SIMULATION-AIDED PROCESS COVERAGE FOR DELIVERY RELIABILITY UNDER SHORT DELIVERY SCHEDULES USING REAL-TIME EVENT BASED FEEDBACK LOOPS

Wilhelm Dangelmaier[1], Andre Döring[1], Christoph Laroque[2], Thorsten Timm[2]

[1]Heinz Nixdorf Institute, University of Paderborn, Business Computing esp. CIM
Fürstenallee 11, 33102 Paderborn, Germany
[2]Fraunhofer Anwendungszentrum für Logistikorientierte Betriebswirtschaft (ALB),
Fürstenallee 11, 33102 Paderborn, Germany

andre.doering@hni.uni-paderborn.de (Andre Döring)

Abstract

This paper describes an approach for collaborative, automated process coverage for event handling in supply networks, based on the integration of an artificial intelligence method, simulation techniques and change planning strategies within an integrated force-feedback loop. The artificial learning system, based on Q-Learning and state abstraction, is introduced, learning state based rule sets for controlling application of change planning algorithm in an event situation. The state abstraction is using k-means clustering evaluating the discrete production plans using a combined weighted quantitative and structural distance function. Using the rule set the relevant scenarios, gained from possible alternative change planning strategies, can be selected automatically for simulation based process coverage. These scenarios will be analyzed through a material flow simulation. According to the specific requirements of real-time reaction to events, it will be discussed whether a dynamic simulation model, scaling based on the selected scenarios, could be used to optimize simulation time to speed or result reliability. Therefore actual research activities will be discussed. A Production Control Center architecture is introduced to integrate all components into a vision of a future control system. To position this paper and future work on one basis, the upcoming and here partly stressed problems of foresight and real-time event handling, implemented by a combined simulation-learning-optimization system, will be classified and structured into a system matrix.

Keywords: Supply chain management, Process coverage, Event handling, Simulation

Presenting Author's biography

Dipl.-Inform. Andre Döring studied informatics with special focus on automatic language processing, machine learning and risk assessment at the University of Bielefeld. Since 2004 he is research assistant at the University of Paderborn, business computing esp. CIM.

ISBN 978-3-901608-32-2

1 Introduction

In today's competition, automotive OEMs and their suppliers are confronted with the demand for shorter delivery times, product lifecycles and a higher number of variants in parts and products. The globalization enables new market chances, but also increases the pressure of competition on the European Automotive Industry. Outsourcing of human resources and processes from the OEMs to their suppliers or other service providers is one consequence; the concentration on core competences is elevating the impact of the efficiency in adding value and also of the efficiency of planning and control of the value chain.

The result is an increasing reduction of the inter-enterprise vertical range of manufacturing. The number of companies involved in the value adding processes of the supply chain increases and consequently the co-ordination processes handling the inter-enterprise material flow become more complex.

The study FAST 2015 [1] stresses the potential in costs and delivery time, which arises from a decrease in process lead times and stabilization of planning and execution processes. Maximal delivery reliability must be enabled by a higher and more efficient collaboration between the companies in the automotive supply network.

One main aspect in the collaborative supply network management is the clearing of events occurring during supply network execution. These clearing processes must be highly reliable to ensure the operation of the entire production network. Today, this is implemented by supply network control processes using change-planning [2]. The predicted effects on production plans calculated by those change planning strategies are depending on the event scenario, the supply network configuration and the type of used change algorithm. Combined with a material flow simulation system like d³FACT insight [3], the different emerging scenarios using the different change planning strategies [4] can be simulated and analyzed to cover the change planning strategies. But processing these emerging scenarios by applying change planning algorithms with simulation methods is a challenging task. It will be more *effective* to cover only those scenarios regarded as most effective for clearing the event.

The selection of those scenarios could be done manually e.g. by a production planner or simulation expert. In order to increase the *efficiency* in covering change planning strategies enabling near real-time reaction an *automated* selection of scenarios used for process coverage is suggested.

This paper describes a concept using change planning algorithms for event handling scenario generation used in a simulation component for real-time process covering. The selection of the scenario will be done with an introduced artificial intelligence system module based on reinforcement learning, learning its rule set for controlling the change planning scenario generation by selecting the best algorithm for event handling. All modules are integrated into a force feedback loop.

