The New EDrives Library:
A Modular Tool for Engineering of Electric Drives

Anton Haumer
Technical Consulting
3423 St.Andrä-Wördern, Austria
anton.haumer@edrives.eu

Christian Kral
Electric Machines, Drives and Systems
1060 Vienna, Austria
christian.kral@edrives.eu

www.edrives.eu

Abstract

Simulation is an indispensable tool for the engineering of systems containing electric drives. Depending on the design phase and the engineering task different levels of modeling details are required: proof of concept, investigation of energy and power consumption, design of control, etc. The new EDrives library provides three levels of abstraction for inverters: quasi static (neglecting electrical transients), averaging (neglecting switching effects) and switching – for serving different demands. The inverters can feed the machine models of the Modelica Standard Library: `Modelica.Magnetic.FundamentalWave` and the new `Modelica.Magnetic.QuasiStatic.FundamentalWave`. The EDrives library copes with arbitrary phase numbers and can be easily extended to develop new control algorithms. In this publication the structure of the library and the implemented control principles are presented. Furthermore, examples comparing the three different levels of abstraction are included.

Keywords: Electric machines and drives, power electronics, control, multi phase, thermal behavior, switching, quasi static, transient

1 Introduction

Engineering of systems containing electric drives is essentially supported by simulation, allowing verification of the overall concept with rapid prototyping against requirements even in early design phases as shown in [1]. Different engineering tasks require different level of modeling abstraction: An early proof of concept can be done without knowledge of details like the control or switching algorithm of the inverter. Determination of energy and power consumption during driving cycles requires the accurate consideration of losses and the interaction of the control with the whole system. Yet, in such cases the switching effects can often be neglected. For more sophisticated investigations of, e.g., the torque ripple, a more detailed model is needed taking switching effects into account. Therefore a modular concept supporting all tasks is indispensable. The new EDrives library provides inverter models with three levels of abstraction:

- Quasi static inverters connected to quasi static machine models; electrical transients are neglected, but losses are taken into account. The models have highest performance and robustness, making them suitable for long term simulations.

- Averaging inverters connected to transient machine models; the inverter models do not consider switching effects. Interaction of control with the whole system can be tested with high performance. Losses are taken into account, power conversion of the power electronics is based on power balance.

- Switching inverters provide the highest level of abstraction, resulting in lower simulation performance (i.e. longer execution time). On this level of abstraction switching effects and new pulse width modulation algorithms can be investigated.

The EDrives library supports arbitrary phase numbers m and is thus fully compatible with the `Modelica.Magnetic.FundamentalWave` and the new `Modelica.Magnetic.QuasiStatic.FundamentalWave` library. Yet the number of phases, m, being equal to 2^n with integer n are currently not supported. The reasons are discussed in [2] and [3]. Segregating the inverter and the machine models allows the modular exchange with more advanced models of machines, power electronics and control. For example, an electric machine model can be coupled with a thermal model and cooling circuit to study the thermal behavior of the machine and system over a whole driving cycle.

The parameters of the entire drive are compiled in a drive parameter record. The drive parameter record contains a record of the electric machine, the power electronics and the control settings. This modular record structure enables the user to handle and exchange parameter sets for different machines and drives in a very convenient way.

In the actual version of the EDrives library control algorithms are provided for permanent magnet (PM) synchronous machines, solely. Control algorithms for in-
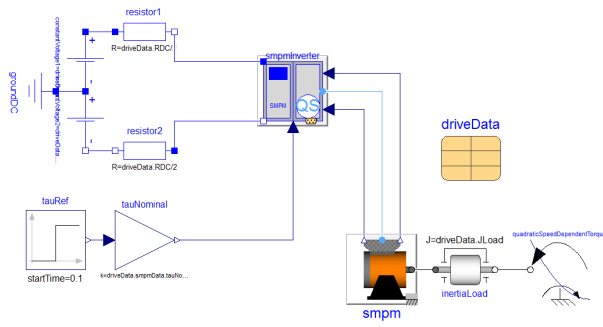
Figure 1: Permanent magnet synchronous machine with inverter

duction machines, electrical excited synchronous and synchronous reluctance machines as well as mains converters feeding the DC intermediate circuit will be implemented in upcoming versions. Therefore, all examples and models presented in this publication refer to permanent magnet synchronous machines.

2 Drive Concepts

Each **machine** model provided in the EDrives library consist of a machine model from the MSL including sensors for temperatures and the rotor flange angular position. The interface connectors are electrical, mechanical, thermal and the sensor outputs. These sensor outputs are connected with the inputs of the **inverter** model. The inverter consists of the power electronics, i.e., a **DC/AC converter**, sensors for electrical quantities and machine **control** and pulse width modulation (PWM) if required. The electrical AC output of the inverter is connected with a machine model and the DC input is connected with an external power supply. The type of DC power supply very much depends on the application. The DC power supply may either be a battery or fuel cell or an AC to DC converter supplied by the mains. An example of a PM synchronous machine drive modeled with the EDrives library is shown in Fig. 1. A detailed description of the main drive components is presented in sections 3–6.

