SMART CONTROL PARAMETER SETTING FOR THE SIMULATION OF ELECTRIC DRIVE SYSTEMS

Hansjörg Kapeller[1], Harald Giuliani[1], Franz Pirker[1]

[1]Arsenal Research, Österreichisches Forschungs- und Prüfzentrum Arsenal Ges.m.b.H., 1210 Vienna, Giefinggasse 2, Austria

phone +43-50550-6606, fax +43-50550-6595, e-mail: hansjoerg.kapeller@arsenal.ac.at

Abstract

Trend analyses show an increasing demand for electrified vehicle components due to economic and technical requirements for tomorrow's vehicles. The challenge for developing innovative electrical vehicle components is the successful exploitation of new technologies and drive concepts, the consideration of the total system in the development process as well as a high degree of fail safety. For shortening the period of development and reducing costs, simulation is a crucial step in the continuous design process.

Special software tools are necessary for this development process because the conventional simulation and calculation programs do not meet interdisciplinary and dynamic demands. Dynamically applicable and real-time capable models are required to simulate real operating conditions in a vehicle and to design the components according to real requirements.

In this contribution a simulation library, the SmartElectricDrives (SED) library [1] developed by arsenal research with focus on automotive applications will be presented. This simulation tool provides the easy and robust controllability of electric machines in order to meet the ever increasing demand regarding efficiency, flexibility and accuracy in a wide range of applications. Moreover, the implemented controller parameter estimation functions for easy controller handling will be presented.

Keywords: Electrified vehicle components, Simulation of electric drive systems, Shortening the period of development, High machine performance, High control precision.

Presenting Author's biography

Hansjörg Kapeller was born in 1978 in Italy. He received the Dipl.-Ing. degree in electrical engineering from Vienna University of Technology, Vienna, Austria, in 2004. Since October 2004, he has been an engineer of the business unit Monitoring, Energy and Drive Technologies with Arsenal Research, Vienna, Austria. His main research interests are monitoring of faulty induction machines, simulation and control of electric drives.

1　Introduction

In electric as well as hybrid electric vehicles, the electric traction machine has to be controlled. The main purpose of electric machine control is to achieve a desired and well defined machine performance. Drives such as traction drives, electrical auxiliary drives in conventional vehicles and drives in hybrid electric vehicles are good examples for applications. Other typical high performance drive applications are industrial drives such as machine-tools and robotics. These applications require high static control precision, fast dynamic response, overload capability and a vast operating speed range. For motor operations, the control system is designed to control the torque, speed or position. For generator operations, e.g. in regenerative braking mode, the control system is designed to control the torque.

1.1　The controlled system

The basic topology of an generalized electric drive is schematically illustrated in Fig. 1. It consists of the electric machine, the power converter, the power supply, cascaded current and process controllers (speed, position), the mechanical load, the current sensor and the position transducer respectively [2]. The speed and position controllers determine the reference stator current of the machine following the desired reference quantities. If no speed and position controller is used, the drive is torque controlled and the reference current is based on the reference torque.
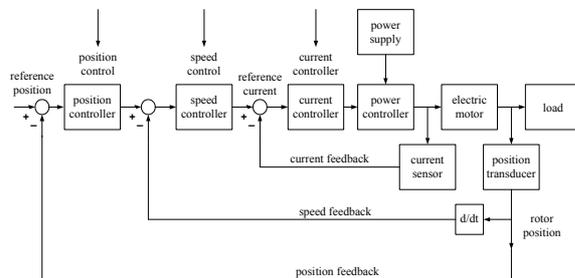
Fig. 1 General diagram of a drive control

However, the basic topology of an electric drive shows a typical application of a cascade controlled system. The main subject of this paper will be to show how each individual controller can be parameterized without a sophisticated analysis of the entire drive control structure.

The implementation occurs in three steps:

1) Parameterization of the controller in the inner closed loop using the subsequently described parameterization method.

2) Embodiment of the controlled inner closed loop to one unit. Then the outer controller can be parameterized as single-loop again.

3) Parameterization of the outer controller using the same parameterization method as in step one.

The controller design (P-, PI- or PID-controller) of each loop depends significantly on the controlled system structure. The structure of the controlled system can be determined either by mathematical description or by various identifying methods [3,4], if the controlled system is unknown. For the here presented examples the controlled system of each loop can be determined by mathematical description, because all electrical (machine) and mechanical parameters (load) are known.