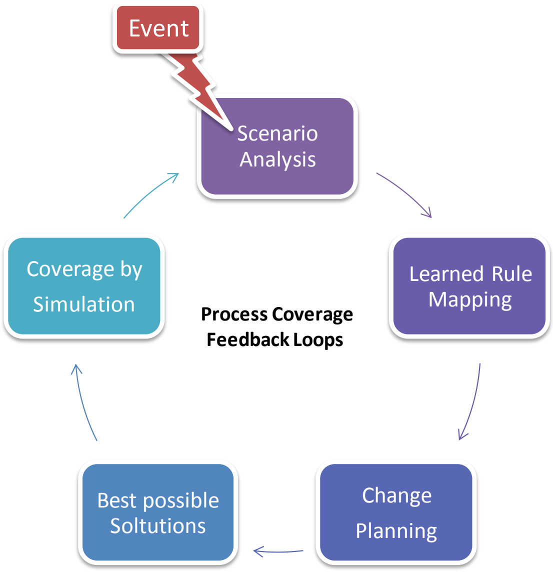


Fig. 1 Force-feedback loop for event management

2 Problem definition

Through occurrence of events, the validity and consistency of production schedules can change tremendously during supply chain execution. These changes can be for example:

- Short-term changes of demand or supply values
- Blackout of capacities caused by machine breakdowns
- Material based loss of supplies caused by logistical delivery problems
- ...

Within supply chain event management, these events are handled by a supply chain event management processor, which executes event-corresponding clearing actions. However, currently existing event handling software systems are only aligned to the use in simple supply chains, not in complex collaborative process environments (supply networks).

The fast and reliable management of events in more dense and inter-enterprise automotive manufacturing networks demands for a new consideration of this topic regarding the influence of network based factors. It is obvious, that only more cooperation and collaboration implemented by efficient, overhead minimizing and robust processes will lead to an efficient event management in supply networks.

The event management and clearing will be done by *change planning strategies* generating various alternative event-cleared production plans, called *scenarios*. To ensure the reliability of these scenarios, they can be safeguarded by a material flow simulation, based on the current production network configuration. Based on the size of the network simulation model and the simulated time periods this task is highly computing intensive [5], increased by the number of scenarios being assessed. The number of scenarios to be assessed depends on the number of applied change planning strategies because every change planning strategy will generate a new alternative scenario which has to be covered by simulation.

It can be summarized that using simulation for process coverage will increase the reliability of production plan scenarios but decrease the reaction time far from real-time event handling. Also the process starting with scenario selection, simulation model generation and at least simulation is interrupted by manual processes done by a production planner or simulation expert. To advance the reaction times for event handling, it is regarded helpful to optimize this manual process by using process automation and integrate the whole clearing process into a force-feedback loop for processing events and covering changed plans as fast as possible.

To fulfill this task several problems arise:

1. A set of change planning algorithms is needed that can be applied on a production plan according to the change planning strategy.
2. A measure is needed to reduce the possible scenarios for simulation to a processable number covering the most promising scenarios for event handling generated by the change planning stratgies.
3. A simulation environment is needed to generate simulation models according to the scenarios that are dynamically configurable, regarding the time horizon offered for simulation. The level of detail of the simulation model must be adaptable to speed (rough model, fast processing, short time horizon for simulation) or reliability (detailed model, accurate processing, longer time horizon for simulation)
4. A foresighted event handling module to increase the time period usable for scenario generation and process coverage by simulation.
5. A real-time event detection system connected to the change planning system.
6. A production control center as an umbrella integrating all modules.

To enable real-time reaction to events by applying change planning strategies the used change planning algorithms must be designed lean and highly effective.

The measure assessing the scenarios for selecting the most effective change planning algorithm could be implemented by a machine learning system like reinforcement learning (RL) [6]. RL is good for learning rule based decisions in complex models. But in the case of production network plans as the state model the RL must handle a high number of states that leads to extremely long processing times. Due to this, the state space must be reduced effectively by abstraction while preserving the characteristic features that enables assessing those abstracted states for applying change planning strategies.

The simulator to be used must be able to process big models and offer a component to abstract the models according to the real-time constraints in real-time event handling.