3 Machines and Sensors

The EDrives library will provide wrapper models for each type of machine to encapsulate the machine model including sensors, as shown in Fig. 2 for a PM synchronous machine. These wrapper models contain:

- an instance of a quasi static or transient machine model of the MSL

- a terminal box, providing the desired terminal connection (star or polygon)

- a rotor angle sensor

- temperature sensors for making stator winding and rotor (PM) temperatures accessible
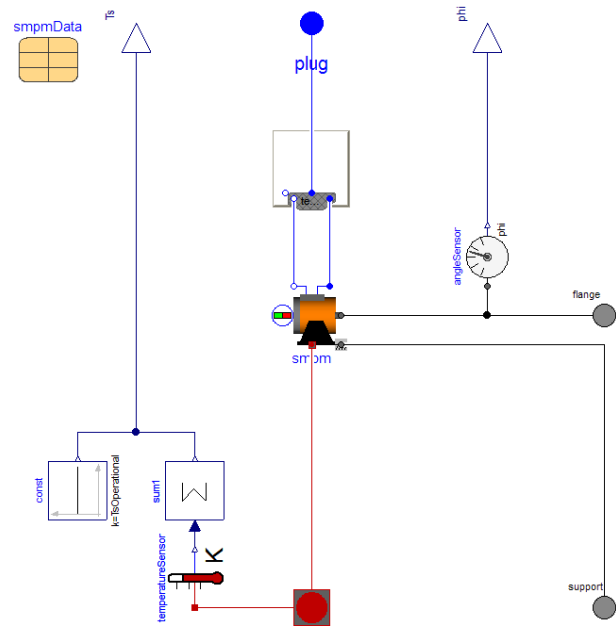

Figure 2: Wrapper model for the transient machine

- the machine parameter record

The machine parameter record allows to propagate all machine parameters with one record. This is achieved by the wrapper model propagating the elements of the parameter record to the machine model. The concept of the wrapper model enables the exchange of a quasi static by a transient machine model in an application and vice versa. The parameter records and all connectors except for the electrical connector are compatible in each wrapper model. This way only the electrical machine supply has to be re-connected. The mechanical flanges, the thermal connector as well as the temperature sensors and angle sensor outputs are the same for all wrapper models of permanent magnet synchronous machines.

Both the quasi static and the transient machine models take losses into account: temperature dependent copper loss, (eddy current) core loss, friction loss, stray load loss and PM loss – see [4, 5]. All these losses are dissipated either to an internal thermal ambient with fixed operational temperature, or exchanged with an external thermal model through the thermal port. An external thermal model allows to simulate the actual temperatures over a driving cycle as discussed, e.g., in [6, 7].

4 Inverters and Parametrization

Each inverter model extends from the same base class since they all have the same connectors except for the electrical plug that has to be connected to the machine. However, each inverter model uses the same drive parameter record. Thus, it is possible to replace one inverter model by another in an experiment with low effort. However, the level of abstraction of the machine model has to match level of abstraction of the inverter model: a quasi static inverter can feed a quasi
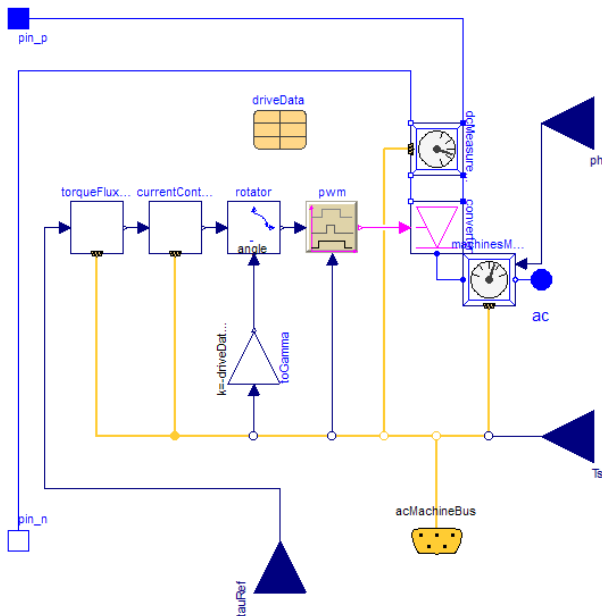
Figure 3: Switching inverter model

static machine; a transient machine can be supplied by an averaging or by a switching inverter.

The inverter model `converter` shown in Fig. 3 can be one of the following power converter models:

- Quasi static model: considers power balance between DC and quasi static AC side

- Averaging model: provides power balance between DC and transient AC side

- Switching model: the m phase converter bridge has two electronic switches and two free-wheeling diodes for each leg and phase, respectively

The sensing and signal conditioning of both DC and AC quantities are encapsulated in two models. The DC sensing model retrieves the actual DC current and voltage. Additionally, the filtered DC current, voltage and power are calculated. The filter settings are automatically adapted to the switching frequency of the application.

The AC sensing model acquires AC currents, voltages and power, and calculates the space phasor d and q components in the rotor fixed reference frame. These quantities get also filtered to suppress switching frequency related effects. All these quantities are fed to a signal bus, including the actual rotor position.

Machine control is also included in the inverter, as shown in Fig. 3 for a switching inverter model. All the signals obtained by sensors and signal conditioning are provided at the internal bus. Control strategies implemented in the EDrives library rely on field oriented control (FOC). This means that all electrical voltages, electrical currents and magnetic fluxes are decomposed into a component aligned with magnetic field of the PM (direct or d axis) and a component perpendicular to the magnetic field orientation (quadrature or q axis). Therefore, the sensed rotor flange

angular position is required to perform the space phasor decomposition. The d and q components represent the real and imaginary part of the respective space phasor in the rotor fixed reference frame. The actual orientation and magnitude of the field is obtained by a controller internal machine model. This model also provides the estimated electrical torque. In a real drive application torque and field magnitude and orientation are usually also modeled, not measured. Therefore, the quality of machine control is very much determined by the quality of the controller internal machine model.

