Modelica Simulation of Electric Drives for Vehicular Applications – The Smart Drives Library

H. Giuliani*, C. Kral*, J.V. Gragger*, F. Pirker*
arsenal research
modelica@arsenal.ac.at

Abstract

This paper presents a new simulation library developed in -language for *Dymola* software- the *Smart Drives Library*. The proposed software was developed to meet the user's needs for easy-to-use components in drive control simulations, specifically for hybrid electric vehicles. Currently there is no software available that was designed at this level of abstraction. Other than conventional software the *Smart Drives Library* was build so that control systems containing electric units as well as mechanical parts can be simulated in an easy and concise way. The library includes all components necessary to simulate a modern drive system in automotive applications (e.g. machines, power electronics, control units, sources, etc.).

1 Motivation

1.1 Purpose and Scope of the Smart Drives Library

Due to well known ecological as well as economical reasons the efficient use of energy resources plays a significant role in automotive design. In the project *Smart Drives for Smart Cars* executed under the framework of *Arsenal Research*, electrifications of conventional car auxiliaries such as water pumps, air conditioning, oil pumps, etc., are going to be developed. With the aim of achieving the highest efficiency possible, which means finding the optimal energy concept for the automotive drive it is necessary to simulate the entire hybrid vehicle. Investigations of hybrid configurations showed that an efficiency increase of 17% in the entire electric drive led to a fuel savings of 13% for the hybrid vehicle. Another library for longitudinal dynamics simulation and electrical auxiliaries was developed by *Arsenal Research* as well. Using this library together with a machine control library the so called *Smart Drives Library* enables the user to model a hybrid vehicle as a single unit. A highly technological and energy usage drive model has been developed.

*Faradaygasse 3, A-1030 Vienna, Austria

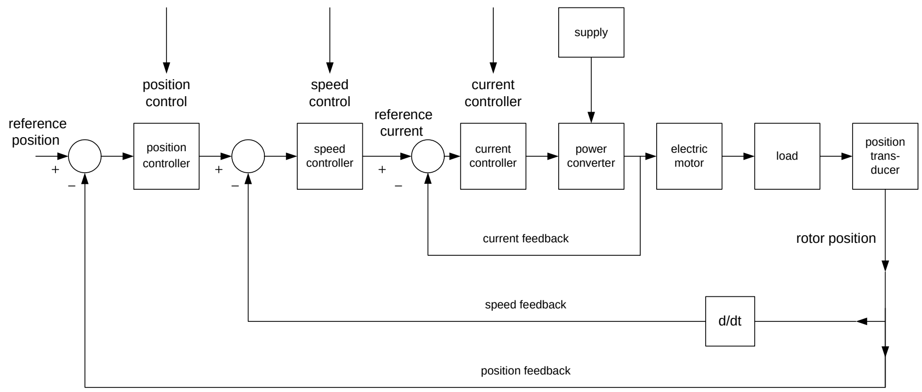

Figure 1: Schema of drive control

1.2 Why using Dymola?

Dymola, Dynamic Modeling Laboratory, is a complete tool for modelling and simulation of integrated and complex physical systems. The unique multi-engineering capabilities of *Dymola* gives rise to solutions for modelling and simulation. With this tool it is possible to simulate the dynamic behaviour and complex interactions between systems of many engineering fields, such as mechanical, electrical, thermodynamic, hydraulic, pneumatic, thermal- and control systems. *Dymola* empowers the user to model and simulate any physical component that can be described by ordinary differential equations and algebraic equations. The *Dymola* environment is open which means that users are free to create their own model libraries or modify standard libraries to better match the user's individual modelling and simulation needs. A powerful graphic editor for composing models is part of *Dymola*. *Dymola* is based on *Modelica*, which is an object-oriented language for physical modelling [1].

A library containing the basic models of electric machines already exists in the *Modelica* standard library [2]. However, without suitable drive control blocks, these machine models cannot be utilized in a resourceful and easy way. Based on the machines library, the *Smart Drives Library* facilitates the modelling of different control structures and control strategies. Power electronics and energy storage models are included as well.

