M. Otter, M. Dempsey, C. Schlegel:
Package PowerTrain. A Modelica library for modeling and simulation of vehicle power trains.
Modelica Workshop 2000 Proceedings, pp. 23-32.

Package PowerTrain: A Modelica library for modeling and simulation of vehicle power trains

Martin Otter, DLR, Germany (martin.otter@dlr.de)
Mike Dempsey, Claytex Services Ltd., U.K. (mike.dempsey@claytex.co.uk)
Clemens Schlegel, Schlegel Simulation GmbH, Germany (cs@schlegel-simulation.de)

Abstract

In this article a new Modelica library to model vehicle power trains is discussed. An overview of the elements within the library is given, such as planetary gearsets, laminar clutches, torque losses, driver, warm up model. The library is demonstrated through the discussion of two illustrative examples, for drive cycle fuel consumption prediction and for investigating gearshift transition dynamics of an automatic gearbox. Together with the Modelica simulation environment Dymola, this library is suited and used for HIL (Hardware-In-the-Loop) simulations.

1 Introduction

Vehicle power trains are 1-dimensional, rotational mechanical systems. They transmit mechanical energy between an engine and the wheels and provide a translation of engine torque and speed. Traditional methodologies for modelling drivelines are described in [5, 7]. In this article a new Modelica library for modelling vehicle power trains is discussed which offers significant advantages compared to traditional methods.

From a modellers point of view, the PowerTrain library window is a set of icons representing the components of a power train, such as engine, torque converter, automatic gearbox, etc. The icons are dragged and moved to a model window and are connected together according to the physical connections. The library together with an appropriate Modelica tool such as Dymola [2] takes care of all complications which may arise in the model, such as inertias directly coupled via an ideal gearbox (which leads to a reduction of the number of states) or locking and unlocking of frictional elements such as bearing friction, clutches or brakes.

The actual implementation of all model components is given in the Modelica language. If in an application some variant of a component is needed, users

can simply make a copy of the appropriate element and change the model according to their needs. Furthermore, it is straight forward to realize and incorporate new elements as needed for the application.

Using Dymola as a simulation engine, an overall model is symbolically processed (a) to transform the system into state space form and (b) to enhance the efficiency. The latter is performed by, e.g., transforming algebraic loops into smaller loops via permutation of equations and via "intelligent" variable substitution. If an algebraic loop is linear in the unknowns, a numerical linear solver is used to solve it. If the coefficient matrix of the linear system is not time varying, an LU-decomposition is performed only once before the simulation starts. Other optimizations are done as well, e.g., constant expressions are only computed once and output variables which need not to be computed for the integrator, are only computed at time instants where the value has to be stored. The result of the symbolic processing is stored as a C-function which is compiled and linked to Dymolas simulation environment. Alternatively, it can be incorporated into other environments, such as SIMULINK. From Dymolas point of view, power trains are small systems and the symbolic processing needs usually only seconds.

In the remaining part of this article, an overview of the PowerTrain library is given, some components are explained in more detail and two application examples are discussed.

2 Overview of PowerTrain library

The PowerTrain library contains 1-dimensional rotational mechanical components. It is normally used in combination with the Modelica.Mechanics.Rotational sublibrary of the Modelica standard library. A screen shot of the Rotational library is shown in figure 1. This library contains basic elements, such as a rotational inertia, ideal gearboxes, shaft elasticity, bearing friction, clutches, brakes, and sensors. The Rotational library
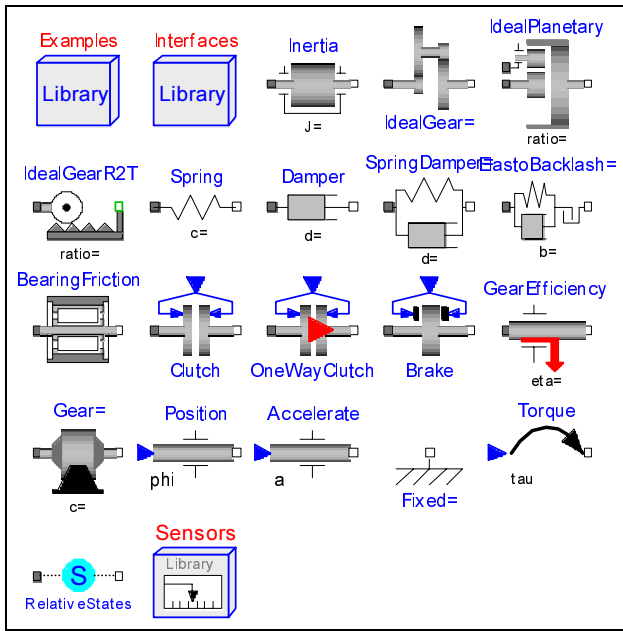
Figure 1: Library Modelica.Mechanics.Rotational.