For a machine with phase number greater than or equal to three, the electrical voltages, currents and magnetic fluxes are decomposed into space phasors, consisting of a d and q component, each. In the EDrives library the signals exchanged between the different controllers are vectors with two components, representing the d and q component of a space phasor. This space phasor may either be an electric voltage, current or magnetic flux.

Control is designed hierarchically. On each level of abstraction the control structure includes a torque-flux-controller, see section 6.2. The output of the torque-flux-controller are the commanded d and q currents, $\hat{I}_d$ and $\hat{I}_q$. The current of the q axis is proportional to the generated torque and the d axis current is used the adjust the actual flux level. These quantities are processed by a current controller. The outputs of the current controller are the components of the commanded voltage space phasor. This space phasor has to be transformed into phase voltages to be provided to the voltage generation of the inverter. Three different levels of inverter abstraction are provided in the EDrives library. The torque-flux-controller, however, is the same for all three levels.

- The quasi static inverter contains a quasi static current controller (section 6.3). The space phasor voltage components are transformed into quasi static phase voltages which are fed to the quasi static power converter.

- The averaging inverter uses a transient current controller (section 6.3) which is also used in the switching inverter model. A delay block is modeling the delay caused by the switching period of a real drive. The space phasor voltage components are transformed into transient phase voltages which are fed to the averaging power converter.

- The switching converter contains the same transient current controller as the averaging inverter. The space phasor voltage components represent the rotor fixed d and q components. These components are transformed into the voltage space phasor with respect to the stator fixed reference frame, which serves as input for the PWM (pulse width modulation) algorithm. The switching signals of the PWM are then fed to the m phase power electronics converter.

The drive parameter record utilized in the inverter model contains the machine parameter record and further param-
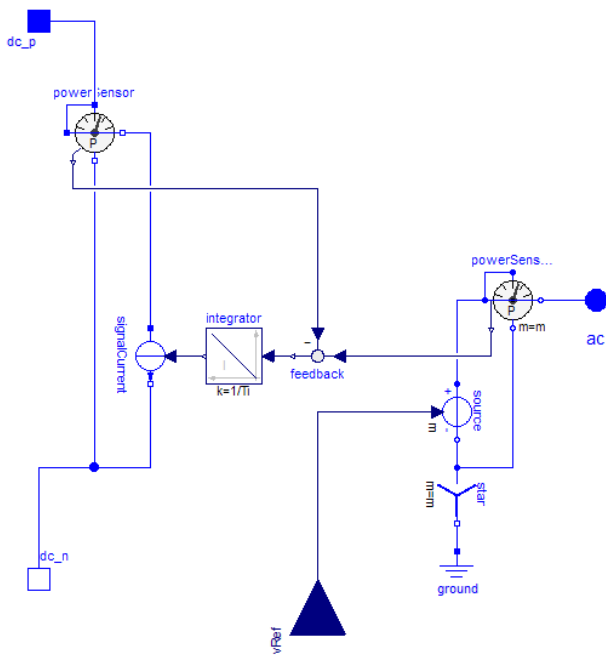
Figure 4: Averaging converter model

eters of the drive configuration. The controller parameters are calculated from the machine parameters and propagated to the controller models. The user can tune the pre-configured control parameters by means of tuning parameters.

5 DC / AC Converter

Three different levels of abstraction for DC to AC converters are provided in the EDrives library.

- First, the quasi static converter is modeled as an ideal quasi static voltage source. Power balance between DC input and AC output is complied. This means the efficiency of the quasi static converter is 100 %.

- Second, in the averaging converter model (Fig. 4) the space phasor voltages are transformed into transient phase voltages. These voltages are fed to an ideal voltage source. Considering power balance between DC input and AC output side, the converter efficiency is 100 %.

- Third, the switching converter consists of *m* legs. Each of them consists of two electronic switches and two anti parallel free-wheeling diodes. Switch and diode models are taken from the Modelica Standard Library (`IdealGTOThyristor` and `IdealDiode` from package `Modelica.Electrical.Analog.Ideal`). The inputs of the switching converter are boolean switching signals (`on|off`) for each leg.

6 Control

Machine control is based on FOC (field oriented control) and consists of the following components:

- optional speed controller

- torque flux controller

- current controller

In each controller model, a space vector representation of voltages and currents in the rotor fixed reference frame is used. These phasors are also distributed to the machine bus. The input for the torque-flux-controller is the commanded torque which can be provided by the user or an optional speed controller.