2 The SmartDrivesLibrary

2.1 Electrical Drives Control

The main purpose of electric machine control systems is to make the machine perform in a desired fashion. For motor operation, the control system is designed to control the torque, speed or position. These objectives are schematically illustrated in Fig. 1.

The speed and position signal make the reference stator current of the machine follow the desired reference quantities. The task of the power converter is to process and control the electric energy supplying voltages and currents in a way that matches the machines specifications. While rectifiers or DC/DC buck-boost converter are used to feed direct current machines, three phase pulse-width-modulation inverters are used for induction machines drives. If the converter works with DC supplies, batteries, super caps or fuel cells can be applied. Based on the specific machine type used, different power electronics converter and diverse control strategies enforce different machine behaviour.

2.2 Structure of Library

The *Smart Drives Library* is divided in different packages. This structure supports the user in finding the desired drive control components. Each package contains sub packages.

1. Controllers: This package contains control components suitable for electric machines (permanent magnet synchronous induction machines, asynchronous induction machines with squirrel cage rotor, direct current machines).

2. Inverters: This package contains ideal and real power electronics modules to feed electric machines. Two groups of inverters got implemented. Ideal inverters with 100% efficiency in which the current flow is based on the energy balance between the input and output ports can be chosen as well as inverters with diode bridges and thyristor bridges in which the output voltage is generated due to defined switching states.

3. Sources: The power inverter can be supplied by batteries, super caps or fuel cells.

4. Sensors: This package contains sensors to measure physical values such as machine currents, machine voltages power, and the rotor angle

5. Connectors: Apart from the standard connectors, three different pairs of connectors are used in this package. Three phase and DC connectors contain all measured electrical and mechanical signals such as machine current, machine voltage and machine rotor angle. The flux phasor connectors got designed to contain magnitude, phase angle and angular velocity of estimated flux signals.

6. Components: This package contains all necessary components that are not machine specific.

7. Icons: This package contains graphical symbols of components which are used in different sub packages.

8. MachineQS: This package contains quasi stationary models of electrical machines (permanent synchronous induction machine, asynchronous induction machine) with integrated ideal inverter and field oriented control including voltage and current limitation as well as flux weakening. The MachineQS components must be fed via dc-bus. Compared to the already existing machines in the *Modelica* standard library the

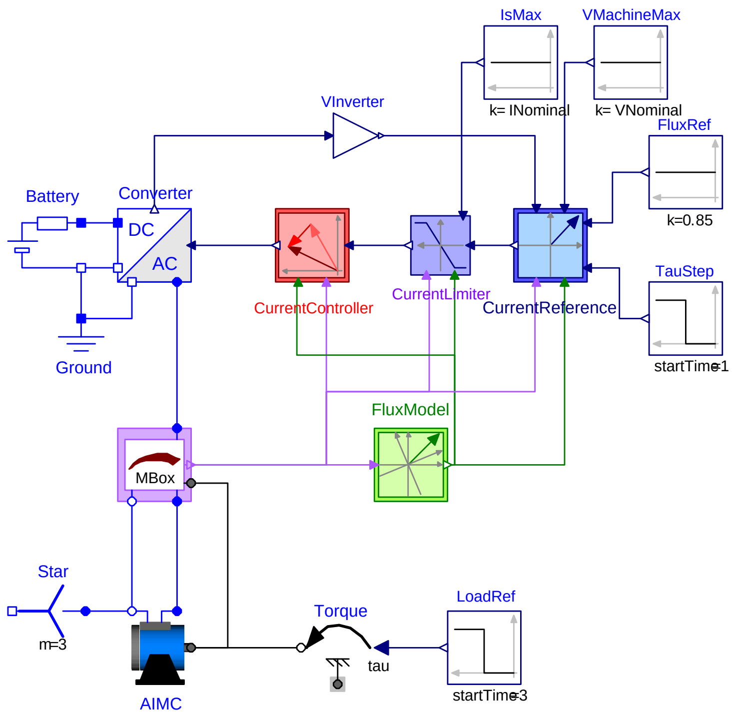

Figure 2: Induction machine with field oriented control

MachineQS components present a model where the transient oscillations are replaced by algebraic steady state terms.