The following chapter 3 summarizes the state of the art and shows the gap, which is to be filled by the concepts of chapter 4. Chapter 5 summarizes the work already done and discusses in brief future activities.

3 State of the Art

Basis of every reaction within the event management is the identification and classification of the raised events, which are to be detected by monitoring mechanisms. In the following section, a possible event classification is presented, which can be enlarged for the described project of simulation-based process coverage for an effective change planning. This section is followed by a small extract from the state of the art within the area of application of material flow simulation method as a forecasting module in the operational level of a production system. The state-of the art is closed by a view on artificial intelligence methods used for change planning of production processes.

3.1 Event classification (changed location)

As a basis for event classification the supply network is modeled using MFERT (MFERT is described in detail in [7]). Based on this model an event classification scheme for supply network event management was defined as described in [4], based on the inter-company event classification scheme developed in [2].

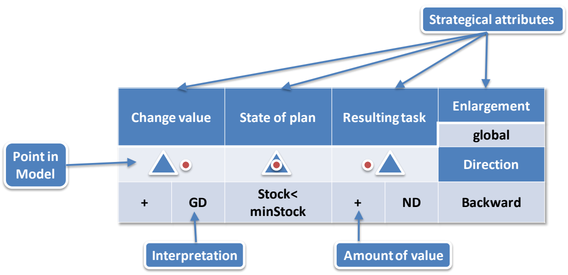


Fig. 2: Event classification scheme

An example for such an event classification is shown on the left hand side of Fig. 2. Each event can be mapped to a change in value (either of positive or negative nature) at a certain point in the model. The point in the model can be either before, behind or at a node (two kinds of nodes exist in MFERT, capacity nodes and object nodes).

An informal collection of events with industry representatives has been performed and these events have been mapped successfully to our event classification scheme. Thereby the relevance to practice has been demonstrated. The concepts described in this paper are based on the systematization of this event classification scheme.

3.2 Reinforcement Learning in Production Network Change Planning

Reinforcement learning has proven to be very successful in different practical domains. The most famous example is TD-Gammon [8], which has become the world's best backgammon computer program and is able to play as good as the best human players. Other famous and successful practical RL systems are the elevator dispatching system [9] by Crites and Barto and the job shop scheduling system by Zhang and Dietterich [10]. Stockheim et. al [11] present an approach for learning in supply chain management based on a very simple job acceptance decision. Riedmiller and Riedmiller are introducing a learning system for learning local dispatching policies [12], optimized for job scheduling on simple production systems. An approach for job control has been introduced by Stegheer [13]. The method is developed for planning and control of local production systems and does not use any state abstraction. Dangelmaier et. al. [14] present a general approach for learning rules for controlling production networks but do not cover state abstraction and efficient training.

In order to use state abstraction to explore the state space, several methods have been developed and applied. For example Sutton and Barto as well as Watkins [15],[16] developed a method called CMAC that partitions the state space into feature spaces. Also neuronal networks [17] have been used to approximate the policy value function. Another method for reducing state space complexity is state aggregation. State aggregation is generally based on distance function that assigns a state to a number of "similar" states while preserving the general features of a state and doing abstraction of the feature values. Mahadevan and Connell [18] aggregate the state space using the binary Hamming-Distance for robot control learning-system. Also several algorithms like *G-Tree* [19] or U-Tree [20] exist for state aggregation using successive aggregation of state space. These algorithms create a decision tree during the learning process that represents the Q-function in a compact way. Sing et al. [21] developed a more general concept (ASA) for state aggregation based on probabilities for state occurrence to be similar to a state cluster. Bertsekas and Tsitsikalis [22] have proofed that reinforcement learning will converge in an aggregated state space. The proof is based on the assumption that all usable actions have the same effect whether used on states or aggregated states and following the convergence could be transferred to learning on aggregated states. Döring et. al. [23] introduced an approach for state aggregation in supply networks using k-means based on a distance function representing the characteristic features of a supply network state. This approach will be used in this paper.

