Tagung "Dynamisches Gesamtsystemverhalten von Fahrzeugantrieben", Haus der Technik Essen, Augsburg, 2007.
Tagungsbuch: "Systemanalyse in der Kfz-Antriebstechnik IV"

Modelling of Vehicle Powertrains with the Modelica PowerTrain Library

Jakub Tobolář
Martin Otter
Tilman Bünte

Abstract

Modern powertrains increasingly include mechatronic components. Moreover, the correlation with vehicle dynamics and comfort is significant for powertrain development. Therefore, a holistic i.e. multiphysics approach is essential for the dynamic analysis in the design process. Hence, the multidisciplinary object oriented modelling language Modelica provides an ideal basis for simulations. This article describes both the basics and some application examples of powertrain modelling using the Modelica ”*PowerTrain*” library. Amongst others, it comprises task-specific driver models, efficiency considerations, and 3D effects such as gyroscopic phenomena. Finally, results of some powertrain application example simulations will be shown.

Zusammenfassung

Moderne Antriebsstränge beeinhalten zunehmend mechatronische Komponenten, zudem ist für die Entwicklung die Wechselwirkung mit Fahrdynamik und Komfort von großer Bedeutung. Daher ist die ganzheitliche, d.h. eine multiphysikalische Betrachtungsweise bei der Untersuchung des dynamischen Verhaltens im Entwurfsprozess unentbehrlich. Für Simulationen stellt die multidisziplinäre, objektorientierte Modellierungssprache Modelica somit eine ideale Ausgangsbasis dar. Dieser Artikel beschreibt die Grundlagen und verschiedene Anwendungsbeispiele der Modellierung von Antriebssträngen mit der Modelica ”*PowerTrain*” Bibliothek. Hierzu gehören u.a. manöverspezifische Fahrermodelle, Wirkungsgradbetrachtungen oder 3D-(insbes. gyroskopische) Effekte. Auch Ergebnisse von Simulationen der Anwendungsbeispiele werden gezeigt.

1. Introduction

Dealing with the modelling of multiphysical automotive applications, the object-oriented modelling language Modelica is widely used, see e. g. [1, 2, 3, 4]. This modelling language is designed to allow convenient, component-oriented modelling of complex physical systems, e.g., systems containing mechanical, electrical, electronic, hydraulic, thermal, control, electric power or process-oriented subcomponents (see [13]). The free Modelica language, free Modelica libraries and Modelica simulation tools are available, ready-to-use and have been utilised in demanding industrial applications, including hardware-in-the-loop simulations.

Based on the Modelica language, a library called ”*PowerTrain*” has been developed at the German Aerospace Center (DLR), see also [9, 10, 11, 12]. It is useful for the modelling of a wide range of powertrain specific problems including optimisation of switch strategies for automatic transmissions, modelling of gearboxes with speed and torque dependent losses or realtime simulations. The library includes both easy to use and rather sophisticated components to model complete powertrains. As a matter of course, it does not supply components for all specific needs. But, the code of all components is transparent and can be modified or extended by the user.
The following sections make a short introduction to the most important packages and components of the *PowerTrain* library (see Section 2). In addition, the interoperability between different automotive model libraries in terms of the *VehicleInterfaces* library is shown in Section 3. Finally, examples of powertrain modelling are discussed in Section 4.

2. Modelica PowerTrain library

In this section the *PowerTrain* library, its structure, conceptual design and some other features will be shortly introduced.

2.1. Components

For Modelica based modelling and simulation of vehicle powertrains, Modelica Standard Library [13] is used utilising mechanical, electrical, electronic and hydraulic elements. Moreover, to facilitate powertrain specific modelling the *PowerTrain* library contains many particular components. Some of the common components are described in more detail in the following.
Especially for manual and automatic transmission models laminar clutches and free wheels are implemented and summarised in the *PowerTrain* library package *Clutches*, see Figure 1 below for package overview. With the lamella clutches – optionally with thermal conduction – the input is the contact pressure to engage the clutch. Connecting in series a free wheel and a laminar clutch, the ”*OneWayLaminarClutch*” component can be used for e. g. planetary gearsets.
The *PowerTrain* package *Shafts* contains shaft components necessary to develop the driveline and transmission models either as one-dimensional or multibody elements. Besides the common rigid shaft, the key component required is the flexible shaft, which allows the twisting of a shaft to be modelled. In its simplest form the flexible shaft consists of two rotational inertias connected by a combined linear rotational spring-damper. This shaft can be used to model low frequency effects such as shuffle which typically occurs in the range between 2 and 10 Hz.
Additionally, the flexible shaft can easily be adjusted to model higher frequency effects as it can contain a variable number of elastic and inertia components evenly distributed across this element.
The mounts typically used to suspend the powertrain within the vehicle chassis are designed using the *MountingSystems* package. The reaction forces in the x, y and z directions are introduced but they leave the powertrain free to rotate. Both linear and nonlinear characteristics are applicable.
For a gearset or a differential gear modelling, basic gear components – included in the package *Gears* – are available; e. g. the two components ”*PlanetPlanet*” and ”*PlanetRing*” enable any type of planetary gearbox to be constructed. With most of the gear elements

both the torque dependent losses as well as mesh losses (gear tooth contact losses) are taken into account. An example is shown in Figure 2 where a Wolfrom type planetary gear with losses is constructed with the *PlanetPlanet* and *PlanetRing* components.
The overall gear ratio and efficiency of a planetary gearbox constructed using these basic lossy elements can be calculated with the example shown in Figure 3. Provided that the number of teeth on each of the gearwheels and the efficiencies of each mesh are known the overall gear efficiency can be calculated.
It would not be possible to determine this value using a static model where the gear shafts are not rotating. This is because the friction between the teeth would be in the stuck mode and the friction torques are then *computed* implicitly from the requirement that the shaft accelerations are zero. This is correctly described by the presented lossy model.

