} = k$.

b) In the following we set for quasi-linear networks $r_z = z$ und $\ell_1 = 1$.

c) Whether a network is quasi-linear depends on a the numbering of nodes. Given some interference graph $(\mathcal{Z}, E)$ and $K \in \mathbf{N}$, the problem of deciding if there exists an appropriate renumbering of nodes such that bandwidth $\Delta(\mathcal{Z}) \le K$ is achieved, is NP-complete (see [6], p. 200).

Fig. 1 represents the interference graph of a special quasi-linear cellular network. The simplest case of a linear network (reuse distance $k = 1$ or 2) are investigated for DCA in [1], [5], [7], [11]. All these papers suppose homogeneous traffic load $\lambda_i = \lambda$ which allows to solve coCAP easily. FCA and DCA may then be compared by simulation.

Due to a representation as generalized Fibonacci numbers [9], even for linear networks the number of admissible 1-channel designs increases exponentially with the number of cells z. However, there is an efficient algorithm to solve coCAP(1) for quasi-linear networks.

Theorem 1. *For any quasi-linear cellular network $\mathcal{Z}$ with reuse distance $k+1$* coCAP(1) *can be solved in $O(z^{2+\log_2(\Delta(\mathcal{Z})+1)})$ steps.*

Proof. Without loss of generality we assume that $G_{\mathcal{Z}}$ is connected such that $r_i > i$ for all $i = 1, \ldots, z-1$. For a 1-channel design $P_{1,z} = (p_1, \ldots, p_z)^{tr}$, $p_i \in \{0,1\}$, $1 \le i \le z$, by $P_{i,j}$, $1 \le i \le j \le z$, we denote the subdesign $(p_i, \ldots, p_j)^{tr}$ of $P_{1,z}$. $P_{i,j}$ stands for the empty design whenever $i > j$. For $i \le k \le j$ designs may be concatenated as $P_{i,j} = P_{i,k}\,P_{k+1,j}$.

Now, let $P^* = (p_1^*, \ldots, p_z^*)$ an admissible optimal 1-channel design and $a \in \{1, \ldots, z\}$. Then there is an index $b \in \{\ell_a, \ldots, r_a\}$ which induces a decomposition of $P^* = P^1\,P^2\,P^3$ in three subdesigns with the following properties.

$$P^1 = P^*_{1,\ell_b-1} \text{ is an optimal 1–channel design for the subnetwork } \{Z_1, \dots, Z_{\ell_b-1}\}. \tag{5}$$

$$P^2 = P^*_{\ell_b,b-1} = (0, \dots, 0)^{tr}. \tag{6}$$

$$P^3 = P^*_{b,z} \text{ is an optimal 1–channel design for the subnetwork } \{Z_b, \dots, Z_z\}, \text{ and } p^*_b = 1 \text{ holds.} \tag{7}$$

$\{Z_1, \dots, Z_{\ell_b-1}\}$ denotes the empty set if $\ell_b - 1 \leq 0$. A decomposition of such type exists for any admissible optimal 1–channel design P^*, because at least one of the numbers $p_{\ell_a}, \dots, p_{r_a}$ is equal to 1, p_b say, since otherwise P^* would not be admissible. Each of the three by b induced subdesigns $P^*_{1,\ell_b-1}, P^*_{\ell_b,b-1}$, and $P^*_{b,z}$ has properties (5) – (7), since otherwise P^* would not be optimal. Observe that the optimality of P^1 und P^3 strongly depends on the particular structure of quasi–linear networks.

Let $t_k(z)$ denote the calculation time (proportional to the number of steps) to determine an optimal 1–channel design for a quasi-linear network of reuse distance $k+1$ and z cells in the worst case. We set $t_k(z) = 0$ when $z \leq 0$. The above decomposition yields the following bound

$$t_k(z) \leq \sum_{b=\ell_a}^{r_a} \big(t_k(\ell_b - 1) + t_k(z - b + 1)\big).$$

Obviously, $t_k(z)$ is increasing in z, such that

$$t_k(z) \leq \sum_{b=\ell_a}^{r_a} 2\max\{t_k(\ell_{r_a} - 1), t_k(z - \ell_a + 1)\} \leq 2(\Delta(\mathcal{Z}) + 1)\max\{t_k(\ell_{r_a} - 1), t_k(z - \ell_a + 1)\}.$$

A good upper bound is obtained by choosing a in such a way that the numbers $\ell_{r_a} - 1 \leq a+k-2$ and $z - \ell_a + 1 \leq z - (a-k) + 1$ are as close as possible to the same value. $a+k-2 = z-(a-k)+1$ leads to $a = \lfloor \frac{z}{2} \rfloor + 2$. By induction we obtain the following upper bound

$$\begin{aligned}
t_k(z) &\leq 2\,(\Delta(\mathcal{Z}) + 1)\, t_k\big(\lfloor \tfrac{z}{2} \rfloor + k + 1\big) \leq 2^2\,(\Delta(\mathcal{Z}) + 1)^2\, t_k\big(\lfloor \tfrac{\lfloor z/2 \rfloor + k + 1}{2} \rfloor + k + 1\big) \\
&\leq 2^2\,(\Delta(\mathcal{Z}) + 1)^2\, t_k\big(\lfloor \tfrac{z}{4} + \tfrac{3}{2}\,(k+1) \rfloor\big) \\
&\;\;\vdots \\
&\leq 2^{\lfloor \log_2(z) \rfloor}\,(\Delta(\mathcal{Z}) + 1)^{\lfloor \log_2(z) \rfloor}\, t_k\Big(\Big\lfloor \frac{z}{2^{\lfloor \log_2(z) \rfloor}} + \frac{2^{\lfloor \log_2(z) \rfloor} - 1}{2^{\lfloor \log_2(z) - 1 \rfloor}}\,(k+1) \Big\rfloor\Big) \\
&\leq z^2\, z^{\log_2(\Delta(\mathcal{Z})+1)}\, t_k(2k+4) = O\big(z^{2+\log_2(\Delta(\mathcal{Z})+1)}\big).
\end{aligned}$$

$\blacksquare$

Observe that $\Delta(\mathcal{Z}) \leq 2k$ for quasi-linear networks with reuse distance $k+1$, therefore the upper bound in Theorem 1 is polynomial in z. The following result is an immediate consequence of Theorem 1. The proof is based on the fact that $\Delta(\mathcal{Z}) = 2k$ for any linear network with reuse distance $k+1$.

Corollary 1. [8] *For any linear cellular network $\mathcal{Z}$ with reuse distance $k+1$* coCAP(1) *can be solved in $O(z^{2+\log_2(2k+1)})$ steps.*

Following the lines of the above proof makes clear how to construct a corresponding algorithm to solve coCAP(1) for quasi–linear networks. To get an efficient implementation, calculating the decomposition (5) – (7) should be preceded by determining $P^3 = P^*_{b,z}$. For if $p^*_b = 0$, then $P^1 = P^*_{1,\ell_b-1}$ need not be calculated any more since P^* is cannot be optimal. That is the reason why the expected calculation time is significantly smaller than the worst–case bound of Theorem 1.

4. Quasi–Cyclic Networks

The idea of quasi–linear networks may be applied to certain cyclic networks. We first transfer Definitions 4 and 5.

Definition 6. *Given an interference graph $G_{\mathcal{Z}} = (\mathcal{Z}, E)$, the set of cyclic right and left neighbours of $Z_i \in \mathcal{Z}$ is defined as*

$$\mathcal{R}_{zyk}(Z_i) = \{Z_j \in \mathcal{Z}\,;\, (j-i) \bmod z \leq \lfloor (z-1)/2 \rfloor,\ Z_i \text{ and } Z_j \text{ are adjacent}\},$$

$$\mathcal{L}_{zyk}(Z_i) = \{Z_j \in \mathcal{Z}\,;\, (i-j) \bmod z \leq \lfloor z/2 \rfloor,\ Z_i \text{ and } Z_j \text{ are adjacent}\}.$$

With $D_{zyk}(\mathcal{Z}_i) = |\mathcal{R}_{zyk}(Z_i)| + |\mathcal{L}_{zyk}(Z_i)|$ we define the maximum and minimum bandwidth, respectively, as $\Delta_{zyk}(\mathcal{Z}) = \max_{1\leq i\leq z} D_{zyk}(Z_i)$, and $\delta_{zyk}(\mathcal{Z}) = \min_{1\leq i \leq z} D_{zyk}(Z_i)$.

Definition 7. a) *A network $\mathcal{Z}$ is called quasi–cyclic with reuse distance $k+1$, if for all $Z_i \in \mathcal{Z}$ there exist integers r_i satifying*

- $\mathcal{R}_{zyk}(Z_i) = \{Z_j \in \mathcal{Z}\,;\, i \neq j \text{ und } (j-i) \bmod z \leq r_i\}$,
- $r_{(i \bmod z)+1} \leq r_{((i+1) \bmod z)+1}$, $i = 1, \ldots, z$,
- $\max_{1\leq i\leq z}\{(r_i - i) \bmod z\} = k$.

b) *A quasi–cyclic network $\mathcal{Z}$ is called cyclic with reuse distance $k+1$, if $z > 2k$ and if the numbers r_i from* a) *satisfy $(r_i - i) \bmod z = k$, $i = 1, \ldots, z$.*

Fig. 2 shows the interference graph of a cyclic network with reuse distance 3. Applying the algorithm for quasi–linear networks iteratively yields an efficient solution of coCAP(1) for quasi–cyclic networks .

Theorem 2. *For any quasi–cyclic network $\mathcal{Z}$ with reuse distance $k+1$* coCAP(1) *can be solved in $O\big((\delta_{zyk}(\mathcal{Z})+1)\,(z - \delta_{zyk}(\mathcal{Z}) - 1)^{2+\log_2(\Delta_{zyk}(\mathcal{Z})+1)}\big)$ steps.*

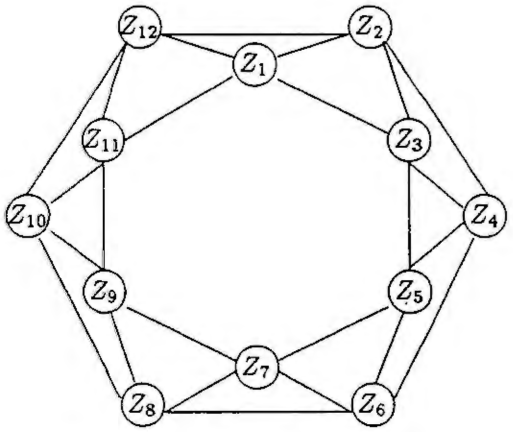


Fig. 2. Interference graph of a a cyclic network ($k = 2$)

Proof. Let i_0 satisfy $|\mathcal{R}_{zyk}(Z_{i_0})|+|\mathcal{L}_{zyk}(Z_{i_0})| = \delta(\mathcal{Z})$. For any admissible optimal 1–channel design $P^* = (p_1^*, \ldots, p_z^*)$ there is an index $a \in \{i_0\} \cup \operatorname{Ind}(\mathcal{L}_{zyk}(Z_{i_0})) \cup \operatorname{Ind}(\mathcal{R}_{zyk}(Z_{i_0})) = A$ with $p_a^* = 1$, where for some $\mathcal{U} \subset \mathcal{Z}$ the set $\operatorname{Ind}(\mathcal{U})$ denotes the indices of cells contained in $\mathcal{U}$. Optimal channel designs may be determined as follows. Set $p_a^* = 1$, one after the other for all indices in A, and calculate some admissible optimal 1–channel design for the quasi–linear network $\mathcal{Z}_a = \mathcal{Z} \setminus \big(\{Z_a\} \cup \mathcal{L}_{zyk}(Z_a) \cup \mathcal{R}_{zyk}(Z_a)\big)$ using the interference graph of $\mathcal{Z}$ restricted to $\mathcal{Z}_a$. Obviously $|\mathcal{Z}_a| \leq z - \delta_{zyk}(\mathcal{Z}) - 1$ holds. By Theorem 1 $O\big((z - \delta_{zyk}(\mathcal{Z}) - 1)^{2+\log_2(\Delta_{zyk}(\mathcal{Z})+1)}\big)$ steps are necessary since $\Delta(\mathcal{Z}_a) \leq \Delta_{zyk}(\mathcal{Z})$. The optimal design is obtained by comparing the $\delta_{zyk}(\mathcal{Z}) + 1$ channel designs thus calculated. In summary, the cumulated number of steps with this procedure is $O\big((\delta_{zyk}(\mathcal{Z})+1)\,(z - \delta_{zyk}(\mathcal{Z}) - 1)^{2+\log_2(\Delta_{zyk}(\mathcal{Z})+1)}\big)$. ∎

Obviously, for quasi–cyclic networks with reuse distance $k+1$ it holds that $\Delta_{zyk}(\mathcal{Z}) \leq 2k$ and $0 \leq \delta_{zyk}(\mathcal{Z}) \leq 2k$. As $\delta_{zyk}(\mathcal{Z}) = \Delta_{zyk}(\mathcal{Z}) = 2k$ for cyclic networks with reuse distance $k+1$, from Theorem 2 we obtain the following result.

Corollary 2. *For any cyclic network $\mathcal{Z}$ with reuse distance $k+1$* coCAP(1) *can be solved in $O\big((2k+1)\,(z-2k-1)^{2+\log_2(2k+1)}\big)$ steps.*

Iterative use of the above algorithms is a reasonable heuristic to solve coCAP(N) for $N > 1$. Though, in general we will not end up with an optimum solution. Let P_N denote a design iteratively calculated, and P_N^* an optimal one. An upper bound of the difference $\Im(P_N) - \Im(P_N^*)$ is obtained along the following lines. Let $I(\mathcal{Z})$ be the maximum number of elements in an independent set of the interference graph $G_{\mathcal{Z}}$, and $b_{(i-1)N+\ell} = w_i \, f_i(\ell)$, $i = 1, \ldots, z$, $\ell = 1, \ldots, N$. By $b_{(1)} \geq b_{(2)} \geq \ldots \geq b_{(Nz)}$ we denote the components of $(b_1, \ldots, b_{Nz})$ in decreasing order. Obviously

$$\Im(P_N) - \Im(P_N^*) \leq \Im(P_N) - \left(1 - \textstyle\sum_{i=1}^{N \cdot I(\mathcal{Z})} b_{(i)}\right)$$

holds, since each channel may be used in at most $I(\mathcal{Z})$ cells. Of course there remains to determine $I(\mathcal{Z})$ which unfortunately is NP–hard for arbitrary graphs. In the quasi–linear and quasi–cyclic case, respectively, $I(\mathcal{Z})$ may be determined effectively by solving coCAP(1) in the special case $f_i(k) = \mathbb{1}_{\{1\}}(k)$, $k \in \mathbb{N}$, and $w_i = 1/z$, $i = 1, \ldots, z$, with the help of the above introduced algorithms. This yields an optimal 1–channel design P_1^* satisfying $I(\mathcal{Z}) = z \, \Im(P_1^*)$.

The bound given above is strict and cannot be improved without further assumptions. Of course, properties of the densities f_i will significantly influence the performance of the iterative approach. Numerical experiments [9] show that decreasing (or unimodal) densities are favourable, as is encountered in the Erlang–B–case [10] (see (4)).

5. References

[1] D.C. Cox and D.O. Reudink. Dynamic channel assignment in high-capacity mobile communication systems. *Bell System Tech. J.*, 50(6):1833-1857, Jul.-Aug. 1971.

[2] M. Duque-Antón, D. Kunz, and B. Rüber. Channel assignment for cellular radio using simulated annealing. *IEEE Veh. Techn.*, 42(1):14–21, Feb. 1991.

[3] D. Everitt and N. Macfadyen. Analysis of multicellular mobile radiotelephone systems with loss. *British Telecom Techn. J.*, 1(2):37–45, Oct. 1983.

[4] D. Everitt and N. Macfadyen. Teletraffic problems in cellular mobile radio systems. *Proc. 11th Int. Teletraffic Congr.*, 1985.

[5] M. Frodigh. Optimum dynamic channel allocation in certain street microcellular radio systems. *IEEE Veh. Tech. Conf.*, 42:658–661, 1992.

[6] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W.H. Freedmann and Co., San Francisco, 1979.

[7] F.P. Kelly. Stochastic models of computer communication systems. *J. Roy. Statist. Soc.*, 47(3):379–395, 1985.

[8] R. Mathar and J. Mattfeldt. Channel assignment in cellular radio networks. *Aachener Informatik-Berichte*, 92–40, 1992 (to appear in IEEE Veh. Tech.).

[9] J. Mattfeldt. Analyse und Optimierung mobiler Kommunikationssysteme. *Ph.D. Thesis, RWTH Aachen*, 1993.

[10] E.J. Messerli. Proof of a convexity property of the Erlang B formula. *Bell System Tech. J.*, 51(4):951-953, Apr. 1972.

[11] L. Schiff. Traffic capacity of three types of common-user mobile radio communication systems. *IEEE Trans. Commun.*, 18:12–21, Feb. 1970.

[12] K.N. Sivarajan and R.J. McEliece. Dynamic channel assignment in cellular radio. *IEEE Veh. Tech. Conf.*, 40:631–637, 1990.

[13] L. Takács. On Erlang's formula. *Ann. Math. Statist.*, 40(1):71–78, 1969.

[14] W. Yue. Analytical methods to calculate the performance of a cellular mobile radio communication system with hybrid channel assignment. *IEEE Trans. Veh. Tech.*, 40(2):453–460, May 1991.

[15] M. Zhang and T.P. Yum. The nonuniform compact pattern allocation algorithm for cellular mobile systems. *IEEE Trans. Veh. Tech.*, 40(2):387–391, May 1991.

Effects of Call–Queueing in a GSM–Network

V. Brass[1)] W. Fuhrmann[1,2)] W.R. Mende[1)]

1) DeTeMobil (FRG), 2) Deutsche Bundespost Telekom (FRG)

Abstract: Mobile communication is characterised by steadily increasing demand. Since in mobile communication networks traffic capacity is limited by the total amount of allocated spectrum, the radio resources have to be used efficiently. It is the aim to assign and operate radio channels in such a way, that the network offers to the user a high quality of service on the one hand and to the network operator a high network performance on the other hand. Therefore the GSM–Network provides queueing/loss operation for the assignment of dedicated traffic channels and the establishment of semi–connections (OACSU, Off Air Call Set–Up) if requests for radio resources by the user cannot be served instantly. This paper will treat the effects of both functions on the dimensioning of the traffic channels and the performance of the signalling channels. Both traffic problems are significant for mobile radio networks.

1 Introduction

The increasing demand for a mobile access to voice and non–voice services has led to the realization of a new generation of digital cellular telecommunication networks. These networks represent an evolutionary step, which is comparable to the transition between the conventional PSTN (Public Switched Telephone Network) and the ISDN. The European GSM (Global System for Mobile Communications) networks have been the first representatives which provide a common services integrated radio interface for speech and data. The GSM system supports an internationally compatible mobile network access and enables personal communication on the basis of personal numbers.

Section 2 describes the specific characteristics of the GSM radio interface organization. A traffic model for the evaluation of call queueing is introduced in section 3. In section 4 the performance of several TCH assignment procedures to the radio interface is evaluated, which is formed by common and dedicated control channels and dedicated traffic channels. Finally the results are summarized and an outlook is given.

2 Organization of the GSM Radio Interface

The GSM system can be considered as a mobile network extension of the ISDN and thus it is modelled according to the concept of circuit–switched channels on the radio interface. The main concern for the design of the radio interface has been a high user capacity relative to the given limited radio spectrum and a metropolitan service area, high grade of service, low costs for infrastructure and a simple implementation of hand–held MS. A detailed description on general mobility requirements, supported telecommunication services on network architecture is given in e.g. [1] and [2].

The GSM system operates at 900 MHz (MS→BTS: 890–915 MHz, BTS→MS: 935–960 MHz) and at 1800 MHz (DCS, Digital Communication System 1800). The subbands are partitioned by a combined Frequency/Time Division Multiple procedure into several physical channels. The capacity of one physical channel is defined by the required transmission rate for the fullrate speech codec with a gross bitrate of 22.8 kbit/s. On the physical channels various information streams have to be carried. According to the

GSM concept logical user information channels and signalling channels are generated from the physical channels by means of appropriate mapping functions. Logical channels can be distinguished into two broad categories: *multipoint* and *point-to-point* channels.

Multipoint channels are used, whenever there is not any point-to-point connection between the MS and the network established. Multipoint channels are e.g.:

- The BCCH (Broadcast Control Channel, BTS→MS) distributes cell specific system information, e.g. descriptions of the channel configurations of the controlling and of neighbouring cells including pointers to frequencies and time-slots, identities of cell and network, radio criteria for cell selection and re-selection and parameters to control network access.
- The CCCH (Common Control Channel) can be further subdivided into three subchannels:
  - The RACH (Random Access Channel, MS→BTS) is used for a S-ALOHA type random access procedure.
  - The AGCH (Access Grant Channel, BTS→MS) is used to acknowledge successful accesses by either assigning point-to-point channels or by sending explicit reject indications.
  - The PCH (Paging Channel, BTS→MS) broadcasts paging requests to localize an MS within its current LA (Location Area) in case e.g. of an incoming call or short message.

The following bidirectional (BTS↔MS) *point-to-point* channels are defined:

- The SDCCH (Stand Alone Dedicated Control Channel) carries signalling information and user short messages.
- Several types of user TCHs (Traffic Channel) can be discriminated by their transmission rates and offered bit error probability, e.g. TCH/FS (Fullrate Speech, 2.4...9.6kbit/s) and TCH/HS (Halfrate Speech, 2.4...4.8kbit/s).
- SDCCH and TCH are always accompanied by an SACCH (Slow Associated Control Channel). It is used to carry frequently transmitted signalling information for radio link control, e.g. power control, to adjust the transmitting frame in the MS or to transmit measurement results of received signals from the MS.
- The FACCH (Fast Associated Control Channel) uses on demand a certain part of the TCH time-slots and marks them by a specific flag. Time critical signalling transactions are performed on the FACCH, if a TCH has been assigned. The *stolen* TCH-blocks may be corrected by forward error correction, if the transmission conditions are sufficient.

The concept of TCH/H, SDCCH and SACCH has been chosen, to utilize the given transmission capacity efficiently, i.e. a physical channel may carry one TCH/F, two TCHs/H or eight SDCCHs including the respective SACCH. At present a basic user access of one TCH and one SACCH is defined.

3 Traffic Models

The basis for the performance evaluation of the GSM radio interface is an appropriate traffic model. It describes the load as a number of MSs generated communication processes. The performance of the GSM system represents the response of the system to the offered load. The quantitative assessment of specific performance criteria should be oriented along the grade of service, which defines e.g. loss probability and call set-up delay. In accordance with ISDN, the MS processes can be subdivided into *control plane* and *user plane* processes.

The *control plane* processes fall into two categories. The first one includes all processes required for ISDN signalling: Signalling for call set-up and release including connection related supplementary services, signalling for the execution of non-call related supplementary services and user-to-user signalling. ISDN-type signalling traffic follows directly from the users' traffic intensity and is referred to as *call-related* signalling in the following. The second category comprises location updating and handover and is referred to as *mobility-related* signalling, cp [2]. Unlike the characteristics of call-related signalling it depends on the subscribers' movement and the system configuration.

3.1 Call–related Signalling

The following parameters for the call–related procedure rates per MS during the busy hour are assumed:

$$\begin{aligned}
RAM &= .5 \text{ (Ratio of Active Mobile Stations)}\\
\lambda_{MOC} &= .8 \text{ Mobile Originated Call attempts/h}\\
\lambda_{MTC} &= .4 \text{ Mobile Terminated Call attempts/h}\\
P_{NOANS\_MOC} &= P_{NOANS\_MTC} = .3 \text{ (Ratio of calls with no answer from called party)}\\
P_{BUSY\_MOC} &= .05 \text{ (Ratio of MOCs with called party busy)}^1\\
t_{RISC\_MOC} &= 15\text{s Ringing–time for successful (answered) calls (MOC)}\\
t_{RISC\_MTC} &= 5\text{s Ringing–time for successful (answered) calls (MTC)}\\
t_{RIUC\_MOC} &= t_{RIUC\_MTC} = 30\text{s Ringing–time for unsuccessful (not answered) calls}\\
t_{BUSY\_MOC} &= 3\text{s Channel occupation–time, if called party is busy}\\
t_{MOC} &= t_{MTC} = 90\text{s Effective call duration.}
\end{aligned}$$

It is assumed that all indicated times follow a negative exponential distribution. With these assumptions a mean channel occupation–time $1/\mu$ can be defined for MOCs and MTCs:

$$\begin{aligned}
1/\mu_{MOC} &= P_{NOANS\_MOC} \cdot t_{RIUC\_MOC} + P_{BUSY\_MOC} \cdot t_{BUSY\_MOC} +\\
&\quad (1 - (P_{NOANS\_MOC} + P_{BUSY\_MOC})) \cdot (t_{RISC\_MOC} + t_{MOC})\\
&= .3 \cdot 30\text{s} + .05 \cdot 3\text{s} + .65 \cdot (15\text{s} + 90\text{s}) = 77.4\text{s}\\
1/\mu_{MTC} &= P_{NOANS\_MTC} \cdot t_{RIUC\_MTC} + (1 - P_{NOANS\_MTC}) \cdot (t_{RISC\_MTC} + t_{MTC})\\
&= .3 \cdot 30\text{s} + .7 \cdot (5\text{s} + 90\text{s}) = 75.5\text{s}
\end{aligned}$$

In the following a common mean value $1/\mu \simeq 75$s is assumed for both call directions. This results in a traffic intensity per subscriber ERL_{MS} of

$$ERL_{MS} = \frac{\lambda_{MOC} + \lambda_{MTC}}{3600\text{s}} \cdot \frac{1}{\mu} = 0.025 \text{ [Erl]}$$

The subscribers' traffic intensity ERL_{MS} is composed of *successful* call attempts with TCH occupation upon the called party has answered and of *unsuccessful* call attempts which are not answered. For unsuccessful mobile originated call attempts the occupation of the TCH can be avoided, if the OACSU procedure is used, which is discussed later in this paper.

3.2 Mobility–related Signalling

For *mobility–related* signalling only handover is considered. If an MS has a call request, the network assigns a TCH in that cell where the MS is currently situated. While the call is active the MS may move through several cells, and in order to avoid an interruption of service, the call is handed over to the new cell entered by the MS (*inter–cell* handover) and the call can be continued without perceptible disruption. In this paper only the *inter–cell* handover case caused by movement is addressed as handover.

To evaluate the rate of handover procedures per MS during the busy hour a mobility model is used, which is described in [2]. It is assumed that movement and call behaviour of an MS are statistically independent. Furthermore it is supposed that a stationary equilibrium exists between calls of a cell which are handed–in and handed–out. Therefore the rate λ_{HO} of handover calls per MS is:

$$\lambda_{HO} \sim P_{BUSY} \cdot \lambda_{LU}(P(1))$$

where $P_{BUSY} = ERL_{MS}$ [Erl] marks the probability that an MS is in call–state and $\lambda_{LU}(P(1))$ means the number of location updating request of a user in a LA with exactly one cell, cp [2].

[1] In case of an MTC for a busy MS the call is already blocked at the MSC and neither a paging procedure will be initiated nor a TCH will be assigned.

4 TCH Assignment

Figure 4.1 shows the signalling procedures for a user information connection between two MSs.

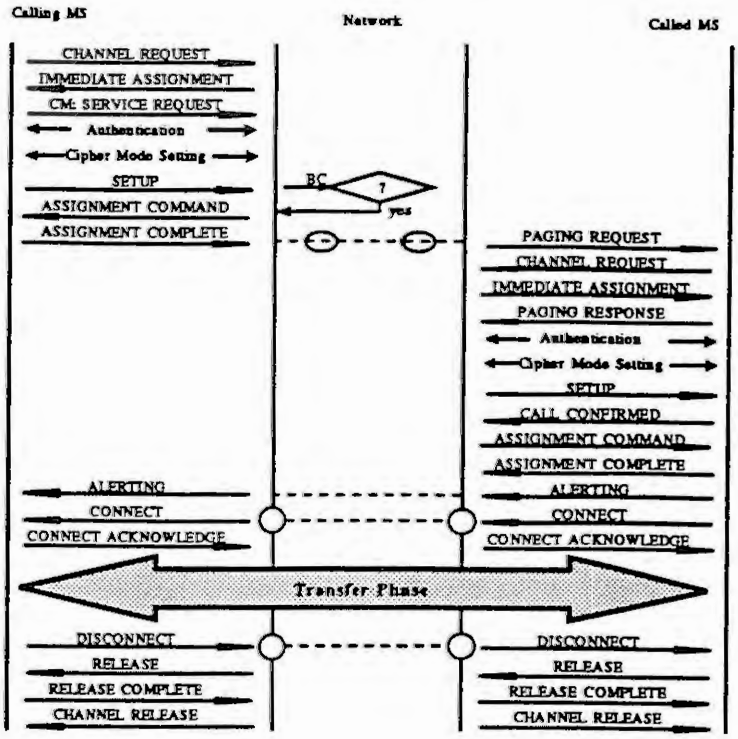


Figure 4.1: Call establishment and release.

The calling MS initiates the establishment of a ciphered radio connection on an SDCCH. With the CM SERVICE REQUEST message a call request is initiated in the MSC. Then the MS sends the SETUP message to the network, which contains the bearer capability information, the MSISDN of tbe called mobile user and service compatibility information. The network identifies tbe requested service from the bearer capability information element together with the supported bitrates of the MS. The network returns the CALL PROCEEDING message and indicates the beginning of the connection establishment in the network. Tbe network assigns an appropriate TCH to the MS by sending an ASSIGNMENT COMMAND message. The MS tunes to the assigned channel and initiates the establishment of a data link connection on the appertaining FACCH. The seizure of the TCH is completed upon the network receiving ASSIGNMENT COMPLETE. Then the call establishment proceeds as specified in ISDN.

On receiving the call request, the destination MSC verifies the subscription of the addressed user and initiates paging. The called MS sets up a ciphered radio connection and responds from its visited cell with a PAGING RESPONSE message. The network delivers the call by sending the SETUP message, which contains the bearer capability information, requirements for service compatibility and additional address information, if required. The MS checks its compatibility to the requested service and confirms the call with CALL CONFIRMED. For an incoming call, there are several possibilities to allocate the necessary resources, e.g. TCH and interworking functions:

- The network identifies tbe requested service from the signalling of the GSM network or the ISDN.
- The service is derived from the service profile allocated to the dialled number (multi–numbering).
- The network has received a call without specific service information and the MS may select the service by returning appropriate service information in the CALL CONFIRMED message (single–numbering).

Tbe network assigns a TCH by sending an ASSIGNMENT COMMAND message. Upon the MS seizing tbe TCH, the call establishment proceeds as in ISDN. In order to utilize the limited number of radio cbannels more efficiently and to remove signalling–times from the TCHs, several optional modifications of the above described call set-up procedure can be applied. Queueing at the transition between SDCCH and TCH can be used to increase the utilization of TCHs. Furthermore the assignment of a TCH can be delayed, until an answer from the called party has been received (OACSU).

The call can be released independently by both users and the network following ISDN procedures.

While a call is in progress, the MS may enter a new cell. In this case the call must be handed over to the new cell. Therefore handover requests have to be considered for the assignment and the length of the occupation-time of TCHs.

In the following various TCH assignment procedures are considered. For comparison a standard GSM configuration of $m = 30$ TCHs and the traffic model of section 3 are assumed. In order to utilize the limited number of radio channels more efficiently and to remove signalling-times from the TCHs, several optional modifications of the call set-up procedure can be applied. Queueing at the transition between SDCCH and TCH can be used to increase the utilization of TCHs. Furthermore the assignment of a TCH can be delayed, until an answer from the called party has been received (OACSU).

4.1 Early TCH Assignment

As a reference model for the various TCH assignment strategies, a pure loss system is considered. In this model a TCH is assigned when a SETUP message or a CALL CONFIRMED message is received from the MS or a handover request is present in the network. All requests for a TCH are treated with equal priority and experience the same blocking probability. Using the Erlang-B formula a total number of $N_{MS} \simeq 830$ users can be served in case of no call type should experience a higher blocking probability than 2%.

4.2 Call Queueing without Off Air Call Set-up

In the following queueing systems with different priority disciplines for various call types are investigated. The initiation of originating and terminating calls is always performed on SDCCHs. Since priority queues with more than two priorities can be analysed only in a very restricted manner, the following results are mainly gained by simulation. The constraints for call queueing in general are limited waiting-times for specific call classes and low blocking probability for handover calls.

4.2.1 Priority Queue with Limited Waiting-Capacity

Figure 4.2 shows the discussed queueing model. The queue is characterized by a finite waiting-capacity ($s = 9$), FIFO discipline and two priorities:

- priority 1: handover calls
- priority 2: normal calls.

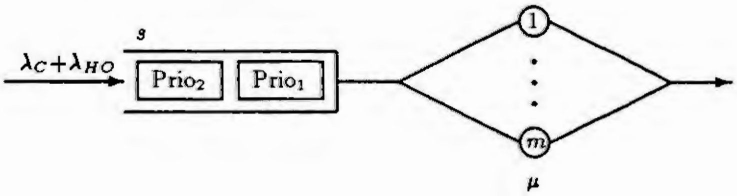


Figure 4.2: Priority queue with limited waiting-capacity.

The blocking probability of the total arrival stream and the complementary waiting-time distribution for both priorities are gained by simulation and shown in figure 4.3.

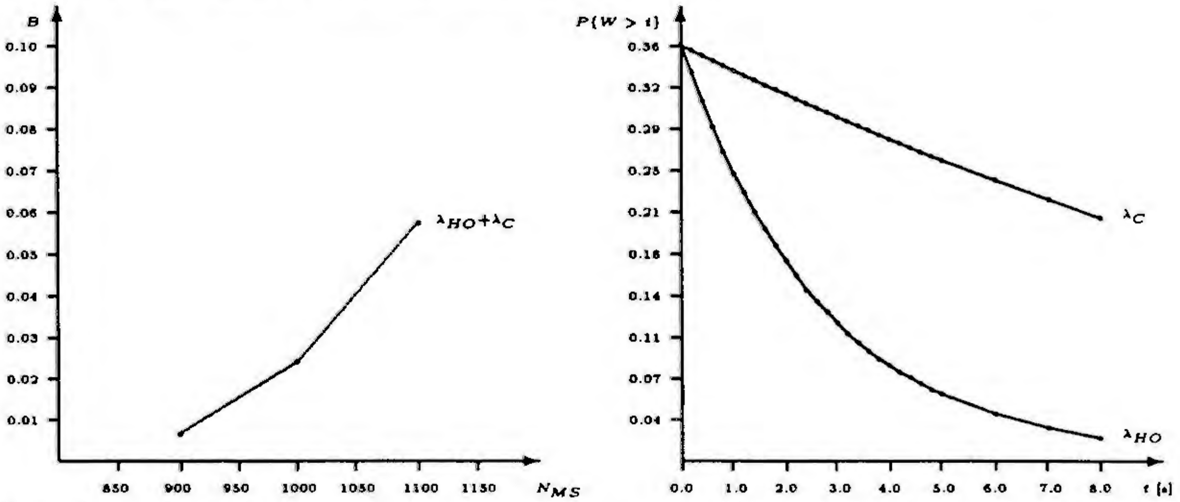


Figure 4.3: Blocking probabilities (dependent on N_{MS}) and the complementary waiting-time distributions for the various priorities ($N_{MS} = 1000$).

A blocking probability of 2% of the arrival stream would allow for a number of $N_{MS} = 980$ users. However, the maximum waiting-time of 8s for handover calls will then be exceeded by more than 2%, which cannot be tolerated. For handover calls the blocking probability may be reduced by allowing a handover call (Prio$_1$) to enter a completely filled queue by removing the last queueing call (Prio$_2$) from the queue. In this case mainly the limited waiting-time may contribute to the blocking probability of handover requests.

4.2.2 Priority Queue With Waiting-Time Control

A user will only wait a limited time until the connection is completed. Therefore the model of the previous subsection is refined by a waiting-time control. It is assumed that the patience of a user waiting for call completion can be described by a truncated normal distribution ($N(\mu, \sigma)$) for the waiting-time ($0 \leq W \leq 145$s with $\mu = 30$s and $\sigma = 15$s). Furthermore, data and speech calls are discriminated according to the traffic model of section 3. Three arrival streams are discriminated by priorities:

- priority 1: handover calls
- priority 2: data calls
- priority 3: speech calls.

The refined queueing model is shown in figure 4.4.

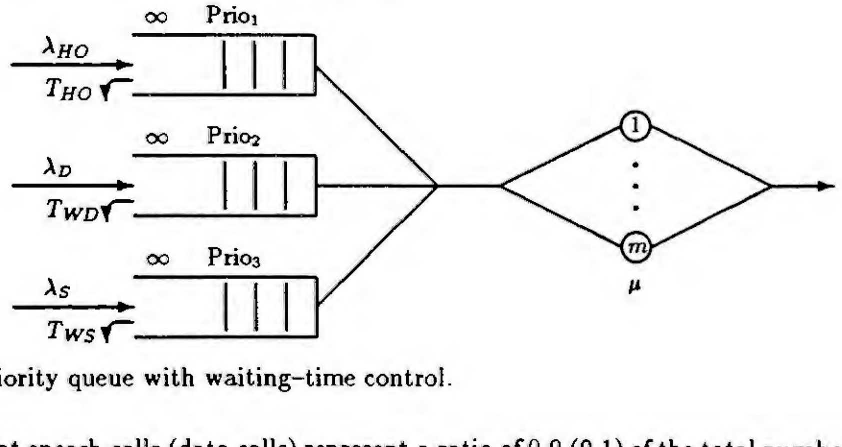


Figure 4.4: Priority queue with waiting-time control.

It is assumed that speech calls (data calls) represent a ratio of 0.9 (0.1) of the total number of normal calls. Since this model cannot be treated analytically, the blocking probabilities and waiting-time distributions for handover calls and normal calls are gained by simulation. If the waiting–time of a call is exceeded, the call is blocked and removed from the queue. Figure 4.5 gives the blocking probabilities and the complementary waiting–time distributions for the various priorities.

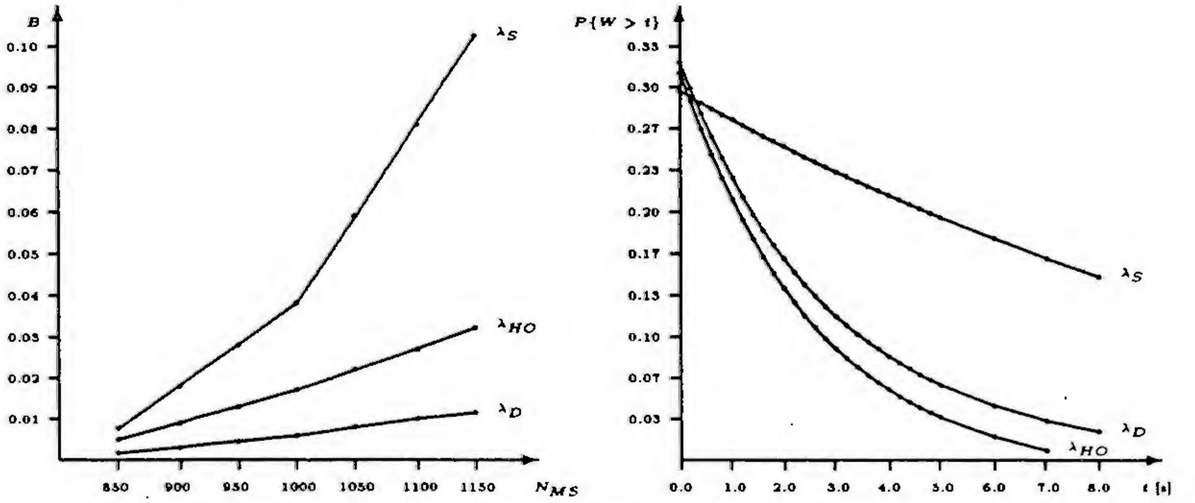


Figure 4.5: Blocking probabilities (dependent on N_{MS}) and the complementary waiting–time distributions for the various priorities ($N_{MS} = 1000$).

4.3 Call Queueing with Off Air Call Set–up

The queueing operation described in the previous section is further refined by performing the complete originating call establishment procedure for speech calls on the SDCCH. In this case a TCH is assigned, if the called party has answered (OACSU). This procedure may lead for a fixed user to the unusual situation that at answer instant a TCH is not yet available for communication. The called party has then to be informed of the waiting situation by an appropriate announcement. In case the originating speech call arrives at a destination, which does not return any answer, a TCH request may be generated by time–out of a supervisory–timer. This timer may comprise e.g. the mean establishment time of the network connection and the mean ringing–time at the called end. This timer is not considered in the following simulation results. Figure 4.6 shows the blocking probabilities and waiting–time distributions of the various call types.

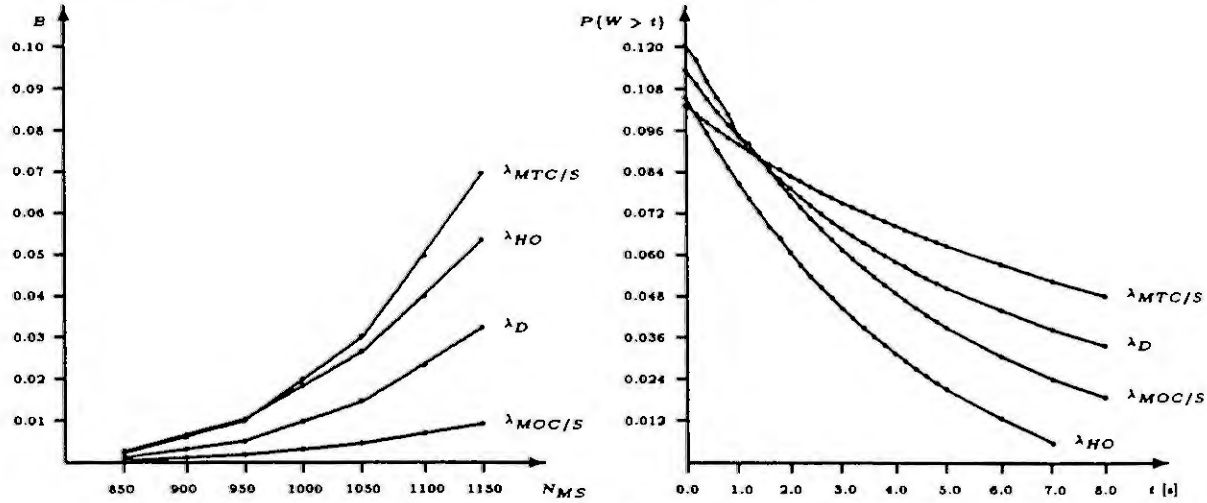


Figure 4.6: Blocking probabilities (dependent on N_{MS}) and the complementary waiting–time distributions for the various priorities ($N_{MS} = 1000$).

4.4 Repercussions of Call Queueing on the SDCCH Occupation

Call queueing and OACSU have been applied to increase the utilization of TCHs. However a call request will spend its waiting-time on an SDCCH and thus increase the mean occupation-time of SDCCHs. For the call-related signalling the mean occupation-time is:

$$\begin{aligned} 1/\mu &= \sum_{i=0}^{2} 1/\mu_i \text{ with:} \\ 1/\mu_0 &= \text{4s, Basic Signalling time} \\ 1/\mu_1 &= \text{Additionally OACSU related signalling time, cp subsection 3.1} \\ 1/\mu_2 &= \text{Mean waiting-time for TCH assignment} \end{aligned}$$

In table 4.1 the additional SDCCH occupation-time is given for the discussed models.

Model of section	Type	Priority	$1/\mu_1$	$1/\mu_2$	$\sum_{i=0}^{2} 1/\mu_i$
4.2.1	MOC, MTC	2	-	4.25s	8.25s
4.2.2	MOC/D, MTC/D	2	-	0.96s	4.96s
	MOC/S, MTC/S	3	-	2.88s	6.88s
4.3	MOC/S	1	18.9s	0.5s	23.4s
	MOC/D, MTC/D	2	-	0.65s	4.65s
	MTC/S	3	-	0.9s	4.9s

Table 4.1: Additional SDCCH occupation-time.

For mobility-related signalling procedures the SDCCH occupancy is:

$$1/\mu = 1/\mu_0$$

5 Summary

In this paper the radio access in a GSM system has been highlighted and the effects on the performance have been investigated. It shows that a very robust protocol solution has been chosen for multipoint and point-to-point access signalling channels. Specific queueing models have been studied to improve the utilization of user information channels. The elements of the radio access, SDCCH and TCH, have first been considered separately. In a following step the interaction between these elements has been investigated, e.g. it has been shown, how the use of OACSU removes load from TCHs and increases the load on SDCCHs. The GSM system may be considered as a potential platform, from which a new generation of mobile telecommunication systems may evolve.

References

[1] DCRC (Digital Cellular Radio Conference), Hagen FRG, Oct. 1988.

[2] W. Fuhrmann, V. Brass: Performance Evaluation of a GSM Public Land Mobile Network, in: *Proc. of the 5th Nordic Seminar on Digital Mobile Radio Communications*, pp 295–302, Helsinki, Dec. 1992.

[3] W. Fuhrmann, V. Brass, U. Janßen, F. Kuehl, W. Roth: Digitale Mobilfunkkommunikation, *Tutoriumsbeitrag für GI/ITG-Fachtagung '93, Kommunikation in verteilten Systemen*, ed. J. Swoboda, pp 124-181, Munic, Mar. 1993.

[4a] L. Kleinrock: Queueing Systems: Volume 1: Theory, John Wiley&Sons, New York, 1975

[4b] L. Kleinrock: Queueing Systems: Volume 1: Computer Applications, John Wiley&Sons, New York, 1975

[5] W.R. Mende: Bewertung ausgewählter Leistungsmerkmale von zellularen Mobilfunksystemen, PhD thesis in Electrical Engineering from the Fern-Universität Hagen, Jan. 1991.

Performance Evaluation of Vehicle-Beacon (Roadside) Communications Proposed for Standardization

Carl-Herbert Rokitansky

RWTH Aachen Technical University, Communication Networks, D-52074 Aachen, Germany
Tel: ++49/241/80-7924 or -7910; Fax: ++49/241/84964; Email: roki@dfv.rwth-aachen.de

Abstract

In this paper, the results of performance evaluations by analysis and computer simulations of beacon-vehicle communication protocols/options for Dedicated Short Range Communications (DSRC), currently being proposed as candidates for standarization of the data link layer in Europe (CEN/TC278/WG9) and in North America are presented. Different approaches (random delay counter versus persist mechanisms, adaptive algorithms, single slot public windows versus multiple slot public windows, variable versus fixed frame structure, etc.) for the medium access control protocols are compared. The non-completion (failure) rate or the percentage of the communication zone which was required in average / maximum to complete the transaction are used as important indicators to evaluate the performance of the analyzed schemes and parameter settings with regard to their capability to recover successfully after a packet collision. The traffic intensity / number of lanes and the speed of the vehicles, the length of the communication zone and the amount of data and the length of messages to support a specific application are taken into account. These performance evaluations are also important for the design of RTI systems to calculate the required communication zone, the transmission / receive range, the required downlink / uplink transmission rate, the number of beacon heads, and antenna characteristics.

1. Introduction

The main important characteristic of beacon-vehicle communications is the limited communication zone / time, depending on transmission parameters, number of beacon heads, orientation and width of beam of antennas on beacons and vehicles, installation height of antennas, traffic intensity, speed of vehicles and supported application(s).

Due to the random arrival of vehicles, appropriate medium access schemes and efficient recovery algorithms after packet collisions are required to satisfy the communication needs of the relevant RTI applications (Automatic Fee Collection & Access Control, Dynamic Route Guidance, etc.), which rely on vehicle-beacon communications. Such access schemes have already been proposed and analyzed in [1], [2], [3] and [4]. The communication requirements of the various applications can vary significantly: e.g. for Automatic Fee Collection (AFC) a sequence of relatively small data packets must be exchanged between the beacon and the vehicles with a very high reliability; while for Dynamic Route Guidance large amount of data has to be transmitted on the downlink (beacon to vehicle direction) but only short messages are transmitted on the uplink (vehicle to beacon direction).

2. Types of Beacon Configurations

Assuming a TDMA scheme [5], in principle it is possible to avoid data collisions on the uplink using short communication zones (e.g. 5 meter in longitudinal direction) in:

- single-lane scenarios (i.e. only one vehicle can be physically present in the communication zone
- multiple-lane scenarios with one (or more) antennas for each lane (acting synchronous in downlink transmissions.

Data collisions on the uplink are possible if:

- short communication zones in longitudinal direction are used (e.g. 5 meter) are used and:
  - more than one motor cycle is present in the scenarios above
  - multiple-lane scenario and one beacon antenna covering all lanes for both uplink and downlink communications
  - multiple-lane scenario and several beacon antennas, but each covering more than one lane for uplink communications
- larger communication zones (e.g. > 5 meter) in longitudinal direction are used

In this paper, we further focus on the scenarios in which collisions can occur, and on the comparision of different medium access schemes and recovery algorithms.

Due to the fact, that the medium access control protocols proposed for standardization will include appropriate algorithms to recover from packet collisions, system designers/ implementers might consider to avoid complex beacon configurations (e.g. several antennas in multiple-lane scenarios) by possibly relying on the recovery capability of the proposed algorithms, taking the performance of the analyzed schemes and parameter settings and communication requirements of the supported application(s) into account.

3. Characteristics of Vehicle-Beacon Communications

European DSRC/MAC Protocols

The European DSRC/MAC protocols proposed for standardization are based on:

- TDMA (Time Division Multiple Access): the vehicles share the medium by transmitting in (different) time slots. In the "Collision" scenarios described above, data collisions occure, when two or more vehicles transmit in the same time slot.

- Half-duplex mode: The beacon and the vehicles share the same physical channel: Either the beacon transmits data on the downlink, which can be received by the vehicles in the commuication zone, or the vehicles transmit data on the uplink, which can be received by the beacon, if no collisions occured.

- Public / Private Windows: The beacon controls the medium by offering two different types of windows to the vehicles:
  - Public Windows: The beacon broadcasts periodically specific messages on the downlink (PWA / BST) including a "Beacon Service Table (BST)" (containing important information on the supported application(s), currently valid protocol parameter settings etc.) and a so called "Public Window Allocation". Immediately after the transmission of this PWA / BST message a "Public Window" is opened, which may be accessed by vehicles in the communication zone for uplink transmissions, (mainly) for address acquisition of newly arrived vehicles in the communication zone.

    In the MAC schemes analyzed below, we distinguish between:

    - Single-Slot Public Uplink Window: The window contains only one time slot for uplink transmission of any vehicle in the communication zone (see Figure 4-1 below).
    - Multiple-Slot Public Uplink Window: The window contains more than one (*n)* time slots for uplink transmission of vehicles in the communication zone (see Figure 4-1 below). The data messages sent by a maximum of *n* vehicles can be received successfully by the beacon in a multiple-slot window, provided that all *n* vehicles are transmitting in different time slots.

    Advantages of the multiple-slot window approach are:

    - If several vehicles transmitted successfully in different time slots, the beacon can first serve vehicles requesting applications with a higher priority
    - Only one PWA / BST message must be transmitted to offer several public uplink slots.

    A disadvantage of the multiple-slot window approach is, that if only one vehicle intends to use the public window, an (unneccesary) delay is caused depending on the number of slots in the window.

    Performance analysis in [3] and results from computer simulations (see Section 6 below) show, that the trade-off between the advantages / disadvantages of the single slot window approach versus the multiple-slot window approach described above depend on the length of (PWA / BST) data, slot duration, PWA timeout values, traffic intensity, amount of application data and other parameter settings, and must be further analyzed.

  - Private Windows: Once the (randomly chosen) address of a vehicle transmitted in a public window is known to the beacon, the beacon can allocate private windows for that specific vehicle for further data exchange. The private window is a time slot reserved for exclusively use by the specified vehicle to protect it against data collisions transmitted by other vehicles.

Vehicle to Roadside Communication (VRC) Frame Structure

While the medium access control protocols, proposed for standardization of DSRC in Europe and described above, are based on a quite flexible sequence of public and private windows / messages on the downlink and uplink, the TDMA protocol [7] used by the Vehicle to Roadside Communication (VRC) system, developed in response to the many short range communication needs of the IVHS User Services in North America, is based on a fixed frame structure (see Fig. 3-1), consisting of a Reader Control Message (RCM), Slot Data Messages (SDM), and Activation Slots (AS). The reader control message is transmitted by the roadside .beacon and is the master signal for the vehicle transponder. The Activation Slots are used to resolve contention for the data slots and is the first message sent by the vehicle transponder when entering the communication area. After receiving an ID(s) in the Activation Slots, the beacon transmits an RCM directing the vehicle transponders to either receive or transmit information during the Slot Data Messages.

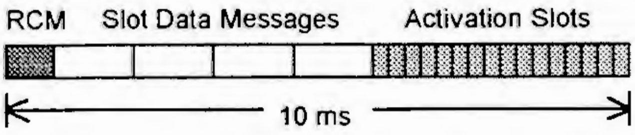


Fig. 3-1: Hughes TDMA Protocol (Open Road Frame)

The RCM can be regarded to correspond to the PWA / (BST) message of the European proposal, the Activation Slots are equivalent to the Public Slots, and the Slot Data Messages correspond to the private downlink / uplink messages / windows of the European proposal described above.

4. Analyzed Medium Access Control Schemes

Basically there are two schemes [3] to reduce the probability of data collisions in public uplink slots and to recover from such collisions once they occured: Random Delay Counter (c) and Persistence Mechanism (p)

Random Delay Counter Mechanism: A vehicle entering the communication zone of a beacon will receive a PWA / BST message opening a public window. Using the random delay counter mechanism it randomly choses a value *C* between 1 and the maximum random delay value *c* specified (e.g. in the BST). This random value indicates in which time slot the vehicle should transmit. The maximum delay counter value specified is sensible with regard to the performance of the access scheme, because a higher value causes longer delays while a smaller value increases the probability of collisions. Optimal values were analyzed in [3] and are validated by the simulation results in Section 6 below. Using a random delay counter a vehicle transmits only once in a specified number of *c* slots, and the probability that a vehicle transmits within a specified number of *c* slots is 1. The random delay counter mechanism can be applied to both the single-slot window as well as to the multiple-slot window approach, in which case the maximum delay counter value should be specified according to the number of slots in the public uplink window.

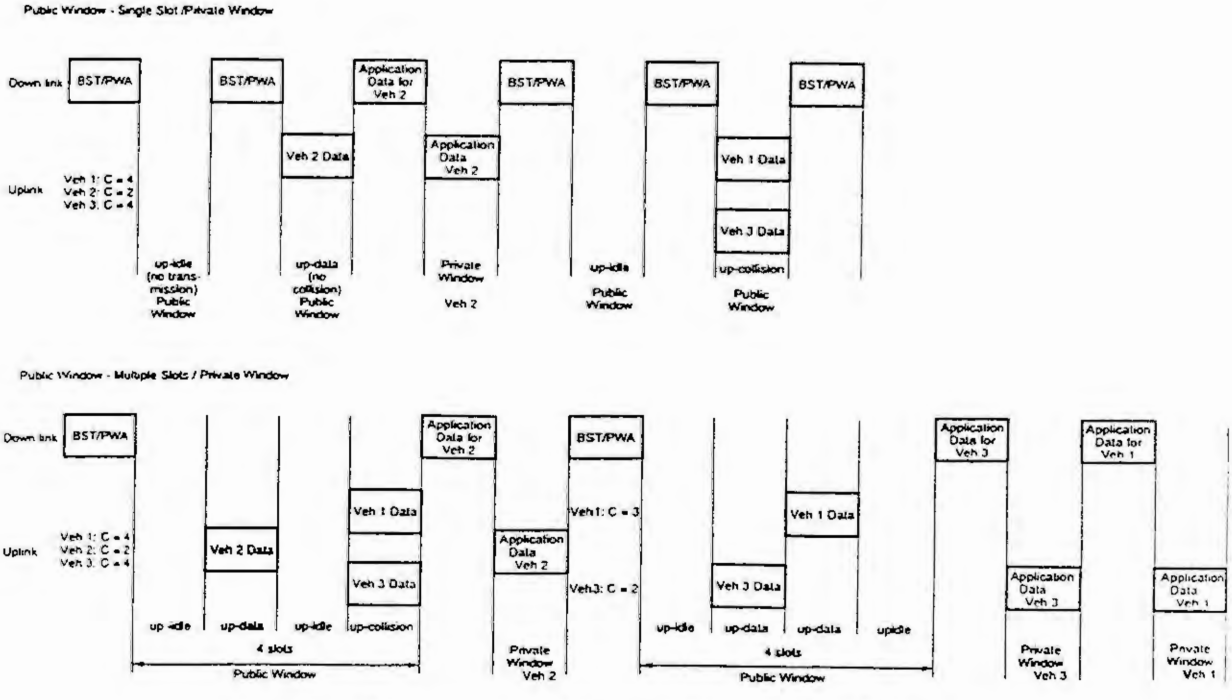


Figure 4-1: Examples of the Random Delay Counter Mechanism

Persistence Mechanism: Using the persistence mechanism, a vehicle determines for each slot whether to transmit in this slot or not by chosing a random value *u* between 0 and 1: If the value *u* is below or equal to the specified persistence value *p* (e.g. in the BST) the vehicle transmits in this slot, otherwise it waits for the next public slot and decides again. Therefore according to the persistence value *p* specified, the vehicle transmits with probability *p* in each time slot. Unsimilar to the random delay counter mechanism described above, using the persistence mechanism a vehicle may transmit more than once in a number of *c* slots, and the proba-bility that a vehicle does not transmit in a number of *c* slots is greater than 0 (for $p < 1$).

Adaptive Schemes: If the probability is low (refer to [3] for details) that there are two or more vehicles in the communication zone, which can cause collisions, it is reasonable to use adaptive, "immediate response" schemes for the single-slot window approach in the following way: Initially a vehicle immediately transmits in the next public uplink slot after a PWA / BST message was received. Only in case of collisions, which are assumed to have occured if no acknowledgement for that vehicle was received before the next PWA / BST message is received, the vehicle switches to the random delay counter or persistence mechanism described above. In the performance evaluations below these schemes are refered to as "Immediate Response" schemes ("1c - counter " or "1p - persistence" mechanism).

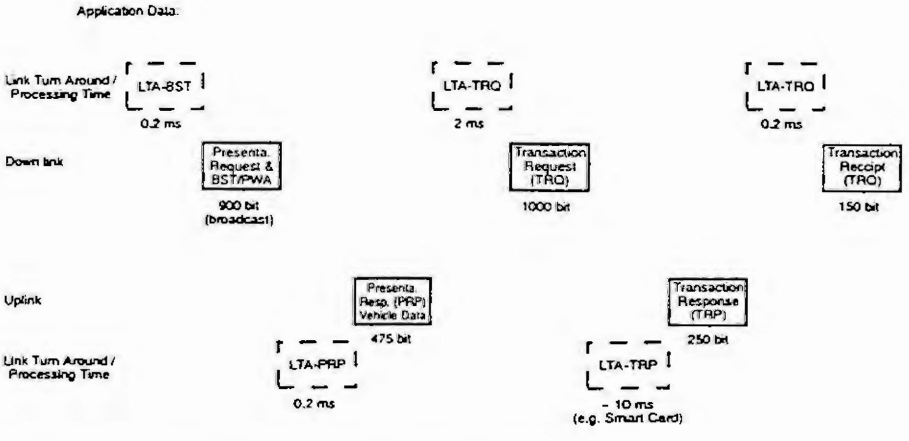
Another option of an adaptive scheme being analyzed is the following: Each vehicle responds immediately as defined above, but instead of switching to the maximum random delay counter value *c* or persistence value *p* once a collision occured, the vehicle increments (decrements) a vehicle specific maximum value whenever a new collision is determined until the overall maximum *c* (or minimum *p* value) is reached. In the performance evaluations below these schemes are refered to as "Increment" schemes ("ic - counter " or "ip - persistence" mechanism). Instead of controling the increment value by the vehicles beacon-controled increment algorithms ("bc - counter" or "bp - persistence" mechanism) are currently being investigated.
For the "ic - counter" mechanism analyzed in the performance evaluations described in Section 6 below, starting from a vehicle specific maximum value *c'* of 1, this value was simply incremented by 1 whenever a collision was determined, until the maximum *c* value was reached. For the "ip - persistence" mechanism the vehicle specific *p"* value was initially set to 1. When the first collision was determined it was set to 0.75, and on further collisions it was set to $p' := p' - (1-p')/2$ until $p' <= p$. If $p' < p$ then $p' := p$.

5. Stochastic Simulation Model and Performance Evaluation Parameters

The performance of the following medium access control schemes (see Section 4 for details) has been evaluated, taking the data exchange of a (typical) Automatic Fee Control (AFC) application into account (Figure 5-1), and using a modified version of the SIMCO2 / SIMCO3++ simulator described in [2] and [6]:

- Basic random delay counter (c): non-adaptive
- Random delay counter - immediate response (1c): adaptive
- Random delay counter - incremented *c* (ic): adaptive
- Basic Persist.(p): non-adaptive
- Persist - imm. resp. (1p): adapt.
- Persist - decrement *p* (ip): adapt.

The downlink transmission rate was assumed to be 500 kbit/s, the uplink transmission rate was 125 kbit/s; the duration for the transmission of PWA / BST messages including the link turn around time was 2 ms (0.5 ms), the assumed PWA timeout value after the end of a public window was set to 10 ms, which was used only if application data to vehicles was sent, otherwise a new PWA / BST message was immediately transmitted. The duration of public slots was set to 4 ms, 1 ms (0.5 ms). An ideal channel (no bit errors) was assumed for first evaluations and comparisions.



Application Data:
Link Turn Around / Processing Time
LTA-BST
0.2 ms
LTA-TRQ
2 ms
LTA-TRO
0.2 ms
Down link
Presenta. Request & BST/PWA
900 bit (broadcast)
Transaction Request (TRQ)
1000 bit
Transaction Reccipt (TRO)
150 bit
Uplink
Presenta. Resp. (PRP) Vehicle Data
475 bit
Transaction Response (TRP)
250 bit
Link Turn Around / Processing Time
LTA-PRP
0.2 ms
LTA-TRP
~ 10 ms (e.g. Smart Card)

Figure 5-1: (AFC) Application Data Exchange and Link Turn Around / Processing Times

The number of lanes was set to 3, and the traffic intensity was 2400 veh./h/lane, resulting in a road intensity of 7200 veh./h. The vehicles were generated using a Poisson arrival process taking the specified road intensity into account. The investigated speed of the vehicles was set to 100 km/h and the length of the communication zone was assumed to be 5 meter.

For the performance evaluations of the VRC protocol (IVHS America), the downlink and uplink transmission rates were both set to 500 kbit/s. In the fixed frame structure (see Fig. 3-1 above), the Reader Control Message (RCM) was set to 0.5 ms, each slot of the Slot Data Messages to 1.4 ms, each slot of the Activation Slots to 0.2 ms duration, and the time duration between frames to 1 ms. The number of Activation Slots was varied between 4, 8, 12 and 16, and the number of Data Message Slots was set to 1, 2, 3, or 4 respectively. Both, the random delay counter mechanism and the persistence mechanism were investigated for the VRC protocol and compared with the European approach, using also 500 kbit/s for the downlink and uplink channel, and multiple-slot public uplink windows with 4, 8, 12, and 16 slots respectively.

6. Results of Performance Evaluations by Simulation

For the performance evaluation of the analyzed medium access control schemes and parameter settings, described in Sections 4 and 5 above, the following performance parameters were defined:
Non-completion (Failure) Rate: Percentage of vehicles which were unable to complete the transaction before the end of the communication zone was reached.
Maximum Distance for Completion of Transaction: If the completion rate is 100%, this parameter indicates the maximum percentage of the communication zone used until the completion of the transaction.
Average Distance for Completion of Transaction: This parameter indicates the average percentage of the communication zone used until the completion of the transaction.

In the diagrams the key for each simulation run is indicated as follows (examples below):

# of slots	Up slot	Method	% of coll. in	% of comms zone	% of comms zone
Up [Data]	in ms	(c,1c,ic,p,1p,ip)	Pub. up slots	at av. dist. of compl.	at 100% completion

Examples:
Flexible down/up message sequence (Non-fixed; European approach):
s3 4 c 3 1.4 23.5 88.3 (Multiple-slot 3, 4 ms, max. c 3, 1.4% coll., 23.5% at aver., 88.3% at 100%)
s1 1 p 0.6 1.1 16.4 41.3 (Single-slot, 1 ms, persist 0.6, 1.1% coll., 16.4% at aver., 41.3% at 100%)
s4 0 0 p 0.6 0.5 15,2 24.8 (Multiple-slot 4, <1 ms, p 0.6, 0.5% coll., 15.2% at aver., 24.8% at 100%)

Fixed frame structure (VRC Protocol):
s8 3 0 c 8 0.10 16.9 36.2 (Act. slot 8, 3 Data slots, <1 ms, c 8, 0.10% coll., 16.9% aver., 42.7% at 100%)
s4 2 0 p 0.4 0.36 18.2 31.2 (Act. slot 4, 2 Data s., <1 ms, p 0.4, 0.36% coll., 18.2% aver., 31.2% at 100%)

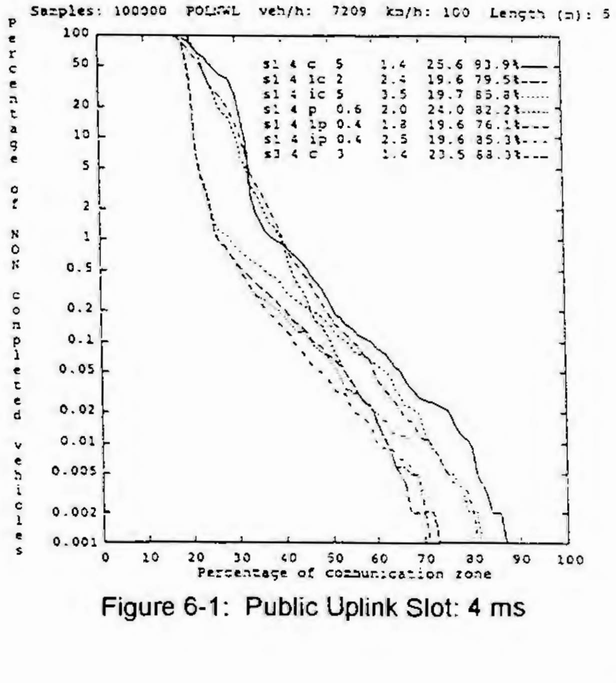


Figure 6-1: Public Uplink Slot: 4 ms

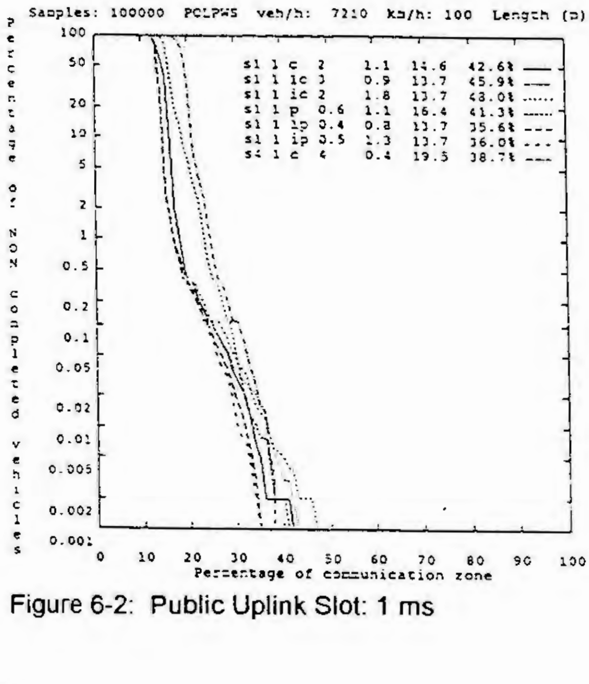


Figure 6-2: Public Uplink Slot: 1 ms

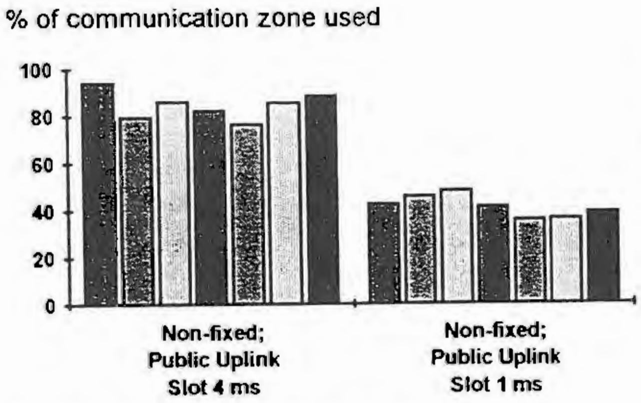


Fig. 6-3: % at 100% of completion

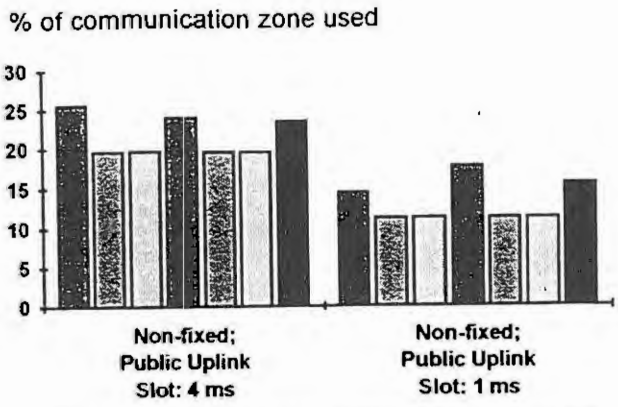


Fig. 6-4: % at average distance of completion

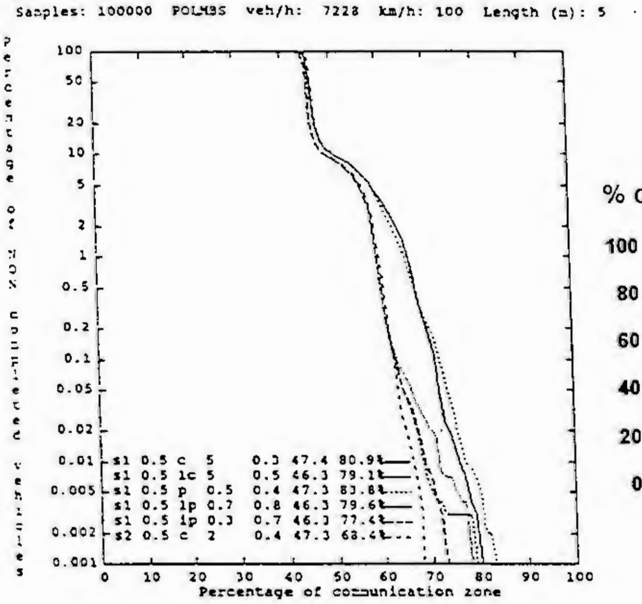


Fig 6-5: Non-fixed (more ph.) PWA/BST 0.5 ms
Public Uplink Slot: 0.5 ms

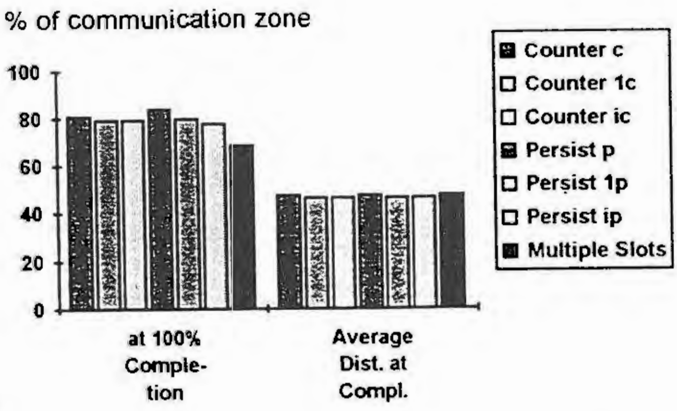


Fig. 6-6: Non-fixed (more phases) PWA/BST 0.5 ms
Public Uplink Slot: 0.5 ms

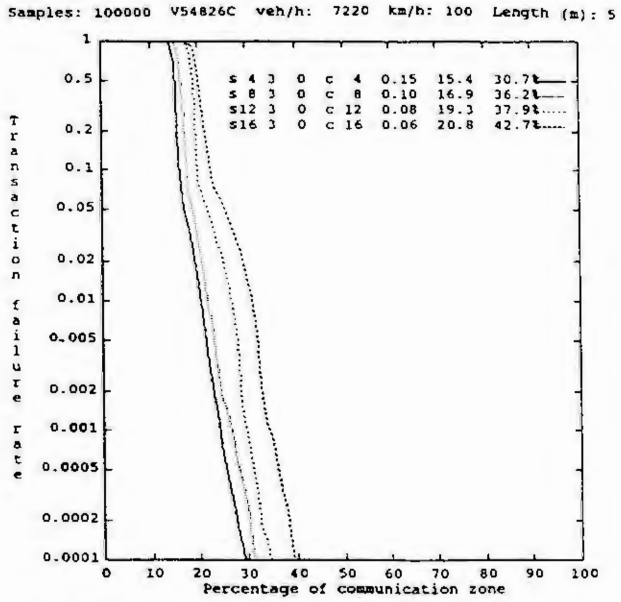


Fig. 6-7: Fixed; Counter c; Act. (Pub.) Slot: 0.2 ms
LTA 1 ms + RCM 0.5 ms

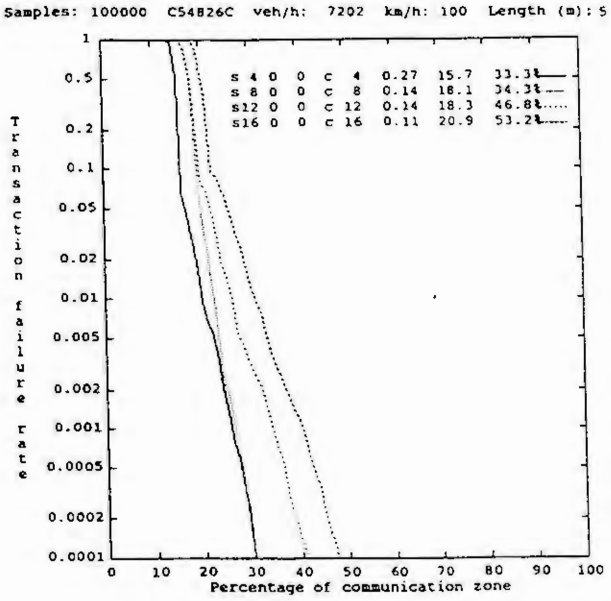


Fig. 6-8: Non-fixed; Counter c; Slot: 0.2 ms
LTA 1 ms + BST/PWA 1.8 ms

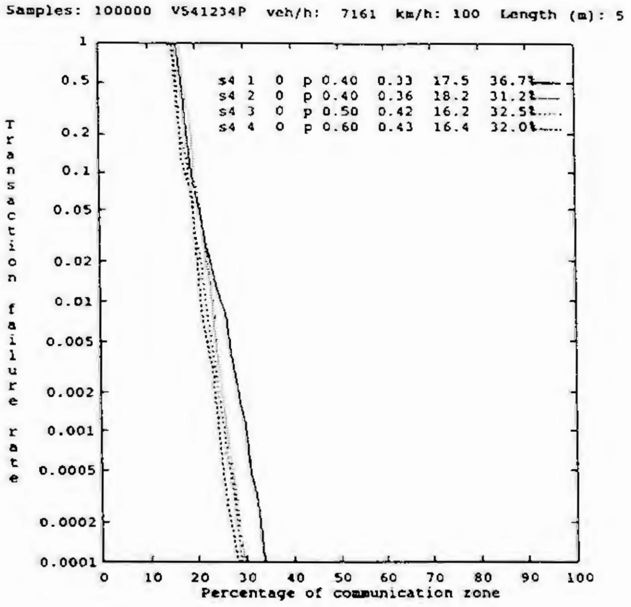


Fig. 6-9: Fixed; Persist p; Act. (Pub.) Slot: 0.2 ms
LTA 1 ms + RCM 0.5 ms

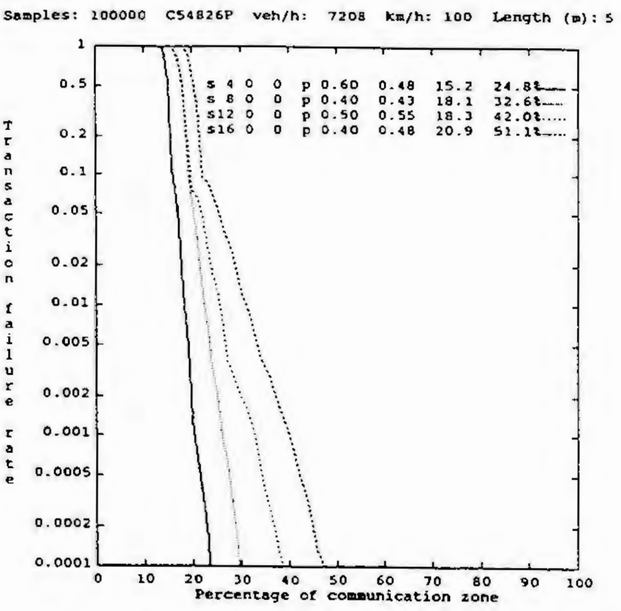


Fig. 6-10: Non-fixed; Persist p; Slot: 0.2 ms
LTA 1 ms + BST/PWA 1.8 ms

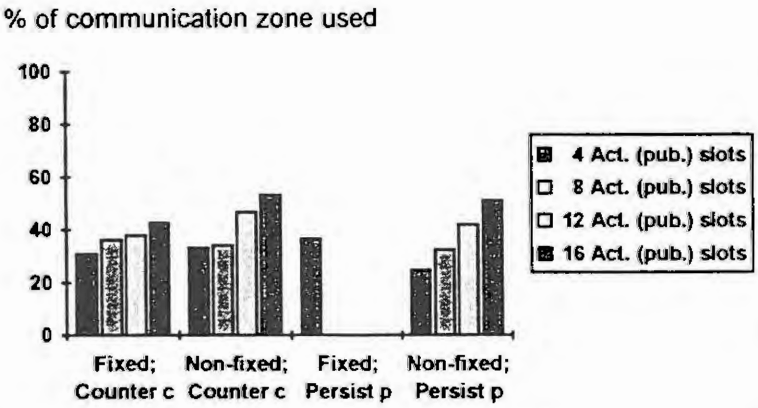


Fig. 6-11: at 100% of compl. (Pub. slot 0.2 ms)

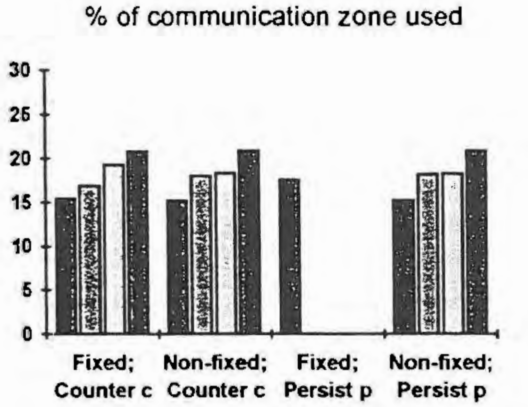


Fig. 6-12: at aver. dist. of comp. (Slot 0.2 ms)

Discussion of Results

For each medium access scheme and parameter settings, several simulation runs with a sample size of 100.000 vehicles were performed using different values for the maximum delay counter *c* or the persistence value *p* respectively. Those simulation runs with the best performance for a specific value of the maximum delay counter *c* or the persistence value *p* are shown in Figure 6-1 to Figure 6-12 for better comparision with the analyzed schemes. The results shown in Fig. 6-5 and Fig. 6-6 above can not be compared directly with the results in Fig. 6-1 to Fig. 6-4, because more phases were specified in those cases. Also, due to the different (fixed) frame structure in the VRC approach, with no / minimum BST being transmitted and with a maximum number of bits in the data slots, the results in Fig. 6-7 to 6-12 (fixed versus non-fixed) should be compared with respect to the fact that a slightly different sequence and size of messages for the AFC application described in Section 5 had to be specified for the fixed frame structure. The peformance evaluation of the analyzed medium access schemes and parameter settings show the following results:

- The adaptive schemes ("immediate response" 1c,1p and "increment" ic,ip mechanism) show always better results than the non-adaptive schemes for the single-slot window approach.
- The basic random delay counter mechanism (c) shows a better performance than the basic persistence mechanism (p); however:
- The adaptive schemes show better results for the persistence mechanisms (1p,ip) than the random delay counter mechanisms (1c,ic).