Figure 2: PowerTrain library.

is part of the standard distribution of a Modelica environment, such as Dymola.

A screenshot of the PowerTrain library is given in figure 2. The following list contains a short description of the most important components of the PowerTrain library. Some of these components are discussed in more detail in subsequent sections:

PlanetPlanet, PlanetRing: Basic building blocks to construct any type of planetary gearbox.

Planetary: Planetary gearbox where the planet may have a different rolling radius with respect to the sun and to the ring respectively.

Differential: Differential gearbox.

VariableGear: Simple model of a gearbox where the gear ratio between the driving and driven shaft is determined by an input signal. The ratio applied can be continuously or step changed.

LaminarClutch: Lamella clutch defined with technological parameters where the input is the pressure to engage the clutch.

OneWayLaminarClutch: Series connection of free wheel and laminar clutch.

LockUpTC: Hydrodynamic torque converter, realized with tables, coupled to a friction clutch to allow the torque converter pump and turbine to be directly coupled together.

GearEfficiency2: Gear efficiency where the efficiency depends on the gear which is supplied as an input.

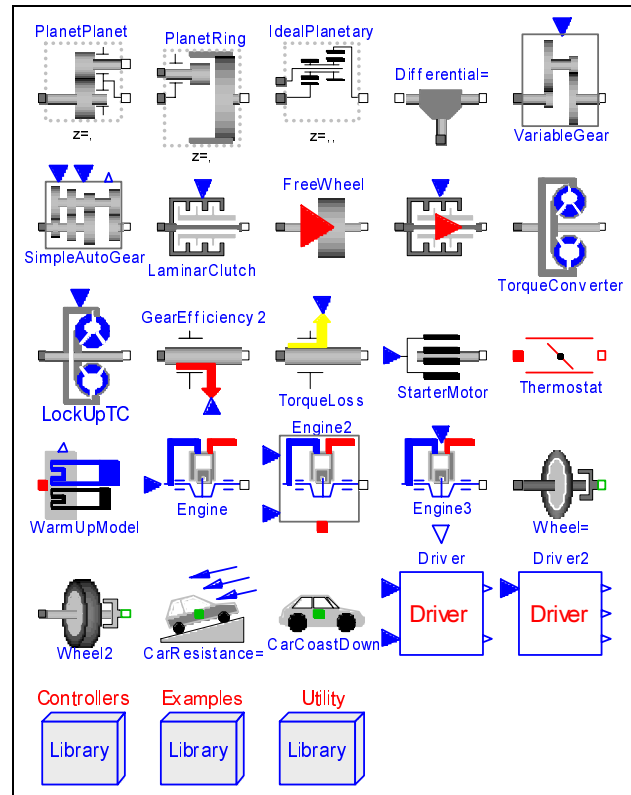*TorqueLoss:* Torque loss element which models the frictional losses in bearings. The friction characteristic is given as input signal to allow any type of functional dependency of the torque loss to be modelled.

WarmUpModel: A simple warm-up model that determines coolant and oil temperature based on a lumped mass system from an energy balance between the engine, block, coolant in the block, coolant in the radiator, oil system and the surrounding atmosphere.

Engine3: An engine model with inertia, warm-up and friction models which is based on steady-state mapped engine data.

Wheel2: A simple model of the wheel where the wheel radius depends on the vehicle speed.