6.1 Optional Speed Controller

The external speed controller is implemented as limited PI controller with anti windup. It feeds the commanded torque via signal connector to the inverter and has to be connected to the inverter via machine bus. The parametrization is defined by a record `speedControllerData` which is a sub-record of the drive parameter record. The record `speedControllerData` calculates the parameters for the PI controller based on the symmetrical optimum method.The subscripts *iq* refer to the settings of the current controller with respect to the *q* axis [8]:

$$k_p = k_{\text{Tune}} \frac{(J_{\text{Machine}} + J_{\text{Load}})\,\omega_{\text{Nominal}}}{\tau_{\text{Nominal}}} \frac{2k_{\text{Tune},iq}}{T_i}$$

$$T_i = T_{i,iq} \frac{4}{k_{\text{Tune},iq}}$$

6.2 Torque Flux Controller

The torque-flux-controller shown in Fig. 5 limits the commanded torque with the maximum speed dependent torque. In the field weakening range torque is limited proportional to one over speed. In the base speed range the maximum torque is constant. The actual machine torque, magnetic flux and stator voltage are determined by a controller internal machine model (section 6.4). The input quantities of the controller internal machine model are the commanded current space vector and the rotor angular position. In a real drive, torque, magnetic flux and stator voltage are either determined by an observer or an on-line model which is implemented in the control software.

The deviation between the commanded and the modeled torque is the input for a fast integrator controlling the *q* current component.

The *d* current component is determined by the flux controller which implements a simple flux control algorithm. It can be replaced by a more sophisticated flux controller with low effort. First, the actual voltage, calculated by the controller internal machine model is compared with the maximum admissible voltage. This is the minimum of the maximum admissible machine voltage defined by a parameter
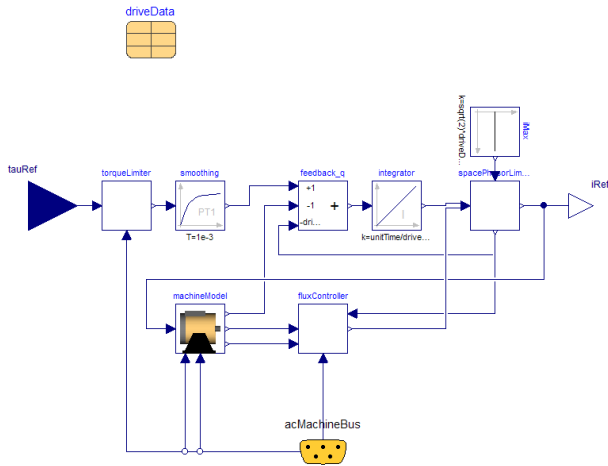
Figure 5: Torque flux controller

and the maximum obtainable voltage according to the actual DC voltage. Second, the actual magnetic stator flux of the machine is compared with the nominal magnetic flux. Nominal flux is calculated by an auxiliary quasi static machine model to determine the nominal operating conditions defined by the machine parameter record. If the actual voltage exceeds the maximum admissible voltage, field weakening occurs. When the actual magnetic flux exceeds the nominal magnetic flux, field weakening ends, entering the base speed range where magnetic flux is kept constant.

In the constant flux region, the d current component is controlled by a fast integrator to achieve constant magnetic flux. In the field weakening region, the d current component is determined by the same integrator to keep the voltage limited.

The outputs of the torque-flux-controller are the commanded d and q current component of the current space phasor in the rotor fixed reference frame. These components are limited to the maximum admissible machine current, which is defined in the drive parameter record. The limited current space phasor is equal to the commanded current space phasor which is forwarded to the current controller. The commanded current space phasor is also fed to the controller internal machine model.

6.3 Current Controller

The current controller for the transient inverters (averaging and switching) utilizes two limited PI controllers with anti windup to determine the stator voltages V_d and V_q, respectively. The stator voltage space phasor is determined such way that the commanded currents, $\hat{I}_d$ and $\hat{I}_q$, and the actual currents, I_d and I_q, coincide.

The voltage equations of the PM synchronous machine are

$$V_d = R_s\hat{I}_d - \omega L_q\hat{I}_q + L_d\frac{d\hat{I}_d}{dt},$$

$$V_q = R_s\hat{I}_q + \omega L_d\hat{I}_d + L_q\frac{d\hat{I}_q}{dt} + V_{PM},$$

where ω is the angular rotor speed. Parameter R_s represents the stator resistance at rated operating temperature, and L_d and L_q are the total inductances of the d and q axis. The controller internal machine model determines the voltage drops $R_s\hat{I}_d + \omega L_q\hat{I}_q$ and $R_s\hat{I}_q + \omega L_d\hat{I}_d + V_{PM}$, which are added as feed forward signal to the PI controller outputs. This way the PI controllers solely controls the transient current deviations $\frac{d\hat{I}_d}{dt}$ and $\frac{d\hat{I}_q}{dt}$; in literature this technique often is called "decoupling". In the field weakening region the desired voltage space phasor is limited to the maximum admissible voltage (section 6.2).

Since the machine transfer function represents a first order delay, the parametrization of the PI controllers is based on the compensation method [8].

$$k_{p,id} = k_{\text{Tune},id}R_s$$
$$T_{i,id} = \frac{L_d}{R_s}$$

$$k_{p,iq} = k_{\text{Tune},iq}R_s$$
$$T_{i,iq} = \frac{L_q}{R_s}$$

These controller parameters are calculated by the drive parameter record.

The quasi static current controller is simpler: The voltages to match the actual and commanded d and q currents are calculated by the controller internal machine model. The complex voltage is limited to the maximum admissible voltage and fed to the output.

6.4 Controller Internal Machine Model

The controller internal machine model shown in Fig. 6 is used to calculate the stator voltage phasor, the magnetic flux and torque related to the the commanded stator current space phasor and rotor position. The actual stator winding temperature, however, is taken into account by the controller internal machine model. This machine model is implemented as the instance of a quasi static machine model as described in [3].