- For the analyzed parameter settings the "increment" mechanism does not show significant advantages compared to the "immediate response" mechanism.
- The single-slot window approach shows always a better performance with regard to the average distance at completion of the transaction.
- The multiple-slot window approach (with 3 slots or 2 slots) using the basic random delay counter mechanism (c), showed better results for very short PWA / BST messages (0.5 ms) and very short public uplink slots (0.5) with regard to the distance at 100% completion.
- The performance decreases with an increasement of the number of activation (public uplink) slots above 4 (or 5) (4, 8, 12, 16 were analyzed in the fixed versus non-fixed comparision).
- For the analyzed AFC application, a fixed number of 4 data slots as specified in the Hughes TDMA frame structure (see Fig. 3-1) does not show the best perfomance; in the simulation results, a number of 3 data slots was the optimal value in most cases.
- In the comparision between the fixed versus the non-fixed approach, the persistence mechanism showed slightly better results than the random delay counter mechanism.
- The (European) approach using a flexible sequence and variable size of messages showed advantages over the fixed frame structure, especially with respect to the average distance of completion.
- The optimal values for the maximum random delay counter *c* and the persistence value *p* (for the investigated scenarios, especially for the fixed versus non-fixed comparision), as well as other optimal parameter settings must be further analyzed.

7. Conclusions

The performance of several medium access schemes proposed for standardization of the DSRC Layer 2 in Europe and in North America (IVHS America) have been evaluated. The adaptive schemes based on a persistence mechanism and applied to the single-slot window approach showed better results in most cases. However the multiple-slot window approach using the basic random delay counter mechanism showed better performance for very short PWA / BST messages and very short public uplink slots with regard to the percentage of the communication zone used at 100 % completion. The (European) approach using a flexible sequence and variable size of messages showed some advantages over the fixed frame structure, especially with respect to the average distance of completion. Further research activities will include adaptive beacon-controlled schemes and optimal parameter settings, e.g. PWA / BST duration, slot duration, number of slots in public uplink windows (activation slots) and in data slots (Slot Data Messages in VRC approach), PWA time-out values, values for adaptive persistence mechanisms or for the random delay counter, etc., in different environments (length of communication zone, speed of vehicles, traffic intensity, etc.) and a further [3] accurate analysis using Marcov chains.

Acknowledgement

Horst Clausen, Berhard Walke, Christian Wietfeld and the members of the WG9 / SG.L2 Group have provided helpful discussions and important suggestions.

References

[1] B. Walke, C.-H. Rokitansky, "Short-Range Mobile Radio Networks for Road Transport Informatics", Proc. MRC '91, Nice, France, Nov. 1991, pp. 183 - 192.

[2]] C.-H. Rokitansky, "SIMCO2: Simulator for Performance Evaluation of Vehicle- Beacon and Inter-Vehicle Communication Protocols (Medium Access and Knowledge-based Routing)", Proc. 41th Vehicular Technology Conference, IEEE, St. Louis, MO, USA, May 1991, pp. 893 - 899.

[3] C.-H. Rokitansky, "Performance Analysis and Simulation of Vehicle-Beacon Protocols", Proc. 42nd Vehicular Technology Conference, IEEE, Denver, CO, USA, May 1992, pp. 1056 - 1061.

[4] Lars Egnell, "A link protocol for vehicle-roadside communication deviced for anonymous road-use in free traffic flow", Proc. 3rd Conference on Vehicle Navigation & Information Systems, Oslo, Norway, 1992, pp. 420-425.

[5] IEEE 802.2 Local Area Network Standard.

[6] C.-H. Rokitansky, Ch. Wietfeld, "Comparision of Adaptive Medium Access Control Schemes for Beacon-Vehicle Communications", in Proc. IEEE-IEE Vehicle Navigation & Information Systems Conference - VNIS '93, Ottawa, Oct. 1993, pp. 295-299.

[7] M.A. Kady, M.P. Ristenbatt, "An Evolutionary IVHS Communication Architecture", in Proc. IEEE-IEE Vehicle Navigation & Information Systems Conf. - VNIS '93, Ottawa, Oct. 1993, pp. 271-276.

User densities in radio communication systems with dynamic channel assignment

Dipl. Ing. Günter Kleindl
Siemens AG Austria, EZE
A-1030 Vienna, Erdberger Lände 26

Abstract

In this paper a multi-cell radio communication system with dynamic channel assignment is considered. For such a system the achievable channel-densities are calculated and the influence of parameters like the required signal to interference ratio, usage of power control or the propagation conditions are investigated.

1 Introduction

Mobile communications is a rapidly growing market and the analogue cordless and cellular systems of today are now being substituted by new digital systems. And already now the work on the next generation of digital radio communication systems, which will lead us into the year 2000, has started. All these evolving new mobile services require spectrum, which is a limited resource. As there is an ever growing demand for higher capacity, the efficient use of spectrum is an essential requirement for modern communication systems. In order to handle very high user densities and not to use up too much spectrum, channel reuse is one of the most powerful means to tackle this problem.

In this paper first a performance measure for the efficiency of a multi-cell system is derived. From this formula it can be seen that for the comparison of different system solutions the channel reuse is an important factor. Under the assumption of evenly distributed users an analytic formula for the channel reuse has been derived. The theoretic result is compared with the result obtained by computer simulations. For the simulations a dynamic selection of the best channel is used. In the case of interference the disturbed connection is switched to a better channel, if available. The channel reuse which can be achieved depends on many parameters, like the required signal-to-interference ratio, the channel selection algorithm, the propagation conditions, handover possibilities, power control, the number of interfering cells, the service quality and many others. The dependency of the channel reuse on such parameters has been investigated by means of computer simulations.

2 Spectral efficiency

In the CCIR report 662 [1] the following definition of spectral efficiency is given:

$$Efficiency = \frac{Communication}{Bandwidth * Time * Space} \tag{1}$$

Using this definition, a simple calculation leads to a formula which is especially useful for judging the efficiency of cellular communication systems:

$$Efficiency = \frac{\text{channel\_informationrate}}{channel\_bandwidth} * channel\_density * \frac{1}{cell\_size} \tag{2}$$

The channel-information rate is the information per time which can be carried by a single communication channel and the channel-bandwidth is the spectrum which is needed for such a single communication channel. The first factor in formula (2) containing the relation of the channel-information rate to the channel-bandwidth depends e.g. on the source coding and on the modulation type. A more narrow band modulation method improves this factor but requires usually a higher signal to interference ratio, which reduces the channel-density and therefore does not automatically improve the spectral efficiency. For this reason it is important to know how the channel-density is influenced by the required signal to interference ratio.

Because of the interference from the other cells, not all available channels are usable in every cell. The number of usable channels per cell divided by the total number of channels is defined as channel-density and is equal to the reverse of the reuse-factor.

As can be seen from (2) reducing the cell size is one of the most effective measures to increase the efficiency. Therefore it is interesting to know, if the channel-density is influenced by the cell-size.

3 Estimation of Channel Reuse and Channel Density

Let us assume a multi-cell radio system with evenly distributed portables. For calculating the signal strength S the following propagation law is used:

$$S = Konst * r^{-n} \tag{3}$$

In the above formula r is the distance from the signal source and n is the propagation exponent which is typically in the order of 4 in the case of indoor propagation but varies a lot depending on the scenario.

When assuming that the channels are used periodically by the portables then the resulting signal to interference ration at the base station for a communication with a portable at the cell boundary can be calculated as:

$$\left(\frac{S}{I}\right)^{-1} = \sum_{j=1}^{\infty}\left(\frac{1}{1+j*\mathrm{Re}use_{min}}\right)^{n/2} = \sum_{j=1}^{\infty}\left(\frac{1}{1+j/Channel\_density_{max}}\right)^{n/2} \tag{4}$$

By using an approximation for the sum the following result can be obtained:

$$\mathrm{Re}use_{min} = \frac{1}{Channel\_density_{max}} \cong C*\left(\frac{S}{I}\right)^{\frac{2}{n}} \tag{5}$$

for S/I > 1 and with $C \cong 1\ldots2$ depending on the value of n.

4 Simulation of user densities

4.1 Simulation-scenario

For investigating the channel-density a multi-cell system consisting of hexagonal or square sized cells is used. As result the channel-density is calculated as a function of the required signal to interference ratio. For comparison purposes one reference curve is defined which is obtained by using the following parameters:

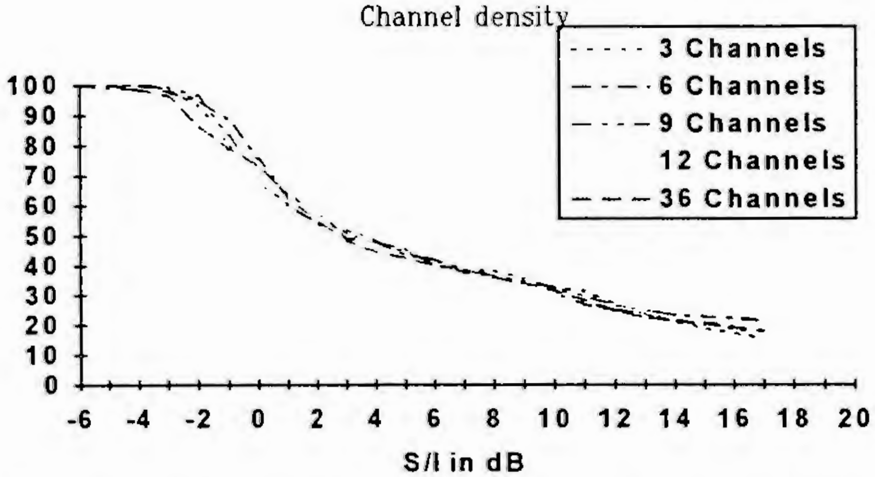
- system with 128 hexagonal cells
- 10 orthogonal communication channels are available
- propagation exponent n = 4
- no power control is used
- the channel-density is increased until a first interruption occurs

The selection of a channel is done on basis of the best signal to interference ration at the fixed and the portable station and is done dynamically and decentralised. If an existing connection is interrupted due to increasing interference and a better channel can be found, then handover is performed.

4.2 Comparison of the analytical estimation with the simulation result

For the calculation of 'channel-density$_{max}$' an optimum channel-distribution has been assumed outside the centre cell. The simulation shows that if a high signal to interference ratio is required, then this maximum channel-density can be almost achieved. But for low signal to interference values the practically achievable channel-density is lower, because of the interference between the outer cells, which does not allow to optimise the channel distribution for the central cell only. It is possible to take this fact into account by using a sub optimum channel-distribution. Under this assumption the same dependency as given in formula (5) can be derived, only the factor C is a little bit higher compared with the optimum case. When assuming such a channel distribution, then the simple analytical result matches very well with the simulation results.

4.3 Influence of the channel number on the channel density

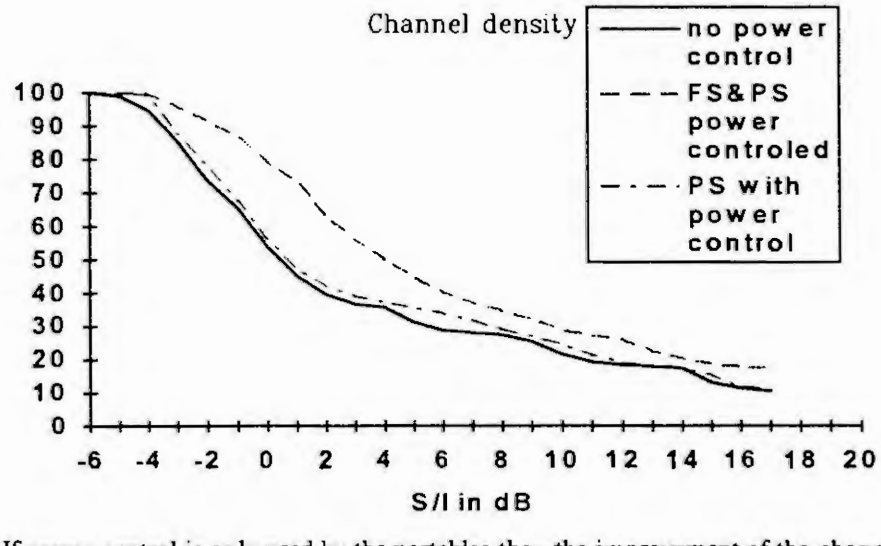


Because the channel-density is defined as the ratio between the usable channels and the maximum number of channels it is independent from the actual number of total available channels. This has been confirmed by the simulations.

4.4 Influence of the cell shape on the channel density

With hexagonal cells a slightly better channel-density can be achieved then using rectangular cells, because the distance between the portables at the cell boundary and the base station is smaller.

4.5 Influence of power control on the channel density

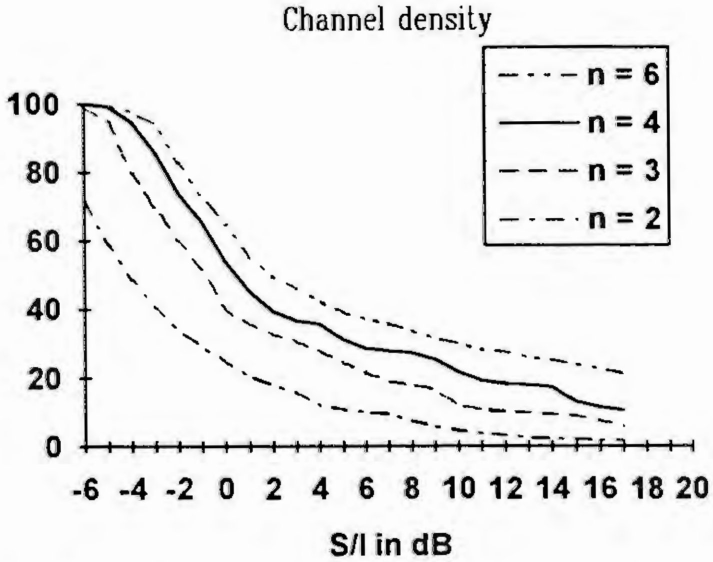


If power-control is only used by the portables then the improvement of the channel-density is small, because in this case the channel-density is mainly limited by the down link. Only if both the portable and the fixed station apply power-control, then a significant improvement of the channel-density can be achieved.

4.6 Influence of handover on the channel density

Handover principally leads to an improvement of the channel-density. In case a good channel selection algorithm is used, this improvement is rather small for static environments however for an environment with moving users handover is a very important feature.

4.7 Influence of the propagation conditions on the channel density



The influence of the propagation conditions on the channel-density is very significant.

4.8 Influence of the cell size on the channel density

If the propagation exponent is constant, then the channel density does not depend on the cell size. Practical results obtained from indoor propagation measurements show that typically the attenuation coefficient is lower for short distances and higher for long distances. Such a propagation law leads to slightly higher channel densities for larger cells then for smaller cells. Instead of using an attenuation coefficient varying with the distance a constant value can be used and gives about the same results, if the value of the propagation coefficient at the cell boundary is used as the constant value. This also explains the slightly higher channel densities for larger cells because of the higher attenuation coefficient at the boundary of a larger cell. But this effect of channel density reduction for small cells is minor compared with the user density increase obtained with smaller cells which is still almost proportional to the inverse of the cell size.

5 Summary of results

For a multi-cell radio communication system with dynamic channel allocation the achievable channel densities have been investigated. A formula has been derived which shows the dependency of the channel density on the required signal to interference ratio and the propagation exponent, which are the two most important factors. This formula has been verified by means of computer simulations, which further show the importance of using an efficient channel selection algorithm. It can also be seen that power control increases the efficiency only if used by the portable and the fixed station and that handover gives only a small improvement in the case of static scenarios but is important for dynamic scenarios.

6 References

[1] CCIR Report 662, "Definition of Spectrum Use and Efficiency", Documents of the XIVth Plenary Assembly, Kyoto, 1978

Macroscopic Simulation of Single Vehicles in Motorway Networks

Dipl.-Inform. Christiane Theis
Universität Karlsruhe, Institut für Verkehrswesen
Kaiserstraße 12, Postfach 6980
76128 Karlsruhe
Germany

Abstract: This paper gives a description of the mesoscopic simulation model DYNEMO which was developed in the 1980s at the Institut für Verkehrswesen in Karlsruhe. It allows to simulate larger motorway networks in a macroscopic way but also to watch single vehicles on their way through the network. Since information about the state of the network can be given to the vehicles, DYNEMO can be used to investigate the influence of route guidance systems on traffic flow. Additionally fuel consumption and emissions of the simulated vehicles can be calculated.

1. Introduction

For estimating the influence of route guidance systems on traffic flow by simulation, a model is needed which can simulate large networks with high traffic loads. Thus a macroscopic model describing the traffic by parameters as density (number of vehicles per kilometre), volume (number of vehicles per hour) or mean speed would be the suitable tool. To obtain information about the route decisions of the vehicles in the simulated network or about their travel time it is useful to watch these single microscopic elements on their way through the network.

The DYNEMO model joins both a macroscopic model for describing the traffic state in the sections of the network and single vehicles as microscopic elements.

DYNEMO was first developed at the Institut für Verkehrswesen in Karlsruhe [4] and was in 1992 integrated in the VISUM system of the consultancy PTV system in Karlsruhe, where further development of the model takes place.

2. The Traffic Flow Model

In DYNEMO a stretch is divided up in sections and is characterized by a speed-density-relation. Each section of the stretch is regarded as an area with a constant traffic state characterized by its mean speed and density during one time step. The length of the section does not change during the simulation. Figure 1 gives an impression of the sections and characteristics a stretch consists of.

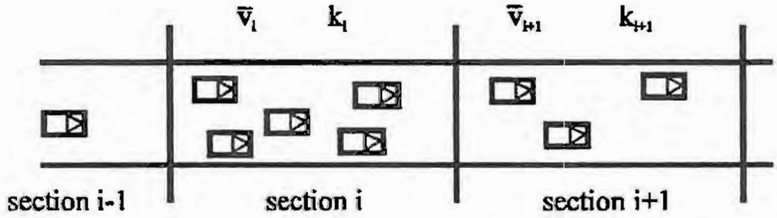


Figure 1: *Sections of a stretch characterized by mean speed and density*

The speed-density-relation describes the possible states of traffic that can occur in the sections of the stretch. Figure 2 shows the graph of such a relation between mean speed and density. The mean speed is calculated from the speeds of all vehicles in a given section, the density means the number of vehicles divided by the length of this section. It can be seen that the mean speed decreases by increasing density. Each point of this curve represents one traffic state.

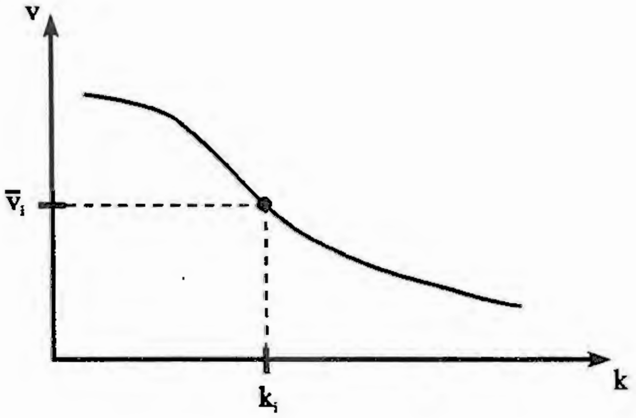


Figure 2: *A speed-density-relation*

During the simulation the whole system is updated in time steps, e.g. every 10 seconds. The states of all sections are calculated from the states in the time step before. First the number of vehicles in a section for the next time step is determined. The number of vehicles in section i is the result of the number, position and speed of the vehicles in sections i-1 and i in the time step before. The speed-density-relation of the regarded stretch gives the mean speed v_i according to that new density. This forms the macroscopic part of the simulation model.

The microscopic part of the model is represented by single vehicles with individual parameters. Each vehicle has a desired speed out of a desired speed distribution and it has a type. This vehicles are now moved on the stretch due to the macroscopic parameters of the section they are located in. The mean speed of all the vehicles in a section must be the mean speed calculated as described above. But the speeds of the single vehicles can differ from this mean value in that way that vehicles with a higher desired speed drive faster than vehicles with a lower one. The variation of the speeds of the vehicles in one section decreases with increasing density, that means that in sections with higher loads the traffic state is of much more importance for the speed of a vehicle than its desired speed.

3. The Network

The network consists of stretches with lanes, stretches are linked together with nodes. At every node the possible turns from incoming to outgoing stretches are defined. A desired speed distribution is assigned to every stretch. Each lane has its own speed-density-relation taking into account that on different lanes a different range of traffic states can occur. Stretches can be closed for certain types of vehicles, e.g. for trucks.

New vehicles are created in traffic sources and put into the system at the beginning of stretches and are taken out at the end of stretches. An origin-destination-matrix must be given to determine the volumes to be put into the system and to find the destination for a vehicle starting its way at a certain point.

4. Ways Through the Network

The way a vehicle takes through the network to reach its destination can be given by the user or calculated by best way algorithms following certain criteria as estimated travel time, length of way or costs. (Costs for passing a stretch or a node can be defined by the user.)

For modeling the influence of route guidance systems (as for example RDS-TMC), vehicles must obtain information about the traffic state of the network and recommendations about new routes to their destination while they are on the way. In DYNEMO decision points can be put anywhere in the network, on stretches or in traffic sources. These decision points contain specific information for the different types of simulated vehicles. For every type of vehicle new routes to every destination can be provided. Additionally for each route the percentage of vehicles obeying it must be defined. Such new routes can be updated during the simulation in given time intervals, so that the route guidance can react dynamically on the actual loads of the stretches. It is also possible to define a delay of providing the information as in reality the driver cannot be informed immediately about the actual state of traffic.

5. Additional Features and New Development

In 1988 a module was added to the simulation model to estimate fuel consumption and emissions of the vehicles [3]. The calculation is based upon engine maps and regards speed and acceleration of the vehicles at every time step.

The DYNEMO model was developed to investigate traffic flow on highways. It is now extended to simulate urban networks as well. This means that nodes are no longer only connections between stretches but they must be modeled as intersections with traffic signals or priority rules. The simulation of buses and trams will also be possible in the next version of DYNEMO.

6. Literature

[1] Cremer, M., *Der Verkehrsfluß auf Schnellstraßen*, Springer-Verlag, 1979 (in German).

[2] Leutzbach, W., *Introduction to the Theory of Traffic Flow*, Springer-Verlag, 1988.

[3] Schnittger, S., ***Simulation des Verkehrsflusses in Fernstraßennetzen unter Beachtung des Kraftstoffverbrauchs, der Abgaswerte und der Reisezeit,*** Forschung Straßenbau und Straßenverkehrstechnik, Heft 559, Bundesminister für Verkehr, Bonn 1988 (in German).

[4] Schwerdtfeger, T., ***Makroskopisches Simulationsmodell für Schnellstraßennetze mit Berücksichtigung von Einzelfahrzeugen (DYNEMO)***, Forschung Straßenbau und Straßenverkehrstechnik, Heft 500, Bundesminister für Verkehr, Bonn 1987 (in German).

Microscopic Simulation of Traffic Flow

Dipl.-Inform. Uwe Reiter
Universität Karlsruhe, Institut für Verkehrswesen
Kaiserstraße 12, Postfach 6980
76128 Karlsruhe, Germany

Abstract. Very often investigations in potential effects of different measures on traffic flow cannot be carried out in real world traffic. Models have to be used, representing traffic as detailled as needed. The analysis concerning changements in individual driving behaviour requires very detailled models. A microscopic approach to modelling traffic is presented in this paper. Its empirical background is described as well as some examples of application of the model.

1. INTRODUCTION

In various disciplines scientists and engineers develop models describing processes of those parts of the reality, they are dealing with. Traffic engineers are looking at the movement of vehicles and were developing models that represent these movements: models of traffic flow. Different types of traffic flow models have been developed, that have to be destinguished by their level of abstraction. On one hand there are the macroscopic simulation models forming the highest level of abstraction. Macroscopic models are looking at quantities of vehicles. They represent traffic flow as the movement of a liquid, and they describe it by aggregated parameters like traffic volume, traffic density and mean speed. Whereas, on the other hand, microscopic models treat the vehicles individually. This is the lowest level of abstaction. Here the movement of individual vehicles is described as a result of actual traffic conditions, for example the relations to surrounding vehicles and road conditions. Other types of models, socalled mesoscopic models, form the inbetween levels of abstraction. One microscopic approach will be described in this paper.

2. A MICROSCOPIC SIMULATION MODEL

The development of microscopic traffic flow models has a long tradition at the ***Institut für Verkehrswesen*** at the University of Karlsruhe. The investigations started in the late sixties and led to the development of a microscopic simulation model for traffic on unidirectional carriageways based on human perception of and on reaction on changes in distance and speed difference to surrounding vehicles. Within this model the movements of individual vehicles on both, urban and interurban carriageways are described. Its kernel is formed by two models representing the basic vehicle movements: the longitudinal and the lateral movement. The first deals with the movement on one lane, concerning models are often called car following models. The latter focusses on the movement between lanes, models are called lane changing models.

2.1 Car Following Model

Wiedemann [8] developed a model representing the longitudinal movement as a reaction to the perception of distance and speed difference to the leading vehicle. He defines four different states of interaction relative to the front vehicle. Each of these states is related to one basic type of driving behaviour. The first is called uninfluenced driving behaviour. No front vehicle is obstructing the own movement, the driver is driving at his desired speed or tries to reach it by accelerating. The second is the closing behaviour. The driver has perceived that he is approaching a slower vehicle in front. He reacts by decelerating, trying to reduce his own speed to the speed of the front vehicle always keeping a safe distance. The third is the following behaviour, where the driver follows the front vehicle at low speed differences and at a distance near the desired following distance. The driver tries to keep speed difference and acceleration low. Finally, the emergency braking behaviour represents the behaviour in case of emergency situations, with the actual distance being lower than the minimum desired distance. The driver will react trying to avoid an accident and to reach the minimum distance again.

In the model the four types of driving behaviour are separated by socalled perception thresholds, representing, that a driver perceives a changement in distance or speed difference only if the physical stimulus exceeds certain minimum values. In this case the physical stimulus is produced by changements in the seen size of the front vehicle, depending on the distance to the front vehicle (for the size) and on speed difference (for the velocity of changements). Therefore, the thresholds are defined to be functions of distance and speed difference and are illustrated in the phase plane below (Fig. 1) with the x axis being the speed difference and the y axis the distance to the front vehicle.

The threshold *SDV* indicates the perception of speed differences at high distances. *BX* is the minimum distance, the driver values to be safe at the actual speed, whereas *AX* is the minimum distance for standing vehicles. *CLDV*, *OPDV* and *SDX* delimit the following behaviour describing that the driver perceives speed differences causing an closing or an opening process or gradually increasing distances respectively.

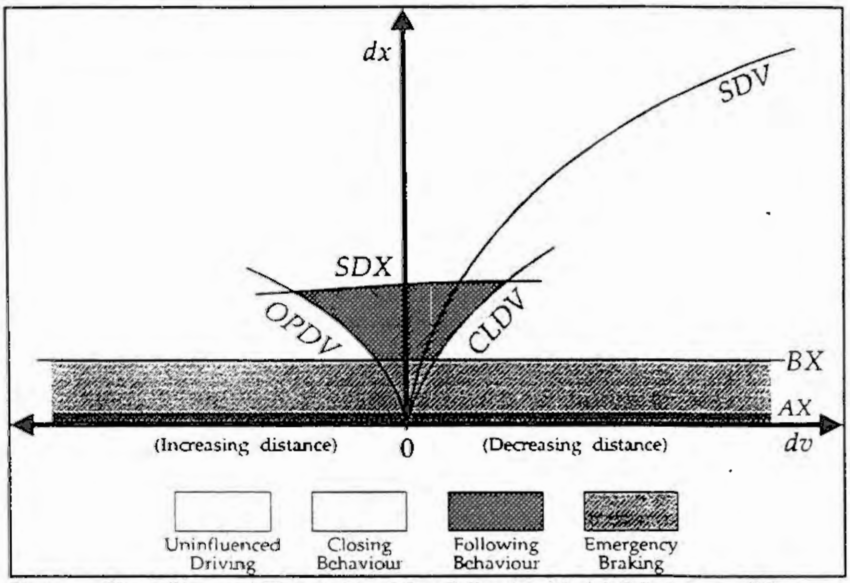


Fig.1: Phase Plan and Thresholds of Car Following Model (Wiedemann [8])

Differences in perception, reaction and driving abilities between drivers and differences in properties between vehicles are considered in the model. The functions calculating thresholds and the procedures representing driving behaviour are not deterministic, but are characterized by the use of normal distributed parametres leading to a natural distribution of the calculated values.

2.2 Lane Changing Model

The lateral movement of interest on unidirectional carriageways is the movement between lanes. Human lane changing behaviour is the result of parallel decisional processes, that have to be simplified in the model. The lane changing behaviour is represented by a sequential algorithm, the decision to change the lane depending on the following three questions: Is a lane change desired, for example because of an obstruction on the actual lane? Is the situation on a neighbouring lane and hence a change to that lane advantageous? Is the movement to the neighbouring lane possible and is it safe?

The decision to change the lane is again based on human perception of the interesting traffic situation. Hence, the lane changing behaviour is modelled in a similar way as the car following behaviour. Human estimation of distances and speed differences needed for the three decisional questions is represented by analogous thresholds as above. This model has been developed by Willmann [9].

3. EMPIRICAL BACKGROUND

The strength of the microscopic model is its empirical background. All stages of model development (model design, calibration and validation) were based on extensive empirical investigations of traffic flow and of human driving behaviour. The model has continously been calibrated and validated against empirical data. The development of the car following model was based on psycho-physical studies and on extensive measurements on driving simulators and in traffic flow: Michaels [3], [4], Todosiev [7], Hoefs [2]. The basis of development of the lane changing model was formed by measurements on a stretch of German motorway, on a two-lane carriageway first (Sparmann [6]) and on a three-lane carriageway later (Busch [1]). In the years that followed continous measurements were undertaken furnishing data for both basic models. Models were calibrated with microscopic data, they were continously validated and revalidated using macroscopic data. The result is a very detailed traffic flow model based on the measurable behaviour of individual drivers.

Since the end of 1992 very detailed measurements were undertaken enabled by the possibility to use an especially equipped car. The equipment consisted in different sensors (radar and infra-red), first only in front and later also at the back of the car allowing to measure distances and speed differences to front vehicles and to following vehicles respectively. In the first two series the driving behaviour of 14 test drivers in the measuring vehicle was recorded and evaluated in relation to the situation in front of the vehicle. In the third series, vehicles following the measuring vehicles were recorded. Both types of measurements are used for a new calibration of the car following model. Results can be presented at the conference.

4. UTILISATION OF THE MODEL

The very detailed representation of driving behaviour was the basic requirement for using the model in different applications, especially when potential changements in driving behaviour itself are to be investigated. The model can be employed to assess potential effects on traffic flow in all kind of investigations that are not possible, too dangerous or too expensive, when undertaken in real world traffic.

Some examples for that kind of investigations are: Analysis of the effects of a general speed limit on German Autobahn of 100 km/h, 120 km/h or 130 km/h; analogous investigations in a speed limit only for a proportion of vehicles: those vehicles without catalytic converter; investigations in vehicle emissions under different general conditions for both pollutant and noise emissions; different investigations of changed driving behaviour, for example a socalled ecological driving behaviour. In most of these investigations, the very detailled way of representing traffic was needed to be able to model the interesting changements correctly. But the model was used to estimate more general, often macroscopic, effects like capacity, safety and emissions.

A very recent investigation has been carried out within the framework of the European R&D Programme DRIVE reported for example in [5]. Here the model was used to estimate possible effects of onboard information systems, supporting the driver in his driving task, such as headway control, speed control or lane changing advice. Two types of influence were defined: a supported mode with the driver being supported by additional information and a controlled mode passing control over vehicle movements to the system. A theoretical model was developed, describing both types of influence of these information technologies on driving behaviour. This model was integrated into the simulation model and connected to the model of natural human driving behaviour. Simulation runs were undertaken looking at potential benefits of these technologies on road capacity and traffic safety. Differences in traffic flow between not supported, partially supported and completely supported driving as well as between different proportions of equipped vehicles are shown to be a result of these investigations.

Furthermore it was possible to connect the model on-line with other programes. One example was an investigation in socalled Intelligent Variable Message Signs (IVMS), where the model was used as a tool to test the IVMS-implementation in a closed-loop-procedure: the model supplied the IVMS with traffic information, while the IVMS used this information to control different message signs for example indicating different speed limits, that did again influence the simulated traffic flow. Another application was the connection of the traffic simulation with an AI-system controlling traffic lights in a simulated urban network (part of the city of Hamburg). The AI-system again used traffic measurements to control the traffic lights.

5. CONCLUSION

The level of detail necessary in modelling real world processes depends on the kind of investigation. The same holds for traffic flow models. It was shown that the described microscopic approach has been especially useful for investigations concerning changements of individual driving behaviour. But for scientific investigations it is important to consider, that the more detailled the used model is, the more emphasis has to be put on empirial studies forming the empirical background of the model (calibration and validation).

6. REFERENCES

[1] Busch, F., Leutzbach, W., *Spurwechselvorgänge auf dreispurigen BAB-Richtungsfahrbahnen*, Forschungsauftrag 1.082 G 81H des Bundesministers für Verkehr, 1984 (in German)

[2] Hoefs, D.H., *Untersuchung des Fahrverhaltens in Fahrzeugkolonnen*, Straßenbau und Straßenverkehrstechnik, Heft 140, Bundesminister für Verkehr, Bonn 1972 (in German)

[3] Michaels, R.M., *Perceptual Factors in Car Following*, Proceedings 2nd International Symposium Theory of Road Traffic Flow 1963, OECD Paris 1965

[4] Michaels, R.M., Cozan, L.W., *Perceptual and Field Factors Causing Lateral Displacement*, Highway Research Board, Highway Research Record Number 25, 1963

[5] Reiter, U., *Modelling the driving behaviour influenced by information technologies*, Proceedings of the International Symposium on Highway Capacity, Balkema, Rotterdam 1991

[6] Sparmann, U., *Spurwechselvorgänge auf zweispurigen BAB-Richtungsfahrbahnen*, Straßenbau und Straßenverkehrstechnik, Heft 263, Bundesminister für Verkehr, Bonn 1978 (in German)

[7] Todosiev, E.P., *The Action Point Model of the Driver-Vehicle-System*, Engineering Experiment Station, The Ohio State University, Report No. 202A-3, 1963

[8] Wiedemann, R., *Simulation des Straßenverkehrsflusses*, Schriftenreihe des Instituts für Verkehrswesen der Universität Karlsruhe, Heft 8, 1974 (in German)

[9] Willmann, G., *Zustandsformen des Verkehrsablaufs auf Autobahnen*, Schriftenreihe des Instituts für Verkehrswesen der Universität Karlsruhe, Heft 19, 1978 (in German)

SIMCO3++: SImulation of Mobile COmmunications based on Realistic Mobility Models and Road Traffic Scenarios

Carl-Herbert Rokitansky, Christian Wietfeld and Christian Plenge
Aachen University of Technology
Communication Networks
Kopernikusstr. 16
52074 Aachen,Germany
Phone:++43/241/807911 Fax:++43/241/84964

Abstract

The performance evaluation and simulation of mobile communication networks requires the realistic and efficient modelling of the movements of mobile stations. In this paper, the mobility model of the integrated simulation tool "SIMCO3++" (SImulation of Mobile COmmunications) for tbe performance evaluation and verification of short-range vehicle-beacon and inter-vehicle communication protocols is presented and validated with motor-way measurements performed by the Dutch Ministry of Transportation (Rijkswaterstaat). The results of a comparison of the motor-way measurements and the traffic scenarios simulated by SIMCO3++ are discussed. The comparison shows a very good correspondence in important aspects like following distances between vehicles, average speed of vehicles, distribution of vehicle classes over the lanes.

1 Introduction

New RTT applications will require more or less extensive communications to exchange relevant information between vehicles and roadside beacons (e.g. Automatic Fee Collection, Route Guidance, Parking Management, Medium Range Preinformation, Intelligent Intersection Control, Emergency Call, etc.) and between vehicles (Intelligent Cruise Control, Intelligent Maneuvering Control, Lane Access, Emergency Warning, etc.). To guarantee the functionality of the developed communication protocols and RTT applications and to optimize the parameters under various environmental conditions, computer-simulations are essential for the system design, as well as for the specification of standards (CEN / TC278) for an operational Integrated Road Transport Environment (IRTE) network [1].

The mobility model used in SIMCO3++ to simulate the movement of vehicles (private cars, trucks, busses, etc.) under various environmental conditions (multi-lane motor-ways / rural roads with section-wise speed limits, intersections, etc.) has been validated with Dutch motor-way measurements performed in 1991. The basic simulation model (vehicle movements, communication protocols, data exchange, and RTT applications) is described in Section 2. In Section 3 the road traffic characteristics, which are relevant for communications, are described. The modelling of various traffic scenarios and mobility characteristics is discussed in Section 4. In Section 5, the Dutch motor-way measurement scenarios are described. The results of the Dutch motor-way measurements and the corresponding SIMCO3++ simulation results are compared in Section 6. Finally, the conclusions from these comparison and a summary of SIMCO3++'s further extension are discussed.

2 SIMCO3++ Simulation Model

For the performance evaluation of communication protocols, a simulation model is required, which allows the integrated simulation of both, vehicle movements in a dynamic network and the communications between vehicles and roadside beacons and between vehicles [7]. Figure 1 shows the basic building blocks of such a simulation model and their interdependency. First, realistic vehicle dynamics, based on mobility mechanics, traffic statistics, environmental conditions and driver behaviour must be simulated. The

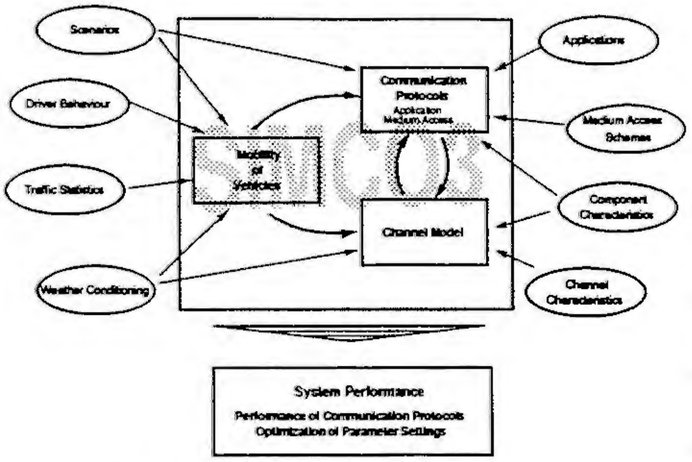


Figure 1: Simulation concept SIMCO3++

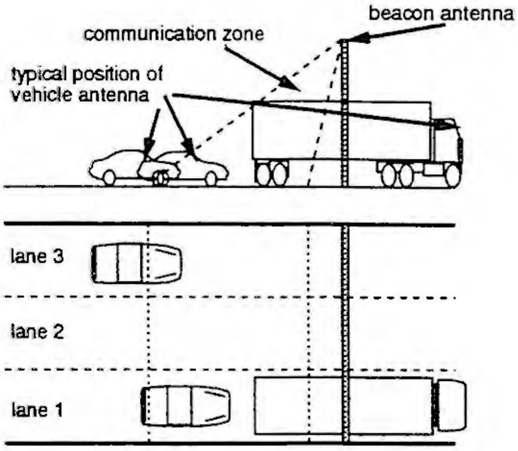


Figure 2: Vehicle-Beacon-Communication

exchange of information of current vehicle and road characteristics, conditions, fixed and dynamic traffic situation and restrictions (e.g. speed limits, traffic lights) are important for the IRTE system and require communication links to roadside infrastructure and/or between vehicles.

For these communications, which might be single- or multi-hop, specific communication protocols (medium access control, logical link control, routing strategies, etc.) are currently being developed by communication groups of the DRIVE II programme and within standardization bodies (e.g. CEN / TC278). Due to their interdependency with the vehicle movements, the environmental conditions, and the current traffic scenario, the protocol performance should be evaluated by integrated simulations of the dynamic network and the corresponding data flow. Communication relevant parameters, like channel characteristics, roadside communication infrastructure, etc. are taken into account in these simulations.

Computer-simulations based on this realistic simulation model, provide the required results for tbe determination of minimum requirements / optimal values of communication characteristics, and allow an accurate performance evaluation of the developed communication protocols.

3 Road Traffic Characteristics Relevant for Communications

The modelling of the traffic should be as accurate as necessary (concerning the effects of the mobility on the performance of the communication protocols) and as lean as possible (in order to allow efficient implementation in the simulator). Therefore it is necessary to analyze, which characteristics of the road traffic are relevant for the communications. In vehicle-beacon communications, a relatively short section of a motor-way (up to around 100 m) is relevant, whereas in vehicle-vehicle communications a longer motor-way section has to be regarded (several kilometres). The topology of the network (relative position of vehicles towards each other) is of special importance for the protocol functionality and performance for inter-vehicle communications. In the following, the specific characteristics regarding vehicle-beacon communications are discussed in more detail:

Speed and vehicle types Each beacon provides a characteristic communication zone, which depends on a number of parameters, such as the antenna configuration and transmission medium. As the length of the communication zone is limited in any case, the speed of the vehicles determines the available communication time of each vehicle. In addition, the vehicle type has to be taken into account. Figure 2 shows the simplified model of a communication zone (microwave): the zone gets shorter, the higher vehicle antenna is positioned. Since the typical position of the vehicle antenna depends on the vehicle types, different vehicle types have to be taken into account.

For the calculation of shadowing effects additional vehicle-type specific parameters have to be taken into account (see also figure 2): the height of the vehicle's antenna, its longitudinal position (distance from vehicle front) and the height of the vehicle in front. Furthermore, the exact knowledge of the following distances between vehicles is necessary (see below).

Traffic Intensities and Distributions of Inter-arrival Times Due to the characteristics of the communication of vehicle-beacon communications, it may occur, that several vehicles transmit data at the same time in the same communication zone (see figure 2). Therefore the characteristics of the free traffic flow have a strong influence on the systems performance: the traffic intensity as well as the distribution of the inter-arrival times (following distances of vehicles on a specific lane) are of importance. The higher the percentage of vehicles, that have very short inter-arrival times, the higher the probability, that the protocols have to cope with data collisions [3] [6] [5].

4 Modelling of Road Traffic Scenarios

The simulation tool SIMCO3++ has been designed to fulfill the simulation requirements of performance evaluation of new communication protocols. In the following section, the model approach for the simulation of various traffic scenarios is discussed in detail.

4.1 Simulation Scenarios

The new mobile communication protocols, currently being developed for vehicle-beacon and inter-vehicle communication systems must provide optimal functionality for traffic scenarios with different characteristics. Therefore the following classes of road traffic scenarios can be simulated by SIMCO3++: motor-ways, rural roads, intersections, road narrowing scenarios, access ramps. Up to 6 lanes and a lane-specific traffic intensity can be specified for each direction. Special road characteristic like speed limit, blocked lanes can be added. All maneuvers of vehicles, that are implemented in the mobility model (see section 4.3) are influenced by the specified road conditions.

4.2 Vehicle Generation

The initial generation of vehicles and their basic characteristics is one of the key problems in realistic traffic models. SIMCO3++ allows to generate (and simulate) several vehicle classes (private cars, vans, trucks, etc) with the following statistical properties: overall percentage of the class (lane specific), average speed and inter-arrival time, reaction time, set of risk factors, maximum speed, intended speed and vehicle length. These characteristics include all those parameters, that are necessary to ensure both a realistic traffic generation and a realistic behaviour of vehicles.

Whenever a new vehicle is generated, its individual intended speed, its set of risk factors, the reaction time, the vehicle length and height, etc. is assigned, using a (pseudo) random number generator according to statistical distribution determined by traffic measurements.

4.3 SIMCO3++ Mobility Model

The mobility model of SIMCO3++ is based on a microscopic view of the traffic. The behaviour of all vehicles depends on a set of rules and was designed to determine the reaction of a vehicle according to its local traffic environment. These rules take into account factors such as acceleration, deceleration, overtaking maneuvers, merging maneuvers, selection of the preferred lane, etc. The periodical update of the mobility scenario in small mobility time steps (several ms) ensure the continuous movement of all vehicles and creates a realistic traffic flow.

As the decisive factor for vehicles' reaction, risk factors were introduced, that are calculated before each mobility step [2]. The actual speed of surrounding vehicles and the distance between them are used to determine a risk factor for each direction. These risk factors are used to assess the actual traffic situation the vehicle has to react in. The factors are compared with a set of 'maximum risk factors': each set is specific for each vehicle and influences the vehicle's traffic behaviour. By parameterizing these maximum factors with a distribution function, different types of driver' behaviour were modelled in the mobility model. In normal situations, the calculated risk factors are smaller than the vehicle's maximum factors. Therefore the vehicle attempts to drive with its intended speed. It also tries to move to its preferred lane if it was caused to change lane by former driving activities. If the risk factors exceed the vehicle's maximum risk factors, a corresponding rule (e.g. about deceleration or overtaking) initiates the required vehicle actions.

5 Road Traffic Measurements on Motor-Ways

The mobility model of SIMCO3++ was validated by comparison with Dutch motor-way measurements [2] carried out as part of a project commissioned by the Transportation and Traffic Research Division of Rijkswaterstaat. Measurements were done during several months in 1991 on the A2 motor-way between Utrecht and Amsterdam, via the research facility of the Motor-way Control Signalling System (MCSS) yielding arrival instants, lane, speed and length of passing vehicles at 16 cross-sections [1] [4]. A study section of 2.9 km was chosen between cross-section A2E53.200 and A2E50.300 in the direction Amsterdam to Utrecht. This is the only section along which there are no entrances or exits. There is however, an exit about 400 m after the end of the study section. This has an influence on the measurement data, as will be discussed in the next session.

For the validation of the mobility model of SIMCO3++, different measurements were analysed. For each measurement site, the following parameters were measured per vehicle:

- lane, in which the measured vehicle was in, indicated by the lane number (1=right/2=middle/3=left lane)
- Arrival time (in hrs:min:sec) of the vehicle
- Current speed of the vehicle (in km/h)
- Length of the vehicle (in m) from which the vehicle class can be derived: e.g. 'Car': *length* <= 5 meter; 'Truck': *length* > 5 meter

To be able to compare the measurements with the simulation results, the following methods were applied:

1. The traffic statistics (average speed of vehicles, standard deviation of speed, traffic intensity, etc.) at the beginning of the study section were computed and specified as simulation input parameters for SIMCO3++ (dataset MES1). Based on these road traffic characteristics, vehicles were generated by SIMCO3++ (dataset SIM1) and simulated according to the mobility model described in section 4. Finally the traffic statistics at the end of the simulation range, corresponding to the distance between the measurement sites were computed and reported in the simulation results file (dataset SIM2) in order to compare them to the measurements (dataset MES2 holds data from the end of the study section).
2. Instead of generating the vehicles according to the measured traffic statistics, it is also possible to feed SIMCO3++ directly with the measured data (arrival time, lane, vehicle class, vehicle-length), to simulate the further behaviour of the injected traffic according to the mobility model of SIMCO3++, and finally report the simulation results (traffic statistics) at the end of the study section (second measurement site)

6 Comparison of the SIMCO3++ Model with Measurements

The table 1 and the figures 3, 4, 5, 6 show characteristic results of a comparison of a 3-hours measurement on the A2 with high traffic intensity and data generated by SIMCO3++ are shown. The data at the beginning of the study section (datasets MES1 and SIM1; vehicle generation using method 1 as described in the previous section) and at the end of the section (after 2.9 km; datasets MES2 and SIM2) is presented. The comparisons of the measurements and the SIMCO3++ results show:

1. Comparison at the beginning of the study section (MES1 vs. SIM1)
The lane-specific traffic intensity, the mean parameters speed and inter-arrival time as well as the distribution of vehicle classes correspond very good: the maximal relative error between the measured data and the data generated by SIMCO3++ is around 6 % (table 1). Furthermore the distributions of inter-arrival times and speeds for the different lanes show as well a good correspondence (figures 3 and 4).

Parameter	Lane	Beg. of Section		End of Section		Comparisons		
		Measur.	SIMCO	Measur.	SIMCO	MES1	MES1	MES2
		MES1	SIM1	MES2	SIM2	SIM1	SIM2	SIM2
intensity (veh/h/lane)	1	1212	1212	1040	1298	1.2 %	7.1 %	24.8 %
	2	1704	1802	1752	1634	5.6 %	-4.0 %	-6.7 %
	3	1333	1353	1460	1456	1.5 %	9.2 %	-0.4 %
mean speed (km/h)	1	95.2	94.7	94.3	87.5	-0.5 %	-8.1 %	-7.2%
	2	109.3	107.6	109.6	106.8	-1.1 %	-2.3 %	-2.6 %
	3	118.3	118.0	118.6	116.4	-0.3 %	-1.6 %	-1.9 %
mean inter-arr. time (sec.)	1	2.66	2.70	3.24	2.53	1.5 %	-5.0 %	-21.0 %
	2	1.98	1.85	1.93	2.05	-6.6 %	3.7 %	6.2 %
	3	2.59	2.53	2.36	2.34	-2.3 %	-9.6 %	-0.8 %
car/truck distr. (%)	1	77/23	77/23	71/29	80/20			
	2	97/3	97/3	98/2	96/4			
	3	100/0	100/0	100/0	100/0			

Table 1: Comparison of measured data and SIMCO3++ data

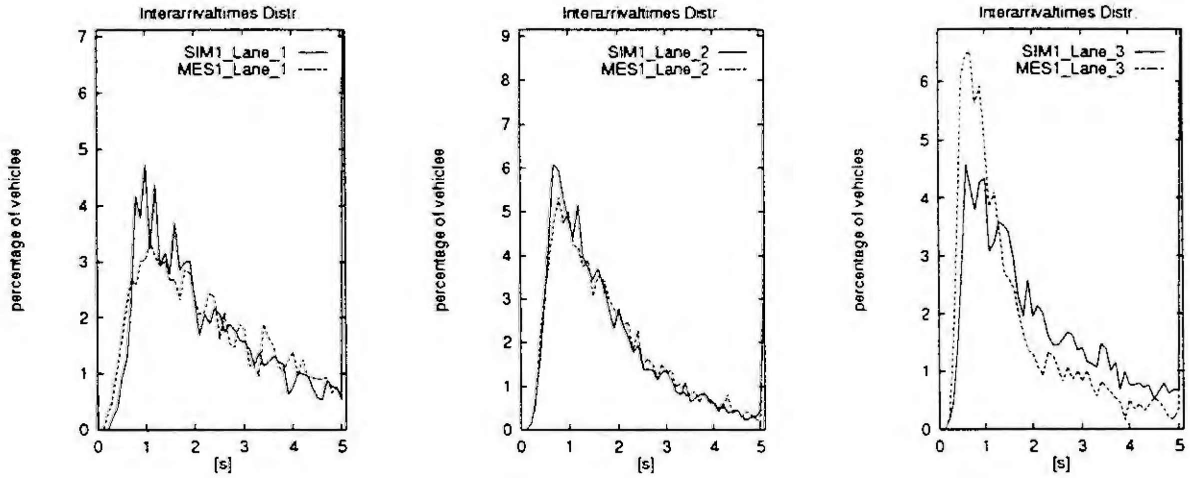


Figure 3: Inter-arrival times (MES1 vs. SIM1: right/middle/left lane)

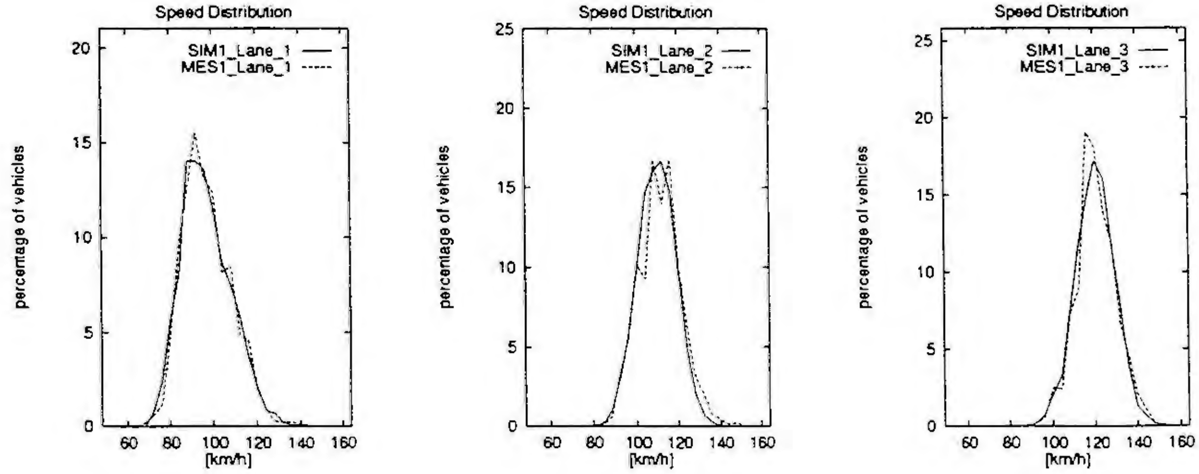


Figure 4: Speed distributions (MES1 vs. SIM1: right/middle/left lane)

2. Comparison of simulated data at the end with measured data at the end of the study section (MES2 vs. SIM2)
   The comparison shows, that some results differ considerably (up to 25 %). This can be explained as follows. The fact, that there is an exit following 400 m after the end of the study section, causes a change in the characteristics of the measured traffic. Therefore the intensity of the traffic (and the mean following distances) on all lanes differs from the data measured at the beginning of the study section. Since SIMCO3++ is currently designed to provide free traffic flow on a straight motor-way (see next paragraph), the influence of an approaching exit is currently not taken into account, but may be included in the set of mobility rules in the future.

3. Comparison of simulated data at the end with measured data at the beginning of the study section(MES1 vs. SIM2)
   In order to prove the ability of SIMCO3++ to provide a free traffic flow with constant statistical characteristics for several kilometres , the simulated data is compared with the data measured at the beginning of the study section (comparison between measured data at the end of the study section and SIMCO3++-data see previous paragraph). The data (see table 1 and figures 5 and 6) shows a good correspondence between simulation and measurements even after 2.9 km (maximal error below 10 %).

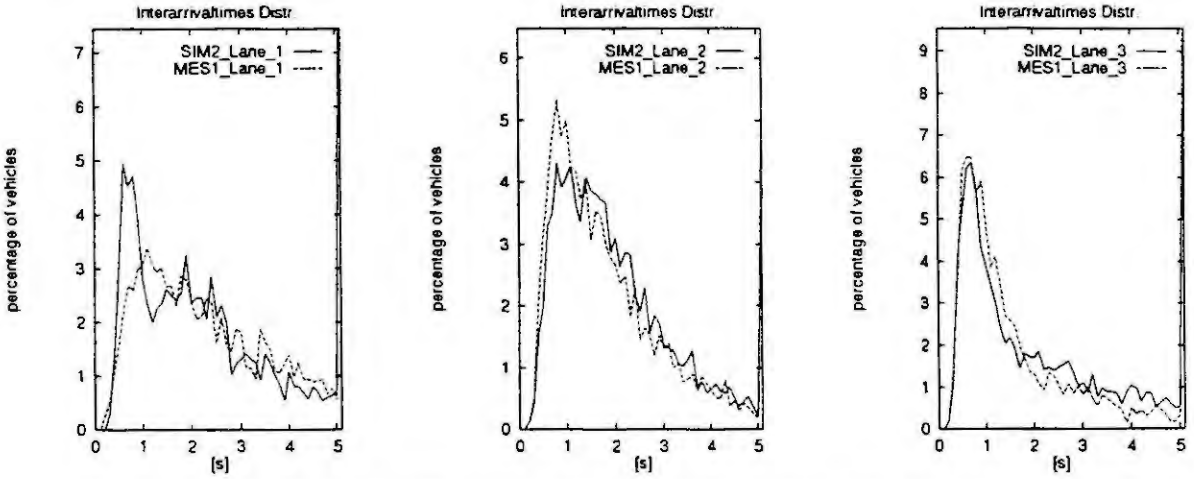


Figure 5: Interarrival times (MES1 vs. SIM2; right/middle/left lane)

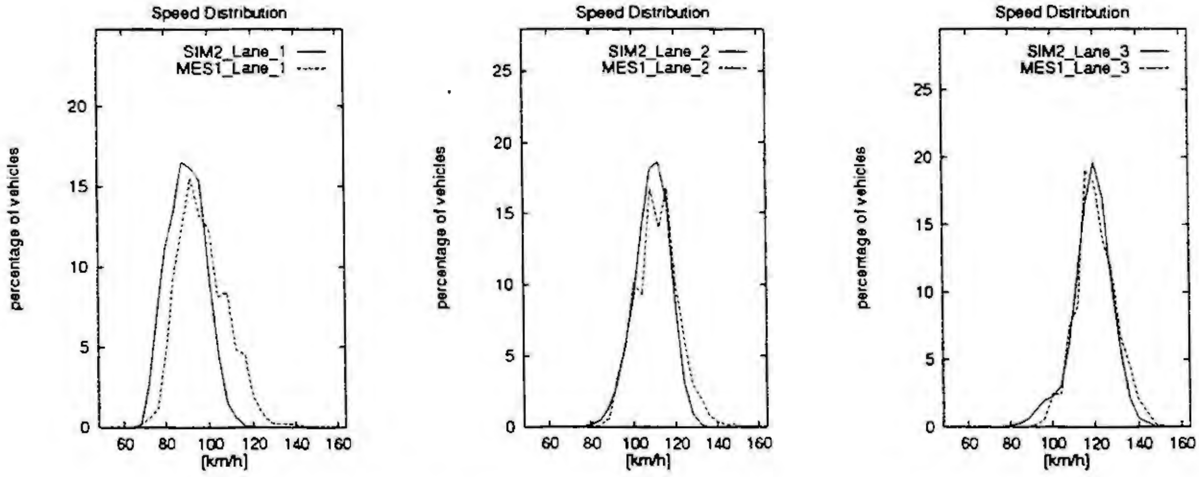


Figure 6: Speed distributions (MES1 vs. SIM2: right/middle/left lane)

7 Conclusions

In this paper, the functionality of the integrated simulation tool SIMCO3++ (SImulation of Mobile COmmunications), which has been designed for accurate analysis and performance evaluation of IRTE specific communication protocols (medium access control, logical link control, multi- hop routing strategies, etc.) for vehicle-beacon and inter-vehicle communications, based on realistic mobility models, road traffic scenarios has been presented.

A comparison of the motor-way measurements and the traffic scenarios simulated by SIMCO3++ shows a very good correspondence in important aspects like following distances between vehicles, average speed of vehicles, distribution of vehicle classes over the lanes especially at the generation point but also after a longer highway section. Therefore it can be concluded, that the simulator SIMCO3++ is a tool, which is very well suited for the performance evaluation of mobile communication protocols (short-range beacon-vehicle and inter-vehicle communications), which require the simulation of realistic road traffic mobility and scenarios.

Due to its sophisticated design, its modular concept, and its characteristic in combining very accurate communication protocol behaviour and channel characteristics with realistic mobility models for a variety of road traffic scenarios, SIMCO3++ provides not only valuable results for the evaluation, refinement and verification of the communication protocols, currently being developed by the communication projects of the DRIVE II programme (COMIS, GERDIEN, etc.), but also for the evaluation of proposed standards for beacon-vehicle communications (CEN TC 278; ISO TC 204), as well as for other mobile communication networks, like GSM (Public Land Mobile Network), UMTS (Universal Mobile Telephone System), MBS (Mobile Broadband System, RACE II Project), and any other large mobile communication network with a rapidly changing topology.

8 Acknowledgement

Bart van Arem, Horst Clausen, Henk Stoelhorst and Bernhard Walke have provided helpful discussions and important suggestions.

References

[1] M. van der Vlist B. van Arem. *Evaluation of the Simulation Program SIMCO2 for Dutch Motorway Conditions.* INRO-VVG 1992-15, TNO, June 1992. 74 pages.

[2] A. Kemeny. *MMI Solutions for Co-operative Driving.* In *Proc. of DRIVE Technical Days Advanced Transport Telematics*, Vol. 2, pp. 348–353, March 1993.

[3] C.-H. Rokitansky. *Performance analysis and simulation of vehicle-beacon protocols.* In *Proc. Vehicular Technology Conference*, pp. 1056–1057, Denver, Colorado, USA, IEEE, 1992.

[4] C.-H. Rokitansky. *Validation of the Mobility Model of SIMCO2 with Dutch Motorway Measurements.* In *Proc. Prometheus Workshop on Simulation*, pp. 209–215, December 1992.

[5] C.-H. Rokitansky. *Performance Evaluation of Medium Access Control Protocols Proposed for Standardization of Dedicated Short-Range (Beacon-Vehicle) Communications.* In *Mobile Kommunikation*, Vol. 124 of *ITG-Fachberichte*. B. Walke [Hrsg.], VDE-Verlag, September 1993.

[6] C.-H. Rokitansky, C. Wietfeld. *Comparison of Adaptive Medium Access Control Schemes for Beacon-Vehicle Communications.* In *Proc. Vehicle Navigation and Information Systems VNIS*, pp. 295–299, Ottawa, Canada, IEEE, October 1993.

[7] B. Walke, C.-H. Rokitansky. *Short-Range Mobil Radio Networks for Road Transport Informatics.* In *MRC'91*, pp. 183–192, Nice, France, November 1991.

Use of the Simulation Model AIRPORT MACHINE for investigating Problems of Air Traffic

The main topic of this paper will not be the design of a simulation model but the experience in using a special one : The AIRPORT MACHINE. This simulation model is obtainable by 'Aviation Simulation International, Inc., USA'.
The Airport Machine is a model for the simulation of an airport with all necessary features as runways, taxiways, positions, aprons and terminals.

1.0 Discussion of the Problem

The goal of our research is the increase of capacity and the safety on airports. Capacity is defined as the maximum number of aircraft in an hour for a special operation (e.g. arrival), which can be handled with a facility (e.g. runway). Many airports around the world have to cope with a shortage of capacity, and the possibilities to build new runways or terminals decrease from year to year. Because of this we have the problem that there is no possibility to make more than little improvements for overloaded airports like a change of the runway utilisation strategy or the building of a few new taxiways.
But before someone can guarantee a capacity increase or the feasibility of a new procedure it is necessary to test the new strategies and this leads us to the main problem:

Where should new improvements be tested ?

Testing on a real airport or introducing of new technology without a simulation is not only very expensive because of the training for the affected controllers and the installation of the necessary technologies or buildings, but also dangerous for the aircraft. Another point is that the controllers become overloaded with the control of normal and new procedures. Finally, such a test would disturb the normal traffic and this could have major impact on the average delay per aircraft.
The solution for this problem is the usage of a well designed and validated simulation model. This model should be able

- to simulate the most important features of an airport such as the complete airport structure and main air traffic control and management procedures and
- to give detailed output reports which should include characteristic airport parameters like capacity and delay.

2.0 Input Data

2.1 Overview

The main work for getting an appropriate baseline simulation is the collection of input data (figure 1). We have 5 different sorts of input data: airport structure, operational parameters, operational procedures, airspace structure and flight-schedules.
Information e.g. about the airport structure as the position of runways can be found on maps or could be obtained from the affected airport agency. Especially the operational data and procedures (e.g. taxi speeds, aircraft separation, use of positions) can be obtained from traffic observations at the considered airport. A very important part of the input data is the flight-schedule. This plan influences the whole simulation run including the results.

2.2 Examples

Aircraft Separation:
Every flying aircraft is trailing a whirl at the end of the wings, called 'wake vortex'. These wake vortices cause great problems in the landing phase of flights. If we have a sequence of two or more landing aircraft a small trailing aircraft may be strongly influenced by the high energy wake vortices of the bigger leading aircraft. This forces the controller to increase the separation between special types of aircraft.
The problem is now to get average values for the separations which depend on the combination of arriving aircraft.
A possibility to get these data is to record radar data of arriving aircraft about a long period of time (perhaps a month) and to analyse it. Because the simulation model needs data for **minimum** separation only periods of high traffic demand should be analysed and false data (e.g. caused by wrong aircraft-id's) should be eliminated.

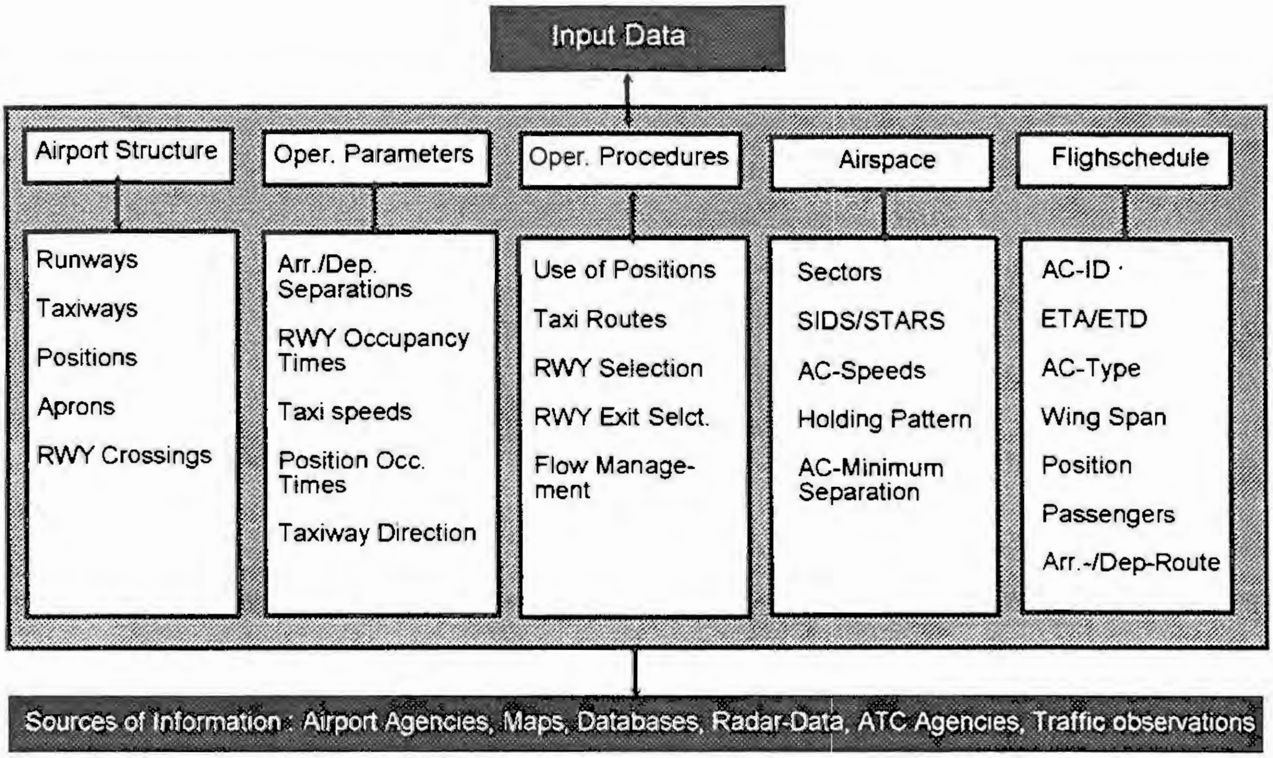


Figure 1: Input Data

Flight-schedule:
As mentioned before the flight-schedule is one of the most important data of the whole simulation and should be selected carefully. Figure 2 shows the structure of the arrival flow over the day for a complete month. The thick curve shows the average over a month for every hour of the day.
For the baseline simulation of an airport it is necessary to choose on the one hand a day which has the same characteristic peaks and valleys as the average day and which has on the other hand a high traffic demand for the simulation of overload situations. To get such a day we have valuated all days in the following manner:

> To every hour *max(0 , 3- (difference in aircraft for this hour relative to the hour value of the average day))* was assigned as hour-value and for the day we have summed up these values. After this we have choosen the day with the highest value and the highest traffic flow.