2.2. Incorporation of 3D effects

In [11], a concept for reproducing three-dimensional (3D) mechanical effects of one-dimensionally (1D) modelled powertrains was presented. The idea is to model transmission elements with their mostly 1D rotating behaviour in a convenient way with 1D model components. Due to the simplicity of the 1D equations, this results in a very efficient simulation code. When these 1D components are mounted on systems moving in 3D space

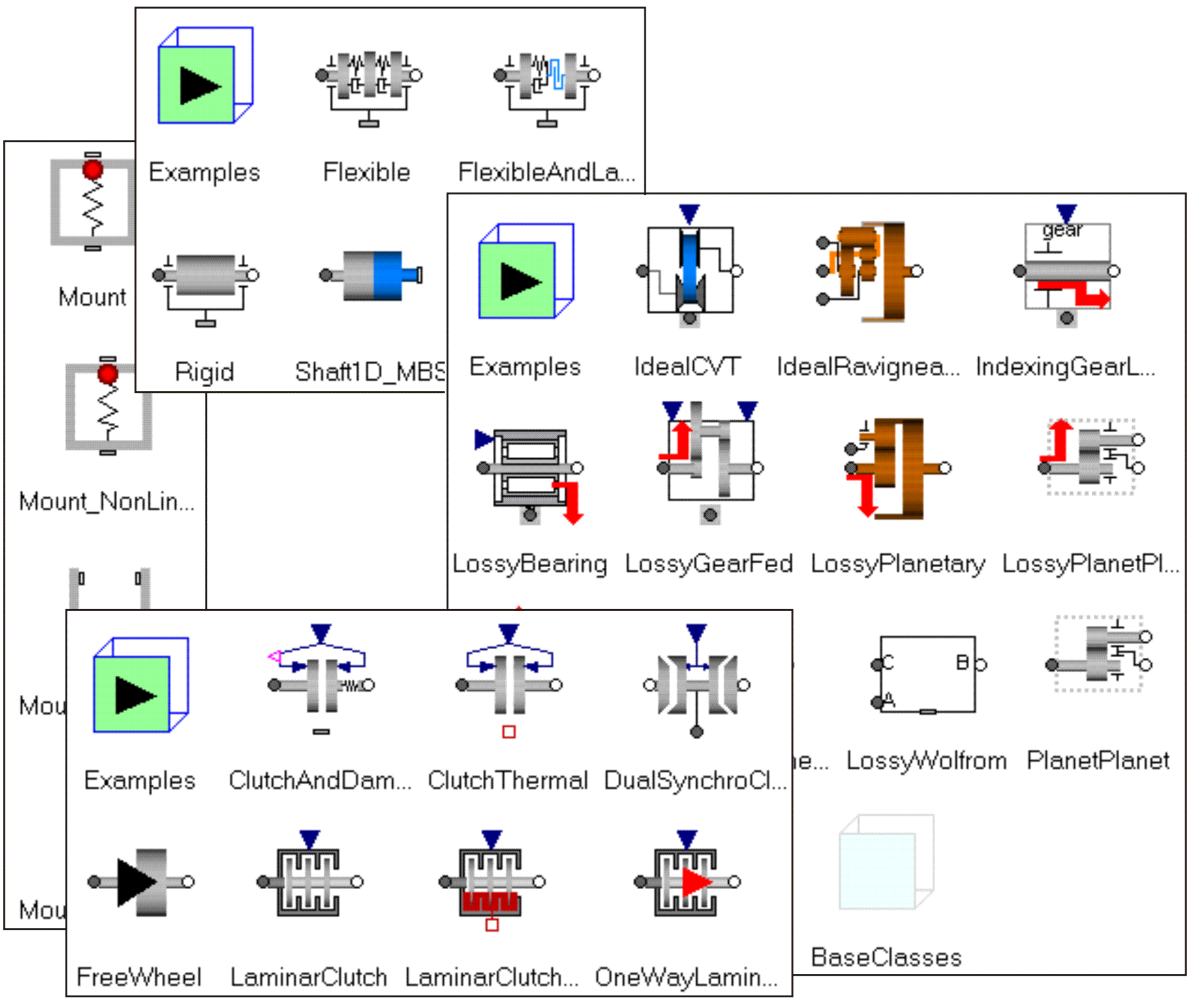

Figure 1: Overview of some component packages of the PowerTrain library. Clockwise from the top, *Shafts*, *Gears*, *Clutches*, *MountingSystems*.

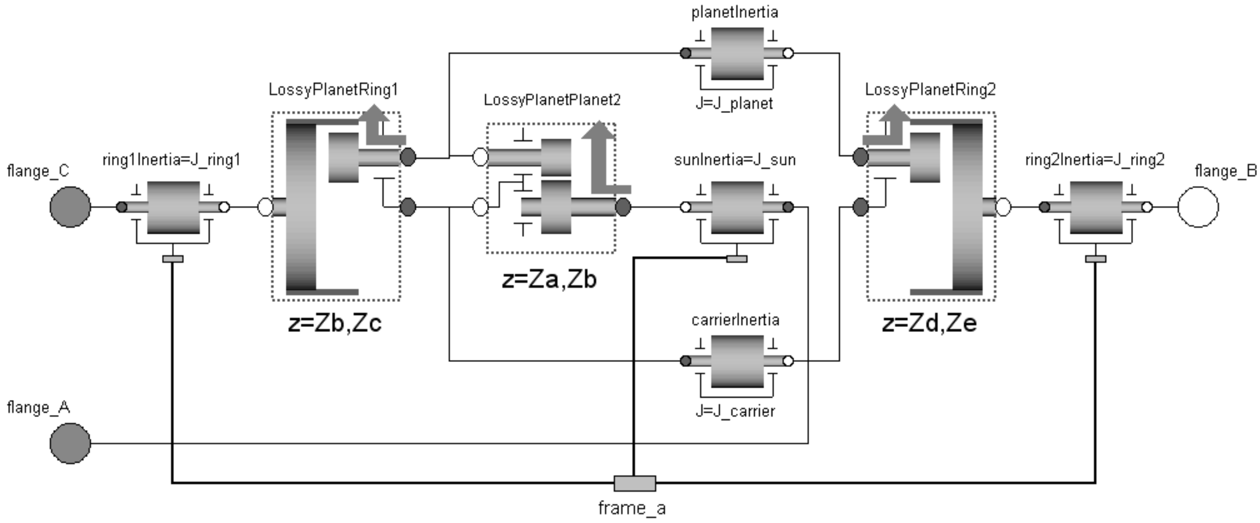

Figure 2: Object diagram of a Wolfrom type planetary gearbox with losses implemented using the improved *PlanetPlanet* and *PlanetRing* components.