It can be shown, that starting from the inner closed loop, every analysis of the feedback loop shows a first order delay system (PT1) ensured by the presented parameterization of the cascade controlled system.

1.1.1　First order controlled system

The structure of a first order controlled system looks like as follows,

$$G(s) = \frac{k_S}{(1 + T_1 s)}, \qquad (1)$$

–　k_S　gain of the controlled system,

–　T_1　time constant of the controlled system.

The step response of a typical first order controlled system is shown in Fig. 2 and illustrates the time behavior of a PT1 element.

1.1.2　Second order controlled system

The structure of the second order controlled system is:

$$G(s) = \frac{k_S}{(1 + T_1 s) \cdot (1 + T_2 s)}, \qquad (2)$$

–　k_S　gain of the controlled system,

–　T_1, T_2　time constants of the controlled system.

The step response of a typical second order controlled system is shown in Fig. 2, too and illustrates the time behavior of a PT2 element.
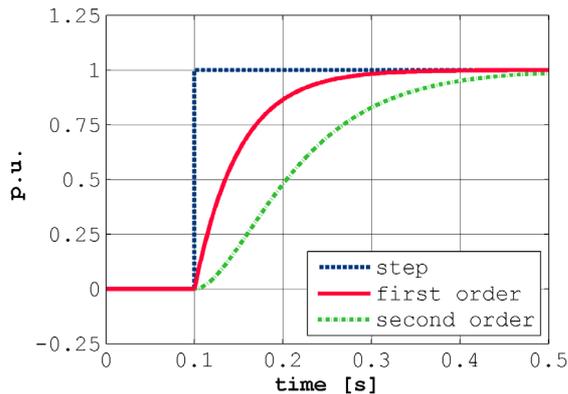
Fig. 2 Step response of a first order and a second order system

2   Parameterization of the controllers and analysis of the cascade controlled system

As mentioned above, the commensurate selection of the controller allows to achieve, that starting from the inner closed loop, every analysis of the closed-loop control system shows a first order delay system (PT1).

2.1 Compensation method

In this paper the compensation method will be used. The idea behind this method is to implement the controller as an inverse function of the already known (or identified) controlled system enhanced with an additional integral time (T_K).

2.1.1   Controller for a PT1 system

According to Eq. (1) the controller for a PT1 system looks like:

$$K(s) = \frac{1}{G(s)} \cdot \frac{1}{T_K s} = \frac{(1+T_1 s)}{k_S} \cdot \frac{1}{T_K s}. \quad (3)$$

This control structure is equivalent to a PI control structure,

$$K(s) = k_R \cdot \left(1 + \frac{1}{T_I s}\right) = k_R \cdot \frac{1 + T_I s}{T_I s}, \quad (4)$$

-    k_R gain of the controller,
-    T_I integral time.

The parameters can be found by comparing the coefficients,

$$T_I = T_1, \quad (5)$$

$$k_R = \frac{T_I}{k_S T_K} = \frac{T_1}{k_S T_K}. \quad (6)$$

Often the system to control shows a delay behavior higher than a first order system, e.g. a second order system (PT2 system). For a PT2 system the controller design (according the compensation method) does not result in a PI structure anymore.

2.1.2   Controller for a PT2 system

According to Eq. (2) the controller for a PT2 system looks like:

$$K(s) = \frac{1}{G(s)} \cdot \frac{1}{T_K s} = \frac{(1+T_1 s) \cdot (1+T_2 s)}{k_S} \cdot \frac{1}{T_K s}. \quad (7)$$

This control structure is equivalent to a PID control structure,

$$K(s) = k_R \cdot \left(1 + \frac{1}{T_I s} + T_D s\right), \quad (8)$$

-    k_R gain of the controller,
-    T_I integral time,
-    T_D derivative time.