For the simulation of a future scenario it is necessary to create a flight schedule with the future traffic flow including the distribution of aircraft types, source and destination airports, positions at the airport and flight routes.

3.0 Interaction Mode

In comparison to other airport simulation models the AIRPORT MACHINE has the advantage of an interaction mode. This means that the user can intervene in a normal simulation run and can e.g.

- change the arrival/departure runway,
- change the preassigned parking position,
- change the taxi route,
- choose the next aircraft for crossing a taxiway intersection,
- choose the next departure,
- stop taxiing aircraft and
- stop the push back (leaving of position) of aircraft.

Within this interaction mode it is possible to test new procedures together with controllers directly in a simulation run. The controllers can handle aircraft in the simulation like they would do with real aircraft.

4.0 Simulation Results

The output of the airport machine includes the following results: Runway-, air-, taxi-, crossing- and gate-delay, taxi-distances and taxi-times.

All results are given for the implemented airlines and the aircraft types (e.g. wake vortex classes). Additionally the taxi-delay is given in form of delay per node of the airport link system and the runway-delay for every hour of the day and per runway. The most important results for the estimation of runway capacity are runway- and air-delay. The performance of the taxiway system is characterized by taxi-delay and -times.

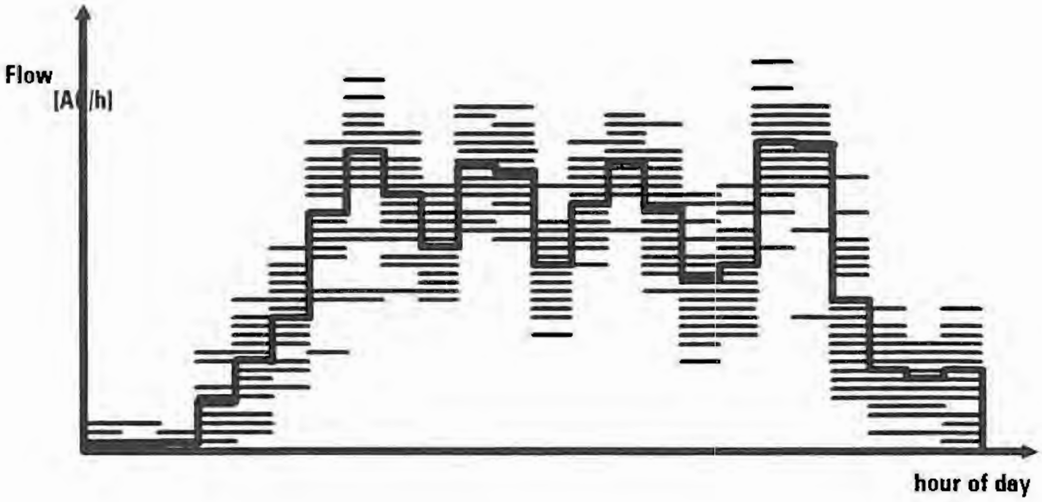
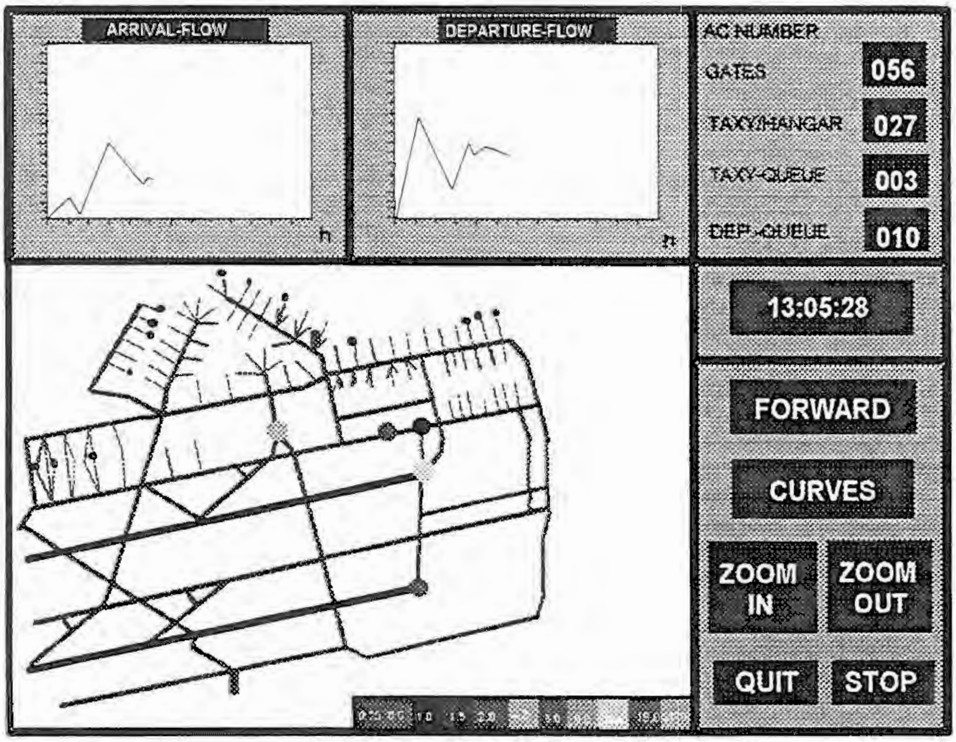


Figure 2: Daily flow of a month together with average flow.



AC NUMBER

GATES 056

TAXY/HANGAR 027

TAXY-QUEUE 003

DEP.-QUEUE 010

13:05:28

Figure 3: Analyzing tool for the AIRPORT MACHINE

5.0 Analysing Tool

Figure 3 shows our additional analysing tool for the AIRPORT MACHINE. With the AIRPORT MACHINE we have had the problem that the graphic interface shows only the activity of aircraft and not directly the values for flow and delay. Therefore we have implemented a program which is able to show the delay in form of colored circles at the current aircraft positions and curves for arrival-/departure- and total-flow, taxi-in-, taxi-out-, air- and runway-delay and the percentage of occupied positions. The user has the possibility to select two curves at the same time. The color of the delay-circles depends on the length of the delay. E.g. yellow stands for a delay between 8 and 15 minutes. In the upper right edge are shown values for the length of runway- and taxi-queues, the number of occupied positions and the number of moving aircraft.

6.0 Example for an Investigation

In the past we have made several investigations with the AIRPORT MACHINE for former BFS (Federal Administration of Air Navigation services) and FAG (Frankfurt Airport Agency). One of these investigations was to simulate the introduction of a new departure controller in Frankfurt. This additional controller should concentrate on the creation of an optimal sequence of aircraft according to the type-dependent separation at the ends of the departure runways.

The test was conducted with 3 different teams, each consisting of an airspace-, an airport-controller and an operator of the DLR (figure 4). If a situation has occured where an interaction was needed, the operator has called a code-word and the affected controller has told back how the traffic should be handled. The special actions for each controller and the codewords of the operator can be found in figure 4.

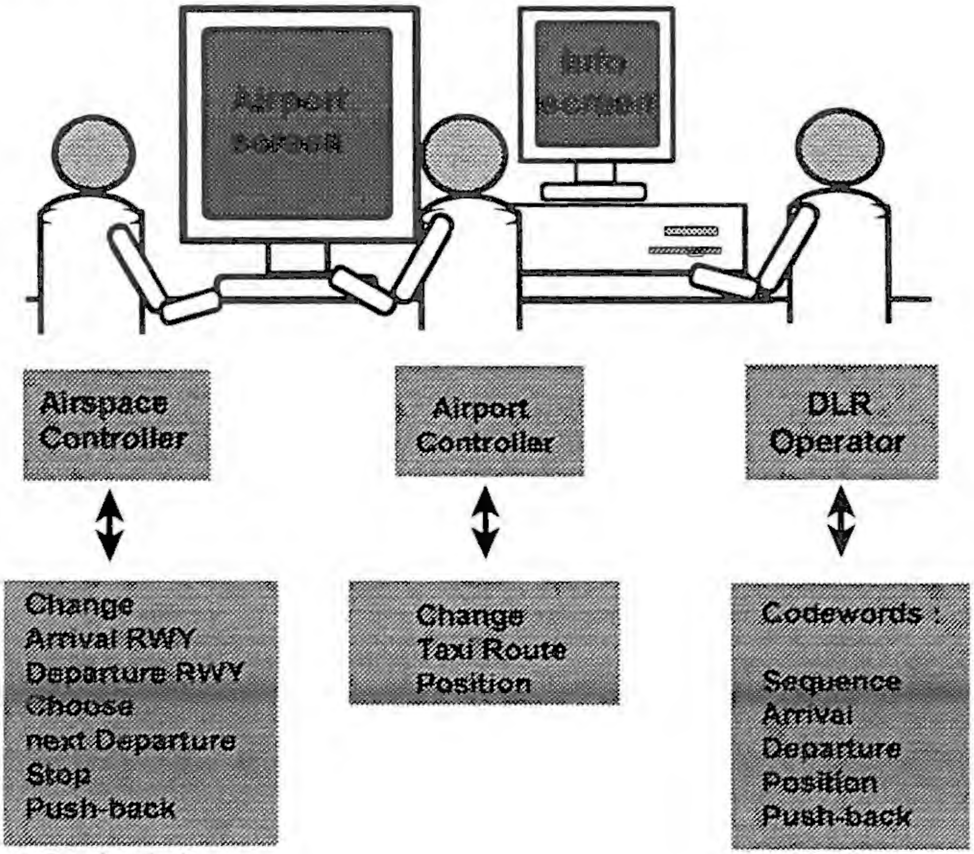


Figure 4: Example for an investigation.

7.0 References

[1] Aviation Simulation International, The Airport Machine User's Manual, Huntington, N.Y., 1991

[2] Gerdes, I., Auswirkungen von unterschiedlichen Sequencing-Strategien auf die Kapazität von Flughäfen. DLR-Mitteilung 90-18, Köln 1990.

[3] Gerdes, I.,Kalibrierung von Simulationsmodellen. To appear in: Wege zur Steigerung der Flughafenkapazität, (1993).

[4] Knabe, F., Gerdes, I., A Look in the Future of Germany's Biggest Hub in Fast Time Simulation. In: Proceedings Airshow Canada Symposium 'Sharing the Skies', Vancouver/Canada, 1991.

MAKSIMOS: MACROSCOPIC SIMULATION AND OPTIMIZATION OF SIGNALIZED URBAN ROAD NETWORKS

Karin Putensen
Technical University Hamburg-Harburg
Lohbrügger Kirchstraße 6S D-21033 Hamburg Germany

Abstract. Dynamic simulation of traffic flow in urban areas may help to solve traffic engineering problems in design and operation of traffic control systems. Within this paper a macroscopic simulation model of urban traffic flow and some of its applications in traffic control are described.

1. INTRODUCTION

Traffic flow in urban areas is strongly influenced by geometric limitations of roads, by the road network topology and by the signalization of intersections. Recent claims for more safety and priority to public transport increase traffic capacity problems in the cities. Dynamic simulation of traffic behaviour becomes more and more relevant and may help to cope with these problems in the design and operation of traffic regulations. In this paper a ***macroscopic model for dynamic traffic flow*** in urban road networks are presented and the ***some applications like optimization of signal control*** using this model are discussed.

2. SIMULATION MODEL FOR URBAN TRAFFIC FLOW DYNAMICS

In this section a macroscopic model for the simulation of urban traffic flow (MAKSIMOS) is presented. The model approach has been derived from hydrodynamics which have been applied successfully to motorway traffic flows (Payne, 1971). The detailed course of individual vehicles is neglected in favour of the advantage of macroscopic approaches, i.e. fast simulation of large networks.

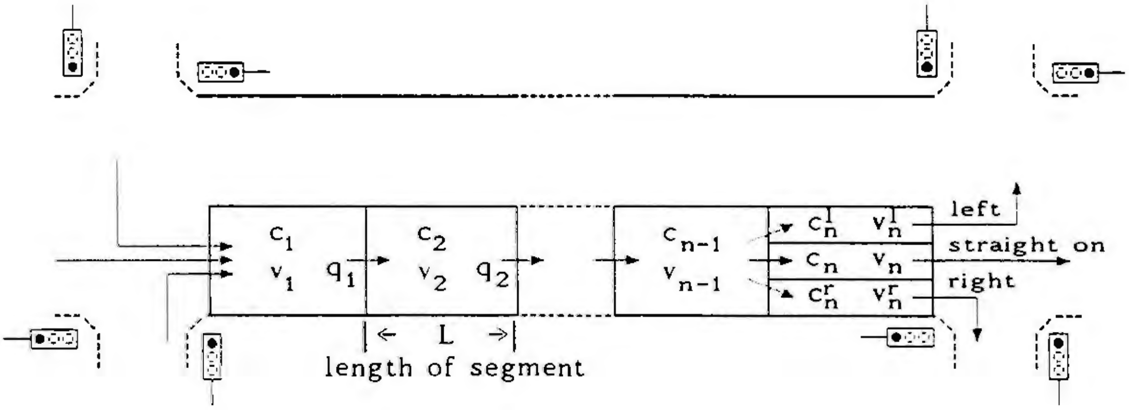


Figure 1: Road section with flow variables

2.1. Model Description

The model is formulated in time and space discrete variables which describe traffic flow as a nonstationary, dynamic process. The urban road network is divided into segments of 40 - 100 m length. The model describes traffic flow dynamics by the aggregate variables **density** per segment, **average speed** per segment and **volumes** at the segment borders (cf. figure1).

Traffic state transition is calculated for time intervals of three seconds based on a set of nonlinear, time discrete difference equations which describe traffic phenomena in the segmentated network in the range from free traffic flow up to saturation and congestion.

Macroscopic Traffic Variables:

$\mathbf{c_i(k)}$ traffic density in segment i for time $k \cdot T$ [veh/km]

$\mathbf{q_i(k)}$ traffic volume leaving segment i during $k \cdot T \le t < (k+1) \cdot T$ [veh/h]

$\mathbf{v_i(k)}$ mean speed of the vehicles in segment i for time $k \cdot T$ [km/h]

Model Equations:

density: $$c_i(k+1) = c_i(k) + \frac{T}{L}\left(q_{i-1}(k) - q_i(k)\right)$$

volume: $$q_i(k) = c_i(k)\, v_i(k)$$

mean speed: $$v_i(k+1) = \beta\, v_i(k) + (1-\beta)\, \mathbf{V}(\bar{c}(k))$$

with $$\bar{c}(k) = \alpha\, c_i(k) + (1-\alpha)\, c_{i+1}(k)$$

with T time interval (3 sec.)

L segment length with $\frac{L}{T} > v_{max}$

$\mathbf{V}(\cdot)$ Speed-Density Characteristic

α, β constant weighting factors

The density ist determined by a balance of vehicles entering and leaving a segment during the time interval $[k \cdot T, (k+1) \cdot T[$. Under stationary conditions, it is assumed that the mean speed $v_i(k)$ depends on the density $c_i(k)$ within segment i and additionally on the density $c_{i+1}(k)$ in the following downstream segment which is foreseen by the driver. This influence on the mean speed is reflected by the speed-density characteristic (see figure 2). For the dynamic behaviour of the mean speed $v_i(k)$ it is assumed that adaptation to the stationary value of the speed-density characteristic **V** takes place according to a first order difference equation.

The **specific characteristics** of urban traffic flow like - flow interruption by signalized and not signalized intersections - and - turning flows with and without separate lanes - can be taken into account based on the described model structure. To model turning flows the segment before the intersection is divided into 3 subsegments for left turning, right turning and straight on traffic. The intersection itself is divided into overlapping segments for each access road (see figure 1).

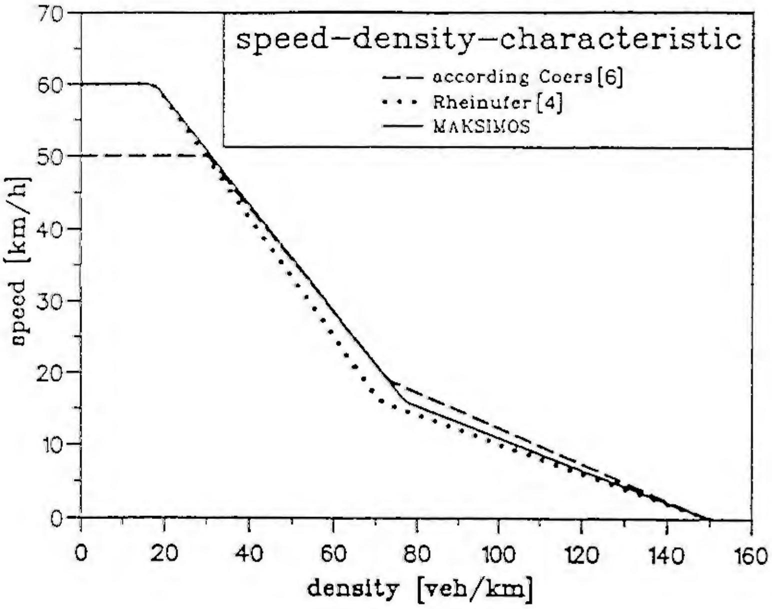


Figure 2: Stationary Speed-Density Characteristic

2.2. Evaluation of Traffic Flow Performance

The simulated traffic flow can be evaluted by four different **performance indices** - travel time, delay time, mean speed, weighted mileage - which have been proved to be suitable measures for the assessment of traffic quality. These criteria can be derived from the state variables by the following formulas (t = number of time intervals, s = numbr of segments):

(1) overall travel time [veh · s]: $$\sum_{k=1}^{t} \sum_{i=1}^{s} c_i(k) \cdot L \cdot T \rightarrow \min!$$

(2) overall delay time [veh · s]: $$\sum_{k=1}^{t} \sum_{i=1}^{s} \left(1 - \frac{v_i(k)}{v\,max_i}\right) \cdot T \cdot L \cdot c_i(k) \rightarrow \min!$$

(3) overall mean speed [m/s]: $$\frac{\sum_{k=1}^{t} \sum_{i=1}^{s} c_i(k) \cdot L \cdot v_i(k)}{\sum_{k=1}^{z} \sum_{i=1}^{s} c_i(k) \cdot L} \rightarrow \max!$$

(4) weighted mileage [veh · m²/s]: $$\sum_{k=1}^{t} \sum_{i=1}^{s} \left(L \cdot c_i(k) \cdot T \cdot v_i(k)\right) v_i(k) \rightarrow \max!$$

2.3. Calibration of Model Parameters

The parameters of the model have been adjusted to measurements from test sites in Hamburg, Germany in 1987 and 1992. The total length of the test site was 450 m with 8 observation points. Measurements have been taken at short distances before and behind a signalized intersection in 1 sec.-intervals, because the model has to represent traffic flow during acceleration and deceleration with the same parameters.

Figure 3 shows for two different locations a comparison of time diagrams of measured and simulated volumes (aggregated to 3 sec-intervals). The results demonstrate that the model reproduces traffic flow dynamics quite well.

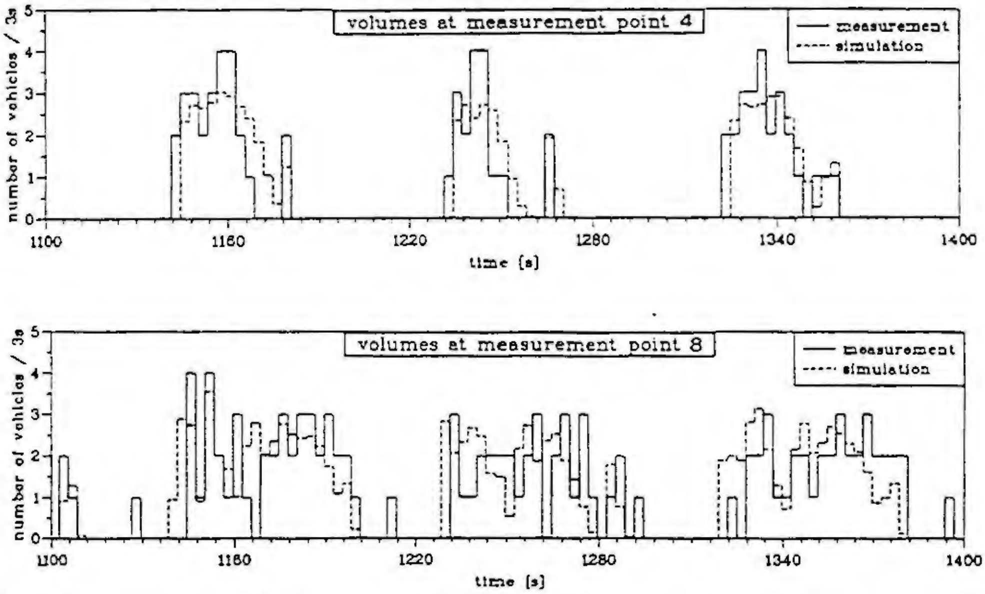


Figure 3: Comparison of measured and simulated volumes at measurement point 4 (stop line) and 8 (end of section)

3. APPLICATION TO TRAFFIC CONTROL PROBLEMS

3.1. Coordination of Signal Control by Optimization

The described model has been used for an ***optimization of the signalization*** in urban road networks. An heuristic optimization algorithm has been developed for the determination of optimal control parameters, like begin and duration of green times, with respect to the chosen performance index. To reduce the number of parameters the signal program of an intersection is determined by only two parameters - begin and duration - of green period of one access road. Each couple of opposite entrances have the same signal times. The signal times of the neighbouring entrances could be calculated using these two parameters. Therefore the total number of optimization parameters is 2 * (number of signalized intersections within the network).

The optimization has to determine those parameters values which minimize (or maximize) the chosen criterion, e.g. delay time for a given (possibly dynamic) demand scenario. This sets up an integer nonlinear optimization problem which can only be solved conveniently by heuristic algorithms. The algorithm developed here is a step-by-step search-algorithm based on steepest descend, which takes into account the characteristics of this special problem.

For example , a simple topology with 3 signalized intersection with a distance of 300 m and 400 m has been chosen (see figure 4). The signal program of intersection 3 and all green times were assumed to be fixed. The coordination of the other two intersections has to be optimized relative to intersection 3 using the performance index delay time. Figure 5 shows the 3-dimensional presentation of the delay time as a two-dimensional function of the begin of red time of intersection 1 and 2. It is interesting to note that already, in this simple example different local optima of this function occur. The path of a particular optimization procedure which leads to the global optimum (52,52) is indicated by a bold line with arrows.

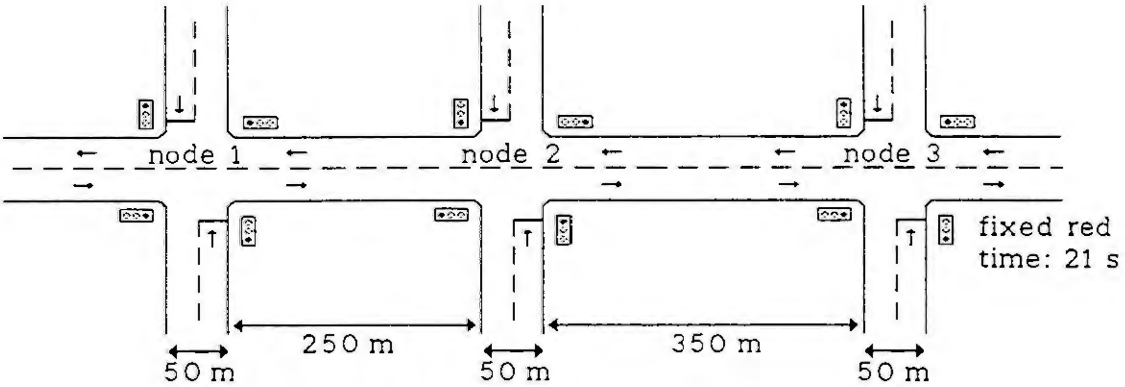


Figure 4: Topology with 3 signalized intersections

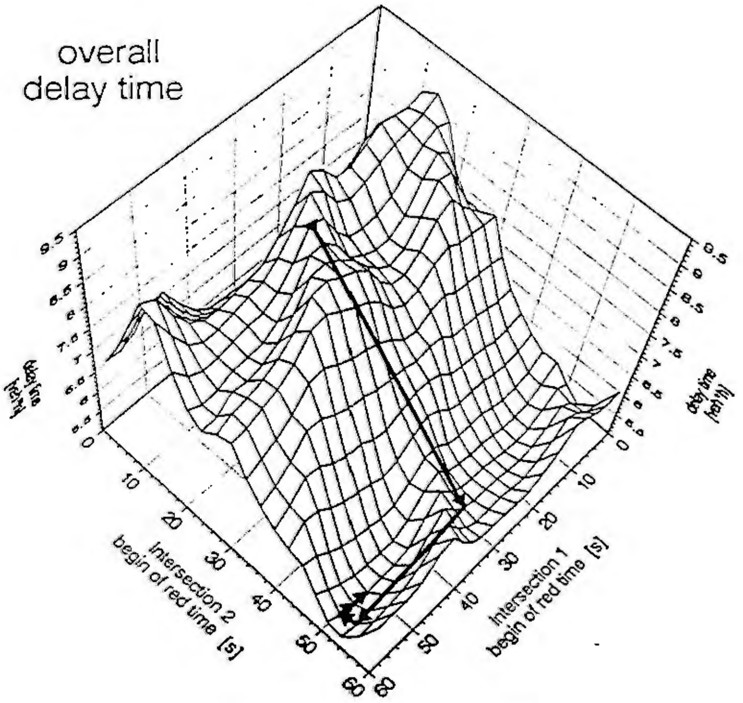


Figure 5: Performance index - delay time - for different signal settings

3.2. Tidal Flow Systems and Traffic Diversion

Dynamic simulation of traffic flow in urban areas may help to solve traffic engineering problems in different areas. For the design and development of traffic systems, simulation can compare different alternatives. Furthermore, the development of control strategies and algorithms can be simplified and improved by simulations. During operation of traffic systems, simulation may help to predict and control the traffic evolution with model based control schemes.

This model has been especially applied to a ***Tidal Flow System*** in Barcelona. Tidal Flow Systems are roads with reversable lanes, which are allocated to the direction with higher demand. These systems are nowadays mainly based on fixed time table. The described simulation model has been applied to the Barcelona system to develop and investigate traffic responsive lane switching strategies based on actual measurements (see DRIVE V1020, 1991). Another example for an efficient application of this model is the ***development of diversion strategies*** for a congested road via alternative routes. The investigations for this application have shown that the traffic diversion has additionally to take into account and to modify the signalization along the alternative routes.

4. CONCLUSIONS

In this paper a macroscopic model for traffic flow in urban road networks has been presented. The model offers to the user high flexibility when specifying a particular topology, allowing the inclusion of signalized and nonsignalized intersections, single-lane and multilane roads, etc.. First verifications of the model show that it reproduces real traffic phenomena quite well. Though the presented application used the model in an off-line mode, the model might be even more valuable for online applications due to its macroscopic nature, e.g. supporting the operator by quick predictions of alternative control decisions. Possible fields of further applications are: model based traffic surveillance and control (e.g. Cremer, Henninger 1993), traffic responsive rerouting, traffic management systems including public transport.

5. REFERENCES

[1] Cremer, M. and Fleischmann, S., Entwicklung eines regelungstechnischen Konzepts zum verkehrsabhängigen Einsatz von Wechselverkehrszeichen und Zufahrtsdosierungen in Schnellstraßennetzen - Forschung Straßenbau und Straßenverkehrstechnik, Heft no. 505, 1987.

[2] Cremer,M. and Henninger,T., Estimation of Queue Length in Urban Road Networks, Proceedings of Pacific Rim TransTech Conference, Seattle USA, July 1993.

[3] DRIVE V1020 Tidal Flow Systems (Commision of the European Communities), Deliverable no.4: Final Report on Simulations, 1991.

[4] Hoffmann, G. , Hoffmann, G. , Leichter, K., Steuerungs- und Bewertungsgrößen, in Verkehrsleittechnik für den Straßenverkehr, Band I: Grundlagen und Technologien der Verkehrsleittechnik, Hrsg.: Lapierre, R. and Steierwald, G. , Springer-Verlag, 1987.

[5] Payne, H.J., Models of freeway traffic control - Simulation Council Proc. 1, 1971.

[6] Stegemann, G.H., Simulation und Bewertung von Verkehr in signalgesteuerten Stadtstraßennetzen - Dissertation RWTH Aachen, 1979.

Nonlinear Models in Population Economics

Feichtinger/Hof/Luptacik/Lutz/Prskawetz/Wirl

Institute for Operations Research
Technical University of Vienna

Abstract. We consider a three sector demoeconomic model and its interdependence with the accumulation of human capital and resources. The primary sector harvests a renewable resource which constitutes the input into industrial production, the secondary sector of our economy. Both sectors are always affected by the stock of knowledge. The tertiary sector (schooling, training) is responsible for the accumulation of this stock that represents a public good for all three sectors. The central focus of this study is to demonstrate the use of nonlinear dynamical systems theory in modelling the interaction of economic processes and population dynamics.

1. INTRODUCTION

We introduce a model that attempts to investigate the interdependence of economic and demographic development. (see e.g. [1], [2], [18], [3], [8]) The employed framework combines Malthusian constraints and economic growth theory. Though the wealth of the industrial countries may distract attention from Malthusian forces, nevertheless they are quite visible in many developing countries. A recent warning in this direction is the book by the well known political scientist Kennedy [7]. The second foundation of the model applies insights from the 'new' theory on economic growth (see e.g. [9], [11],[12]), where the state of human knowledge is crucial in order to fuel development and growth.

The model is fairly general and may in principle cover as limiting case a society of hunters and gatherers (see [10]) that lacks collective action and highly civilized and organized societies observed in industrialized countries. However, and despite this attempt of generality one has to take a humble view of such a modelling exercise. That is, it remains a 'toy model' in the words of Romer [13], and it is necessarily so, because the growth of human knowledge by definition escapes the description by simple formulas.

Our objective is to investigate mechanics underlying the joint development of the population and the economy (at the sectoral level). These stylized patterns of development are contrasted with empirical evidence compiled to support or refute our model.

2. THE MODEL

We consider an entirely competitive economy, where the labour force L is divided and migrates between three different kinds of employment:

1. the primary sector (L_1), which harvests natural renewable resources (agriculture, mining industry, etc.)
2. the secondary or industrial sector (L_2) and
3. the tertiary sector (research and education) (L_3)

The output H of the primary sector is given by a standard production function in the inputs labour L_1 and the available resource stock R. In addition, technology A improves the labour productivity (e.g. by using tractors instead of horses).

$$H(AL_1,R) = R^{\alpha_1}(A^{\varepsilon_1}L_1)^{\alpha_2}, \qquad 0<\alpha_1,\alpha_2,\varepsilon_1<1,\ \alpha_1+\alpha_2=1 \tag{1}$$

The net growth of the renewable resource stock R is affected by two counteracting factors, indigenous, biological growth $g(R)$ and the harvest H.

$$\dot{R} = dR / dt = g(R) - H(AL_1, R) \qquad \text{(I)}$$

The growth function g is assumed to be of the logistic type illustrated in Clark [5].

$$g(R) = \theta R(\overline{R} - R), \qquad g(R) > 0, g'' < 0 \quad for \quad 0 < R < \overline{R} \qquad (2)$$

The coefficient $\overline{R}$ determines the saturation level of the resource stock and the parameter θ determines the speed at which the resource regnerates.

Using the harvest H and labour L_2, the secondary sector produces the output Y. Again, technoloy A affects the labour productivity and may also increase the production frontier over time.

$$Y(A,H,L_2) = A^{\varepsilon_2} H^{\beta_1} (A^{\varepsilon_3} L_2)^{\beta_2}, \qquad 0 < \beta_1, \beta_2, \varepsilon_2, \varepsilon_3 < 1, \beta_1 + \beta_2 = 1 \ . \qquad (3)$$

Finally, the output of the tertiary sector E adds new technologies and ideas to the stock of knowledge A.

$$E(Z, L_3, P, A) = \left[(Z / P)^{\gamma_1} (L_3 / P)^{\gamma_2} A^{\gamma_3}\right] P^{\gamma_4} \quad 0 < \gamma_1, \gamma_2, \gamma_3, \gamma_4 < 1 \qquad (4)$$

The output E depends on two kinds of inputs, industrial products Z and of course, labour L_3. As usual, this production function is itself affected by progress A. In addition, the output E may depend on the population or some measure of density P.

Human knowledge, as already mentioned, is a stock variable that also depreciates. The net growth rate of knowledge can be described by the differential equation

$$\dot{A} = dA / dt = E(Z, L_3, P, A) - \delta A \qquad \text{(II)}$$

where δ stands for the depreciation factor.

The total ratio of the population P entering the labour force, the labour participation ratio l , depends positively on the real wage W net of lump sum taxes τ raised by the government on wage income.

$$L = l(w - \tau)P = \left[(w - \tau) / (w_0 + w - \tau)\right]P \qquad (5)$$

Endogenous population growth n follows a Malthusian law (see Woods [16]). As long as output per capita $y = Y / P$ is below some exogenous given subsistence level y^{sub}, the population will decline, while a positive population growth rate will set in when output per capita exceeds y^{sub}. For analytic and numerical convenience we use the function

$$n = b \log(y / y^{sub}) \qquad (6)$$

where b represents a constant scaling factor.

The evolution of the population is described by

$$\dot{P} = n(y, y^{sub})P \qquad \text{(III)}$$

The economy set out exhibits two externalities that are not properly reflected in competitive markets. Firstly, free access to renewable resource harvesting may lead to the tragedy of the commons. Secondly, the existence of a public stock 'knowlege' which shifts the production frontiers outward in all three considered sectors so that increasing returns to scale apply for the entire economy.

The tragedy of the commons associated with resource exploitation may require restricted entry into the primary sector to sustain the resource basis. This restriction might be achieved by taxing the output of the primary sector. This tax T raises the factor price p of the primary inputs for the secondary sector. Hence, the optimal choice of inputs for the primary sector follows from equating the price p to marginal costs MC plus taxes T.

$$p = MC + T = w / H_{L_1} + T \qquad (7)$$

Marginal costs are determined by the wage w paid for one additional worker divided by his (marginal) product, H_{L_1} (Varian [15]). (Subscripts denote partial derivatives of the function with respect to the corresponding argument.) Now, the cost minimizing input allocation of the industry follows from the condition that the ratio of the marginal products must equal the factor prices.

$$F_{L_2} / F_H = w / p \tag{8}$$

Employing one more worker, the secondary sector has to pay the common market wage w, while the price p has to be paid for increasing the factor input H.

Additionaly the assumption of constant returns to scale production functions leads to equating marginal costs MC and average costs AC in each sector. This yields a further constraint e.g. in the secondary sector.

$$AC = (pH + wL_2) / Y = \pi \tag{9}$$

where π is the price of the industry good, which is equal to marginal costs under competitive allocations.

Secondly, as already mentioned, the stock of knowledge A must be properly managed. Failures to organize collective actions properly will hinder economic growth. Total tax revenues $TR = TH + \tau L$ given by the sum of taxes levied on the harvest TH plus taxex levied on the workers τL are used to finance the inputs L_3 and Z into the production of E. Admitting the possibility of corruption c (governmental institutions may divert part of the total tax revenue into their own pocket) only a part of the collected taxes $(1-c)(TH + \tau L)$ is used for financing the tertiary sector. The distribution of this amount is assumed to follow a constant pattern determined by a proportion $(1\text{-}s)$ spent on the factor costs of the input labour wL_3 and a proportion s spent on the industrial products πZ:

$$(1-c)(TH + \tau L)s = \pi Z \tag{10}$$

$$(1-c)(TH + \tau L)(1-s) = wL_3 \tag{11}$$

Summarizing our model can be described by a three dimensional system of differential equations (I)-(III) and six implicit constraints (5), (7)-(11).

3. EMPIRICAL EVIDENCE: A CROSS SECTION STUDY

In the following a short empirical assessment of the functional relationships outlined in section 2 is given (see also Hazledine [6]). We use a cross-section of 23 low-income countries (per capita GNP ≤ US $400 in 1989) in Africa. Data are taken from three sources: Britannica [4], World Resources [18] and the Weltbevölkerungsbericht [14].

Firstly we estimate the production functions (1), (3), (4) by ordinary least squares. The results are shown in table 1 (t-values are given in the brackets).

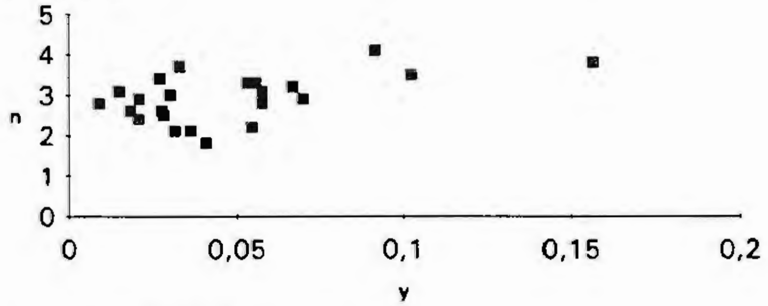
TABLE 1: Estimated production functions

ln(H)		ln(Y)		ln(E)	
constant	5.998 (35.42)	constant	1.163 (0.65)	constant	4.375 (9.24)
$\ln(L_1)$	0.587 (2.91)	$\ln(L_2)$	0.199 (1.12)	ln(Z/P)	0.570 (7.72)
ln(R)	0.356 (1.63)	ln(H)	0.695 (3.09)	$\ln(L_3/P)$	0.298 (2.15)
				ln(P)	0.979 (14.70)
R^2	0.741	R^2	0.719	R^2	0.933

As a measure of the labour force we use the percentage of population at working age (15-65 years). The percentage of population working in agriculture, industry and services yields the distribution of the labour force to each sector. The output H, Y and E of each sector is taken as the percentage of GDP out of the primary, secondary and tertiary sector. Resources are equated with arable land. The percentage of GNP spent on education yields an estimate for Z. As there was no reliable estimate available for A, we excluded it from the list of independent variables. The estimated coefficients are the marginal productivities of the corresponding factor inputs and contain information about the overall returns to scale. All coefficients are positive and their sum suggests that each of the less developed regions suffers from decreasing returns to scale in the primary and secondary sector while the tertiary sector exhibits increasing returns to scale. The independent variables used in our model are likely to be correlated, such that the estimates might be biased upwards.

Furthermore Figure 1. illustrates the positive correlation between population growth n and per capita output y of the secondary sector. The Malthusian law seems not to be refuted by the data.

FIGURE1:



4. BALANCED GROWTH PATHS

The complexity of the model - a three dimensional system of nonlinear differential equations and six implicit constraints - complicates numerical investigations of our system. Alternativley one can study the systems behaviour along *balanced growth paths*, which are characterized by constant growth rates of the state variables. Using this framework several questions can be raised and studied:

- What are the forces to escape Malthusian traps and sustain economic growth?
- What are the conditions of a take off in societies with low industrialization at the outset?
- What are potential contributions to efficiently use tax revenues?
- Does there exist any optimal population growth or technological progress?

These numerical simulations and thus necessarily stylized patterns of development can be contrasted with empirical evidence compiled to support or to refute our results.

REFERENCES

[1] Blanchet, Modélisation Démo-Économique: Conséquences Économiques des Évolutions Démographiques, Institut National d´Études Démographiques, cahier n^0 130, INED, PUF, Paris, 1991.
[2] Bonneuil, N., 1992, Malthus, Boserup and population viability, Working paper.
[3] Boserup, E., Population and Technological Change. The University of Chicago Press, Chicago, 1981.
[4] Britannica, 1992.
[5] Clark, C.W, Bioeconomic Modelling and Fisheries Management. John Wiley, New York, 1985.
[6] Hazledine, T. and R. S. Moreland, Population and economic growth: a world cross-section study. The Review of Economics and Statistics 3 (1977), 253-263.
[7] Kennedy, P., Preparing for the Twenty-First Century, Random House, 1992.
[8] Lee, R.D., Malthus and Boserup: a dynamic synthesis. In: Coleman, D. and R.S. Schofield (Eds.), The State of Population Theory Forward from Malthus. Basil Blackwell, Oxford, 1986, 96-129.
[9] Lucas, R.E. Jr., On the mechanics of economic development. Journal of Monetary Economics 22, (1988) 3-42.
[10] Prskawetz, A., G. Feichtinger and F. Wirl, Endogenous population growth and the exploitation of renewable resources. To appear in Mathematical Population Studies, 1994.
[11] Romer, P.M., Increasing returns and long-run growth. Journal of Political Economy 94, (1986) 1002-10037.
[12] Romer, P.M., Endogenous technical change. Journal of Political Economy 98 (1990), 71-103.
[13] Romer, P.M., Tow strategies for economic development: using ideas and producing ideas. In: Proceedings of the World Bank Annual Conference on Development Economics 1992.
[14] UNFPA, Weltbevölkerungsbericht 1993, Deutsche Gesellschaft für die Vereinten Nationen, 1993.
[15] Varian, H.R., Microeconomic Analysis, W.W. Norton & Company, New York, 1984.
[16] Woods, R., On the long term relationship between fertility and the standard of living. Genus 39 (1983), 21-36.
[17] World Bank, Population Change and Economic Development. Oxford University Press, New York, 1984.
[18] World Resources Institute, World Resources 1990-1991. Oxford University Press, New York, 1990.
[19] Zhang, W.B., Economic growth and technological change. International Journal of Systems Science 21 (1990), 1933-1949.

Iterative Design of Economic Models via Simulation, Optimization and Modeling

M. H. Breitner, B. Koslik, O. von Stryk, H. J. Pesch

Technische Universität München, Mathematisches Institut,
D–80290 München, Germany.
E-mail: breitner@mathematik.tu-muenchen.de

Abstract

Microeconomic models, e.g., concern models, usually suffer from too simple dynamic equations and from too unrealistic economic data. We investigate a new iterative design method, including numerical simulation, numerical solution of optimal control problems by direct optimization methods and modeling. The capability of the method is demonstrated by the refinement of a quite simple first concern model. In the end, very complex micro- and macroecononomic phenomenona known from reality, have been obtained for various numerical calculations.

Introduction and first concern model

Mathematical models of microeconomic processes are very important for many purposes. These models can, e.g., help to explain macroeconomic phenomenona or help to improve the mangement of a concern. Especially concern models are well known in literature, see, e.g., Feichtinger and Hartl (1986), Hilten et al. (1993), Kamien and Schwartz (1981), Kort (1989), Kort et al. (1991) and Lesourne and Leban (1978). The proposed concern models generally can be (and should be!) improved by an additional refinement. Usual insufficiencies are:

- Quite simple dynamic equations to enable an analytic calculation of the optimal open-loop or feedback controls or/and the optimal limit cycles.
- Inexact economic data and inexact data of the concern due to the difficulties to get and to use these data for numerical calculations.
- Fuzzy definition of the business policy due to different possible management approaches.

In the sequel we will illustrate the iterative refinement of a concern model via numerical simulation, numerical solution of optimal control problems by direct optimization methods and modeling. For the first model we use, see Lesourne and Leban (1978), Feichtinger and Hartl (1986) and Will (1992):

$$\begin{aligned} \dot{X} &= (1-\tau)P - D\,, & (1)\\ \dot{Y} &= I - \delta\,(X+Y) - (1-\tau)P + D\,, & (2)\\ \dot{J} &= e^{-rt}D & (3) \end{aligned}$$

with $P = pF - \omega L - \rho_k Y - \delta\;(X+Y)$ and $F = \alpha\,(X+Y)^{\alpha_k} L^{\alpha_l}$. The dot denotes the derivative w. r. t. the independent variable $t \in [t_0, t_f]$. Initial time t_0 and terminal time t_f are fixed. The state variables X, Y and J denote equity capital and loan capital of the concern and the accumulation of the discounted dividends, respectively. With the output F, the profit P is the difference between sales proceeds pF and labour costs ωL, loan capital costs $\rho_k Y$, and depreciation $\delta\;(X+Y)$. τ denotes the tax rate and r denotes the notational interest on equity capital. The performance index $J(t_f)$ is to be maximized with the control functions D, I, and L, which denote dividend, investment, and number of employees. The state constraints $0 \le Y$ and $Y \le \kappa X$ (borrowing limit) and the control constraints $0 \le D$, $0 \le I \le I_{\max}$, and $0 \le L$ have to be fulfilled for all $t \in [t_0, t_f]$. For analytical and numerical calculations with this first concern model, see LESOURNE and LEBAN (1978), FEICHTINGER and HARTL (1986), WILL (1992) and KOSLIK et al. (1993).

Refinement of the first concern model

New direct optimization methods, e.g., the direct collocation method DIRCOL, see VON STRYK (1993), VON STRYK and BULIRSCH (1992), enable an comfortable, fast and reliable numerical solution of optimal control problems. In detail, major advantages of the direct collocation method are:

- Even non differentiable model functions, e.g., i and ρ_k in the new model, can be handled, since no explicit numerical integration of the differential equations is done, see Fig. 1.
- The large domain of convergence enables the computation of the optimal solution even with a poor initial guess.
- Although the calculation of the adjoint differential equations is not necessary, the direct collocation method DIRCOL yields accurate estimates for the adjoint variables. These estimates facilitate the use of an highly accurate indirect optimization method, e.g., the multiple shooting method.

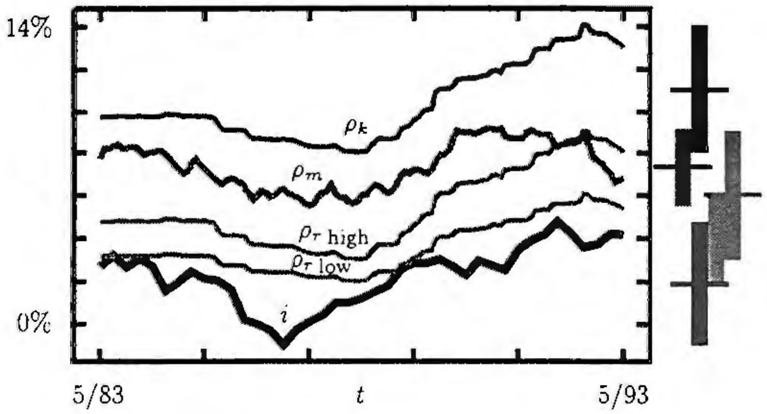


Fig. 1: Real economic functions with mean value and fluctuation for the period May 1983 to May 1993 in Germany (West): Inflation rate i (thick black curve), interest rate ρ_k for loan capital (thin dark gray curve), current yield ρ_m (thick gray curve) and two risk premium rates $\rho_{r\,\text{low}}$ and $\rho_{r\,\text{high}}$ for the equity capital in the concern (thin light gray curves).

Various refinements of the first concern model (1) - (3) lead to the new concern model:

$$\dot{S} = S_c\,, \tag{4}$$
$$\dot{L} = L_c\,, \tag{5}$$
$$\dot{Y} = Y_c\,, \tag{6}$$
$$\dot{X} = +I + (1-\tau)\,(P - \rho_r X)\;, \tag{7}$$
$$\dot{X}_m = -I + (1-\tau)\,X_m\,\rho_m\,, \tag{8}$$
$$\dot{X}_r = (1-\tau)\,\rho_r\,X\,, \tag{9}$$
$$\dot{d} = -d\,\ln(1+i) \tag{10}$$

with

$$F = \alpha\,(X+Y)^{\alpha_k}\,L^{\alpha_l}\;,$$

and

$$P = \frac{1}{d}\,(p\,(F - S_c) - \sigma\,S - \omega\,L) - \rho_k\,Y - \delta\,(X+Y)\;.$$

Equations (4) and (5) represent the level of stocks $S \in [S_{\min}, S_{\max}]$ and number of employees $L \geq 0$ (new controls $S_c \in [S_{c,\min}, S_{c,\max}]$ and $L_c \in [L_{c,\min}, L_{c,\max}]$). The loan capital $Y \in [0, \kappa X]$ can be controlled via the new control $Y_c \in [Y_{c,\min}, Y_{c,\max}]$ replacing the former I, see (2) and (6). The owner of the equity capital $X \geq 0$ in the concern holds a remaining part $X_m \geq 0$ of his capital in alternative capital assets, e.g., fixed-interest stocks. With the help of the investment $I \in [I_{\min}, I_{\max}]$, the capital flow between X and X_m can be directed, see (7) and (8). The demand for a risk premium by the owner of the equity capital $X \geq 0$ in the concern is modeled in Eq. (9). For numerical calculations with real economic data or realistically modeled economic cycles, see BREITNER et al. (1993) and KOSLIK et al. (1993), it is necessary, to calculate an **exact** discounting function $d(t)$ for a **variable** inflation $i(t)$. The derivation of the Eq. (10) and the initial condition $d(t_0) = 1$ for $d(t)$ can be found in KOSLIK et al. (1993). For the investigation of realistically modeled economic cycles it is comfortable, to add the equation

$$\dot{k}_p = \frac{2\pi}{k_l(t)} \tag{11}$$

for the position k_p in an economic cycle with the initial condition $k_p(t_0) = k_{p,0}$. The duration k_l of the economic cycle can be chosen even discontinuous. With the help of k_p, the economic functions can be modeled as trigonometric functions, e.g., $i(t) := i_m + i_v \sin k_p(t)$. All the economic parameters and function, e.g., τ and δ have been determined carefully. The owner of the capital $(X + X_m)$ and the management of the concern try to maximize the total profit,

$$Z = X(t_f) + X_m(t_f) + (1-\tau)\,p\,\frac{S(t_f)}{d(t_f)} \quad \longrightarrow \quad \max\,!$$

with respect to the control functions S_c, L_c, Y_c and I.

The derivation of the Eqs. (4) – (11), the related initial conditions and the state and control constraints for the design of a realistic concern model requires some iterations of simulation, optimization and modeling. Optimal solutions for various economic settings have to be calculated numerically for all preliminary models. These solutions must be compared to well known real phenomenona in micro- and macroeconomy, see, e.g., HEINEN (1985). A further refinement of the model – equations, boundary conditions, constraints or model functions – is required, as long as unrealistic solutions are obtained. The final, complex and very realistic concern model can be found in KOSLIK et al. (1993).

Numerical results

Various numerical calculations with the realistic concern model have shown its validity. The optimal solutions gain insight into the optimal management of a concern on the one hand and can help to understand macroeconomic phenomenona on the other hand. The numerical results include optimal solutions and optimal limit cycles

- for the real data of the inflation i, the interest rate for loan capital ρ_k and the current yield ρ_m, see Fig. 1 and Fig. 2;
- for realistic economic cycles including non-constant cycle duration $k_l \in$ [3 years, 9 years];
- different planing horizons t_f (1 year, 3, 5 and 10 years);
- for the best possible duration $k_l^*(t)$ of the economic cycle related to the initial conditions (best case analysis), see Fig. 3;
- for the worst possible duration $k_{l*}(t)$ of the economic cycle related to the initial conditions (worst case analysis).

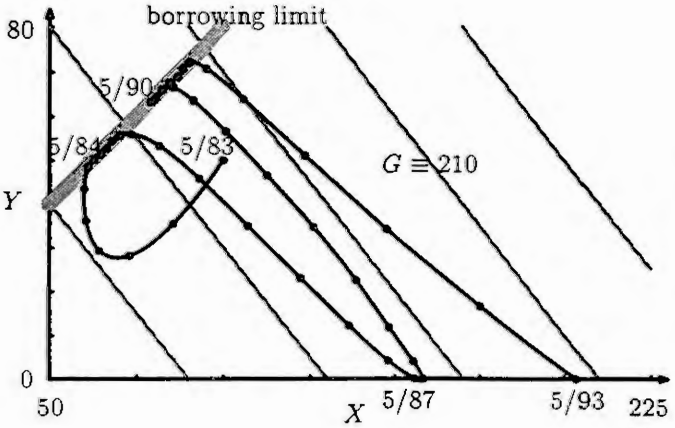


Fig. 2: History of loan capital Y versus equity capital X (black curve) for the moderate risk premium $\rho_{r\,\text{low}}$. The dots on the curve mark the collocation nodes of the direct collocation method DIRCOL. The borrowing limit (thick gray line) and lines of constant joint capital $G := X+Y$ (thin gray lines) are also depicted.

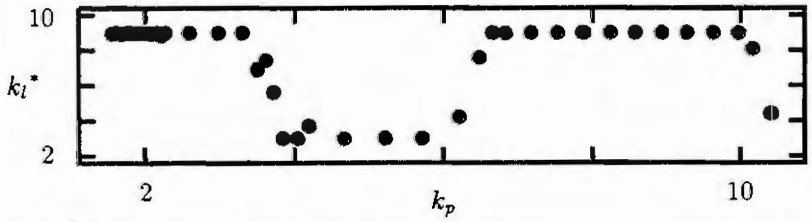


Fig. 3: Best case $k_l^* \in$ [3 years, 9 years] versus k_p (dots): Good estimate with the help of the direct collocation method DIRCOL.

Many of the calculated optimal solutions and optimal limit cycles can be found in KOSLIK et al. (1993). Our future research in this area is devoted to the numerical calculation of optimal solutions with indirect optimization methods, e.g., the multiple shooting method, too. Furthermore, the application of differential game theory, see, e.g., BREITNER et al. (1993), is planed for the handling of unknown, future economic data.

Acknowledgements

We would like to thank PROF. DR. DR. H.C. R. BULIRSCH and PROF. DR. H. J. OBERLE for helpful discussions. The authors would like to thank Bankdirektor P. SOLLINGER, HYPO-Bank Zentrale München, for providing us with the real data of i, ρ_k and ρ_m for the last ten years. This work has been supported by the Deutsche Forschungsgemeinschaft within the Project "Anwendungsbezogene Optimierung und Steuerung" and FORTWIHR, the Bavarian Consortium on High Performance Scientific Computing.

References

BREITNER, M. H., KOSLIK, B., VON STRYK, O. and PESCH H. J.: *Optimal Control of Investment, Level of Employment and Stockkeeping.* In: Proceedings: 18. Symposium über Operations Research, Köln, 1. – 3. September, Physica-Verlag Heidelberg, 1993.

BREITNER, M. H., PESCH, H. J. and GRIMM, W.: *Complex Differential Games of Pursuit-Evasion Type with State Constraints, Part 1: Necessary Conditions for Optimal Open-Loop Strategies, Part 2: Numerical Computation of Optimal Open-Loop Strategies.* Journal of Optimization Theory and Applications **78** **(3)**, pp. 419 – 441, pp. 443 – 463, 1993.

FEICHTINGER, G. and HARTL, R.F.: *Optimale Kontrolle ökonomischer Prozesse.* Walter de Gruyter, Berlin, New York, 1986.

HEINEN, E.H.: *Industriebetriebslehre.* Gabler, Wiesbaden, 1985.

HILTEN, O., KORT, P. M. and VAN LOON, P. J. J. M.: *Dynamic Policies of the Firm: An Optimal Control Approach.* Springer, Berlin, 1993.

KAMIEN, M. I. and SCHWARTZ, N. L.: *Dynamic Optimization: The Calculus of Variations and Optimal Control in Economics and Management.* North-Holland, New York, 1981.

KORT, P. M.: *Optimal dynamic investment policies of a value maximizing firm.* Springer, Berlin, 1989.

KORT, P. M., VAN LOON, P. J. J. M. and LUPTÁCIK, M.: *Optimal Dynamic Environmental Policies of a Profit Maximizing Firm.* Journal of Economics (Zeitschrift für Nationalökonomie) **54** **(3)**, pp. 195 – 225, 1991.

KOSLIK, B., BREITNER, M. H., VON STRYK, O. and PESCH H. J.: *Modeling, Optimization and Worst Case Analysis of a Management Problem.* Report **465**, Deutsche Forschungsgemeinschaft, Schwerpunkt "Anwendungsbezogene Optimierung und Steuerung", TU München, Mathematisches Institut, 1993.

LESOURNE, J. and LEBAN, R.: *La Substitution capital-travail au cours de la croissance de l'entreprise.* Rev. d'Economie Politique **4**, pp. 540 – 564, 1978.

VON STRYK, O.: *Numerical solution of optimal control problems by direct collocation.* In: R. BULIRSCH, A. MIELE, J. STOER and K.-H. WELL (eds.), Optimal Control. International Series in Numerical Mathematics **111**, Birkhäuser, Basel, pp. 129 – 143, 1993.

VON STRYK, O. and BULIRSCH, R.: *Direct and indirect methods for trajectory optimization.* Annals of Operations Research **37**, pp. 357 – 373, 1992.

WILL, O.: *Numerische Untersuchung im Wachstumsmodell von Lesourne und Leban.* Diploma thesis, Fachbereich Mathematik an der Universität Hamburg (under direction of PROF. DR. OBERLE), 1992.

ECONOMY, DEMOGRAPHY AND OCCUPATION OF THE TERRITORY INSIDE A FUNCTIONAL ECONOMIC AREA (FEA)

José Manuel González
Departamento de Economía Aplicada, Universidad de La Laguna
La laguna, Canary Islands, Spain

Abstract The phenomenon of Mass Tourism in the 1960's has converted the islands of Tenerife in Canary Islands into typical Functional Economic Area where the booming economic development has been disturbed by several crises affecting the welfare of the islanders. In particular these crises have induced perverses efects in the economy as the unemployment and the black employment. In this paper we simulate the generation of this black employment and we find that such pernicious results is generated by the high level of unemployment.

1.INTRODUCTION

The economic model which has taken form historically in the Canary Islands was substained by Agriculture for Export of different cash-crops, controlled by a dominant minority (generally associated with foreign entrepreneurs). It based its high level of productivity in superb climatic and environmental conditions and on its coexistence with a subsistence agriculture or an economy of scarcity that left available plenty of cheap labour for the commercial plantations. This model gave rise to a succession of crises which cyclically impoverished the islands causing great waves of emigration as an ecological escape valve (according to Antonio Machado Carrillo's terminology [10]), whenever the demographic capacity was reached.

In addition, the phenomenon of Mass Tourism in the 1960's substantially modified the productive structure producing a clear imbalance in the economic ecological health mechanism known up to then. Precisely the impact of this dominant economic activity on population, land occupation and standard of living conditions has converted the two main islands of the Archipelago: Gran Canaria and Tenerife into typical Functional Economic Areas, where we can find the following trends in economic development during the last twenty years:

1) Firstly an accelerated growth in the number of tourist places is perceived, which has expaned the tourist resorts to a 150000 beds in Tenerife.

2) Together with this development of the tourist industry there has been an overall growth in population and urbanized land alerting us to the appearance of phenomena like overcrowding and excessive urbanization in the not too distant future.

3) This urban sprawl is due not only to the high index of natural increase on the islands but also to the foreign residents occupying villas near the tourist complexes. In addition, this new form of settlement has given rise to a change in the net direction of migration. Previously negative in sign, the emigrants were mainly young single males, it is now positive in sign, the loss being exceeded by the influx of outsiders.

4) Lastly, the spectacular increase in per capita income among the inhabitants is not directly corresponded by a comparable increase in the social conditions and facilities determining the citizens' Quality of Life.

Indeed, the Standar of living of the population and simiraly the booming economic development in the islands has been disturbed by several crises affecting the welfare of the islanders. In particular as consequence these economic crises have had pernicious results on the present development of the economy. The most significant of these is the high level of unemployment seriously affecting the future of the islands. This has reached more than 20% of the working population, threatening imminent strangulation of the prospects for economic development in the Archipelago.

Associated with the high index of unemployment there is another factor affecting negatively the economic situation in the islands. This is the phenomenon of black employment, fruit and motive force of a Black Economy unquantifiable in terms of Regular Accountancy which contributes to a certain financial chaos in the islands.

In this article we aim to evaluate quantitatively the degree to which this type of employment has repercussions on the economy in general. Concretely, we study this phenomenon in the island of Tenerife and with the aid of diverse econometric modelling and simulation techiniques we undertake an analysis contrasting real data with diverse economic theories intended to explain the underground economy within the framework of the world economy as a whole, and in particular that of Spain.

We can already put forward that our study is sen to be highly significant both in explaining black employment in the Canaries and in the practical justification of the theories which set out to explain such a phenomenon.

2. BLACK EMPLOYMENT AND THE 'BLACK' ECONOMY

In the opinion of all the experts: "The growth of black employment is found to have the existence of relatively low incomes as one of its fundamental cause". In consequence therefore, periods of economic crisis give rise to the appearance of this type of employment, the driving force and basis of the Underground Economy. Similarly the accelerated growth of the Canary economy, resultant population overflows and the weakness of the development model have provoked these succesive crises giving rise to the phenomenon under study here. For this reason we set off from the hypothesis that the Canary economic scenario is one of the clearest in having trigged the appearance of this type of employment.

We consider that the 'Black' economy must be understood as the set of:

activities that should be accounted for within the concept of gross interior product, but may be omitted in practice because one or more of the participants attempts to hide them from the public authorities

(Official definition offered by the OECD, Ruesga Benito [11], page 55)

Under this definition enter activities such as legal but underclared production drug-dealing, prostitution, income in kinf-goods rather than money, and an enless variety of hidden activities. However, it seems that the quantitative evaluation of this sort of economic activities may be difficult or even impossible. For this reason we need to restrict the black employment in such a way as to make its evaluation feasible, so we must resort to the satistical series at our disposition. It should also appreciated that there is a substantial divergence between the data collected by the EPA (Encuesta de la Población Activa[1] quantifying the total number of adults considered active in employment, and the real figure for those affiliated to the Social Security (reflecting those employded officially). Contrasting the annual difference between both series, we are able to evaluate the amount of black employment at least at its minimum limit.

Thus being established the concept of this employment we wish to

[1] Survey of Active Population

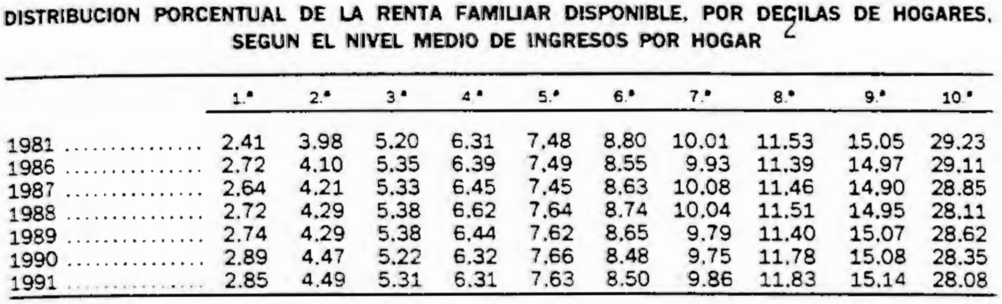
evaluate, let us then proceded to examine the various theories that purport to explain its appearance. We have already put forward that the existence of lower incomes largely explains this. What is more, the fall in income for the average Canary islands family stimulates a search for additional earnings outside the legally regulated labour market. However, this income-drop:

> *... is due not so much to the behaviour of the real salary as to the rise in unempolyment levels and in consequence to the decrease in the number of earners per family. The pool of unemployment in this way becomes one of the fundamental sources of labor supply for the unregulated sector.*
>
> (M. González, A. Vega,..., page. 45)

Thus in essence, the existence of low incomes and the unemployment pool or reserve must be considered as indicators that to a greater extent enlarge or justify black labour. We are thus interested in estimates the size of each.

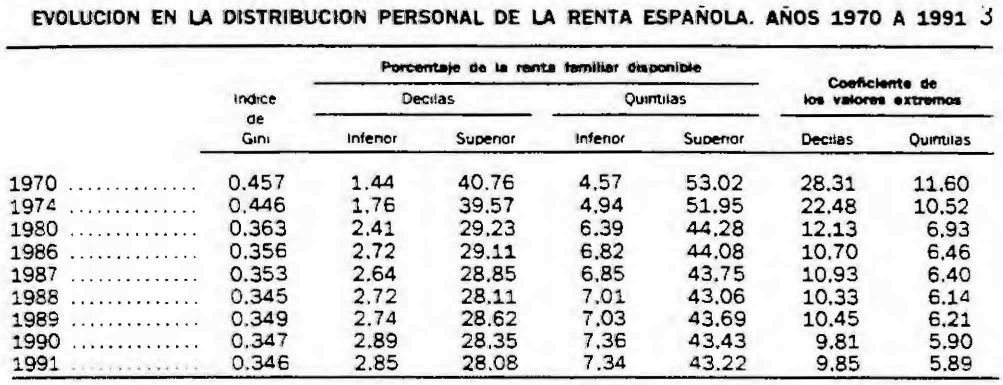
Annual unemployment for Tenerife is quoted in the satistical series put together by the INEM (Instituto Nacional de Empleo), and so these data are known for the last 12 years at least. Under other conditions, the existence of low incomes is found to be intimately related to the personal income distribution and its greater or lesser degree of concentration. So we can turn to the data collected by the annual reports (Informes Anuales) elaborated by the Banco Bilbao-Vizcaya, which being based on information obtained by the Family Budget Survey (Encuesta de Presupuestos Familiares) appear in the following tables:

Table I.



DISTRIBUCION PORCENTUAL DE LA RENTA FAMILIAR DISPONIBLE, POR DECILAS DE HOGARES, SEGUN EL NIVEL MEDIO DE INGRESOS POR HOGAR [2]

	1.ª	2.ª	3.ª	4.ª	5.ª	6.ª	7.ª	8.ª	9.ª	10.ª
1981	2.41	3.98	5.20	6.31	7.48	8.80	10.01	11.53	15.05	29.23
1986	2.72	4.10	5.35	6.39	7.49	8.55	9.93	11.39	14.97	29.11
1987	2.64	4.21	5.33	6.45	7.45	8.63	10.08	11.46	14.90	28.85
1988	2.72	4.29	5.38	6.62	7.64	8.74	10.04	11.51	14.95	28.11
1989	2.74	4.29	5.38	6.44	7.62	8.65	9.79	11.40	15.07	28.62
1990	2.89	4.47	5.22	6.32	7.66	8.48	9.75	11.78	15.08	28.35
1991	2.85	4.49	5.31	6.31	7.63	8.50	9.86	11.83	15.14	28.08



EVOLUCION EN LA DISTRIBUCION PERSONAL DE LA RENTA ESPAÑOLA. AÑOS 1970 A 1991 [3]

	Indice de Gini	Porcentaje de la renta familiar disponible				Coeficiente de los valores extremos	
		Decilas		Quintilas			
		Inferior	Superior	Inferior	Superior	Decilas	Quintilas
1970	0.457	1.44	40.76	4.57	53.02	28.31	11.60
1974	0.446	1.76	39.57	4.94	51.95	22.48	10.52
1980	0.363	2.41	29.23	6.39	44.28	12.13	6.93
1986	0.356	2.72	29.11	6.82	44.08	10.70	6.46
1987	0.353	2.64	28.85	6.85	43.75	10.93	6.40
1988	0.345	2.72	28.11	7.01	43.06	10.33	6.14
1989	0.349	2.74	28.62	7.03	43.69	10.45	6.21
1990	0.347	2.89	28.35	7.36	43.43	9.81	5.90
1991	0.346	2.85	28.08	7.34	43.22	9.85	5.89

As one may appreciate from these, in the second tables a coefficient is set out that could well be used as an indicator of the separation of wealth between Spanish families knowing the coefficient between the 1st and 5th

[2] Porcentual Distribution of the familiar income by deciles of homes, according to the medium level of earnings

[3] Evaluation of the personal distribution of the spanish income, years 1970-1991

quintile. If this coefficient increases, a greater difference arises between the incomes of the richer and poorer families. Therefore an increase is perceived in the very low incomes which respect to the mean. We can accept the value of this coefficient as an index accurately defining the appearance of low incomes and so are able to work further with it.

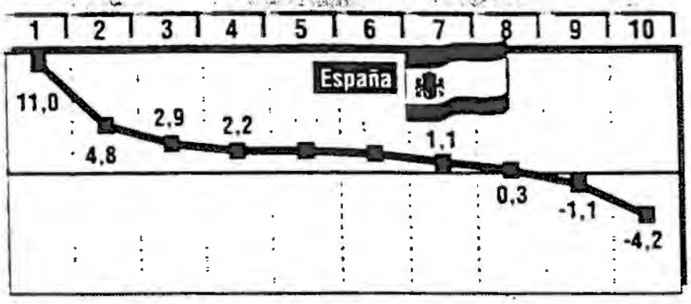
However, it happens we only have available six consecutive values for this coefficient, besides this deriving from the overall Spanish national income figures. We therefore find ourselves obliged to both enlarge the series of observations and relate the data to conditions in Tenerife.

With this intention in mind, let us try to relate our indicator with some other for which we might have a wider range of observations at our disposal, preferably without being aggregated at the provincial level. What should this new indicator be?

It is known that the family or household income distribution is intimately linked with the increase in Public Expenditure that enrich the Welfare Benefits system, etc. should result in a more equal income distribution. By the Spanish government has given special attention to this instter through it policies during the last few years (EL PAIS, Tuesday 1 June :993, see graph)

Crecimiento de la renta 1980-1990

Niveles de renta ordenados desde 1, más pobres, a 10, más ricos.[4]



We must admit that in Spain the behaviour of the state has achieved the reduction of difference between rich an poor, at least during the 1980's

On the other hand:

Policies which expand social expenditure in a recessive economic situation (the case of Spain) *will result in acceleration the decrease in GNP by contributing to a lower rate of saving.*

Banco Bilbao-Vizcaya, page 94

In other words, Social expenditures and the GNP growth rate seem closely related in a reverse correlation. This fact already recognized as a perverse effect of the Welfare economy (Cameron, [5]) drives us to seek a functional relationship that will interconnect the increase in equidistribution of income with the GNP variation. If:

X_t denotes the coefficient of the family income distribution extreme values in quintile in time t, deflated by the contribution of public expenditure and

Y_t the GNP varaiation rate in time t

the equation:

$$(1)\ X_t = A + B \times Y_t$$

allows us a relationship to be established between the two indicators. In this

[4] Income Growth 1980-90
Income levels ranging from 1, poorest to 10, richest.

way if we evaluate the parameters A and B with the aid of the least sqaure method we will obtain:

$A = 4.6626$ and $B = 0.32034$

When we perform this linear regression adjustament with tha aid of the six values collected for X_t in Table 2.. We then confirm that the test of the significance for the constant A and the variable Y_t, gives us signifciantly high values (23.6362 and 7.2571 respectively), that the factor R-squared is very near unity: $R^2 = 0.94611$, and that the Durbin-Watson correlation test justifies the goodness of fit giving a value of 1.8972 (very near the optimum of 2). That is, the regression carried out using equation (1) involves a series of indicators which gives a good test of the significance of the adjustment as a whole, permiting us to accept as valid the hypothesis that the income differential compares well with the GNP growth rate.

Table 2. Variables which are used in equation (1)

Years	Variable Y_t	Variable X_t
1986	3.20	6.46
1987	5.64	6.58
1988	5.16	6.29
1989	4.75	6.14
1990	3.69	5.71
1991	2.29	5.49

With all the foregoing discussion, we can make use of the equation (1) to widen the range of observations of variable X_t, limiting ourselves to the economic conditions of Tenerife in order to undertake a simulation of the black employment found in its economy

3. SIMULATION OF BLACK EMPLOYMENT

Having collected the hypothesis relative to the nature and development of the black or unrregulated employment UE_t in the coordination of the variables X_t and U_t (unemployment in period t, let us proceed to simulate the action of these on UT_t applying various multiple linear regressions. We begin establishing a direct relationship between X_t, U_t and UE_t through equation (2)

$$(2) \quad UE_t = A + B \times X_t + C \times U_t$$

Given the linear regression of (2), we can find the following values of the parameters:

$A = 5840.6$ $B = -41.2778$ and $C = 0.83163$

But it happens that the different test for the goodness of fit of this regression do not indicate the pressence of optimal fit, since the values obtained for the sattistical test of significance of the parameters A and variable X_t: 0.29950 and -0.0094131 and the significance of the regression as a whole with a R^2 of 0.66163 invalidate the good use of equation (2). We must thus recur to a new hypothesis to better describe the generation of black employment.

For this we set off from a theory of frequent use in Econometrics. It is one which justifies persistence in the repetition of habits, or the propensity to repeat a certain behaviour. That is, we are speaking of the same circumstance recognized by Brown [4] in the propensity to consume, that in his words:

The habits, customs, standards and levels associated with real consumption previously enjoyed become "impressed" on the human physiological and psychological systems and this produces an inertia or "hysteresis" in *consumer behaviour. Because of this inertia, consumer demand reacts to changes in consumer income with a certain slowness, and thus past real consumption exerts a stabilizing effect on current consumption...*

Therefore taking this idea up again in our discussion, the black employment must be a function of the present uneployment and of the black employment found in the past. So, parting from the hypothesis that the effect of the persistence of habits is, besides being continuous, an inverse function of time, the migration flow in the immediately previous period is that which must exert a greater influence on the present. In consequence we may formulate the following equation:

(3) $UE_t = A + B \times X_t + C \times U_t + D \times UE_{t-1}$

Proceding therefore to make an test of regression to equation (3) with the aid of the variables refering to a period of 12 years, we find the following parameter values:

A = -10566.0, B = 92.9095, C = 0.18980. and D=1.0548

We also find the the t'Student test of significance give us the values:

-1.6129 for the coefficient A

0.071049 for the variable X_t

1.3793 for the variable U_t, and

8.2202 for the delayed variable UE_{t-}

It being the test R^2 that gives us de significance of the regression as a whole equal to 0.97537 and the Durbin-Watson test leads us to the value 2.2441 (near to 2). That is, the simulation we carried out using equation (3) models the generation of the black employment with a very good fit. Taking this in account, the values found in the test of significance of the variables invite us to vary equation (3) slightly, and by this means we can proceed to pay less attention to the comparatively meanningless value X_t, obtaining the equation (4) which follows:

(4) $UE_t = A + B \times U_t + C \times UE_{t-1}$

In this case, the multiple regression gives us the following results!