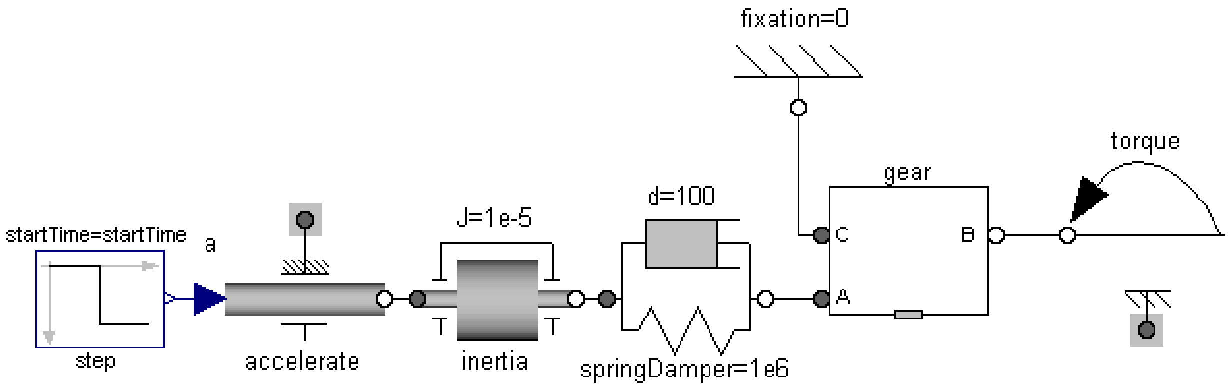

Figure 3: Object diagram of test model to determine gear ratio and gear efficiency between flanges A and B of a Wolfrom planetary gearbox.

a number of important effects, such as support torques and gyroscopic torques, are missing. By including a specific adaptor model and a 3D inertia component it is possible to incorporate these missing effects.
These 3D effects are now incorporated in the *PowerTrain* library. By default, 3D effects are turned off to get fast simulations, which is especially important for real-time purposes [7, 12]. Using the new Modelica feature *conditional declarations*, the 3D effects can be turned on if needed. The idea of conditional declarations in Modelica is that the components concerning 3D effects are instantiated only if required. Otherwise these components are not instantiated and the connect statements referring to them are ignored. The advantage is that the equations of the disabled components are removed from the model and from the generated code, leading to more efficient simulations. The described approach is used everywhere in the *PowerTrain* library where 3D effects may be relevant.

2.3. Assemblies

In the *PowerTrain* library, there are introduced assemblies to support the user when modelling overall powertrains together with a vehicle in the context of vehicle architectures (see also Section 2.5). The packages containing assemblies, such as engines, transmissions, drivelines etc., are built up consequently using the same structure. Basically, every assembly package contains template models – i. e. assembly models of different level of detail and for diverse purposes. For example, the *Engines* package contains simple models just providing a driving torque as a function of engine speed and throttle position as well as more complicated models which include friction, heat losses and fuel flow. The latter model is used e. g. for fuel consumption calculation.
Inherited from the template models, various meaningfully parametrised models of realistic assemblies are grouped in a subpackage called *ParametrizedModels*. Parametrised models of all assemblies are then used in an overall vehicle architecture thus representing a particular vehicle model.
Besides the template and parametrised assembly models, the assembly packages contain *Controllers* and *Components* subpackages which include all the assembly specific submodels. Finally, to check the functionality of assembly models, the package *Examples* is available which includes different simulatable demonstration examples. The palette of preconfigured assemblies provides rather detailed models of engine accessories and wheel brakes, 4- and 6-gear automatic transmissions as well as a three-shaft 6-gear manual transmision, different chassis models and a number of tyre models for longitudinal slip.
To fulfil requirements on the modelling of a growing number of vehicles with all-wheel drive, e.g. sports utility vehicles (SUV) or commercial vehicles, there is a wide range of predefined driveline assemblies in the *PowerTrain* library. Necessary components have been included to enable the modelling of the most common driveline types and some of the most advanced. Available models include e. g. simple open differentials, viscous differentials, and torsen or torque vectoring differentials. As an example, the diagram of an all-wheel drive model is shown in Figure 4. Moreover, a multibody differential gear model has been introduced to calculate the support torques of the driveline at the differential mount points.

2.4. Driver models

The palette of driver models provided with the *PowerTrain* library cover a wider range of tests. Besides the driver models provided to perform drive cycles, there are also other models designed to carry out performance tests and driveability tests. There are variants available for use with both manual and automatic gearboxes.
The driver models performing drive cycles for fuel consumption calculation are based on a situation specific PI controller that actuates either the brake or accelerator pedal to control the vehicle speed so that it follows a defined speed-time profile. A number of drive cycles are included such as the NEDC, EPA City, and Highway cycles. It is also possible to specify user-defined drive cycles for use with the driver model. The cycle driver model also controls the clutch pedal and gear lever in the case of manual transmissions. The gear shifts are usually defined in the drive cycle to occur at particular points in time and the driver starts to change gears at these points.
The driveability driver models are used to perform tip-in and tip-out tests at fixed gears, or in fixed gearbox mode in the case of automatic transmissions, respectively. The tests start with the driver controlling the vehicle speed to an initial value and then decelerating