Using a quasi static machine model to implement the controller internal machine model is much more structured and modular than an model just implementing text book equations describing the machine's behavior. Additionally, all improvements of the machine model from the MSL can be utilized. Machine losses and temperature effects are fully taken into account, resulting in a flexible and adaptive machine model.

6.5 Space Vector PWM

SVPWM (space vector pulse width modulation) is considered in literature as state of the art PWM, as discussed, e.g., in [9–13]. The PWM block included in the actual version
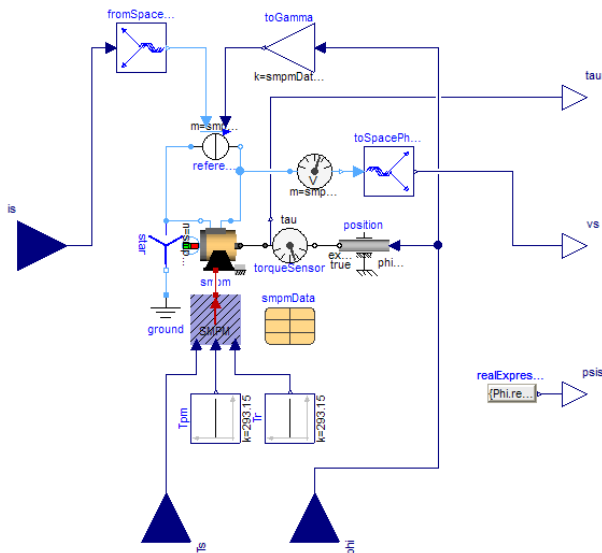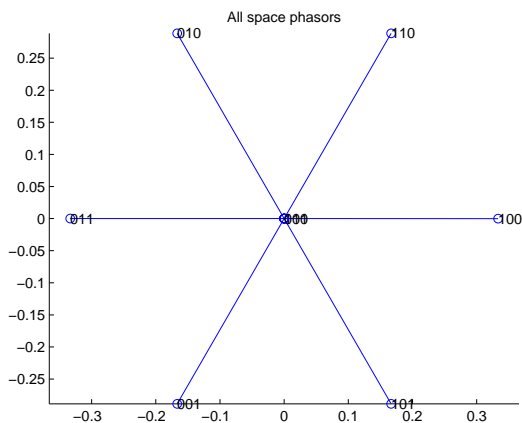
Figure 6: Controller internal machine model

Figure 7: SVPWM switching states in d-q-voltage plane [p.u.] for $m = 3$ phases

of the EDrives library currently implements a SVPWM algorithm for $m = 3$ phases. An extension to arbitrary number of phases is under development. The algorithm samples the desired voltage space phasor and the actual DC voltage according to the switching frequency. The possible space phasor locations according to the switching states shown in Fig. 7 are calculated by a function. First, the sector is determined where the commanded voltage space phasor is located. This voltage space phasor can be composed of the two neighbored switching states and either of the two zero switching states. The boolean switching states are fed to the output, to control the electronics switches of the converter (section 5).

7   Examples

The first example shown in Fig. 1 simulates a quasi static synchronous machine with permanent magnets, fed by a

quasi static inverter. For the given drive configuration three different levels of abstraction will be compared. First, the original simulation based on the quasi static machine and inverter is performed. Second, the permanent magnet synchronous machine model smpm is redeclared as transient machine. Additionally, the inverter model smpmInverter feeding the permanent magnet synchronous machine has to be redeclared as averaging inverter. Then an additional simulation run is be performed. Third, the averaging inverter is redeclared by a switching inverter.

The machine parameters of the simulation example are derived from the default MSL machine parameters summarized in Tab. 1. No damper cage is used; copper losses are caused by heat dissipation of the stator resistors. All other losses are neglected in the preformed example. The machine is mechanically loaded by an inertia and a quadratic speed dependent torque which reaches nominal torque at nominal speed, representing a fan or a pump. The inverter is connected to a DC voltage source with resistors, simulating a battery. At $t = 0.1\,\text{s}$ a step from zero to nominal torque is applied to the reference torque input. After accelerating, the drive reaches an equilibrium with the given load characteristic.

The following figures compare the results of

- a transient synchronous machine fed by a switching inverter,

- a transient synchronous machine fed by an averaging inverter and

- a quasi static synchronous machine fed by a quasi static inverter.

In Fig. 8 the actual torques of the three cases follow the reference torque step very well. As expected, the quasi static torque is nearly identical with the reference torque. The switching case shows a torque ripple according to the switching frequency (5 kHz). The speed trajectories in Fig. 9 show only small differences. The plots of the RMS stator currents in Fig. 10 reveal that the d current increases at $t = 0\,\text{s}$ to set the nominal magnetic flux. At $t = 0.1\,\text{s}$ the q current component is controlled to generate the reference torque. Again the quasi static case shows no transients, whereas the switching current is superimposed by a rest of current ripple left by filtering. The unfiltered phase currents at the end of the simulation are depicted in Fig. 11. A fast Fourier transform (FFT) of phase current of phase 1 is shown in Fig. 12. Note that the largest harmonics – apart from the fundamental wave – are located at multiples of the switching frequency (5 kHz). The filtered stator RMS voltages in Fig. 13 rise with speed since induced voltage increases. Yet the maximum voltage of 100 V is not exceeded. The second example shown in Fig. 14 is derived from the first example, but the number of phases is changed to $m = 5$. The desired torque step is replaced by a speed controller which input is a desired speed cycle, and the parametrization of the speed controller is taken from the record speedControllerData, embedded in the drive parameter record.