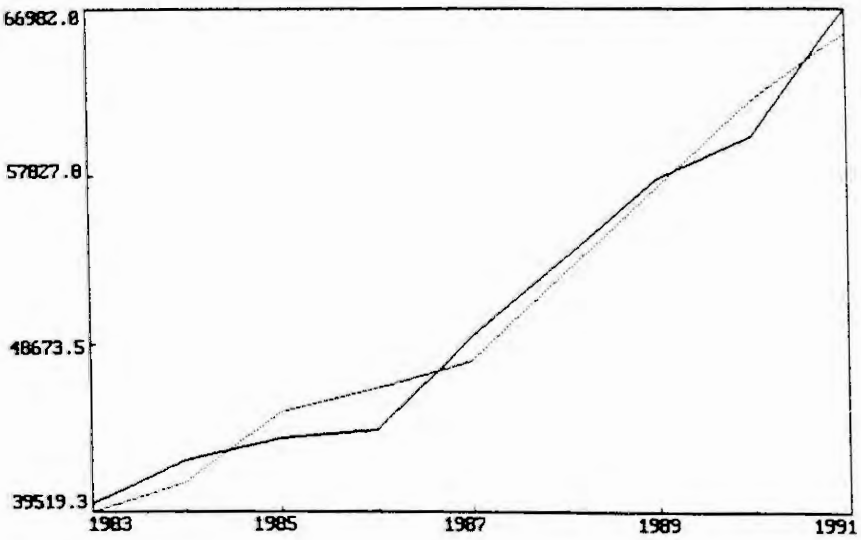
A = -10244.7 with a significance level of -2.3685

B = 0.19393 which leads us to a better test of significance than previously for the variable U_t: 1.7018

and C = 1.0546 with the significance that assumes a t'Student test of 9.005 for the delayed variable UE_t.

In addition, the goodness of fit improves overall as R^2 increases to 0.97535, as does the Durbin'Watson test from which an estimation of 2.2318 is obtained -nearer to the optimal value of 2-. This improvement in the simulation of UE_t is clearly reflected in the following graph where the accuracy in the calculation of equation (4) is seen.

Plot of Actual and Fitted Values



We can thus conclude that the generation of black or unrregulated employment in the island economy of Tenerife responds jointly as much to the unemployment increase as to the persistence in the habit of population to generate and maintain a black economy. The income differential does not substantially affect this unrregulated employment, but we must accept that unemployment itself include the income difference as expressed in the qoute from M. González et al.,([6])

4. REFERENCES

[1] Ando, A. and Modigliani, F., The life'cycle hypothesis of savings: Aggregate implications and tests, American Economic Review, 53, 55-84

[2] Banco de Bilbao-Vizcaya, Informe Económico 1992, Madrid, 1993

[3] Beltrami, E., Mathematics for Dynamic Modelling, Academic Press Inc, San Diego, 1987

[4] Brown , T. M., Habit persistence and lags in consumer behaviour, Econometrica, 20, 355-371

[5] Cameron, D., Public Expenditure and Economic Performance in International Perspective, The future of Welfare, Basil Blackwell, Oxford, (1985), 8-22

[6] González, M. - Vega Martín, A. - Morín Vargas, V. and Delgado García, F., Grado de implantación de las distintas modalidades de contratación y situaciones fraudulentas en el mercado de trabajo en Canarias, Consejería de Trabajo , sanidad y Seguridad Social, Gobierno de Canarias, S. C. de Tenerife, 1986

[7] Intriligator, M. D., Econometric Models, Techniques and Applications, North-Holland, Oxford, 1978

[8] Johnston, J. J., Econometric Methods, MacGraw Hill Inc, 1984

[9] Lafuente Felez, A., Una medición de la economía oculta en España, Boletín de estudios Económicos, no. 11, Diciembre, (1980)

[10] Machado Carrillo, A., Ecología, Medio Ambiente y Desarrollo Turístico en Canarias, Consejería de la Presidencia, Gobierno de Canarias, S. C. de Tenerife, 1990

[11] Ruesga Benito, S., Economía oculta y mercado de trabajo: aproximación al caso epañol, I.C.E., no. 607, Madrid, 1984

[12] Sandefur, J. T., Discrte Dynamical Systems. Theory and Applications, Clarendon Press, Oxford, 1990.

COMPREHENSIVE QUANTITATIVE CALCULUS AS A TOOL IN POLICY MODELING: A BROAD DYNAMIC SIMULATION MODEL OF R&D INPUT

CHEN JIN
Institute of Management Science, Zhejiang University.
Hangzhou, 310027, P. R. China.
WU MINGHUA
Department of Mathematics, Zhejiang University,
Hangzhou, 310027, P. R. China.
XU QINGRUI
Research Center of Management Science & Strategy,
Zhejiang University, Hangzhou, 310027, P. R. China.

Abstract. In recent years quantitative calculus has become an increasingly important tool in policy impact analysis. In this paper, we introduce a newly comprehensive mathematical model based on System Dynamics, and apply this model to a dynamic policy simulation on the resource allocation of Research & Development (R&D) in China. System isomorphism is used to evaluate the main functional loops of the model, and the variable structural control is also used to make out more rational R&D policy.

I. INTRODUCTION

Nowaday quantitative calculus has become an increasingly important tool in policy impact analysis and precise policy-making. One of the analysis of quantitative relationships in complex social & economic was originated by M. I. T. System Dynamics Group with the theory and methodology of System Dynamics (Forrester, 1961; Roberts, 1978, Senge, 1990). For decades System Dynamics models were used to analyse the interrelationship and dynamics change over time among the social & economic variables, and got some existing findings. As China is still one of the less developed economics, large scale investment on R&D as developed countries is impossible, but the low increasing rate of R&D input also affects the catch up with the scientific & technological more advanced. As the prospects for entering into modern industrialisation are not so obvious nor can be taken for granted that late-comers have the advantages, less development countries including China must pay attention to the rational R&D input so that the development can take place through synchronization of build up science & technology capacity and elimination of obstacles to development. Concerning this issue, we have developed several System Dynamics models (Xu Qingrui et al, 1984, 1986, 1989, 1992).

As any social & economic model is not perfect, System Dynamics modeling usually lacks of a more formalized procedures of mathematical analysis, and trial—and error approach was always used to the realising acceptable policies, the number of policy alternatives is limited by the analyst's own experiences & judgement rather than the attainment of objective criteria (Toyoda, 1991), and the cost of policy test & simulation is relatively high. The broad dynamic simulation model is based on our further study of mathematical analysis of system behavior and system structure. In this paper, we introduce a System Dynamics model on the R&D input, based on System isomorphism we mathematically make out the main functional loops which is helpful to fing the varing characteristics of this model. In the meantime we also intend to induce variable structure control in the model for more rational test & simulation of R&D resource allocation policy.

II. SYSTEM ISOMORPHISM ANALYSIS OF R&D INPUT MODEL

By system thinking, the input of Research & Development must be coordinated with the development of economy, education, finance as well as modification the internal structure of science & technology per se. So there are 5 subsystems in this model: economy subsystem, population subsystem, education subsystem, finance subsystem as well as science-technology subsystem.

The actual System Dynamics model of R&D input consists of around 250 variables and 50 feedback loops, so the evolving rule of system is more complicated, and traditionally a great deal of tests are needed to observe the impact of each parameter and feedback loop on the system behavior. The focus of isomorphism analysis is trying to change the structural analysis to find out the functional loop of the total system so as to the effectiveness of policy test and simulation of this R&D input model could be highly raised.

Let system is X, x, y, ... are the elements of system, the structural relationship is $x \rightarrow y$. Assume the mapping from X to the directed graph is F, then we have:

$$F(x) = \text{apex } x, \ \forall\, x \in X$$
$$F(x \rightarrow y) = \text{apex } x \rightarrow \text{apex } y = F(x) \rightarrow F(y).$$

For each loop C in the directed graph, define

$$C_i=\begin{cases}1 \text{ if are i is in } C_{ir} \text{and direction is the sameas the loop direction;}\\ -1 \text{ if are i is in } C_{ir} \text{and arc direction is reserved to to the loop direction;}\\ 0 \text{ if arc i is not in } C_{ir}.\end{cases}$$

Based on above difinations, the algorithm of finding the functional loops is done as follows:

(1) Symbolizing the system X with its equivalent directed graph D, evaluating out the total related matrix R_e and related matrix R.

Here, D is the directed & connected graph with n apexes and l arcs,

$$\text{then: } R=(r_{ij})_{n\times l}, r_{ij}=\begin{cases}1 & \text{if apex i is the starting apex of arc i;}\\ -1 & \text{if apex i is the finishing apex of arc i;}\\ 0 & \text{if apex i is not in arc i,}\end{cases}$$

(2) Finding the non—singular sub—matrix, and assuming it is R_{12}, then $R=(R_{11}, R_{12})$;

(3) Calculating the basic loop matrix C_f, C_f is defined as:

$$C_f=(C^{1T},C^{2T},\ldots,C^{(l-n+1)T})^T=[I-R_{11}^T(R_{12}^T)^{-1}]$$

(4) For every loop C_{ir}^k, C_{ir}^l the corresponding vector of C^k and C^l, verifying whether the direction number of common arc in these two arcs is the same or not, continuing step (5) when it is true, or turning to step (7);

(5) Calculating $g(C^k,C^l)$, and checking up whether the its non—vanishing vector has the same sign, if it is true then $C_{ir}^k \oplus C_{ir}^l$is the functional loop of the system;

(6) Returning to step(4);

(7) Calculating $R_e(D,C_{ir}^k)$ and $R_e(D,C_{ir}^l)$, where

$$R_e(D,C_{ir})=\delta[C_1E_n,C_2E_n,\ldots C_nE_n]\otimes R_e(D)$$

$$E_n^T=(1,1,\ldots,1)_{n\times 1}$$

(8) Finding the adjacent matrix $P(D, C_{ir}^k)$,

$$P(D,C_{ir})=R_e^+(D,C_{ir})\cdot R_e^{-T}(D,C_{ir});$$

(9) Checking up the whether there exist non—zero element in every row and column of $P(D,C_{ir}^k) \oplus P(D,C_{ir}^l)$, if it is true then $C_{ir}^k \oplus C_{ir}^l$ is the functional loop of the system;

(10) Returning to step (4).

We applied this algorithm to figure out the functional loops of the R&D input model as Figure 1 showed.

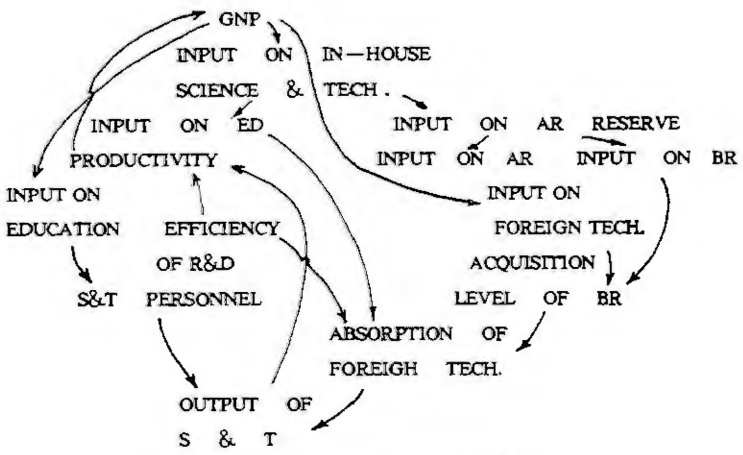


Figure 1. Main Functional Loops in the Model

III. VARIABLE STRUCTURE SIMULATION OF R&D INPUT MODEL

The dominant characteristics of System Dynamics model lies in the fact that the behavior of the model is mainly relied on the model's structure. Actually many modelers tried to forecast the long—range behavior of the system by using the fixed model structure, and it is expected to be improved. Recent years, variable structure control becomes the main pillar of control theory and application (Ryan, 1976; Gao Weibing, 1988). And the form of variable structure control is described as follows:

$$\dot{x}=f(x,u,t)$$

$$u_i(x)=\begin{cases}u_i^+(x), & \text{when } s_i(x)>c,\\ u_i^-(x), & \text{when } s_i(x)<c.\end{cases}$$

Here $x \in R^n$, $u \in R^m$, $t \in R$, and $s_i(x)$ is the switch function vector, and c is the constant, usually $u^+(x) \neq u^-(x)$.

Then the adjustment of model structure was realized by using DYNAMO language stated as follows:

```
A STATE. K=CLIP(STATE1. K, STATE2. K, TIME. K, CTIME)
C CTIME=NTIME (TIME OF CHANGE)
L STATE1. K=...
L STATE2. K=...
```

And STATE. K refers to x, STATE1. K to $u_i^+(x)$, STATE2. K to $u_i^-(x)$, CTIME to c .

Traditionally some of the models of Chinese R&D input were developed by the thought of planned economy. With the open—policy and introduction of market economy in China, the sources of R&D input could be foreign capital (like joint venture), own—raised money, government funding, venture capital, loan et al. And the model structure was changed step by step during model simulation as Figure 2 showed.

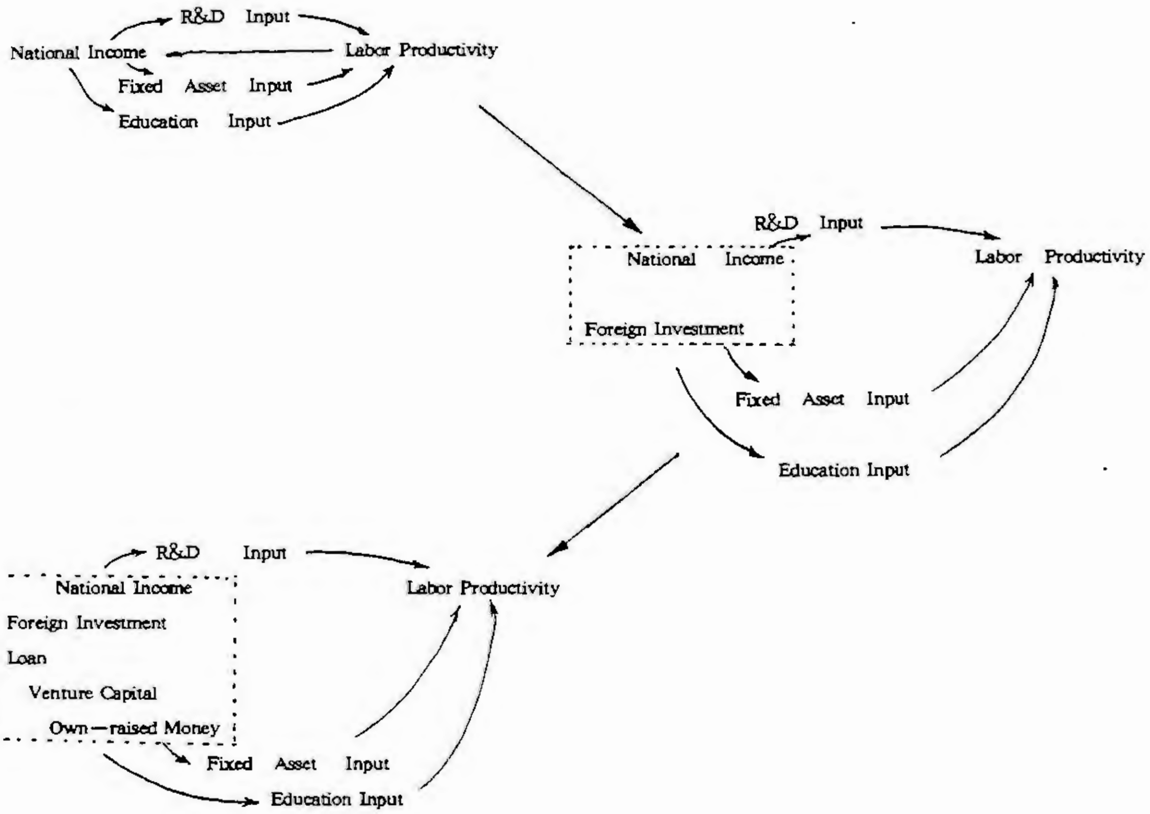


Figure 2.

The simulation results of fixed model structure and variable model structure is showed as Figure 3. From Figure 3, we can see that under the fixed model structure, the education input over national income should be greater that the R&D input over National Income (NI) before 2010. But considerring the varity of model structure induced by Chinese economic reform, the input of education should greater that the input of R&D until 2026 (see Figure 4). The delay of emphasizing the R&D input compared with education input was to some extent caused by the diversification of sources of R&D resource, and the variable structure adjustment of the model is made to simulation the new R&D input policy under changeable environment.

IV. CONCLUDING REMARKS

In this paper, we think that the relationship between model structure and model behavior, variable structure control are the two main pillars for the development of System Dynamics and the development of social & economic models.

We tried to analyse the relationship between model structure and model behavior in a mathematical framework of systemisomorphism, the knowledge of system isomorphism could be used to find the main functional loops in the R&D resource allocation model, and then the time and cost of test and simulation could be reduced to a great extent, and the effectiveness of policy test could be highly raised.

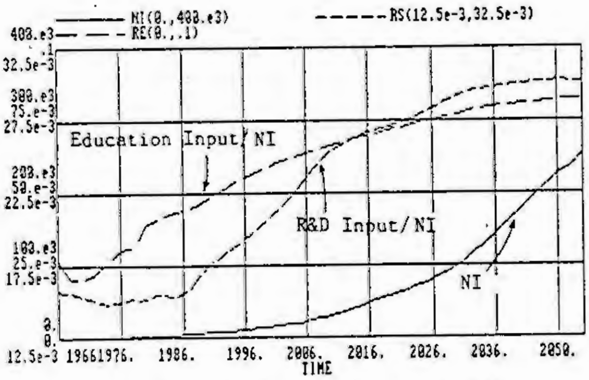


Figure 3. Simulation Result (1).

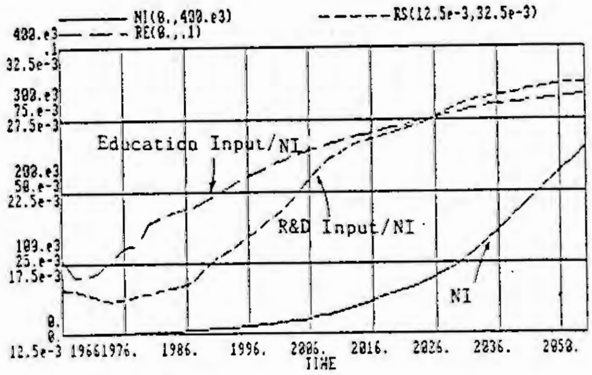


Figure 4. Simulation Result (2)

we also tries to construct and simulate variable structure model on R&D input based on the thought of variable structure control theory. As the variable structure control theory is mathematically difficulty for many social & economic modelers, we realized the variable structure adjustment step by step on the R&D input model by using the System Dynamics language DYNAMO, then made out R&D input policy. According to our research, the R&D input in China should take an exponential growth form, the ratio of R&D input over National Income should be 2.2% in 2000, and in 2050 the figure should be 3.2%. But priority of input by government should be given to the education before 2026, and then the coordinative development between economic, science & technology and education as well as sustainable development of China could be realized.

V. REFRENCES

[1] Xu Qingrui, Li Junjie, Jiang Shaozhong and Jiang Jiong, 1988. Science—Technology, Education and Economy System Dynamic Model. Proceeding of ICSSE'88. Beijing: International Academic Publishers.

[2] Xu Qingrui, Chen Jin et al, 1991. System Modeling on the Resource Allocation of Technological Innovation and R&D , Proceeding of 1991 International Conference of System Dynamics Society. Bangkok.

[3] Xu Qingrui, Chen Jin et al, 1993. Application of System Dynamics on the Resource Allocation of Scientific Research, Proceeding of 1993 International Conference of System Dynamics Society. Mexico.

[4] Xu Qingrui, Wu Gang, Chen Jin, 1992. On the Scale & Speed of the Input of Fundmental Research, Proceeding of International Conference of Inproving & Development of Science Foundation System. Beijing.

[5] Xu Qingrui, Zhu Keqin, Li Junjei, 1989. A System Model and Policy Analysis for National Education and Economy Development, Proceeding of International Conference of Computer — Based System. Spring Verger.

A MODELING SYSTEM OF SOCIOECONOMIC SYSTEMS BASED ON SYSTEM DYNAMICS

Hu Xiaohui

Beijing Institute of System Engineering
P.O. Box 9702-19
Beijing 100101
P.R. China

Abstract. In this paper a modeling system of socioeconomic systems is proposed. The system combines quantitative management methods and mathematical methods with system dynamics method. All of these methods can be constructed as system objects. Using object-oriented approach people can build models conveniently. The system is a useful and appealing tool for fast model building by users who are less familiar with computer programming.

1.INTRODUCTION

Socioeconomic systems are very complex. In the past, people simulated or modeled the systems using general-purpose languages (e.g., C, FORTRAN) or simulation languages such as GPSS, DYNAMO, SLAM, and so on. The methods employed were the event-scheduling approach and process approach. Adopting these languages to simulate a socioeconomic system is very time-consuming and costly, especially for complex or large systems. The purpose to simulate (or model) such a system is to solve difficult problems and make decisions as soon as possible. Sometimes the time needed to simulate a system is too long to do simulation research or to use the results of simulation. So it is necessary to develop a modeling tool in which people can simulate socioeconomic systems easily.

System dynamics can describe socioeconomic behaviors systematically. System dynamics is a rigorous method for qualitative description, exploration and analysis of complex system in terms of their processes, information, organizational boundaries and strategies, which facilitates quantitative simulation modeling and analysis for the design of system structure and control[5]. The original system dynamics software was DYNAMO and DYSMAP2. Both of these programs facilitate rigorous development of large models. In addition, the most recent development has been that of STELLA. It takes full advantage of the graphics interface of computers[6].

For modeling socioeconomic systems, these system dynamics software, DYSMAP2 and STELLA, have two disadvantages. First, system dynamics can not describe all behaviors and processes of socioeconomic system. Secondly, these software still needs programming with computer. In this paper, we propose a modeling tool which makes use of system dynamics method, quantitative management methods, and mathematical methods. Some of quantitative management methods can be directly used. And some of them needs integrating into system dynamics method. So that we can get a powerful tool. The system is built through object-oriented approach. We use object-oriented approach to realize the organic relationships among these methods.

Each of all these methods is coded into modules. In this paper, we call them system objects. Some of these modules can be integrated into bigger module. So this tool can describe all kinds of methods. Users can add modules into this system. The modules and variables of the diagrams are presented by icons. It is a particularly useful and appealing tool for fast model building by those users who are less familiar with computer programming. It is excellent for demonstrating the relationship between system feedback structures and system behaviors and for involving system actors more closely in the model building and analysis process.

2.THE DESCRIBING AND MODELING OF SOCIOECONOMIC SYSTEMS

2.1 System dynamics method and object-oriented approach

System dynamics came into being with its sharp system views. The system is objective, universal and it exists everywhere. All things are part of a system. Both the boundless universe and micro-world, without exception, are made up of systems. A system contains subsystems, some related and conditioned systems can make up of a much larger system. System objects are representations of some real-world things[3]. And systems and subsystems can be considered as system objects. The interaction of system objects can make up of another system object.

System dynamics is a branch of system science and a tool for understanding and analyzing certain kinds of complex problems. It is a unique tool for dealing with problems about the way systems behave through time. The field is based on developed concepts in system theory, and on the available techniques of computer simulation. The behavior of a system is determined by the nature of the relationships between its constituent parts. System dynamics has the character of relating events in the organization or its environment to possible actions that realize lawful state transitions[5]. Only actors are able to change the state of the system. Object-oriented approach can more naturally simulate the style in which mankind understand the real world. It can reflect the relationships between system behaviors and time. The interaction between objects makes it possible to send message from one object to another[3], and·the state transitions can be realized.

In system dynamics, the basic unit of a system is the feedback loop which integrates state, rate or, say, action and information of a system. They correspond to the three constituents of the system, i.e. unit, motion and information. The change of state variable is determined by the result of decision or action. The feedback loop can be realized through the interaction of system objects. The influence of information is expressed by message of object-oriented approach. Action and information of system dynamics correspond to the behavior and communication of object-oriented approach. So the operators (or methods) of system dynamics can be abstracted as a series of system objects. They consist of *State*, *Rate*, etc.

2.2 Quantitative management methods and system objects

System dynamics method is an approach to study socioeconomic behaviors with time-varying. Quantitative management methods are basically static methods. There is no obvious boundary between dynamic methods and static methods. Static methods can be used in dynamic methods. So we can use some of quantitative management methods in system dynamics method. Quantitative management methods in this system include mathematical programming, networks, operations management, and so on[2].

The mathematical programming methods are the most extensive. These comprise three major classes: linear programming, nonlinear programming, and dynamic programming. The linear programming approaches are most widely used. They address a wide variety of problems such as resource allocation, optical scheduling, network flow analysis, and transportation or routing problems.

Many decision problems can be depicted as a set of junction points interconnected by a series of lines. The junction points may represent geographic locations, physical facilities, or activities. Connecting lines may portray routes, layout designs, or job relationships. Usually, manager wants to find the series of connections that best achieves some specified objective.

Frequently, decision makers must cope with a variety of planning and operations problems that cannot be strictly characterized as mathematical programming, or network situations. Instead, the problems involve unique features that require specialized treatments or tailored solution approaches. So we should use operations management, including inventory, queuing theory, and sequential problems.

In foregoing paragraphs, we set forth several typical quantitative management methods. All of these methods can be coded as system objects. In this system, we call them *Decision Trees*, *Bayesian*, *Utility Analysis*, *Integer Programming*, *Goal Programming*, *Assignment*, *Inventory*, *Queuing* and so on.

Mathematics is an essential method of quantitative decision making for socioeconomic systems. Many mathematical methods were integrated into this system (tool). Mathematical methods as system objects mainly include *Square*, *Expo*, *Sin*, *Sum*, *Max*, *And*, *Nor*, *Delay*, *Smooth*, *Switch*, *Multiply*, ·*Divide*, *Integral*, *Coefficient*, *Correlation*, and so on.

2.3 Building system objects

Properties of system objects include encapsulation, or hiding their internal contents from other system components, composite data and activity specification; and abstraction to support reusability by property inheritance from super classes to sub classes[1]. Reuse in Object-oriented development is achieved either by inheritance and specialization of generic objects or by using more specific objects as building blocks in applications.

When abstracting all kinds of methods mentioned above as system objects, we followed these regulations:

(1)Objects should be modelled as composite specifications of data and activity.

(2)System should be modelled as a network of synchronously communicating objects.

(3)The interface for each objects should be specified as events/message types, it will accept and produce, with their triggering effects on objects methods.

(4)Object oriented models should embody classification and inheritance mechanisms.

(5)Object models should abstract and not contain low level application detail. Reuse should be facilitated by making object model generic for subsequent specialization in new applications.

3.USING OBJECT-ORIENTED APPROACH TO BUILD MODELS

3.1 Organizing the objects

There are a number of criteria to use for the classification and organization of objects and classes of objects. One classification originates by considering how similar the classes of objects are to each other. This is normally the basis of inheritance hierarchy; a class can inherit another class. Another classification can be made by considering which objects work together with which other objects or how an object is a part of another.

3.2 Objects interaction

In order to obtain a picture of how the object fits into the system, we can describe different scenarios or use cases in which the object takes part and communicates with other objects. In this way, we can fully describe the object's surroundings and what the other objects expect from our objects. The object's interface can be decided from these scenarios. We then also consider how certain objects are part of other objects.

3.3 Operations on objects

The object's operations come naturally when we consider an object's interface[4]. The operation can also be identified directly from the application, when we consider what can be done with the items we model. They can be primitive or more complex such as putting together some report of information form several objects. If one obtains very complex information, new objects can be identified from them. Generally, it is better to avoid objects that are too complex.

3.4 Object-oriented model construction

Object-oriented construction means that the model is designed and implemented. It is executed in the target environment. Using object-oriented approach people can develop models through the relationships, behaviors and interactions of system objects[3]. The interaction is expressed by means of system dynamics. System dynamics has the character of relating events in the organization or its environment to possible actions that realize lawful state transitions. The proposed dynamic modeling system enables an extension of the object-oriented paradigm in the possibilities they offer to make a dynamic model of a dynamic reality.

4.THE SYSTEM OR TOOL FOR USERS

For users, this system is a powerful vision tool. The interface consists of windows and menus. According to the contents of windows and menus, users can select system objects, and then build the relationships between them. In this way, the models can be built.

Three steps for model development are applied: (a)Identifying a problem and conceptualizing a system model representing the problems by system objects; (b) developing a causal-loop diagram of system functions and building the relationships and interactions of system objects; (c)running and testing the model. The system (tool) can indicate the unstable factors of developed models.

5.CONCLUSIONS

The characteristic of the system (tool) is saving time, convenient, and practical for users. For a specific problem, users should select different system objects first, and then model the problem through the relationships and interactions of the system objects. In this way, the new model to develop can be created. This system is a powerful tools for developing models. It is a users-oriented and an automation system. This tool is built especially for modeling socioeconomic system. We can still use this idea to build other similar systems for modeling. This system is an open system. The methods or socioeconomic behaviors that not existed in the system can be created through the interactions of the system objects existed in the system.

REFERENCES

[1]Assche F.V., Moulin B., Object Oriented Approach in Information Systems, North-Holland, 1991, IFIP.

[2]Guisseppi A.F., Quantitative Management, The Dryden Press, 1990.

[3]Hu Xiaohui, An Object-Oriented Model Management Approach, In the Proceedings of Qualitative Reasoning and Decision Technologies, Barcelona Spain, 1993.

[4]Jacobson I., Object-Oriented Software Engineering, Addison-Wesley Publishers, 1993.

[5]Wang Qifan, Theory and Application of System Dynamics, New Times Press, 1987.

[6]Wolstenholme, E.F., System Enquiry-- A System Dynamics Approach, Wiley Publishers, 1990.

Teaching Mathematics with Modelling

Dr Judy A.H.Wilkinson,
Department of Electronics and Electrical Engineering,
University of Glasgow.

Abstract: Engineers should be taught mathematical techniques through the exploration of models based on physical systems. Working in teams students exploit commercial software packages to provide solutions to these models directly. They are aware of the numerical methods used but do not write algorithms or perform arduous calculations by hand. The emphasis is on realising and criticising the model and the solutions.

1. Introduction

The body of knowledge required by electronics and electrical engineers is expanding; this means that we should teach subjects in a way that is relevant to current industrial practice. Therefore we believe that in order to teach mathematics to Engineers today we should start with the modelling process and introduce the mathematical techniques by solving the equations arising from the models. This shows the students the reasons for learning the mathematical language necessary to solve engineering problems.

The growth of computing power and the availability of sophisticated software packages enables us to teach students to select a relevant method for solving a problem and realise the scope and limitations of that method.

In our course a new topic is introduced by the discussion of a physical system. After the mathematical equations have been obtained students explore solutions using software packages and then compare them with the practical situation. At the same time the students must have some realisation of the basis of the numerical methods available and this is given using a traditional approach with examples solved using paper and pencil or a simple calculator.

First students are taught how to set up the model and understand the assumptions on which it is based; then to use an exploratory approach rather than proceeding by rote, to methods of solution and finally to appreciate the limitations of the methods. Students need to know the methods that are available but how far they must be able to produce computer code is questionable. However they do need to be able to critically assess the answers they obtain using commercial packages, discuss why the programmers have used these methods and evaluate the different approaches.

2. Context

The class is composed of second year students, most of whom are 18 to 19 years old. They have spent one quarter of the first year on Mathematics and have acquired basic skills in numerical calculations, graph plotting and algebra with an introduction to differential equations. This course replaces a numerical methods course that was taught in a Mathematical rather than an Engineering context. The students are expected to spend about 120 hours on the material; 40 hours is spent with the lecturer present, half on blackboard discussions and half around the computers, 40 hours using computers with minimal support and 40 hours reading the lecture notes and text book references.

The class size is about 36 students and 18 p.c's are available with MATHCAD software installed. Each p.c. is connected to a printer and a copy of the manual is chained to the desk. This set-up allows the students to explore solutions to equations using MATHCAD and present their results in the appropriate form either as a table or as a graph. The ease of access to software and printing facilities encourages this exploration and gives students confidence in presenting and interpreting their results in a professional manner.

3. Teaching Technique

The syllabus is divided into topics each of which is covered in two, one hour sessions. Before the session the students have lecture notes containing references to other texts and precise instructions on which section should be read and what background information they should acquire before the class. Usually the topic is introduced with a physical system which is made manifest. For example a large pendulum is brought in, two coupled springs are demonstrated or a filter circuit is displayed on an oscilloscope. This physical realisation of the model is vital in providing the students with concrete images that relate to the abstract mathematics; at this stage in their development as Engineers it reminds them of the physical world they are attempting to model. By questioning they are led to the bare bones of the problem and they learn how the assumptions can be refined to make models of different levels of complexity.

A few minutes is often spent at the beginning of a lecture with the technician or craftsman who has made the physical system explaining what information they needed in order to build it to a given specification. For instance to obtain a given frequency of swing from the pendulum it is necessary to adjust the length of the rod. How this is done depends on the accuracy which is required. This introduces discussion of the size of error expected from the model and assumptions about the tolerances of the components. Meeting the person who has built the circuit or system reinforces the importance of these ideas and relates the discussion to a manufacturing context.

Deliberate gaps have been left in the students lecture notes where answers to questions can be inserted following guided discussions in class. This is an important technique to ensure that the students are participating in the lecture rather than passively acquiring information.

Once the topic has been introduced and the equations built up the students go off to a computer to work on an investigation of the model.

4. Assessment

The form of the assessment dictates what students do; thus its structure affects the learning process. We believe that group learning can be very effective in teaching modelling. Students help each other and gain confidence in tackling the software packages in a group situation. They are allowed to chose with whom they work. All members of the group sign the assignments stating they have participated in it and agree with the results.

Each topic is assessed by a tutorial and an investigation. The tutorial consists of " back-of -the-envelope" calculations which can be done by hand or using very simple calculators. The investigation is a more open ended exploration of the model using software packages. The students work in groups of four on all these assessments.

Two case studies are carried out over the Christmas and Easter period and the first two weeks of the second and third terms These are based on the previous investigations or practical work but are more open ended with students modelling a system as far as they wish. The case studies can be undertaken as an individual exercise or in a group. Students are given specific guidelines and recommended texts but encouraged to make their own assumptions and specific models.

The mark for the course is an average of the marks from the group assignments and case studies but in order to ensure that all students acquire a modicum of technical mathematical ability they have to get 90% or over in two class tests. They can sit the tests (obviously with the same format but using different examples) at weekly intervals in the second and third terms until they achieve the required standard.

5. Course Design

The choice of physical systems which the students will investigate is extremely important for a modelling course and we are building up a library of suitable models. This is not easy because we need physical systems that can be demonstrated in the class room and are of direct interest to an electrical or electronic engineer is required. Figure 1. illustrates the method by which the topics were selected. The lecturer decides that a mathematical technique is necessary for an electrical or electronic engineer in the second year of their course and seeks a suitable physical system, the mathematical model of which can be solved by the specific technique.

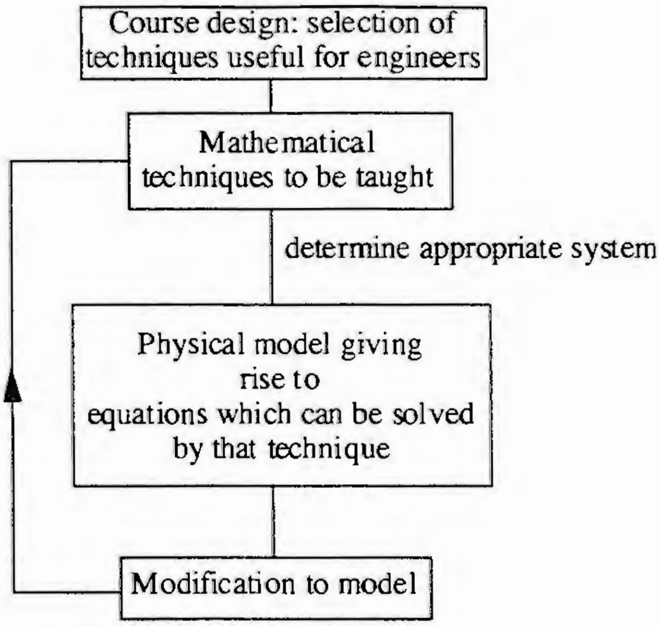

Course design: selection of techniques useful for engineers
Mathematical techniques to be taught
determine appropriate system
Physical model giving rise to equations which can be solved by that technique
Modification to model

Figure 1: Logical path of lecturer in designing course

The students path is different for they must model the system and in doing so discover the techniques to solve the equations but this is placed in a richer context that the primary concerns of the lecturer.

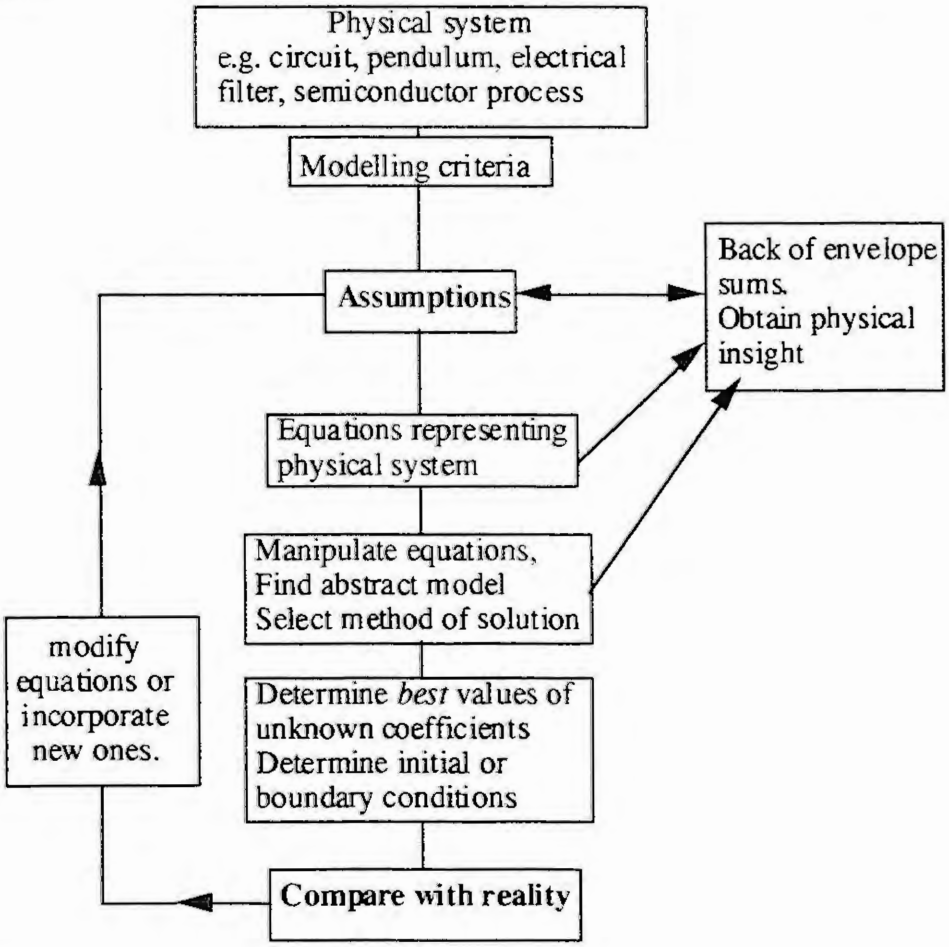

Physical system
e.g. circuit, pendulum, electrical filter, semiconductor process
Modelling criteria
Assumptions
Back of envelope sums.
Obtain physical insight
Equations representing physical system
Manipulate equations,
Find abstract model
Select method of solution
modify equations or incorporate new ones.
Determine *best* values of unknown coefficients
Determine initial or boundary conditions
Compare with reality

Figure 2: Student's path

6 Examples of Physical Systems used

The physical systems used at the beginning of the course are basic, fundamental models that have wide applications in Engineering such as the pendulum or two coupled masses on a spring. The pendulum is used to introduce second order differential equations with linear and then non-linear terms. Adding loss terms and then allowing forced oscillations gives rise to a variety of mathematical techniques for solution and a range of physical interpretation of the results.[1]. The natural progression through the model highlights the different methods of solution from complementary functions, Laplace transforms to the Runge Kutta method for solving non-linear equations. The analysis using computer software reinforces the ideas of frequency, critical damping and decay time, phase plane plots and resonance.

The two coupled masses on a spring is used to introduce matrices, eigenvalues and eigenvectors. The relationship of the results of the calculations to the motion of the masses on the spring introduce important concepts for second year students. They should have a physical understanding of eigenvalues and eigenvectors before they acquire the mathematical techniques needed to solve complicated matrix equations.

Another example of physical systems are electrical circuits and filters which relate to their practical electronics course.

For example in the filters topic we start from a practical circuit but then synthesise mathematics and circuit design to consider such aspects as

i) The relation between the shape of a function and its physical properties
ii) The scaling of quantities
iii) The normalisation of a physical model to see basic underlying features.
iv) The insight gained from different scales in graphing functions, in particular linear and log plots.
v) Curve sketching and approximations to functions.
vi) Complex algebraic manipulations.

before returning to the model of practical filter circuits and comparing the data from the design with the practical implementations.

This is only a second year course so we are looking at the frequency response for a stable system but the aspects introduced above are very important for engineers to grasp. Again we are relating the abstract mathematical forms to concrete examples.

Other models which engineers should be familiar with are the lsing spin model which can describe hysteresis or neural networks, harmonic oscillators as a model for the wave representation of matter, Markov chains for reliability.

7. Results

The quality of work from students working in groups and using computer packages is extremely high. They present their work well and clearly which we find is lacking in hand written solutions. They are prepared to annotate the output far more than in a written mathematical submission. By enabling them to explore a model using fairly powerful packages it is possible to check that they understand the basic concepts and can relate them to the physical system. The mathematics becomes a tool but the students become craftsmen and can select and use the appropriate tool.

By starting from the physical system and then building up a model we relate the abstraction of the mathematics which many students find difficult to concrete phenomena at the beginning. We do not start from the mathematics and then give the model as a hypothetical example at the end of the topic but we start from the physical system and ask questions from the beginning, introducing the mathematical techniques as needed.

We use the strengths of engineers- the ability to work in teams to solve problems and the development of lateral thinking to illuminate and enhance the approach to the mathematics.

8. Acknowledgements

The students in the second year Mathematical Modelling course in the Department of Electronics and Electrical Engineering at the University of Glasgow.

9. References.

[1] R.Abraham and C.D.Shaw, Dynamics: the geometry of behaviour Addison Wesley, 1993.

Simulation By Active Fuzzy Petri Nets

He Xingui
Beijing Institute of System Engineering
P. O. Box 9702-19, 100101, Beijing, China

Abstract

In the recent years, fuzzy technology [1]has been successfully applied into many areas, such as process control, diagnosis, evaluation, decision making and scheduling, especially, in simulation where accurate mathematical models can not or very hard be established. In this paper, the concepts of fuzzy Petri nets and active fuzzy Petri nets will first be presented, which are quite suitable for modeling the concurrent systems with fuzzy behavior. Then, the active simulation will be introduced, in which the simulation model described by an active fuzzy Petri net not only can show its fuzzy behavior, but also has the ability actively triggering some very useful actions , such as automatic warning, real-time monitoring, simulation result checking, self-adapting, error recovery, simulating path tracing, state inspecting and exception handling, by a unified approach while some specified events occur. As a powerful simulation tool, an active fuzzy Petri net is concurrently driven by a network interpreter and an event monitor.

Keywords: Simulation, Fuzzy, Active, Petri net

1 Introduction

Incompleteness, uncertainty and parallelism almost are three general properties of the real world. Concerned about uncertainty, randomness has been discussed thoroughly in probability and statistics, but we know fuzziness much less than that we know about randomness. In the recent years, fuzzy technology based on fuzzy mathematics has been successfully applied in many areas, such as process control, diagnosis, evaluation, decision making and scheduling, especially, in simulation where accurate mathematical models can not or very hard be established. Ideally, simulation of the real world should have a set of suitable tools for describing, testing and running its model conveniently, including being able to reflect its Incompleteness, uncertainty and parallelism. In this paper, we are planning to solve some of the problems. Besides, a concept of active simulation will be introduced, in which the simulation model not only can show its fuzzy behavior, but also has a certain ability which can actively trigger some very useful actions by a unified approach while some specified events occur. Some simulations about production activities and economical plans will be presented as examples. The fuzzy approach is able to be used in the real-time area.

2 Fuzzy Petri Nets

In Petri nets, mainly there are two kinds of nodes, transition nodes and place nodes, represented by bars and circles respectively. There are two kinds of lines connecting from transition nodes to place nodes or from place nodes to transition nodes. Petri nets are executable. At the beginning of execution, a Petri net is initiated so that some place nodes in the net are set some tokens represented by points in the node. During execution, the transition nodes are checked to see whether all their input lines have already been connected to those place nodes with at least one token, if they are, then the transition nodes are fired. The result of firing is to delete one token from each of their input place nodes, and to add one token to each of their output place nodes with a line from the transition node. A Petri net is executed so step by step continuously changing the tokens in its place nodes.

Fuzzy Petri nets are generalization of Petri nets using a fuzzy approach. A fuzzy Petri net is composed of two kinds of fuzzy nodes: fuzzy transition nodes and fuzzy place nodes represented respectively by the following two diagrams(figure 1.):

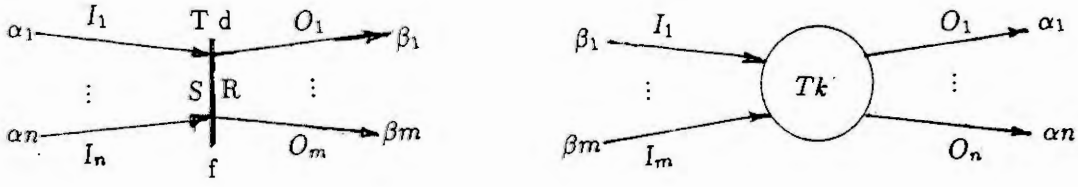


figure 1.

where α_k and β_j represent fuzzy degrees of the input lines and output lines respectively with their values in $[0, 1]$;

I_k and O_j represent allowed maximum input and output on the input lines and output lines respectively with their values in $[0, \infty)$;

T is the threshold of the transition node, which controls state transition of the node;

f is the state transition function of the transition node, which is a monotonous increasing function defined on input strength $S(i_k, \alpha_k)$ of the input lines, where i_k is the allowed maximum input I_k when the token in the corresponding input place node is greater than or equal to the allowed maximum input I_k, otherwise, it is the token in the corresponding input place node, $S(i_k, \alpha_k)$ are some non negative monotonous increasing functions;

d is a non negative number which represents a delay of the transition node from getting the inputs to sending out the outputs.

Tk is the token of the place node represented by a non negative real number.

In fuzzy Petri nets, transition nodes are allowed having input lines or output lines only from/to place nodes and place nodes are allowed having input lines or output lines only from/to transition nodes. Fuzzy Petri nets are also executable. At the beginning of execution, a fuzzy Petri net is initiated so that some of its place nodes are set certain non negative real numbers as their tokens. During execution, the fuzzy transition nodes are checked to see whether their current state transition function values are greater than or equal to their corresponding thresholds, if they are, then those transition nodes are fired. The result of firing is to subtract each input i_k of the fuzzy transition node from the tokens of its corresponding input place nodes, and after a delay d, to add corresponding output strength $R(O_j, \beta_j)$ of the fuzzy transition node to each token of its output place nodes, where the output strength $R(O_j, \beta_j)$ is a non negative monotonous increasing functions defined on the allowed maximum output O_j and the fuzzy degree of the output line β_j. A fuzzy Petri net is executed so step by step continuously changing the tokens of their fuzzy place nodes.

Strength computation functions $S(i_k, \alpha_k)$ and $R(O_j, \beta_j)$ can be defined differently in different applications. For example, they can be defined as $\min(i_k, \alpha_k)$ and $\min(O_j, \beta_j)$ or $i_k * \alpha_k$ and $O_j * \beta_j$.

Obviously, Petri nets are the special cases of fuzzy Petri nets. In order to prove that, it is just needed that let fuzzy degrees α_k and β_j be I or 0 (In fact, 0 means no connection), let the threshold T be 1, the fuzzy place nodes can only assume non negative integers as their token values, and let both functions f, S and R be minimum functions.

Fuzzy Petri nets can be widely applied in simulation of many physical, economical, even social systems. It is quite suitable for describing and analyzing various large complicated concurrent systems. Now, we are going to give an example to explain their applications.

A modern enterprise usually is composed of many sub-enterprises or factories. It can be considered as a network composed of these sub-enterprises and factories according to "supply and demand" relationship, in which, one factory maybe needs other factories to supply raw materials, and on the other hand, also to supply its products to some other factories. During production, many complicated materials and products flows are running concurrently inside the whole enterprise. It is easy to see that a fuzzy Petri net could be used to describe the dynamic behavior of the enterprise. Here, a fuzzy transition node can be used to represent the production department or manufacturing division, and a fuzzy place node can be used to represent the warehouse or depository of materials or products of the factories, where the tokens of the place nodes represent the numbers of materials or products. The "supply and demand" relationship among the factories could described by input or output lines, where the fuzzy degrees of the connecting lines might be used to represent the proportion of materials or products damaged from the starting points to the ending points of the lines. Accordingly, the function S and R can be defined as the product of two arguments, and the thresholds of the transition nodes are used to represent the needed minimal materials for manufacturing their products, and accordingly the state transition function f can be defined as the minimum function. The inputs and outputs of the nodes represent the inputting materials and outputting products respectively. Thus, we have established a simulation model for the manufacturing process of an enterprise.

Using this simulation model, we can do many meaningful things, such as testing the situation of overstock or shortage about materials or products during manufacturing in different distribution schedules of raw materials, and answering the questions such as "where the bottleneck of manufacturing is ?", "how to adjust the production schedule ?" and "how to improve the original distribution of raw materials". In fact, the fuzzy Petri nets can be used to simulate many large systems which connect many nodes together by a certain relationship relevant to "supply and demand" or "producing and consuming", such as planing and scheduling systems on traffic networks, resource distribution systems, even economy macroscopic adjustment and control systems.

3 Fuzzy Active Petri Net and active simulation

A fuzzy Petri net, like a computer programs, can be executed by an interpreter. Once a fuzzy Petri net is established and initiated, the behavior of the net is fixed. It can not do any additional actions needed according to varying states or situation of the net, such as automatic warning, real-time monitoring, simulation result checking, self-adapting, error recovery, simulating path tracing, state inspecting and exception handling. However, these functions are so important and useful for almost all the practical systems. To meet these demands, we are going to present a kind of network named "Fuzzy Active Petri Net", which combines the ideas about fuzzy Petri nets and active databases. In an active fuzzy Petri net, besides a fuzzy Petri net which can be passively executed by an interpreter as mentioned last section, there is another rule base composed of a set of rules which can be actively fired by an event monitor according to some events occurred currently. So, the simulation model using active fuzzy Petri nets not only can represent various complex objects with fuzzy behavior, but also can actively perform some specified actions according to a condition when a certain event occurs. This would make the simulation able automatically to trigger many very useful functions during simulation by a unified approach.

A rule in the rule base has the following unified form:

```
RULE <rule name> [(<parameter>,...)]
PRIORITY <priority>
WHEN <event expression>
IF <fuzzy logical expression> THRESHOLD <threshold>
THEN <action>,...ELSE <action>,...
END RULE
```

Where `<rule name>` identifies the rule; `<parameter>` is optional, if any, it may occur in the following event expressions or fuzzy logical expressions. Usually, the parameters can be the states of the nodes of fuzzy Petri nets, for example, the time a transition node fired, the token numbers of some place nodes, the duration between two times of firing of a transition node or two transition nodes, the duration between two times of changing of a place node, or frequency or number of firing of a transition node, etc. . Through these parameters, the rule base is related with the active fuzzy Petri net, so the behavior of the active fuzzy Petri net can be monitored by an event monitor, and actively trigger some needed actions;

`<event expression>` is an expression of some more basic events presented in [2], which is used to describe various events possibly to occur during simulation time;

`<fuzzy logical expression>` is a well-formed logical formula in a certain fuzzy logic [1] which usually has truth values in $[0,1]$;

`<threshold>` is a number in $[0,1]$;

`<action>` represents an action needed to be performed when an event specified by `<event expression>` occurs, which could be any function mentioned above. When the current truth value of `<fuzzy logical`

expression> is greater than or equal to <threshold>, the actions following THEN are performed, otherwise the actions following ELSE are performed, where the actions in the action sequence are performed from left to right;

<priority> is a number in [0,1], which represents the priority of the actions in the rule. It is used to determine which actions should be performed first when there are other actions belonging to other rules needed to be executed.

An active fuzzy Petri net is concurrently driven by a network interpreter and an event monitor. The network interpreter has been described in last section. The event monitor continuously checks the rules in the rule base to see whether certain events in some rules already have occurred, if they have, then to fire the rules and perform the actions in the rules according to current truth value of their fuzzy logical expressions. It can be implemented by software or hardware. That means, the event monitor can be implemented as a process under the control of the operating system with high priority which makes the monitor executed constantly. In a multi-processor environment, the monitor can be implemented on a specially dedicated processor. The event monitor also can be implemented under the help of some special hardware, especially in the case where the events are some interrupting signals. It can make some rules fired directly when corresponding events occur. It would get more efficiency but higher cost.

Obviously, the active fuzzy Petri nets can be used for simulation with some fascinating active functions in many areas. For example, in order to control the execution of a plan, usually, a PERT diagram is used, from which we can see whether the plan has been delayed or advanced. As we know, a PERT diagram can only represent a static plan, and has no any active behavior. However, an active fuzzy Petri net does. It can be used to describe the execution procedure of a plan dynamically. Furthermore, an active fuzzy Petri net has combined a fuzzy Petri net with an event monitor, it makes us able to simulate the execution of the plan vividly and to trigger various needed actions automatically when some events occur, for example, when a certain crucial task has been delayed, the materials or products in some storehouses have been piled too much, a factory has been lacking of materials for manufacturing for two days, etc. . According to [2], some atomic events can be used to compose an event expression which can express a very complicated and meaningful event.

4 Conclusion

In the paper, we have presented two networks, fuzzy Petri nets and active fuzzy Petri nets, which can represent much more complicated systems with fuzzy behavior than the Petri nets can. Especially, they are very suitable to be used for simulating those concurrent systems with "supply and demand" or "producing and consuming" relationship between components of the systems. Furthermore, the active fuzzy Petri net combines the fuzzy Petri net with the active database technology. A system represented by an active fuzzy Petri net can be simulated vividly and trigger various needed actions automatically when some specified events occur. It makes the simulation process able actively to conduct various remarkable actions, such as automatic warning, real-time monitoring, result checking, self-adapting, error recovery, simulating path tracing, state inspecting and exception handling. Here, particularly we want to emphasize the self-adapting ability, which can adjust the parameters of the active fuzzy Petri net, such as the thresholds of the transition nodes, the fuzzy degrees of the input/output lines of the nodes, the delay of the transition node from getting the inputs to sending out the outputs and allowed maximum inputs and outputs on the input lines and output lines, according to running situation of the network. Thus, the network will evolve with the simulation process going on and become a changeable network. Therefore, it is possible to improve the network through simulation processing itself. Now, in Beijing Institute of System Engineering a fuzzy Petri net interpreter and an event monitor have been established in a PC/DOS environment.

References

[1] He Xingui, Fuzzy theories and fuzzy technologies in knowledge processing, Defense industrial publishing house, Beijing, 1993.

[2] He xingui, Fuzzy active knowledge base systems, Proceedings of CJCAI 92, pp. 89-96, 1992.

FMS RELIABILITY MODELING AND ANALYSIS USING COLORED GENERALIZED STOCHASTIC PETRI NETS

Dr. Yushun Fan Professor Cheng Wu
State CIMS/ERC, Dept. of Automation
Tsinghua University, Beijing 100084
P. R. CHINA

ABSTRACT

In this paper, an index---reliable productivity is put forward to describe the FMS reliability. We also presented an FMS reliability modeling method using Colored Generalized Stochastic Petri Nets. An model is built for a manufacturing system.

INTRODUCTION

FMS is a capital invest intense and high complex system. In the operation period, there are many faults happened in the FMS. Besides the faults caused by hardwares and softwares, there are also some human interference errors. The faults have serious effect on the efficiency of the FMS. Though, there are tons of papers about the reliability analysis work, there lacks a good method of modeling and analysis the reliability of FMS. Considering the significant difference between FMS and conventional continuous time system, the Mean Time To Failure index which is generally used in continuous time system can not reflect the reliability of FMS.

The problems which made FMS reliability analysis difficulty are as follows:
1). The meanings of reliability is quite different with conventional continuous time system.
2). The main purpose of FMS is to improve productivity, while any kind of fault will certainly affect the productivity of FMS, so, it is necessary that an FMS reliability index should reflect the effect of any kind of faults happened in the system.
3). The main advantages of FMS is the flexibility, the routing and manufacturing flexibility have important effect on FMS productivity and hence reliability. While the flexibility of routing and manufacturing is also an important basis for the scheduling system to impose control to the FMS, hence it must be seen that the structure of the FMS is a dynamic structure, an model used to evaluate the reliability of FMS should be an dynamic model.
4) In the FMS, there are many human interference, the faults caused by the human operation errors are transfered in the system through the movement of workpieces and tools.

Considering the above mentioned factors, an FMS reliability model built should take into consideration of the various distinct properties of FMS. In this paper, we put forward a new index--reliable productivity to describe the reliability of FMS. An Colored Generalized Stochastic Petri Nets (CGSPN) method is used to model and analyze the FMS reliability.

Definition 1. The FMS reliable productivity R_p is the probability of the FMS to complete its ideal ptoductivity, ideal means there are no any kind of faults in the FMS.

$$R_p = R_o / R_i \quad .$$

where
R_o represents productivity of FMS under the affect of faults.
R_i represents ideal productivity of FMS without the affect of faults.

FMS RELIABILITY MODELING

Miriyala and Viswandham[1] used Process Spanning Graph method to evaluate the reliability of FMS. Two reliability indexes ----Part Reliability and System Reliability are given and evaluated. The two indexes can reflect the reliability of FMS to some extent. Windahl and Winkelhake[2] have reported the availability analysis results after the survey on 40 assembly lines. They also gave some methods to increase the availability of the assembly lines.

In this section, we will present an FMS reliability modeling and analysis method using CGSPN. The CGSPN method for FMS modeling is using modular approach. The modules (subnets) for the machining centers, buffers, load/unload stations, and automatic guided vehicle(AGV) are similar to that of the normally used as in [3], but we made some modifications for load/unload stations and machining centers in order to fit for the purpose of reliability modeling of FMS.

The load/unload stations model is shown in Fig. 1. In which, the human operation error is added by using a random switch. The probability h refers to the correct operation probability, and 1-h refers the error probability. A color RED will be assigned to the token in place p_1 if the error occurs, then the RED color is gone with the token which represents the type of workpiece to place p_4 and so on. If the RED color token reaches the machining center and been machined, then the machine will break down. This is what we said the human error transfer in the system with the movement of workpieces. It is the same situation for the human operation errors happened to the tools. The random switch before place p_2 represents the mix of the product. Different switch is on will send a token with different colors(represents different workpieces) to the place p_2. Place p_4 represents the workpiece is ready for loading. the transition t_6 represents the fixture time for the workpiece. p_5 represents workpieces is ready to be machined, p_6 represents the unloading station, transition t_7 represents the time needed to unload the workpiece. place p_3 denotes the empty palletes.

The machining centers model are represented by Fig.2. In which, p_7 represents workpieces to be machined. p_9 represents the workpiece is being machined, p_8 represents the idle state of the machining centers, id1, id2 are two different colors represents machine 1 and machine 2. We can also add id3, id4,.. if want to use more machines. p_{10} represents .machine is broken down and needs to be repaired, the time distribution for the machine break down is denoted by the firing ratio of t_{10} . The selection between transition t_{11} and t_{12} means whether the workpiece is damaged by the machine breakdown. The damaged workpiece will go out the system through transition t_{30}, so it will not be added to the product count. The dot line block represents the transfer of the workpiece from one machining center to another. The place P_A is defined on Fig.4. It represents the AGV module.

We can now use the above modules to built the FMS reliability model. Consider a simple manufacturing system shown in Fig.3. Where, M1, M2 represents two machining centers, each one has one input buffer and output buffer. LU represents the load/unload station which has two working plateforms. AGV represents the automatic guided vehicle, BUFFERS represents common buffers. We assume two kinds of workpieces will bw manufactured. Workpiece 1 can be machined either or machine 1 or machine 2 by one time. Workpiece 2 should first be machined by machine 1 and then on machine 2. The product mix is q:1-q. The human operation error probability is 1-h. The other parameters can also be assigned, but we did not need it now.

Fig. 4 shows the CGSPN model built for the manufacturing system Fig.3. In which, places p_1 , p_2 , p_3 , p_4 , p_5 , p_6 , p_7 , p_8 , p_9 , p_{10} , p_{11} , p_{12} , p_A , and t_1 , t_2 , t_3 , t_4 , t_5 , t_6 , t_7 , t_8 , t_9 , t_{10} , t_{11} , t_{12} , t_{13} have the same meaning as stated above. place p represents common buffers. Now we give the colors set used in the model.

C1=(RED,NORMAL), RED---human operation error, NORMAL--human operation correct.

C2=(1,2) 1--workpiece 1, 2---workpiece 2.

C3=(1RED,2RED), 1RED--wrong workpiece 1, 2RED--wrong workpiece 2.

C4=(1f,2f,2h1), 1f--finished workpiece 1, 2f--finished workpiece 2, 2h1--workpiece 2 has finished first machining.

C5=(id1,id2), id1--machining center 1 is idle, id2--machining center 2 is idle.

C6=(m1p1,m1p2,m1p1RED,m1p2RED,m2p1,m2p2,m2p1RED), m1p1--workpiece 1 on machine 1, m1p2--workpiece 2 on machine 1, m1p1RED--wrong workpiece 1 on machine 1, m1p2RED --wrong workpiece 2 on machine 1, m2p1--workpiece 1 on machine 2, m2p2---workpiece 2 on machine 2, m2p1RED--wrong workpiece 1 on machine 2.

ε ---normal token, no character.

μ ---finished workpiece waiting to be unloaded.

null--no token.
null1--no token in machine 1's relevant place.
null2--no token in machine 2's relevant place.

Then the places in Fig.4 have the following colors which are combinations of the above color set.
$C(p_1)=\{C1\}$, $C(p_2)=\{C2\}$, $C(p_3)=\{\varepsilon\}$, $C(p_4)=\{C2,C3\}$, $C(p_5)=\{C2,C3\}$, $C(p_6)=\{C2\}$, $C(p_7)=\{C6\}$,
$C(p_8)=\{C5,null1,null2\}$,$C(p_9)=\{C6\}$,$C(p_{10})=\{C6\}$, $C(p_{11})=\{C6\}$,$C(p_{12})=\{C6,null1,null2\}$,
$C(p_{13})=\{\varepsilon,C2,C3,C4\}$ $C(p_{14})=\{\varepsilon\}$, $C(p_{15})=\{C2,C3,C4\}$,$C(p_{16})=\{C2,C3,C4\}$.

When the time distribution in the model belongs to the exponential distribution, then the index of R_o and R_i for the above CGSPN model (Fig.4) can be calculated by adopting the method of solving Markov Chain State Equations[4] with only the modification in the generation of the reachability tree and the consequent Markov State Equation. If the time distribution does not belong to the exponential distrbution, then simulation method should be used to get the evaluation results about the reliability index R_p .

CONCLUSION

In this paper, we just present the method of applying Petri nets in the FMS reliability modeling and analysis. It may be interestering to study the following problems from the FMS reliability point of view. First, since there will have faults during the operation of FMS, whether the proposed method can be applied to the justification of the scheduling systems. The second problem may be the application of our method in the design phase of FMS in order to determine the requirements on the hardwares and the detection devices, so as to improve the efficiency of the FMS. The last problem may be the application of the proposed method in the reliability optimization, such as to find the optimal maintainence strategy.

REFERENCES

[1] Miriyala,K., N. Viswandham, "Reliability analysis of FMS", Int. J. FMS, Vol.2, 1989, pp145-162
[2] Wiendahl, H. P., U. Winkelhake, "Strategy for availability improvement", Int. J. Advanced Manufacturing Technology, Vol.1, No.4, 1986, pp69-78.
[3] Al-Jaar, R. Y., A. A. Desrochers, "Performance evaluation of automated manufacturing systems using generalized stochastic Petri nets", IEEE Trans. on Robotics and Automation, Vol.6, No.6, 1990, pp621-639
[4] Marsan, A. , et al, "A class of generalized stochastic Petri nets for the performance evaluation of multiprocessor systems", ACM Trans. on Computer systems, Vol.2, No.2, 1982, pp44-53.

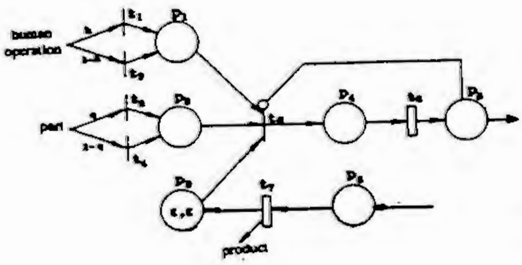


Fig.1 load/unload station module

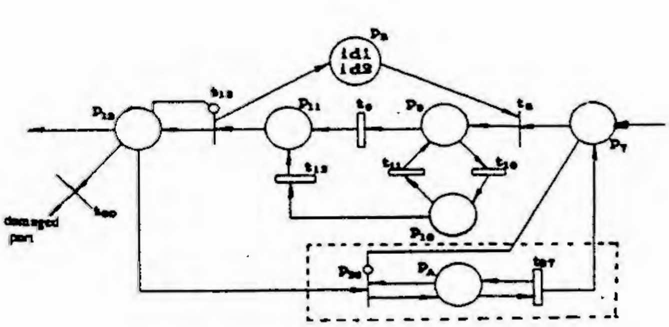


Fig.2 machining centers module

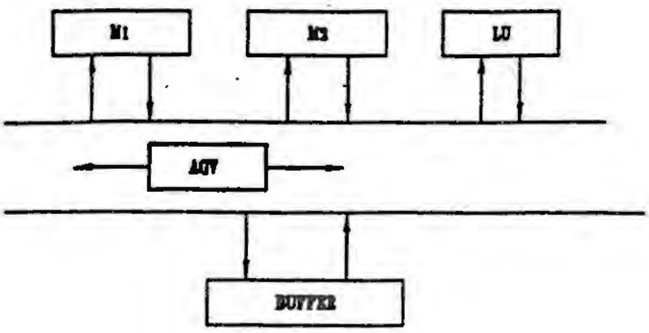


Fig.3 A simple manufacturing system

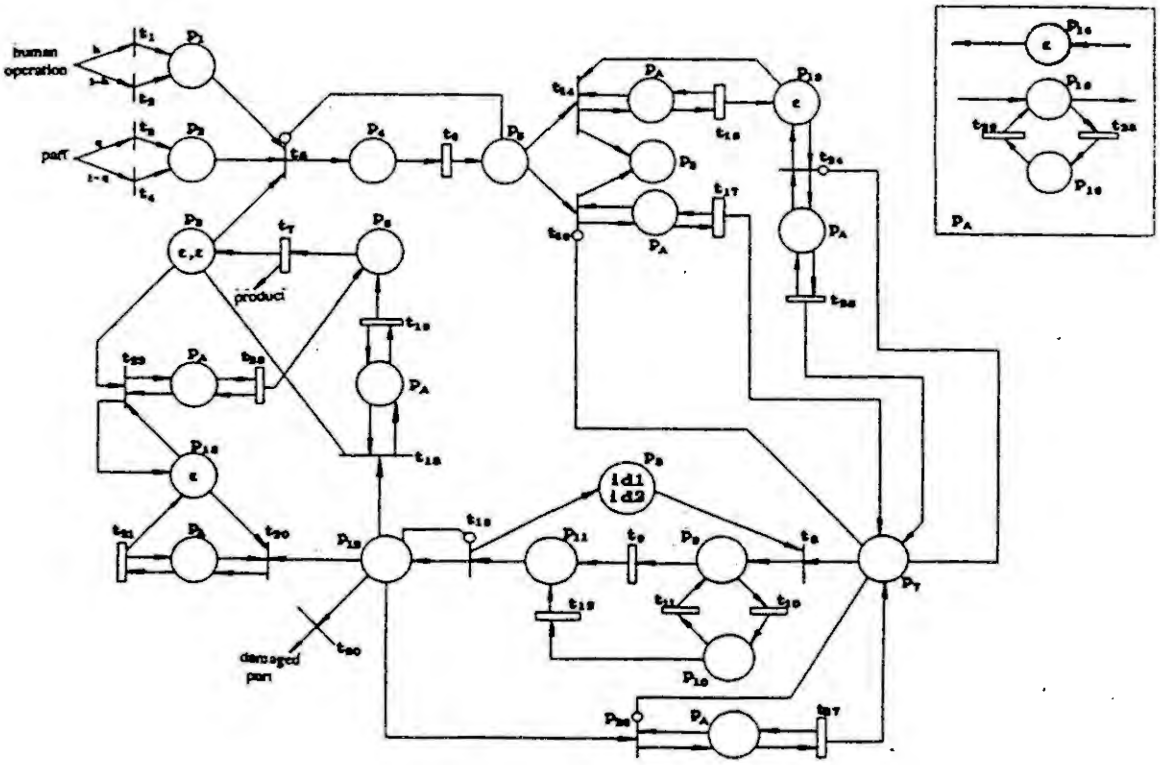


Fig.4 CGSPN model of the manufactruring system

A Modelling Method for Parallel Simulation

Xiaofeng Liu Bo Hu Li
Beijing Institute of Computer Application
and Simulation Technology
P.O.Box 3929, Beijing 100854, P.R.China

Abstract

In this paper, an efficient simulation modelling method for parallel simulation (SMPS) is presented suited for the parallel multiprocessor system and the problem oriented ICSL-2 continuous system sumulation language[1].At first the data relational graph representing any inherent parallelism, data dependence and communication among the simulation tasks of a simulation model, which has been implemented, is contructed automatically by analysing the simulation model writen in the problem-oriented ICSL- 2 continuous system simulation language. Second an improved heuristic algorithm is presented to get suboptimal parallel simulation model. Finally an improved branch-and-bound algorithm is suggested to optimize the result above to get a optimal parallel simulation model suited for the multiprocessor systems. This algorithm has been proved through emulation on PC 386.

Introduction

As we know, the parallel multiprocessor technology has become an important issue and many parallel algorithms aimed to improve the parallel multiprocessor efficiency and speed up computation speed have been studied.Parallel simulation algorithms for the continuous system simulation can be divided into three kinds according to their parallelism granularities: the basic operator level parallelism, the equation level parallelism and the submodel level parallelism.

In the equation level parallelism, the parallel computation of the right-hand side function and the parallel integration algorithms are considered. The work we proceed belongs to this parallelism. It has the following parts:

.Based on the problem-oriented simulation model represented in ICSL- 2 continuous system simulation language,the data relational graph to denote the relationship among the right-hand side functions,algebraic equations and data communication is constructed automatically.

.An improved heuristic allocation algorithm is presented to obtain the suboptimal parallel simulation model. Finally an improved branch-and-bound algorithm is suggested to optimize the result above.

Definitions

In order to set up a framework for our discussions, we first introduce a set of definitions which are used in the description and analysis of the algorithm.

1. The priority of task i, denoted as level(i), is recursively defined as follows:

a. level(i) = Weight(i), if task i has no subtasks

b. $\text{level(i)} = \text{Weight(i)}+\max_{k \in Si} (\text{level(k)}+C(k,i))$, otherwise

where the Si is the set of the subtasks of task i. C(k,i) is the communication time between task i and subtask k. Weight(i) is the execution time of task i.

2. The subtask of a task is referred to the task which musted be scheduled after this task.

3. The schedule upper bound:the maximum time when the last task on each processor

is completed.

4. Feasible schedule:If all the tasks on the processors have been scheduled, the schedule is called a feasible schedule.

5. Speedup is defined as the ratio of the run time on the single processor to the run time on the multiprocessor.

6. Parallel efficiency is the ratio of the speedup to tbe number of the processors.

7. System efficiency is the ratio of the sum of all task's execution time to the sum of tbe end time of the last task on each processor.

8. Computation length is the maxinum of the end time of the last task on each processor.

SMPS algorithm and its implementation

The vector form of the dynamic model of a continuous system can be given below:

$$\vec{X}' = \vec{F}\,(t, \vec{X}, \vec{U}) \qquad (1)$$

$$\vec{Y} = G(\vec{X})$$

where $\vec{X}$ is the state vector,$\vec{U}$ is the input vertor of the system, $\vec{F}$,$\vec{G}$ are the state transfer function, generally $\vec{F}$ is called right-hand side function, $\vec{X}'$ is the derivative of $\vec{X}$ and $\vec{Y}$ is the output vector.