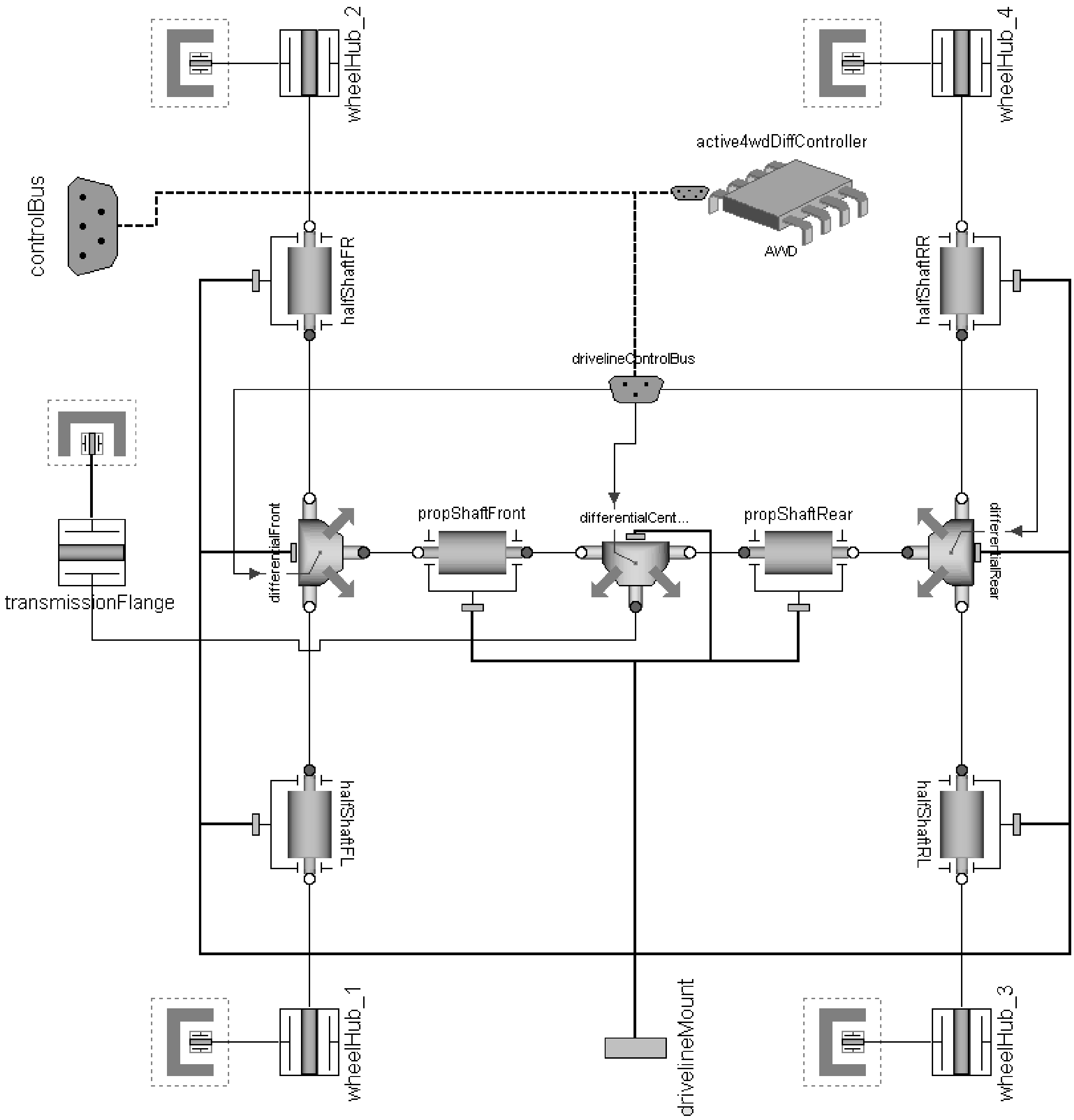

Figure 4: 3D driveline model with active differentials available in the *PowerTrain* library.

and accelerating the vehicle between defined speeds while using only the throttle, not the brakes.
The performance driver is used to perform standing start acceleration tests. The version used with automatic transmissions can perform both an idle start or stall start acceleration test. In both versions the accelerator pedal position for the acceleration test can be defined so it is possible to assess the part-throttle acceleration performance as well as the wide open throttle (WOT) performance. The performance driver model used with manual transmissions will change gear when a defined engine speed is reached.

2.5. Vehicle architectures

As base platforms for different powertrain analyses two standard vehicle architectures have been defined – one for automatic, one for manual transmissions. Both architectures are intended especially for scenarios involving the longitudinal dynamics of a vehicle.

They consist of the following assemblies: engine accessories, combustion engine, transmission, driveline, chassis including wheels, brake assemblies and driver model.
The first architecture is designed to be used for vehicle models with automatic transmissions. Contrary, the second one assumes manual transmission as well as appropriate driver model (see Section 2.4).
Utilising the predefined vehicle architectures, any user specified powertrain model can be created in a very efficient way, since within a vehicle architecture it is just necessary to redeclare the default assemblies by the user-defined parametrised assemblies, see also Section 2.3 for details. Therefore, the modelling approach is similar to the design of an entire vehicle – for a vehicle model different assembly variants can be used (e. g. petrol or diesel combustion engines of different cubic capacity and power).
Both architectures can be used for different purposes as demonstrated in a variety of demo examples delivered with the *PowerTrain* library. Besides the two predefined vehicle architectures, of course any arbitrary architecture can be defined by the user.

3. Interaction with other Modelica libraries

As a common effort of multiple developers of Modelica automotive libraries, a general architecture for the modelling of vehicles was created with the scope to enable the interoperability of developed libraries. The resulting Modelica library called "*VehicleInterfaces*" was presented in [6].
The development focused on standardising the assemblies interface definitions without enforcing a standard vehicle model architecture, so that the same assembly models can be reused in different model architectures. For example, the chassis assembly uses the same interface definition regardless of it being a basic 1D longitudinal model or a complex multibody vehicle dynamics model.
The resulting interface definitions are utilised in commercial automotive libraries such as *SmartElectricDrives* [8], *Transmission* [5] or *VehicleDynamics* [1, 2]. In order to promote easy interoperability with the aforementioned libraries, the *VehicleInterfaces* standards have also been adopted in the *PowerTrain* library. In Figure 5 an example model following *VehicleInterfaces* is shown. Moreover, the new Modelica based library called *Alterna-*