number of phases	m		arbitrary
number of pole pairs	p		2
rotor's moment of inertia	J_r	$kg \cdot m^2$	0.29
nominal frequency	$f_{s,\text{Nominal}}$	Hz	50
open circuit voltage at nominal speed	$V_{s,\text{OpenCircuit}}$	V	112.3
nominal voltage per phase	$V_{s,\text{Nominal}}$	V	100
nominal current per phase	$I_{s,\text{Nominal}}$	A	$100 \cdot \frac{m}{3}$
stator resistance per phase at 20°C	R_s	Ω	$0.03 \cdot \frac{m}{3}$
stator stray inductance per phase	$L_{s\sigma}$	H	$\frac{0.1}{2\pi f_{s,\text{Nominal}}} \cdot \frac{m}{3}$
stator main field inductance per phase, d axis	L_{md}	H	$\frac{0.3}{2\pi f_{s,\text{Nominal}}} \cdot \frac{m}{3}$
stator main field inductance per phase, q axis	L_{mq}	H	$\frac{0.3}{2\pi f_{s,\text{Nominal}}} \cdot \frac{m}{3}$

Table 1: Machine parameters
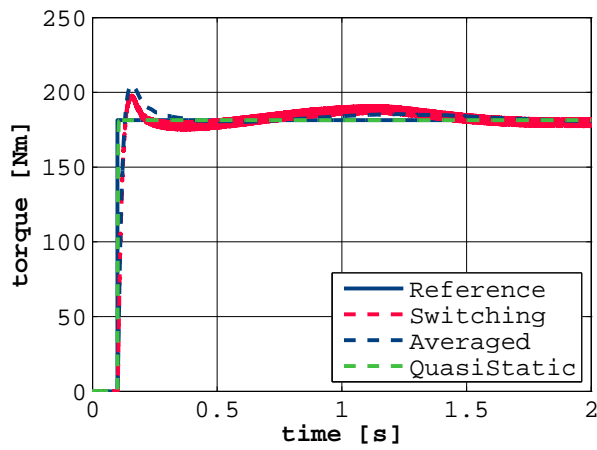
Figure 8: Example 1: simulation results of electrical torque

Figure 9: Example 1: simulation results of speed

The simulations results are obtained with a transient machine model and an averaging inverter model. Fig. 15 shows the reference torque given by the speed controller and the machine electrical torque. The actual speed follows reference speed very well, as shown in Fig. 16. Fig. 17 shows the 5 phase currents during 1 second of the initial acceleration. This proves that the control system works well in both driving directions, both for motor and generator mode, and for phase numbers $m > 3$.

8 Outlook

It is planned to release the new EDrives library as commercial tool-independent library. The modular concept of the library supports the following development steps:

- Enhancing the SVPWM algorithm to arbitrary numbers of phases $m > 3$.

- Extending both the switching converter model and the SVPWM algorithm to multilevel inverter configurations.

- Considering alternative flux control strategies.

- Implementing the control structures for induction machines with squirrel cage; power electronic converters
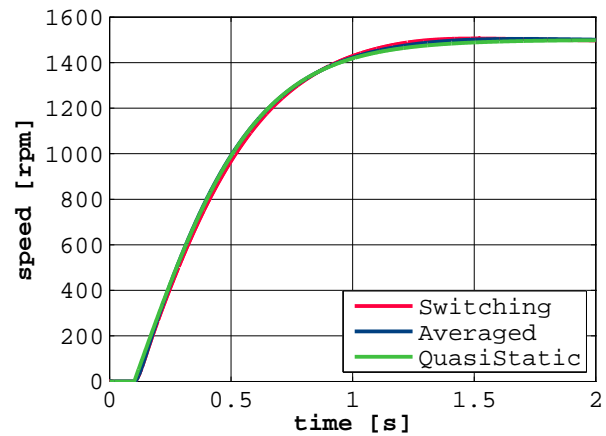
(quasi static, averaging and switching) as well as the SVPWM algorithm can be re-used.

- Providing diode bridge converters as mains converters (quasi static, averaging and switching).

- Developing active mains converters with voltage-oriented control as described in [14]; power electronic converters (quasi static, averaging and switching) as well as the SVPWM algorithm can be re-used.

- In the current implementation, the switching power converter models using the `IdealGTOThyristor` and the `IdealDiode` model from package `Modelica.Electrical.Analog.Ideal` take conduction losses into account. Enhancements include implementation of switching losses for switching power converters, and both conduction and switching losses for averaging and quasi static power converters. One particular focus of development is a user friendly parametrization technique. Furthermore, thermal effects shall be considered, including a thermal port for coupling the electrical models with thermal inverter models and cooling circuits.