In formula (1), the computations of some right-hand side functions might include the same computation units, such as the table function generators etc, which can be abstracted and represented in algebraic equations. So we can think each right- hand function or each algebraic equation contained in some right-hand side functions as a task to construct the simulation task set of system simulation model. In order to generate the set of simulation task which can run efficiently on the multiprocessor, it is necessary to give a data structure which can represent any inherent parallelism, data dependence and communication among the simulation tasks. So we introduce the data relational graph (DRG). The generation procedure is given as Fig 1.

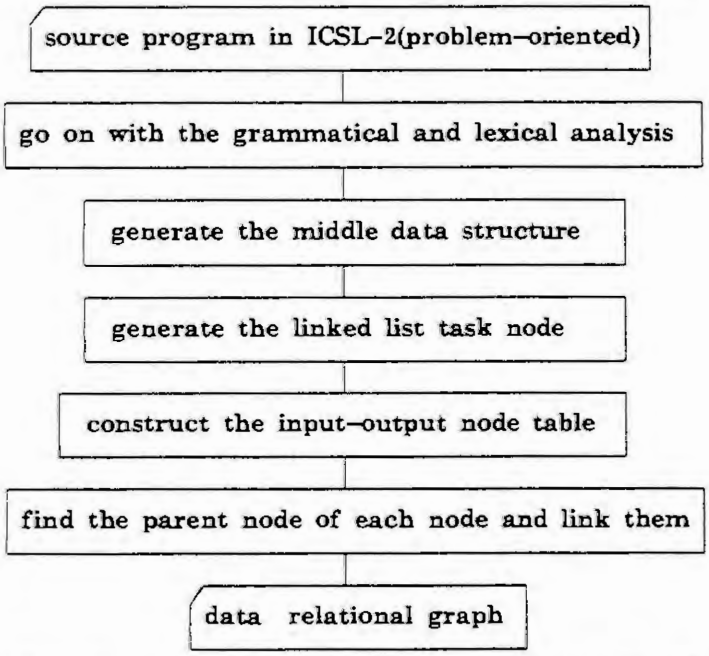

source program in ICSL-2(problem-oriented)
go on with the grammatical and lexical analysis
generate the middle data structure
generate the linked list task node
construct the input-output node table
find the parent node of each node and link them
data relational graph

Fig 1. The flow diagram of the generation of the DRG in SMPS algorithm

As an example, according to the flow diagram above, we can generate the data relational graph of the PHYSBE blood circulation model[2] as Fig 2:

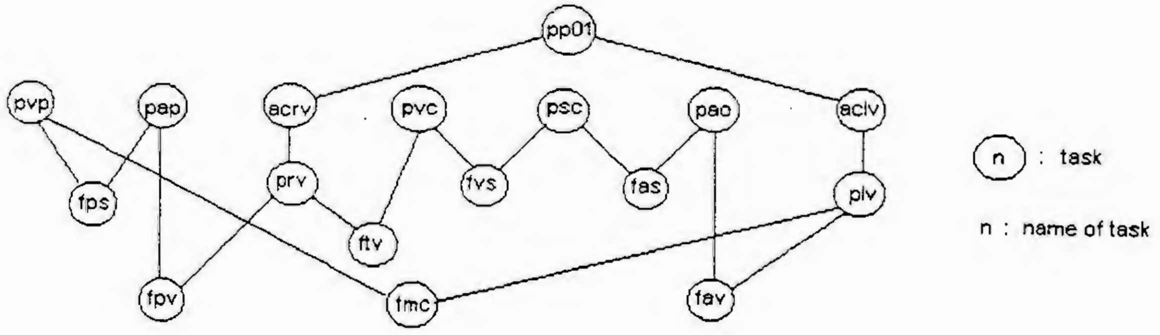


Fig 2. The data relational graph of PHYSBE model

The parallel schedule of the simulation tasks and the allocation of their processors are carried out on the DRG generated. Because of the characteristics that the relationship between the tasks of the continuous system simulation model are determinstic and can be got from the input and output set,the computation time of each task can be preevaluated. As experience shows, the static task allocation strategy is suitable. At first, an reasonable task occupying processor priority is presented in the algorithm(SMPS), in which the data communication time is also considered, to get a suboptimal parallel allocation result.The algorithm is given below:

1. Compute the priority(level(i)) of each task.

2. Determine the priority of the task's occupying processor according to the level(i). The larger the level(i), the higher priority of task i is. When more than one task's level are same, the larger the sum of subtask's level the higher priority is. Otherwise the higher priority is allocated to the task i whose precedent task are more on the idle processor. If any condition is not met, the priority of the task i is determined arbitrarily.

3. Repeat the procedure above to allocate the simulation task on idle processor and compute the schedule upper bound(UBOUND). (See definition 3)

4. Compute the schedule lower bound(LBOUND):

$$\text{LBOUND} = \text{Max}(\sum_{\substack{1,m \\ j\in Si}} (\text{Weight}(i)+C(i,j))/n, \text{Max}_{1.m} (\text{level}(i)))$$

5. If UBOUND = LBOUND, it shows that the optimal schedule has been got, otherwise go on to optimize the result above.

6. The optimization algorithm:

(1) Return to the initial schedule state.

(2) Select arbitrarily a task different from the initial task in the subptimal schedule at the schedule initial point.

(3) Compute the lower bound of this schedule:

LLB = max(LLB1,LLB2,LLB3)

where LLB1 = the maxinum of the end time of the last task on each processor currently.

LLB2 = the mininum of the end time of the last task on each processor currently + the maximum of the level of the task not scheduled.

LLB3 = (the sum of the end time of the last task on each processor currently + the sum of the execution time and the communication time of the tasks not scheduled)/(the number of the processors).

(4) If LLB >=UBOUND return.

(5) If the feasible schedule has been got, compare it with the last optimal schedule and select the shorter schedule length as the current optimal schedule length. Judge whether the schedule is optimal or not. If yes,the algorithm ends, else return to select again.

(6) If the new feasible schedule is not got, recure to schedule.

Case study and experimental results

This algorithm has been proved through emulation on PC 386.We run it with some cases. First the PHYSBE blood circulation model[2] has been selected to illustrate the function and efficiency of our algorithm because this model has been widely discussed in literatures. The runing result is as follow:

Number of CPUs	Computation Length	Parallel Efficiency	System Efficiency	Speedup
1	282	1	1	1
2	147	0.97	1	1.92
3	97	0.96	1	2.91
4	79	0.89	0.90	3.6

We also test our algorithm with a complex model, which includes 213 algebraic equations and 21 one-order ODEs, the result is as follow:

Number of CPUs	Computation Length	Parallel Efficiency	System Efficiency	Speedup
1	7665	1	1	1
2	3833	0.99987	1	1.9997
3	2555	0.9996	1	3
4	1918	0.999	1	3.996
5	1536	0.998	1	4.99
6	1447	0.882	1	5.30

From the cases we can draw our conclusion:

(1) From the examples selected, the number of the processors can reach up to 4-6, which is far higher than the number of the processors fit for the general continuous system parallel integration algorithms.

(2) The speedup obtained becomes larger with the increase of the number of the processor.

(3) The speedup is not infinite, the reason for it lies in the relational constraints among the simulation tasks of the simulation model.

Conclusoin

By runing our algorithm with the cases, we can summarize the advantages of SMPS:

(1) Our algorithm is the problem oriented. It is the extension of the ICSL- 2 continuous system simulation language. It is transparent for the users.

(2) It runs faster than other algorithms and its result is optimal.

(3) It exploits the internal parallelism of the models. If it is used with the existed parallel integration algorithms, the result will be more effiecient.

(4) It is implemented in ANSI C, the transplantation is very easy.

We are now developing a new version which can run on the minisuper computer developed by BICAST.

Reference

[1] Song Xin,Bo Hu Li,The Proceedings of the 1991 Summer Computer Simulation Conference

[2] ACSL — Advanced Continuous Simulation Language, Mitchell and Ganthier Associates Inc. 1975

A HIERARCHICAL MATHEMATICAL MODEL FOR THE CONTROL OF FLEXIBLE MANUFACTURING SYSTEMS

Olivier De Smet and Hisham Abou-Kandil

Ecole Normale Supérieure de Cachan - LURPA, 61 Av. du Président Wilson,
94235 Cachan Cedex - France. email: de_smet@lurpa.ens-cachan.fr

Abstract. The purpose of this paper is to give a mathematical model for flexible manufacturing systems (FMS). Such a model should take into account machine failures and should remain relatively tractable while tackling the control design problem.

1 INTRODUCTION

Recently, there has been a great deal of interest for the problem of production control in manufacturing systems with failure prone machines [KG83] [AKS90] [AK86] [Sha88]. For the modeling of such a system, discrete events dynamic systems (DEDS) are considered here. Two main approaches are often used to model FMS with DEDS: stochastic queuing networks with finite state Markov processes [Ho87] and flow models [MG88].

When using stochastic queuing network, such a DEDS consists of jobs (parts) and resources (machines, operators, ...). Jobs travel from resources demanding and competing for services. The dynamic of the systems is determined by interactions of timing of various discrete events associated with the jobs and resources. With this approach, jobs occupy the resources for a random/deterministic period of time. To model DEDS analytically, the most used mathematical tool is the *finite state markov process.* It is assumed that the state, input and output sets are finite. The transition state and output map can be modeled by matrix of transition probability. But the number of states in a typical DEDS can be combinatorially large. This formulation is unfortunately only useful when used for evaluating transfer lines over a long horizon to obtain results corresponding to the steady state of the system. These models came with two criticisms. First, the demand cannot be used to control the system, the input of the system is continuous, so only the output can be evaluated. Second, a minor change in the system may lead to major changes in the model.

On the other hand, flow models—which are deterministic—consider the flow of parts trough the system. Let $x_n(t)$ be the surplus (if positive) or backlog (if negative) of type parts n at time t. It is the difference between production and demand, and is given by:

$$dx_n/dt = u_n(t) - d_n$$

with d_n the demand rate for type parts n, and $u_n(t)$ the production rate for part type n at time t. Here, results on the transient response of the system can be obtained. However it is difficult to include machine failures in this model.

To solve this problem, a different approach is proposed in this paper. We describe each state of our system by one flow model. Each state correspond to a configuration of the system. Each configuration is generated by the state of all the machines. But we have to model the transitions between models, i.e. during the evolution of the system, machine failures appear, leading to a change from one model to another.

2 PROBLEM FORMULATION

The problem could be defined as follows: for p different type of parts, we have to produce $n_i, i \in \{1, 2, \ldots, p\}$ items of each type at the end of a period of time $\bar{T}$.

We consider a flexible manufacturing system with m machines where each machine can be in either functional or breakdown state. With this system we may produce the p types of part. Each part type requires processing for some specified operations. The system is flexible in the sense that each machine can process different operations with virtually no setups but not necessarily at the same production rate.

For a part type, a machine is considered as having two states: – the part needs to be processed on this machine – the part does not use this machine. But a machine can be either in functional or in breakdown state, without consideration for the part types it can process. Therefore, only two states are important for a part type , the machine is used or the machine is not used (failure or not in use). A model with the state of the machines for all the part types leads to an exponential number of states for the whole FMS. So as to reduce the number of states, we introduce the concept of "*path*". A path is a sequence of machines where a part type can be processed. For example, part of type p_1 may be processed at machine M_1, then at M_2, then at M_3; they may also follow another path $\{M_1, M_2, M_4\}$ or $\{M_3, M_4, M_6\}$. If the production on the FMS is complex enough, the number of states with the paths approach is much less than the number of states with the machines approach, i.e. with m machines and s paths, the 2^m states of machines do not lead to 2^s states of paths, some of them cannot be reached due to the fact that different paths can share common machines.

3 Hierarchical model

A hierarchical model is based on separating different tasks with a time criterion. Gershwin [Ger89] proposed a structure for production planning problems in manufacturing systems. The main idea is to treat events and variables according to the speed of their dynamics, so that at any control level some events may be considered as static at one end of the time scale. A slightly modified hierarchy is used here.

3.1 First-level

At this level, we must take the changes in machine states into consideration in order to obtain the state of paths. Our FMS consist of m machines producing p types of parts. For each part type we have some possible paths; for example:
6 machines producing 3 part types with 3 paths per part type.

Operation	Part type 1			Part type 2			Part type 3		
1st	M1	M4	M1	M2	M5	M1	M2	M1	M3
2nd	M2	M5	M4	M3	M6	M5	M4	M2	M6
3rd	M3	M6	M6	M4	M1	M4	M6	M3	M2
4th			M3			M6	M3		

Each operation can have a different duration, i.e., for part type 1, the second operation on path 1 (on machine M2) can be the same as the second operation on path 2 (on machine M5), but with different processing time. To define the behavior of a machine between the functional state and the breakdown state, a discrete-markov chain with an exponential law is used. For each machine, this leads to a description by two parameters: the mean time between failure ($MTBF$) and the mean time to repair ($MTTR$). As all machine are assumed independent, probabilities of events can be summed. With this law, we can calculate off-line the different models and the transition probabilities matrix of the set of modes:

- Enumerate all the states for the machines, $\Xi = \{\xi_1, \xi_2, \ldots, \xi_{2^m}\}$, $\xi_i = \{\alpha_1, \alpha_2, \ldots, \alpha_m\}$. $\alpha_j, (j \in \{1, 2, \ldots, m\})$ is the binary state of the i^{th} machine (functional or breakdown). ξ_i is a configuration of functional or breakdown machines of the real system. 2^m is the number of system states with the machines (*states of machines*).

- With the previous table, calculate the path state ϕ_l corresponding to the machine state ξ_i. But we must take into account the fact that different states of machines $\{\xi_i, \xi_j\}$ can lead to tbe same state of paths ϕ_l.

- The set of states of paths $\Phi = \{\phi_1, \phi_2, \ldots, \phi_f\}$ is obtained, with $f = Card\{\Phi\}$ the number of system states with the paths (*states of paths*).

- For all the possible transitions between machines states, we can obtain this transition probability:

$$\xi_i = \{\alpha_1, \alpha_2, \ldots, \alpha_m\}, \xi_j = \{\beta_1, \beta_2, \ldots, \beta_m\}$$

$$Prob\{\xi(k+1) = \xi_i | \xi(k) = \xi_j\} = p_{ij} = \prod_{k=1}^{k=m} \eta(\alpha_k, \beta_k)$$

α_k	β_k	$\eta(\alpha_k, \beta_k)$
0	0	$1 - \mu_k$
0	1	μ_k
1	0	λ_k
1	1	$1 - \lambda_k$

with $\lambda_k = 1/(MTBF_k * \delta t)$ and $\mu_k = 1/(MTTR_k * \delta t)$, δt is the sampling period (T).

- With the link between $\{\xi_i, \phi_l\}$ and $\{\xi_j, \phi_m\}$ we sum p_{ij} to $Prob\{\phi(k+1) = \phi_l | \phi(k) = \phi_m\}$ to obtain π_{ij}, the probability to jump from the state of paths ϕ_i to ϕ_j due to a machine failure.

With the above example, only 16 states of paths are obtained versus 64 states of machines:

State of path	N	State of path	N	State of path	N	State of path	N
0 0 0 1 0 0 1 0 1	1	0 0 1 0 0 0 0 0 0	1	1 0 0 0 0 0 0 1 1	1	0 0 0 0 0 0 0 0 1	2
0 0 0 1 0 0 0 0 0	2	1 0 0 0 0 0 0 1 0	2	0 1 0 0 0 0 0 0 0	3	0 0 0 0 1 0 0 0 0	3
0 0 0 0 0 0 0 0 0	40	0 1 1 0 1 1 0 0 0	1	1 0 1 1 0 0 1 1 1	1	1 0 0 0 1 0 0 1 1	1
0 1 0 1 0 0 1 0 1	1	1 0 0 1 0 0 0 1 0	2	0 1 0 0 1 1 0 0 0	2	1 1 1 1 1 1 1 1 1	1

N is the number of system states (machines states) that lead to this path state. Each machine has an $MTBF$ equal to 10 units of time and an $MTTR$ of 1 unit of time.

A breakdown state for a path is due to machine failure for at least one of the machine involved in this path. The number of state f is less than 2^m because different configurations of machines can lead to the same configuration of paths.

For a system with only one configuration (no machine breakdown), the evolution of the number of parts produced is modeled by the linear discrete-time equation:

$$x(k+1) = x(k) + Bu(k); \qquad x(0) = x_0$$

with:

$x(k) \in \Re^p$ $\quad x(k)_i$ is the stock available for part type i at the beginning of k^{th} period, p is the number of products (part type).
$u(k) \in \Re^s$ $\quad$ production rates at period k,
$s = \sum_{i=1}^{p} q_i$
q_i is the number of possible paths to produce part type i
$u(k)$ $\quad$ is made up with subvectors $u_i(k)_j$ $(1 \leq i \leq p; 1 \leq j \leq q_i)$.
$u_i(k)_j$ $\quad$ is the production rate for part type i considering path j.

Matrix B is the control matrix which depends on the configuration of operating paths. The elements of B will be zero when a path is not used to process the considered part type, otherwise an element represents the fraction of processing time for a part type in terms of the sampling period. The sampling period T is constant and represents the interval between any two successive periods k and $k+1$.

Since we have modeled our real-system as a set of exclusive systems, we must now define the way the real-system evolves from one system to another. Different parameters must be introduced to represent the change of model for the system in the inventory balance equation:

$$x(k+1) = x(k) + B_{r(k)}u(k)$$

If the system is operating in mode i, i.e. the configuration of the machines gives us the i^{th} path configuration, the form process $\{r(k) : k = 0, 1, \ldots, N\}$ is a finite-state Markov chain taking values in $F = \{1, 2, \ldots, f\}$, with transition probabilities π_{ij} such that:

$$\begin{aligned} Prob\{r(k+1) = j | r(k) = i\} &= \pi_{ij} \\ Prob\{r(0) = i\} &= \pi_i \\ \sum_{j=1}^{f} \pi_{ij} &= 1 \end{aligned}$$

Each state of the markovian chain is associated with a control matrix $B_l, l \in \{1, \ldots, f\}$. These matrices will depend on the configuration of operating paths. f matrices B_l should be computed to represent the control matrices for paths in function of the state of machines. T is chosen such that:

$$T \ll min\{MTBF, MTTR\}$$

This choice is made to ensure that all machine breakdowns and repairs are known to the controller i.e. no machine can breakdown and be repaired within a single period T.

This model, in term of control, leads to a discrete time markovian jump linear system. Associated with a quadratic cost criterion, a set of coupled Riccati matrix equations should be solved to determine the control law. An algorithm to solve such a problem is described in [AKS93], and many results in optimal control can be exploited.

3.2 Second-level

The second-level controller has a longer time-scale and longer period than the first-level controller. The changes in system state are not considered at this level. The demand for each part type must be taken into account for this level. The inventory balance equation for this level can be written as:

$$\tilde{x}(k+1) = \tilde{x}(k) + \tilde{B}\tilde{u}(k) - d(k) + e(k)$$

with:

$\tilde{x}(k) \in \Re^p$ — $\tilde{x}(k)_i$ is the stock available for part type i at the beginning of k^{th} period, p is the number of products (part type)

$\tilde{u}(k) \in \Re^s$ — production rates at period k

$d(k) \in \Re^p$ — demand for all part types at the end of period k

$e(k) = x(N) \in \Re^p$ — production made by first-level during the previous horizon

$\tilde{x}(0)$ — is given

Here $\tilde{B}$ is a control matrix which sums the production of different paths for each part type. To ensure that we can use an average value for the control matrix $\tilde{B}$ within a single period $\tilde{T}$, the sampling period is constant and is chosen such that:

$$\tilde{T} \gg MTTR$$

To guaranty the continuity between levels, we can choose the sampling period of the second level equal to the horizon of the first level. Capacity constraints must be taken into account at this level to ensure a feasible control. For that purpose, they can be included in the cost criterion of this level, using penalty functions.

4 Concluding remarks

The problem of mathematical model for a FMS with failure prone machine is studied in this paper. To cope with the demand, we use a discrete-time flow model. And to take breakdowns of machines into account, we propose to use a markovian jump parameter in the above model. A hierarchical model is used to ensure capacity constraints on the machines. The method proposed in this paper is a compromise between fully deterministic models and stochastic queuing networks.

References

[AK86] R. Akella and P. Kumar. Optimal control of production rate in a failure prone manufacturing system. *IEEE Trans. Automatic Control*, 33:116–126, 1986.

[AKS90] R. Akella, B.H. Krogh, and M.R. Singh. Efficient computation of coordinating controls in hierarchical structures for failure-prone multi-cell flexible assembly systems. *IEEE Trans. Robotics and Automation*, 6:659–672, 1990.

[AKS93] H. Abou-Kandil and O. De Smet. A first level control strategy for failure-prone multi-machine manufacturing systems. In *Proceedings of IEPM 93*, volume II, pages 771–780, june 1993.

[Ger89] S.B. Gershwin. Hierarchical flow control: a framework for scheduling and planning discrete events in manufacturing systems. *Proceedings of IEEE*, 77:195–209, 1989.

[Ho87] Y. H. Ho. Performance evaluation and perturbation analysis of discrete event dynamic systems. *IEEE Trans. Automatic Contr.*, 32:563–572, 1987.

[KG83] J.G. Kimemia and S.B. Gershwin. An algorithm for the computer control of production in flexible manufacturing systems. *IIE Trans. Automat. Contr.*, 15(4):353–363, 1983.

[MG88] O.Z. Maimon and S.B. Gershwin. Dynamic scheduling and routing for flexible manufacturing systems that have unreliable machines. *Operations Research*, 36:279–292, 1988.

[Sha88] A. Sharifnia. Production control of a manufacturing system with multiple machines states. *IEEE Trans. Automatic Control*, 33:620–625, 1988.

Transient Analysis of Manufacturing Systems Using Continuous Petri Nets

N. Zerhouni, M. Ferney and A. El Moudni

Laboratoire de Mécanique et Productique (LMP)
Ecole Nationale d'Ingénieurs de Belfort
8 Bd Anatole France, 90016 Belfort (FRANCE)

Abstract :
The analysis of manufacturing system is a complex problem. A manufacturing system is often considered as a discrete event system and has always a discrete model. These models give an exact description of discrete systems and give analytical results about steady state. But it don't always allow the transient analysis. The continuous approach of Petri nets modelling gives an analytical expression and permits some analysis of these systems.

Keywords: Continuous Petri net, transient analysis, manufacturing system.

1 .Introduction

The analysis of manufacturing system is a complex problem [5][6]. A manufacturing system is often considered as a discrete event system and has always a discrete model. These models give an exact description of discrete systems and give analytical results about steady state. But they don't always allow the transient behaviour studies. Among the modelling tools, the Petri nets (PN) are known and used [1][2]. But the discrete form of this tool does not give a transient analysis. The continuous Petri Nets have been defined [3][4]. It can be seen as an approximation of timed Petri net. The marking of a place which is an integer in discrete Petri nets becomes a real number in the continuous Petri nets and the transition firing at accurate times becomes a continuous crossing. The continuous approach of this modelling gives an analytical expression of these systems.

In this paper, continuous Petri nets model with variable firing speeds is presented. This model is called variable continuous Petri net : VCPN. The application of this tool to study the transient behaviour of manufacturing systems is pointed out on an example of manufacturing system.

2. Continuous Petri nets

This model is based on the following idea : with each transition is associated a maximal firing speed. In the net, this firing speed will depend on the marking of upstream places.It's called variable firing speeds continuous Petri nets.
Let us consider a station (Fig. 1-a). This station is composed by a queue and two servers.

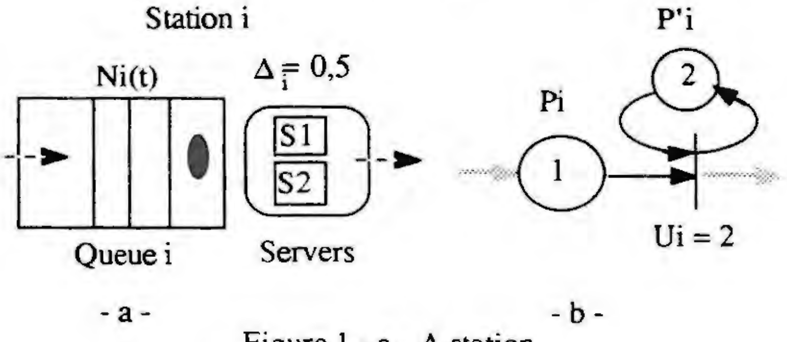


Figure 1.- a - A station.
- b - Continuous PN associated with the station.

In Fig 1-b, the tokens in Pi represent the number of customers in the queue. The place P'i represents the servers. The place P'i corresponds to a reading operation and has a constant marking. The maximal speed of transition Ti (Ui = 2) is equal to the inverse of the service time. If the number of customers is greater than the number of servers, the production is limited by the number of servers. Otherwise, it is the input number of customers which limits the production speed.

2.1 Firing speeds

In a VCPN, the firing speed of a transition depends on all the input constraints. Hence, the firing speed of a transition is proportional to the smallest marking of upsream places.

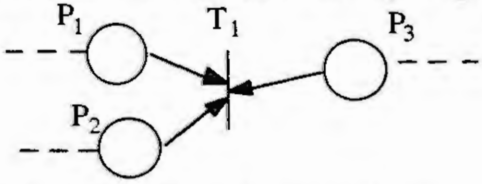


Figure 2. A transition with 3 input places

The expression of firing speed is :

$$\mathbf{v_j = U_j \min (m_i) \qquad , m_i \in {}^\circ T_j}$$

Where $^\circ T_j$ is the set of input places for T_j.

The expression for the firing speed in Fig. 1-b is :

$v_i = U_i \min (m_i,\ m'_i) = U_i \min (m_i, 2)$ because m'_i is always equal to 2.

If $0 \le m_i \le 2$, then $v_i = m_i.U_i$ And If $m_i \ge 2$, then $v_i = 2.U_i = 4$

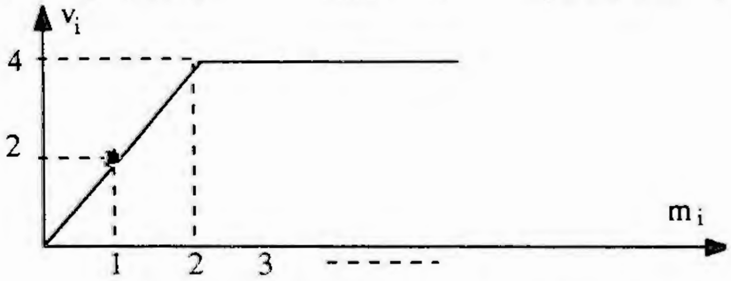


Figure 3. The firing speed function of marking

2.2 Markings

The variation of marking dm_i during the interval of time dt is given by the following expression

$$dm_i = (\sum_{j=1}^{n} w_{ij}.v_j).dt \qquad \text{for each place Pi}$$

Where "$\sum_{j=1}^{n} w_{ij}.v_j$" represents the balance of the marking for

Then $$\frac{dM}{dt} = W.v$$

Where $M = [\ m_i\ ]$ is the marking vector, $W = [\ w_{ij}\ ]$ is the incidence matrix which represent the net structure and $v = [\ v_j\ ]$ the Firing speed vector.

This system of equations represents the marking behaviour of each place. It is a non-linear system. Its evolution is obtained by the successive resolution of a system of equations. A change in the equations corresponds to the change of a phase. A phase is defined as the period during which there is no change in the expression for the speed.

3.Application to transient analysis of manufacturing system

The manufacturing systems studies are generally in steady state. We are interested about the transient behaviour of these systems. We limit this study to an example of particular class of manufacturing system : the manufacturing lines. A manufacturing line is made up of a number of production stations in series. We distinguish between the "open" manufacturing lines which have a variable number of pallets and the "closed" manufacturing lines which have a constant number of pallets. We are interested in the "closed" manufacturing lines in this paper (fig.4).

This study of manufacturing lines is done with the following assumptions.
Assumption 1 : The buffer preceding the manufacturing system is never empty and the buffer succeeding the manufacturing system is never full.
Assumption 2 : The machines has no failure.
Assumption 3 : The capacity of the buffer stocks is not limited. (This assumption is not retrictive. The behaviour of the line can be easily deduced from the analysis of the unlimited capacity case)

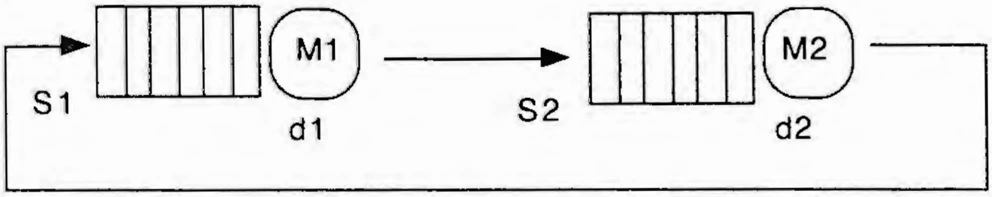


Figure 4. Example of manufacturing line

The maximal firing speed Ui corresponds to the inverse of service time for each station ($d_i=1/U_i$). Initially, all the pallets are in the first station and all the other buffer is empty.

Let us consider the VCPN with two places and two transitions given in fig.5 which modelled the example of manufacturing line (fig.4). The place P1 is marked initially by n and the place P2 have no marking. n represents the total number of the pallets.

In the VCPN of this example, the firing speed of T_i is : $v_i = U_i \min(m_i, 1)$ because there is one machine per station. v_i represents the production rate of each machine.

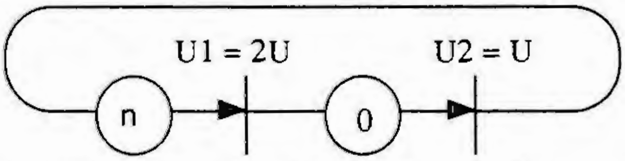


Figure 5. The VCPN representing the example manufacturing line

The evolution of the second buffer S2 is pointed out thanks to the marking of the place P2 in the continuous Petri nets as follows. The evolution of buffer S1 i.e the marking of place P1 can be deduced thanks to the invariant $[m_1(t) + m_2(t) = n]$.

Phase 1 : [0, t1[

$m_1(0) = n$ and $m_2(0) = 0$ => $v_1 = U_1 = 2U$ and $v_2 = m_2.U_2 = m_2.U$

$dm_1/dt = m_2U_2 - U_1 < 0$ and $dm_2/dt = U_1 - m_2U_2 = 2U - m_2U > 0$

$$\Rightarrow \mathbf{m_2(t) = 2\ [1 - \exp(-Ut)]}$$

The change of phase occurs at time t_1 when the marking of P2 is 1($\mathbf{t_1 = (\ln 2)/U}$). The marking has an exponential evolution in the first phase.

Phase 2 : [t1, t2[

$m_1(t_1) = n - 1$ and $m_2(t_1) = 1$ => $v_1 = 2U$ and $v_2 = U$

$dm_1/dt = -U$ and $dm_2/dt = +U$

$$\Rightarrow \mathbf{m_2(t) = Ut + 1 - \ln 2}$$

The change of phase occurs at time t_2 when the marking of P1 is 1 : **[$t_2 = (n - 2 + \ln 2)/U$]**. The marking has a linear evolution in the second phase.

Phase 3 : $[\ t_2, +\infty[$

$m_1(t_2) = 1$ and $m_2(t_2) = n - 1$ => $v_1 = 2U.m_1$ and $v_2 = U$

$dm_1/dt = U - 2U.m_1$ and $dm_2/dt = 2U.m_1 - U$

=> $\mathbf{m_2(t) = n - 0{,}5 - 2\ \exp[2(-Ut + n - 2)]}$

The marking has an exponantial evolution in the third phase. In this example, there are three phases. In the third phase, the markings tends towards its stationnary values (Fig. 6).

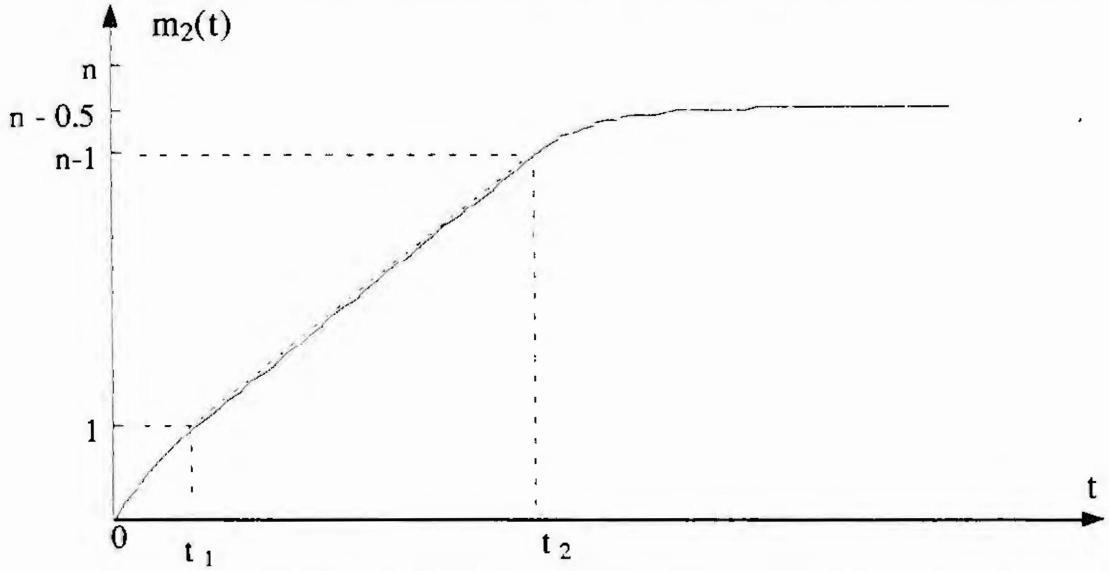


Figure 6. The P2 marking illustrate the transient phases

One can notice that in this example, three phases. Two transient phases and the third stationnary phase. In each phase, the analytical expression of place P2 markings corresponding to the buffer S2 is presented. The analytical values of the steady state corresponds to the average marking in the discrete system. The duration of transient phase is equal to [(n-2+ln2)/U]. It's depend on both n (number of palets) and U (machine speeds).

Conclusion

We first have introduced the continuous model of Petri nets by the model which have variable firing speeds. The study presented in this paper shows the interest of the continuous Petri nets tool in the transient analysis of the manufacturing lines. It is possible thanks to the analytical expression of markings in the continuous Petri nets. Much work remains to be done to define a dynamic properties of general manufacturing systems and extending these results.

References

[1] G.W.BRAMS, "Réseaux de Petri : Théorie et Pratique", Masson Ed., Paris, 1983

[2] P.CHRETIENNE, "Les réseaux de Petri temporisés": Thèse de doctorat d'état, Univesité Pierre et Marie Curie, Paris, 1983

[3] R. DAVID, H.ALLA, Autonomous and Timed Continuous Petri Nets, 11th International Conference on Application and Theory of Petri Nets, Paris, June 1990, PP. 367-386.

[4] R.DAVID et H.ALLA, "Du Grafcet aux réseaux de Petri", Ed.HERMES, Paris, 1989

[5] H.P.HILLION et J.M.PROTH, "Analyse de fabrication non linéaires et répétitives a l'aide de graphes d'événement temporisés", septembre 1986

[6] N. ZERHOUNI, H. ALLA, Dynamic analysis of manufacturing systems using continuous Petri nets, 1990 IEEE International Conference on Robotics and Automation, May, 1990, Cincinnati, USA.

A Boolean algebra for a formal expression of events in logical systems

Bruno Denis ; Jean-Jacques Lesage ; Jean-Marc Roussel

Laboratoire Universitaire de Recherche en Production Automatisée
Ecole Normale Supérieure de Cachan
61, avenue du président Wilson
F-94235 Cachan cedex - France
e-mail : denis@lurpa.ens-cachan.fr

Abstract. The dynamic modelling of logical systems widely calls upon the event notion. In terms of Function Chart Grafcet or Petri Nets for instance, events are generally represented by "rising or falling edges" of logical variables. However, numerous ambiguities are encountered in the models because the translation of events into edges is not formal enough. In this paper, we propose a Boolean algebra the definition-set of which allows us to describe the time behavior of the inputs and the outputs of any logical system. In this algebra, we have defined two unary operations in order to formally express the events. Then, we are giving 14 properties related to these rising and falling edge operations and their composition with the operations AND, OR, and NOT.

Key Words. Boolean algebra, event, logical system, Grafcet, Interpreted Petri Nets.

1. INTRODUCTION

The notion of event is widely used in describing the dynamic behavior of logical systems, mostly when it is represented by combinatory equations, Grafcets or Interpreted Petri Nets (IPN). The use of events in the transition conditions of a graph introduces a wide variety of descriptions that turns out very useful for the representation of real time systems [15]. It actually allows for the conditioning of the clearing of transitions by the occurrence of an event (event - connected approach) and not only through the checking of a condition (condition-connected approach) [14]. Both these approaches are joint in the Interpreted Petri Nets if one associates a predicate and an event to each transition [10].

In this paper we will show that, even though the relevance of the notion of event cannot be questioned, the same cannot apply to the rigor of its definition or its use in the transition conditions of Grafcets or Interpreted Petri Nets. We will first demonstrate the limits and inaccuracies of the notion of "edges" and the interest of building an "extended" Boolean algebra so as to reach a more formal approach of events in transition conditions. This Boolean algebra will then be described, and its main properties given.

2. PROBLEMATICS OF THE TAKING INTO ACCOUNT OF EVENTS

From a theoretical point of view, the notion of event corresponds to a zero time spectrum information, i.e the information that translates the supposedly instantaneous change of state of a logical variable or of the function of logical variables [1], [7], [2]. The concept is then translated in the notation "rising edge : ↑ " or "falling edge : ↓ " of the variable or of the function of variables [11] (Fig. 1).

Yet in practice, the use of edges is often limited to the sole elementary logical variables as the evaluation of expressions that hold edges of functions of logical variables - as in equation (1) - is not mastered.

$$E_1 = \uparrow (a \cdot \bar{c} + b) \cdot \downarrow (a \cdot b + c) \tag{1}$$

When the restrictive hypotheses of non-simultaneous events or of total independence between the a, b, c variables can be emitted, heuristics such as (2) or (3) may be used [4] :

$$\uparrow (a \cdot b) = \uparrow a \cdot b + a \cdot \uparrow b \tag{2}$$

$$\uparrow(a+b) = \uparrow a \cdot \bar{b} + \bar{a} \cdot \uparrow b \qquad (3)$$

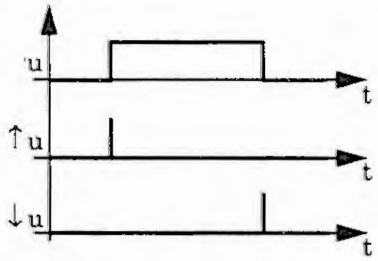


Fig. 1 Rising edge and falling edge of a boolean variable.

One only has to examine such an expression as (1) to identify the cause of the difficulties met in evaluating such logical equations. In fact, in this expression, "+" and "." represent the two operations of composition of Boole's Algebra and "‾", the unary complementary operation ; these operations are thus perfectly defined by their truth table. However, "↑" is only a mere notation that shows that the designer of this expression is interested in the change of state of the function $(a \cdot \bar{c} + b)$ and not in its logical level "1". In [3] the authors actually stress that grafcet transition conditions that hold edges of variables are not defined according to Boole's Algebra.

So as to develop and evaluate such expressions, whatever the number of variables that make up the logical functions, and taking into account the possibility of distinct simultaneous events, we now wish to build an "extended" Boolean algebra that holds two event-connected unary operations : the "rising edge" and the "failing edge". Thus equipped with an algebraic definition of edges, we will establish a set of properties that allows for the development and the evaluation of expressions such as (1).

3. BUILDING OF AN "EVENT-CONNECTED" BOOLEAN

3.1 Set of definition

The set of definition for the researched algebra must :

- faithfully represent the inputs and outputs of any logical system,
- allow for the temporal taking into account of events,
- hold at least two elements,
- be closed under all the operations defined on it.

Taking these four criteria into consideration, we have retained and will note $\mathbb{I}$ the set of functions defined on $\mathbb{R}^{+*}$, whose range is $\mathbb{B} = \{0, 1\}$, that verify the following property.

$$\mathbb{I} = \{u : \mathbb{R}^{+*} \to \mathbb{B} \mid$$
$$\forall t \in \mathbb{R}^{+*} : (\exists \varepsilon_t > 0 : (\forall (\varepsilon_1, \varepsilon_2) \in]0, \varepsilon_t[^2, u(t-\varepsilon_1) = u(t-\varepsilon_2)))\ \}$$

By definition, all u functions of $\mathbb{I}$ are then piecewise continuous and can admit a double discontinuity at certain points. The general shape of a function of the set $\mathbb{I}$ is represented in Fig. 2.

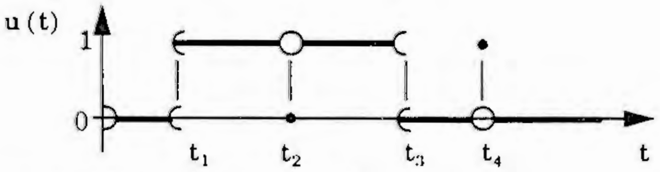


Fig. 2 Example of an element function of the set $\mathbb{I}$.

At the points of discontinuity, the problem of the value of the function is posed. The function can actually be considered as right-continuous or left-continuous. As J.P. Frachet, we will hold for the right - continuity, as it is more natural to the physicist, it being causal [5]. At the date of occurrence of an event, we will therefore consider that the function has already changed its value.

We admit the existence of points that present a double discontinuity $(u(t_2) = 0 ; u(t_4) = 1)$ so as to ensure that $\mathbb{I}$ is closed under the edge operations.

It may be stressed here that such a definition of the elements of $\mathbb{I}$ is perfectly in conformity with the practice of automaticians who frequently represent the time evolution of boolean values as timing diagrams.

3.2 Convention of notation

To make the reading of this paper easier, and to avoid all possible mixing of the operations on the elements of $\mathbb{I}$ (function $u : \mathbb{R}^{+*} \rightarrow \mathbb{B}$) and the booleans (values taken by these functions at a given time), we will from then on note "$\wedge$" the logical operation AND, " $\vee$" the logical operation OR, "$\neg$" the NOT operation on a boolean. The notations ".", "+", et "$^{-}$" will be dedicated to the operations on $\mathbb{I}$.

Furthermore, we have carefully distinguished the function from the boolean, i.e the value taken at a given time by this function. For instance, u, v, w are three functions element of $\mathbb{I}$ while u(t), v(t), w(t) are three booleans.

3.3 Definition of operations on $\mathbb{I}$

After having explained our set of definition and precised the notations, we can defined the following operations on $\mathbb{I}$.

the **AND** operation

$$\begin{array}{c} \mathbb{I}^{2} \rightarrow \mathbb{I} \\ (u, v) \rightarrow (u \cdot v) \end{array} \text{ with } \forall t \in \mathbb{R}^{+*},\ (u \cdot v)(t) = u(t) \wedge v(t)$$

the **OR** operation

$$\begin{array}{c} \mathbb{I}^{2} \rightarrow \mathbb{I} \\ (u, v) \rightarrow (u + v) \end{array} \text{ with } \forall t \in \mathbb{R}^{+*},\ (u + v)(t) = u(t) \vee v(t)$$

the **NOT** operation

$$\begin{array}{c} \mathbb{I} \rightarrow \mathbb{I} \\ u \rightarrow \bar{u} \end{array} \text{ with } \forall t \in \mathbb{R}^{+*},\ \bar{u}(t) = \neg u(t)$$

By definition, $\mathbb{I}$ is closed under these operations and $\mathbb{I}$ is a boolean algebra [6] [8] [9].

3.4 Taking event into account in this algebra

All the interest of this algebra resides in the fact that we can now strictly define two extra unary operations to formally express the notion of event.

the **RE** operation (rising edge)

$$\begin{array}{c} \mathbb{I} \rightarrow \mathbb{I} \\ u \rightarrow \uparrow u \end{array} \text{ with } \forall t \in \mathbb{R}^{+*},\ \uparrow u(t) = u(t) \wedge (\exists \varepsilon_0 > 0 : \forall \varepsilon \in \,]0, \varepsilon_0[,\ u(t - \varepsilon) = 0)$$

the **FE** operation (falling edge)

$$\begin{array}{c} \mathbb{I} \rightarrow \mathbb{I} \\ u \rightarrow \downarrow u \end{array} \text{ with } \forall t \in \mathbb{R}^{+*},\ \downarrow u(t) = \bar{u}(t) \wedge (\exists \varepsilon_0 > 0 : \forall \varepsilon \in \,]0, \varepsilon_0[,\ u(t - \varepsilon) = 1)$$

The images of t under the function $\uparrow u$ (respectively $\downarrow u$) are thus determined at all moments as the logical AND between two booleans. The first boolean is the value of the u-function (respectively the complement of the value) at that moment, whereas the second boolean is the value of a predicate at the same moment. The truthfulness of the predicate depends on the value taken by the function u on the interval $]t - \varepsilon_0, t[$.

For the function represented on Fig. 2 for instance :

- $\uparrow u(t) = 1$ if $t \in [t_1, t_2[\cup]t_2, t_3[\cup \{t_4\}$ and if $t \in]0, t_1] \cup]t_3, t_4] \cup]t_4, \infty[$ i.e. $t = t_1$ or $t = t_4$.
- $\downarrow u(t) = 1$ si $t \in]0, t_1[\cup \{t_2\} \cup [t_3, t_4[\cup]t_4, \infty[$ and if $t \in]t_1, t_2] \cup]t_2, t_3]$ i.e. $t = t_2$ or $t = t_3$.

By definition, $\mathbb{I}$ is closed under these two operations as the functions $\uparrow u$ and $\downarrow u$ are defined in $\mathbb{R}^{+*}$, with boolean values and verify the property of the elements of $\mathbb{I}$ developed in paragraph 3.1.

3.5 FUNDAMENTAL PROPERTIES

The following properties ((4) to (17)) have been demonstrated on the set $\mathbb{I}$ (these proofs as well as an example have been developed in [12] :

$$u + \uparrow u = u \quad (4) \qquad u \cdot \uparrow u = \uparrow u \quad (5) \qquad \uparrow \bar{u} = \downarrow u \quad (6)$$

$$\bar{u} + \downarrow u = \bar{u} \quad (7) \qquad \bar{u} \cdot \downarrow u = \downarrow u \quad (8) \qquad \downarrow \bar{u} = \uparrow u \quad (9)$$

$$\uparrow(\uparrow u) = \uparrow u \quad (10) \qquad \uparrow(\downarrow u) = \downarrow u \quad (11) \qquad \downarrow(\uparrow u) = 0^{*} \quad (12) \qquad \downarrow(\downarrow u) = 0^{*} \quad (13)$$

$$\uparrow\left(\prod_{i=1}^{n} u_i\right) = \sum_{i=1}^{n}\left(\uparrow u_i \cdot \prod_{(j=1),(j\neq i)}^{n} u_j\right) \quad (14) \qquad \downarrow\left(\prod_{i=1}^{n} u_i\right) = \sum_{i=1}^{n}\left(\downarrow u_i \cdot \prod_{(j=1),(j\neq i)}^{n} \downarrow u_j + (u_j \cdot \overline{\uparrow u_j})\right) \quad (15)$$

$$\downarrow\left(\sum_{i=1}^{n} u_i\right) = \sum_{i=1}^{n}\left(\downarrow u_i \cdot \prod_{(j=1),(j\neq i)}^{n} \overline{u_j}\right) \quad (16) \qquad \uparrow\left(\sum_{i=1}^{n} u_i\right) = \sum_{i=1}^{n}\left(\uparrow u_i \cdot \prod_{(j=1),(j\neq i)}^{n} \uparrow u_j + (\overline{u_j} \cdot \overline{\downarrow u_j})\right) \quad (17)$$

4. CONCLUSION

In this paper we have presented the results of a theoretical work that aims at making up for the lack of a formal definition of the notion of event as it is practiced in Grafcet. We have therefore built up an algebra on a set $\mathbb{I}$ of functions defined in $\mathbb{R}^{+*}$ with boolean values. In this algebra, two operations have been defined for the formal definition of events : the rising edge operation and the falling edge operation. Fourteen properties have then been gived in relation with the edge operations and their combinations. This work, though it has an inner finality, is actually part of a global project that aims at the analysis of the coherence and of the dynamic behavior of complex systems. The Boole's algebra that we have presented has actually allowed us to design a module of formal calculation and of simplification of combinatory expressions. This module of formal calculation is itself used for the analysis of the dynamics of grafcets and the automatic generation of the equivalent automaton (AGGLAE Project of the LURPA [13]).

5. REFERENCES

[1] French AFCET working group, Pour une représentation normalisée du cahier des charges d'un automatisme logique. AII, 61 (1977) 27-32 & 62 (1977) 36-40, Dunod (Ed.) France.

[2] Blanchard M., Comprendre maîtriser et appliquer le GRAFCET. Cépaduès (Ed.), Toulouse-France, 1979.

[3] Bouteille N. & al., Le GRAFCET. Cépaduès (Ed.), Toulouse-France, 1992.

[4] David R. & Alla H., Petri Nets and Grafcet tools for modelling discrete event systems. Prentice Hall (Ed.), London, 1992.

[5] Frachet J.P. & Colombari G., Elements for a semantics of the time in GRAFCET and dynamic systems using non-standard analysis. APII Hermès (Ed.), 27-1 (1993), 107-125.

[6] Garding L. & Tambour T., Algebra for Computer Science. Springer Verlag (Ed.) New-York, 1988.

[7] GREPA, Le GRAFCET de nouveaux concepts. Cépaduès (Ed.), Toulouse-France, 1985.

[8] Halmos P., Lectures on Boolean Algebras. Van Nostrand Mathematical Studies (Ed.), New-York, 1967.

[9] MacLane S. & Birkhoff G., Algebra. The MacMillan Compagny (Ed.), London, 1967.

[10] Moalla M. & al., Synchronized Petri Nets : A model for the description of non-autonomous systems. Mathematical Foundations of Computer Sciences, Springer Verlag (Ed.), (1978) 374-383.

[11] IEC 848 Standard, Preparation of function charts for control systems. 1988.

[12] Roussel J.M. & Lesage J.J., Une algèbre de Boole pour l'approche événementielle des systèmes logiques. To appear in : APII Hermès (Ed.), 27-5 (1993).

[13] Roussel J.M., Analyse de grafcets par génération logique de l'automate équivalent. PhD thesis of E.N.S. de Cachan-France , To appear, 1994.

[14] Sayat B. & Ladet P., Control specification of a production system using GRAFCET and Petri nets. APII Hermès (Ed.), 27-1 (1993) 53-64.

[15] Zahnd J., Machines séquentielles. Dunod (Ed.), Paris, 1987.

PROBABILITY MODELS OF RELIABILITY OF INFORMATION - COMPUTER SYSTEMS WITH HIERARCHICAL STRUCTURE

Andrei CORLAT, Nicolai ANDRONATY, Yurii ROGOZHIN

Institute of Mathematics of Academy of Sciences of Moldova

Academiei str. 5, Kishinev 277028, Moldova

Abstract. A mathematical model of information-computer systems with hierarchical structure is build in general assumptions concerning the distributions of failure and repair times of the systems unit. The main characteristics of the reliability of such systems are obtained.

The Information - Computer Systems have in main a specific structure: the information from some principal control unit is forthcoming to several next units, each of that in its turn communicates the information to the following units etc. This takes place in systems in which are carried out the circular transsmition of information signals (or managment signals) or collection of information from downstairs elements.

We will consider the system S with hierarchical structure: the principal unit a_0 is connected with a_1 units of first level, each of that is connected with a_2 units of second level etc. The units of the last n - level are called extreme elements, their number is $N = a_1 \cdot a_2 \cdot \ldots \cdot a_n$ and they formed $K = a_1 \cdot a_2 \cdot \ldots \cdot a_{n-1}$ groups.

The system's unit may be unable to operate either it is failured, or it is disconnected as a result of failure of some unit. The failure of (i,j) unit (i - indicates the level, $i = \overline{0,n}$, j - the number of the unit of i - level, $j = \overline{1, N_i}$, $N_i = a_1 \cdot a_2 \cdot \ldots \cdot a_i$) leads to the disconnection of all units that are connected with this unit and are controled by it and of all preceding units connected with it and that do not belong to any efficient way. We will understand here under an efficient way a chain of functional connected operating units from the principal $(0,1)$ to one of the extreme.

The restored unit is included in system simultaneously with all previously disconnected operative units (with that level of efficiency at the moment when these units are disconnected) that formed an efficient way with the restored unit. Moreover, the disconnected early units unde repair continued (but not start again) their repair if these units are functionaly connected with restored unit.

The system is considered in failure (total failure) if the number of efficient ways is less than R $(1 \leq< R < N)$ and at this moment all operative units are disconnected.

It is assumed that

- the failure times $\alpha_1^{(ij)}$ and the repair times $\alpha_0^{(ij)}$ are independent in totality random variables with limited mean $0 < E\alpha_k^{(ij)} = T_k^{(ij)} < \infty, i = \overline{0,n}, j = \overline{1,N_i}, k = \overline{0,1}$,
- the distribution functions of failure and repair times are considered absolutely continuous with respect to Lebesgue measure, $\overline{F}_k^{(ij)}(t) = 1 - F_k^{(ij)}(t) > 0,\ t > 0$,
- the restored unit is as good as new,
- there are not queuie to repair,
- the disconnection and including of the units in system as failure as are taking place instantaneously.

The functioning of such system is described (see [1]) by the semi-Markov processes $\xi(t) = \{\xi_{01}(t), \xi_{11}(t), \xi_{12}(t), \ldots, \xi_{21}(t), \ldots, \xi_{ij}(t), \ldots, \xi_{nN}(t); v_{01}(t), v_{11}(t), v_{12}(t), \ldots, v_{21}(t), \ldots, v_{ij}(t), \ldots, v_{nN}(t)\}$ where

$$\xi_{ij}(t) = \begin{cases} 1, & \text{if the (i,j) unit is operative at the moment t,} \\ 0, & \text{if the (i,j) unit is under repair at the moment t.} \end{cases}$$

$v_{ij}(t)$ - is (if $\xi_{ij}(t) = 0$) the repair time of (i,j) unit from it last failure and (if $\xi_{ij}(t) = 1$) the lifetime of (i,j) unit from it last join in the system (without taking in consideration the possible time of disconnection).

The phase space of system's states is $(Z, \mathcal{Z})$, (see [2]) where $Z = \{(d; x^{(ij)}) : d \in D\ ,\ x^{(ij)} = (x_{01}, x_{11}, \ldots, x_{1N_1}, x_{21}, \ldots, x_{2N_2}, x_{i1}, \ldots, x_{ij-1}, 0, x_{ij+1}, \ldots, x_{nN}),$
$x_{km} > 0, k = \overline{0,n},\ m = \overline{1,N_k},\ (k,m) \neq (i,j)\}$;
$D = \{d\ :\ d = (d_{01}, d_{11}, \ldots, d_{1N_1}, d_{21}, \ldots, d_{2N_2}, \ldots, d_{km}, \ldots, d_{nN}), d_{km} = \overline{0,1}, k = \overline{0,n}, m = \overline{1,N_k}\}$
x_{km} - points the time passed by the last change of "phisical" state of the (k,m) unit,
d_{km} - describes the "phisical" state of the (k,m) unit:

$$d_{km} = \begin{cases} 1, & \text{if it is operative (or disconnected in an operative state),} \\ 0, & \text{if it is under repair (or disconnected in an failured state).} \end{cases}$$

$\mathcal{Z}$ – σ-algebra of Borel sets in Z.

We define the set of operative system's states Z_1 and the set of failure system's states Z_0 proceeding from the concept of total system's failure.
$Z_1 = \{(d; x^{(ij)}) \in Z : d \in D_1\},\ Z_0 = \{(d; x^{(ij)}) \in Z : d \in D_0\}$ where

$$D_1 = \{d \in D : \sum_{u=1}^{N} S_u \geq R\}\ ,\ D_0 = \{d \in D : \sum_{u=1}^{N} S_u < R\},$$

$$S_u = d_{nu} \cdot d_{n-1,u-1} \cdot \ldots \cdot d_{iu_i} \cdot \ldots \cdot d_{1u_1} \cdot d_{01},$$

$$u_i = \begin{cases} [\frac{u}{a_n \cdot a_{n-1} \cdot \ldots \cdot a_{i+1}}] + 1, & \text{if } u \neq 0(mod(a_n \cdot a_{n-1} \cdot \ldots \cdot a_i)), \\ [\frac{u}{a_n \cdot a_{n-1} \cdot \ldots \cdot a_{i+1}}], & \text{otherwise,} \end{cases}$$

where $[\cdot]$ denotes entire part of the number.

The mean life time T_1 of the system S is given by

$$T_1 = \{ \sum_{d \in D_1} \prod_{i=0}^{n} \prod_{j=1}^{N_i} T^{(ij)}_{d_{ij}} \} \cdot \{ \sum_{d \in D_0} \sum_{(i,j) \in I} \prod_{s=0}^{n} \prod_{\substack{v=1 \\ (s,v) \neq (i,j)}}^{N_s} T^{(sv)}_{d_{sv}} \}^{-1}$$

and the mean repair time T_0 of the system S is given by

$$T_0 = \{ \sum_{d \in D_0} \prod_{i=0}^{n} \prod_{j=1}^{N_i} T^{(ij)}_{d_{ij}} \} \cdot \{ \sum_{d \in D_0} \sum_{(i,j) \in I} \prod_{s=0}^{n} \prod_{\substack{v=1 \\ (s,v) \neq (i,j)}}^{N_s} T^{(sv)}_{d_{sv}} \}^{-1}$$

where I – denotes the set of units under repair that are not desconnected at the state d.

Suppose now that the system S is homogeneous: the units of the i-level are of the same type $T^{(ij)}_k = T^{(i)}_k, k = \overline{0,1}, i = \overline{0,n}$. The system will be considered under total failure when the number of operative extreme groups is less then P $(1 \leq P < K)$, an extreme group is operative if it contains Q or more operative units from a_n.

Then may be suggested an iterative algorithm for determining T_1 and T_0. For example, when $P = 1$

$$T_1 = \frac{T^{(0)}_1 S_+(1)}{T^{(0)}_1 S_*(1) + S_+(1)};$$

$$T_0 = \frac{T^{(0)}_0 S_+(1) + T^{(0)}_1 S_-(1)}{T^{(0)}_1 S_*(1) + S_+(1)};$$

where

$$S_+(n-i) = \left[T^{(n-i)}_1 S_+(n-i+1) + A_{n-i}\right]^{a_{n-i}} - S_-(n-i),\ i = \overline{1,n-1};$$

$$A_{n-i} = T^{(n-i)}_0 S_+(n-i+1) + T^{(n-i)}_1 S_-(n-i+1),\ i = \overline{1,n-1};$$

$$S_-(n-i) = [A_{n-i}]^{a_{n-i}},\ i = \overline{1,n-1};$$

$$S_*(n-i) = a_{n-i}\left[S_+(n-i) + T^{(n-i)}_1 S_*(n-i+1)\right] A^{a_{n-i}-1}_{n-i},\ i = \overline{1,n-1};$$

$$S_+(n) = \left(T^{(n)}_1 + T^{(n)}_0\right)^{a_n} - \sum_{k=a_n-Q+1}^{a_n} C^k_{a_n} \left(T^{(n)}_1\right)^{a_n-k} \left(T^{(n)}_0\right)^k;$$

$$S_-(n) = C^{a_n-Q+1}_{a_n} \left(T^{(n)}_1\right)^{Q-1} \left(T^{(n)}_0\right)^{a_n-Q+1};$$

$$S_*(n) = C^{a_n-Q+1}_{a_n} \left(T^{(n)}_1\right)^{Q-1} \left(T^{(n)}_0\right)^{a_n-Q};$$

It should be mentioned that the results are obtained in terms of structure and means of failure and repair times and in a suitable for coding form.

References

[1] A.Corlat, V.Kuznetsov, M.Novicov, A.Turbin. *Semimarkovian models of systems with repair and queueing systems.* Shtiintsa, Kishinev, 1991 (in Russian)

[2] A.Corlat *Modelling of systems reliability by means of semi-Markov processes.* This volume.

MODELLING OF SYSTEMS RELIABILITY BY MEANS OF SEMI-MARKOV PROCESSES

Andrei CORLAT

Institute of Mathematics of Academy of Sciences of Moldova

Academiei str. 5, Kishinev 277028, Moldova

Abstract. A new method of reliability analysis of complex systems is suggested. It is based on modelling of their evolution by means of semi-Markov processes. The results are obtained under general assumptions concerning the distributions of failure and repair times of the system's unit.

We will consider a complex system S with repairable units. It consists from $N, (N \geq 1)$ units and has a definite functional structure.

The failure time of the i - unit is a random variable (r.v.) $\alpha_1^{(i)}$ with the following distribution function (d.f.) $F_1^{(i)}(t) = P(\alpha_1^{(i)} \leq t), i = \overline{1, N}$, and the repair time is a r.v. $\alpha_0^{(i)}$ with d.f. $F_0^{(i)}(t) = P(\alpha_0^{(i)} \leq t), i = \overline{1, N}$. It is assumed that d.f. $F_k^{(i)}, k = 0, 1$, $i = \overline{1, N}$ are absolutely continuous with respect to Lebesgue measure, and r.v. $\alpha_k^{(i)}$ are independent in totality and have finite means $(0 \leq E\alpha_k^{(i)} < \infty)$

The i - unit may be in one of the following states: operative, operative disconnected, under repair or disconnected under repair.

The disconection of i- unit (or a totality of units) may be a result of total system's failure or of some functional connected with it system's unit and the i-unit does not belong to any efficient way. We will understand here under an efficient way a chain of functional connected units, whose functioning involves the system's viability. When the system's failure occurs all the remained operative units are disconnected. At this moment the repair of units under repair in system is done only.

The disconnected units are included in system with that level of operation or repair which find theirs at the moment of disconnection or of total system failure. These inclusions are taking place simultaneously with some restored unit in condition that these units generate an efficient way.

Moreover it is assumed that

- the restored unit is as good as new,
- there are not queuie to repair,
- the disconnection and including of the units in system as failure as are taking place instantaneously.

The concept of system's failure is introduce proceeding from its functional structure: it may be as a result of the failure of one or a group of units.

The i- system's unit functioning $i = \overline{1, N}$, represents a sequence of alternate periods of operating and repair (with possible disconnection) and the system's functioning - an analogue of superpossition of the N independent alternate renewal processes (see [2]).

Let $\xi_i(t)$ $i = \overline{1, N}$- alternate processes with possible disconnection periods that model the functioning of i-unit with $P\{\xi_i(0) = 1\} = 1$ and $\xi_i(t) = 1$, if in the moment t the i-unit is operating or disconnected in an operative state, $\xi_i(t) = 0$ if it is under repair or desconnected under repair. Following the approach suggested in [1] we will consider the semi-Markov processes
$\xi(t) = \{\xi_1(t), \ldots, \xi_i(t), \ldots, \xi_N(t); u_1(t), \ldots, u_i(t), \ldots, u_N(t)\}$
that is modelling the system's evolution. Here $u_i(t)$ is the repair time of the i-unit from its last failure if $\xi_i(t) = 0$ and is the life time of the i-unit from its last join in system if $\xi_i(t) = 1$ (without taking in consideration the possible time of disconnection).

The phase space of system's states is $(Z, \mathcal{Z})$,
where $Z = \{z = (d; x^{(i)}) : d \in D\ ,\ x^{(i)} \in R_+^{(i)}\}; D = \{d :\ d = (d_1, \ldots, d_i, \ldots, d_N, d_i = \overline{0, 1}, i = \overline{1, N}\}$
$R_+^{(i)} = \{x^{(i)} : x^{(i)} = (x_1, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_N), x_k > 0, k = \overline{1, N}, k \neq i\}$
$\mathcal{Z}$ – σ-algebra of Borel sets in Z.

We will define the semi-Markov kernel of the processes $\xi(t)$

$$Q(t, (d; x^{(i)}), \{b; [0, y]^{(n)}\}) = \begin{cases} P\{x_n + \alpha_{d_i}^{(i)} > \alpha_{d_n}^{(n)}, \alpha_{d_j}^{(j)} > x_j + \alpha_{d_n}^{(n)} - x_n, x_j + \alpha_{d_n}^{(n)} - x_n \in \\ [0, y_j]; j \notin I_d, j \neq i, \alpha_{d_l}^{(l)} > x_i, l \in I_d, \alpha_{d_n}^{(n)} - x_n \leq t \mid \alpha_{d_m}^{(m)} > \\ > x_m, m = \overline{1, N}\}, i \neq n; \\ P\{\alpha_{d_n}^{(n)} > x_n + \alpha_{d_i}^{(i)}, x_n + \alpha_{d_i}^{(i)} \in [0, y_n]; n \notin I_d, n \neq i; \\ \alpha_{d_l}^{(l)} > x_l, l \in I_d, \alpha_{d_i}^{(i)} \leq t \mid \alpha_{d_m}^{(m)} > x_m, m = \overline{1, N}\}, i = n; \end{cases} \tag{1}$$

where $I = \{i : d_i = 0, i \notin I_d\}$; I_d - the set of indices of disconnected units in the state $(d; x^{(i)})$ $\{b; [0, y]^{(n)}\} = \{z \in Z : z = (b; y^{(n)}), y^{(n)} \in [0, y]^n\}$;
$[0, y]^{(n)} = ([0, y_1], [0, y_2], \ldots, [0, y_{n-1}], 0, [0, y_{n+1}], \ldots, [0, y_N])$;

$$[0, y_i] = \begin{cases} [0, x_i + y], & if\ i \notin I_d, i \neq n; \\ [0, x_i], & if\ i \in I_d, i \neq n; \end{cases}$$

The states d and b differs by the n- component only here. In other cases the transition probability from $(d; x^{(i)})$ to $\{b; [0, y]^{(n)}\}$ is zero.

Let us define the mean values of the being time of $\xi(t)$ in the states from Z
$\Theta_{(d,x^{(l)})} = min\{[\xi_n^{(d_n)} - x_n]^+, n \notin I_d\}$, **(2)**
$where[\xi - x]^+$ $r.v.$ $with$ $d.f.$ $P\{[\xi - x]^+ \leq t\} = P\{\xi - x \leq t \mid \xi > x\} = [F(x+t) - F(x)]/\overline{F}(x)$.

The relations **(1)**–**(2)** (see [2]) completely define the semi-Markov processes $\xi(t)$ that model the evolution of the system S.

The mean life time T_1 of the system S is given by

$$T_1 = \frac{\sum_{d \in D_1} \prod_{i=1}^{N} T_{d_i}^{(i)}}{\sum_{d \in D_0} \sum_{n \in I} \prod_{j=1 j \neq n}^{N} T_{d_j}^{(j)}}$$

and the mean repair time T_0 of the system S is given by

$$T_0 = \frac{\sum_{d \in D_0} \prod_{i=1}^{N} T_{d_i}^{(i)}}{\sum_{d \in D_0} \sum_{n \in I} \prod_{j=1 j \neq n}^{N} T_{d_j}^{(j)}}$$

where

$$T_{d_i}^{(i)} = \begin{cases} E\alpha_1^{(i)}, & if\ d_i = 1; \\ E\alpha_0^{(i)}, & if\ d_i = 0; \end{cases}$$

$D_1(D_0)$ - is the set of operative (repair) states.

References

[1] A.Corlat, V.Kuznetsov, M.Novicov, A.Turbin. *Semimarkovian models of systems with repair and queueing systems.* Shtiintsa, Kishinev, 1991 (in Russian)

[2] V.Korolyik A.Turbin *The Markov renewal processes in reliability problems of systems.* Nauk. dumka, Kiev, 1982. (in Russian)

THE AUTOMATIZATION OF DECISION SUPPORT PROCESS FOR SOFTWARE QUALITY ANALYSIS

Viorica Madan **Maia Ungureanu**

Software Department, Institute of Mathematics, Academy of Sciences of Moldova
5 Academiei Street, Kishinau, THE REPUBLIC OF MOLDOVA

Abstract. In the last 10 years, an increasing number of measures have been developed in software quality analysis. The interactive decision support process is meant to provide assistance in selecting the most qualitative program for solving any particular problem. The notion of distance as mechanizm ensuring a quality control and classification of software is used.

Keywords: Software measures, Classification, Decision Maker (DM), Interactive Decision Support System (IDSS).

1. INTRODUCTION

An abundant growth of software means is noticed at present. The main idea of this article is to show what we can automatize the decision support process for software quality like an Interactive Decision Support System (IDSS). The role of the IDSS is to help the DM to:

- Analyse and give an expertyze of program quality;
- Choice of the qualitative program for solving his problem;
- Construct the optimal program.

The measurement of the program features has a capital importance for the assurance of software quality, for the processes of quality determination and evaluation , by facilitating software control during its entire life-cycle.

No less important is the task of assessment of the already existing programs as a large knowledge base concerning programs in the context of which each program would be characterized by a diversity of quality measures The Interactive Decision Support System includes:

- Analysts of programs displaying as many as possible characteristics of their qualities;
- Multicriterial analyser for optimal program determination (Pareto optimal set of programs [2]);
- Classificators of programs ensuring possibility to perform an optimal partition into classes and a rapid search of the required program;
- Programs controling the of the programming process (able to signal the existence of drawbacks, the level of closeness to the priori estimation, or to the "etalon" of the class) and displaying the program time qualitative profile (i.e. change of the program quality characteristics in case of debugging and testing).

All the above enumerated tools will help to exert a quality control over the software development and support processes.

2. PARTITION OF PROGRAM SETS INTO CLASSES

Let's have the following marking: **E** - stands for the analysed finite set of programs, where the programs are denoted by $x_1, x_2, ..., x_n$. **F** is the mark for the set of measures characterizing the quality of program. We assume $| \mathbf{F} |= p$ and mark the measures characterizing the x program as $x^1, x^2, ..., x^p$. $M_x[i] \subset \mathbf{R}$ is the ith component of M_x vector i.e. it is the assignment of the i-characteristic for the x program. The d distance between the correct programs for this space makes:

$$d(x,y) = \sqrt{\sum_{i=1}^{p}(M_x[i] - M_y[i])^2}$$

Within a claster analysis the d magnitude represents the dissimilarity measure [1]. This measure is used in the algorithm of the automatic partition and classification of n programs into nonintersecting classes.

3. THE STRUCTURE OF IDSS

Having done the partition of programs space into non- intersectional sets the DM can proceed to the multicriteria analysis of programs within each class [2] in order to find " the best" program (a kind of an "etalon") for this class. Different measures of program evaluation, can serve as criteria. Actually, the representative of the ith class is not by all means a real physical point but an averaged index of the class. Such a standart becomes the real program (or some programs) with the best quality coefficients. Before writing any program it is possible to predict its complexity by using the program specification. One can find different measures of kind in literature [3]. In case we fail to find an appropriate class for solving the given problem we procced to constructing an optimal program for this problem. The optimal program having been constructed it is included in the **E** spase of programs. This information is colected in the Program Data Base . Construction of automatic systems performing accounting and analysis of the existing program base as well as the program quality control at a developing stage are both necessary and possible. This can be the IDSS. The main subsystems of the Interactive Decision Support System are:

- Quality Analysis;
- Program Choice;
- Classification of Program Set;
- Program Constraction;
- Program Data Base.

The DM has acces at all these subsystems.

All enumetated above means will help to carry out a quality control during the processes of the software development and support processes.

4. REFERENCES

[1] E.Diday et al. Optimization et Classification Automatique. INRIA. Domaine de Voluceau. Rocquencourt. 1979.

[2] V.Madan, M.Ungureanu. Control of Software Quality. Studies in Informatics and Control. March 1992. vol.1. No. 1. Bucharest.

[3] Coulter, N.S., Cooper, R.B.,Solomon, M.K., Information Theoretic Complexity of Program Specification. Computer Journal, vol 30, No.3, 1987.

AUTOMATION OF MODELLING FOR PERSONAL HYBRID COMPUTER

Michael G. LEVIN, Olga M. PETROVA, Alexander M. BURCHAKOV, Igor A. BEIU

Moldova, Academy of Sciences, Institute of Power Engineering
5 Academy str, 277028, Moldova, Kishinev.

Features of a system software for the professional hybrid computer (PHC) [1] are presented in this paper. The core of PHC software is modelling automation system, including a number of subsystems and providing the experiments preparation and conducting.

The PHC intellectualization level is conditioned by automation degree of the analogue processors and system on the whole, by perfection of interactive means, control and hardware diagnostics automation, comfort of analog–digital programs debugging.

The necessity of the intellectualization of user–PHC relations processes manifests itself on the all stages of analog–digital problems preparation and solution, but two steps are of particular importance:

– input of initial mathematical description of a modelling object;

– calculation of the variables and time scales, transmission coefficient values of the analogue processor operational elements.

The importance of these stages are caused, in the first place, by difficulty of formalization and a great computational efforts and, in the second place, there is a necessity «to remove» analogue processors with its demerits upon model preparation from PHC users.

The PHC can include up to 3 MAP–45 with hard configuration, each of them is intended for modelling of objects, described by a system of differential equations of 6th order and a system of linear and non–linear algebraic equations.

The PHC system software has been realized for PHC architecture. Its core is a system of computer–aided simulation based on a principle of decomposition into subsystems. Functionally the subsystems are represented by three levels of program modules:

– the first (lowest) level provides communication with MAP–45 on the physical level;

– the second level is composed of programs forming the address, digital and control data;

– the third level includes programs, intended for communication with user; these have reference to the programs of the second and partially to the first level.

The computer–aided simulation system is designed on a module principle which allows to obtain an «open» system providing a rather simple substitution or inclusion of new program modules, if necessary.