The parameters can be found by comparing the coefficients,

$$T_I = T_1 + T_2, \quad (9)$$

$$T_D = \frac{T_1 T_2}{T_1 + T_2}, \quad (10)$$

$$k_R = \frac{T_I}{k_S T_K} = \frac{T_1 + T_2}{k_S T_K}. \quad (11)$$

2.2 Dynamic tune

When $T_K = T_I$ is assumed, from Eq. (5), Eq. (6) and respectively from Eq. (9), Eq. (11) the coherence $k_R = k_S^{-1}$ can be deduced. The transfer function of the closed loop is, however, a first order system. The value of k_R determined using the assumption $T_K = T_I$ can be used as an initial setting during the initial operation phase. It is possible to increment the gain of the controller to enhance the dynamic performance of the whole controlled system (e.g. five times the value of k_R could be a feasible setting). Further down the parameter *kdynVariablename* will represent a size factor, which allows the user to magnify or reduce the gain of the controller until the desired dynamic behavior is achieved.

Important comment: if the determined structure of the controlled system shows a behavior like an element without self-regulation, e.g. a salient integral action, then the parameterization of the controller via compensation method is dispensable. The choice of a P-controller in such case results in stable performance.

2.3 Current controller design

The control process $G'_{1,d(q)}(s)$ of a permanent magnet synchronous induction machine (SMPM) and the designed current controllers $K_{C,d(q)}(s)$ are shown in Fig. 3. Since a decoupling network is used, the structure of both parts (stator current d-component and stator current q-component) are equal. This allows identical current control design for both parts with high control performance. The decoupling components added to the output of the controller guarantee, that the designer can regard $G'_{1,d(q)}(s)$ as a first order lag element [5].

Therefore two PI-controller (for stator current d-component and stator current q-component) are used and parameterized via the compensation method. The user has the possibility to change the dynamics of the current controllers by vary the parameter *kdynCurrent*.
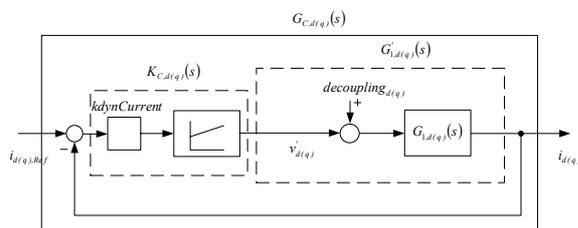
Fig. 3 Control process of the machine and designed current controllers

$$L_{d(q)} = L_{ssigma} + L_{md(q)}, \qquad (12)$$

$$G'_{1,d(q)}(s) = \frac{i_{d(q)}}{v'_{d(q)}} = \frac{k_S}{1 + T_1 s} = \frac{\dfrac{1}{R_S}}{1 + \dfrac{L_{d(q)}}{R_S} s}, \quad (13)$$

with

- $i_{d(q)}$ stator current d(q)-axis,
- $v'_{d(q)}$ regulated quantity of the decoupled stator voltage d(q)-axis,

and the known electrical machine parameters

- $L_{d(q)}$ total inductance d(q)-axis,
- L_{ssigma} stator stray inductance per phase,
- $L_{md(q)}$ main field inductance d(q)-axis,
- R_S warm stator resistance per phase.

The coherence that the gain of the controller should be the reciprocal of the gain of the controlled system (cp. section 2.2) delivers

$$k_{R,d(q)Current} = \frac{kdynCurrent}{k_S}, \qquad (14)$$

$$T_{I,d(q)Current} = T_1 = \frac{L_{d(q)}}{R_S}. \qquad (15)$$

2.4 Speed controller design

Figure 4 depicts the outer closed loop consisting of the speed controller $K(s)$ and the controlled system $G(s)$ including the embodiment of the inner closed loop of the q-component current controller $G_{C,q}(s)$ and the transfer block $G_m(s)$, which represents the mechanical behavior of the machine.
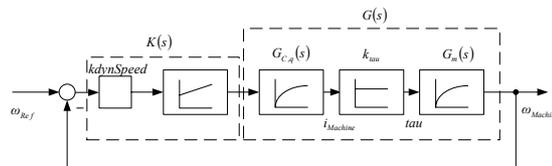
Fig. 4 Outer closed loop consisting of the speed controller and the controlled system

If the model of the mechanical behavior considers friction and mass inertia [6], the transfer block $G_m(s)$ looks like,

$$G_m(s) = \frac{1}{\mu_{fr} + T_m \cdot s}, \qquad (16)$$

with

- μ_{fr} friction coefficient of the mechanical system,
- T_m mechanical time constant.

The mechanical time constant depends on the mass inertia, the torque provided by the motor, the desired speed range and can be calculated as follows,

$$T_m = J \cdot \frac{\omega_{Max}}{tau_{Max}}, \qquad (17)$$

with

- J total mass moment of inertia relating to the shaft of the SMPM,
- ω_{Max} maximum admissible angular velocity of the SMPM,
- tau_{Max} maximum admissible torque of the SMPM.

