

Stavanger, Norway, May 13 - 16, 2009

Simulation of Electric Drive System Faults in Hybrid Electric Vehicles

Dietmar Winkler, Clemens Gühmann

Technische Universität Berlin

School of Electrical Engineering and Computer Science

Chair of Electronic Measurement and Diagnostic Technology

{Dietmar.Winkler|Clemens.Guehmann}@tu-berlin.de

Abstract

With the increased number of electric drive systems in hybrid electric vehicles also the probability of electric faults within those systems rises. We have developed a Modelica[®] library to simulate field-oriented drive systems, including faults. The library is called `freeFOClib` (short for “free Field-Oriented Control library”) and can be used to build a field-oriented control system for existing machine models from the *Modelica Standard Library*, investigate the impact of electric faults (e.g., battery faults, inverter faults, machine faults) on a electric drive system, and run simulations to estimate the fuel consumption of hybrid electric vehicles. This paper will show how to model and simulate a faulty electric drives systems in combination with a hybrid electric vehicle.

Keywords: electric drive, HEV (hybrid electric vehicle), modeling, simulation, power-train, battery model

1 Introduction

In automotive applications the number of electric motors used is increasing rapidly. Most of them are doing their work in the background unnoticed by the car owners/users. Whilst a power window is quite an obvious application for an electric motor, the active controlled throttle valve might not be. And with more and more tasks going to be performed *by-wire* (e.g., braking, steering) the number of electric motors used is due to increase even more. But not only *small* electric motors are present. With the electric motor being used for active propulsion in hybrid electric vehicles (HEV) also the power rating of motors used grows bigger.

But how do all these little and large motors work together? How should the manufacturer develop the controller? What happens if there is a fault in the system? Will the faults cause serious damage or just minor inconveniences?

All these question could be answered by using simulations to investigate the normal and faulty behaviour. For the creation of such simulation models we need both, appropriate machine mod-

els and the suitable machine controllers. The overall simulation model will contain signals from different physical domains (i.e., electrical and mechanical).

The modelling language Modelica was especially developed to simplify the simulation in different physical domains within one simulation model. The multi-domain capability allows us to build simulation models of hybrid electric vehicles easier than with other simulation languages thus can we concentrate on the physics of a model rather than building models which represent mathematical equations which in turn then represent the actual physical behaviour [1, 2].

2 The new library

For the simulation of machines the Modelica Standard Library contains a sub-library called: `Modelica.Electrical.Machines` [3]. This library contains basic three-phase models of asynchronous and synchronous machines as well as DC machine models. To control these machines

the modeller still has to provide his own controllers since currently there is no *free* Modelica library available to provide complete electric drive models.

So in order to simulate more complex electric drive applications a new “free Field-Oriented Control Library” (`freeFOCLib`) is being developed [4].

The `freeFOCLib` should allow the user to model and simulate all aspects of the an electric drive. A standard electric drive normally consist of components like power sources, power electronics, controllers, electric machines, and interfaces. The communication of the blocks can be done either via the classic approach by use of input and output connectors or by the use of so-called bus signals. The bus structure orients itself on the new bus structure of the Vehicle Interfaces Library [5]. This, for example, should allow easier simulation of power-train simulations of hybrid electric vehicles.

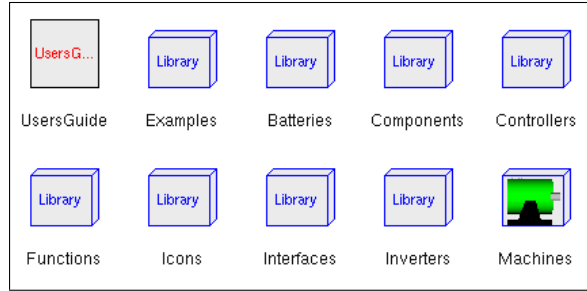


Figure 1: Top-level packages of the `freeFOCLib`

Figure 1 shows a graphical representation of the uppermost hierarchy level of the library.