CarResistance, CarCoastDown: Longitudinal dynamics of a vehicle. Energy losses can either be defined by resistance forces of wind and rolling resistances given by tables, or the resistance forces can be given as 2nd order polynomial.
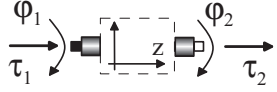
Driver2: A driver model, for use with automatic transmissions, that is based on a PI controller and that is used to determine the throttle and brake pedal positions. The driver will only control one pedal at once.

Controls: Several controllers, such as gearbox, fuel flow, idle controller.

Examples: Example models to demonstrate the usage of the library.

3   Interfaces

The connector interfaces used for 1-dimensional rotational components are *mechanical flanges*:

$$\tau_1 = \tau_1 \cdot \mathbf{e}_z$$
$$\tau_2 = \tau_2 \cdot \mathbf{e}_z$$
$$\omega_1 = \dot{\varphi}_1 \cdot \mathbf{e}_z$$
$$\omega_2 = \dot{\varphi}_2 \cdot \mathbf{e}_z$$

As interface variables the *absolute rotation angle* φ_1, φ_2 and the *cut-torque* τ_1, τ_2 in a flange are used. In every flange a *coordinate system* is defined which is fixed in the flange. The z-axis of this coordinate system is directed into the positive rotation direction. All vectors in the flange are resolved in this flange coordinate system. Usually, in a component the coordinate systems of all flanges of this component are parallel to each other. This is the reason why the cut-torque τ_1 of the left flange in the figure above is directed *in* the left flange and the cut-torque τ_2 of the right flange is directed *out* of the right flange. If two flanges of different components are connected together, this means that these two flanges are rigidly attached and that the two *flange coordinate systems are identical*. In the Rotational library, a 1-dimensional rotational connector is formally defined as:

```
connector Flange_a
   Modelica.SIunits.Angle       phi;
   flow Modelica.SIunits.Torque tau;
end Flange_a;
```

In this library two identical connectors Flange_a and Flange_b are defined. The only difference of these two connectors is the graphical representation (filled and non-filled square) in order that different flanges of a component can be more easily identfied in the graphical view.

4   Ideal planetary gears

There are various kinematical setups of planetary gearboxes. It would therefore be inconvenient to derive the equations for every type of gearbox manually. It is, however, possible to build up any type of planetary gearbox by using two basic components (see the first two rows in table 1): A planet-planet and a planet-ring combination. This is demonstrated by the examples in the two last rows of table 1:
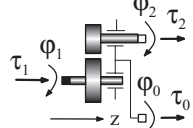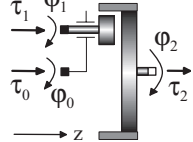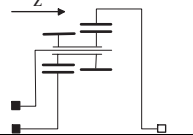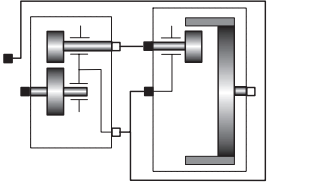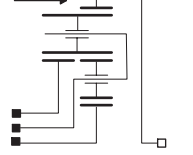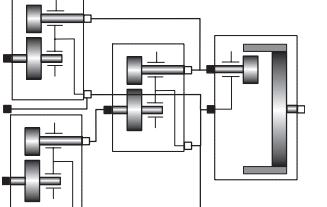
Planet-Planet combination	z_j: Tooth number of wheel j $i_1 = z_1/(z_1 + z_2)$ $i_2 = z_2/(z_1 + z_2)$ $\varphi_0 = i_1 \cdot \varphi_1 + i_2 \cdot \varphi_2$ $0 = \tau_1 + i_1 \cdot \tau_0$ $0 = \tau_2 + i_2 \cdot \tau_0$
Planet-Ring combination	z_j: Tooth number of wheel j $i_1 = z_1/(z_2 - z_1)$ $i_2 = z_2/(z_2 - z_1)$ $\varphi_0 = i_2 \cdot \varphi_2 - i_1 \cdot \varphi_1$ $\tau_1 = i_1 \cdot \tau_0$ $0 = i_2 \cdot \tau_0 + \tau_2$
Planetary gearbox	
Ravigneaux gearbox	

Table 1: Ideal planetary gears.