Then the transfer function of the controlled system $G(s)$ looks like

$$G(s) = G_{C,q}(s) \cdot k_{tau} \cdot G_m(s), \qquad (18)$$

with

- k_{tau} torque constant of the SMPM.

Thus the system appears as a second order system (PT2 element) and the compensation method yields a PID-controller. In many applications the inner closed

loop of the machine current control shows a much faster dynamic behavior compared to the mechanical transfer block with the mechanical time constant T_m. If only the time constant T_m is regarded, the controlled system simplifies to $G(s) = k_{tau} \cdot G_m(s)$, a PT1 system. If in Eq. (16) the friction coefficient is neglected as well ($\mu_{fr} = 0$), the controlled system $G(s)$ turn into an integral gain. Then the speed controller can be designed as a simple P-controller, where the parameter *kdynSpeed* changes the gain of the speed controller and, respectively, the dynamics of the closed speed control loop.

3   Simulation in Dymola

The SED library was developed using the software platform Dymola with the object oriented programming language Modelica. This language allows the software designer to create models of any kind of objects that can be described by algebraic and ordinary differential equations [7]. Dymola is an environment for modelling and simulating integrated and complex systems using the objects described in Modelica syntax. With Dymola models of electrical (resistors, inductances, electrical sources) and mechanical (dampers, inertias, springs) components simulations of the behavior and interaction betweens systems of different engineering fields, such as mechanical, electric, thermodynamic, hydraulic, pneumatic, thermal and control systems are possible. The modeling language itself is open which means that users are free to create their own model libraries or modify standard libraries to better match the users individual modelling and simulation needs. A library containing the basic models of electric machines such as direct current, asynchronous induction or permanent magnet synchronous induction machines already exists in the Modelica Standard library, the Machines library [8]. However, for many drive applications a drive control is essential to fulfill the required specification and machine behavior. Without suitable control blocks the machines in the standard library cannot be utilized in an easy way. Based on the Machines library the SED library facilitates simulations of any electric drive application using different control structures and strategies.

4   The SmartElectricDrives library

The SED library contains models for all important components contained in these state-of-the-art electric drive systems. Starting with sources (batteries and a PEM fuel cell), converters (ideal and power balanced), loads, process controllers, sensors, etc. are provided. The machines package, based on the Machines library of the Modelica Standard library, is split up into two categories, the quasi stationary models and the transient models. Besides all elementary components that give the user the freedom of building up an entire controlled machine on himself, 'ready to use' models

of drives for a convenient and quick composition of simulations are provided [1,9].

Elementary components, drive components as well as the 'ready to use' models are available and can be placed in the simulation just by drag and drop. These models contain converters and a measurement device which are essential for the implemented field oriented control (FOC).

4.1   Example of speed control for automotive application

The example of an electrically operated water pump in an automotive application with a speed controlled permanent magnet synchronous induction machine (SMPM) and an ideal integrated power converter is presented here.
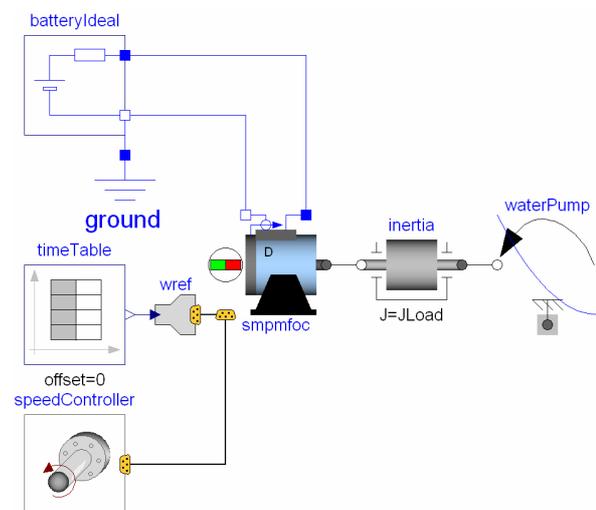
Fig. 5 Field oriented controlled permanent magnet synchronous induction machine with speed controller and load characteristic (water pump)

Figure 5 shows this typical automotive application simulated with the SED library. The 'ready to use' model of the torque controlled permanent magnet synchronous induction machine with field oriented control (smpmfoc) is supplied by an ideal battery where the state of charge remains constant (the battery is not discharged). The reference speed of the drive is defined by a time table and controlled by the speed controller. A load which rises quadratically with speed represents a typical characteristic of the water pump connected as load. The inertia describes that of the water pump.