The library started of with some basic battery models and controller types for field-oriented control. Over time more and more bits were added to enhance the models and the field of application. In the beginning we only had a simple ideal battery model (i.e., an ideal controlled voltage source) and some Nickel-Metal-Hydrate battery models from the Advisor library [6]. Over time we added dynamic models Lithium-Ion battery with different levels of detail to select from.

The controller package contains different controller types for the field-oriented control. Simple controllers using different flux-models for asynchronous and synchronous machines.

The machines packages contains models of asynchronous and synchronous m -phase machines. This means an extension to the 3-phase machine models currently available in the *Modelica Standard Library*. The special thing about the machine models in the `freeFOCLib` is the they are using the m -phase presentation. We do this in order to be able to introduce faults directly into the components without the need of a $d-q$ -transformation.

3 Modelling of electrical induction machines

The classical approach to describe an electrical machine is by the use of the so called *equivalent phase circuit diagram*. In the following the machine equations shall be explained taking an asynchronous induction machine as an example. There are however much similarities to the synchronous machine which shall not be investigated further in this short paper (see for example [7]).

3.1 Physical equations

Figure 2 represents the equivalent circuit diagram for one phase of the electrical machine. It consists of resistances representing the ohmic losses in the field windings (R_s and R_r) and the stray inductance ($L_{s\sigma}$ and $L_{r\sigma}$) in the stator and the rotor. The rotor and stator are coupled via the mutual inductance (L_m).

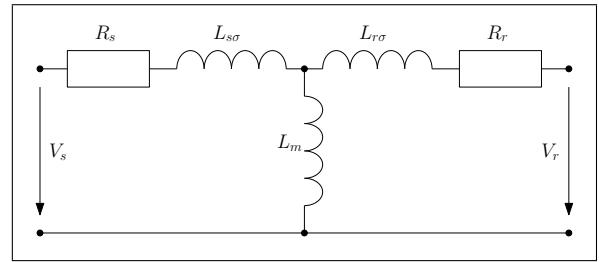


Figure 2: Equivalent circuit diagram of an electrical induction machine

The standard machine equations for the rotor and stator voltages can be written as:

$$\vec{v}_s = R_s \vec{i}_s + L_{s\sigma} \frac{d\vec{i}_s}{dt} + \frac{d\vec{\Psi}_{ms}}{dt} \quad (1)$$

$$\vec{v}_r = R_r \vec{i}_r + L_{r\sigma} \frac{d\vec{i}_r}{dt} + \frac{d\vec{\Psi}_{mr}}{dt} \quad (2)$$

where Ψ_{mx} are the flux vectors and defined as

$$\vec{\Psi}_{ms} = \mathbf{L}_{ss} \cdot \vec{i}_s + \mathbf{L}_{sr}(\gamma) \cdot \vec{i}_r \quad (3)$$

$$\vec{\Psi}_{mr} = \mathbf{L}_{rs}(\gamma) \cdot \vec{i}_s + \mathbf{L}_{rr} \cdot \vec{i}_r \quad (4)$$

In the equations (3) and (4) you can see the four inductance matrices. Each of these have the dimension ($m \times m$) where m stands for the number of phases (three in most cases). \mathbf{L}_{ss} and \mathbf{L}_{rr} represent the self-inductance matrices for the stator and the rotor, respectively.

$$\mathbf{L}_{ss} = \begin{bmatrix} L_m & L_{sm} & L_{sm} \\ L_{sm} & L_m & L_{sm} \\ L_{sm} & L_{sm} & L_m \end{bmatrix} \quad (5)$$

and

$$\mathbf{L}_{rr} = \begin{bmatrix} L_m & L_{rm} & L_{rm} \\ L_{rm} & L_m & L_{rm} \\ L_{rm} & L_{rm} & L_m \end{bmatrix} \quad (6)$$

If the machine is symmetrically built then the matrices can be further simplified by providing the equations:

$$L_{sm} = L_{rm} = L_m \cdot \cos\left(\frac{2}{m}\pi\right) \quad (7)$$

This also means that stator and rotor self-inductance would be identically.