The standard planetary gearbox, shown in the third row of table 1, is constructed in such a way that the planet may have a different rolling radius with respect to the sun and to the ring wheel, as is common in modern load gearboxes. This corresponds to the case where two planets with different rolling radius are rigidly attached.

The Ravigneaux gearbox consists of 2 sun wheels, 2 planet wheels and one ring wheel (see, e.g., [4]) and is an example how non-standard gearsets can be modelled using the two mentioned basic elements. In the right column of table 1 it is shown, how this gearbox can be modelled by using 3 planet-planet and one planet-ring combination.

The derivation of the equations for the planet-planet combination shall be discussed briefly: The needed auxiliary variables are defined in figure 3. Basically, the well known formula is used, which computes the absolute velocity $\mathbf{v}_P$ of a point P from the absolute velocity $\mathbf{v}_B$ of a reference point B on the same rigid body, from the absolute angular velocity ω of this
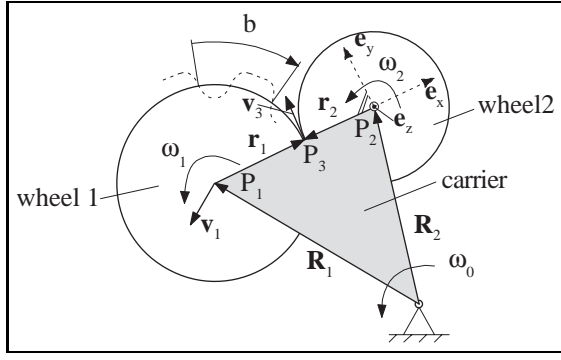
Figure 3: Planet-Planet combination.

body and from the position vector $\mathbf{r}_{BP}$ from B to P:

$$\mathbf{v}_P \;=\; \mathbf{v}_B + \boldsymbol{\omega} \times \mathbf{r}_{BP} \qquad (4.1)$$

Application of (4.1) to the points P_1, P_2, P_3 in figure 3, and utilizing the closed vector chain of the position vectors $\mathbf{R}_1, \mathbf{r}_1, \mathbf{r}_2, \mathbf{R}_2$ leads to:

$$
\begin{aligned}
\mathbf{R}_1 + \mathbf{r}_1 &= \mathbf{R}_2 + \mathbf{r}_2 & (4.2a)\\
\mathbf{v}_1 &= \boldsymbol{\omega}_0 \times \mathbf{R}_1 & (4.2b)\\
\mathbf{v}_2 &= \boldsymbol{\omega}_0 \times \mathbf{R}_2 & (4.2c)\\
\mathbf{v}_3 &= \mathbf{v}_1 + \boldsymbol{\omega}_1 \times \mathbf{r}_1 &\\
&= \mathbf{v}_2 + \boldsymbol{\omega}_2 \times \mathbf{r}_2 & (4.2d)
\end{aligned}
$$

Inserting (4.2a–4.2c) in (4.2d) yields:

$$\boldsymbol{\omega}_0 \times (\mathbf{r}_2 - \mathbf{r}_1) + \boldsymbol{\omega}_1 \times \mathbf{r}_1 \;=\; \boldsymbol{\omega}_2 \times \mathbf{r}_2 \qquad (4.3)$$

When the vectors are resolved in the coordinate system of the carrier (see figure 3), i.e.,

$$
\begin{aligned}
\boldsymbol{\omega}_0 = \dot\varphi_0 \cdot \mathbf{e}_z, \quad \boldsymbol{\omega}_1 = \dot\varphi_1 \cdot \mathbf{e}_z, \quad \boldsymbol{\omega}_2 = \dot\varphi_2 \cdot \mathbf{e}_z, \\
\mathbf{r}_1 = r_1 \cdot \mathbf{e}_x, \qquad \mathbf{r}_2 = r_2 \cdot (-\mathbf{e}_x)
\end{aligned}
$$
$$(4.4)$$

inserted in (4.3) and integrate once, the result is:

$$(r_1 + r_2) \cdot \varphi_0 = r_1 \cdot \varphi_1 + r_2 \cdot \varphi_2 \qquad (4.5)$$