3.3 Simulation-Based Forecasting

Software tools like UGS Plant Simulation, Delmia's Quest or Taylor ED by Enterprise Dynamics [24],[25] are used for material flow simulation regularly. With these tools it is possible to create, to validate, to verify and to compute models of the focused production process. Although mostly traditional areas of application such as process planning and dimensioning are regarded today, there are existing prototypes, which use the method 'material flow simulation' as a forecasting module within the production planning and control processes in the daily business [26] in order to enrich their production control. Based on a model of the actual production process and the actual production state as well as the intended production plan as an input data set for a simulation run, this allows the detection of upcoming disruptions within the production process as early as possible, in the ideal case before its incidence and, by that, gives the affected company the possibility to avoid these expectable events in the production process or react to them.

As a major deficit of this approach, it has to be pointed out, that the possibilities of simulation are not used entirely, because simulation gives no information about the optimal reaction on this forecasted event. This is due to the reasonable selection of possible reaction strategies, which depends on type and consequence of the foreseen event. By using a given event classification (cp. section 3.1), a pre-selection of reaction scenarios can be achieved by methods of artificial intelligence in order to simulate these scenarios for a simulation based benchmark.

4 Concept and Implementation

The approach establishes two time horizons for the methods to be developed:

- In real-time event management only the current state of the supply network is examined.
- In foresighted event management a time horizon of a certain time interval depending on the size of the network simulation model, e.g. one week, is examined. All methods begin with the current state of the system.

The architecture of the system to be developed is shown in Fig. 3:

- On the data layer, standing and dynamic data is regarded. This information is condensed from the company's ERP system (e.g. SAP) and additional data stored by the special production planning and control system OOPUS-Web designed to integrate the methods described in this paper.
- The user interfaces are designed to fit the requirements of effective control and the ability of fast reactions to occurring events.
- The event management is divided into modules by the techniques used: simulation, optimization and artificial intelligence. These modules are closely integrated by process definitions.

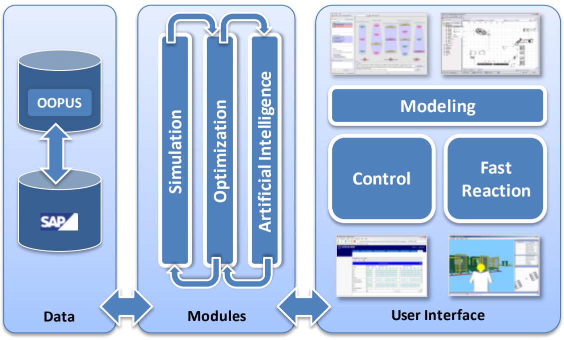


Fig. 3 System architecture

In detail the following components implement the event management:

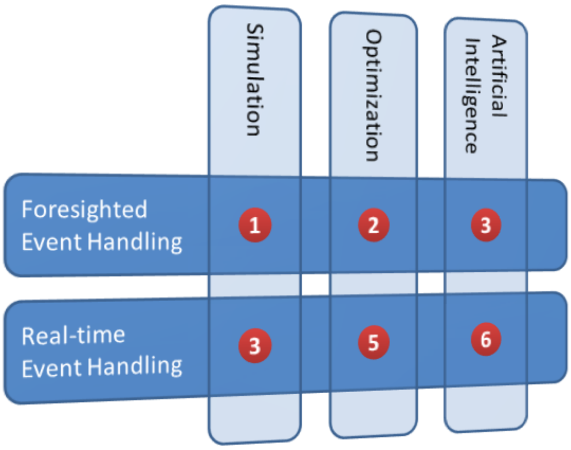


Fig. 4 System matrix of research topics

(1) Foresighted event detection, situated in the simulation module. Bottlenecks and probably occurring events are detected and resolved as necessary.

(2) Foresighted event handling, situated in the optimization module. Standard planning algorithms are used to react to predicted events above a pre-defined probability threshold t_A.

(3) Foresighted definition of rule sets for event reaction, situated in the artificial intelligence module. Events below the pre-defined probability threshold t_A but above the threshold t_B are regarded here. Reaction measures fitting these events are developed in advance.

(4) Real time event detection, situated in the simulation module. A continuous control of the system state and comparison to the plan detects events. A simulation determines the relevance of these events and the effected partners.