We still got the two matrices $\mathbf{L}_{sr}(\gamma)$ and $\mathbf{L}_{rs}(\gamma)$ left. These represent the mutual coupling between the stator and the rotor side and vice versa. Unfortunately the coupling effect depends on the rotor position γ . For a three phase machine ($m = 3$) this gives:

$$\begin{aligned} \mathbf{L}_{sr}(\gamma) &= [\mathbf{L}_{rs}(\gamma)]^T \dots \\ &= \begin{bmatrix} \cos(\gamma) & \cos(\gamma + \frac{2}{3}\pi) & \cos(\gamma - \frac{2}{3}\pi) \\ \cos(\gamma - \frac{2}{3}\pi) & \cos(\gamma) & \cos(\gamma + \frac{2}{3}\pi) \\ \cos(\gamma + \frac{2}{3}\pi) & \cos(\gamma - \frac{2}{3}\pi) & \cos(\gamma) \end{bmatrix} \end{aligned} \quad (8)$$

Now this equation can be used directly in a Mod-
elica model.

4 Library applications

The development of the `freeFOClib` was started with specific purposes in mind:

- field-oriented control of induction machines
- fault-simulations to investigate electrical and mechanical impacts of machine faults
- state of charge estimations for batteries in HEV applications
- investigate adaptive controller algorithms for electric machines

The `freeFOClib` contains an example model which allows for simulation of three different types of faults. These could be faults of the battery, faults in the inverter, and faults in the machine. In Figure 3 an example of an electric drive system is depicted.

4.1 Inverter and battery faults

To simulate faults in the battery and/or in the inverter, the example model in Figure 3 contains a fuse component `fuse_DC`. This model disconnects the DC current when a surge current is detected that violates the maximum rating of the

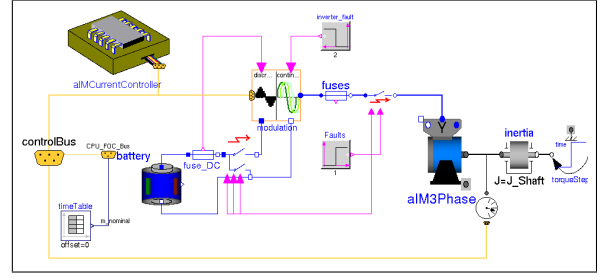


Figure 3: Example model of an electric drive system.

battery. The surge-proof fuse is not triggered right away but after a first order delay time which can be parametrised. Whenever the fuse is triggered the inverter gets a signal which will switch of the firing signals for the inverter bridge in turn. Another kind of battery fault would be a short circuit of the supply side of the inverter. This is accomplished by a simple switch that is triggered by a Boolean signal and which connects both support voltage connectors of the inverter. And at last a fault of firing signals can be applied directly via a signal `inverter_fault`. The combination of the different switches gives the ability to build even more fault scenarios.

4.2 Machine faults

In the electric machine models of our library the following fault scenarios can be simulated:

- open-circuit of a stator phase (e.g., a connecting cable is broken)
- short-circuit of one or more phase windings (e.g., insulation failure because of thermal stress within the stator or rotor)
- short-circuit phase to ground (e.g., insulation failure because of mechanical damage)

Each of these faults will have some influence of the torque produced by the electric drive.

4.2.1 Open-circuit fault

Here an example simulation of a synchronous machine having a open-circuit fault. This could occur if a connecting cable comes loose. In Figure 4 you can see the three phase currents and the mechanical torque depicted over time just before and after the connection of one phase was opened at the time of two seconds.

As you can see, the torque drops instantaneously and but the controller try to keep the torque output until at some point the fuses for the remaining phase currents get triggered.