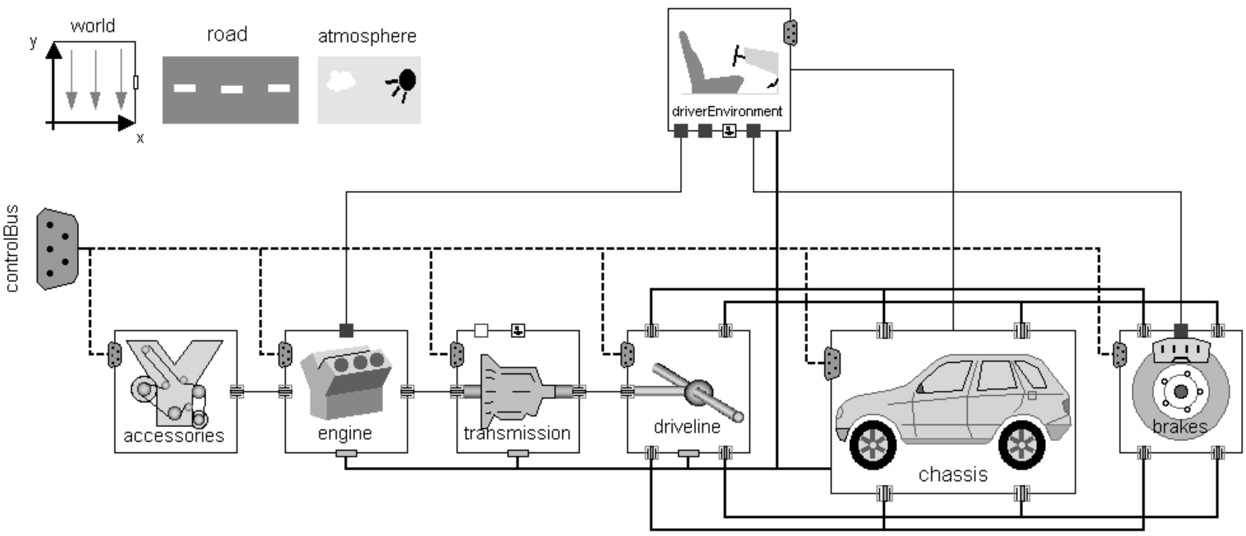

Figure 5: A typical vehicle model based on *PowerTrain* and *VehicleInterfaces* libraries.

tiveVehicles is currently developed at the DLR. This library also complies with the *VehicleInterfaces* and is intended to be used for simulations on hybrid or fuel-cell vehicles, see also [4] and an application example in Section 4.3.

4. Examples

The functionality of the *PowerTrain* library is demonstrated by means of two examples. The first one shows a typical use of the predefined vehicle architecture. In the second example a simple all-wheel drive model with controlled active differentials is analysed. The last example focuses on the utilisation of the *PowerTrain* library while modelling a hybrid vehicle.

4.1. Car performance simulation

In the first application example of the *PowerTrain* library, a vehicle model with an automatic transmission is discussed which is suitable for carrying out standing start car performance work. Since the model is mainly used for predicting the acceleration time, assembly models of different appropriate level of detail are employed.
The combustion engine assembly contains an engine model which is based on steady-state engine maps. The engine accessories such as power steering pump or alternator are simply modelled in terms of their effective inertia. No losses are considered.
The transmission is a detailed model of controlled automatic transmission with six gears and incorporates a torque converter with a lock-up clutch. The gearbox itself is of Lepelletier type, which provides six different gear ratios. It is modelled using base gearboxes, inertias and different clutches and brakes. The different gear ratios are a result of applying different pressures to the clutches and brakes, thus engaging and disengaging them. The control system determines the shift point based on throttle position and vehicle speed when compared to the defined shift map.
The driveline model is for a rear-wheel drive vehicle and is essentially a model with no compliance in the drive shafts. The vehicle itself is modelled as a lumped mass and the resistance forces associated with the vehicle are modelled as different physical effects. Tyre slip is included using the Pacejka (Magic Formula) tyre model. The driver model is formed by an open-loop controller. The level of detail of the whole model is sufficient for the car performance analysis which this model has been developed for.
In the simulation, the performance test starts at 5 s where the driver applies full throttle. The transmission gear shift is dependant on the angular velocity of the transmission output shaft. The simulation stops at 40 s.
The corresponding results are shown in Figure 6. In the upper part of the figure the vehicle velocity is displayed. In the lower part, the current gear is shown.

4.2. Active controlled all-wheel drive model

The driveline model for a four-wheel drive vehicle with three active differentials is discussed here. The model includes propshaft and halfshaft inertias. The losses in each differential are taken into account as well. See Figure 4 for the diagram of a driveline which may be a part of a vehicle model or thought to be installed on a test rig.
During a test rig simulation which lasts 20 s, the drive torques applied on the wheels are changing whereas the input shaft of the centre differential is driven with constant velocity.