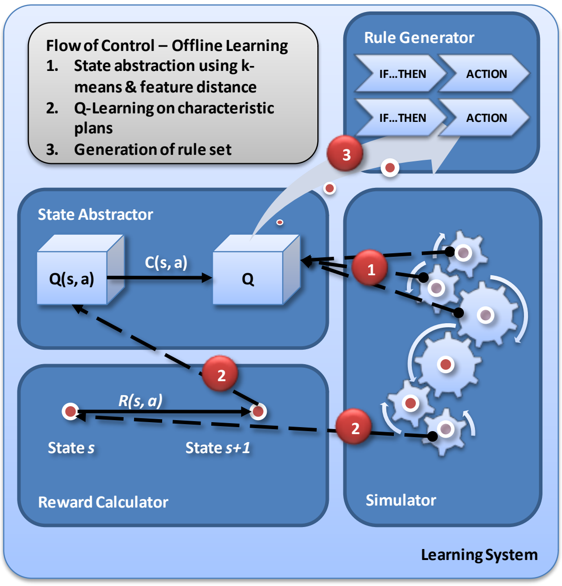
(5) Real-time optimization using optimization methods.[1]

(6) Real time event handling, situated in the artificial intelligence module. Either a pre-defined reaction measure is selected (if existent) or a new one is created using methods of artificial intelligence.

The realization of these components is described in 4.1 (artificial intelligence module) and 4.2 (simulation module) in detail.

4.1 State aggregation and rule learning for supporting simulation scenario selection

The basis for learning rules in production networks is an aggregated state space to ensure the convergence of the learning algorithm. For this, a feature based state abstractor will be introduced. The reinforcement learning system based on Q-Learning [6],[15] will operate on the aggregated states called clusters. The learning task itself will be done offline based on a given product network representation. Fig. 4 shows the basic architecture of the learning module.



.

Fig. 5 Learning system architecture

4.1.1 State abstractor

The clustering is needed to accelerate the learning process based on which will now work on a

[1] Not stressed in this paper.

characteristic state space of the production network with less states to observe. Every production network and its plans for each partner is characterized by its interconnections, associated policies and procurement strategy and their local production system restrictions like order penetration point. Based on this framework, a specific interaction of the partners and thereby specific characteristics in the individual production plans will emerge. Based on this assumption the state space can be reduced to its characteristic states.

For training the clustering algorithm and the simulator generates the specific states based on real data or generated data gained by representative production plans generated by a specific algorithm using the above mentioned frame conditions. The Centroids of the clusters are defined as *characteristic production plans* according to the specific features of the states the production network could reach normalized over the inventory of the periods.

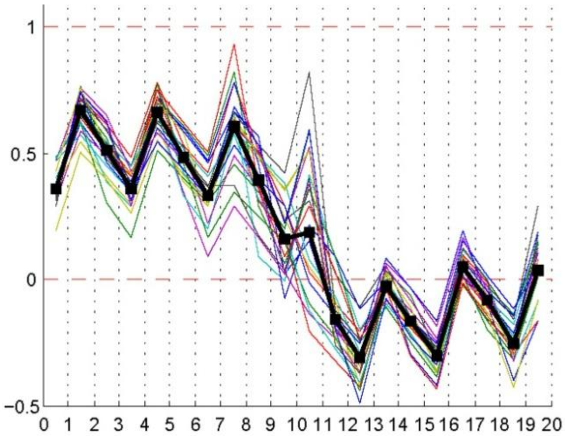


Fig. 6 Example cluster with normalized characteristic plan over 20 planning periods

The characteristics are gained from the specific restriction violations that occur after an event, their distance in the plan and the proportion of inventory in the periods. The full concept and results are presented by Döring et. al. in [23].

4.1.2 Learning and rule generation

The whole learning process is done by a Q-Learner. For this, a reward function has been developed that calculates rewards based on the weighted sum of restriction violation quantities in a normalized interval. The reward will be calculated based on states (1) while the Q-value will be (2) associated to the cluster that is associated with state *t*.