The rated operating conditions are given by:

- $\omega_{Nominal} = 408.4$ rad/s (nominal angular velocity),

- $tau_{Nominal} = 0.4907$ Nm (nominal torque, equals tau_{Max} here).

The permanent magnet synchronous machine has the following nominal values:

- $V_{Nominal} = 7$ V (nominal phase voltage),

- $I_{Nominal}$ = 12.7 A (nominal phase current),

- $f_{Nominal}$ = 130 Hz (nominal frequency).

If these boundary conditions are completed with the electrical machine parameters (essential for the current controller), the mechanical parameters of the machine and the load (essential for the speed controller), then the cascade control system (anticipated in Fig. 1) can be parameterized regarding the presented method.

4.1.1   Parameter estimator for SMPM

This parameter estimator is implemented as a function which allows the estimation of the control parameters for a drive application with a permanent magnet synchronous induction machine. Furthermore the function provides the calculation of the nominal torque, the nominal rotor speed (rad/sec) and the nominal stator voltage of the permanent magnet synchronous induction machine. For the parameter estimation the presented control design method is implemented.

The optional flux weakening (in order to extend the speed range) follows common strategies of flux weakening as published in [10]. The parameterization of the flux weakening controller was done by the compensation method, again.

The control parameters have been calculated with the adjustable dynamic parameters (cp. Section 2.2),

- *kdynCurrent*=1 (dynamic gain of the current controller),

- *kdynFluxWeak*=0.1 (dynamic gain of the flux weakening controller),

- *kdynSpeed*=5 (dynamic gain of the speed controller − the friction coefficient is neglected and only the mechanical time constant T_m is regarded),

when executing the function:
'parameterEstimationSMPMControllersFOC'.
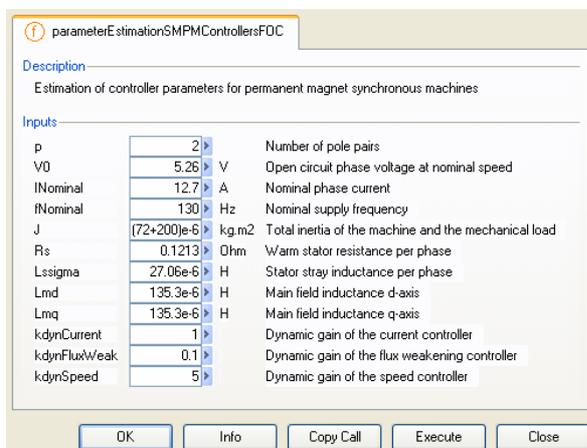
Fig. 6 Parameter estimator for SMPM (call function)

In Fig. 6 a screen shot of the completed parameter estimator input layout is depicted. This input layout appears, when calling the function (right mouse click).

Furthermore in the SED library similar parameter estimators for asynchronous induction machines (AIMC), permanent magnet DC machines (DCPM) and electrical excited DC machines (DCEE) are available.

The parameter estimator returns the specific controller parameter values listed in Tab. 1. With these values the example of the speed controlled electrically operated water pump can be simulated.

Tab. 1 Returned parameter values of the controllers

Controllers		
Parameters of the current controller		
$k_{R,dCurrent}$	0.1213	Gain of the stator current d-component
$T_{I,dCurrent}$	0.0013	Integral time constant of the stator current d-component [s]
$k_{R,qCurrent}$	0.1213	Gain of the stator current q-component
$T_{I,qCurrent}$	0.0013	Integral time constant of the stator current q-component [s]
Parameters of the flux weakening controller		
$k_{R,FluxWeak}$	1.5081	Gain
$T_{I,FluxWeak}$	0.0013	Integral time constant [s]
Parameter of the speed controller		
$k_{R,Speed}$	5.0	Gain

Regarding the speed controller the friction coefficient is neglected and only the mechanical time constant T_m=0.226 s estimated by Eq. (17) is considered: the condition $T_m >> T_{I,qCurrent}$= 0.0013 s is fulfilled and the speed controller acts as a P-controller.