The initial dialogue subsystem organizes user interface in regime of menu. The model description subsystem (MDS) is intended for the introduction of initial mathematical description of the modelling object. For this purpose the authors worked out specialized language HYBRID, with grammar rules similar to those of C language and permit to generate the language constructions describing ordinary differential equations in the Cauchy's form, linear and non–linear algebraic relations, functional dependencies and auxiliary information. The file, containing initial object's description in the HYBRID language can be created with assistance of any text editor (Fig.1).

```
@nam [Task_01.Section_X]      { Task name }

@dif [X]                      { Differential equations }
    d(X01) = -1.0*X01 + 25.0;
    d(X02) = -76.0*X01 + 1.0*X07;
    X01(0) = -5.5;  X02(0) = 0.0;
    -2.0 < X01 < +10.0;
    -5.9 < X02 < +9.9;
@alg [X]                      { Linear algebraic equation }
    X07 = +0.5*X02 - 12.0*X14;
    -7.5 < X07 < +7.5;
@mul [X]                      { Non-linear algebraic equations }
    X14 = 23.0*X01/X02;
@end [X]
```

Fig. 1

The translator, realized in C language, transforms this description into the internal format, creating the files «Initial problem» and «Control» (Fig.2).

The subsystem of computer–aided analog programming (SAP) reads the information from the file «Initial problem», calculates optimal scales values of the analogue variables, transmission coefficients of the MAP's operational elements and writes received information in the file «Computer problem». The problem of optimal scaling can be formulated as follows: maximize the sum of all peak voltages at the outputs of design elements of the analog unit with the weight coefficients with the limitations on the variables and time scales, summators and integrators transmission coefficients, which are determined by analogue equipment physical features and requirement to increase precision of modelling. These limitations can contradict each other; therefore, the algorithm of preliminary decision of optimal scales vector existence problem was suggested [2,3]. The optimal scaling problem is a problem of linear programming and can be solved by two author's algorithms, taking into account all peculiarities of this problem.

The subsystem of automatic commutation (SAC) reads data of model from the file «Computer problem» and, interacting with MAP—45 hardware by means of the interprocessors exchange subsystem (IES), makes a complete automatic setup of the design simulation.

The subsystem of control with decision of the analogue part (SCD) through IES gets access to the all operational elements, control blocks and time service (initial information for SC is stored in the file «Control»). The results of modelling are written into the files «Tables» and «Graphics».

The subsystem of graphic data (SGD) provides simulation design of graphs, phase—plane picture and other data output onto the screen of display or printer.

The tests subsystem (ST) allows to check the capacity for work of the analogue coprocessor and system on the whole in the static and dynamic regimes.

The extension of the system software is possible by means of the additional subsystems inclusion. Now the work on creation a bank of analog—digital models for energetical electronics devices is under progress.

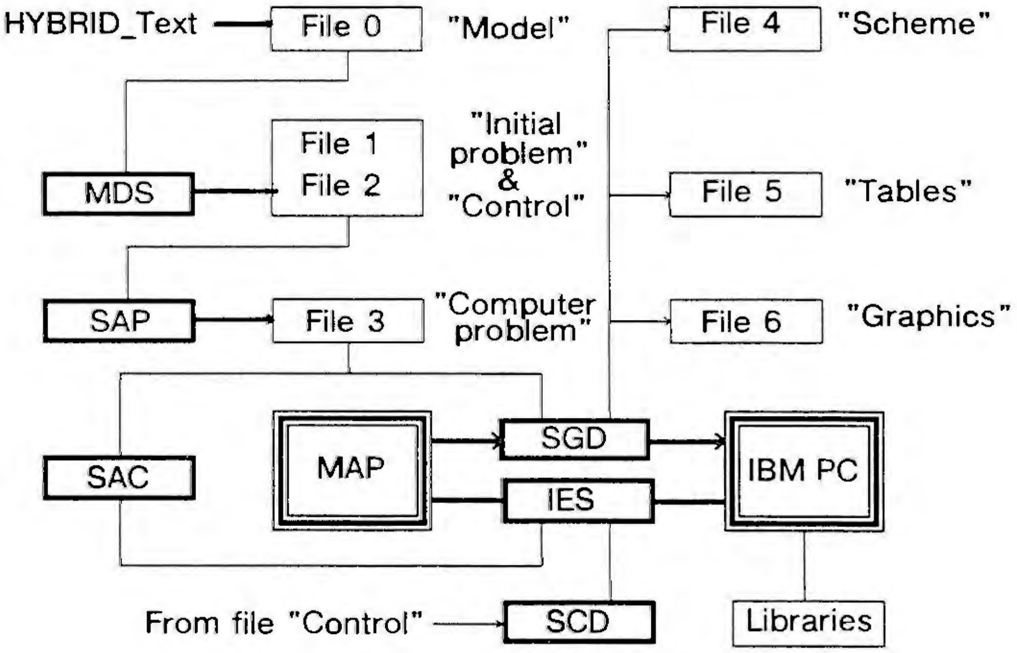


Fig. 2

REFERENCES

[1] Levin, M.G., Petrova, O.M., Zhuravlev, A.A. Architecture of the Analog—Digital Computational System with a Matrix Organization of the Analogue Processors. This volume.

[2] Levin, M.G., Petrova, O.M. Solution of an Optimal Scaling Problem for a Matrix Analog Processor. Advances in Modelling & Simulation, AMSE Review, Vol.19, 3 (1991), 19—23.

[3] Shor, I.Ya., Levin, M.G., Burchakov A.M., Petrova, O.M. System Software for a Professional Personal Hybrid Computer. Advances in Modelling & Simula—tion, AMSE Review, Vol.16, 2 (1989), 23—33.

ARCHITECTURE OF THE ANALOG-DIGITAL COMPUTATIONAL SYSTEM WITH A MATRIX ORGANIZATION OF THE ANALOGUE PROCESSORS

Michael G. LEVIN, Olga M. PETROVA, Anatoly A. ZHURAVLEV

Moldova, Academy of Sciences, Institute of Power Engineering
5 Academy str., 277028, Moldova, Kishinev.

Features of the professional high—performance hybrid computer, based on the IBM PC family and original matrix analogue co—processor, are presented in this paper.

We shall define the personal hybrid computer (PHC) as the desktop inhomogeneous computer system based on personal digital computers and small—scale computer—aided analog co—processors designed for personal computations and simulation of the most prevailing continuous and continuous discrete processes to be carried out by non—programming professionals.

The productivity of analogue processor presenting the main calculational power of each hybrid computational system in considerable degree is conditioned its ability to execute commutation of the operational elements, puttings the required values to transmission coefficients , initial conditions and etc. according to program. Without full automatic modelling scheme setup, the PHC creating is impossible. «Free setup» programming technology based on the matrix—module or matrix—commutational principles of designing a computer—aided analog processor is elaborated by the authors [1].

The matrix—module principle is based on the merging a series of matrix analog processors—modules with fixed internal communications by means of autocommunication of intermodule connections between main variables. Each matrix processor—module is supplemented through a series of small—scale intramodule commutators completed with nonlinear units and comparators which lead to the increase of the columns and rows in a matrix. This principle allows to bring down to minimum the number of switches in a computer—aided processor and to decrease maximally the crossing communications between the design units via analog gates, which results in the increased accuracy and reliability of a processor while decreasing by few tens the number of switches. However, it requires application of additional digitally controlled potentiometers.

The matrix—commutational principle allows to decrease the number of potentiometers through elimination of a full matrix of coefficients stipulated by matrix in real—time problems.

These principles were taken as a basis when creating PHC with inhomogeneous matrix analogue multiprocessor (MAP—45) and digital IBM PC family computer. The matrix integrating analogue processor (MIP), linear and nonlinear analogue processors (LAP and NAP), functional analogue processor (FAP) and parallel logical processor (PLP) are united by commutational analogue processor (CAP) in the analogue multiprocessor (AMP). AMP is completed with digital control device (DCD), time and interrupt service (TIS), processor of analog to digital and digital to analog conversion of information (PCI), digital RAM, input and output registers (RGI and RGO) forming functionally complete analogue multiprocessor MAP–45 (Fig.1).

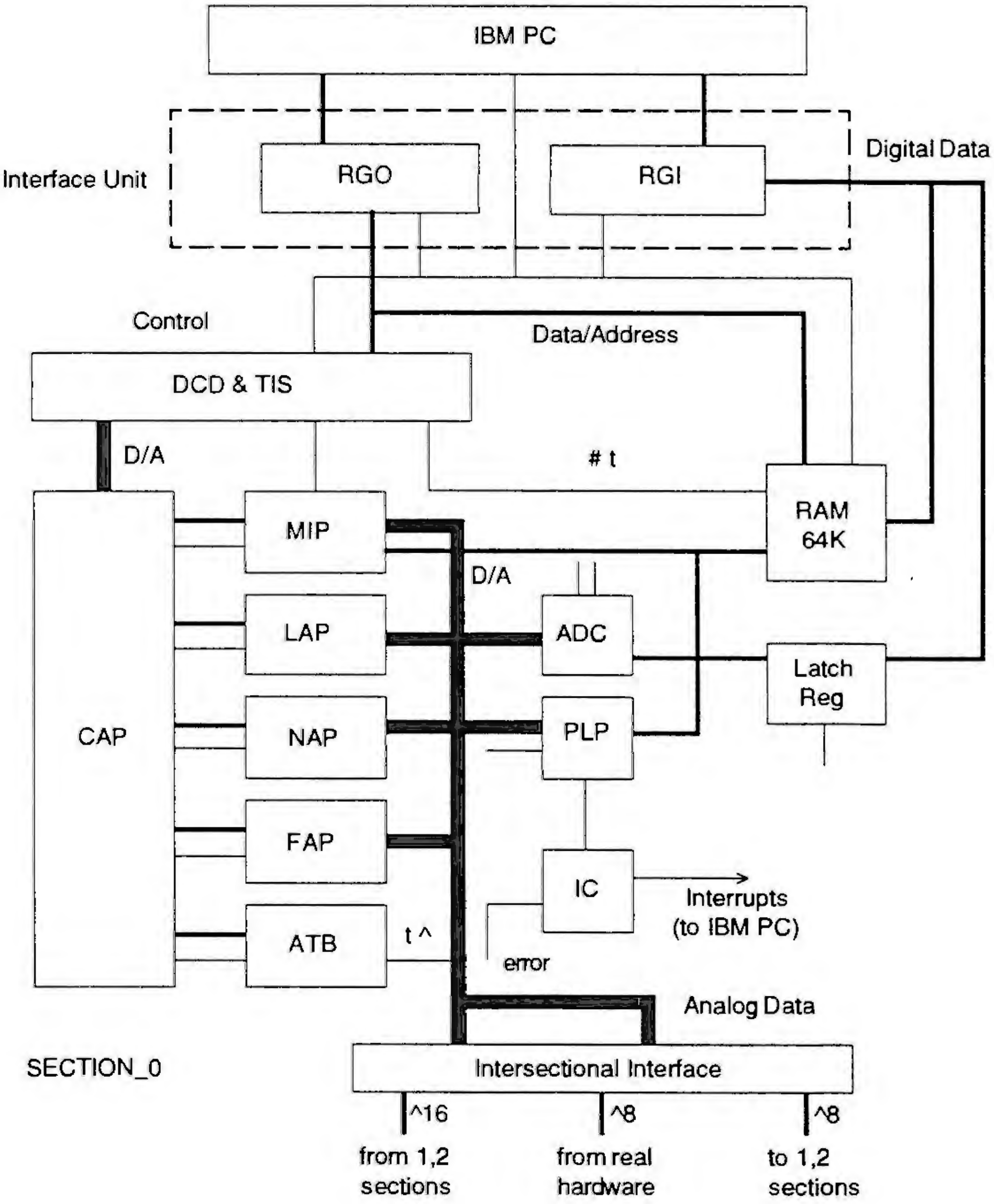


Fig. 1

MAP — 45 operational elements are connected to each other by means of automatic commutation elements, forming the analog circuit for the complete set of nonlinear differential equations system of 6th order and linear and non — linear algebraic equations system.

The important features of MAP — 45 are the presence of internal working memory, which makes it possible to store up to 128 values for each of 29 variables, generated by the operational elements, and the extension of variable set due to the addition of the analog equivalent for the current time, leading to possibility to solve a more general class of problems.

There is an additional possibility to increase the order of the considered differential equations by means of connection of 3 MAP — 45 co — processors. MAP — 45 enables to communicate with real hardware through 8 separate channels.

The procedure of modelling taken one's bearings on the «free setup» programming technology of MAP — 45 was elaborated by the authors, the ways of organization of effective analog — digital computing processes in the one class automated systems of scientific research of continuous — discrete power engineering electronics devices were proposed and investigated.

REFERENCES

[1] Analog — digital matrix processors / Zhuravlev, A.A., Levin, M.G., Shor, I.Ya., Trahtenberg, A.S. Kishinev. 1989.

MATHEMATICAL MODELLING OF COMPOSITE MATERIALS MANUFACTURING PROCESSES

Yury I. DIMITRIENKO

Professor, Scientific Research & Production Corporation "NPO Mashinostroeniya"
Gagarina 33a, Reutov, Moscow region, 143952 Russia
FAX: (095)-302-20-01, E-mail: DIMIT@COMPNET.MSU.SU

Abstract. In this paper a new mathematical model for production processes of composite materials on polyimide, carbon and ceramic matrixes is developed. The model is based on the idea of combination of three independent before theories: structural mechanics of composites describing the dependence of material characteristics on structural parameters, consolidation theory considering conditions of liquid and gaseous to solid phases transition and their features, and heat-mass-transfer theory in multiphase media modelling technological processes in composites. The mathematical problem statement has been given, and its applications for specific types of composites and some computed results are shown.

1. INTRODUCTION

Prediction methods existing up to now don't allow to perform completely the problem on designing new materials with properties given before, for example with a given strength, stiffness, thermostability etc. One of the principal reasons of this fact is the absence of mathematical models taking account of formation and development of microdefects of a material structure (cracks, pores etc.) and also the effect of technological factors on a change of structure and properties of composite materials.

2. MODELLING OF A COMPOSITE STRUCTURE

For composite materials on polyimide, carbon and ceramic matrixes independently of a production technology the same model of a structure can be formulated applicable practically for all up-to-date structures of reinforcing by fibers.

Composite material is considered to be a multilevel structure in the overwhelming majority of cases, with six structural levels (Fig.1). Each n-th structural level is represented by a collection of repeating structural elements being periodicity cells of the n-th type (PCn) n=1,...6. Periodicity cells are domains of definition for boundary problems of the special type the solution of which allows to define material properties at the n-th structural level by properties of components at the n-th level. As such properties only mechanical features will be considered (strength and stiffness).

At the first structural level there is a composite material as the whole having either a fabric structure or a layer-fiber two-directional structure produced by winding, or multidirectional structure. For $2D$ winded composite a periodicity cell consists of S layers of unidirectional material, the layers are laid at different angles Φ_s in planes orthogonal to the same direction. Periodicity cell of a multidirectional composite consists of bundles of unidirectional material guided in S different directions (usually in three, four or five ones). For a fabric composite its periodicity cell consists of s layers placed in the orthogonal direction to the same Ox_1 direction, therewith each the layer represented by the element of the second structural level is formed by the system of curved threads of unidirectional material and consists of a collection of PC2. PC2, in its turn, consists of a continuous collection of components characterized by curving angle $\Phi(\xi^2_{(2)})$ in the certain plane $O\xi^1_{(2)}\xi^2_{(2)}$, $-\pi \leq \Phi \leq 0$, where $\xi^i_{(n)}$ are local coordinates in the periodicity cell of the n-th structural level. Each component from the continuous collection PC2 being the element of the third structural level, as components of periodicity cells 2D

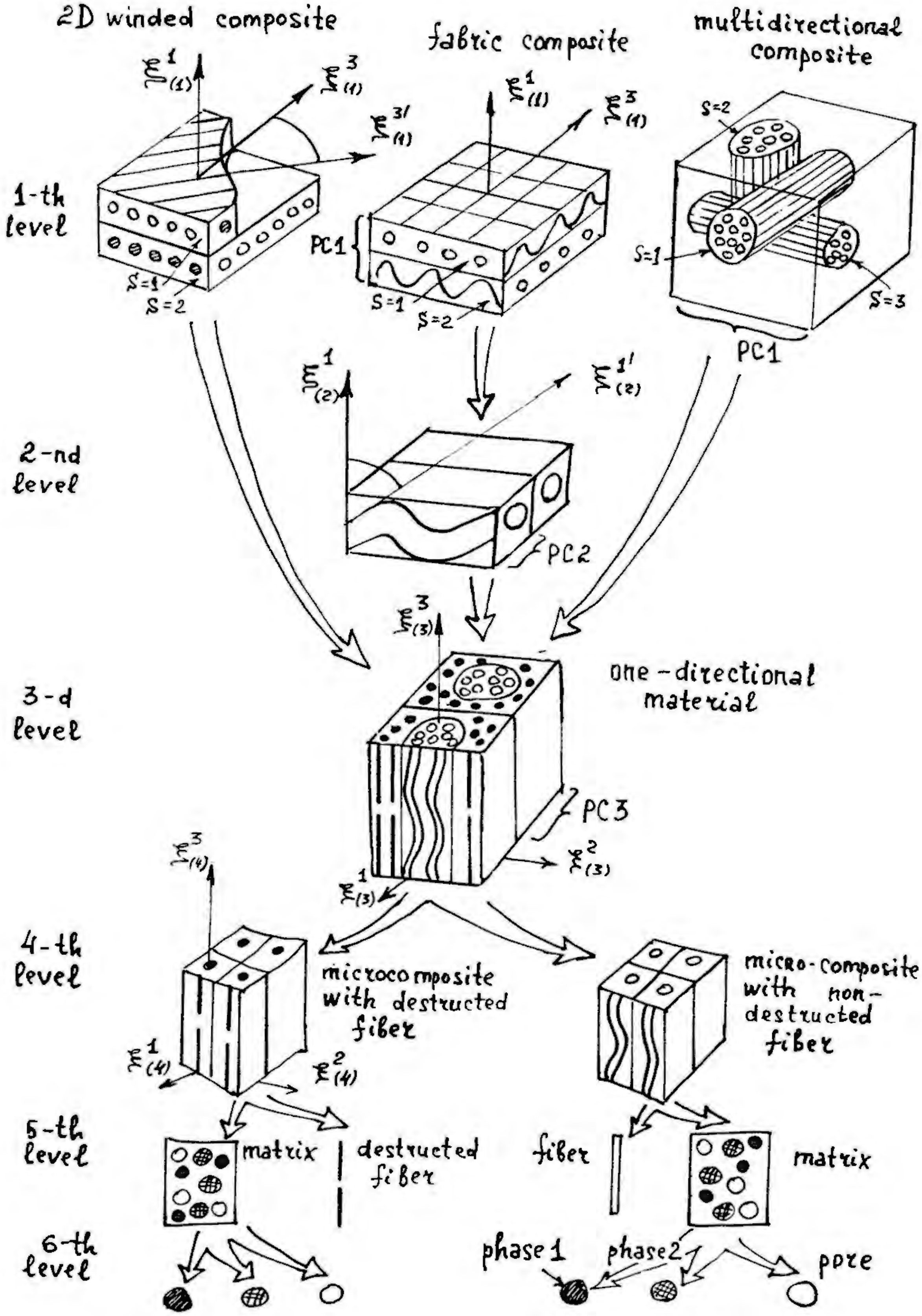


Figure 1. Model of a multilevel structure of composite material.

and multidirectional composite, is represented by unidirectional material.

Unidirectional material being, in essence, a thread of separate fibers consists of repiodicity cells PC3. According to the model developed in the paper this cell is formed by two components: microcomposite with a destructed fiber and microcomposite with non-destructed fiber but curved at the certain angle that is a technological defect of the thread.

Microcomposite with a destructed fiber is a collection of periodicity cells PC4$_d$, the component of which is a destructed fiber and its surrounding matrix. Microcomposite with a non-destructed fiber consists of PC4, the component of which is a non-destructed fiber and matrix being elements of the 5th structural level.

And finally a matrix undergoing phase transformations in technological processing consists of several solid phases, for example amorphic and crystalline phases, and also voids (pores, cracks).

Mechanical properties of these phases and also all fibers are described by the model of a linear thermoelastic body. On solving successively boundary problems for the periodicity cells of the 6th and then 5th etc. levels mechanical features of the whole composite are determined which correspond to the model of nonlinear-viscoelastic medium of the non-stable type:

$$\varepsilon = F\Big(\sigma(\tau), \theta(\tau), \varphi_i(\tau), z_s(\tau, \xi^j_{(n)})\Big)^{\tau=t,\ \xi^j_{(n)}=\xi^j_{(n)0}}_{\tau=0,\ \xi^j_{(n)}=0}, \tag{1}$$

where σ is a stress tensor, ε - strain tensor, θ - temperature, φ_i - volumetric concentrations of matrix phases, $\xi^j_{(n)}$ - local non-dimensional coordinates in PCn, t- time.

Viscoelasticity and non-stability of a composite are caused by phase transformations in its matrix in technological processing, and the cause of non-linearity is the presence of defects at the 3rd and 4th structural levels. At the 3rd level due to breakage of separate fibers a part of microcomposite with a destructed fiber φ_d gradually increases depending on stresses in fibers. Semilength of periodicity cell PC4 called non-effective fiber length depends on stresses in the destructed fiber that allows to simulate the effect of fibers' breaking in real composites. Complete destruction of the composite occurs when stresses in the matrix in PC4 or PC5 reach its strength limit, therewith damage parameter z_s reaches the value equal to 1:

$$z_s(t, \xi^j_{(4)}) = z_s\Big(\sigma_{(4)}(\tau, \xi^j_{(4)}), \theta_{(\tau)}\Big)^{\tau=t}_{\tau=0}, \tag{2}$$

where $\sigma_{(n)}$ are tensors of microstresses at the n-th structural level connected to σ by functionals of stress concentrations $\Upsilon_{(n)}$:

$$\sigma_{(n)}(\tau, \xi^j_{(n)}) = \Upsilon_{(n)}\Big(\xi^j_{(n)}, \sigma(\tau), \theta(\tau), \varphi_i(\tau)\Big)^{\tau=t}_{\tau=0}. \tag{3}$$

3. MODELLING OF HEAT-MASS TRANSFER IN COMPOSITES' MANUFACTURING

Relations of composite mechanics obtained according to the model described in the above part contain the collection of structural parameters: φ_1 - volumetric part of fibers, φ_2 and φ_3 - volumetric parts of amorphic and crystalline phases in the matrix, φ_g, φ_l - volumetricparts of gas and liquid phases placing in pores, and also φ_d - part of destructed fibers, φ_s - part of threads in the sth direction, therewith $\varphi_1 + \varphi_2(t) + \varphi_3(t) + \varphi_g(t) + \varphi_l(t) = 1$, $0 \le \varphi_d(t) \le 1$, $\sum_s \varphi_s = 1$. Presence of a greater number of matrix solid phases is possible.

In technological processing structural parameters change with time that is described by the following system of heat-mass-transfer equations:

$$\frac{\partial \varphi_1}{\partial t} = 0; \qquad \rho_i \frac{\partial \varphi_i}{\partial t} = J_{gi} - J_{ig} + J_{li} - J_{il} + (J_{32} - J_{23})(-1)^i, \quad i = 2, 3;$$

$$\frac{\partial \rho_j \varphi_j}{\partial t} = \mathrm{div}(K_j \mathrm{grad} p) + J_{2j} - J_{j2} + J_{3j} - J_{j3} + (J_{lg} - J_{gl}), \quad j = l, g;$$

$$\rho c \frac{\partial \theta}{\partial t} = \mathrm{div}(\lambda \mathrm{grad}\theta) - (c_g K_g + c_l K_l)\mathrm{grad} p \ \ \mathrm{grad}\theta + \\ + J_{2g}\Delta e^0_{2g} + J_{3g}\Delta e^0_{3g} + J_{lg}\Delta e^0_{lg}, \tag{4}$$

where ρ_i - phase densities, $p = R\rho_g\theta$ - pressure in gas and liquid phases, K_l, K_g - gas-permeability and filteration tensors, λ - heat-conduction tensor, c_i - heat capacities, J_{ij} - mass-transfer intensity, Δe^0_{ij} - heat effects of phase transformations, θ - the same temperature for all phases.

4. TECHNOLOGICAL STRESSES

Calculation of technological stresses σ in a composite is based on the solution of the equilibrium equations system:

$$\mathrm{div}(\varphi_s\sigma) - \mathrm{grad}(\varphi_g + \varphi_l)p = 0; \quad \varphi_s = 1 - \varphi_g - \varphi_l, \tag{5}$$

whereinto the relation (1) should be substituted for σ and deformation should be expressed in terms of displacement vector: $\varepsilon = (1/2)(\mathrm{grad}\vec{u} + (\mathrm{grad}\vec{u})^T)$.

5. EXAMPLE OF COMPUTED RESULTS

In Figure 2 the dashed curve shows the temperature technological regime of heating the cylindric shell made of fabric composite on carbon matrix. The first peak of heating corresponds to the stage of matrix solidification, i.e. transition of liquid to solid (amorphic) polymer state, and the second peak of heating corresponds to the stage of matrix carbonization and amorphic to new solid (crystalline) state transition. Solid curves show changes of elasticity modules E_1 and E_2 and Poisson coefficients ν_{23} in manufacturing processes calculated according to the mathematical model above-developed.

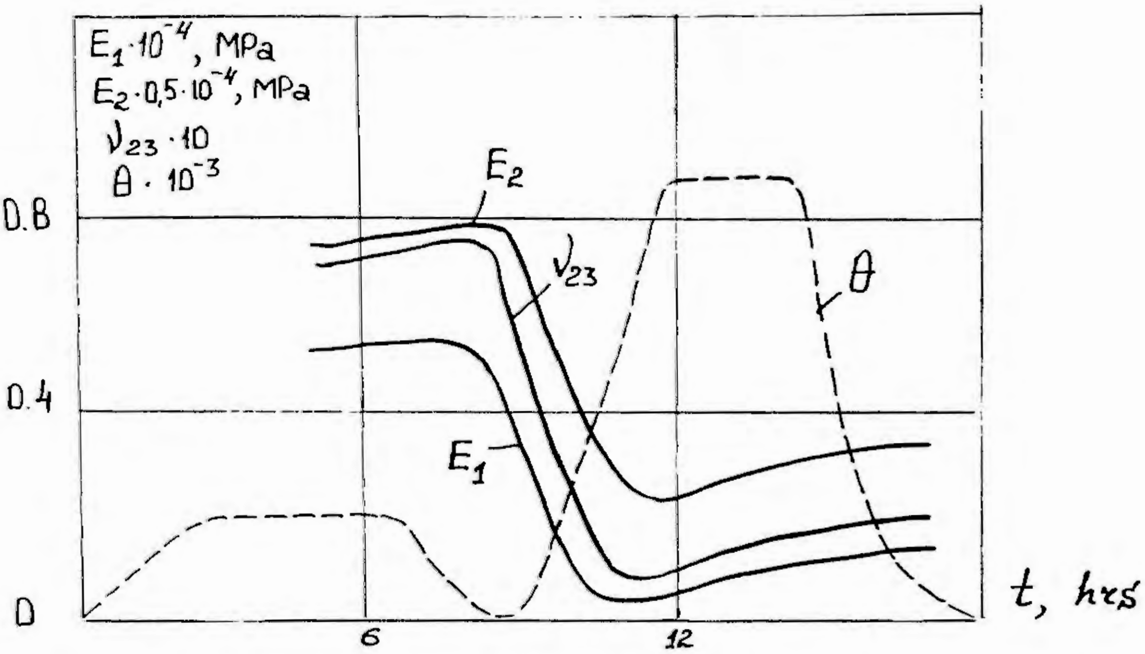


Figure 2. Computed results for changing mechanical features of carbon-carbon composite material in technological processing.

6. CONCLUSION

Numerical investigations conducted allow to establish that the mathematical model developed describes adequately enough changes of composite properties in technological processing.

7. REFERENCES

[1] Dimitrienko Yu., Technological stresses in carbon-carbon composite materials. Mechanics of Composite Materials (Riga), 1 (1992), 43-55.

MATHEMATICAL MODELLING OF FRAGMENTATION OF THIN SHELLS IN EXPLOSION

A.B.KISELEV

Department of Mechanics and Mathematics, Moscow M.V.Lomonosov State University
Moscow, 119899, Russia

Abstract. A simple one-dimensional mathematical models which allow to calculate a number of fragments of thin elastoviscoplastic cylindrical and spherical shells and their initial velocity in internal explosion are present in this paper.

1. INTRODUCTION

Increase of orbital debris in consequnce of explosion of spacecraft and their elements is one of main sources of obstruction of spase near Earth. Before the estimations of number of fragments in explosion of spacecrafts, distribution of their mass and velocity were empirical, on the basis of data of separate experiments on the Earth or on orbits (such the well-known models DEBRIS, EVOLVE, PIB, IMPACT). In this paper presents a simple mathematical models which allow to calculate a number of fragments in explosion of spacecraft and their initial velocity.

2. SETTING TO A PROBLEM

The next simplifical assumptions were done:

1. The body of spacecraft is simulated by cylindrical or spherical shell.

2. The shell is thin: $h/r \ll 1$ (h - thickness, r - radius of shell).

3. Influence of inside explosion is simulated of pressure $p = p(t)$, which depends on time and uniform distribution along inside surface of shell. Characteristic time of action of loading is $\tau \gg h/a_0$ (a_0 - speed of sound in material of shell).

4. Material of shell is considered as elastoviscoplastic and process of its deformation is adiabatical [1].

5. A condition for the specific (per mass unit) dissipation D to attain some limit value D_* is adopted as criterion of beginning of failure:

$$D = \int_0^{t_*} \frac{1}{\rho}\, d\, dt = D_*.$$

Here t_* is the time of beginning of failure, D_* is the material constant which determined experimentally of investigating the flat collision of two plates with break-off failure in a plate target, ρ is the density, d is the dissipation [2].

6. It is assume that break-up of shell is the result of circular tensile stresses at the expense of expenditure of supply elastic energy which was accumulated in shell to moment $t = t_*$ of beginning of failure [3].

As the result were obtain formulas for the number of fragments N. For cylindrical shell

$$N = [\frac{\pi \rho_0 d_0 E_*}{\gamma}],$$

Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX where ρ_0 is initial density, d_0 is initial diameter of shell, γ is the specific (per area unit) energy which spend to form unit of failure surface, E_* is density of accumulating elastic energy to moment $t = t_*$, the mark [ ] is the whole part of number. For spherical shell the number of fragments is

$$N = [\pi k(\frac{\rho_0 d_0 E_*}{\gamma})^2],$$

where k is form coefficient ($k = s/p^2$, s is area of external surface of fragment, p is its half-perimeter).

3. RESULTS

Many calculations were carry out for shell with varios h and d_0 and varios dependence $p = p(t)$ which is sumulate influence of powerful and weak explosions. Some results of calculations are present in table 1 (cylindrical shell) and table 2 (spherical shell). Here $(\varepsilon_\theta^p)_*$ is plastic deformation at moment $t = t_*$, l is typical size of fragment ($l = 2\pi r/N$ for cylindrical shell and $l = 4r\sqrt{\pi k/N}$ for spherical shell), $k = 0,2$, material of shells is alluminium alloy, $r = 1,5m$, $h = 3mm$, $p = p_0/(1 + t/\tau)^3$, p_0 is initial pressure, τ is typical duration of loading.It is show that number of fragments N and their velocity V_0 is function of impulse of pressure

$$I_* = \int_0^{t_*} p(t)dt.$$

The dependences $V_0 - I_*$, $N - V_0$ were constracted. They satisfactory agree with experemental data ([4-6] etc.).

p_0, GPa	τ, mc	N	l/h	$V_0, m/c$	t_*, mc	$(\varepsilon_\theta^p)_*$	$I_*, KPa \cdot c$
0,01	1,00	979	3,21	439	1,640	0,207	4,28
0,01	10,00	990	3,13	1057	0,951	0,189	8,31
0,01	100,00	992	3,17	1148	0,904	0,186	8,92
0,10	0,10	980	3,21	541	0,996	0,196	4,95
0,10	1,00	1026	3,06	2539	0,277	0,145	9,40
0,10	10,00	1044	3,01	3080	0,242	0,137	23,30
1,00	0,01	980	3,21	547	0,899	0,194	4,95
1,00	1,00	1242	2,53	7481	0,063	0,085	57,70
1,00	100,00	1268	2,48	7903	0,061	0,083	60,80
10,00	0,01	1181	2,66	6096	0,043	0,079	47,70
10,00	0,10	2009	1,56	18203	0,018	0,048	141,00
10,00	1,00	2213	1,42	21114	0,017	0,046	164,00

p_0, GPa	τ, mc	$N \cdot 10^{-3}$	l/h	$V_0, m/c$	t_*, mc	$(\varepsilon_\theta^p)_*$	$I_*, KPa \cdot c$
0,01	10,00	1097	1,52	597	0,688	0,0402	6,23
0,01	100,00	1097	1,52	653	0,660	0,0402	6,53
0,10	1,00	1098	1,51	2021	0,205	0,0400	15,50
0,10	10,00	1100	1,51	2409	0,187	0,0401	18,20
1,00	0,01	1098	1,52	447	0,415	0,0402	4,95
1,00	10,00	1257	1,41	7632	0,057	0,0366	56,50
5,00	0,01	1099	1,51	3105	0,078	0,0398	21,40
5,00	0,10	1670	1,23	12390	0,026	0,0273	93,30
5,00	1,00	1919	1,15	15104	0,023	0,0252	113,00
10,00	0,01	1173	1,47	6145	0,040	0,0357	47,50
10,00	0,10	2294	1,05	18170	0,017	0,0221	137,00
10,00	1,00	2613	0,97	20999	0,016	0,0210	157,00

Tables 1, 2.

4. ACKNOWLEDGEMENTS

This work has been supported by Russian Foundation of Fundamental Investigation (project code 93-013-16746).

5. REFERENCES

[1] Perzina, P., The Constitutive Equations for Rate Sensitive Plastic Materials. Quart. Appl. Mech., 20, N 3 (1963), 321-332.
[2] Kiselev, A.B., Yumashev, M.V., Deformation and Fracture under Impact Loading. Model of Damaged Thermoelastoplastic Medium. Prikladnaya Mekhanika i Teknicheskaya Fizika, 31, N 5 (1990), 116-123 (in Russian).
[3] Ivanov, A.G., About Possibility of Construction of United Theory of Rupture. Prikladnaya Mekhanika i Teknicheskaya Fizika, 31, N 1 (1990), 109-116 (in Russian).
[4] Banks, E.F., The Fragmentation Behaviour of Thin-walled Metal Cylinders. J. Appl. Phys., 40, N 1 (1969), 437-439.
[5] Olive, F., Nicaud, A., Marillean J., Loichot, Rupture Behaviour of Metals in Explosive Expansion in Mechanics Properties High Rates Strain. In: Mechanics Properties High Rates Strain. Proc. 2nd Int. Conf. Bristol and London, 1980, 242-251.
[6] Stelly, M., Legrand, J., Dormeval, R., Some Metallurgical Aspects of the Dynamic Expansion of Shells in Shock Waves and High Strain Rate Phenomena in Metals: Concepts and Applications. Proc. Int. Conf. Metallurgical Effects High- Strain-Rate Deformation and Fabrication. New York and London , 1981, 113-125.

MATHEMATICAL MODELLING OF DYNAMIC PROCESSES OF DEFORMATION AND MICROFRACTURE OF METALS AND SOLID FUELS

A.B.KISELEV and M.V.YUMASHEV

Department of Mechanics and Mathematics, Moscow M.V.Lomonosov State University
Moscow, 119899, Russia

Abstract. Two new models of deformable solids are being considered, which takes into account the accumulation of micro-structural damage in the material, in the process of dynamic deformation, the influence of damage to a stressed-strained state and the temperature effects. These models (the damaged and porous thermoelastoplastic media) serve as a description of deformation and the ductile failure of metals, solid fuels and explosives and belong to a class of models with internal variables. The entropy criterion of macro-failure and methods of concerning the parameters of the models are proposed. The testing of the models and the numerical solutions of some dynamic problems are discussed too.

1. INTRODUCTION

The evaluation of the durability of materials under intensive short-term stress is one of the basic problems of the mechanics of solids. Dynamic fracture is a complicated multistage process, including the appearance, development and confluence of microdefects and the formation of embryonic micro-cracks, their growth right up to the break-up of the bodies with their division into separate parts. Three basic types of dynamic fracture can be singled out: ductile, brittle and the mechanism of adiabatic shear failure. Ductile fracture, observed under normal conditions in metals, solid rocket fuels and explosives, are characterized by the appearance and development under plastic deformation of dispersed spherical micropores. A large number of orientared, coin-type micro-cracks, capable of growing in the process of deformation are formed in the brittle fracture of the material. Fracture of this type can be observed in berilium, concrete, mineral rock and certain types of steel. The mechanism of shear failure is observed under high speeds of deformation, for example, when a "plug" is forced out of the target. In this case the resulting hear is concentrated in thin layers with a thickness of up to several tens of a micron, are positioned along surfaces with maximum tangent stresses. This leads to the development of intensive plastic flow.

Below, two new models which describing the initial stages of ductile fracture (formation and growth of microdefects) are discussed. These models are geared towards modern usage of the numerical modelling of nonstationary non-homogenous processes of deformation and fracture of bodies in a complex stress-strained state.

2. MODELS OF CONTINIOUS DUCTILE FRACTURE

2.1. Model of the Damageable Thermoelastoplastic Medium.

This model belongs to the class of model media with internal variables, in which additional scalar or tensor variables of state, that characterise damages are introduced [1, 2]. An understanding of the measure of damage to material was first introduced in the works [3-5]. The scalar internal variables of state ω is used for models of the damageable media [6]. This describes the appearance and growth of the damaged material in the deformatory process(ω varies from 0 in an undamaged material to 1 in a

Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX

complete fracture). Let is be assumed that the full deformation ε_{ij} can be expessed in the form of the sum $\varepsilon_{ij} = \varepsilon^e_{ij} + \varepsilon^p_{ij}$, where ε^e_{ij} - elastic deformations and ε^p_{ij} - plastic deformations, while: $\varepsilon^p_{ij} = 0$.

Returning to the heat equation and the second law of thermodynamics, expressed in the form of the Clausius-Duhame inequality we get

$$\dot{\eta}T = (\frac{1}{\rho}\sigma_{ij} - \frac{\partial F}{\partial \varepsilon^p_{ij}})\dot{\varepsilon}^p_{ij} - \frac{\partial F}{\partial \omega}\dot{\omega} - \frac{1}{\rho}div\vec{q}, \tag{1}$$

$$d = d_M + d_F + d_T \geq 0,\ d_M = (\sigma_{ij} - \rho\frac{\partial F}{\partial \varepsilon^p_{ij}})\dot{\varepsilon}^p_{ij},$$

$$d_F = -\rho\frac{\partial F}{\partial \omega}\dot{\omega},\ d_T = -\frac{\vec{q}\ gradT}{T},\ \sigma_{ij} = \rho\frac{\partial F}{\partial \varepsilon^e_{ij}},\ \eta = -\frac{\partial F}{T},$$

where U - specific internal energy, ρ -density, σ_{ij} - components of the stressed tensor, $\vec{q}$ - heat flow, η-specific entropy, T - absolute temperature, d_M - mechanical dissipation, d_F - dissipation of continuum fracture, d_T - thermal dissipation, $\tau_{ij} = \sigma_{ij} - \rho\frac{\partial F}{\partial \varepsilon^p_{ij}}$ - tensor of "active" stress.

In the frameworks of the linear thermodynamics, with assumptions of small elastic deformation and the nonnegativity of each of the components of functions of dissipation; introducing a specific heat capacity under constant stress c_σ, accepting that module K and μ depend on the variables of damage ω in the following way:

$$K = K_0(1-\omega),\ \mu = \mu_0(1-\omega), \tag{2}$$

where K_0, μ_0 - are the module of the undamaged material, assuming that the behavior of the material can be described by a flow equations with Mizes' criteria of plasticity, and that the variable of damage ω is expressed by a kinetic equation of the Tooler-Butcher type, finally we end up with the following system of constitutive equations:

$$\sigma' = K_0\varepsilon_{kk} - \alpha_V(T - T_0) + \frac{\Lambda}{3}\int_0^\omega \frac{\partial\dot{\omega}}{\partial\omega}d\omega, \tag{3}$$

$$(\tau'_{ij})^\nabla + \lambda\tau'_{ij} = 2\mu_0(\dot{\varepsilon}_{ij} - \frac{1}{3}\dot{\varepsilon}_{kk}\delta_{ij}),\ \tau'_{ij}\tau'_{ij} \leq \frac{2}{3}Y^2,$$

$$\rho c_\sigma\dot{T} + \alpha_V\dot{\sigma}T = \tau_{ij}\dot{\varepsilon}^p_{ij} + \Lambda\dot{\omega}^2 + div\vec{q},$$

$$\dot{\omega} = B(\sigma' - \sigma_*)^m H(\sigma' - \sigma_*),\ \tau'_{ij} = \frac{\tau_{ij}}{1-\omega},\ \sigma' = \frac{\sigma}{1-\omega},$$

where $\sigma = \sigma_{kk}/3$, $H(x)$ - function of Heaviside, B, m, σ_* - constants of the material. The symbol ∇ designates Jauman' time derivation. Here,the yield strength Y and the shear modulus μ depend on temperature, pressure and other variables of state [7].

Model (3) generalizes the Prandtler-Reuss model of elastoplastic flow and takes into account the anisotropy of plastic deformation (in the case where $\Gamma \neq 0$), the accumulation of damage in area of intense tension, the effects of the processes of the deformation and accumulation of micro-structurial damage, and termal effects. Model (3) is used to express the behavior of metals [8].

2.2. Model of Porous Thermoelastoplastic Medium

The model of porous thermoelastoplastic medium [9] is suggested to explain the dynamic behavior solid rocket fuel and explosives, which even in their initial condition have scattered micropores.The system of constitutive equations of the porous medium turns out to be analogous to that of the model of the damaged medium (3), if intend of the variable of damage ω the variable of porosity α ($0 \leq \alpha \leq 1$) is inserted - volumetric total contents of micropores (the voids in the materials). As a kinetic equation for variable a the equation of the ductile growth of pores is used, taking into account the influence of its gases [10]:

$$\frac{\dot{\alpha}}{\alpha} = \frac{\sigma - \sigma^+}{4\eta}H(\sigma - \sigma^+) + \frac{\sigma - \sigma^-}{4\eta}H(\sigma^- - \sigma), \tag{4}$$

$$\sigma^{+} = -\frac{2}{3}Y\ ln\alpha - p_0(\frac{\alpha_0}{\alpha})^k,\ \sigma^{-} = \frac{2}{3}Y\ ln\alpha - p_0(\frac{\alpha_0}{\alpha})^k.$$

Here η is the dynamic ductility of the material, α_0 - initial porosity, p_0 - initial pressure of the gas in a pore, k - index of the adiabate of the gas. The first term in (4) explains the process of the expansion of micropores, the second it's plastic swelling.

3. CRITERION OF BEGINNING OF MACROFRACTURE

Criterion of beginning of macrofracture (the origin of cracks - the new free surface in the material) is the condition for the achievement of the specific dissipation of maximum meaning D_* [6, 11]:

$$D = \int_0^{t_*} \frac{1}{\rho}(d_M + d_F + d_T)dt = D_*.$$

Here t_* - time of fracture, D_* - the constant of the material, experimentally defined.

Tnis criterion may be referred to the class of entropy criteria of failure. Such a criterion makes it possible to describe in principle the process of failure using the mechanism of cumulative microstructural damages occurring, for instance, at break-off failure in tension waves (in this case a decisive contribution to d is made by the term $d_F = \Lambda\dot{\omega}^2$ the power of continual failure dissipation along with the power of mechanical dissipation $d_M = \tau_{ij}\dot{\varepsilon}^p_{ij}$. Use can also be made of the mechanism of shear failure which is the case, for example, in problems of punching plate targets of a finite thickness with a flat-face striker. In this particular case, narrow zones of intensive adiabatic shear are known to develop in the target in places of stress concentration. The work of plastic deformations converts almost completely to the heat that, because of high local deformation velocities, has no time to extend over any significant distance from the zones of developed plastic deformations. As a result, the temperature in the zones rises and great thermal gradients occur which causes an additional plastic flow and a further concentration of local plastic deformations and eventually forces a "plug" out of the target. At shear failure, a decisive contribution to dissipation d is made by the terms $d_M = \tau_{ij}\dot{\varepsilon}^p_{ij}$ and $d_T = -\frac{\vec{q}\ gradT}{T}$. The latter is the power of thermal dissipation and is given by $d_F = \varkappa(gradT)^2/T$ in the case of the Fourier law of heat conduction $\vec{q} = -\varkappa gradT$.

4. CONCERNING THE PARAMETERS OF THE MODELS

The material characteristics are selected from the experimental data from the flat collision of two plates with results of numerical modelling [6, 11]. The deformation anisotropic parameter Γ can be defined from the experiments of tension-pressure or normal shear [12]. In future calculations let $\Gamma = 0$ given the absence of necessary experimental data for the investigation of material. In particular experiments [13] for the break-off fracture of a $10mm$ titanium alloy targets which impact from $2mm$ aluminum plates with a wide range of velocities were used. Numerical investigation was carried out with an adiabatic approach. The significance of the limit of the strength $D_* = 75kJ/kg$ was defined for titanium alloys.

For the definition of porous, thermoelastoplastic model parameters, the problem of the impact-compression of micropores [9] is used. This is used to define the interactive parameter Λ of deformation and micropore evolution (Λ is replaced by A in the equation for the model of porous media). Firstly the problem of the adiabatic compression of individual micropore with initial inner-radius a_0 and external radius b_0, with or without gas was solved. External pressure was defined as the following with t representing the duration of the process: $P(t) = P_0 H(\tau - t)$. From the average temperature of pores when $t = \tau$ can be defined:T_{av} in the case where $a_0 \neq 0$ and T^0_{av} in the case $a_0 = 0$ The increment of temperature $\Delta T_{av} = T_{av} - T^0_{av}$ was attained due to the porosity of the individual cell. The problem of dynamics of micropore can be solved with a one-dimensional approach. The gas in the pores is considered ideal. As constitutive equation for the material of pores the equation for thermoelastoplasticity is used [13] with the same coefficient of dynamic ductility η which is found in equation (4). Then the problem of plate collision with velocity V_0 is numerically solved. However in the plate of the investigated material initial porosity $\alpha_0 = (a_0/b_0)^3$ is introduced. The velocity of collision V_0 and the thickness of the striker h are selected so that the surface pressure equals p_0 and the duration τ. The selection a parameter A must make sure that the increment of surface temperature equals ΔT_{av}. Fof VRA-fuel we managed to attain the value $A : A = 5kPa \cdot s$. The results of the calculations correspond to experimental data and the calculations of other scientists.

5. ACKNOWLEDGEMENTS

This work has been supported by Russian Foundation of Fundamental Investigation (project code 93-013-16746).

6. REFERENCES

[1] Coleman, B.D., Gurtin, H.E., Thermodynamics with Internal State Variables. J. Chem. Phys., 47, N 2 (1967), 597-613.

[2] Kondaurov, V.I., Nikitin, L.V., Theoretical Principles of Reology of Geomaterials. Nauka, Moscow, 1990 (in Russian).

[3] Il'yushin, A.A., On a Theory of Long-term Strength. Izv. Akad. Nauk SSSR. Mekhan. Tverd. Tela, N 3 (1967), 21-35 (in Russian).

[4] Kachanov, L.M., On Failure Time under Fatigue. Izv. Akad. Nauk SSSR. Section of Engeneering Sciences, N 8 (1958), 26-31 (in Russian).

[5] Rabotnov, Yu.N., Fatigue of Structural Components. Nauka, Moscow, 1966 (in Russian).

[6] Kiselev, A.B., Yumashev, M.V., Deformation and Fracture under Impact Loading. Model of Damaged Thermoelastoplastic Medium. Prikladnaya Mekhanika i Teknicheskaya Fizika, 31, N 5 (1990), 116-123 (in Russian).

[7] Wilkins, M.L., Modelling the Behaviour of Materials. In: Structural Impact and Crashworthiness. Proc. Int. Conf. London and New York, 1984, V. 2, 243-277.

[8] Kiselev, A.B., Yumashev, M.V., Numerical Investigation of Dynamic Processes of Deformation and Microfracture of Damaged Thermoelastoplastic Medium. Vestnik Moscovskogo Universiteta, seriya matematika i mekhanika, 49, N 1 (1994), 66-72 (in Russian).

[9] Kiselev, A.B., Yumashev, M.V., Mathematical Model of Deformation and Failure of Solid Fuel under Impact Loading. Prikladnaya Mekhanika i Technicheskaya Fizika, 33, N 6 (1992), 126-134 (in Russian).

[10] Kiselev, A.B., Yumashev, M.V., Numerical Investigation of Impact Compression of Micropores in Thermoelastoviscoplastic Material. Vestnik Moskovskogo Universiteta, seriya matematika i mekhanika, 47, N 1 (1992), 78-83 (in Russian).

[11] Kiselev, A.B., Yumashev, M.V., On Dynamic Failure Criterion for a Thermoelastoplastic Medium. Vestnik Moskovskogo Universiteta, seriya mathematika i mekhanika, 45, N 4 (1990), 38-43 (in Russian).

[12] Bykovtsev, G.I., Lavrova, T.B., Model of Anisotropic Hardening Medium, which Have Different Hardening Laws at Tension and at Pressure. Izv. Akad. Nauk SSSR. Mekhan. Tverdogo Tela, N 2 (1989), 146-151 (in Russian).

[13] Kanel', G.I., Razorenov, S.V., Fortov, V.E., Break-off Strength of Metals in a Wide Range of Amplitudes of Impact Loads. Dokl. Akad. Nauk SSSR, 294, N 2 (1987), 350-352 (in Russian).

[14] Perzina, P., The Constitutive Equations for Rate Sensitive Plastic Materials. Quart. Appl. Mech., 20, N 3 (1963), 321-332.

Mathematical Modelling of the High-Velocity Penetration

Boris P. Rybakin

Institute of Mathematics Academy of Science of Moldova

Academy 5, Kishinev 277028, Moldova

E-mail: 31boris@mathem.moldova.su

Abstract. The given paper presents mathematical modeling of the process of the high velocity deformation of the finite-thick target when objects of different shapes strike it. The differential Eiller-Lagrange scheme is constracted, which makes it possible to calculate in two-dimension coordinates with cylindrical symmetry setting great deformations of elastic-plastic materials.

According to estimates of Russia's National Space Agency and NASA the Earth is orbited by about 70000 objects which, for same reason or other, have stopped their activity. These are disabled satellites,the last stages of carrier rockets, various fragments and debris that remained after unsuccessful or emergency launchings, etc. The ground equipment makes it possible to trace the objects whose effective reflecting surface is not less than 0.1 metre. They constitute about 1/3 of their total number and can be further prognosticated. The other objects are not seen from the Earth and there are only some probability characteristics of their location.

Besides, this "debris" is concentrated near inclination planes of orbits, which is typical for Baikonur and Canaveral launching sites. This leads to the appearance of a serious danger both for piloted ships and for non-piloted space vehicles. It is all the more complicated because these objects collide and this leads to their further defragmentation and redistribution of velocities of orbits' inclination angles.

Thus mathematical modeling of objects behavior during their collision is of great interest. The present work contains the results of numerical investigations of compact medium high-speed deformation processes occurring during fragments collision. The investigations were carried out in the 1 - 20 km/s range of collision velocities. In the same time we estimated the influence of the shape of colliding bodies on the processes of load and unload waves formation, the formation of the crater. Also studied was the distribution of the initial striker impulse on the axial and radial components in the target. For the sake of definitiveness we'll confine ourselves to the investigation of the impact of spherical, elliptical and cylindrical strikers upon the target which is a final thickness plate. The information obtained as a result of calculations may be used thereupon when calculating the durability of the cover construction which constitutes the space vehicle hull.

The interaction of the striker and target within the above-mentioned range of velocities is a complex process. In the initial time moments that follow the collision the pressure in the impact zone reaches very high values which leads to the melting or even evaporation of the substance. A little later the pressure decreases and the account for durable properties of materials comes in the foreground. The above-indicated factors should be given account to in a correct mathematical model of high-speed deformation.

In a two-dimensional axially symmetric system of coordinates the laws of mass conservation, of the quantity of motion and energy written for the substauce volume $\omega(t)$, restricted by the moving contour $\gamma(t)$, has the following form:

$$\frac{\partial}{\partial t}\int_{\omega(t)} \rho\, d\omega = \int_{\gamma(t)} \rho\,(W_n - D_n)\, d\gamma;$$

$$\frac{\partial}{\partial t}\int_{\omega(t)} \rho\, \overline{W}\, d\omega = \int_{\gamma(t)} \rho\, \overline{W}\,(W_n - D_n)\, d\gamma + \int_{\gamma(t)} p\, \overline{n}\, d\gamma;$$

$$\frac{\partial}{\partial t}\int_{\omega(t)} \rho\, E\, d\omega = \int_{\gamma(t)} \rho\, E\,(W_n - D_n)\, d\gamma + \int_{\gamma(t)} (p\, \overline{n}\, \overline{W})\, d\gamma;$$

here ρ - is the density, $\overline{W}$ - the velocity, $E = e + \frac{W^2}{}2$ - the specific full energy, p - pressure, W_n - the mass velocity of the substance, directed according to normal to $\gamma(t)$, D_n - the velocity of surface $\gamma(t)$, directed according to normal to it. The system of equations is completed by

$$p = p(\rho, E).$$

The collision of the striker and the target at high speeds results in substantial deformations of the substance. In this connection special care is required when choosing the difference scheme. The application of difference schemes, which use the Lagrange representation, is difficult since the difference net becomes highly distorted at strong deformations. The restructuring of such a net is quite a difficult procedure, moreover - not always possible.

In the schemes that use the Eiler system of coordinates there occur difficulties in tracing the free and contact boundaries and in fulfilling the corresponding boundary conditions on them.

The idea of the method of big particles (MBP) [1] was taken as the basis in this work. The MBP scheme was modified as follows: 1. the scheme was transformed into an invariant one in respect to coordinates transformation; 2. an additional algorithm was introduced for giving and tracing the contact and free boundaries during their movement on the motionless Eiler net [2]. This led to the appearance of additional calculation stages connected with the movement of boundaries, with giving and redetermining new boundary cells as well as satisfying the corresponding boundary conditions.

Fast Numerical Algorithms for Evaluating Characteristics of Priority Queueing Systems

Mishkoy G.K. and Grama I.G.

Institute of Mathematics
Academy of Sciences of Moldova
Academiei str.5, Kishinau 277028
Moldova

Analytical means

In this section we shall present some typical analytical formulas that arise in studying priority queueing systems $M_r|G_r|1$ with orientation. For more details we refer the reader to the book of Klimov, G.P. and Mishkoy, G.K. [1].

For the sake of definiteness we assume that the absolute priority is realized. This means that arrival of the higher priority requests in the system, busy with queuing orientation or servicing the lower priority request, interrupts both the orientation and the service. Let us assume that when the system becomes free of the higher priority requests the interrupted orientation will be continued, while the interrupted service is restarted and the orientation gets instantly annulled as soon as the busy period is completed.

Let us denote by $B_i(t)$, $C_j(t)$ and $\Pi(t)$ the distribution functions of duration of service of requests of the i-th priority class, duration of orientation of the device for servicing the requests of j-th class, $i \neq j$, $i,j = 1,...,r$ and busy period, respectively. Let a_i be the parameter of the arriving Poisson flow of priority i and $\sigma_k = a_1 + ... + a_k$, $\sigma_0 = 0$, $\sigma = \sigma_r$. Let $\beta_i(s)$, $c_j(s)$, $\pi(s)$ be the Laplace-Stieltjes transforms of the distribution functions $B_i(t)$, $C_j(t)$, and $\Pi(t)$ respectively.

Statement 1 *The Laplace-Stieltjes transform $\pi(s) = \pi_r(s)$ of the distribution function of a busy period is determined from the system of recurrent functional equations*

$$
\begin{aligned}
\sigma_k \pi_k(s) &= \sigma_{k-1}\pi_{k-1}(s + a_k) \\
&\quad + \sigma_{k-1}\{\pi_{k-1}(s + a_k(1 - \bar{\pi}_{kk}(s))) - \pi_{k-1}(s + a_k)\} \qquad (1) \\
&\quad \times \nu_k(s + a_k(1 - \bar{\pi}_{kk}(s))) + a_k \pi_{kk}(s), \\
\pi_{kk}(s) &= \nu_k(s + a_k(1 - \bar{\pi}_{kk}(s)))\bar{\pi}_{kk}(s), \qquad (2)
\end{aligned}
$$

$$\begin{aligned}\bar{\pi}_{kk}(s) &= h_{k-1}(s + a_k(1 - \bar{\pi}_{kk}(s))), && (3)\\ h_k(s) &= \frac{\beta_k(s+\sigma_{k-1})}{1 - \frac{\sigma_{k-1}}{s+\sigma_{k-1}}[1-\beta_k(s+\sigma_{k-1})]\nu_k(s)}, && (4)\\ \nu_k(s) &= c_k(s + \sigma_{k-1}[1 - \pi_{k-1}(s)]), && (5)\end{aligned}$$

where $k = 1, ..., r$ *and* $\pi_0(s) = 0, ...$.

The functions $\pi_k(s)$, $\pi_{kk}(s)$, $\bar{\pi}_{kk}(s)$, $h_k(s)$, $\nu_k(s)$ included in the above formulas are the Laplace-Stieltjes transforms of distribution functions of some supplementary random intervals having rather determined independent meaning.

We write down now expressions for the first moments π_{k1}, π_{kk1}, $\bar{\pi}_{kk1}$, h_{k1}, ν_{k11} of the above random intervals. Let $\rho_k = \sum_{i=1}^{k} a_i b_i$, where $k = 1, ..., r$,

$$b_1 = \frac{\beta_{11} + c_{11}}{1 + a_1 c_{11}}, \qquad b_i = \Phi_1 \dots \Phi_{i-1} \frac{1}{\sigma_{i-1}}[\frac{1}{\beta_i(\sigma_{i-1})} - 1](1 + \sigma_{i-1} c_{i1}),$$
$$\Phi_1 = 1, \quad \Phi_i = 1 + (\sigma_i - \sigma_{i-1}\pi_{i-1}(a_i))c_{i1}, \quad i = 2, ..., k,$$

and β_{k1}, c_{i1} are the first moments of the distribution functions $B_i(t)$, $C_j(t)$ respectively.

Statement 2 *If* $\rho_k < 1$ *then*

$$\sigma_k \pi_{k1} = \frac{\Phi_2 \dots \Phi_k + \rho_{k-1}}{1 - \rho_k}, \quad \bar{\pi}_{k1} = \frac{b_k}{1 - \rho_k},$$
$$h_{k1} = \frac{b_k}{1 - \rho_{k-1}}, \quad \nu_{k1} = \frac{\Phi_2 \dots \Phi_{k-1}}{1 - \rho_{k-1}} c_{k1}.$$

Numerical Algorithms

In this section we deal with the numerical methods for solving the system of functional equations (1-5). We need them in order to evaluate the Laplace-Stieltjes transforms $\pi_k(s)$, $\pi_{kk}(s)$, $\bar{\pi}_{kk}(s)$, $h_k(s)$, $\nu_k(s)$ and the first moments π_{k1}, π_{kk1}, $\bar{\pi}_{kk1}$, h_{k1}, ν_{k11} as well.

Let us observe first that in order to evaluate $\pi_k(s)$ for fixed k ($k = r, ..., 1$) and s we have to solve equation (3). For this a simple iteration process

$$\bar{\pi}_{kk}^{(0)} = 1, \qquad \bar{\pi}_{kk}^{(m)} = h_{k-1}(s + a_k(1 - \bar{\pi}_{kk}(s)^{(m-1)})), \qquad m = 1, 2, ... \quad (6)$$

allows to obtain any prescribed accuracy. The convergence of processes of such type is proved in Gnedenko, B.V. et al. [2].

¿From equations (1-5) it follows that at any iteration of the procedure (6) we have to evaluate two values $\pi_{k-1}(s_{k1}(s))$ and $\pi_{k-1}(s_{k2}(s))$ with

$$s_{k1}(s) = s + a_k, \quad s_{k2}(s) = s + a_k(1 - \bar{\pi}_{kk}(s)). \tag{7}$$

For the last the above reasons are available too but with new values $s = s_{k1}(s)$ and $s = s_{k2}(s)$ respectively.

The same situation arise when we want to evaluate all other characteristics. Such type of procedures has been used in Mishkoy, G.K. et al. [3]. where the software for queueing priority systems with orientation were elaborated. Unfortunately such type of procedures are very much time consuming especially for large r ($r \geq 5$).

In this paper we propose a new effective method for evaluating these characteristics by using a single iteration process.

In the sequel it will be convenient to make use of the following times

$$\begin{aligned} s(r,1) &= s, \\ s(k-1,2j-1) &= s_{r1}(s(k,j)), \\ s(k-1,2j) &= s_{r2}(s(k,j)), \end{aligned}$$

$k = r, ..., 1$, $j = 1, ..., 2^{r-k}$, that may be arranged in a binary tree. The main idea of this procedure is to solve the system of equations (1-5) for fixed r and s with respect to the collection of the unknown values $\bar{\pi}_{kk}(s(k,j))$, where $k = r, ..., 1$, $j = 1, ..., 2^{r-k}$. To this end the we define an iteration process each iteration m of which represents a recurrent procedure constructed by means of the above binary tree in the following manner

$$\begin{aligned} \sigma_k \pi_k(s(k,j)^{(m)}) &= \sigma_{k-1}\pi_{k-1}(s(k-1,2j-1)^{(m)}) \\ &+ \sigma_{k-1}\{\pi_{k-1}(s(k-1,2j)^{(m)}) - \pi_{k-1}(s(k-1,2j-1)^{(m)})\} \\ &\times \nu_k(s(k-1,2j)^{(m)}) + a_k \pi_{kk}(s(k,j)^{(m)}), \\ \pi_{kk}(s(k,j)^{(m)}) &= \nu_k(s(k-1,2j)^{(m)})\bar{\pi}_{kk}^{(m)}(s(k,j)^{(m)}), \\ \bar{\pi}_{kk}^{(m)}(s(k,j)^{(m)}) &= h_{k-1}(s(k-1,2j)^{(m)}), \\ s(r,1)^{(m)} &= s, \\ s(k-1,2j-1)^{(m)} &= s_{k1}(s(k,j)^{(m)}), \\ s(k-1,2j)^{(m)} &= s_{k2}(s(k,j)^{(m)}), \end{aligned}$$

where $k = r, ..., 1$, $j = 1, ..., 2^{r-k}$, $s_{ki}(s(k,j)^{(m-1)})$, $i = 1,2$ are determined by (7) with $\bar{\pi}_{kk}^{(m-1)}(s(k,j)^{(m-1)})$ instead of $\bar{\pi}_{kk}(s)$ and $h_k(s)$, $\nu_k(s)$ are determined by (4-5),

$m = 1, 2, \ldots$. The initial values for $\bar{\pi}_{kk}^{(0)}(s(k,j)^{(0)})$, $k = r, \ldots, 1$, $j = 1, \ldots, 2^j$ may be taken arbitrary in the interval $[0, 1]$.

This iteration process may be easily performed at the computer and turns out to provide very rapid calculations for the above mentioned characteristics even for enough large values of k ($k \geq 5$).

We have evaluated the function $\pi_k(s)$, for different values of k in the following table. The distribution functions $B_i(t)$, $C_i(t)$ are taken to be exponential with parameter 1 and $a_i = 1$, $i = 1, \ldots, k$. The first number inside each column represents the value of the characteristic $\pi_k(s)$, the second one being the processing time in msec. All calculations were performed at the computer IBM PC 386.

t	$k = 5$		$k = 7$		$k = 9$		$k = 10$	
.001	0.0308	33	0.0166	104	0.0104	363	0.0085	725
0.50	0.0257	27	0.0146	99	0.0094	368	0.0078	719
1.00	0.0219	22	0.0129	93	0.0085	363	0.0071	720
2.00	0.0164	22	0.0103	88	0.0071	363	0.0060	621
5.00	0.0085	22	0.0060	82	0.0045	308	0.0040	620
10.0	0.0039	17	0.0031	76	0.0025	263	0.0023	522
20.0	0.0015	16	0.0013	66	0.0011	263	0.0010	522

Bibliography

1. Klimov, G.P. and Mishkoy, G.K. (1979): *Priority service systems with orientations.* Moscow State University, 222 p. (Russian).

2. Gnedenko, B.V. etc. (1973): *Priority service systems.* Moscow State University, 446 p. (Russian).

3. Mishkoy, G.K. etc. (1990): *Software for priority systems with orientation.* Stiintsa, Kishinev, 190 p. (Russian).

THE "NESTED HASHING" METHOD APPLIED FOR SIMULATION OF CHAOTIC WALK

Yu.G.Gordienko and E.E.Zasimchuk

Institute of Metal Physics of the Ukrainian Academy of Sciences
36 Vernadsky Blvd, Kiev, 252142, Ukraine

Abstract.The new method ("nested hashing"), which substantially reduces the number of calculations for simulation of many phenomena related to chaotic walks of particles, is proposed.

1.INTRODUCTION

Many phenomena, like the size distribution of clusters and appearance of spatial inhomogeneity in plastically deformed crystals can be understood as the final product of a dynamical processes which include chaotic walk, chaotic generation, annihilation, agglomeration etc. Simulation of these processes can be achieved through very different algorithms using random walks of N particles on 2D lattice with $S \times S$ sites. Calculating new coordinates for particle and determining presence of empty site at the new position of particle are necessary stages of these algorithms. The latter often requires execution of many comparison operations and takes more time than the former does. Thus, search in large systems is a serious obstacle for investigation of their temporal evolution.

2.ORDINARY METHODS OF SEARCH

The primitive successive search can be carried out on the $2 \times S \times S$ matrix with $S \times S$ values of coordinates and $S \times S$ values of emptiness (or fullness) of sites. One discrete time step (i.e. moving all of particles for one spatial step) requires $C \sim S \times S$ operations (magnitude comparisons) and $M \sim S \times S \times (E+D)$ bytes of computer memory, where D and E are the length in bytes of the values of coordinates and emptiness respectively. Usually, $N \ll S$ and it is more convenient to use ciphered form to describe the coordinates. For example, for the lattice with S = 255 the ciphered form of particle coordinates X=00001101 (X=13 in decimal form) and Y=11111111 (Y=255 in decimal form) is XY=0000110111111111. This technique demands $C \sim N \times N$ comparisons (the number of other operations is negligible) and $M \sim 2 \times N \times D$ bytes of computer memory. However, the more substantial gain in speed can be obtained by means of hash-coding [1]. The hashing function H is supposed to provide uniform distribution of all particles in $G = \sqrt{N}$ groups which contain $P = \sqrt{N}$ particles with the same value of hashing function. For this method the number of calculations is $C \sim 2 \times N \times \sqrt{N}$, and the required memory is $M \sim (2 \times D + h)N$ bytes, where h is the length of hashing function value H(XY). But this method can be significantly improved.

3. "NESTED HASHING" METHOD

Applying the hashing procedure L to the G hashing function values H(XY) we can obtain the $\sqrt{G}$ new hashing function values L(H(XY)). Applying the hashing procedure R to each of the G groups of P ciphered coordinates XY we can get the $\sqrt{P}$ new hashing function values R(XY) for each group. If L and R meet the requirements offered for H then the number of calculations can be reduced up to $C \sim 2\times N\times(\sqrt{P}+\sqrt{G})$ at the expense of the more complex algorithm and larger required memory $M \sim (2\times D+h+l+r)$ of bytes, where r and l are the lengths in bytes of values R(XY) and L(h(XY)). After k repeatitions of these operations we can obtain the search tree with k hashing stages and the reduction in calculations up to value

$$C(k,N) \sim 2^{k}\times N^{2^{-k}+1}$$

with inevitable complication of algorithm and increase of required memory. The number of "nested hashing" stages (Kmin) needed for the minimal number of calculations (Cmin) for certain N can be easily obtained:

$$Kmin = \log_{2}(\ln N), \qquad Cmin \sim N\times\ln N\times N^{1/\ln N} = N\times\ln N\times e.$$

The "nested hashing" method can be compared with the binary search in ordering tables [1]. The number of operations required for realization of the binary search is $Cbin \sim N\times\ln N$. But for the "nested hashing" method all changes of particles' coordinates do not cause restructuring in the search tree, because all insertions related to particles' displacements are take place in "leaves". As a result the structure of the search tree remains constant. Moreover, the method proposed does not demand the magnitude ordering after one of the particles was displaced. The demerits of the method proposed are the large requested memory $M \sim (2\times D+(l+r)\times(2^{k}-1))$ and significant complexity of the algorithm.

This method is successfully used for simulation of behavior of vacancies during plastic deformation of crystals. The hashing functions values L(A) (and R(A)) are obtained by left (and right) masking the argument A (i.e. by extracting the left (and right) half of the argument A). In reality, the number of calculations is smaller than Cmin, because agglomeration causes decrease of the number of mobile particles. The advantages of the method allow to solve the above mentioned problems with computers possesing the modest computational resources (for example, on personal computers).

ACKNOWLEDGEMENTS

This work is supported in part by the Ukrainian State Committee for Science and Technology and Soros Science Foundation grant awarded by the American Physical Society.

REFERENCES

[1] Knuth, D.E., The Art of Computer Programming, Volume 3, Sorting and Searching. Addison-Wesley, Reading, MA, 1973.

ON A CLASS OF LIMIT RELIABILITY FUNCTIONS FOR SERIES–PARALLE AND PARALLEL–SERIES SYSTEMS

KRZYSZTOF KOLOWROCKI
Department of Mathematics
Maritime University
Morska 83, 81-962 Gdynia, Poland

Abstract. In the paper some ten-element closed classes of limit reliability functions for series–parallel and parallel–series systems with identical components are fixed. Next, the results are transfered to the systems with unalike components. These systems are such that at least the number of their series or the number of their parallel components tends to infinity.

1. INTRODUCTION

The results of the investigations of order statistics limit distributions are contained in the works of the following authors M. Frechet (1927), R. A. Fisher and H.C.Tippett (1928), E. J. Gumbel (1935), R. von Mises (1936) and B. W. Gniedenko (1943), L. de Haan (1970), N.W. Smirnow (1949). W. Dziubdziela (1977), H.Chernoff and H. Teicher (1965), L. Smith (1982,1983), E. J. Gumbel (1962), B. Kopocinski (1973), R. E. Barlow and F. Proschan (1975) and W. Feller (1978). The paper presents the results of the author's effort to extend and to put an end to the current state of the investigations on the limit distributions of maximin and minimax statistics in the sequence of independent random variables not necessarily identically distributed. The terminology proper to reliability theory is used.

2. ESSENTIAL NOTIONS

Denote by E_{ij} , where $i = 1,2,\ldots,k$, $j = 1,2,\ldots,l_i$, system components and by X_{ij} their lifetimes being , by the assumption , independent random variables . A system is called series–parallel if its lifetime is given by

$$(1) \qquad X = \max_{1\le i\le k}\{\min_{1\le j\le l_i}\{X_{ij}\}\} ,$$

and it is called parallel–series if its lifetime is given by

$$(2) \qquad X = \min_{1\le i\le k}\{\max_{1\le j\le l_i}\{X_{ij}\}\} .$$

Moreover , these systems are called regular if

$$l_1 = l_2 = \ldots = l_k = l \text{, where } l \in N$$

and they are called homogeneous if random variables

$$X_{ij} \text{ , } i = 1,2,\ldots,k, \ j = 1,2,\ldots,l,$$

have the same distribution function

$$F(x) = P(X_{ij} \le x) ,$$

i.e. components E_{ij} have the same reliability function

$$R(x) = P(X_{ij} > x) = 1 - F(x) \ , \ x \in (-\infty, \infty).$$

Assuming $k = k_n$ and $l = l_n$,where n tends to infinity and k_n and l_n are some sequences of natural numbers such that at least one of them tends to infinity, we get some sequences of the regular homogeneous systems corresponding to the sequence (k_n, l_n) . Replacing n by a positive real number t and assuming that k_t and l_t are positive real numbers, we get some families systems corresponding to the pair (k_t, l_t). For these families of systems there exist families of reliability functions. The family of reliability functions of systems there exist the families of reliability functions given by

$$\mathbf{R}_t(x) \ = \ 1 \ - \ [1 \ - \ (R(x))^{l_t}]^{k_t} \ , \ x \ \in \ (-\infty, \infty) \ , \ t \ \in \ (0, \infty)$$

for a series–parallel system and

$$\overline{\mathbf{R}}_t(x) \ = \ [1 \ - \ (F(x))^{l_t}]^{k_t} \ , \ x \ \in \ (-\infty, \infty) \ , \ t \ \in \ (0, \infty)$$

for a parallel–series system.

3. RESULTS

We assume the following definition :

A reliability function $\mathbf{R}(x)$ is called an asymptotic reliability function of a regular homogeneous series–parallel system if there exist functions $a_t > 0$ and $b_t \in (-\infty, \infty)$ such that

$$\lim_{t \to \infty} \mathbf{R}_t(a_t x + b_t) \ = \ \mathbf{R}(x) \text{ for } x \ \in \ C_{\mathbf{R}} \ ,$$

where $C_{\mathbf{R}}$ is a set of continuity points of $\mathbf{R}(x)$. A pair (a_t, b_t) is called a norming functions pair.