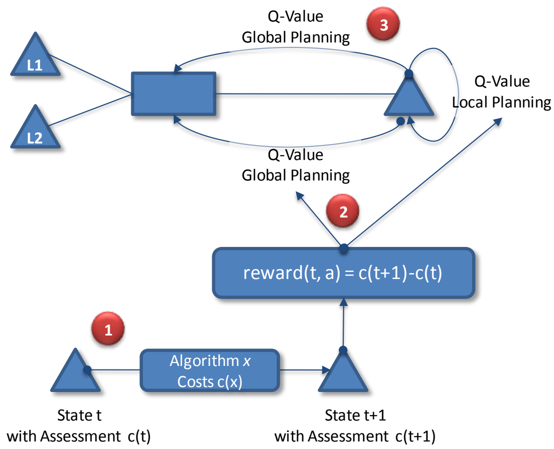


Fig. 7 Illustration of the learning system principles

As fig. 6 shows, the main learning task is to learn the decision (3) whether to execute a global or a local planning strategy. Secondly it has to determine whether during global or local planning a specific change planning action *a* is the best algorithm for this characteristic production network state could be expressed as

$$\max(r(t, a_i)) \text{ for all } a_i \xrightarrow{assoziated} C_t .$$

This quantified sorting by Q-values mapped to those actions a_t that are executable in a plan state of cluster C_t is the numerical representation of the rules that are needed to select the best actions in the specific characteristic scenarios.

For integration of the learned rules into the event handling system see fig. 6. After an event occurred, the analyzer maps the system scenario to a cluster. The best *x* change planning algorithms are selected based on the rule set. Then the change planning strategy will be executed on the event scenario bringing *x* new plans with cleared events. For process coverage, the simulation will simulate the clearing scenarios according to the actual network status and the scenarios to take out specific effects that could happen in the operative production execution using the specific plan.

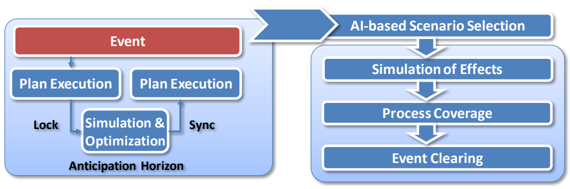


Fig. 8 Integrating learning, rules, scenarios and simulation in real-time event handling

4.2 Simulation

During the real-time event handling process the simulation time will be fixed by a certain anticipation horizon which defines the time period from *now* until the event will probably take direct effect to the production system or systems in the supply network.

The simulation must be based on a rough detail level optimized to processing speed using the best promising scenario based on the sorted rule set for this state. Therefore, a method for a dynamic adjustment of the simulation models level of detail during or before the start of the simulation run have been worked out [26]. Core insight of this solution is the decrease in computation time, if the level of detail of a simulation model is reduced.

Three technologies form the basis for dynamically detailed simulation. First, a *hierarchy* is needed, which assigns all existing building blocks to a certain level in the simulation model, and defines which building block represents others on a higher level of detail, and by whom it is represented on a less detailed level of degree. In each simulation step, a special configuration within this hierarchy is active, and leads to an executable simulation model, which can cover different levels of detail on the different areas of a simulation model.

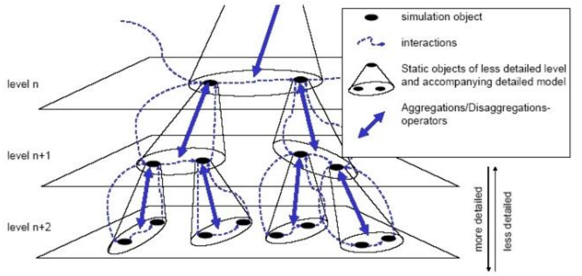


Fig. 9 Construction of a dynamic detailing simulation model with its cross-hierarchical links

Secondly, *indicators* decide which parts of the simulation model are to be executed on a higher or lower level of detail. There are, e.g., indicators existing, which decide by different geometrical aspects, for example the position and the user's line of sight within the immersive, virtual environment. Dependant on the user's avatar position in the virtual scene, the level of detail is adopted within the simulation kernel. Moreover, there are indicators, which are based on the logical connections between the building blocks. Especially for real-time event handling by a simulation model, the use of these indicators are strongly recommended, to be able to define those areas in the simulation model, which are affected by the identified event.