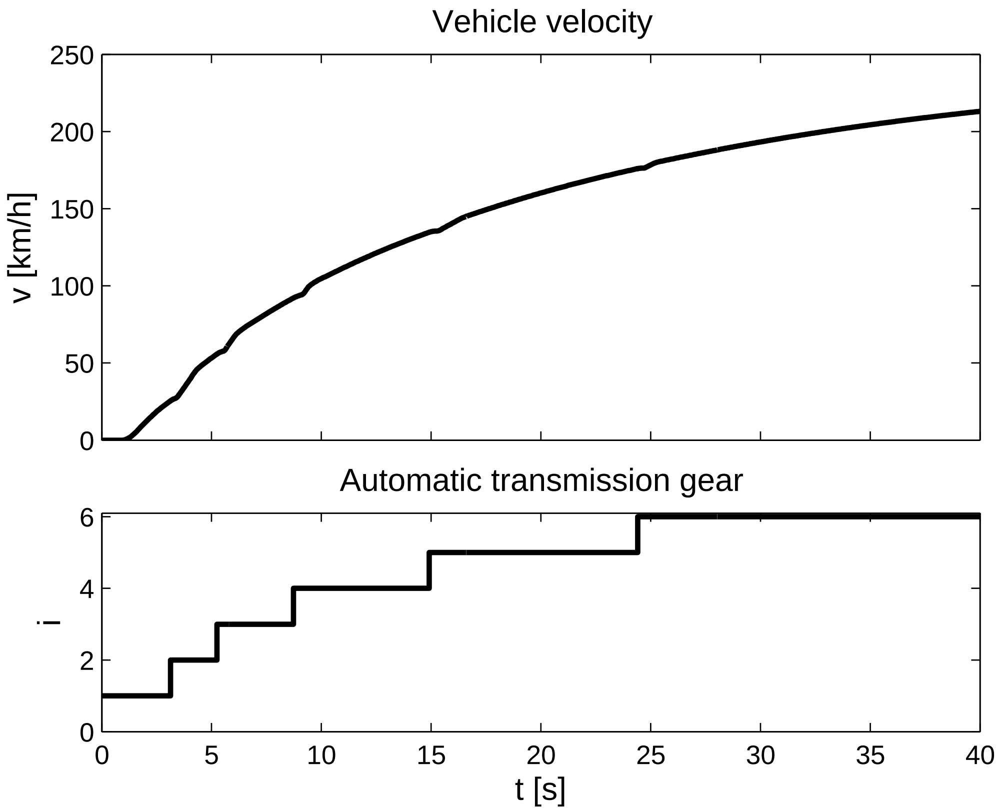

Figure 6: Simulation results of standing start car performance work.

Each of the three active differentials is controlled in such a way that the allowed slip between output shafts does not exceed 5 rad/s. The resulting velocities of wheels as well as the drive torques are displayed in Figure 7.

4.3. Parallel hybrid electric vehicle

In Figure 8 the model structure of a parallel hybrid electric vehicle (PHEV) is shown as a typical example of an alternative vehicle. Hybrid-electric vehicles combine the benefits of combustion engines and electric motors to improve fuel economy and reduce emissions or to supply additional torque and extended performance.
The model is constructed as a full hybrid vehicle, i. e. it can operate on the electric motor alone, the internal combustion alone or both together. Furthermore, it shows the ability to convert and store energy in a battery with regenerative braking.
The model partly consists of assemblies and components from the *PowerTrain* library such as combustion engine or transmission and has been completed by components from the *AlternativeVehicles* library (compare Section 3) for additional devices like battery, electric drive and a special control module to handle the overall strategy in dependence on the different operating conditions.
In the simulation scenario, the New European Drive Cycle (NEDC, lasting 1200 s) is to be followed by the driver. Figure 9 shows a period from the driving cycle simulation results. In the upper graph of Figure 9, the resulting vehicle velocity is presented. Below 8 m/s the vehicle is exclusively propelled by the electric drive. Hybrid propulsion is between 8 m/s and 20 m/s. Above 20 m/s, the vehicle operates on the internal combustion engine alone, whereas deceleration is performed exploiting regenerative braking.

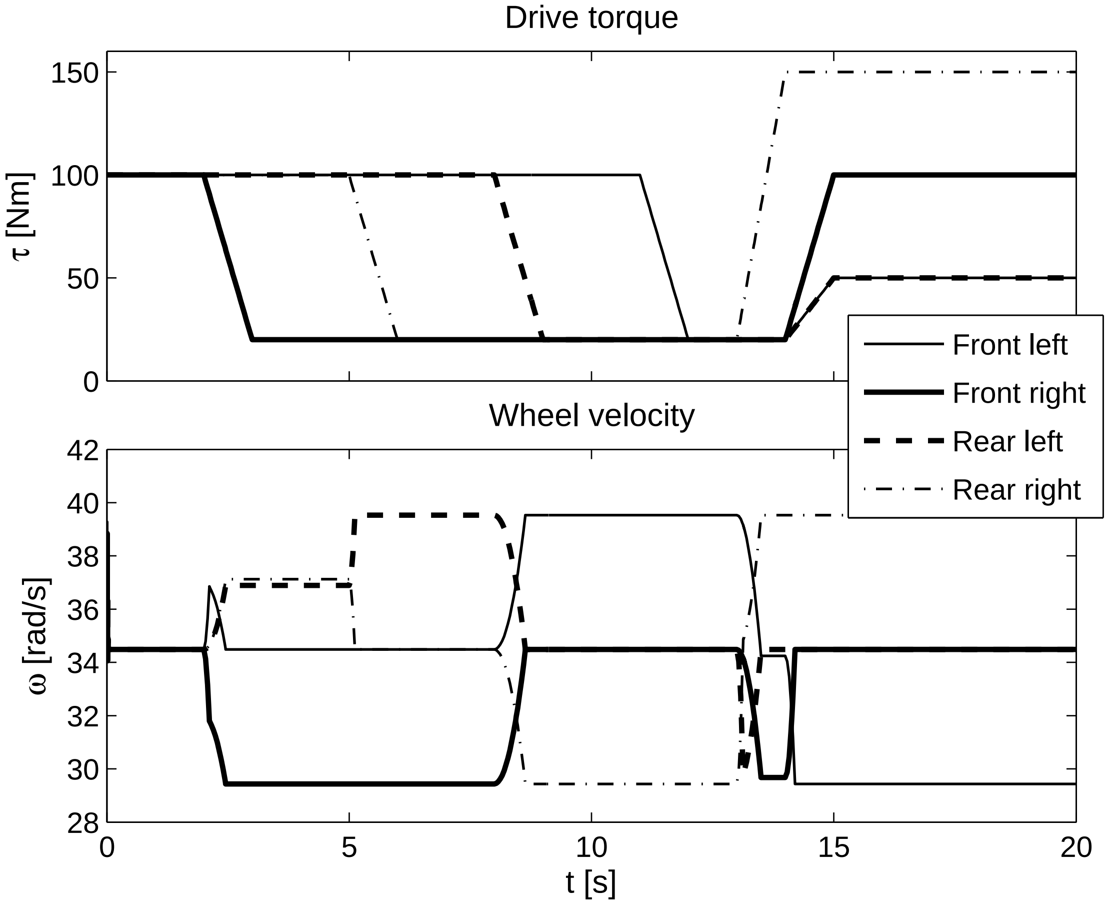

Figure 7: Test rig simulation: Drive torques (above) and wheel velocities (below) of a driveline of an all-wheel drive with controlled active differentials.