The up to now known results relative to the possible limit distribution functions of the sequence of statistics (1) i (2), fixed for

$$k_t \ = \ l_t \ = \ n, \ n \ \in \ N,$$

can be formulated as follows :

The only possible nondegenerate asymptotic reliability functions of the regular homogeneous system with equal numbers of series and parallel components are one of the following types :

$$\mathbf{R}_1 \ = \ \begin{cases} 1 & \text{for } \ x \leq 0 \\ 1 - \exp[-x^{-\alpha}] & \text{for } x > 0 \ , \ \text{ where } \ \alpha > 0 \ , \end{cases}$$

$$\mathbf{R}_2 \ = \ \begin{cases} 1 - \exp[-(-x)^{-\alpha}] & \text{for } x < 0 \ , \text{ where } \alpha > 0 \\ 0 & \text{for } \ x \geq 0 \ , \end{cases}$$

$$\mathbf{R}_3 \ = \ 1 \ - \ \exp[-\exp(-x)] \text{ for } x \in (-\infty, \infty).$$

for a series-parallel system and

$$\overline{\mathbf{R}}_i(x) \ = \ 1 \ - \ \mathbf{R}_i(-x), \ x \ \in \ C_{\mathbf{R}_i} \ , \ i \ = \ 1, 2, 3,$$

for a parallel-series system.

In a natural way the problem of the existence of asymptotic reliability functions for series–parallel and parallel–series systems with different numbers of series and parallel components and with unalike components appears. This problem is partly solved in [1-7] and the results may be formulated as follows :

The only possible nondegenerate asymptotic reliability functions of the regular homogeneous series–parallel are one of the following types :

1° $\mathbf{R}_i(x)\ ,\ i = 1,2,3,$

if (under some condition on the way of changing of l_t)

$$k_t\ =\ t,\ |\, l_t\ -\ c\ln t\,|\ >>\ s,\ s\ >\ 0\ ,\ c\ >\ 0.$$

2°

$$\mathbf{R}_4\ =\ \begin{cases} 1 & \text{for } x < 0 \\ 1 - \exp[-\exp[-x^{\alpha} - \frac{s}{c}]] & \text{for } \mathrm{x} \geq 0, \text{ where } \alpha > 0\ , \end{cases}$$

$$\mathbf{R}_5\ =\ \begin{cases} 1 - \exp[-\exp[(-x)^{\alpha} - \frac{s}{c}]] & \text{for } x < 0\ , \text{ where } \alpha > 0 \\ 0 & \text{for } \mathrm{x} \geq 0\ , \end{cases}$$

$$\mathbf{R}_6\ =\ \begin{cases} 1 - \exp[-\exp[\beta(-x)^{\alpha} - \frac{s}{c}]] & \text{for } x < 0\ , \text{ where } \beta > 0 \\ 1 - \exp[-\exp[-x^{\alpha} - \frac{s}{c}]] & \text{for } \mathrm{x} \geq 0, \text{ where } \alpha > 0\ , \end{cases}$$

$$\mathbf{R}_7\ =\ \begin{cases} 1 & \text{for } x < x_1 \\ 1 - \exp[-\exp[-\frac{s}{c}]] & \text{for } x_1 \leq x < x_2 \\ 0 & \text{for } \mathrm{x} \geq x_2, \text{ where } \mathrm{x}_1 < x_2, \end{cases}$$

if

$$k_t = t\ ,\ l_t - c\ln t \sim s\ ,\ s \in (-\infty,\infty),\ \ c > 0.$$

3°

$$\mathbf{R}_8\ =\ \begin{cases} 1 - [1 - \exp[-(-x)^{-\alpha}]]^k & \text{for } x < 0\ , \text{ where } \alpha > 0 \\ 0 & \text{for } \mathrm{x} \geq 0\ , \end{cases}$$

$$\mathbf{R}_9\ =\ \begin{cases} 1 & \text{for } x < 0 \\ 1 - [1 - \exp[-x^{\alpha}]]^k & \text{for } x \geq 0\ , \text{ where } \alpha > 0\ , \end{cases}$$

$$\mathbf{R}_{10}\ =\ 1\ -[1-\ \exp[-\exp x]^k \text{ for } x\ \in\ (-\infty,\infty).$$

if

$$\lim_{t\to\infty} k_t = k\ ,\ \lim_{t\to\infty}\ l_t = \infty.$$

4° Under the same assumptions on a pair (k_t, l_t) as in the cases 1°, 2° and 3° , the only possible nondegenerate asymptotic reliability functions of the regular homogeneous parallel-series system are one of the following types :

$$\bar{\mathbf{R}}_i(x) = 1\ -\ \mathbf{R}_i(-x),\ x\ \in\ C_{\mathbf{R}_i}\ ,i\ =\ 1,2,\ldots,10.$$

The above results may be transformed to the nonhomogeneous systems, i.e. the systems with unalike components. These results are given in [8], [9] and [10] and summarised in [11]. The fixed classes for nonhomogeneous systems are also ten-element but these classes may be regarded as more extensive in the sense that ten types of asymptotic reliability functions, which forms depend on some properties of the reliability functions of particular components and on the frequences of their appearance in the system, have been fixed.

4. REFERENCES

[1] Kolowrocki, K., Asymptotic reliability functions for series–parallel systems. Advances in Modelling and Simulation, AMSE Periodicals,Vol. 25, N^o 3, France (1991), 49-63.

[2] Kolowrocki, K., Remarks on a class of limit reliability functions of some regular hmogeneous series–parallel systems. Advances in Modelling and Analysis, AMSE Periodicals, Vol. 34, N^o 4, France (1992), 57-63.

[3] Kolowrocki, K., On a class of limit reliability functions of some regular homogeneous series–parallel and parallel- -series systems. Advances in Modelling and Analysis, AMSE Periodicals, C, Vol. 37. N^o 2, France (1993), 55-60.

[4] Kolowrocki, K., On a class of limit reliability functions of some regular homogeneous series–parallel systems. Reliability Engineering and System Safety 39, N^o 1, Elsevier Science Publishers LTD, England (1993), 11-23.

[5] Kolowrocki, K., On a class of limit reliability functions of some regular homogeneous series–parallel systems. Applied Mathematics 36, Polish Mathematical Society Publications, Poland, (in press).

[6] Kolowrocki, K., Limit reliability functions of some series–parallel and parallel–series systems. Applied Mathematics and Computation, Holland, (submitted for publication).

[7] Kolowrocki, K., On asymptotic reliability functions of series–parallel and parallel–series systems with identical components. Reliability Engineering and System Safety 41, Elsevier Science Publishers LTD, England, (in press).

[8] Kolowrocki, K., On limit reliability functions of some series–parallel and parallel–series systems with unalike components, IEEE Transactions on Reliability, USA, (resubmitted for publication).

[9] Kolowrocki, K., Limit reliability functions of some nonhomogeneous series–parallel and parallel–series systems. Applied Stochastic Models and Data Analysis, John Wiley and Sons, (submitted for publication).

[10] Kolowrocki, K., Limit reliability functions of some nonhomogeneous series–parallel and parallel–series systems. Reliability Engineering and System Safety, Elsevier Science Publishers LTD, England, (resubmitted for publication).

[11] Kolowrocki, K., On a class of limit reliability functions for series–parallel and parallel–series systems. Maritime University Press, monograph, Gdynia, (1993), pp. 125.

STABILITY OF PERIODICAL AND CHAOTIC VIBRATIONS IN SYSTEMS WITH MORE THAN ONE EQIULIBRIUM POSITIONS

Olga A. Chernoivan, Yuri V. Mikhlin

Dnepropetrovsk State University

Kazakova 4, 34, Dnepropetrovsk 320050 Ukraine

Abstract. Models that may be obtained by the discretisation of elastic systems that have lost stability are considered. The stability of one-dimensional periodical and chaotic vibrations in a spase with a greater number of dimensions has been examined (over a sufficiently wide finite range of time) with the use of computer.

Some recent papers discussed regular and chaotic behavior of the model governed by nonautonomous Duffing's equation with two potential pits [1,3]

$$x'' + dx' - px + ax^3 = f\cos(wt) \qquad (a,p,d,f>0) \qquad (1)$$

Here (') refers to the differentiation with time. Computer analysis have shown that as the amplitude of external factor increases, the vibration period successively doubles; beginning with certain values of f, a chaotic behavior is observed in the system, the region of the chaotic vibrations possessing all properties of a "strange attractor".

This model (and more general models) may be obtained by the discretisation (Bubnov - Galerkin procedure) of an elastic system that has lost stability under a constant compressive force. Consider, for example, nonlinear bend vibrations of a rod within the context of Kirchhoff hypotesis, the rod being subjected to an axial compression and a distributed external force of the form Fsin(pi*x/l)cos(wt). Supercritical behavior of the rod is investigated. The displacement of a nonlinear system and the distributed effects are approximated by two harmonics Fourier series expansion for space coordinates. One obtains a set of two second order o.d.es, coupled only in the nonlinear terms (a system with two degrees of freedom):

$$\begin{aligned} &x'' + dx' - px + ax^3 + cxy^2 = f\cos(wt) \\ &y'' + dy' - by + ay^3 + cx^2y = 0 \qquad (a,p,d,b,c,f>0) \end{aligned} \qquad (2)$$

The same system of equations may be obtained in the problem about nonlinear forced oscillations of long cylindrical shells within the context of Karman-Tsien equations linked diametrical displacement and stress function [4] in assumption that shell compressed along generatrix axial compression forces applied to the arc of the circle. Diametrical displasement are approximated by reduced Fourier series for space coordinates. After discretisation we receive again the system (2). So equation (2) have sense for the various elastic systems after branching of the initial eqilibrium state.

It is possible energy "pumping" from one space mode of vibrations to another in the model (2). Therefore, it is possible to formulate a problem of stability of periodical and chaotic vibrations of system with one degree of freedom ($x=x(t)$, $y=0$) within the limit of model with a greater number of dimension. As perturbation variables y, y' are chosen here.

If for initial values of this variables (which are sufficiently smaller then initial values of x, x' variables) inequality $|y(t)| < |R(y(t0))|$ is true for the rather large interval of variable t value change, we say about stability of investigated solution on the finite time interval, otherwise - about instability. In such a way correctness of the model (1) is estimated within the limit of more general elastic system.

We suggest the following view of the test function:

1) $R(z) = 10z$,

2) $R(z) = 10z\ (1 + |\ln(1 + |xmax/x(0)|\ )|\ )$,
$xmax = max|x(t)|$, $0 <= t <= T$.

In the case 1) instability is fixed after exceeding of variable y it's initial value "on the order": in the case 2) taking into account that in the regime of chaotic oscillations unlike of the periodic solution maximal value variable $x(t)$ may exceed initial one in many times, that, in its turn, influence on the $y(t)$ behavior. The calculation are realized on the interval of time $0 <= t <= T$ in such a way that furthering increasing T wouldn't change boundaries of stability and instability in the space of parameters.

Numerical results were obtained for the following value of parameters: $a=100$; $p=10$; $d=0.1,5$; $c=400$; $w=3.76$. The parameter b was varied from -440 to $+8800$, correspond to te real geometrical characteristics of rods and shells. The amplitude of the internal force f varyes from 0 to 2.5, and it is known [1,3] that chaotic oscillations take place for the values of f nearing 1 and more. Chosen of initial conditions almoust doesn't influence on the stability of solution in correspondence with above proposed criteries. Some results are displayed below. Symbol y marks stability, n - instability. On the horizontal axis value of parameter b with variable step (step 10 near zero and step was increased far from the zero), on the vertical - value of parameter f: 0, 0.2, 0.75, 0.9, 1.2, 1.8, 2.5.

```
   b=-440          0                      -8800
f=0    nnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyyyyy  Shell.
 0.2   nnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy  T=20.
 0.75  nnnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyy
 0.9   nnnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyy  d=1,y(0)=0.01,
 1.2   nnnnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyy      y'(0)=0
 1.8   nnnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyy  R(z)=10z
 2.5   nnnnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyy

f=0    nnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy Shell.
 0.2   nnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy T=20.
 0.75  nnnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyy
 0.9   nnnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyy d=1,y(0)=0.003,
 1.2   nnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyyy     y'(0)=0.001
 1.8   nnnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyy R(z)=10z
 2.5   nnnnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyy

f=0    nnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyyyyy Shell.
 0.2   nnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy T=20.
 0.75  nnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyyy d=1, y(0)=0.01
 0.9   nnnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyy       y'(0)=0
 1.2   nnnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyy R(z)=10z(1 + |ln(1 +
 1.8   nnnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyy         +|xmax/x(0)|)|),
 2.5   nnnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyy xmax=max|x(t)|, 0<=t<=T

f=0    nnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy Shell.
 0.2   nnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy T=20.
 0.75  nnnnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyy d=1, y(0)=0.003,
 0.9   nnnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyy       y'(0)=0.001
 1.2   nnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyyy R(z)=10z(1 + |ln(1 +
 1.8   nnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyyyy         +|xmax/x(0)|)|),
 2.5   nnnnnnnnnnnnyyyyyyyyyyyyyyyyyyyyyyyyy xmax=max|x(t)|, 0<=t<=T
```

Results. The stability (over a finite range of time) of one-dimensional models with periodical and chaotic behavior, in a space with a greater number of dimension has been examined with the use of computer.

References.

[1] Holmes, P.J., A Nonlinear Oscillator with a Strange Attractor.Philos.Trans.R.Soc.London, A 292 (1979),419-448.

[2] Kauderer, H., Nichtlinear Mechanik. Springer-Verlag, Berlin,1958

[3] Moon, F.C., Chaotic vibrations. A Wiley-Int.Publication, New York,1987.

[4] Karman, Th. and Tsien H.S., The bukling og thin cylindrical shells under axial compression. J.Aeron.Sci.8, N8 (1941), 303-312.

Mathematical Model of Allocation Problem of Geometric Objects of Arbitrary Spatial Form

Yu. G. Stoyan
Institute for Problems in Machinery
of Ukrainian Academy of Sciences
2/10 Pozharsky Str., 310046, Kharkov,Ukraine

Abstract. A mathematical model of optimization problem of allocation of geometric objects of arbitrary spatial form in a given domain is constructed. At this it is assumed that allocation objects are the sources of physical-mechanical fields and they are connected between themselves by routes of definite profiles according to the given scheme of connections.

1 Introduction

During synthesis of technical systems consisting of discrete sources of physical-mechanical fields (loads, thermal sources, electromagnetic sources, etc.) with given metrical characteristics (dimensions) connected between themselves with routes of given profiles (roads, pipelines, cables, etc.) a number of optimization problems arises.

To solve these problems successfully both in systems of automated design and in different intelligent technological systems it is necessary to have its such mathematical model that would allow to take advantage of both modern mathematical methods and computational facilities.

We choose a class of point sets (called φ-objects [2]) in the arithmetic Euclidean space R^k $(k = 1, 2)$ which may be taken as a model of allocation objects (supports of physical-mechanical fields).

Definition 1. Nonempty set $T \subset R^k$ $(k = 2, 3)$ is called a φ-object if T - canonically closed $(T = \mathrm{Cl}^* T = \mathrm{Cl}\,\mathrm{int}\, T)$ or canonically open $(T = \mathrm{int}^* T = \mathrm{int}\,\mathrm{Cl}\, T)$ and in any point $x \in \mathrm{Cl}\, T$ (closure T) there is its neighbourhood such that is its interior and its closure have the same homotopic type [1].

As it is known, the position of φ-object T_i in R^3 is fully determined by three coordinate of origin of its eigen coordinate system $v_i = (x_i, y_i, z_i)$ and its three angular parameters $\gamma_i = (\varphi_i, \psi_i, \omega_i)$. $\nu_i = (v_i, \gamma_i)$ are called parameters of allocation of φ-objects $T_i (i = 1, 2, ..., m)$. Any φ-object T_1 is determined by the equation of its $\mathrm{Fr}\, T_1$(frontier T_1) with respect to eigen coordinate systems OX' as follows $f_1(X') = 0$ where $f_1(X') \geq 0$ if $X' \in T_1$.

Making use of the equation $f_1 = 0$, we shall write the equation of the φ-object T_1 with respect to fixed coordinate system OX in R^k as follows $F_1(X, v_1, \gamma_1) = f_1[L(X - v_1)] = 0$ where L is an orthogonal operator depending on the angular parameters $\gamma_1 = (\varphi_1, \psi_1, \omega_1)$.

On the basis of equations of type $F_1 = 0$ Φ-functions $\Phi_{ij}(\nu_i, \nu_j, \gamma_i, \gamma_j)$ are constructed for any pair of φ-objects T_i and T_j which possess characteristic properties

$$\begin{gathered} \Phi_{ij}(v_i, v_j, \theta_i, \theta_j) > 0, \text{ if } \mathrm{Cl}\, T_i \cap \mathrm{Cl}\, T_j = \emptyset, \\ \Phi_{ij}(v_i, v_j, \theta_i, \theta_j) = 0, \text{ if } \mathrm{int}\, T_i \cap \mathrm{int}\, T_j = \emptyset \text{ and } \mathrm{Fr}\, T_i \cap \mathrm{Fr}\, T_j \neq \emptyset, \\ \Phi_{ij}(v_i, v_j, \theta_i, \theta_j) < 0, \text{ if } \mathrm{int}\, T_i \cap \mathrm{int}\, T_j \neq \emptyset. \end{gathered} \tag{1}$$

The physical field $u(x, y, z, t) = u(X, t)$ in a synthesis system is described in the general case by the non-stationary boundary value problem of the type

$$Au = P, B_j u = f_j, j = 1, 2, ..., n^*, \tag{2}$$

where $X \in T_0$, A is the given operator; B_j are the given operators, defining boundary conditions, initial conditions and conjugation at boundary surface of a medium; f_j are the given functions;

$$P = \begin{cases} P_i(X,t) \text{ if } X \in T_i \\ 0 \text{ if } X \notin \left(\bigcup\limits_{j=1}^{m} T_j \right) \setminus T_i. \end{cases} \tag{3}$$

As it follows from (2) and (3), the physical field of a synthesis system is the function of kind

$$u = u(x,t,\nu), \tag{4}$$

where $\nu = (\nu_1, \nu_2, ..., \nu_m)$.

We suppose that each φ-object T_i has points $a_{ih} \in \operatorname{Fr} T_i (i = 0,1,...,m; h = 1,2,...,h_i)$ which determine the location of the "entrances-exits" of routes $K_{ij}^{\tau h} (i < j = 1,2,...,m-1; r = 1,2,...,k_i; h = 1,2,...,k_j)$ connecting these objects according to the given matrix

$$C = \left| \begin{array}{cccc} c_{00} & c_{01} & \dots & c_{0n} \\ c_{10} & c_{11} & \dots & c_{1n} \\ \multicolumn{4}{c}{\dotfill} \\ c_{n0} & c_{n1} & \dots & c_{nn} \end{array} \right|, c_{ij} = \left| \begin{array}{cccc} e_{11}^{ij} & e_{12}^{ij} & \dots & e_{1k_j}^{ij} \\ e_{21}^{ij} & e_{22}^{ij} & \dots & e_{2k_j}^{ij} \\ \multicolumn{4}{c}{\dotfill} \\ e_{n1}^{ij} & e_{n2}^{ij} & \dots & e_{nk_j}^{ij} \end{array} \right| \tag{5}$$

where $c_{ij} = 0$ if $i = j$.

We assume that the profile $S_{ij}^{\tau h}$ is determined by the following equation $\varpi_{ij}^{\tau h}(x, y, m_{ij}^{\tau h}) = 0$ where $m_{ij}^{\tau h} = (m_1^{\tau h}, m_2^{\tau h}, ..., m_{k_{ij}}^{\tau h})$.

Definition 2. Axis $\sigma_{ij}^{\tau h}$ of route $K_{ij}^{\tau h}$ is the path $\sigma_{ij}^{\tau h}(t) : [t_{ij}^{\tau}, t_{ij}^{h}] \to T_0$ such that $\sigma_{ij}^{\tau h}(t_{ij}^{\tau}) = a_{ij}^{\tau} \in \operatorname{Fr} T_i, \sigma_{ij}^{\tau h}(t_{ij}^{h}) = a_{ij}^{h} \in \operatorname{Fr} T_j$ and satisfies the requirements:

$$1.\ \sigma_{ij}^{\tau h}(t) = (x_{ij}^{\tau h}(t), y_{ij}^{\tau h}(t), z_{ij}^{\tau h}(t)), \tag{6}$$

2. $\sigma_{ij}^{\tau h}(t) \in C^2_{[t_{ij}^{r}, t_{ij}^{h}]}$. 3. Intersection of any plane normal to axis $\sigma_{ij}^{\tau h}(t)$ of route $K_{ij}^{\tau h}$ forms the profile $S_{ij}^{\tau h}$ of this route.

2 The Basic Optimization Problem

It is necessary to allocate φ-objects $T_i (i = 1,2,...,m)$ generating physical fields (2), (3) in the domain T_0 and form routes $K_{ij}^{\tau h}$ of the given profile $S_{ij}^{\tau h} (i < j = 1,2,...,m-1;\ r = 1,2,...,k_i;$ $h = 1,2,...,k_j)$ according to the matrix (5) and with account of the given additional requirements on the allocation of the φ-objects $T_i (i = 1,2,...,m)$ and routes $K_{ij}^{\tau h} (i < j = 1,2,...,m-1;$ $r = 1,2,...,k_i;\ h = 1,2,...,k_j)$ their connecting is so that some functional κ would reach its best value.

Remark. Additional restrictions on the allocation of the φ-objects $T_i (i = 1,2,...,m)$ in the domain T_0 may be requirements to the physical field $u(X,t,\nu)$ (4). These restrictions may be represented in the general case by the following system of inequalities

$$D_k u(X,t,\nu) \left.\right|_{\Omega_k} \le (\ge) u_k, k = 1,2,...,p, \tag{7}$$

where D_k is the given operator; u_k is the limit value of the physical field in the domain $\Omega_k \subset T_0$.

As it is known, in the work [3] the inequality determining the route $K_{ij}^{\tau h}$ of given profile $S_{ij}^{\tau h}$ has the kind

$$\Omega_{ij}^{\tau h}(X, \sigma_{ij}^{\tau h}, (\sigma_{ij}^{\tau h})', (\sigma_{ij}^{\tau h})'', \theta_{ij}^{\tau h}, m_{ij}^{\tau h}) \ge 0, \tag{8}$$

where $\sigma_{ij}^{\tau h}$ is axis (6); $(\sigma_{ij}^{\tau h})'$ -i-th derivative of the function $\sigma_{ij}^{\tau h}$; $\theta_{ij}^{\tau h}$ -is the angle determining orientation of profile $S_{ij}^{\tau h}$; $\Omega_{ij}^{\tau h} = \varpi_{ij}^{\tau h}(E_{ij}^{\tau h} \cdot A_{ij}^{\tau h}(X - \sigma_{ij}^{\tau h}), m_{ij}^{\tau h})$; $E_{ij}, A_{ij}^{\tau h}$ are operators of rotation.

The conditions of transversality [3] of routes $K_{ij}^{\tau h}$ and φ-objects T_i in points a_{dl} are determined by the system of equations

$$\begin{cases} \sigma_{ij}^{\tau h}(t_{ij}^{l}) - B_i^l a_{d\tau} - v_i = 0 \\ \mu_{ij}^{\tau h}(t_{ij}^{l}) - \mu_d^l + \mu_d^{l_*} = 0 \\ \psi_{ij}^{\tau h}(t_{ij}^{l}) - \psi_d^l + \psi_d^{l_*} = 0 \\ \varphi_{ij}^{\tau h}(t_{ij}^{l}) - \varphi_d^{l_*} = 0, \end{cases} \tag{9}$$

$d = i, j; l = \tau, h; i < j = 1, 2, ..., m - 1; \tau = 1, 2, ..., k_i; h = 1, 2, ..., k_j$,
where B_i^l is the operator of rotation.

To provide the requirements of item 3 in Definition 2 it is necessary to fulfill the following inequality

$$\rho_{ij}^{\tau h}(t) - R_{ij}^{\tau h}(t) \geq 0, \tag{10}$$

where $\rho_{ij}^{\tau h}(t)$ is the radius of curvature of axis $\sigma_{ij}^{\tau h}$; $R_{ij}^{\tau h}(t)$ is the radius of curvature of the "external" Fr $S_{ij}^{\tau h}$.

To simplify the designations we shall redesignate all routes $K_{ij}^{\tau h}(0 < i < j = 1, 2, ..., m-1; \tau = 1, 2, ..., k_i; h = 1, 2, ..., k_j)$ as follows $K_i(i = 1, 2, ..., \gamma)$.

3 Mathematical Model

Making use of Φ-function (1), the mathematical model of the basic optimization problem may be represented as follows:

To determine

$$\operatorname{extr} \kappa(t^b, t^e, \nu, \sigma, m, \theta), \tag{11}$$

on the feasible region of solution W, which is given by inequalities (7), (8), equations (9) and also inequalities (10) and systems of inequalities which determine:

conditions of allocation of routes K_i in the domain T_0

$$F_{0i}(t_i^b, t_i^e, \sigma_i(t), m_i(t), \theta_i(t)) \geq 0, i = 1, 2, ..., \gamma,$$

where t_i^b is the "origin" of route K_i; t_i^e is the "end" of route K_i; $\sigma_i(t)$ is the axis of route K_i; $m_i(t)$ are the metric characteristics of the route profile S_i; $\theta_i(t)$ is the angle defining the orientation of the profile S_i;

conditions of mutual non-intersection of routes K_i and K_j

$$F_{ij}(t_i^b, t_j^b, t_i^e, t_j^e, \sigma_i, \sigma_j, m_i, m_j, \theta_i, \theta_j) \geq 0, 0 < i < j = 1, 2, ..., \gamma;$$

conditions of mutual non-intersection of routes K_i and φ-objects T_j

$$H_{ij}(t_i^b, t_j^b, \sigma_i, m_i, \theta_i, \nu_j) \geq 0, i = 1, 2, ..., \gamma; j = 1, 2, ..., m;$$

conditions of mutual non-intersection of routes φ-objects T_i and T_j

$$\Phi_{ij}(\nu_i, \nu_j) \geq 0, 0 < i < j = 1, 2, ..., m;$$

conditions of allocation of φ-objects T_i in the domain T_0

$$\Phi_{0i}(\nu_i) \geq 0, i = 1, 2, ..., m;$$

additional requirements to the location of φ-objects T_i in the domain T_0

$$F_i(\nu_i) \geq 0, i = 1, 2, ..., k;$$

additional requirements to the location of routes K_i in the domain T_0

$$P_i(t_i^b, t_i^e, \sigma_i, m_i, \theta_i) \geq 0, i = 1, 2, ..., l;$$

4 Conclusions

The stated problem (11) includes optimization both on a set of functions (variational problem) and on a set of vectors in the Euclidean arithmetical space.

If this problem has a strict hierarchical structure allowing to perform optimization on the set of vectors ν, and then on the set of functions $\sigma_i(t), \theta_i(t), m_i(t) (i = 1, 2, \ldots, \gamma)$, then the process of its solution is considerably simplified [4].

If the profile of all routes is represented as a circle and each route has one and the same profile over the whole length (circle of one and the same radius), then the variational problem may be reduced to determination only of route axes $\sigma_i(t) (i = 1, 2, \ldots, \gamma)$.

If the routes axes consist of several parts which are specified, though unknown are the coordinates of the points of connection of these parts, and, in addition, the orientation of the routes profiles and their metric characteristics are defined on finite sets, then the stated problem is reduced to optimization on a set of vectors in the corresponding Euclidean arithmetical space.

5 Results

Mathematical model of problem of allocation of φ-objects in the given domain is constructed with simultaneously forming of routes of given profile.

For some class of boundary value problems the continuous dependence of function $u(X, t, \nu)$ by parameters of allocation ν of φ-objects [5] has been proved.

A number of optimization problem of allocation of non-point sources of physical field in the given domain are solved [5], [6].

A number of problems of optimal allocation of geometric objects of arbitrary spatial form in the given domain are solved [7].

Programming system for layout of a box of a complex technical system of block structure having hierarchical structure is created [4].

References

[1] Borisovich, Yu.G., Bliznyakov, N.M., et al, Introduction into Topology. Moscow: Vysshaya Shkola, 1980, 295 pp. (Russ.)

[2] Stoyan, Yu.G., On One Generalization of the Dense Allocation Function. Rept. Ukr. SSR. AS Ser.A., 1980, N 8, 70-74. (Russ.)

[3] Stoyan, Yu.G., Mathematical Model of Optimization Problem of Geometric Design with Account of Routes.- Kharkov, 1993, 29 pp., (Preprint / AS of Ukraine, Inst. for Problems in Machinery, N 368). (Russ.)

[4] Stoyan, Yu.G., Ponomarenko, L.D., Programming System KTS Automatical Layout of Box of Complex Technical System of Block Structure, 1989, 22 pp., (Preprint / AS of Ukraine, Inst. for Problems in Machinery, N 310). (Russ.)

[5] Stoyan, Yu.G., Putyatin V.P., Optimization of Technical Systems with Sources of Physical Fields. Kiev: Naukova Dumka, 1988, pp. 190. (Russ.)

[6] Stoyan, Yu.G., Kolodyazhny, et al, Optimal Construction of Radioelectronic Device of the Cassette Design According to Temperature Control. In book: Heat Transfer in Electronic and Microelectronic Equipment (Proceedings of the International Centre for Heat and Mass Transfer), Hemisphere Publishing Corporation, N-Y, 1990, 809-823.

[7] Stoyan, Yu.G., Mathematical Methods for Geometric Design. In Advances in CAD/CAM: Proc. PROLOMAT 82 (Leningrad, USSR. 16-18, May 1982), Amsterdam etc. 1983, 67-86.

NEW TECHNOLOGIES OF REAL-TIME MODELLING

Anatoly A. NAUMOV
Nikolay P. Kislenko

Novosibirsk State Technical University
P.B. 241, Novosibirsk-104, 630104, Russia

Abstract. The considered problem is software for experimental strategy synthesis and real-time data processing. The researched method is the variant of FCS called as Immertion method. It consists in such transformation of initial problem that solution can be realized more efficiently .

During more than thirty years beginning from J.Kiefer's publications [1],[2] theorems of optimality and equality of experiment designs for regression model are used and developed in theory and practice in the next form:

$$f(x,a) = a^T\varphi(x),\ y_i = f(x_i,a) + e_i,\ i = \overline{1, N_0},$$

a -vector of unknown parameters, $\varphi(x)$ - basis vector of regression model, e_i -realizations of errors of observations. Then the theorem of optimality and equality for D-optimal experiment design

$$\xi^* = \begin{pmatrix} x_1, x_2, \dots, x_N \\ \xi_1, \xi_2, \dots, \xi_N \end{pmatrix},\ x_i \in \mathfrak{X},\ \xi_i \geq 0,\ \sum_{i=0}^{N} \xi_i = 1,$$

is written as

1) Designs ξ^* maximize $|M(\xi)|$(minimize $|D(\xi)|$);
2) Designs ξ^* minimize $\max\limits_{x \in \mathfrak{X}} \lambda(x)\, d(x,\xi)$;
3) $\max\limits_{x \in \mathfrak{X}} d(x,\xi^*) = m$.

Statements 1)-3) are equivalent. Here $|M(\xi)|$ is determinant of design ξ information matrix ,$|D(\xi)|$ is determinant of dispersion matrix , $\lambda(x)$ is function efficiently, $d(x,\xi) = \varphi^T(x) D(\xi) \varphi(x)$, m is number of parameters in the model.

However , in 1988 we are published the result refuting the statement of optimality and equality theorem and thus serving as the beginning of Kiefer's theorem recomprehesion and overproof. In particular , the next statement was proved:

In general case D-optimal experiment design ξ^* cannot be invariant relatively unknow model parameters α.

This fact except a practical usage of Kiefer's theorem in the form of it was written above and therefore computer software on this basis can't be used always (see, f.e. [3],[4]). There is showed that the usage of J. Kiefer's and V. Fedorov's results in the problem of complicated dynamic system active identification can result in essential deteriorotion of received solutions and models quality. In accordance of our statement represented above some new original reproaches for active identification problem are suggested. In particular , there is an ε-invariant immersion method.

REFERENCES

[1] Kiefer, J., Optimal experimental design , J. Roy. Statist. Soc.,Ser. B,21 ,1959, 272-319.

[2] Kiefer,J. , General equivalence theory for optimum designs (approximate Theory) , Ann. Statist.,2,1974, 849-879.

[3] Fedorov, V., Malyotov , M., Optimal designs in regression problems,Math. OF Statist.,3,1972,281-308.

[4] Denisov, V., Matematicheskoe obespechenie sistem EVM-eksperimentator,1977 (in Russian).

[5] Naumov, A., Metod invariantnych pogrugeniy v sadachach aktivnoy identifikazii, In: Identificaciay,izmerenie characteristic i imitaciay,Novosoborsk,1991 , 58-59.(in Russian).

[6] Naumov, A., Activnaya express-identificaciya dynamicheskich sistem v realnom vremeni, In: Proceedings of international symposium "Engineering Ecology-91", Zvenigorod,1991, 283-287.

A Discrete Model of a Dynamic System : a multistand mill - a strip

Leonid Kuznetsov
Lipetsk Polytechnic
30 Moskovskaya Street, 398055 Lipetsk, Russia

Abstract. We consider a system digitization of differential - difference equations in this paper. It is important to stress that various difference equations have different delays. The discrete description preserves delays which are the basis for the real time's decomposition. On the basis of the initial description we build a modular vector-matrix discrete model which is close to a conventional form.

The initial system description has the form [2]:

$$\dot{x}(t)=A_x x(t)+A_h \theta h(t)+A_w w(t), \qquad (1)$$

$$h(t)=B_x x(t)+B_h \theta h(t)+B_w w(t), \qquad (2)$$

$$z(t)=C_x x(t)+C_h \theta h(t)+C_w w(t), \qquad (3)$$

where $A_x, A_h, ..$ are factors matrixes; $x(t), h(t), ..$ are the state vectors, external effects and output values; θ is the shift operator which is determined in the following way:

$$\theta h(t)=\theta(h_1(t),h_2(t),...,h_\iota(t))^T= =(h_1(t-\tau_1),h_2(t-\tau_2),...,h_\iota(t-\tau_\iota))^T, \qquad (4)$$

τ_ι, $\iota=1,2,...,\iota$, are time delays.
(1) is a system of differential equations, (2) is a system of difference equations, (3) is a system of algebraic equations.

While digitizing the time moments are used

$$t=\alpha_1\tau_1+\alpha_2\tau_2+...+\alpha_\iota\tau_\iota, \quad \alpha_\iota\in Z_0=\{0,1,2,...\}, \; \iota=1,2,...,\iota. \qquad (5)$$

Here α_ι, $\iota=1,2,...,L$, may be considered the decomposition coordinates of t on the basis of τ_i, $i=1,2,\ldots,L$. Time moments having the same coordinates sum

$$\alpha_1+\alpha_2+\ldots+\alpha_l=\|\alpha\|=K \qquad (6)$$

are united into $L(K)$ group which k number plays the role of associated discrete time. For each $L(k)$ group the modular vectors $X(K),h(K),\ldots$ are built and they include all vectors $X(t),h(t),\ldots$ such as $t\in L(K)$ in the certain order [1].

The digitization of a differential system (1) is carried out by its solution in the interval τ_i, $i=1,2,\ldots,l$. The solutions of a differential linear system (1) are the matrix exponents with the arguments of the form (5). From here in accordance with the order of vectors following $X(t),h(t),\ldots$ we build a modular matrix in modular vectors $X(t),h(t),\ldots$ which determine the effect $X(k-1)\to X(k),h(k-1)\to X(k),W(k-1)\to X(k)$.As a result on the basis of the system (1) an equation is built with modular vectors and matrixes

$$X(k)=\Psi_x(k)X(k-1)+\Psi_h(k)h(k-1)+\Psi_w(k)W(k-1). \qquad (7)$$

A discrete analogue of the difference system (2) is constructed for modular vectors $X(k),h(k),\ldots$ by modular matrixes designing $\Phi_x(k),\Phi_h(k),\Phi_w(k)$ from B_x,B_h,B_w matrixes of the initial system (2). As a result we get:

$$h(k)=\Phi_x(k)X(k)+\Phi_h(k)h(k-1)+\Phi_w(k)W(k). \qquad (8)$$

A discrete analogue of the equation (3) is built similarly:

$$Z(k)=\Gamma_x(k)X(k)+\Gamma_h(k)h(k-1)+\Gamma_w(k)W(k). \qquad (9)$$

Unlike the initial presentation (1)-(3) a system (7)-(9) is not stationary due to changing of vectors dimensionality $X(k),h(k),\ldots$ and matrixes $\Psi(k),\Phi(k)$. The presentation (7)-(9) allows the usage of space states of a modern control theory of control analysis and synthesis.

REFERENCES

[1] Барышев, В.Г., Блюмин, С.Л., Кузнецов, Л.А., К управлению системами с многомерным параметром. Автоматика и телемеханика, 4 (1977), 37-42.

[2] Кузнецов, Л.А., Применение УВМ для оптимизации тонколистовой прокатки. Металлургия, Москва, 1988.

ANALYSIS OF SOME MATHEMATICAL MODELS IN NON-LINEAR DYNAMICS

A.A.Zevin

Transmag Research Institute, Ukraine Academy of Sciences.
320005 Dniepropetrovsk, Piesarzhevskogo 5, Ukraine

Abstract. The cases are indicated where the use of a quasi-conservative model for a non-linear dynamic system may result in a mistaken conclusion on the stability of periodic solutions. It is shown that the oscillations of swing cannot be explained within the framework of the universally adopted model.

1. INTRODUCTION

Analytical investigations of non-linear dynamic systems are usually conducted via perturbation methods. The corresponding unperturbed system admits an exact analytical solution; an approximate solution of the initial problem and the multipliers of the variational equation are expressed as a power series in a small parameter μ (see, for example, [1,5,6]).

When nonconservative and nonautonomous terms are small, a quasi-conservative model is often employed; here the unperturbed system is usually integrable or consists of disjoint oscillators (note that this approach is utilized in studies of the synchronization of weakly connected objects, in particular, planets [1]). The result of stability analysis is certainly true for $\mu\in(0,\mu_*)$; however, the critical value μ_* remains unknown. It proves out that for some systems, μ_* is quite small, so the quasi-conservative model leads to a wrong conclusion on the stability even for small μ. Such systems are indicated in two subsequent sections. In the last section the mathematical model of swing is discussed.

2. NONAUTONOMOUS SYSTEM

Oscillations of a quasi-conservative system with one degree of freedom are usually governed by the equation

$$\ddot{x}+f(x)=\mu\varphi(x,\dot{x},\omega t) \qquad (1)$$

where $\varphi(x,\dot{x},\omega t)=\varphi(x,\dot{x},\omega t+2\pi)$. The small parameter method enables to find a T-periodic solution $x(t,\mu)$ close to the solution $x_0(t)$ for $\mu=0$. However, a rigorous qualitative analysis shows that the value μ_* may be very small, provided that the position force $f(x)$ is not monotonous. To illustrate it, let us consider forced oscillations of a pendulum ($T=2\pi/\omega$, $f=\omega_0^2 sinx$, $\varphi=cos\omega t$). Let $x_0(t)=x_0(-t)$, $x_0(0)>0$, then there exist solutions $x_1(t,\mu)\longrightarrow x_0(t)$, $x_2(t,\mu)\longrightarrow -x_0(t)$ as $\mu\longrightarrow 0$. Using the procedure of small parameter method [5], one can find that $x_1(t,\mu)$ is unstable, $x_2(t,\mu)$ is stable for small μ. In fig.1 the amplitude-frequency curves for the solution $x_2(t,\mu)$ obtained via direct numerical calculations are shown (1.μ=0.1, 2.μ=1, 3.μ=2); solid and dotted lines correspond to stable and unstable solutions. As seen, $x_2(t,\mu)$ is unstable for $\omega\in(0,\omega_*(\mu))$. Therefore, for any small μ there is a frequency interval where the quasi-conservative model yields the wrong conclusion on the stability of the solution $x_2(t,\mu)$.

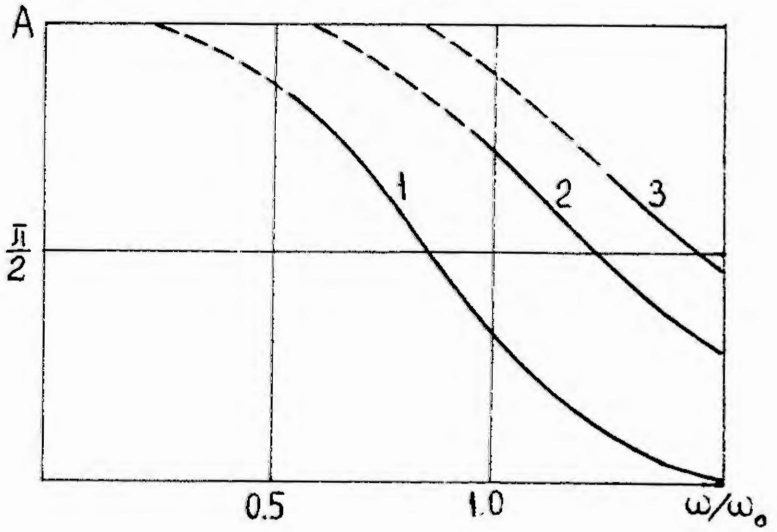


Fig.1

If the position force is repulsive ($f(x)x<0$ for $x\neq 0$), the unperturbed system has only the trivial periodic solution $x(t)\equiv 0$. An asymptotic method testifies to instability of the periodic solution $x(t,\mu)$. However, a qualitative analysis [3] shows that $x(t,\mu)$ may be stable, provided that $f(x)$ does not decrease monotonically. For example, forced oscillations of a pendulum about the upper equilibrium point ($f(x)=-\omega_0^2 sinx$) become stable beginning with some amplitude $A<\pi$.

3. AUTONOMOUS SYSTEM

In the case of weakly connected objects the unperturbed system consists of n disjoint oscillators. It proves out that such approach may result in error even for very weak ties when the position forces of oscillators are hardening ($f_i(x)/x$ increase with x). As an example, let us consider the system shown in fig.2.

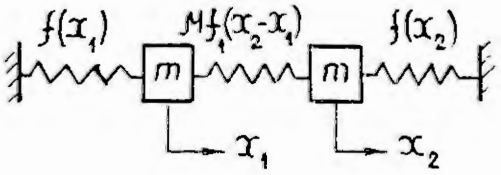


Fig.2

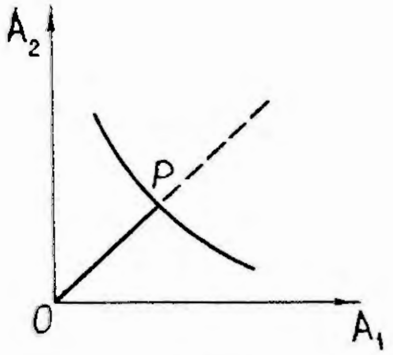


Fig.3

Due to the symmetry, for any μ, there exist in-phase and out-of-phase solutions $x^1(t,h)$ and $x^2(t,h)$ $(x_1^1(t,h)=x_2^1(t,h)$, $x_1^2(t,h)=-x_2^2(t,h))$ where h is the total energy of the system. For small h they are stable, however at some $h=h_*$ two periodic solutions branch off the solution $x^2(t,h_*)$ (fig.3 where A_1 and A_2 are the amplitudes)). A detailed analysis shows that both of them are stable while $x^2(t,h)$ becomes unstable for $h>h_*$. Since the branching point $P\longrightarrow 0$ as $\mu\longrightarrow 0$, then the asymptotic approach points to the instability of $x^2(t,h)$ for all h and, therefore, gives the wrong result for $h<h_*$.

4. MATHEMATICAL MODEL OF SWING

Swing are usually modelled with a pendulum having a periodically varying length, so they often serve as an example of a parametrically excited system [4]. Upon neglecting the nonconservative terms, the corresponding differential equation is wtitten as follows

$$\frac{d}{dt}\left(l\,\frac{dx}{dt}\right) + glx = 0 \qquad (2)$$

where l and x are the length and angle coordinate of the pendulum, g is the accelleration due to gravity; $l(t)=l(t+T)$.

Equation (2) admits $2T$-periodic solutions $x_1(t)$ and $x_2(t)$ for which the length reaches respectively its minimum and maximum at the point x=0. The stability analysis shows that $x_1(t)$ is stable, $x_2(t)$ is unstable [2]. Meanwhile, actually the oscillations of swing resemble the solution $x_2(t)$ (the length increases under squating and diminishes under raising). This contradiction is explained by the next consideration: in fact, the length is not an independent function of time but is defined by the swing motion, so that any perturbation (say, a small delay) of the motion causes a change in $l(t)$. Therefore, the universally adopted model of swing is not correct; they should be considered as a self-sustaining system $(l=l(x,\dot{x}))$.

5. REFERENCES

[1] Blehman,I.I.,Synchronization of Dynamic Systems. Nauka, Moscow, 1971.

[2] Zevin,A.A., Qualitative Analysis of the Stability of Periodic Oscillations and Rotations of Parametrically Excited Second Order Nonlinear Systems. Izv. AN SSSR., Mekh. Tverd. Tela, 2(1983),38-44.

[3] Zevin,A.A. and Filonenko L.A., Forced Oscillations of a Non-Linear System Having a Repulsive Position Force. AN SSSR, Prikl.Mat.Mekh., 54 (1990), 6, 944-950.

[4] Kauderer,H., Nichtlineare Mechanik. Springer-Verlag, Berlin, 1958.

[5] Malkin, I.G., Some Problems of the Theory of Non-Linear Oscillations. Gostehizdat, Moscow, 1956.

[6] Nayfeh, A.H., Perturbation Methods. Wiley, New York, 1973.

odically varying length, so they often serve as an example of a parametrically excited system [4]. Upon neglecting the nonconservative terms, the corresponding differential equation is wtitten as follows

$$\frac{d}{dt}\left(l\frac{dx}{dt}\right) + glx = 0 \qquad (2)$$

where l and x are the length and angle coordinate of the pendulum, g is the accelleration due to gravity; $l(t)=l(t+T)$.

Equation (2) admits $2T$-periodic solutions $x_1(t)$ and $x_2(t)$ for which the length reaches respectively its minimum and maximum at the point $x=0$. The stability analysis shows that $x_1(t)$ is stable, $x_2(t)$ is unstable [2]. Meanwhile, actually the oscillations of swing resemble the solution $x_2(t)$ (the length increases under squating and diminishes under raising). This contradiction is explained by the next consideration: in fact, the length is not an independent function of time but is defined by the swing motion, so that any perturbation (say, a small delay) of the motion causes a change in $l(t)$. Therefore, the universally adopted model of swing is not correct; they should be considered as a self-sustaining system ($l=l(x,\dot{x})$).

5. REFERENCES

[1] Blehman,I.I.,Synchronization of Dynamic Systems. Nauka, Moscow, 1971.

[2] Zevin,A.A., Qualitative Analysis of the Stability of Periodic Oscillations and Rotations of Parametrically Excited Second Order Nonlinear Systems. Izv. AN SSSR., Mekh. Tverd. Tela, 2(1983),38-44.

[3] Zevin,A.A. and Filonenko L.A., Forced Oscillations of a Non-Linear System Having a Repulsive Position Force. AN SSSR, Prikl.Mat.Mekh., 54 (1990), 6, 944-950.

[4] Kauderer,H., Nichtlineare Mechanik. Springer-Verlag, Berlin, 1958.

[5] Malkin, I.G., Some Problems of the Theory of Non-Linear Oscillations. Gostehizdat, Moscow, 1956.

[6] Nayfeh, A.H., Perturbation Methods. Wiley, New York, 1973.

An Improved Model of a Propeller Aircraft

W. Dunkel
Technical University of Braunschweig
Institute for Flight Guidance and Control
Rebenring 18, D-38106 Braunschweig

1. INTRODUCTION

To develop models for flight simulators and numerical flight simulations [4] as well as for the design of automatic flight control systems (e.g., autopilots) [2] a sufficient knowledge of the process is essential. Often a significant mismatch is encountered between flight test data and computer simulations based on theoretical models [1] of the aircraft motion. This mismatch may be caused by systematic measurement errors or by inadequate modeling of the aircraft and/or of the measurement system. The system identification approach is a powerful tool for the detection and correction of these errors [6, 7]. For the identification of nonlinear aircraft models, flight-test data from a twin engine research aircraft DORNIER DO128 are used.

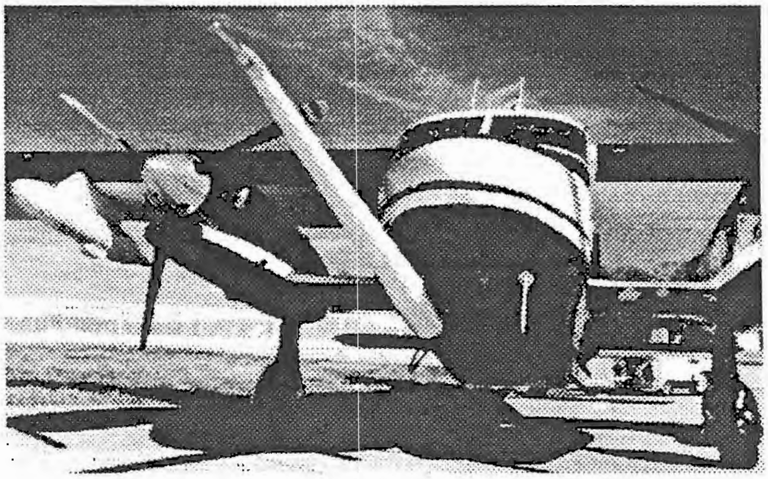


Fig. 1: The research aircraft DORNIER DO 128 - 6

2. IDENTIFICATION

The off-line identification is preferably done in two main steps according to the "Estimation Before Modeling" technique [7] which uses a Maximum-Likelihood identification. In a first step the well-known differential equations of the kinematic airplane model are used to estimate the unknown parameters of the measurement model and of the quasi static wind model. In the second step the propulsion system model and aerodynamic model are used to estimate the unknown aircraft-specific parameters. Advanced sensor systems allow to improve both parts of the model.

2.1. Expansion of the kinematic model

The measurement check as the first step of the identification makes use of a new satellite navigation system (Global Positioning System, GPS) with its output values latitude, longitude and altitude. In differential mode (DGPS) positions are determined with an uncertainty of less than one meter [5, 8]. If the model is extended accordingly this DGPS-data can be used to improve the accuracy of the measurement check (e.g., bias estimation of the accelerometers). To obtain the required accuracy of the identification model the equations in [7] have to be expanded to the description of the earth as a rotating ellipsoid.

Now the inertial system is the WGS 84 [9]. Again all measurements are assumed to be affected by bias, scaling factors and time shifts. These parameters are estimated by integrating the measured body-fixed angular velocities and accelerations and comparing the estimated and measured output quantities (Φ, Θ, Ψ, χ, V_{GS}, $\dot{h}$, h, φ, λ, V_A, α, β) as described below.

The measured angular velocity $\underline{\Omega}_b^{ib}$(between inertial- and body-fixed coordinate system: index superscript ib, measured in body-fixed coordinates: index subscript b) is measured by laser gyros. This velocity includes the angular velocity between the inertial system and the earth-fixed coordinate system $\underline{\Omega}_g^{ie}$ and the angular velocity between earth-fixed and geodetic coordinate system $\underline{\Omega}_g^{eg}$. The transformation matrix $\underline{M}_{bg}$ from geodetic to body-fixed coordinates (a function of the Euler angles) leads to $\underline{\Omega}_b^{gb}$ and to the equation

$$\partial \underline{\Phi}/\partial t = \underline{\Omega}_b^{gb} = \underline{\Omega}_b^{ib} - \underline{M}_{bg} \cdot (\underline{\Omega}_g^{ie} + \underline{\Omega}_g^{eg}) \qquad (1)$$

with $$\underline{\Omega}_g^{ie} = \left[\Omega_e \cdot \cos\varphi,\ 0,\ -\Omega_e \cdot \sin\varphi\right]_g^T \qquad (2)$$

$\Omega_e = 7292115 \cdot 10^{-11} \text{rad s}^{-1}$; [9] is the angular velocity of the earth

$$\underline{\Omega}_g^{eg} = \left[\dot{\lambda} \cdot \cos\varphi,\ -\dot{\varphi},\ -\dot{\lambda} \cdot \sin\varphi\right]_g^T \qquad (3)$$

for determining the vector of the Euler angles $\underline{\Phi}$ (roll angle Φ, pitch angle Θ , yaw angle Ψ)

$$\underline{\Phi} = \left[\Phi,\ \Theta,\ \Psi\right]_b^T \quad . \qquad (4)$$

The acceleration $\underline{a}_b^{ib}$ which can be measured in the aircraft's center of gravity includes the acceleration from earth mass attraction $\underline{G}_g$, centripetal and Coriolis acceleration. Therefore the transport acceleration $\partial\underline{V}_{Kg}/\partial t$ can be written as

$$\partial\underline{V}_{Kg}/\partial t = \underline{M}_{gb} \cdot \underline{a}_b^{ib} + \underline{G}_g - \underline{\Omega}_g^{ie} \times (\underline{\Omega}_g^{ie} \times \underline{r}_g) - 2\underline{\Omega}_g^{ie} \times \underline{V}_{Kg} - \underline{\Omega}_g^{eg} \times \underline{V}_{Kg} \qquad (5)$$

with $$\underline{G}_g = \left[0,\ 0,\ \mu \cdot (R_M + h)^{-2}\right]_g^T \qquad (6)$$

$\mu = 3986001.5 \cdot 10^8\, \text{m}^3\text{s}^{-2}$; [9] is the earth's gravitational constant

$$\underline{r}_g = \left[0,\ 0,\ (R_M + h)\right]_g^T \qquad (7)$$

which leads to the transport velocity in cartesian coordinates

$$\underline{V}_{Kg} = \left[u_K,\ v_K,\ w_K\right]_g^T \qquad (8)$$

and further to the ground speed V_{GS} and the true track χ

$$V_{GS} = (u_{Kg}^2 + v_{Kg}^2)^{1/2} \qquad (9)$$

$$\chi = \arcsin(v_{Kg} \cdot (u_{Kg}^2 + v_{Kg}^2)^{-1/2}) \quad . \qquad (10)$$

If the acceleration cannot be measured in the aircraft's center of gravity the distance from the center of gravity to the accelerometers $\underline{X}_b^m$ must be taken into account as follows

$$\underline{a}_b^{ib} = \underline{a}_b^m - \underline{\Omega}_b^{gb} \times \left[\dot{\underline{X}}_b^m + \underline{\Omega}_b^{gb} \times \underline{X}_b^m\right] - \dot{\underline{\Omega}}_b^{gb} \times \underline{X}_b^m \quad . \qquad (11)$$

Normally the variation of the distance e.g. by fuel consumption is negligible $\dot{\underline{X}}_b^m \approx 0$.

The position of the aircraft in earth-fixed coordinates (latitude φ, longitude λ, altitude h) can be obtained by

$$\partial\varphi/\partial t = u_{Kg} \cdot (R_M + h)^{-1} \qquad (12)$$

$$\partial\lambda/\partial t = v_{Kg} \cdot ((R_P + h) \cdot \cos\varphi)^{-1} \qquad (13)$$

$$\partial h/\partial t = -w_{Kg} \quad . \qquad (14)$$

The up to now unknown quantities in the equations above are the local earth's meridian radius of curvature R_M and the local earth's transverse radius of curvature R_P

$$R_M = a \cdot (1 - e^2) \cdot (1 - e^2 \sin^2\varphi)^{-3/2} \qquad (15)$$

$$R_P = a \cdot (1 - e^2 \sin^2\varphi)^{-1/2} \qquad (16)$$

with the earth's semimajor axis a (= 6378145 m) and the first eccentricity of the earth e (= 0.0818191908426) [9].

To demonstrate the effect of the expanded kinematic model the old model (with the earth described as a flat non-rotating system) and the model described here (the earth is a rotating ellipsoid) are compared. A simulation result of the determined earth-fixed positions of both models can be found in Fig. 2. The simulation uses a DO128 aircraft model without sensor errors and without wind. As an input signal an aileron doublet with an amplitude of 5 degrees and a duration of 2 seconds is used. This results in a total position error of 9.25 m after 100 seconds and of already 618.51 m after 200 seconds.

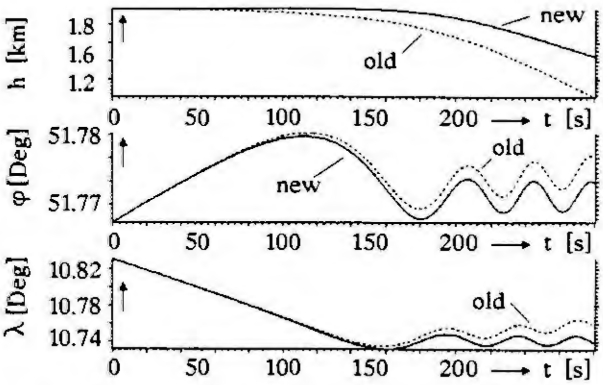


Fig. 2: Comparison of new and old model

2.2. Parameter identification of the aerodynamic and propeller model

So far the major problem of the second step of the identification is the lack of experimental data to validate the models of the coupled aerodynamic and propeller processes in a normal aircraft. It is merely possible to measure some input signals as there are the control deflections of the elevator η, the rudder ζ, the aileron ξ and the shaft power P of the engine as well as the aircraft velocity $\underline{V}_A$ (airspeed V_A , angle of attack α, angle of sideslip β). The forces $\underline{R}$ and moments $\underline{Q}$ of aerodynamic and propulsion system are not available through direct measurement. The closest measurable output quantities are the accelerations $\underline{a}$ and the angular velocities $\underline{\Omega}_K$ (see 2.1.), which are state variables of the kinematic model. There are no directly measurable aerodynamic and propeller output values to strictly separate the effects of both systems. To validate this part of the aircraft model it is advantageous to measure additional quantities in the research aircraft like the pressure behind the propeller (output value of the propulsion system) and the wing pressure (variable of the aerodynamic; a function of C_L). This paper will concentrate on the additional measurement of the pressure/airspeed behind the propeller by a Pitot tube.

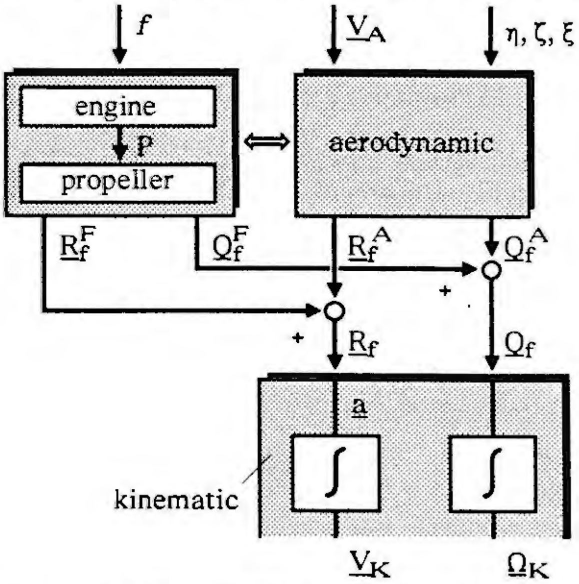


Fig. 3: Block diagram of the aircraft model

The connection to the equations in chapter 2.1. is given by the equation

$$\partial \underline{V}_{Kg} / \partial t \; = \; \underline{M}_{gb} \cdot m^{-1} \cdot (\underline{R}_b^A + \underline{R}_b^F) \qquad (17)$$

with the aerodynamic force (drag coefficient C_D, side force coefficient C_Q, lift coefficient C_L /see [3] / and the wing area S)

$$\underline{R}_b^A \; = \; \underline{M}_{ba} \cdot (0.5 \cdot \rho \cdot V_{Ab}^2) \cdot S \cdot \left[-C_D, C_Q, -C_L \right]_a^T \qquad (18)$$

and the propeller force (σ is the angle between the x-axis of the airplane and of the engine)

$$\underline{R}_b^F \; = \; (P \cdot V_{Ab}^{-1}) \cdot \eta_{prop} \cdot \left[\cos \sigma, \, 0, \, -\sin \sigma \right]^T \; . \qquad (19)$$

In this equation the term $(P \cdot V_{Ab}^{-1}) \cdot \eta_{prop}$ represents the thrust F. The propeller efficiency η_{prop} is given by the propeller charts of the manufacturer as a function of airspeed, air density and speed of the propeller. The deterioration of the propeller efficiency caused by the installation of the propeller (e.g. by drag in the propeller flow) is neglected.

To put things into perspective the already mentioned Pitot tube behind the propeller is integrated into the flight test equipment. This sensor in the propeller area measures the air pressure and allows to determine the airspeed at this point. By an additional measurement of the radial pressure distribution in the propeller area (see Fig. 4) in a ground test the accompanying distribution for axial stream is known. For non-axial stream the distance of the sensor to the propeller (safety requirements) and the speed distribution near to the sensor has to be taken into account.

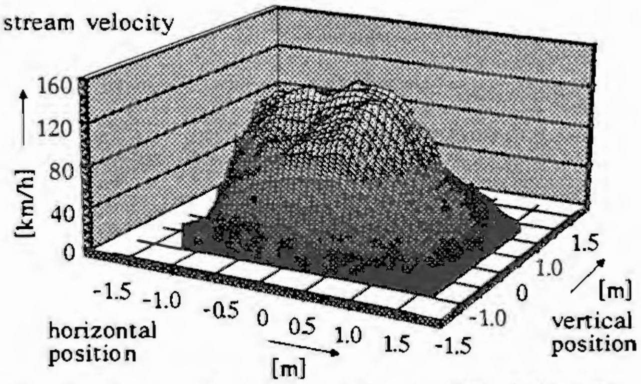


Fig. 4: Measured stream velocity (F = 2490.9 N)

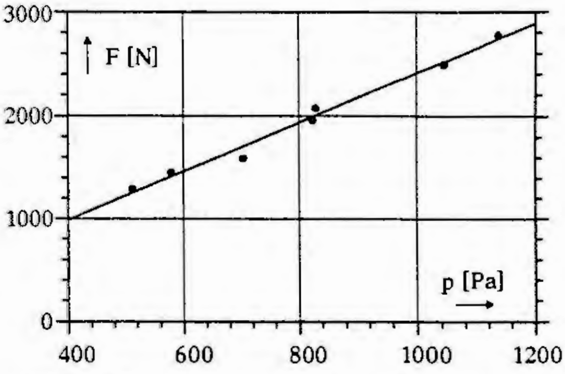


Fig. 5: Thrust as a function of P. pressure

This knowledge is used to extend equ. (19) with equations describing the drag of the parts of the aircraft behind the propeller (e.g. the nacelle of engine and tank, see Fig. 1) according equ. (18) where V_A is replaced by the measured speed behind the propeller. The mean air velocity behind the propeller (calculated from the measurement as mentioned above) can also be used to refine the calculation [6] of the propeller flow effects on the wing.

There is a second application for the Pitot tube. The integration of the pressure over the whole propeller area (measured in the ground tests) leads to the propeller thrust valid for the airplane at rest and axial stream. Figure 5 shows this calculated thrust as a function of the pressure measured by the Pitot tube.

3. SUMMARY AND CONCLUSION

The additional measurement signals available from DGPS and Pitot tube mounted behind the propeller allow to improve the aircraft model. Additional measurement equations containing further unknown parameters are introduced in the model. The introduced uncertainties can be reduced by additional information obtained from ground tests of the propeller and from wind tunnel test data of the wing profile.

4. REFERENCES

[1] Brockhaus, R., A mathematical multi-point model for aircraft motion in moving air. *Z. Flugwissenschaft und Weltraumforschung*, 11, 174 - 184, 1987.

[2] Brockhaus, R., Autopilots: Feedback Structure. *Systems & Control Encyclopedia, Pergamon Press*, Oxford, 1990.

[3] Dunkel, W., Brockhaus, R., A Nonlinear Observer for Sensor Fault Detection in an Airplane. *IFAC/IMACS Symposium*, Baden-Baden, 1991.

[4] Dunkel, W., Erprobung nichtlinearer Schätzfilter zur Sensorfehlererkennung, *DGLR Jahrbuch*, 1992.

[5] Jacob, Th., Beitrag zur Präzisionsortung von dynamisch bewegten Flugzeugen. *Dissertation, TU Braunschweig*, 1992.

[6] Proskawetz, K.O., Ein Beitrag zur Genauigkeitssteigerung bei der Parameteridentifizierung nichtlinearer Prozesse am Beispiel der Flugbewegung. *Dissertation, TU Braunschweig*, 1989.

[7] Proskawetz, K.-O., System-identification of airplanes using the "Estimation Before Modeling" Technique. *Z. Flugwissenschaft und Weltraumforschung*, 15, 401-407, 1991.

[8] Yuan, J., Schänzer G., Gu. X., Jacob, Th., Error correction for Differential GPS with long separated ground station and user for aircraft landing. *Symp. on Precise Positioning with GPS*, Ottawa, 1990.

[9] Department of Defense World Geodetic System 1984, DMA Technical Report 8350.2, 1987.

Nomenclature according ISO 1151, DIN 9300.

Hamiltonian Systems in Biomathematics

Josef Hofbauer

In this paper I discuss some Hamiltonian (and more general conservative) systems arising from ODE models in Mathematical Biology. I will present few results and many more open problems.

Lotka–Volterra Systems. The most well-known case is the classical predator–prey model:

$$\dot{x} = x(a - by) \qquad \dot{y} = y(-c + dx). \tag{1}$$

The phase portrait (restricted to the biologically interesting nonnegative quadrant $x, y \geq 0$) consists of a one parameter family of closed orbits. Hence the system is Hamiltonian, although not in the usual sense: One either has to make a (noncanonical) coordinate transformation $u = \log x, v = \log y$, or consider a suitable symplectic form, different from the standard one, on $\mathbb{R}^2_+$. The Hamiltonian is the well-known function of Volterra,

$$H = -dx + c\log x - by + a\log y. \tag{2}$$

If we allow the coefficients a, b, c, d in (1) to be *time-periodic* (with the same period, say 1), which in biological terms would model seasonal variations, the dynamics of (1) gets already complicated. The time 1 map is an area preserving map, often a twist map. Due to parametric resonance, interesting bifurcation phenomena may arise locally. The main problem, however - which I think is open - is the stability of (1) in the large: Is each solution of (1) bounded, as $t \to \infty$? Or is this true at least for small periodic variations? Equivalently, does this area-preserving map have arbitrarily large invariant curves (which bound the possibly chaotic orbits starting inside such a curve)? This calls for KAM theory. However, as far as I see, the standard versions produce only invariant curves of finite size.

The general Lotka Volterra equation in n dimensions reads

$$\dot{x}_i = x_i\big(r_i + \sum_{j=1}^{n} a_{ij}x_j\big). \tag{3}$$

The best framework for Hamiltonian systems in higher dimensions are *Poisson structures*. See [Pe] for a discussion of this concept, that generalizes the more familiar symplectic structures, and that allows to consider Hamiltonian systems also in odd dimensions. It turns out that, in this sense, (3) is Hamiltonian e. g. in the case of a skew–symmetric interaction matrix A, i. e. $a_{ij} = -a_{ji}$. The Hamiltonian function is again Volterra's well-known constant of motion, similar to (2), see [SZ]. More generally, it was recently shown by Plank [P], that (3) is Hamiltonian whenever there exists a constant of motion of the form $H = \prod x_i^{\alpha_i} (\beta_0 + \sum \beta_i x_i)$ with $\beta_i \neq 0$. In dimensions 2 and 3, this covers essentially all Hamiltonian systems of type (3), as shown in [P]. In higher dimensions, this is open.

A concrete example would be a two prey–two predator system, as studied in [K]. The Hamiltonian function is again Volterra's, but now indefinite. Hence local stability is not clear, even though the equilibrium is elliptic, but could be deduced from KAM theory. Again, stability in the large is an open question.

Another interesting problem is which of these Hamiltonian systems are completely integrable. This appears to be rather difficult. The only nontrivial example known seems to be the so-called *Moser–Caligero systems*, [M, (1.1)], where $r_i = 0$ and A is a tridiagonal matrix (with skewsymmetric signs).

Game Dynamics. We consider two separate populations of players, with n possible strategies in the first, and m in the second population. If A, B are the two payoff matrices, then such a game can be modelled by an evolutionary dynamics, which describes the change in time of frequencies x_i and y_j of strategies in the two populations:

$$\begin{aligned} \dot{x}_i &= x_i \left((Ay)_i - x \cdot Ay \right) \quad i = 1, \dots, n \\ \dot{y}_j &= y_j \left((Bx)_j - y \cdot Bx \right) \quad j = 1, \dots, m. \end{aligned} \tag{4}$$

For zero–sum games $A = -B^T$, this differential equation on the product $S_n \times S_m$ of two probability simplices, can be shown to be Hamiltonian for a suitable Poisson structure.

Actually, also for general A and B, (4) has a conservative dynamics: If one linearizes at interior equilibrium points, then the eigenvalues are symmetric with respect to the imaginary axis (like the eigenvalues of a linear Hamiltonian system), and one can find an invariant volume (see [HS, chs. 17, 27]). This raises the question, whether (4) is Hamiltonian in general. However, it was recently shown in [H], that this is not the case.

References

[H] J. Hofbauer, *Evolutionary dynamics for games between two populations: Is game dynamics a Hamiltonian system?*, Preprint. Vienna (1993).

[HS] J. Hofbauer, K. Sigmund, *The Theory of Evolution and Dynamical Systems*, Cambridge University Press, 1988.

[K] G. Kirlinger, *Permanence in Lotka–Volterra equations: Linked prey–predator systems*, Math. Biosciences **82** (1986), 165–191.

[M] J. Moser, *Three integrable Hamiltonian systems connected with isospectral deformations*, Advances in Math. **16** (1975), 197–220.

[Pe] A. M. Perelomov, *Integrable Systems of Classical Mechanics and Lie Algebras, Vol. I*, Birkhäuser, 1990.

[P] M. Plank, *Hamiltonsche Systeme und Lotka–Volterra Gleichungen*, Diplomarbeit, Universität Wien, 1993.

[SZ] F. Scudo, J. Ziegler, *The Golden Age of Theoretical Ecology, 1923–1940*, vol. 22, Springer Lecture Notes in Biomathematics, 1978.

Institut für Mathematik, Universität Wien, Strudlhofgasse 4, A-1090 Vienna, Austria

Automatic Generation of Software for Large Scale Control Systems

P. Bader, W. Bücker, H. Heller, K. Nökel, R.Schmid
SIEMENS AG, ZFE BT SE 11, Otto-Hahn-Ring 6, 81730 München
e-mail: bader@ztivax.zfe.siemens.de

Producing reliable software for the control of large and complex discrete event systems, especially where a very high degree of safety is required, is a very time consuming and expensive task. Today's solutions lack a number of desired properties like clearness and an appropriate description level of the control task, leading to a clumsy software development process which requires a high degree of testing and tends to result in a product with poor modifiability. We propose a new method of control software design which shows its qualities especially in those cases, where the description of the system can appropriately be separated into a description of the types (abstract data types) and a description of the instances.

Our approach has the following advantages:

- The control task is modeled in a *formal*, but natural and *domain specific language*, combining object orientation with a typed temporal predicate logic.
- From the specification (the task description) an executable control software module is *automatically generated* by a well defined and verifiable transformation process.
- The system's overall behavior is modeled by the behavior of its classes rather than the instances thereof. Thus *producing variants* of the control software is often reduced to the task of exchanging the set of instances and their structural relationship (topology).
- The generated software is efficient, therefore real-time constraints can be met. The representation of the software is *machine independent*. It can be adapted to different hardware platforms. Instead of a software module also a hardware solution can be derived.
- It enables the *automatic, formal proof of important properties* of the software (especially those which are regarded as highly safety critical), thereby contributing the reliability of the software and to considerable savings of testing.
- With automatic generation of the control module and the reduced need of testing, modifications of the specification get much cheaper.

An outline of our approach is as follows: the control task is regarded in principle as an automaton, with the events of the system as input symbols and the resulting control actions as output symbols. The states of the automaton are expressed by predicate logic formulae based on a number of predicates, which are given through the attributes of the system components.

The transition function is specified in the same predicate logic language, except that a temporal operator ("next_state") has been added. For each input symbol one specifies a number of so called *partial transitions*, each of which describes certain aspects of the whole transition function. More precisely each partial transition describes a binary transition relation, i.e., the set of pairs (*actual state, next state*), it allows.

The partial relations are constraints. By solving these constraints we derive the state transition function and the output function of the automaton. In case of contradictions in the specification of the automaton there is no solution. This fact can be translated into a meaningful error message in terms of the specification language.

The formal representation of the automaton allows to model check the automaton. Important properties of the system (liveness and safeness properties) can be proved automatically. If a property is shown not to hold, the checker comes up with an input symbol sequence which leads to the violation of the property.
Properties and the output of the model checker are formulated in the same control specification language as is used for the specification of the automaton.

Our approach has been used successfully to describe and generate automata with a state space of 2^{5000} states. Most recently our solution has been extended in order to deal with infinite domains.

ECG SIMULATION BASED ON A NEW MATHEMATICAL MODEL OF DYNAMICS WITH PIECEWISE FUNCTIONS

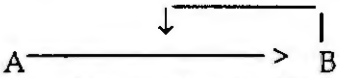
V. Gontar, M. Gutman, I. Erukhimovich and I. Ovsyshcher*

The International Group for Scientific and
Technological Chaos Studies (IGCS), The Institutes for
Applied Research, *Faculty of Health Sciences, Ben-Gurion
University of the Negev, Beer-Sheva,
P.O. Box 1025, 84101 Beer-Sheva, Israel

One of the key indices of bioelectrical activity of different tissues in the heart is the cardiac transmembrane potential (CTP) [1]. From CTP recordings it has been established that myocardial tissue has exceptional electrophysiological characteristics, a special feature being its slow rate of repolarization coupled with rapid depolarization. A shematic representation of electrogram and CTP curves is given in Fig. 1. This curve may be divided into three segments: 1. rapid upstroke or depolarization, 2. slow segment or repolarization, and 3. isoelectric line or resting membrane potential.

The biochemical processes taking place in a myocardial cell can be formally represented by the following gross reaction:

(1)



denoting transformation of a substrate A into the new state B, which gives rise to the dynamics of CTP variation; B exercises an autocatalytic influence on the process of its own formation. Stating the problem in this way enables us to apply our new model (NM) to calculate the dynamics of CTP variation [2].