If deactivated model elements are reactivated by the decision of an indicator, they don't have a regular state for this simulation time, as it exists in the building block, activated directly before. In order to keep *consistency* within the simulation, thirdly, the current states for the activating building blocks are to be computed again. Within the developed method, several techniques have been implemented, starting from a recalculation of this building block up to a generated mapping by special event functions.

Based on the actual production state as well as the pre-selected scenarios, the simulation has to be calculated and evaluated according to defined indices, so that the generated result of the scenario simulation can be compared on an objective level and, by that, the ideal reaction strategy can be selected automatically. As a technical basis for the implementation, the material flow simulator d³FACT insight [27] is used and enhanced in those interface parts, which deal with the analysis connecting to the additional modules of the production control umbrella. Fig. 9 shows the modular structure of d³FACT insight:

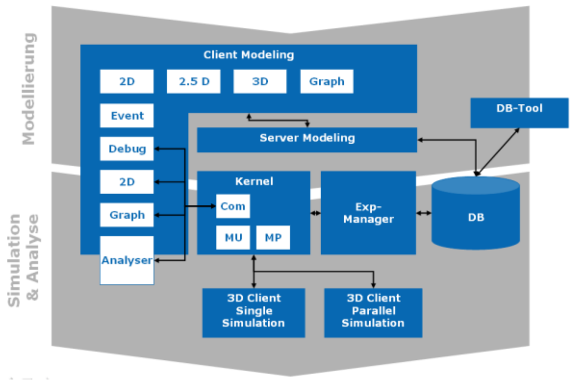


Fig. 10 Modules of d³FACT insight

During foresighted event handling there will be much more time for simulation. In this case the simulation model can be simulated on a higher level of detail. By the use of distributed simulation methods, the generated forecasting simulation results can be enlarged and validated on a statistic level, if not a single simulation run is taken into account but a set of simulation runs in a simulation experiment.

By augmenting the simulation from the real-time reaction benchmark to the forecasting level, the process coverage effects can also be shifted from the "emergency" reaction operational planning level to a tactical process coverage form a future time period, and thereby accretes the time horizon for a reaction and selection of the ideal re-planning scenario.

5 Conclusion

This paper introduces a specific concept for an integrated control center using simulation and artificial intelligence methods, powering force feedback loops for an automated event handling of production processes. Detailed problems, that arise during the realization of these concepts are solved for themselves, but yet not integrated:

- A solution for learning a rule set for optimizing change planning by selecting scenario based change planning algorithms on effective and efficient abstraction and Q-learning have been introduced.
- A simulator, suitable for material flow simulation is available.

- A matrix to classify and structure the related research topics has been developed.

Future work will combine all components to one force feedback loop. For this main research field, the dynamic scaling of simulation models as well as the optimization of the actual status of the learning system is essential. Actually, all processes are not integrated into a "real" system. Also learning is actually an offline process and will be improved by pushing an online learning strategy, which adapts to network configuration changes automatically.

6 References

[1] Mercer Management Consultants; Fraunhofer Gesellschaft. (2004). Future Automotive Industry Structure (FAST) 2015.

[2] J. Heidenreich (2006). Adaptive Mengen- und Kapazitätsplanung in kollaborativen Produktionsnetzwerken der Serienfertigung. Dissertation Universität Paderborn: HNI Verlagsschriftenreihe.

[3] C. Laroque (2007). Ein mehrbenutzerfähiges Werkzeug zur Modellierung und richtungsoffenen Simulation von wahlweise objekt- und funktionsorientiert gegliederten Fertigungssystemen. Dissertation, Universität Paderborn.

[4] W. Dangelmaier, A. Döring, T. Timm & B. Klöpper (2006). Collaborative Event Networking (CEN) - Automatisierte Absicherung der operativen Materialflussplanung in Supply Networks der Automobilindustrie. In 12. Magdeburger Logistiktagung: Sicherung von Prozessketten (S. 182-193). Magdeburg.

[5] W. Dangelmaier, M. Aufenanger; D. Huber, C. Laroque (2007): Ideen zur Entwicklung der Ablaufsimulation. In: Simulation und Visualisierung 2007, S. 3-14, 8. - 9. Mrz. 2007 SCS – Europe.

[6] T. M. Mitchell (1997) Machine Learning. Mac Graw-Hill Book Co, New York.