4.2 Simulation Results

The simulation results are shown in Fig. 7 and Fig. 8. As depicted in Fig. 7, the real speed of the pump follows the reference speed at the beginning of the simulation. Beginning at 0.5 seconds the reference speed rises too quickly. The maximum admissible torque tau_{Max} of the SMPM is reached (Fig. 8) and therefore the acceleration of the inertia is limited.
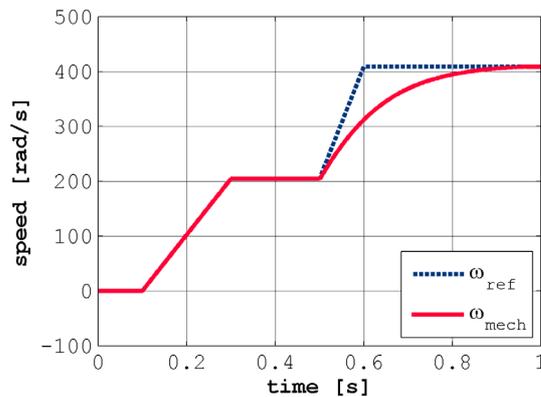
Fig. 7 Reference speed and real water pump speed
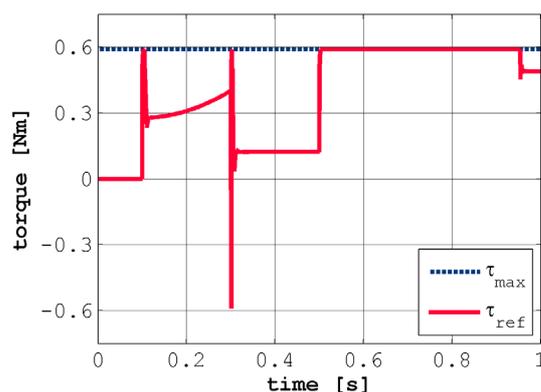
Fig. 8 Maximum admissible torque and reference torque of the electric machine

4.3 Example of position control for industrial application

The example of an industrial application where a position controlled milling feed driven by an asynchronous induction machine (AIMC), is presented here.
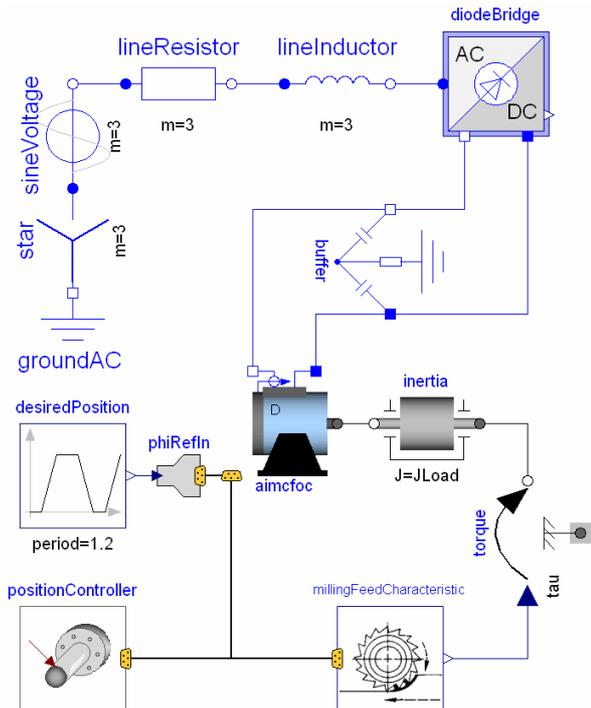
Fig. 9 Field oriented controlled asynchronous induction machine with position controller, secondary speed controller and load characteristic (milling feed)

Figure 9 shows this industrial application simulated with the SED library. The 'ready to use' model of the torque controlled asynchronous induction machine with field oriented control (aimcfoc) is supplied by a three-phase network. The reference position of the drive is defined by a trapezoidal curve shape and controlled by the position controller (2 revolutions in positive sense of rotation and back). The acting load rises as a coherent rectangular function representing the characteristic of the milling feed. The inertia describes that of the milling feed.

The rated operating conditions are given by:

– $\omega_{Nominal}$ = 154.2 rad/s (nominal angular velocity),

– $tau_{Nominal}$ = 130.9 Nm (nominal torque).