- Adaptive control techniques shall be considered, taking temperature dependencies and saturation effects into account.
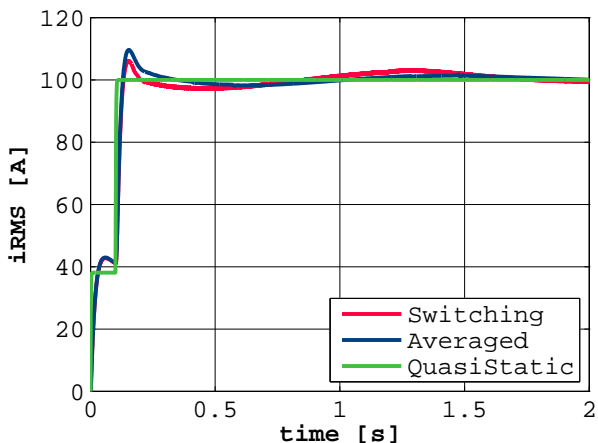
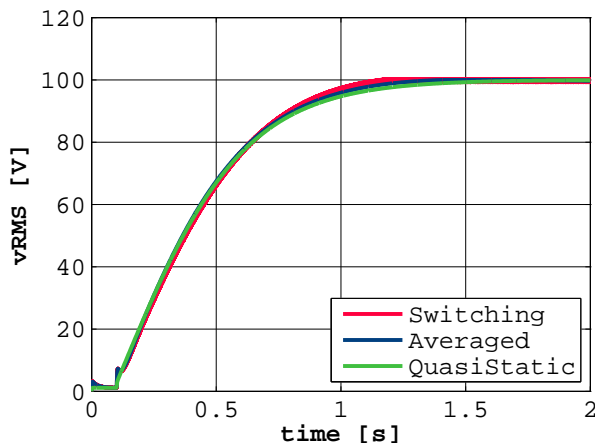Figure 10: Example 1: simulation results of stator RMS current
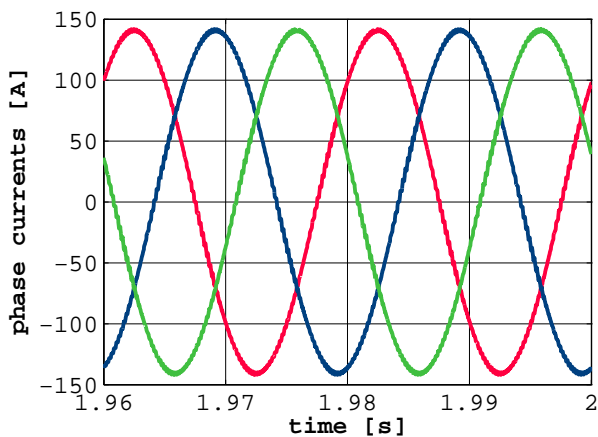

Figure 11: Example 1: simulation results of stator phase currents

9   Conclusions
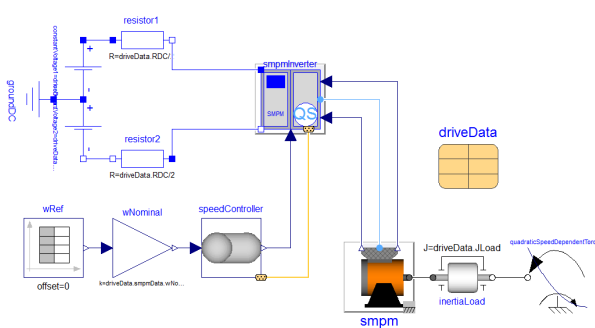
The new EDrives library for modeling inverter drives utilizes the machine models from the MSL. The provided in-

Figure 13: Example 1: simulation results of filtered stator RMS voltage

Figure 14: Example 2: speed controlled PM synchronous machine drive

verter models allow phase numbers equal or greater than 3, like the `Modelica.Magnetic.FundamentalWave` and the new `Modelica.Magnetic.QuasiStatic.FundamentalWave`. The control structures are based on FOC. The parametrization of the controllers is described. Simulation examples demonstrate the usage of matching inverter and machine models with respect to different levels of abstraction.
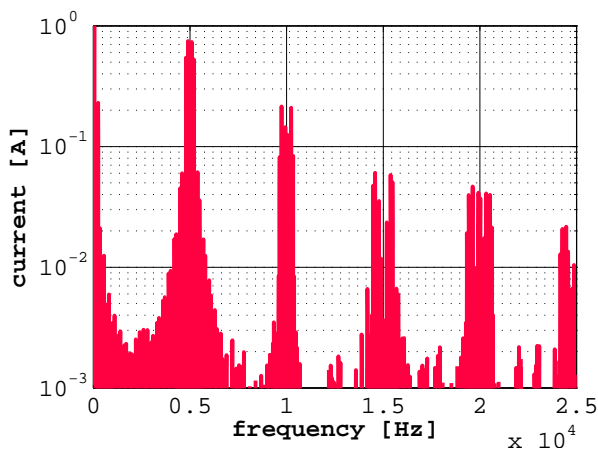

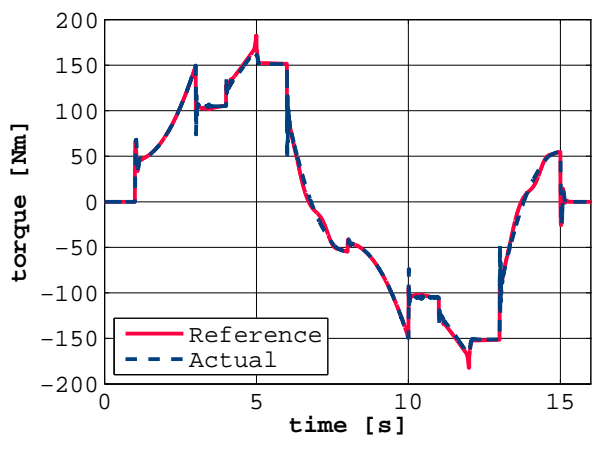Figure 12: FFT of simulation of stator RMS current at equilibrium