9. Examples: This package contains different setups to demonstrate the usage of packages such as components, inverter, sources and connectors etc. in combination with different machines.

2.3 Example of an Induction Machine with Field Oriented Control

In this section, the application of the *Smart Drives Library* is explained by outlining a short example (Fig. 2). The aim is to control a star connected (`Star`) induction machine (`AIMC`) over a wide speed range while the torque is kept constant at the nominal value. Field oriented control (FOC) [3] is employed. In electrical machines the developed electrical torque is a function of the electrical rotor flux linkage and the stator current. The main feature of FOC is that the stator current is controlled in such a way that it is oriented with respect to the rotor flux. The induction machine is connected to a voltage con-

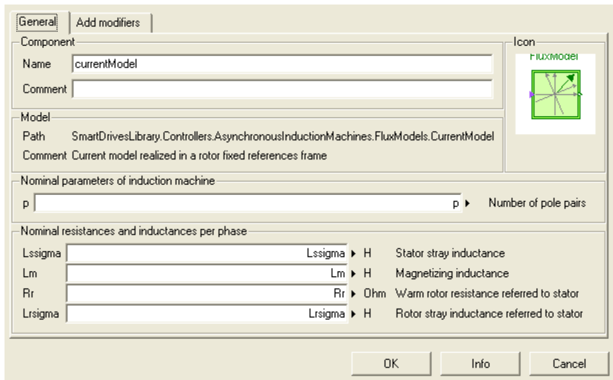

Figure 3: Set of parameters for the flux model

trolled DC/AC converter (`Converter`) which is supplied by a battery (`Battery`).Since there is no speed and no position controller the inner torque is commanded (`TauRef`) and provides a reference current in the so called block `CurrentReference`. The reference stator current is limited such way that the stator voltage does not exceed the maximal inverter voltage (`VInverter`) or the maximal machine voltage (`VMachineMax`). Command values of stator voltages get processed in the block named `CurrentControl`. FOC is based on the separation of the stator current into a torque generating and a flux generating component. Therefore it is possible to control both torque and flux independently in the block `CurrentReference`. Flux is determined from the measured voltages and currents (`MBox`) by a flux model (`FluxModel`). At high speed, the flux generating current component is weakened so that the voltage reserve is satisfactory. The `CurrentLimiter` limits the reference stator current by means of the maximum stator current (`IsMax`). Additional reference blocks are load torque (`LoadRef`) and reference rotor flux (`FluxRef`). All specific block parameters are kept at their default values and can be adjusted according to the user's choice (Fig. 3).

2.4 Example of a Vehicular Application

The use of electric machines in vehicular applications as hybrid drive claims sophisticated control concepts to provide optimal efficiency concerning the interaction between the supply and the electric machines. In the application example shown in Fig. 4 the combination of two electric machines and a rechargeable battery drive a mechanical water pump [4] which works in a car cooling system. One electric machine (`IMQS2`) works as generator and is driven by the internal combustion engine. The other machine (`IMQS1`) works as motor and provides the water pump drive. Since simulation time is drastically reduced using non transient equations, the quasi stationary machines implemented in the *Smart Drives Library* were used. If the battery's state of charge reaches the lower limit the generator pushes back power into the battery. In Fig. 5 characteristic and concise system values are