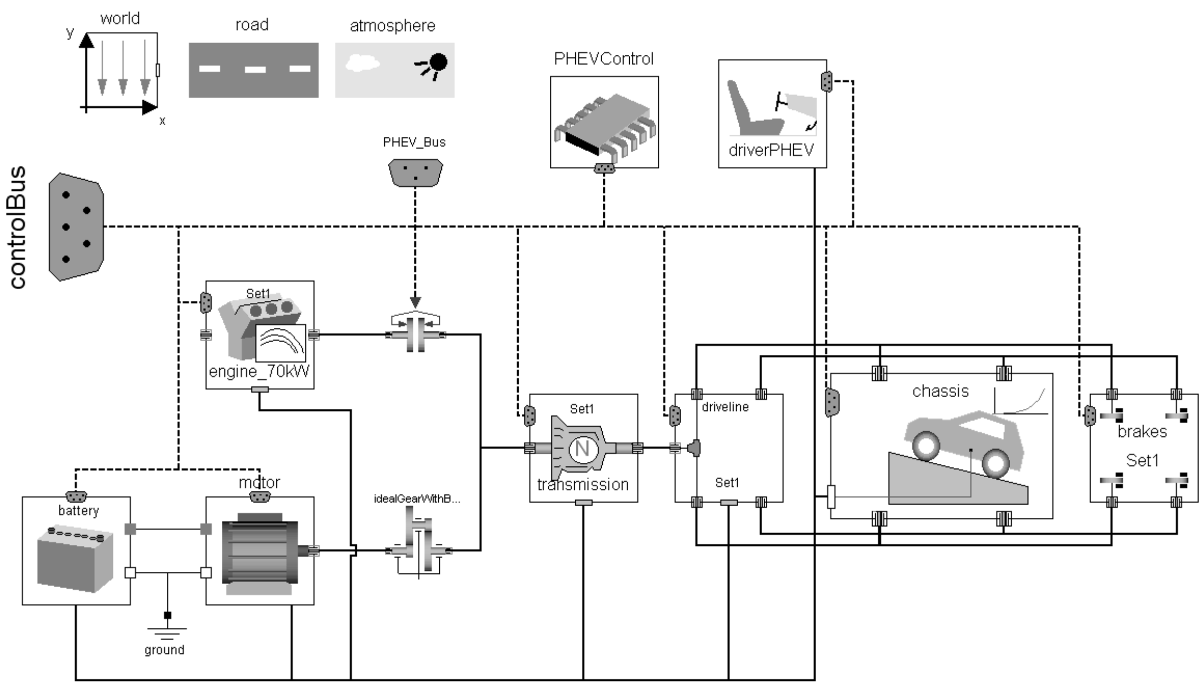

Figure 8: Model structure of a parallel hybrid electric vehicle.

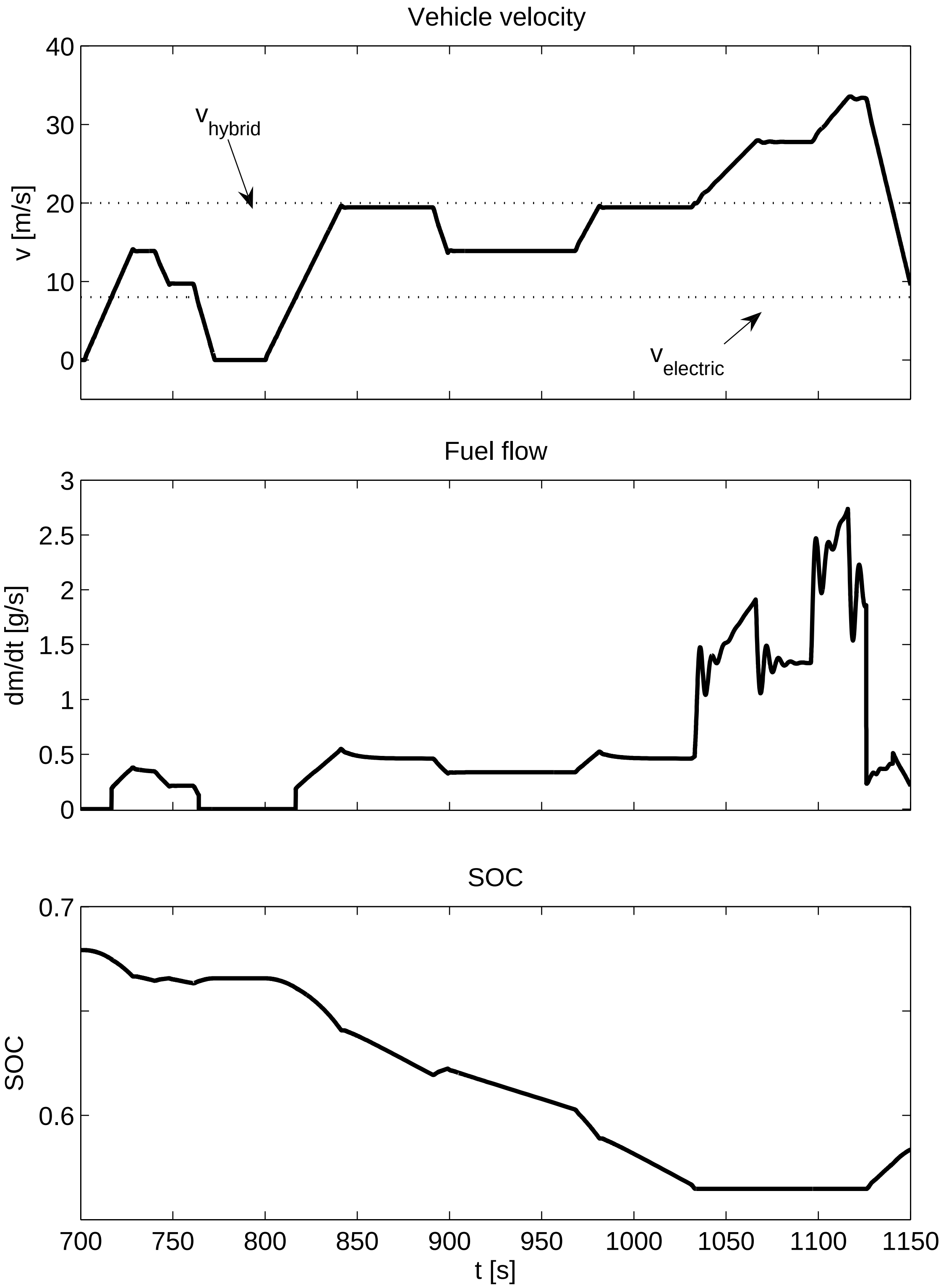

Figure 9: Results of fuel consumption simulation of PHEV. Vehicle velocity (above), fuel flow (middle) and state of battery charge (below) in time range between 700 s and 1150 s.