The asynchronous induction machine has the following nominal values:

– $V_{Nominal}$ = 400 V (nominal phase voltage),

– $I_{Nominal}$ = 18.93 A (nominal phase current),

– $f_{Nominal}$ = 50 Hz (nominal frequency).

If these boundary conditions are completed with the electrical machine parameters (essential for the current controller), the mechanical parameters of the machine and the load (essential for the speed controller), then the cascade control system can be parameterized again.

4.3.1 Parameter estimator for AIMC

This parameter estimator is implemented as a function which allows the estimation of the control parameters for a drive application with an asynchronous induction machine. Furthermore the function provides the calculation of the nominal torque and the nominal rotor speed (rad/sec) of the asynchronous induction machine. For the parameter estimation the presented control design method is implemented.

Compared to the SMPM the AIMC builds up its own magnetic field. Therefore an additional flux controller is needed [10] and designed via compensation method, too.

The optional flux weakening (in order to extend the speed range) follows common strategies of flux weakening as published in [10]. The parameterization of the flux weakening controller was done by the compensation method, again.

The control parameters have been calculated with the adjustable dynamic parameters (cp. Section 2.2),

- *kdynCurrent*=5 (dynamic gain of the current controller),

- *kdynFlux*=1 (dynamic gain of the flux controller),

- *kdynFluxWeak*=0.1 (dynamic gain of the flux weakening controller),

- *kdynSpeed*=134 (dynamic gain of the speed controller– the friction coefficient is neglected and only the mechanical time constant T_m is regarded),

when executing the function: 'parameterEstimationAIMCControllersFOC'.

Fig. 10 Parameter estimator for AIMC (call function)

In Fig. 10 a screen shot of the completed parameter estimator input layout is depicted. This input layout appears, when calling the function (right mouse click).

The parameter estimator returns the specific controller parameter values listed in Tab. 2.

Tab. 2 Returned parameter values of the controllers

Controllers		
Parameters of the current controller		
$k_{R,xCurrent}$	4.76	Gain of the stator current x-component
$T_{I,xCurrent}$	0.0086	Integral time constant of the stator current x-component [s]
$k_{R,yCurrent}$	4.76	Gain of the stator current y-component
$T_{I,yCurrent}$	0.0086	Integral time constant of the stator current y-component [s]
Parameters of the flux weakening controller		
$k_{R,FluxWeak}$	0.00065	Gain
$T_{I,FluxWeak}$	0.00688	Integral time constant [s]
Parameter of the speed controller		
$k_{R,Speed}$	134.0	Gain

Regarding the speed controller the friction coefficient is neglected and only the mechanical time constant T_m=0.46 s estimated by Eq. (17) is considered: the condition $T_m >> T_{I,yCurrent}$= 0.0086 s is fulfilled and the speed controller acts as a P-controller.

The parameterization of the position controller can be done as last step, during the initial operation phase of the entire cascade control drive.

4.3.1.1 Position controller

If the speed control of an electrical drive is extended with a position control loop as like in Fig.1, then the position controller has to be parameterized, too. The fact, that the position transducer act as an integral gain leads to the conclusion that the determined structure of the controlled system shows a behavior like an element without self-regulation. Therefore the choice of a P-controller delivers stable performance and sufficient phase reserve to avoid overshoot effects [6]. A feasible gain value for the P-controller can be found experimentally [6] and is listed here in Tab. 3.

Tab. 3 Parameter value of the position controller

Parameter of the position controller		
$k_{R, Position}$	47.0	Gain

The innermost closed loop represents the fastest controlled system, the next outer closed loop is faster then the outermost closed-loop controlled system.

With all parameterized controllers it is possible to get satisfactory simulation results for the position controlled milling feed.