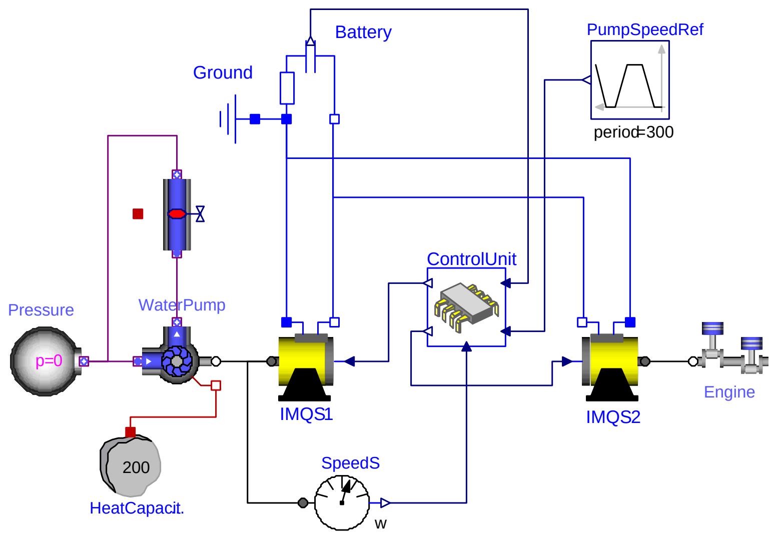

Figure 4: Example of a water pump driven by an electric motor

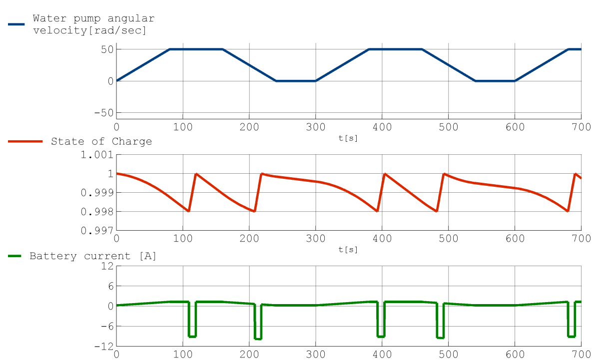

Figure 5: Some characteristic values of water pump example

displayed. A speed controller integrated in the control unit (`ControlUnit`) handles the water pump angular velocity based on a reference pump speed (`PumpSpeedRef`). Furthermore the control unit includes the battery charge and recharge strategy. The state of charge determines if the generator is to couple into the pump system in order to push back energy to the battery. The battery current sign changes depending on battery energy flow direction.

3 Conclusion

As it is shown in this work, the *Smart Drives Library* is a powerful tool to simulate drive systems for automotive applications. The straightforward structure of the library minimizes the user's efforts to simulate specific drive control setups for vehicular applications. Apart from the *Smart Drives Library* there are a number of other simulation tools for solving control system problems on the software market. However, most of these tools do not facilitate manageable simulation setups that hold mechanical units and electric components as well as their interfaces. Specifically drive control simulation in vehicular applications, furthermore in hybrid vehicles, is massively challenged with combinations of mechanical units and electric components in one full system. The *Smart Drives Library* was specifically designed to ease simulating such complex systems in *Dymola*. Another important feature of the proposed *Smart Drives Library* is the user-friendly design of the components included. Working with conventional software packages users are forced to grasp theories of many different fields (e.g. control design, power electronics, mechanics, etc.) before building vast, abstract systems for simulation. However, in the *Smart Drives Library* most of the basic theories are already incorporated right in the specific components. This enables the users to shorten development cycles and time to market. Additional advantages are the variety of examples included and the documentation of the specific components focussing on practical applications and aiming on leading the user to build and tune own simulations in little. Since simulation usually is a preliminary design step before embedded systems design the coming development aim in the *Smart Drives Library* project is to include the *Dymola* generated c-code into a hardware environment.

References

[1] H. Elmquist, S. Mattsson, and M. Otter, "Object-oriented and hybrid modeling in modelica," *Dynasim AG, DLR Oberpfaffenhof*, 2001.

[2] C. Kral and A. Haumer, "Modelica libraries for dc machines, three phase and polyphase machines," *Modelica Conference*, pp. 549–558, 2005.

[3] S. Nasar and I. Boldea, *Electric Machines; Dynamics and Control*. 2000 Corporate Blvd., N.W. , Boca Raton, Florida, 33431, USA: CRC Press, Inc., 1st ed., 1993.

[4] D. Simic, C. Kral, and F. Pirker, "Simulation of the cooling circuit with an electrically operated water pump," *IEEE Vehicle Power and Propulsion Conference, VPPC*, 2005.