During halts and periods of electrical driving no fuel is consumed (middle graph of Figure 9). When the vehicle stands idle the state of charge (SOC) of the battery is constant, whereas it decreases when the vehicle is electrically accelerated and it increases through regenerative braking during deceleration, respectively – see lower graph of Figure 9.

5. Conclusions

The paper has described typical applications of modelling and simulation of powertrains utilising a multidisciplinary modelling approach. The examples given demonstrate the capabilities of the Modelica *PowerTrain* library. The models presented include a simple test benchmark of an all-wheel drive driveline with controlled active differentials as well as overall vehicle models with engine, transmission and other powertrain relevant assemblies. Additionally, the vehicle architecture model of a parallel hybrid electric vehicle for fuel consumption simulation shows the interplay of different Modelica based automotive libraries.

Acknowledgements

The authors would like to thank Christian Schweiger (formerly DLR) and Mike Dempsey (Claytex Services Ltd.), who significantly contributed to the development of the *PowerTrain* 2.0 version.
The development of *PowerTrain* 1.0 was partly supported by the *Bayerisches Staatsministerium für Wirtschaft, Verkehr und Technologie* under contract 300-3245.2-3/01 for the project *Test und Optimierung elektronischer Fahrzeug-Steuergeräte mit Hardware-in-the-Loop-Simulation*.

References

1. J. Andreasson, *VehicleDynamics Library*, in Proceedings of the 3rd International Modelica Conference, P. Fritzson, ed., Linköping, November 2003, The Modelica Association and Linköping University.

2. J. Andreasson and M. Gäfvert, *The VehicleDynamics Library – Overview and Applications*, in Proceedings of the 5th International Modelica Conference, Vienna, September 2006, The Modelica Association and arsenal research, pp. 43–51.

3. J. J. Batteh and P. J. Kenny, *Modeling the Dynamics of Vehicle Fuel Systems*, in Proceedings of the 5th International Modelica Conference, Vienna, September 2006, The Modelica Association and arsenal research, pp. 147–155.

4. M. Baur, J. Ungethüm, P. Treffinger, M. Otter, and C. Schweiger, *Alternative Vehicles - Modelica Bibliothek zur Simulation von alternativen Fahrzeugkonzepten*, in Steuerung und Regelung von Fahrzeugen und Motoren – AUTOREG 2006, Wiesloch, Germany, 2006. VDI/VDE-GMA, ed., no. 1931 in VDI-Berichte.

5. F. Brandao and P. Harman, *An integrated simulation approach: Ricardo Transmission and Driveline Dynamic Simulation Library*, in IMechE Integrated Powertrain and Driveline Systems, 2006.

6. M. Dempsey, M. Gäfvert, P. Harman, C. Kral, M. Otter, and P. Treffinger, *Coordinated automotive libraries for vehicle system modelling*, in Proceedings of the 5th Interna-

tional Modelica Conference, Vienna, September 2006, The Modelica Association and arsenal research, pp. 33–41.

7. H. Elmqvist, S. E. Mattsson, H. Olsson, J. Andreasson, M. Otter, C. Schweiger, and D. Brück, *Realtime Simulation of Detailed Vehicle and Powertrain Dynamics*, in Electronics Simulation and Optimization (SAE 2004 World Congress), Detroit, USA, March 2004, pp. 8–11. SAE International Document Number: 2004-01-0768.

8. J. V. Gragger, H. Giuliani, C. Kral, T. Bäuml, H. Kapeller, and F. Pirker, *The SmartElectricDrives Library – Powerful Models for Fast Simulations of Electric Drives*, in Proceedings of the 5th International Modelica Conference, Vienna, September 2006, The Modelica Association and arsenal research, pp. 571–578.

9. M. Otter, M. Dempsey, and C. Schlegel, *Package PowerTrain. A Modelica library for modeling and simulation of vehicle power trains*, in Proceedings of the Modelica Workshop, Lund, Sweden, October 2000, pp. 23–32.

10. C. Schweiger, M. Dempsey, and M. Otter, *The PowerTrain Library: New Concepts and New Fields of Application*, in Proceedings of the 4th International Modelica Conference, Hamburg–Harburg, March 2005, The Modelica Association and Hamburg University of Technology, pp. 457–466.

11. C. Schweiger and M. Otter, *Modeling 3D Mechanical Effects of 1D Powertrains*, in Proceedings of the 3rd International Modelica Conference, P. Fritzson, ed., Linköping, November 2003, The Modelica Association and Linköping University, pp. 149–158.

12. C. Schweiger, M. Otter, and G. Cimander, *Objektorientierte Modellierung mit Modelica zur Echtzeitsimulation und Optimierung von Antriebssträngen*, in Steuerung und Regelung von Fahrzeugen und Motoren – AUTOREG 2004, Düsseldorf, Germany, 2004, pp. 639–650. VDI/VDE-GMA, ed., no. 1828 in VDI-Berichte.

13. *Modelica*. http://www.Modelica.org.

Ing. Jakub Tobolář, Ph.D., DLR, Institut für Robotik und Mechatronik, Oberpfaffenhofen
Prof. Dr.-Ing. Martin Otter, DLR, Institut für Robotik und Mechatronik, Oberpfaffenhofen
Dr.-Ing. Tilman Bünte, DLR, Institut für Robotik und Mechatronik, Oberpfaffenhofen