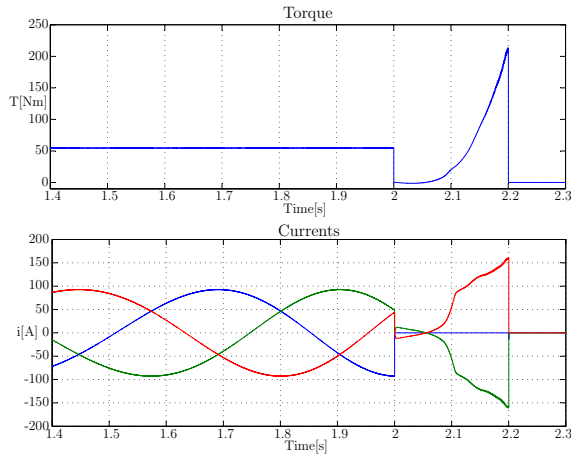


Figure 4: Fault scenario: open-circuit of one stator phase

4.2.2 Short-circuit within a phase winding

In Figure 5 a fault of the insulation between the phase windings of a stator coil is modelled. Such a fault can be caused by, for example, over-temperature or overload which in turn leads to an overheated stator winding. This behaviour is modelled by reducing the inductance value abruptly by 20 percent. In Figure 5 you can see the three phase currents and the mechanical torque over time just before and after the connection of one phase was opened at the time of $T = 2\text{sec}$.

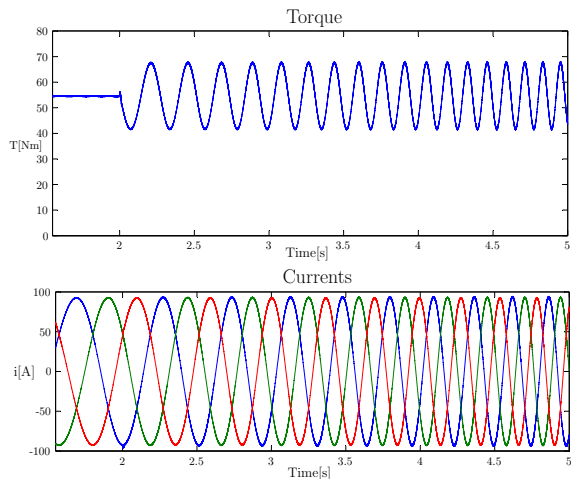


Figure 5: Fault scenario: short-circuit of phase windings

At first sight the electrical impact seems not to be very drastic. However since the field-oriented control now calculates the wrong control values the torque starts to oscillate quite considerable. If this electric drive is applied in a hybrid electric vehicle, for example, this could lead to reduction

of drive comfort. But not only this, depending on the mechanical system such oscillation could become unstable and cause major damage.

5 Faults in the HEV's drive system

The `freeFOClib` allows us to simulate faults in a electric drive system. The next step is to include the fault simulation into the drive system of a hybrid electric vehicle. In one of our earlier works we already build a simulation model for real-time simulation of hybrid electric vehicles [8]. Figure 6 shows the structure of such a drive system from that work.

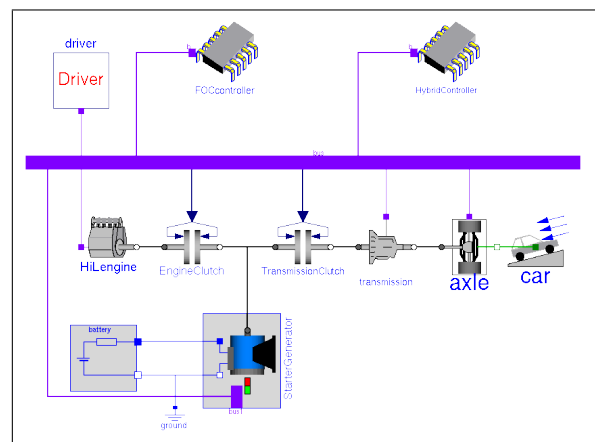


Figure 6: Model of the old HEV's drive system.

In the meantime there has been a steady improvement in the Modelica language and the libraries. The new version 2.0 of the PowerTrain library makes it much easier to model complex drive-train models by the heavy use of expandable bus connectors and replaceable model components. Using expandable bus connectors means for the modeller that he can simply connect a new signal to a bus and the signal variable will automatically be added to the bus and is therefore known throughout the whole bus architecture. The use of sub-buses then makes it even easier to organise the signals of the drive-system. Replaceable model components in turn mean that the modeller can give the end-user of the model a predefined choice of what kind of models he can replace (e.g., use of different combustion engines, different transmission types).