[7] U. Schneider (1996). Ein formales Modell und eine Klassifikation für die Fertigungssteuerung. Dissertation Paderborn: HNI-Verlagsschriftenreihe.

[8] G. Tesauro (1995). Temporal Difference Learning and TD-Gammon. Communications of the ACM 38(3), pp. 58-68.

[9] R. H. Crites, A. G, Barto (1996). Improving Elevator Performance Using Reinforcement Learning. In: Touretzky, D. S.; Moser: M. C.; Hasselmo, M. E.: Advances in Neural Information Processing Systems 8. Cambridge, Mass., MIT Press.

[10] W. Zhang, T. G. Dietterich (1996). A Reinforcement Learning Approach to Job Shop Scheduling. Technical Report, Department of Computer Science. Oregon State University.

[11] T. Stockheim, M. Schwind, & W. A. Koening, (2003). Reinforcement Learning Approach for Supply Chain Management, 1st European Workshop on Multi-Agent Systems Proceedings.

[12] S. Riedmiller, M. A. Riedmiller (1999). A neural reinforcement learning approach to learn local dispatching policies in production scheduling, Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI´99), p. 764-771.

[13] T. Stegherr (2000). Reinforcement-Learning zur dispositiven Auftragssteuerung in der Variantenreihenproduktion, Herbert Utz Verlag.

[14] W. Dangelmaier, A. Döring, T. Rust, & B. Klöpper (2006). A reinforcement learning approach for learning coordination rules in Production Networks. In International Conference on Intelligent Agents, Web Technologies. IEEE Computer Society Press.

[15] R. S. Sutton, A. G. Barto.(1998). Reinforcement Learning: An Introduction, MIT Press, 1998

[16] C. J. Watkins (1989). Learning from Delayed Rewards, University of Cambridge.

[17] S. Russell, P. Norvig, (2003). Artificial Intelligence: A Modern Approach Planning Prentice Hall.

[18] S. Mahadevan, J. Connell. (1992). Automatic programming of behavior-based robots using reinforcement learning Artificial Intelligence, Elsevier Science Publishers Ltd., 311-365.

[19] D. Chapman, L. P. Kaelbling (1991). Input generalization in delayed reinforcement learning: an algorithm and performance comparisons, Proceedings of the 1991 International Joint Conference on Artificial Intelligence, 726-731.

[20] W. T. B. Uther, M. M. Veloso (1998). Tree based discretization for continuous state space reinforcement learning, IAAI '98: Proceedings of the tenth conference on Innovative Applications of Artificial Intelligence, American Association for Artificial Intelligence, 769-774.

[21] S. P. Singh, T. Jaakkola & M. Jordan (1995). In: G. Tesauro, D. Touretzky, & T. Leen. (ed.): Reinforcement Learning with Soft State Aggregation, Advances in Neural Information Processing Systems, The MIT Press, 7, 361-368.

[22] D. P. Bertsekas, & J. N. Tsitsiklis.(1996). Neuro-Dynamic Programming, Athena Scientific.

[23] A. Döring, W. Dangelmaier & C. Danne (2007). Using k-means for clustering in complex

automotive production systems to support a Q-learning-system, ICCI 2007 - The 6th IEEE International Conference on Cognitive Informatics, Accepted for publishing, August 2007.

[24] A. M. Law, W. D. Kelton (2000). Simulation Modeling and Analysis; Boston; Third Edition; McGRAW-Hill International Series.

[25] VDI-Richtlinie 4499 (2007): Digitale Fabrik - Grundlagen; Stand, Januar 2007; VDI Verlag Düsseldorf.

[26] B. Mueck (2005). Eine Methode zur benutzerstimulierten detaillierungsvarianten Berechnung von diskreten Simulationen von Materialflüssen, HNI-Verlagsschriftenreihe, Paderborn.

[27] W. Dangelmaier, D. Huber, C. Laroque, & B. Mueck (2005). d³FACT insight - An immersive material flow simulator with multi-user support. In: Bruzzone, Agostino; Williams, Edward (Eds..) Proceedings of the 2005 Summer Computer Simulation Conference, Simulation Series, Band 37, S. 239-242.