Using the NM we obtain the following one-dimensional map describing the dynamics of CTPs:

(2)
$$x_n = \cfrac{b}{1 + \cfrac{1}{w_0^k + w^k x_{n-1}}} \qquad \begin{array}{ll} k=1 & 0 < x_{n-1} \leq p_1 \\ k=2 & p_1 < x_{n-1} \leq p_2 \\ k=3 & x_{n-1} > p_2 \end{array}$$

where b equals the sum of the concentrations of the components A_n, B_n of reaction (1) at the instant t_n, p_1, p_2 are control parameters responsible for the transitions between limbs of transform function (2), and empirical parameters $w_0{}^k$, w^k can be found by statistical analysis of the experimental data.

Fig. 2 presents the phase diagram of map (2). Each limb corresponds to a particular segment of the CTP: the first limb characterizes the process of depolarization, the second—repolarization, and the third describes the isoelectric line. Together with the bisector y = x, the middle portion of the second limb and the third limb form narrow "corridors" corresponding to the long "quasicontinuous" regions of the trajectory characteristic of the second repolarization phase and of the isoelectric region.

By altering the relative position of the different sectors of the map with the help of control parameters, we can obtain a wide range of CTP (Fig. 3A).

Fig. 3B presents the derivatives of modelled CTP trajectories, the dynamics of which reproduce the characteristic features of actual ECGs. The model makes it possible to vary the form and amplitude of the ECG over a wide range. Fig. 4 illustrates the variability of R-R intervals—heart rate variability (HRV). Reduction of HRV is typical of the most highly variability can easily be linked to changes in the shape of the CTP and ECG; this point is of practical interest in real-time diagnosis of cardiac abnormalities.

Thus, the model has the potential to encompass a broad spectrum of cardiac abnormalities.

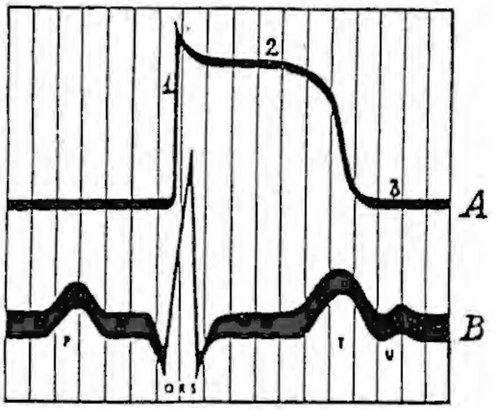


Fig.1

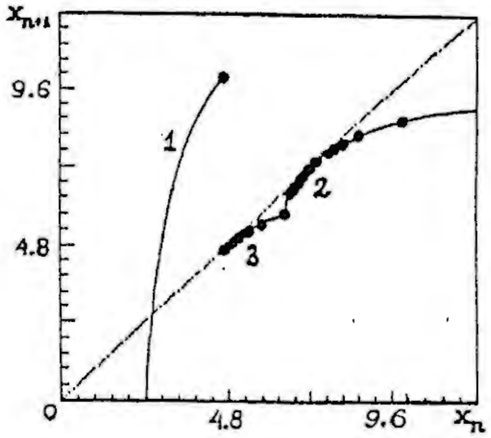


Fig.2

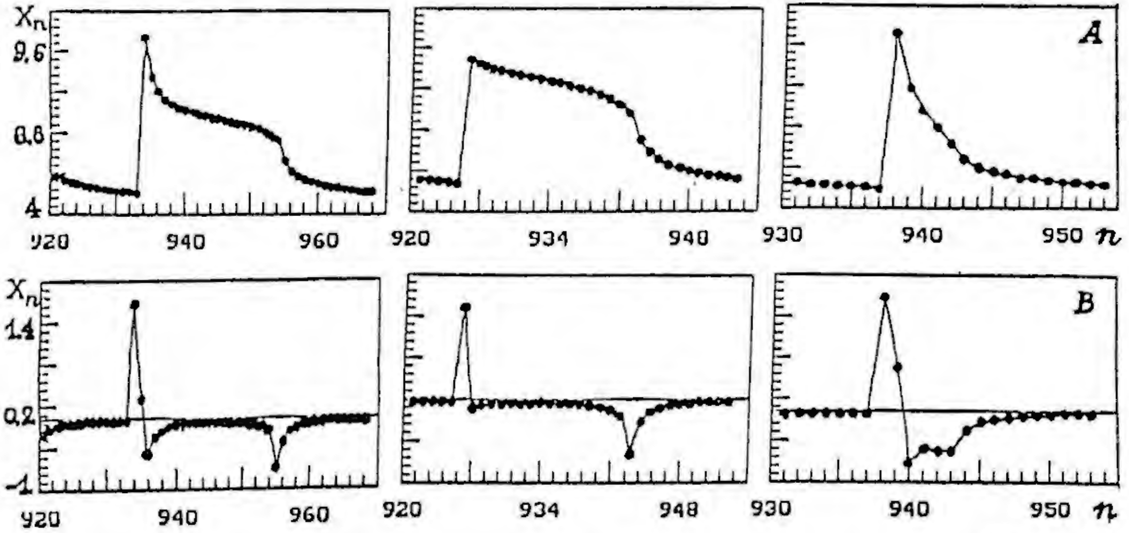


Fig.3

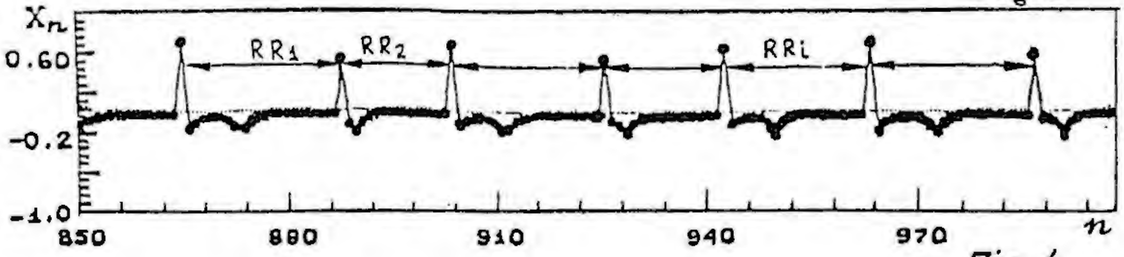


Fig.4

References:

1. Katz, A.M. Physiology of the heart (1992) Reuven Press, New York.
2. Gontar, V. Chaos in Chemistry and Biochemistry (1993) World Scientific, London, 215-232.
3. Goldberger, A.L. and West, B.J. Application of nonlinear dynamics to clinical cardiology (1987) Ann. NY Acad. Sc., 504, 195-273.

DEVELOPMENT OF SKILL OF MATHEMATICAL MODELLING USING FAMOUS PITFALLS IN SCIENCE AND ENGINEERING

Krzysztof Arczewski, Józef Pietrucha
Warsaw University of Technology
Institute of Aeronautics and Applied Mechanics
ul. Nowowiejska 24, 00-665 Warsaw, Poland

Abstract. The paper proposes a new item of mathematical modelling curriculum. An essential feature of this item is the extensive use of pitfalls encountered in scientific literature.

1. INTRODUCTION

The skill of the building of an adequate mathematical model of a real phenomenon is very important and difficult problem that a teacher of modelling must develop in his students. Since mathematical modelling is a multi-stage activity requiring a variety of concepts and methods, there are many "opportunities" to produce wrong models, which lead to absurd solutions. It seems that too small stress is laid on the role of pitfalls in the teaching of mathematical modelling. That is why in our book [1] we have written: "In reality there are many doubts and even errors before the modeler reaches a satisfactory model. It is good to learn of several setbacks suffered by great exponent of mathematics and mechanics, as a warning and... comfort."

This paper proposes a new item to teaching of mathematical modelling. An essential feature of this item is the extensive use of pitfalls made by prominent scientists. We suppose that the knowledge about such "specific" pitfalls can be used as a valuable encouragement for the beginning modelers.

All the pitfalls appearing in a vast scientific literature can be classified into four groups suited the four stages of a mathematical modelling process. It is why we start with a short description of modelling process.

2. AN OUTLINE OF MATHEMATICAL MODELLING PROCESS

There are two main ways of formulating mathematical models. The first, theoretical in principle, is based on direct application of physical laws, while in the second, the fundamental role is played by the experiment. In this paper we shall concern only about the first way.

The complex process of investigation of phenomena may be schematically illustrated by means of notions from the set theory. Let us introduce the following four sets:

- **R**, whose elements **r** are real objects or phenomena,
- **P**, whose elements **p** are physical models,
- **M**, whose elements **m** are mathematical models, and
- S, which elements **s** are solutions of the mathematical models.

Now we are in position to discuss Fig.1. The four circles marked by **R**, **P**, **M**, S represent respective sets introduced above, while arrowhead continuous lines represent four main activities of the investigation process. The picture shows that there is no unique relationships between the elements of different sets. It is easy to accept, knowing that the same physical model may represent different phenomena and vice versa, i.e. a given phenomenon, when modelled according to different modelling purposes may result in different physical models.

Similar situations appear for successive pairs of sets and this fact is reflected in the Fig.1. Additionally, let us note that not every physical model **p** has its counterpart in the set **M** of mathematical models, and not each mathematical model **m** has its solution within the set S. Except for continuous lines there are also broken lines. These show an influence of respective modelling stages on the previous ones. It often happens that certain observations, results, or, even, difficulties encountered at the subsequent stages imply necessary modifications of previous results. Thus, all these activities which are symbolized by broken lines will be called modifications.

It should be stressed that in practice most modelling does not take the precise form shown in the Fig.1. The figure is there just to give some idea of the underlying relationship between real world problems and the mathematical techniques used to find solutions to them.

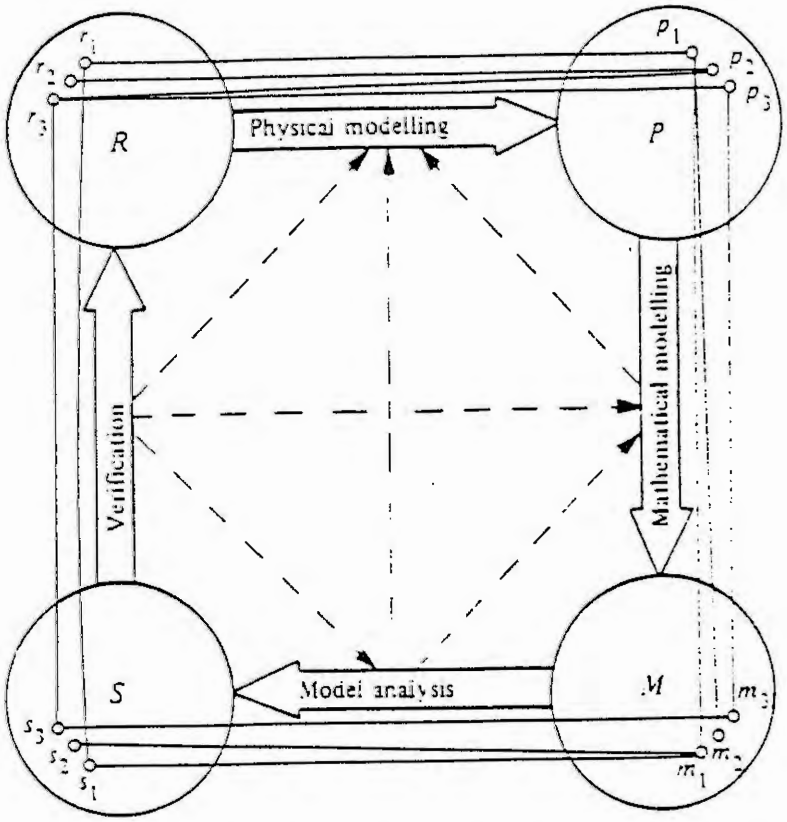


Fig.1. Scheme of the process of investigation of the phenomena

3. CLASSIFICATION OF PITFALLS

All the pitfalls within the mathematical modelling process can be classified into four following groups:

1. missing an important factor/component of a physical model, e.g., [2],
2. falsification of a theory applied, e.g., [3],
3. errors on a stage of model analysis, e.g., [4],
4. errors at a stage of model verification, e.g., [5].

The examples of each kind of pitfalls will be shown during the conference in a poster session.

4. ACKNOWLEDGEMENTS

This work was supported by grant No. 3 0955 91 01. The author wishes to acknowledge the State Committee for Scientific Research which offered this grant.

5. REFERENCES

[1] Arczewski, K., Pietrucha, J., Mathematical Modelling of Complex Mechanical Systems, vol.1. Chichester, E.Horwood, 1993, p.286.

[2] Euler, L., 1757. In: Panovko, J.G., Gubanova, I.I., Stability and Vibrations of Elastic Systems, (in Russian), Nauka, Moscow, 1987, p.12.

[3] Lindelöf, E., 1895. Sur le mouvement d'un corps de revolution roulant sur un plan horizontal. Acta Soc. Scie. Fenn., v.XX, No.10.

[4] Ashley, H., McIntosh S.C., 1968. Applications of Aeroelastic Constrains in Structural Optimization. In: Proc.12th Inter. Cong. of Appl. Mech. Springer-Verlag, Berlin, 1969, p.100.

[5] Oseen, C.W., 1912. Über Wirbelbewegung in einer reibenden Flüssigkeit. Ark. för Mat., Astr. Och. Fis., v.7, p.14.

Collision-Avoidance Control for Redundant Articulated Robots

N. Rahmanian-Shahri
Technical University Vienna
Wiedner Hauptstrasse 8-10
A-1040 Wien
Austria

Abstract. This work presents a suitable mathematical formulation of a robot and obstacles such that for on-line collision recognition only robot joint positions in the workspace are required. In addition to this, three collision avoidance methods are presented which allow the use of redundant degrees of freedom such that a manipulator can avoid obstacles in work space while tracking the desired end-effector trajectory. These collision avoidance methods are based on optimization of a suitable function, on influence of joint angle rates and on control of the self-motion of the manipulator, respectively. The effectiveness of the porposed methods is discussed first theoritically and then illustrated by simulation results.

INTRODUCTION

A robot manipulator is defined to be kinematically redundant if it possesses more degrees of freedom than are required to achive the desired position and orientation of the end-effector. One of the adventages of robot redundancy is the potential to use the extra degrees of freedom to maneuver in a congested workspace and avoid collisions with obstacles, where obstacles are defined as objects in the robot workspace [1].

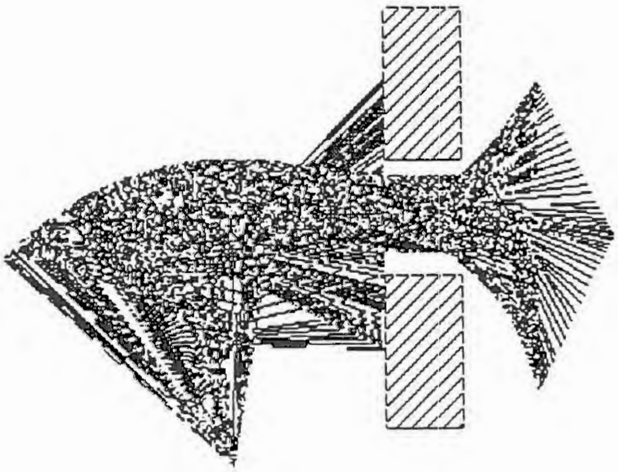
The majority of the work reported to date concerning obstacle avoidance for robot manipulators has dealt with high-level path planning, in wich the end-effector path is planned off-line so as to avoid collision with workspace obstacles. Alternatively, the obstacle avoidance problem of redundant robots can be solved on-line by the robot controller at the low level such that a redundant robot closely tracks the desired end-effector trajectory and simultaneously avoids workspace obstacles [2].

This work presents suitable mathematical formulation of robot and obstacles such that for on-line collision recognition only robot joint positions in the workspace are required. This can essentially reduce the calculation time because the joint positions in the workspace can be computed from the jointvariable through robot geometry at any time. It is supposed that the obstacles in the workspace of the manipulator are represented by convex polygons. For every link of a redundant robot and every obstacle in the workspace a boundary ellipse is defined such that there is no collision if the robot joints are out of this ellipsis [3].

In addition to this, three collision avoidance methods are presented that allow the use of redundant degrees of freedom such that a manipulator can avoid obstacles in the work space while tracking the desired end-effector trajectory.

The optimization method is based on the generalized inverse with boundary ellipse functions as optimization ceriteria. The method permits the tip of the

hand to approach any arbitrary point in the free space while the kinematic control algorithm maximizes the boundary ellipse function of the critical link.
A new and simple approach to collision-avoidance control through influence of joint angle rates over the entire motion is developed. This method is based on augmentation of the manipulator forward kinematics and the Moore-Penrose inverse of the corresponding augmented Jacobian matrix. As the variable vector of the augmented system has elements with different physical units, a Moore-Penrose inverse of the augmented Jacobian results which influences joint angle rates.
A further approach to avoid collision of the links of redundant robots with obstacles consists in control of the self-motion of the manipulator. This method is based on coordinate transformation and inverse kinematic and leads to the favorable use of the ability of redundant robots to collision-avoidance.
The effectivness of the porposed methods is discussed by theoritical consideration and illustrated by simulation results in [3].
The figure below gives a shot of simulation of the motion of a manipulator between obstacles, a further demonstration is provided in a video film.



[1] **Allgeuer H.** *Kinematische Steuerung von Robotern mit redundanten Freiheitsgraden.* Dissertation, Technical University Vienna, TNF, 1992.

[2] **Colbaugh R., Seraji H., Glass K.L.** Obstacle Avoidance for Redundant Robots Using Configuration Control. *Journal of Robotic Systems*, Vol. 6, Nr. 6, S. 721-744, 1989.

[3] **Rahmanian-Shahri N.** *Steuerungsalgorithmen zur Vermeidung von Kollisionen der Glieder redundanter Roboter mit Hindernissen.* Dissertation, Technical University Vienna, TNF, 1993.

The Knowledge Based System for Process Modelling DIOPRAN-EXPERT II

Possekel, F.; Weiß, B.; Winkler, W.

Dipl.-Ing. Beate Weiß
Technische Universität Ilmenau
Fakultät für Informatik und Automatisierung
Institut für Automatisierungs- und Systemtechnik
PSF 327
98684 Ilmenau

Abstract. The knowledge based system DIOPRAN-EXPERT II is an interactive software system for the application of experimental methods in process identification. It supports the user during the selection and application of appropriate methods. The present paper explains the basic structure of the system, the rules being used, the extraction of the needed facts, and the storage of the expert knowledge in decision modules.

1. Problemspecification

The design of signal and system models based on measured data requires both extensive knowledge in the field of experimental process analysis and comprehensive knowledge of the system itself. The support of the control engineer requires therefore besides the provision of a multitude of well-known methods in the field of process analysis as a software package interactive suggestions for the selection and application of appropriate methods depending on the problem and the available information [1]. With the system DIOPRAN-EXPERT II developed by us the information necessary for the solution of the problem is mostly automatically derived from the available data. DIOPRAN-EXPERT II is the consistent further development in conception, content, and software technology of the expert-system DIOPRAN-EXPERT [2] in which the algorithms for experimental process analysis established with DIOPRAN 88 were joined with suggestions for their application (as PROLOG-rules) for the first time. The areas of application for DIOPRAN-EXPERT II include:

- the design of signal models;
- the design of dynamic and static, linear as well as non-linear input/output models;
- the provision of suitable test signal patterns and experiment strategies;
- the data pre-processing.

2. Structure and operation of the knowledge based system DIOPRAN- EXPERT II

Depending on the user's level of experience it is distinguished between an interactive mode of operation and an automatic execution according to standard settings. According to figure 1 first of all the required information (the a-priori facts) is queried starting from the problem to be solved and the available measured data. With the available facts the rules for the method selection are activated and a suitable method is selected. Its results are stored in the process database and lead to the generation of further facts. The illustrated processing steps are repeated until a solution of the problem is reached. Typical cycles are the data pre-processing, the determination of system properties (linear/non- linear, disturbed/undisturbed), the structure search, as well as model estimation and validation [3].

2.1 Fact extraction and storage

The facts contain statements about the properties of the signals involved with the process and the system behaviour. They are derived by calculation and evaluation of characteristic values and functions, utilisation of derived signals as well as determination of the order of precedence and structure search or they are entered by the user as a-priori facts. They have the purpose to store the derived knowledge about the system or the signals

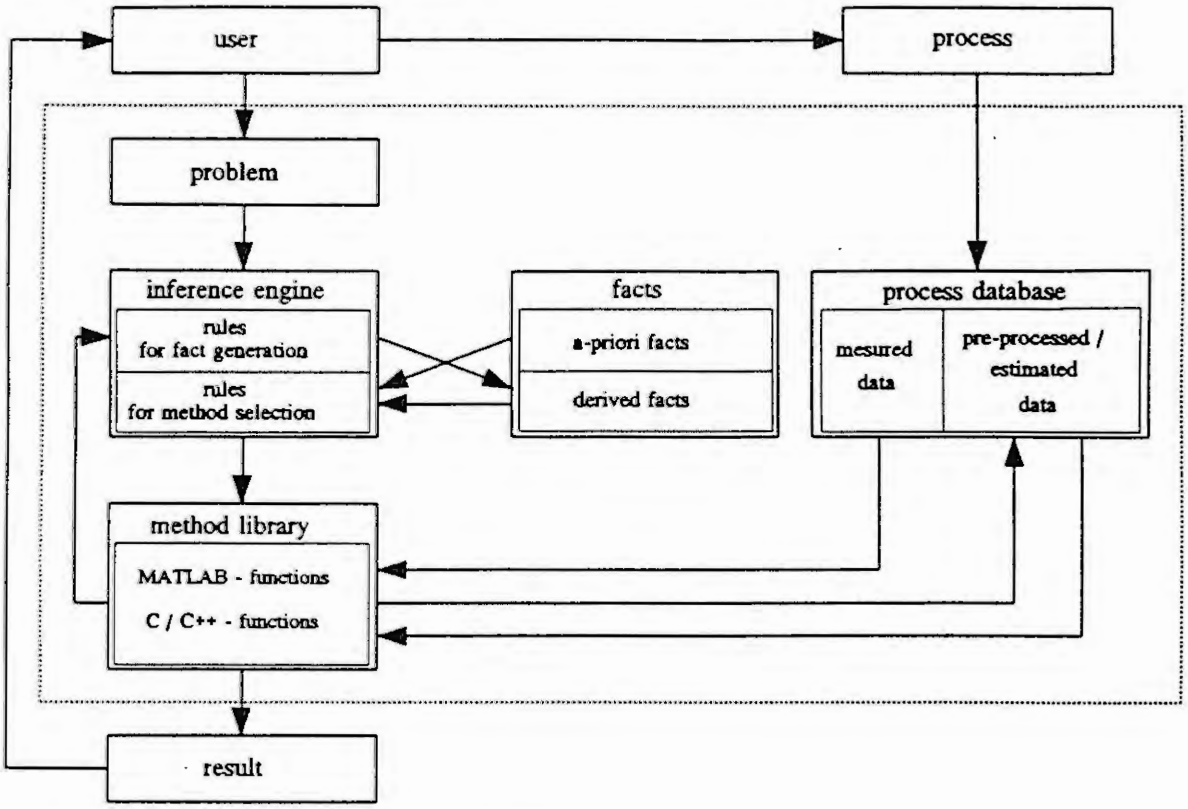


Fig. 1: Componets of the knowlege based System

for the user in a natural language way. They also appear in the conditional section of the rules for method selection, i.e. lead to the activation of the rules. They are stored in the fact base as object-attribute-value--triplets. With the object the kind of the fact described is characterised. It can be:

- a formal description of the system (e.g. dynamic system),
- a description of the available measured data (e.g. signal model),
- a characterisation of the properties of the methods being applied (e.g. test signal pattern).

The attributes characterise the features or properties of the objects, like for instance the strength of the disturbance, the auto correlation of the disturbance, the time variance of the parameters of the object "dynamic system". The values contain the specific properties as numerical values, sets of values, or qualitative statements (e.g. correlated/uncorrelated disturbance). Only those facts are generated that are required for the further processing (rule activation), that represent the result of problem being processed, or that contain information about the data being used. The facts are arranged in a way that depending on the respective values new objects can be generated. Therefore a structuring from common to increasingly more specific statements takes place.

2.2 Storage of methodological expert knowledge

The methodological knowledge is stored in DIOPRAN-EXPERT II as decision modules. They contain rules for the selection of methods or for the generation of facts, establish the dialogue with the user and the output of intermediate results [3]. The modules allow to capture the typical strategy of an expert in process analysis, which approximately turns out as follows:

- definition of the sub task,
- acquisition of the required data,
- analysis of the data with the methods,
- evaluation of the results obtained based on gathered experience,
- repetition of the previous steps until a satisfying solution is reached.

In the result of the rule decision a successive module is called. Therefore a module is executed only, if the necessary preconditions are met. This way a local restriction of the scope of the rules is achieved, i.e. only those facts are used, which contribute to the further decision. The rule base is therefore structured into subareas. This way the system can relatively easily be restructured or extended. For the notation of the modules a special syntax was developed, with which the methodological knowledge is stored in a readable form. This comprises the specification of the module name, the definition of required data, the list of functions, the definition of inputs and outputs, the definition of the rules for decision making as well as suggestions and comments to the theoretical background. Figure 2 shows the notation of a module for trend detection. In it first of all the signal is loaded from the process database. With the function "sigOrd" the order of the trend is determined and following that it is checked whether a trend is present and has to be corrected. The result of the investigation is stored in the fact base and output as an intermediate result. In presence of a trend a correction has to be made in the successive module.

```
:NAME             TestForTrend;
:DATEN            STRING <fTrend>;
                  EINGANG VEKTOR <signal>;
                  AUSGANG WERT <order>;
                  AUSGANG FAKT <fact> = ["signal model":"trend": <fTrend>];
:FUNKTION         <order> = sigOrd(<signal>);
:TEST             correction        <-  (<order> != 0);
                  nocorrection      <-  (<order> == 0);
:VERZWEIGUNG      correction        ->  <fTrend> = "present", AUSGABE correction,
                                        MAKEFAKT, MODUL TrendCorrection;
                  nocorrection      ->  <fTrend> = "not present", AUSGABE nocorrection,
                                        MAKEFAKT, MODUL Continue;
:EINGABE
:AUSGABE          correction "The signal contains a trend.";
                  nocorrection "The signal contains no trend.";
:BESCHREIBUNG     "The function 'sigOrd' is called. It calculates the order of a trend contained in the
                  signal. The order search is made via the assessment of the progress of the information
                  criteria and the certainty measure over the order.";
:ENDE
```

Fig. 2: Decision module for trend detection

2.3 Storage of procedural knowledge

The procedural knowledge is present as MATLAB and C or C++ functions, which are called during the execution of the modules [3]. To avoid reprogramming all methods the internationally accepted software package MATLAB was utilised. Missing procedures were programmed in MATLAB as M-files, in case they were mainly based on matrix operations. Algorithms that generate dynamic structures or contain many logical decisions were established as C or C++ functions. To the individual problems the following methods among others are applied:

- data pre-processing:
  * significance test,
  * test for normal distribution,
  * removal of data inconsistencies via statistical error limits,
  * filtering and trend correction;
- estimation of static system models:
  * determination of the order of precedence of input variables,
  * structure search via generation methods,
  * determination of the strength and auto correlation of the disturbance,
  * determination of the correlation of the inputs,
  * determination of the time variance of the parameters;

- estimation of dynamic system models (parameter estimation for the impulse response or difference equation model):
  * sample period check,
  * determination of the delay time,
  * model order search,
  * determination of disturbance strength and auto correlation;
- estimation of signal models (polynomial, periodic, and auto regressive models):
  * model order search,
  * determination of dominant frequencies;
- experiment strategies for static systems:
  * design of complete, partial and saturated experiment strategies of 1st and 2nd order,
  * design of Plackett-Burman and Hartley plans;
- test signal patterns for dynamic systems:
  * design of optimal test signal patterns for impulse response models based on Plackett-Burman plans,
  * design of PRB (pseudo random binary) sequences for difference equation models.

3. Software technological approach

The storage of knowledge in DIOPRAN-EXPERT II is in contrast to most of the already known systems which make use of expert-system shells combined with identification packages implemented using decision modules as described already earlier. For the execution of the decision modules corresponding C++ classes with functions belonging to them were developed. They are programmed such that the source code is usable without change on IBM-compatible PCs and workstations HP 9000. Only standard C++ and the class library "common view" by "Glockenspiel C++" were used. Therefore the system can run under Microsoft Windows and X-Windows. For the realisation of the algorithms MATLAB with its toolboxes SIGNAL PROCESSING, SYSTEM IDENTIFICATION, and CONTROL SYSTEMS were also available.

4. Assessment and future development

With the knowledge based system DIOPRAN-EXPERT II developed as part of the DFG project "Measured Data Information" a tool is made available which enables the user to sensibly select and apply appropriate methods in experimental process analysis. Based on the problem specification and depending on the current level of knowledge a sequence of investigations is undertaken to derive missing information and finally a satisfying solution of the problem. Experiments appropriate for the acquisition of process data required for model improvement are suggested. The present system can easily be extended with further methods. This is particularly recommended if particular preconditions must be met for the application of a method, if the user is required to enter suitable initial values, or an iterative solution requires intervention after each cycle.

5. References

[1] Meier zu Farwig, H.; Unbehauen, H.; Knowledge-Based System Identification. 9th IFAC/IFORS Symposium Identification and System Parameter Estimation, 1991, Budapest Hongary, Vol. 1.

[2] Wernstedt, J.; Otto, P.; Puhlman, R.; Roß, F.; A consulting/ Expert System for Experimental Process Analysis. 9th IFAC/IFORS Symposium Identification and System Parameter Estimation, 1991, Budapest Hongary, Vol. 1.

[3] Winkler, W.; Weiß, B.; Possekel, F.; Entwurf und Realisierung von Metehoden zur Transformation von Meßdaten in eine geeignete Wissensrepräsentationsform. Forschungsbericht für die deutsche Forschungsgemeinschaft, 1992.

This work is sponsored by the Deutsche Forschungsgemeinschaft, project-No. WE 1520.

MODERN TAYLOR SERIES METHOD AND STIFF SYSTEMS

Jiří Kunovský

Department of Computer Science and Engineering,
Faculty of Electrical Engineering,
Technical University of Brno,
Božetěchova 2, 612 66 BRNO, Czech republic,
E-Mail kunovsky@dcse.vutbr.cz

Abstract

A number of one-step, multi-step, explicite and implicite methods have been developed for solving stiff systems. They are mostly modifications of well-known numerical methods. This paper aims at evaluating the possibilities of solving stiff systems by the Modern Taylor Series Method.

1 Introduction

The best-known and most accurate method of calculating a new value of a numerical solution of a differential equation

$$y' = f(t, y) \qquad y(t_0) = y_0$$

is to construct the Taylor series in the form

$$y_{n+1} = y_n + h * f(t_n, y_n) + h^2/2! * f^{[1]}(t_n, y_n) + \ldots + h^p/p! * f^{[p-1]}(t_n, y_n),$$

where h is the integration step.

The simulation language TKSL/386 (an implementation of the Taylor Kunovsky Simulation Language on an Intel 80386 based personal computer) has been created to test an algorithm of the Modern Taylor Series Method.

The main idea behind the Modern Taylor Series Method is an automatic integration method order setting, i.e. using as many Taylor series terms for computing as needed to achieve the required accuracy.

2 Stiff systems

The paper lists a number of stiff systems of linear differential equations, nonlinear differential equations and differential equations with discontinuities. In all the cases the analytic solution is known to verify the accuracy of the computation. For each system a source code of the system in TKSL/386 will be shown. The results of the solutions will be given in corresponding figures. In the left part of each figure the solution as a function of time will be shown, in the right part of the figure separate values of time and of the variables will be listed (corresponding to the position of the cursor — a small circle in the graph).

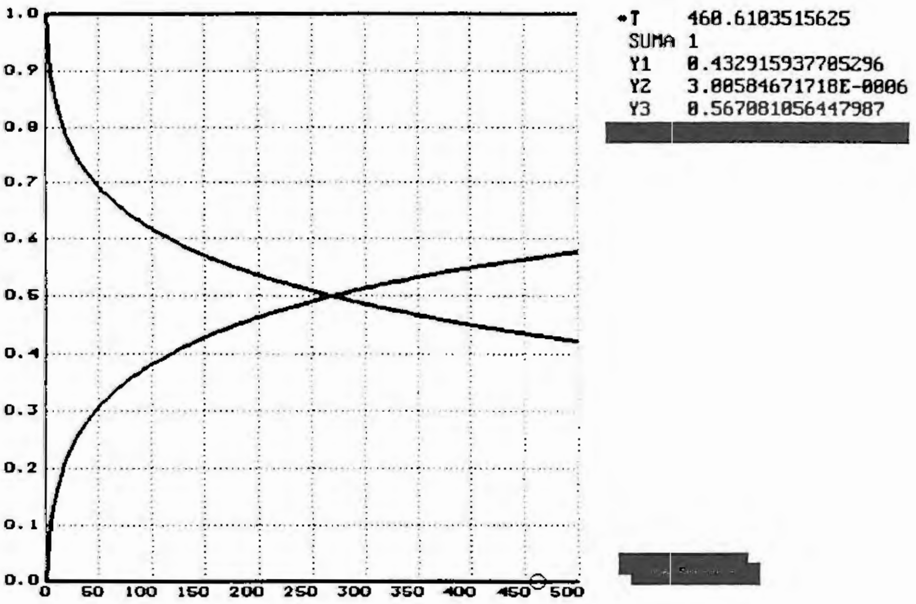
As an example the following nonlinear stiff system has been solved by the TKSL/386:

$$\begin{array}{ll} y_1' = -0.04 * y_1 + 10000 * y_2 * y_3 & y_1(0) = 1 \\ y_2' = 0.04 * y_1 - 10000 * y_2 * y_3 - 30000000 * y_2 & y_2(0) = 0 \\ y_3' = 30000000 * y_2 * y_2 & y_3(0) = 0 \end{array}$$

The corresponding source text in TKSL/386 is:

```
var     y1,y2,y3,SUMA;
const   tmax=1000,eps=1e-20;
system  y1'=-0.04*y1+10000*y2*y3             &1;
        y2'= 0.04*y1-10000*y2*y3_30000000*y2  &0;
        y3'= 30000000*y2*y2                   &0;
        SUMA=y1+y2+y3;
sysend.
```

The results are in Fig.1. It is typical of this system that $SUMA = 1$ in the entire time interval ($SUMA = y_1 + y_2 + y_3$).



```
•T     460.6103515625
SUMA   1
Y1     0.432915937705296
Y2     3.00584671718E-0006
Y3     0.567081056447987
```

Figure 1:

3 Conclusion

The Modern Taylor Series Method yields an extreme accuracy of the computation — the Modern Taylor Series Method used in the simulation language TKSL/386 increases the method order automatically, i.e. the values of the terms

$$h^p/p! * f^{[p-1]}(t_n, y_n)$$

are computed for increasing integer values of p until adding the next term doesn't improve the accuracy of the solution.

EQUATION FORMULATION AND SOLUTION METHODS BEHIND DYNAST - A MULTIPURPOSE ENGINEERING SIMULATION TOOL

Heřman MANN

The Czech Technical University, Zikova 4, CZ-166 35 Prague 6, the Czech Republic
Phone: +42-2-3112454, fax: +42-2-24310271, e-mail: mann@csearn

Abstract. *DYNAST is a versatile software tool for modeling, simulation and analysis of general linear as well as nonlinear dynamic systems, both in time and frequency domain. DYNAST admits system descriptions in the form of a set of equations, of a causal or acausal block diagram, of a multipole diagram respecting physical laws, or in the form combining the above approaches.*

DYNAST was designed as a '**simulation tool**' for practicing engineers rather then as a 'computational environment' for mathematically oriented enthusiast. Many operations were automated, and up-to-date computational methods were utilized to make the program robust and readily available for the most typical simulation tasks with the aim of distracting the user from the investigation of her/his dynamic system as little as possible.

DYNAST can be used as an **equation solver** for systems of nonlinear first-order algebro-differential and algebraic equations in the implicit form. The equations can be submitted in a natural way (without converting them into block diagrams) using a rich variety of functions including the Boolean, event-dependent and tabular ones. The equations, as well as any other input data, are directly interpreted by the program without any compilation.

The main purpose of DYNAST is, however, to simulate dynamic systems decomposed into subsystems defined independently of the system structure. The structure can be hierarchical. There are three types of subsystem models available in DYNAST. The **causal blocks**, specified by explicit functional expressions or transfer functions, are typical for any simulation program. But the variety of basic blocks is very poor in DYNAST, as its language permits to define the block behaviour in a very flexible way. Besides the built-in basic blocks, also user specified multi-input multi-output macroblocks are available. The causal block interconnections are restricted by the rule, that only one block output may be connected to one or several block inputs. In the DYNAST block variety, however, also **acausal blocks** are available with no restrictions imposed on their interonnections, as they are defined by implicit-form expressions.

The physical-level modeling of dynamic systems is based in DYNAST on subsystem **multipole models**. These models respect the continuity and compatibility postulates which apply to all physical energy-domains. (The former postulate corresponds to the laws of conservation of energy, mass, electrical charge, etc., the latter one is a consequence of the system connectedness.) The multipole poles correspond directly to those subsystem locations in which the actual energetic interactions between the subsystems take place (like shafts, electrical terminals, pipe inlets, etc.). The interactions are expressed in terms of products of complementary physical quantity pairs – the 'through' variables 'flowing' into the multipoles via the the individual poles, and the 'across' variables identified between the poles.

Thanks to the multipole modeling, the physical model of a system in the form of a **multipole diagram** can be constructed in a kit-like way from multipoles exactly in the same – isomorphic – way as the real system is formed from the corresponding real subsystems. The system structure is submitted to DYNAST by specifying the mutual incidence between the multipole poles (without any need for construction of a bond graph, for example).

The causal and acausal blocks as well as the multipoles can be combined to form **metamodels**. Any of the subsystem models can be stored in independent data files. This allows for creating **model libraries** for typical reusable subsystems at different level of their abstraction and idealization.

The equation formulation approach used in DYNAST both for multipole and block diagrams evolved from the 'extended method of nodal voltages' developed for electrical systems [5]. As DYNAST formulates all the equations of diagrams simultaneously, it has no problems with the 'algebraic loops'. As the formulated equations are in the implicit form, neither it makes any problems with the 'causality' of the physical models.

DYNAST uses only one, but robust and efficient **integration** procedure to solve nonlinear equations, both algebro-differential and algebraic (much more efficient that the procedures used in MATLAB, for example [7]). It is based on a stiff-stable implicit backward-differentiation formula [3]. During the integration, the step length as well as the order of the method is varied continuously to minimize the computational time while respecting the admissible computational error. Jacobians necessary for the integration are computed by **symbolic differentiation**. Their evaluation as well as their LU decomposition, however, is not performed at each iteration step if the convergence is fast enough. Considerable savings of computational time and memory are achieved by a consistent **matrix sparsity** exploitation.

The solution can start either from **initial conditions** specified by the user, or from the initial conditions computed by DYNAST and corresponding to a **steady-state** of the system, either static or periodic. In the former case, DYNAST automatically sets the derivatives of all the variables to zero. To accelerate the computation of periodic responses of weakly damped dynamic systems the iterative ϵ **algorithm** is utilized [4]. Also, fast Fourier transformation is available in DYNAST for **spectral analysis** of the periodic-steady state responses.

The **automatic linearization** of nonlinear equations by DYNAST allows for the **small-excitation analysis** of nonlinear systems in the vicinity of a user-specified or computed operating point. For the linearized systems, DYNAST is then able to compute numerically their rational **transfer functions** (and also transforms of their initial-condition responses) in a **semisymbolic form**. The functions are then expressed with the Laplace operator s as a symbol, and with the polynomial roots and coefficients as numbers. The computation is based on such transformations of the linearized system matrices [1], that the polynimial roots of the rational functions can be computed by solving the **two-set-eigenvalue problem** using the **QR algorithm**. To avoid the solution of the generalized eigenvalue problem, the formulated equations are reduced first using a sparse matrix transformation procedure [2].

For the resulting semisymbolic-form functions, the program can yield **semisymbolic-form frequency- and time-domain characteristics**. The latter characteristics are computed by partial fraction decomposition of the semisymbolic functions followed by the inverse Laplace transformation using closed-form formulas without resorting to any approximations.

In case of linear or linearized dynamic systems, DYNAST provides also an option for their **numeric frequency analysis** by direct solving the formulated equations at discrete frequency points. This approach is especially useful for **distributed-parameter** dynamic **systems** [6]. In this way, DYNAST is also able to compute **sensitivities** and **tolerances** of the frequency characteristics with respect to the system or ambient parameter deviations.

DYNAST runs on PC's, but it is easily transportable to other computers as it is coded in FORTRAN 77 and C languages. It is accompanied by a menu-driven **graphical environment** designed for the approach differentiated according to the user's experience. The block and multiport **diagrams** can be submitted **in a graphical form** by setting them up on the computer screen using a schematic capture editor. DYNAST can be easily augmented by various pre- and postprocessors because all its input and output data are available in the ASCII code.

DYNAST has been preceded by several simulation programs developed by groups headed by this author. In case of DYNAST and his imminent predecessor, SADYS, the author wishes to express his grateful **acknowledgment** first of all to the late Prof. Thomas Rübner-Petersen from the Danmarks Tekniske Hojskøle in Lyngby and to Dr. Zdeněk Oliva, the former author's graduate student. Also the feedback from the numerous users of the programs was very helpful.

References

[1] Mann, H.: *An algorithm for the formulation of state-space equations.* Proc. 1979 Int. Symp. on Circuits and Systems ISCAS IEEE, Tokyo 1979, pp. 161-162.

[2] Rübner-Petersen,T.: *SFORM1 and SFORM2 – two FORTRAN IV subroutines for sparse matrix transformation of the general eigenproblem to standard form.* Report IT-41, DTH, Lyngby 1979.

[3] Rübner-Petersen, T.: *ALGDIF – a FORTRAN IV subroutine for solution and perturbated solutions of algebraic-differential equations.* Report, DTH, Lyngby 1979.

[4] Skelboe, S.: *Time-domain steady-state analysis of nonlinear electrical systems.* Proc. IEEE 70 (1982), pp. 1210-1228.

[5] Mann, H.: *Analysis of combined circuit-block diagrams.* Proc. Int. Symp. on Circuits and Systems ISCAS IEEE, Rome 1982, pp. 639-642.

[6] Mann, H.: *Computer applications in electrical engineering design* (in Czech). SNTL Publishing House, Prague 1984.

[7] Mann, H.: Comparison 1 - DYNAST. *EUROSIM Simulation News Europe*, November 1991, p. 32.

A model of a centrifugal separator for corn

V.I. Gorelov

Moskow Institute of Food Technology
Ryazanscy prospect, 70-2-24, 109542, Moskow,Russia

Abstract. A model of centrifugal separator is considered to improve technological processes of separation in food industry without considerable expenses. Trajectory, stability, control parameters and other main motion propeties are defined for this model.

Corn separation process is the division of mixture in fractions with sieves. The existence of the force being able to push fractions through sieves is indispensable condition of division. On the whole, it's the gravitational force with small vibrations. Intensification of separation can be realized with help of introduction more powerful centrifugal force than gravitational one. We can express that this force easily yield to the modification and sieves form is a cylinder (from condition of technical simplicity).

I consider the relative motion of the rough solid in the interior sufrace of cylinder with vertical axis rotating by law

s=ut+asinwt.

In this case differencial equations of the motion have the form

$$\begin{cases} \ddot{z}=g-fr(u+a\omega\cos\omega t+\dot{y})^2\dfrac{\dot{z}}{\sqrt{(r\dot{y})^2+\dot{z}^2}} \\ \ddot{y}=a\omega^2\sin\omega t-f(u+a\omega\cos\omega t+\dot{y})\dfrac{r\dot{y}}{\sqrt{(r\dot{y})^2+\dot{z}^2}} \end{cases}$$

where g - acceleration of the gravitation;

f - coefficient of the fraction;

r - radius of cylinder;

z - displacement down;

ry - displacement by sufrace.

Depending on the relation between u and aw there are 4 possible variants of the motion: stoping, the motion with pauses, the continuous motion, free falling. The continuous motion is the best case for our separation.

The system of differential equations is solved by introducing small parameter $e=fu/w$ from the change of variables and time

$t_1 = wt,\ z=rp,\ y=aq$

Arbitrary integral constantes are defined with help of integral constantes of averaged system. The solutions have the form

$z=F_1\ (r,w,a,u,f,t),$

$y=F_2\ (r,w,a,u,f,t).$

There solutions have stability with respect to small perturbations.

If we have optimization test in the form productivity of separators; that control law for the parameter a is an algebraic equation.

MODELLING OF THE POWER CONSUMPTION IN AGROINDUSTRIAL COMPLEX

L.Batirmurzaeva
Institute of Mathematics, Moldova

The intensive rise of consumption of the power resources intensistently demands to find the ways of their economy. This problem became especially pressing for Republic Moldova. Lack of the own fuel power resources and rapid rise in their prices make worse the complicated state of the republic economy.

The proposed mathematical model of the regional AIC serves as a base for solving of this problem.

The designations for description of the model of the AIC are: i - index of items of the agricultural production ($i \in I$); I - set of indexes of items of the agricultural production ($I = I_1 \bigcup I_2 \bigcup I_3 \bigcup I_4$); I_1 - subset of indexes of items of the vegetables ($I_1 \in I$); I_2 - subset of indexes of items of the perennial plantation ($I_2 \in I$); I_3 - subset of indexes of items of the stock-raising production ($I_3 \in I$); I_4 - subset of indexes of items of the bread grains, technical and fodder crops ($I_4 \in I$); j - index of items of the industrial production ($j \in J$); J - set of indexes of items of the industrial production; t - index of directions of utilisation of the agricultural production ($t \in T$); T - set of indexes of directions of utilisation of the agricultural production ($T = T_1 \bigcup T_2$); T_1 - subset of indexes of directions of utilisation of the agricultural production for the local market ($T_1 \in T$); T_2 - subset of indexes of directions of utilisation of the agricultural production for the export ($T_2 \in T$); T_3 - subset of indexes of directions of utilisations of the agricultural production for the industrial production ($T_3 \in T$); n - index of items of technical and biological resources ($n = 1, 2, \ldots, N$); f_i - crop yields or productivity of one head of the animal; A - land under crop; P_i - area to the i perennial plantations or numbers of the basic head of the animal giving the i production; a_{ij} - rate of inputs of i agricultural production for output of unit of j industrial products; Q_i - demand of inhabitants for i products; Q_j - demand of inhabitants for j food products; C_i - volume of export of the i agricultural products; C_j - volume of export of the j industrial products; b_{ni} - expenses of the n resource for output of unit of the i agricultural products; b_{nj} - expenses of the n resource for output of unit of the j industrial products; H_n - quantity of the n resource at the beginning of the forecasting period; g_i - power capacity of unit of the i farm products; g_j, g_j^* - power capacity of unit of the j industrial products for the local market and export; g_i^* - bioenergy content of the agricultural land in 1 hectare or 1 head of the animal for the production of the i agricultural products; g_n - power capacity of unit of the n resource; x_i - land under i crop in forecasting period ($i \in I \setminus I_3$); X_{it} - volume of output of the i agricultural production for the t direction of utilisation ($\sum_{t \in T} X_{it} = X_i$); χ_j, χ_j^* - volume of the output of the j industrial products for the local market and export; Y_i - increase of the land under perennial plantations or increase of the basic head of the animals giving the i agricultural products in the forecasting period; Y_n - increase of the n resource in the forecasting period.

It is necessary to determine minimum of the power expenses:

$$F = \sum_{i \in I} g_i X_i + \sum_{j \in J} g_j \chi_j + \sum_{j \in J} g_j^* \chi_j^* + \sum_{i \in I_2 \bigcup I_3} g_i^* Y_i + \sum_{n}^{N} g_n Y_n.$$

Conditions of the problem:

1. Balance of production and consumption of the agricultural products:

$$X_i - f_i x_i = 0, \qquad i \in I_1 \bigcup I_4,$$

$$X_i - f_i Y_i = f_i P_i, \qquad i \in I_2 \bigcup I_3.$$

2. Conditions of utilisation of the agricultural raw materials for the processing industry:

$$\sum_{j \in J} a_{ij}(\chi_j + \chi_j^*) - \sum_{t \in T} X_{it} = 0, \qquad i \in I.$$

3. Conditions of satisfaction of the solvent demand of inhabitants for foodstuffs:

$$\sum_{t \in T_1 \setminus T_3} X_{it} \geq Q_i, \qquad i \in I \setminus I_4,$$

$$\chi_j \geq Q_j, \qquad j \in J.$$

4. Conditions of export of the agroindustrial products:

$$\sum_{t \in T_3 \setminus T_1} X_{it} \geq C_i, \qquad i \in I,$$

$$\chi_j^* \geq C_j, \qquad j \in J.$$

5. Conditions of utilisation of the technical and biological resources for the production of agroindustrial products:

$$\sum_{i \in I} b_{ni} X_i + \sum_{j \in J} b_{nj}(\chi_j + \chi_j^*) \leq H_n$$

   (for hard limited resources) or

$$\sum_{i \in I} b_{ni} X_i + \sum_{j \in J} b_{nj}(\chi_j + \chi_j^*) - Y_n \leq H_n$$

   (for less limited resources).

6. Non negativeness of variables.

At present the multiversion statement of a problem is expedient. It allows to use several criteria and different values of the limiting conditions.

On Analysis of r x c x t Contingency Tables

Hedayat Yassaee, Sharif University of Technology, Iran.

October 26, 1993

In this paper we plan to analyze data which are obtained in r x c x t contingency table form. Data are obtained in this fashion. Suppose that we have N observations which are categorised in t different classifications in a random fashion. Then we classify each observed categorised frequency into an, for example, $m_k \times n_k$ contingency table form. The main perpose of this paper is to analyze data according to some models. We use some distribution and models on probability cells and test some hypotheses on parameters.

Analysis of contingency tables has brought attentions of several research workers. Some have investigated on estimating cell frequencies of tables and other have studied on parameters of models used in analysis of contingency tables and some researchers on both. We shall give references which are recently published, see Kullback and Cornfield (1976), Goodman (1970), Koch (1973), and (1969), Yassaee (1977).

In this paper we assume that there are N observations which are categorized according as a multinomial distribution. These observations may be taken from a continuos distribution or even a multivariate distribution. We assume there are t categories. We let p_k be the probability that an observed value from N is categorised in kth classification. If we take $f(x)$ to be the probability density function of the distribution in which N observatin is taken from, then we define kth category as the interval (x_{k-1}, x_k) and

$$p_k = \int_{x_{k-1}}^{x_k} f(x)dx$$

We define N_k be the frequency of observations in kth category. Now, we classify N_k observations in $m_k \times n_k$ contingency table form. We assume p_{ijk} be the probability that an observation in N is categorized in ijkth cell. Suppose we use the model

$$p_{ij(k)} = \frac{p_{ijk}}{p_k} = \mu + \alpha_{i(k)} + \beta_{j(k)} + \gamma_k + \epsilon_{ijk}$$

$$\sum_i \alpha_{i(k)} = \sum_j \beta_{j(k)} = 0, \quad i = 1, 2, \cdots, m_k; \; j = 1, 2, \cdots, n_k$$

We plan to estimate the model as well as testing on the model and parameters involved in the model. We use a new definition of discrimination information statistic to estimate the model and its parameters. We take all $m_k \times n_k$ table to be $r \times c$ contingency tables.

$$\begin{aligned} I^* &= \sum_i \sum_j \sum_k N p_{ijk} \ln \frac{N p_{ijk}}{n_{ijk}} \\ &= \sum_k \sum_i \sum_j N p_k p_{ij(k)} \ln \frac{N p_k p_{ij(k)}}{n_{ijk}} \end{aligned} \tag{1}$$

In the model we used, we have $p_k p_{ij(k)} = p_k(\mu + \alpha_{i(k)} + \beta_{j(k)} + \gamma_k)$. For simplicity, we take $\alpha_{i(k)}$ and $\beta_{j(k)}$ as α_i and β_j. Therefore,

$$\begin{aligned} I^* &= N\sum_i \sum_j \sum_k p_k(\mu + \alpha_i + \beta_j + \gamma_k) \ln \frac{N p_k(\mu + \alpha_i + \beta_j + \gamma_k)}{n_{ijk}} \\ &= N\sum_i \sum_j \sum_k p_k(mu + \alpha_i + \beta_j + \gamma_k)(\ln \frac{N}{ijk} + \ln p_k + \ln(\mu + \alpha_i + \beta_j + \gamma_k \end{aligned} \tag{2}$$

We now minimize this distrimination information statistic with respect to parameters $p_k, \mu, \alpha_i, \beta_j, \gamma_k$.

To give an application of our research problem, suppose that N units of a currency money is specified as the budget for spending on technical and theoretical educational programs in t regions. In each region there are rc different categories of technical-theoretical program. Suppose that the probability that a unit of the money is devoted to kth region is p_k and assume that given this unit of money for kth region, the probability that it is spent for ijth category of technical-theoretical program is $p_{ij(k)} = \mu + \alpha_i + \beta_j + \gamma_k$. Then $p_{ijk} = p_k p_{ij(k)}$, where p_{ijk} is the probability that a unit of the budget N is spent for ijth category at kth region, or simply ijth category. If we are given n_{ijk}'s, the budget for each possible classification, then we may be interested in investigating whether the model is applicable (or it is applied for distributing the budget N in this program).

1 References

Goodman, L. A. (1970), The Multivariate Analysis of Qualitative Data: Interactions among Multiple Classifications, JASA 65, 226–56.

Koch, G. G. (1969), The Effect of Non-sampling Error on Measures of Association in 2 Contingency Tables, JASA 64, 581–64.

——— (1973), An Alternative Approach to Multivariate Response Error Models for Sample Survey Data with Applications to Estimators Involving Sub-Class Means. JASA 68, 906–13.

Yassaee, H. (1977), On Properties of Estimators in $rxcx2$ Contingency Tables: Logit Linear Model IRIA: Data Analysis and Informatics, 329–37.

——— (1978), On Matrices Whose Row and Column Totals Given. Technical Report. Sharif University of Technology, Tehran, Iran.

CONTROLLED BIFURCATION TRANSITIONS FOR SIMULATION OF CHAOTIC AND COMPLEX PERIODIC OSCILLATIONS

V. Gontar and M. Gutman

The International Group for Scientific and
Technological Chaos Studies (IGCS), Ben-Gurion
University of the Negev, Beer-Sheva,
P.O. Box 1025, 84101 Beer-Sheva, Israel

We propose a method for obtaining solutions of one-dimensional discrete maps in the special form of quasi-continuous orbits (QCOs), the segments of which resemble orbits of differential equations. These QCOs may be effectively applied for computation simulation of complex dynamical processes. The proposed technique also has heuristic value, in that it can be used to obtain new types of oscillation modes and new bifurcation phenomena [1]. Some of these orbits have been examined by one of present authors (V.G.) for discrete map modelling of the dynamics of physico-chemical processes [2].

We demonstrate our method for the discontinuous logistic map:

$$x_{n+1} = p_1 x_n (1 - x_n) \bmod p_2, \qquad (1)$$

whose phase diagram is given in Fig. 1a (p_1, p_2 are the control parameters). This scheme corresponds to the conditions necessary for type V intermittency, where the point of discontinuity is a stable fixed point [3]. The presence of a "trap"—the right-hand side of the scheme crosses the bisector twice—gives the orbits a form characteristic of intermittency, that is, with upper laminar (continuous) segments.

At the same time, the discrete orbit under consideration can be interpreted as the Poincaré map generated by the heteroclinic phase trajectory of a three-dimensional dynamic system coming out of a single saddle periodic motion and entering another. As a result, one obtains a rich selection of chaotic and periodic orbits of varying complexity sharing one specific feature—long laminar segments (Fig. 2a, b, c). We use the function mod p_2 to obtain the above-mentioned heteroclinic motions between two saddle fixed points. The value of the parameter p_2 is determined on the initial bifurcation diagram $\left(\sum_{n=1000}^{1500} x_n = f(p_1),\ p_2 >> x_{max}\right)$ corresponding to its maximum value x_{max} at the selected point of the bifurcational transition (BT). Next, a BT is constructed for the map with the obtained value of p_2 (controlled BT), and the value of the parameter p_1 corresponding to the QCO accumulation region of required shape is determined on this BT. For example, if we select the value of the p_2 at a point corresponding to the second doubling bifurcation (Fig. 3a), we obtain orbits with laminar segments of the second order (every second iteration of an upper segment lies on a smooth curve) (Fig. 4a); and we obtain orbits with gaps at every second peak (laminar segments of the 3rd order) (Fig. 4b) if we select p_2 at the point of third doubling bifurcation (Fig. 3b).

This method makes it possible to combine laminar segments of different orders in a single orbit. Thus, for the following logistic map with two points of discontinuity:

$$x_{n+1} = p_i x_n (1-x_n) \bmod p_{i+1}, \qquad (2)$$

where $i = 1$ for $x_n < p_2$ and $i = 3$ for $x_n \geq p_2$, the phase portrait of which forms two "traps" (Fig. 1b), we obtained orbits with laminar segments of order 1 and 2 (Fig. 5). The control parameters p_1, p_2 and p_3, p_4 for this orbit are respectively equal in value to p_1 and p_2 for the orbits in Fig. 2b and Fig. 4a.

A natural extension of the method might be the use of the parameter p_2 as a complex function, e.g. $p_2 = f(p_1)$. Further, we demonstrate that it is possible to obtain BTs of different types by means of appropriate functional manipulation of the control parameters and argument of the initial one-dimensional map. Application of controlled BTs to maps transformed in the indicated manner is another way of modifying the form of quasi-continuous orbits.

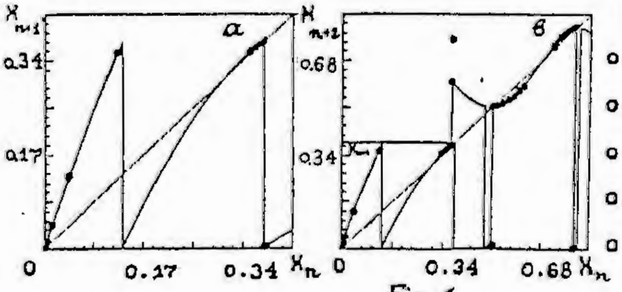


Fig. 1

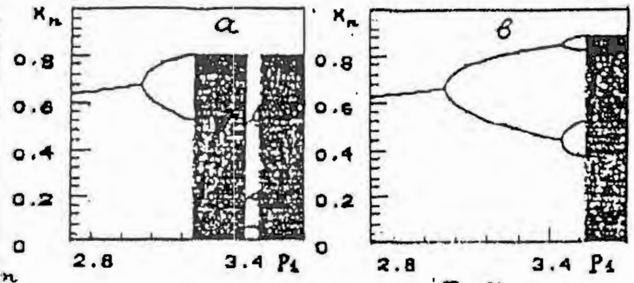


Fig. 3

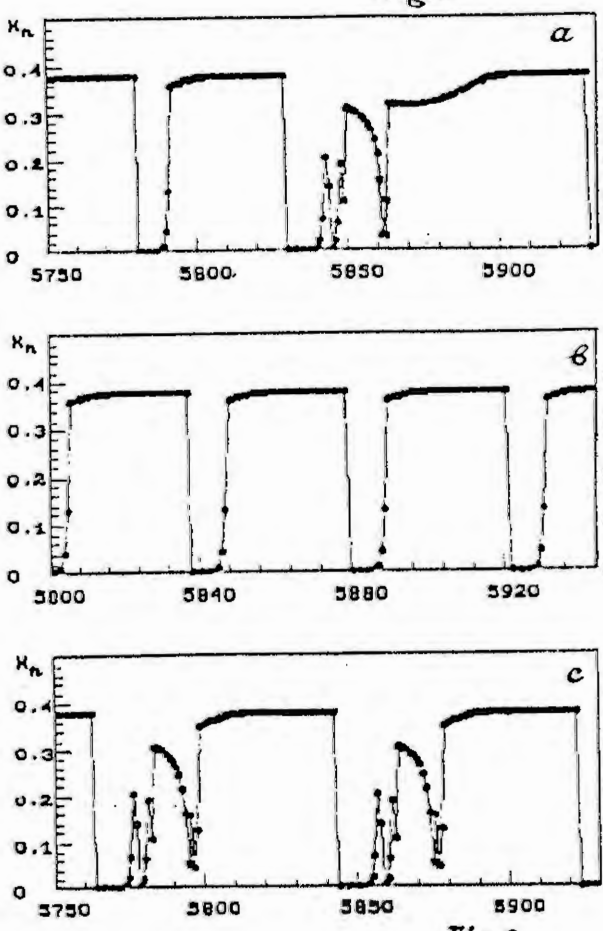


Fig. 2

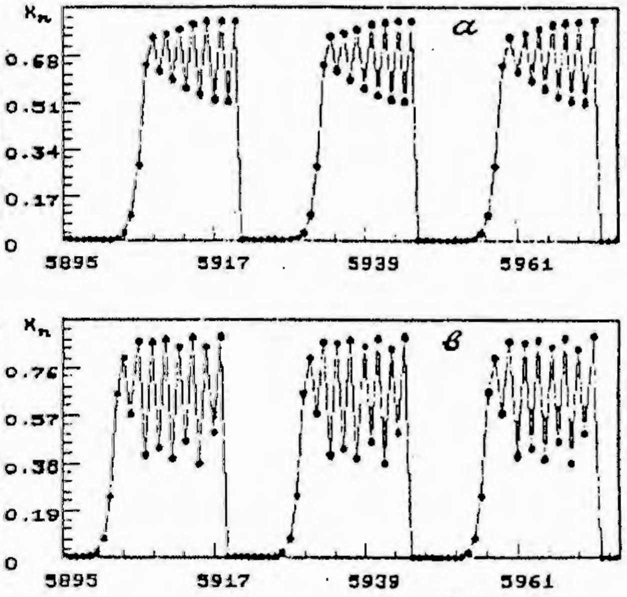


Fig. 4

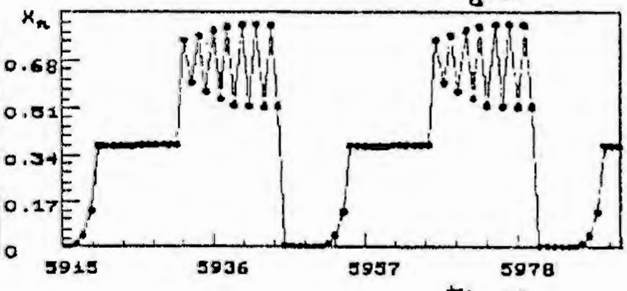


Fig. 5

References

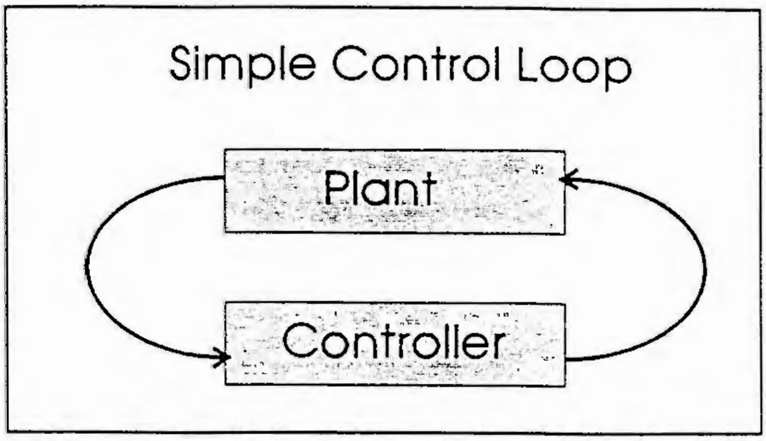
1. Gutman, M. and Gontar, V. (1993) Route to chaos via cascade of continuous inverse bifurcations. Submitted to Int. J. Bifurcation and Chaos.
2. Gontar, V. and Ilin, A. (1991), New mathematical model of physico-chemical dynamics. Contrib. Plasma Phys. 31, 6, 681-690.
3. He, D.R. et al. (1992) Type V intermittency, Phys. Lett. 171A, 61-65.

The Parallel Simulation System "mosis"

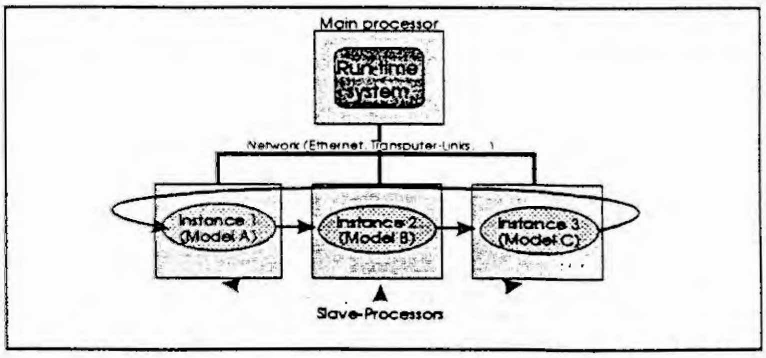
G.Schuster, F.Breitenecker
Technical University of Vienna, Dept. Simulation technique

"mosis" is a simulation system for continuous systems with the aim of efficient parallelisation of simulation models on a multi-processor network with distributed memory. It is based on the so-called "Model Interconnection Concept" which provides modular model development for big simulation models and is based on an object-oriented point of view. This has been a research project on the Technical University of Vienna and its first step - the first release of the free experimental simulation system "mosis" - will soon be completed.

The name of the system - "mosis" - stands for the term "modular simulation system" which also explains the main concept based on. The "model interconnection concept" assumes that every large simulation model (e.g. the description of a complex mechanical system) consists of several smaller models that are interconnected via special uni-directional links (where the data are transferred). The part models can be implemented and tested independently and as a last step they can be linked together to describe the whole system.



A very easy example of this is the algorithmic description of a *control loop*: one part describes the plant; its output is used by the controller (represented by the second model) - its output is sent to the first model which changes the behavior of the real system.



In this concept, the definition of a *model* is a very general one: It can be used for very different things, as

- the usual meaning of a model: the mathematical description of a real system in a continuous and/or time-discrete manner.
- a test model (i.e. a model specially implemented to verify the behavior of another model) or a model for statistical evaluation the output of other models
- a function table or even a single constant
- even an interface to the real world via A/D and D/A converters (Hardware in the loop)

In mosis, models are defined in a CSSL style simulation language based on "C". They are translated to "C", compiled and linked with the run-time system. In the resulting program they can be prepared for simulation. But when the system is started, the information about the models is only inherently present on the main processor. In order to simulate it, an instance must be created. (Object oriented view: Class definition -> object creation). Those instances can be connected with (physical or logical) links. With this, it is possible to create more than one instance of a single model, even on different processors in the network. This can be useful especially for parameter studies, optimization etc.

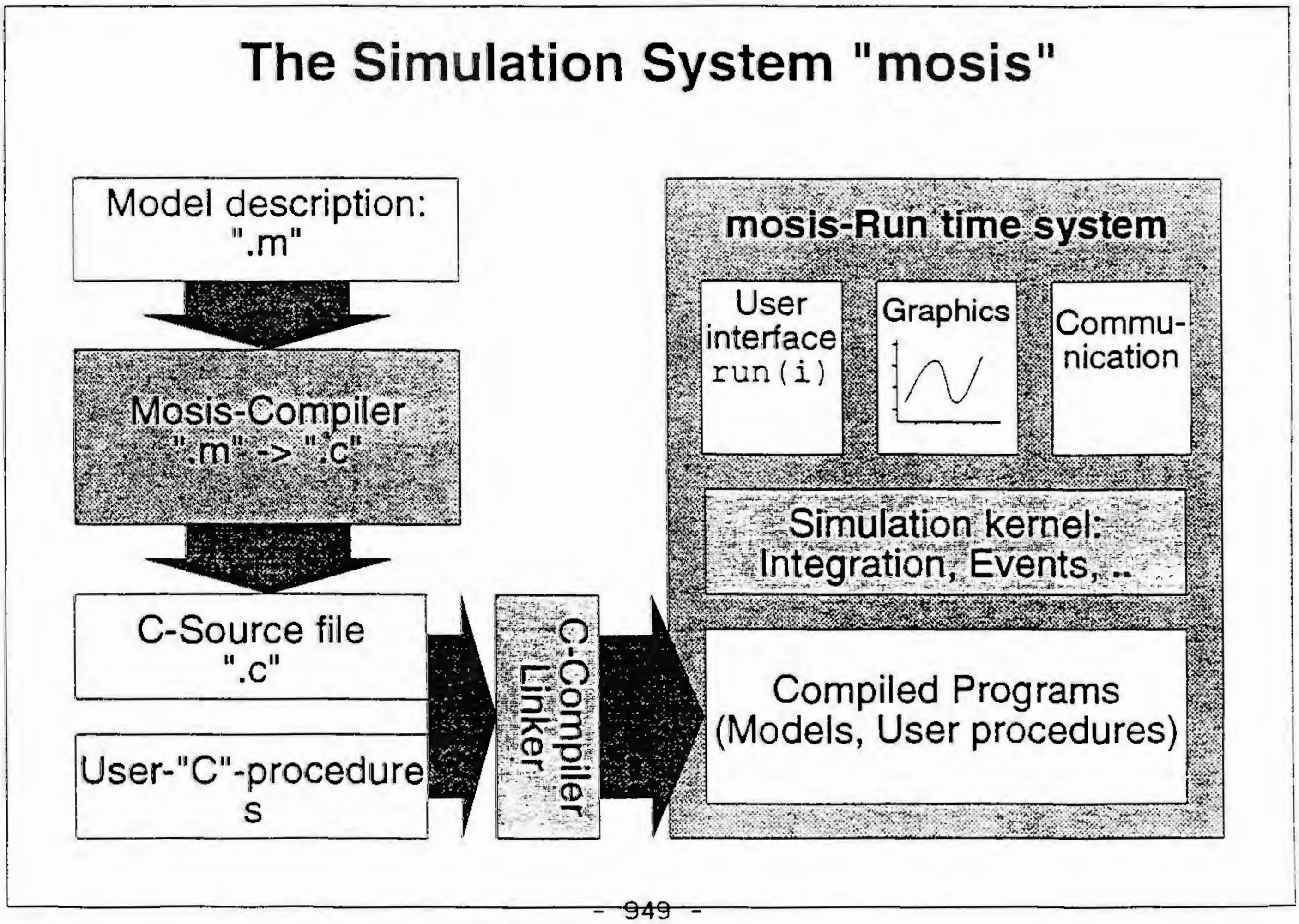
The biggest problem is to keep data interchange between the instances as little as possible; therefore two restrictions have to be done:

- the communication is done strictly one way from output to input (no questions back)
- data transfer is only done at fixed times (usually at equidistant time intervals)

These restrictions result on special care that must be taken for choosing the integration algorithm: When the values of input signals are used between two communication points, they have to be interpolated/extrapolated which can result in numerical problems.

Efficiency: Care has to be taken at the partitioning of the whole problems into small models. They should not be too big (which limits the maximum number of processors that can be used in a useful manner) but on the other hand they should not be so small that the communication between the processors takes much more time than the calculation of the state variables. Besides that, the user should care that most instances are capable to calculate the output values at time t_{i+1} with the inputs only from time t_i. If this is not possible, the second model has to wait for the results of the first one; this would delay the whole execution.

"mosis" has been implemented on PC's with MS-DOS, UNIX-Clusters under PVM and XWindows and the Transputer System Cogent XMP. It will be freely available on the simulation server of the Technical University of Vienna.



The Simulation System "mosis"

Optimization in Discrete Event Simulation
- An approach with Genetic Algorithms

M. Salzmann, F. Breitenecker
Technical University of Vienna, Dept. Simulation technique

One of the first scientists that worked on optimization using Genetic Algorithms (GAs) was John Holland. In 1975 his work consisted of two main topics:

- encoding complicated structures with simple representations
- looking for simple transformations to improve those structures

Holland proofed that GAs are probabilistic algorithms which start with an initial population and improve the solutions of problems by means of genetic operators. These operators are similar to natural reproduction. In many cases the initial population consists of a certain number of solutions represented by bitstrings and a simple genetic operator would be "crossover": Two parent solutions are taken, a cross site is chosen and an exchange of parts of bitstrings is performed (see figure below).

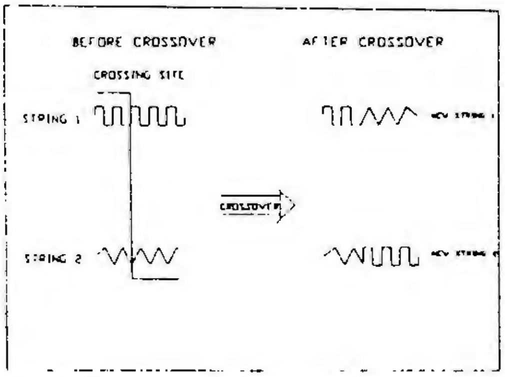


FIGURE 1 A schematic of simple crossover shows the alignment of two strings and the partial exchange of information, using a cross site chosen at random

This and many other genetic operators are the backbone of the GAs which can be applied in a wide range of applications.

One of these applications is the discrete event simulation. For example the job scheduling problem:
Certain jobs are given which have to be done until different points of time. If there is a delay a fine has to be paid. The problem now is to minimize the costs on the production and therefore to avoid this fine.
A solution is reached the following way:

- An initial population of possible solutions is chosen
- A couple of simulation runs (e.g.: using GPSS/H) is performed
- The total costs (= fitness function) are computed and stored
- Using selection, crossover and mutation a new population is generated
- This continues until a satisfying job schedule is found

Literature:
Bloc, L., Cytowski, J. : Search Methods for Artificial Intelligence.
Goldberg, D. : Genetic Algorithms in Search, Optimization, and Machine Learning.