In Figure 7 the new model which contains drive-train with a starter/generator is depicted. Such an HEV architecture is also referred to as mild hybrid architecture. That means that the capability of the electric motor is mainly for assisting the combustion engine and for recuperation of brake energy. Depending on the electric motor rating it could also be used for electric driving at low speeds only.

We now exchange the normal electric drive model with a special type from the `freeFOClib`

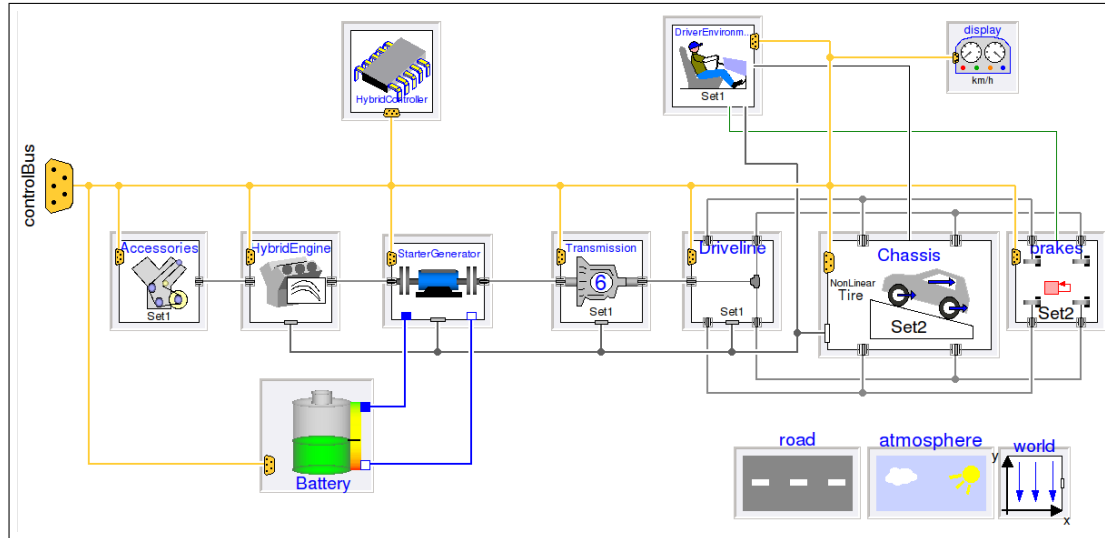


Figure 7: HEV drive system using the new PowerTrain library.

that can be used to introduce electrical faults. Thus we can then investigate the impact of electric faults on the drive train of the hybrid electric vehicle. The simulation results can then be used for designing different safety measures that should avoid major damage to the vehicle.

6 Electric drive test-bench

On important aspect of every simulation is the verification of the model parameters and the simulation behaviour in general.



Figure 8: Back to back test-bench set-up.

We use a test bench consisting of a asynchronous induction machine (the test object) as drive and a DC-machine as load coupled together via a torque-meter (see Figure 8). The torque-meter an analogue voltage signal for the load torque. This signal is then read in via high-speed analogue-digital converter card. For the speed signal an incremental resolver is used which is attached to the machine shaft. The TTL signal of this speed sensor is then used by the electric inverter, a *Siemens SINAMICS G120* for the field-oriented control algorithms.

In a first step different methods for parameter identification are used to verify the simulation models for the non-faulty behaviour. Afterwards we introduced faults using fuse switches to disconnect or short-circuit different phases of the test object and measure the dynamic values of currents, voltages and torques. With the measured data we could then verify also the fault-simulations of our machine models.

7 Future work

The current state of the library can be used to test different scenarios in order to investigate the mechanical and electrical impact of electric faults on drive train and the power system, respectively. This gives an idea of what could happen and how severe the impact would be on the drive experience and as a result improve safety functions of control units.