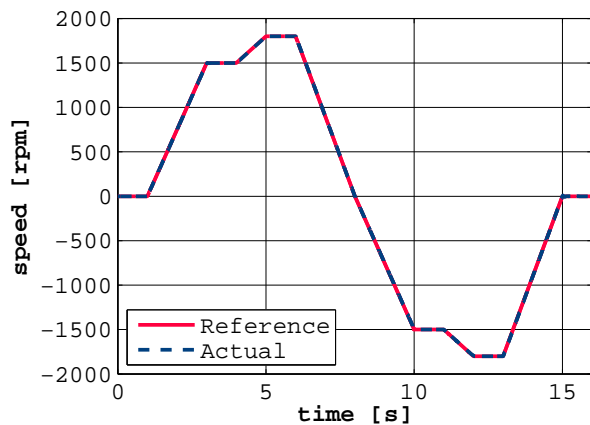Figure 15: Example 2: simulation results of electrical torque

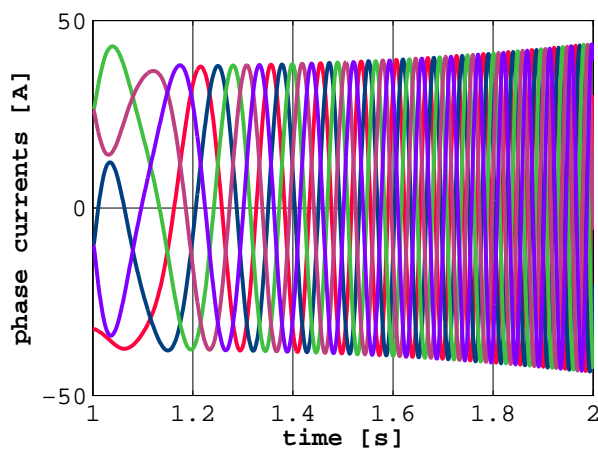Figure 16: Example 2: simulation results of speed

Figure 17: Example 2: simulation results of phase currents

The new EDrives library opens a new field of applications for mobility and industrial applications: Quasi static models with highest performance allow long term energy consumption simulations. Averaging models enable testing the interaction of the drive control with the whole system at high performance. The most detailed switching models allow the investigation of ripple effects and the development and customization of controller concepts and designs.

The package shall be released as a commercial library, the first version comprises quasi static, averaging and switching inverter models for synchronous machines. Enhancements and extensions are currently under development.

References

[1] A. Haumer, T. Bäuml, and C. Kral, "Multiphysical simulation improves engineering of electric drives," *7th EUROSIM Congress on Modelling and Simulation*, September 2010.

[2] C. Kral, A. Haumer, and R. Wöhrnschimmel, "Extension of the fundamentalwave library towards multi phase electric machine models," *submitted for possible publication at Modelica Conference*, 2014.

[3] C. Kral and A. Haumer, "New multi phase quasi static fundamental wave electric machine models for high performance simulations," *submitted for possible publication at Modelica Conference*, 2014.

[4] A. Haumer, C. Kral, H. Kapeller, T. Bäuml, and J. V. Gragger, "The AdvancedMachines library: Loss models for electric machines," *Proceedings of the 7th Modelica Conference*, pp. 847–854, 2009.

[5] C. Kral and A. Haumer, *Object Oriented Modeling of Rotating Electrical Machines*. INTECH, 2011.

[6] C. Kral, A. Haumer, and M. Plainer, "Simulation of a thermal model of a surface cooled squirrel cage induction machine by means of the SimpleFlow-library," *International Modelica Conference, 4th, Hamburg, Germany*, pp. 213–218, 2005.

[7] A. Haumer, C. Kral, V. Vukovic, A. David, C. Hettfleisch, and A. Huzsvar, "A parametrization scheme for high performance thermal models of electric machines using Modelica," *MATHMOD VIENNA 2012*, 2012.

[8] P.-I. H. Lutz and P. D. I. W. Wendt, *Taschenbuch der Regelungstechnik, 5. erweiterte Auflage*. Frankfurt am Main: Wissenschaftlicher Verlag Harri Deutsch, 2003.

[9] M. J. Duran, F. Barrero, and S. Toral, "Multi-phase space vector pulse width modulation: Applications and strategies.," *Inertnational Conference on Renewable Energies and Power Quality, ICREPQ 2007*, 2007.

[10] S. Halasz, "PWM strategies of multi-phase inverters," *IECON 2008. 34th Annual Conference of IEEE Industrial Electronics*, 2008.

[11] D. Dujic, M. Jones, and E. Levi, "Generalised space vector PWM for sinusoidal output voltage generation with multiphase voltage source inverters," *International Journal of Industrial Electronics and Drives*, vol. 1, no. 1, 2009.

[12] M. Mengoni, *Modulation Techniques for Multi-Phase Converters and Control Strategies for Multi-Phase Electric Drives*. PhD thesis, University of Bologna, 2010.

[13] D. G. Holmes and T. A. Lipo, *Pulse Width Modulation for Power Converters*. Wiley Interscience, 2003.

[14] A. Haumer and C. Kral, "Modeling a mains connected PWM converter with voltage-oriented control," *Modelica Conference*, 2011.