4.4 Simulation Results

The simulation results are shown in Fig. 11, Fig. 12 and Fig. 13. The acting load rises as a coherent rectangular function representing the characteristic of the milling feed (Fig. 11). As depicted in Fig. 12, the real position of the shaft follows the reference position of the drive (defined by a trapezoidal curve shape – 2 revolutions in positive sense of rotation and back). Figure 13 shows the reference speed characteristic for the position controlled milling feed.
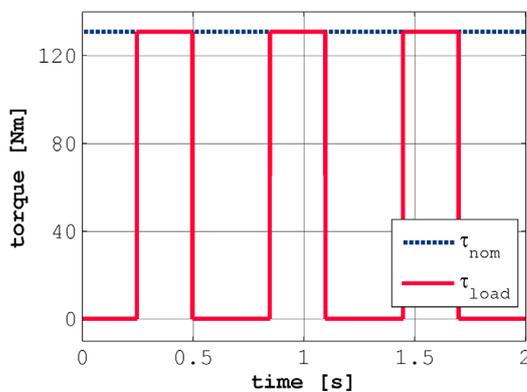
Fig. 11 Nominal torque of the electric machine and load torque (milling feed)
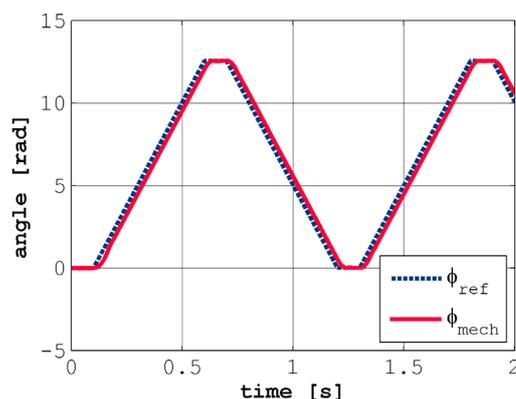
Fig. 12 Reference shaft position and real shaft position
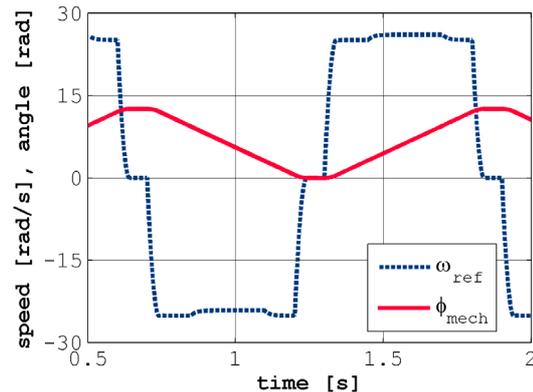
Fig. 13 Reference speed and real shaft position

5 Conclusion

In this paper the application of the compensation method in cascade controlled systems is investigated. This method allows the user to parameterize controllers without a sophisticated analysis of the entire drive control structure. Furthermore the SmartElectricDrives (SED) library was presented in which controller parameter estimation functions for easy controller handling are available. The simulation results of an automotive and an industrial application substantiate the claim of the presented smart control parameter setting for the simulation of electric drive systems.

6 References

[1] J.V. Gragger, D. Simic, C. Kral, H. Giuliani, V. Conte and F. Pirker. A Simulation Tool for Electric Auxiliary Drives in HEV's – The Smart ElectricDrives Library. *FISITA*, 2006.

[2] S. Nasar and I. Boldea. Electric Machines; Dynamics and Control. 2000 Corporate Blvd, USA: CRC Press, 1st ed., 1993

[3] A. Weinmann. Regelungen, Analyse und technischer Entwurf, Vol. 1. *Wien: Springer Verlag, 3rd ed.*, 1983.

[4] O. Föllinger. Regelungstechnik. *Heidelberg: Hüthig Verlag, 8th ed.*, 1994.

[5] M. Schrödl. Drehstromantriebe mit Mikrorechnern, Vienna University of Technology. *Wien: Lecture notes*, 2000.

[6] H. Lutz, W. Wendt. Taschenbuch der Regelungstechnik. *Frankfurt am Main: Wissenschaftlicher Verlag Harri Deutsch, 5th enlarged ed.*, 2003.

[7] H. Elmquist, S. Mattsson and M. Otter. Object-Oriented and Hybrid Modeling in Modelica. *Dynasim AG, DLR Oberpfaffenhof*, 2001.

[8] C. Kral and A. Haumer. Modelica Libraries for Dc Machines, Three Phase and Polyphase Machines. *Modelica Conference, 549-558*, 2005.

[9] H. Giuliani, C. Kral, J.V. Gragger and F. Pirker. Modelica Simulation of Electric Drives for Vehicular Applications. The Smart Electric Drives Library, *ASIM*, 2005.

[10] D. Schröder. Elektrische Antriebe- Regelungen von Antriebssystemen. *Berlin, Heidelberg, New York: Springer Verlag*, 2001.