We are steadily improving the library by adding new drive controllers, new battery models, for example. The developed models of unbalanced electrical machines are going to be a part of the library as well as all components which are necessary for a field-oriented control (hence the name).

The *freeFOClib* will be released as free library which everybody can use and adapt. The idea is to provide a Modelica[®] library which gives the possibility to model a modern electrical drive system and still leaves the user the freedom to look at the underlying code and perhaps even contribute improvements to it.

One drawback of the library as currently used is the decrease in simulation speed as the simulation of the time-dependent inductance and its derivatives is very complex and uses a lot of computational power, i.e., time. One possible solution would be to investigate the effects of faults

in the frequency-domain. This approach is already applied in a different free library for power transients called SPOT and the use of which shall be investigated further in order to cut down the simulation time needed for complex fault simulations.

References

- [1] M. Tiller, *Introduction to Physical Modeling with Modelica*. Kluwer International Series in Engineering & Computer Science, Kluwer Academic Publishers, 2001.
- [2] P. A. Fritzson, *Principles of Object-Oriented Modeling and Simulation with Modelica 2.1*. No. 0-471-471631, John Wiley & Sons, Januar 2004.
- [3] C. Kral and A. Haumer, "Modelica libraries for dc machines, three phase and polyphase machines," in *Proceedings of the 4th International Modelica Conference* (G. Schmitz, ed.), pp. 549–558, Modelica Association, March 7-8 2005.
- [4] D. Winkler, E. Bakhach, F. Döring, S. Rinderer, F. André, and C. Schoepe, "freeFOClib - A free Field-Oriented Control library for Modelica." unreleased, see www.freefoclib.org for any news.
- [5] M. Dempsey, M. Gäfert, P. Harman, C. Kral, M. Otter, and P. Treffinger, "Coordinated Automotive Libraries for Vehicle System Modelling," in *Proceedings of the 5th International Modelica Conference* (D. C. Kral and A. Haumer, eds.), vol. 1, (Vienna), pp. 33–41, Modelica Association, arsenal research, September 2006.
- [6] T. Markel and K. Wipke, "Modeling grid-connected hybrid electric vehicles using ADVISOR," in *Applications and Advances, 2001. The Sixteenth Annual Battery Conference on*, pp. 23–29, 9-12 Jan. 2001.
- [7] D. Novotny and T. Lipo, *Vector Control and Dynamics of AC Drives*. Clarendon Press Oxford, 1996.
- [8] D. Winkler and C. Gühmann, "Hardware-in-the-Loop simulation of a hybrid electric vehicle using Modelica/Dymola," in *The 22nd International Battery, Hybrid and Fuel Cell Electric Vehicle Symposium & Exposition* (E. . Secretariat, ed.), (Yokohama, Japan), pp. 1054–1063, EVS, Japan Automobile Research Institute, 23-28 October 2006.

Authors

Dipl. Ing. Dietmar Winkler studied Electrical Engineering at the Technische Universität Berlin and specialised in the field of control of electric drives. Since January 2004 he works as a research and teaching assistant and doing his PhD at the Chair of Electronic Measurement and Diagnostic Technology at the TU Berlin.



web: claimid.com/dietmarw

email:

Dietmar.Winkler@tu-berlin.de

Prof. Dr.-Ing. Clemens Gühmann is head of the Chair of Electronic Measurement and Diagnostic Technology at the Technische Universität Berlin. The chair's research focuses on: Modern Data Processing Methods for Automotive Systems; Simulation, Test, and Calibration Methods for Automotive ECUs (HiL, MiL, SiL-Simulation); Modelling and Real-Time Simulation of Automotive Systems; Pattern Recognition and Technical Diagnosis; and Hybrid Vehicle Control Strategies.

mail: TU Berlin, Sekr. EN 13, Einsteinufer 17, 10587 Berlin, Germany

phone: +49 30 314-29393

fax: +49 30 314-25717

web: www.mdt.tu-berlin.de

email:

Clemens.Guehmann@tu-berlin.de