If z_1 is the number of teeth of wheel 1, z_2 the number of teeth of wheel 2 and b the width of one tooth together with the distance to the next tooth along the rolling circle, the girth of the two wheels is given by:

$$2\pi r_1 = z_1 \cdot b, \quad 2\pi r_2 = z_2 \cdot b \qquad (4.6)$$

Solving these two equations for r_1 and r_2, inserting in (4.5) finally yields the desired kinematic relationship between the rotational angles and the teeth numbers:

$$(z_1 + z_2) \cdot \varphi_0 = z_1 \cdot \varphi_1 + z_2 \cdot \varphi_2 \qquad (4.7)$$

Note, that (4.7) is better suited than (4.5), because the teeth number of a given gearbox can be determined

exactly by counting the teeth, where as the radii r_1, r_2 can only be determined up to a certain *tolerance*.

The kinematic relationship (4.7) is part of the model equations of the planet-planet combination, see first row of table 1. The remaining equations can be derived based on conservaton of energy: Since no energy is stored in an ideal planet-planet combination and no energy is transmitted to the environment, the sum of the energy flowing into this component through all connectors has to vanish, i.e.,

$$
\begin{aligned}
0 &= P \;(= dE/dt)\\
&= \dot\varphi_0 \cdot \tau_0 + \dot\varphi_1 \cdot \tau_1 + \dot\varphi_2 \cdot \tau_2\\
&= (i_1\dot\varphi_1 + i_2\dot\varphi_2) \cdot \tau_0 + \dot\varphi_1 \cdot \tau_1 + \dot\varphi_2 \cdot \tau_2\\
&= (\tau_1 + i_1 \cdot \tau_0)\dot\varphi_1 + (\tau_2 + i_1 \cdot \tau_0)\dot\varphi_2 \quad (4.8)
\end{aligned}
$$

The two speeds $\dot\varphi_1, \dot\varphi_2$ can be selected independently from each other. In order that the overal power $P(t)$ always vanishes, the factors in front of $\dot\varphi_1$ and $\dot\varphi_2$ have to vanish, which leads to the equations in the first row of table 1

Note, that the usage of ideal gearboxes, and especially of the planet-planet and planet-ring combination leads always to small algebraic loops. These loops can be treated efficiently with the techniques used in Dymola.

5 Friction and loss elements

Both the Rotational library and the PowerTrain library contain friction elements, such as bearing friction, clutches, brakes, and gear efficiency. All these elements are based on a thorough theory which will be introduced by a brief explanation of a basic Coulomb friction element. For more details, see [6].

Friction may lock the relative motion of two surfaces. This situation may be modeled (a) with stiff springs, which only allow small relative displacements in the stuck mode or (b) in a slightly idealized way as exact locking element which removes one degree of freedom if in stuck mode. It is easy to provide a model for approach (a). However, the corresponding system becomes stiff and is not suited for realtime simulations. Furthermore, it is usually difficult to determine an appropriate spring and damping constant which reflects the reality and does not induce vibrations due to the numerics. For these reasons, in the Rotational and PowerTrain library, approach (b) is used. However, this leads to difficulties if two or more friction elements are dynamically coupled.

The essential idea is discussed using translational friction elements: Here, the friction force f acts between two surfaces and is a function of the relative

velocity v between the friction surfaces when they are sliding relative to each other. When v becomes zero, the two surfaces are stuck to each other and the friction force is no longer a function of v. The element starts sliding again if the friction force becomes larger than the maximum static friction force f_0.

The key to the solution is the observation that $v = 0$ in the stuck mode and when forward sliding starts, but $\dot{v} > 0$ when sliding starts in forward direction and $\dot{v} = 0$ in the stuck mode, see figure 4. Since the friction characteristic in figure 4 at zero velocity is no functional relationship, i.e., a function $f = f(a)$ does not exist, a parameterized curve description

$$a = a(s_a)$$
$$f = f(s_a)$$

with a curve parameter s_a is used leading to the following equations for zero velocity:

$$a = \dot{v}$$
$$a = \textbf{if } s_a > 1 \textbf{ then } s_a - 1 \textbf{ else}$$
$$\textbf{if } s_a < -1 \textbf{ then } s_a + 1 \textbf{ else } 0$$
$$f = \textbf{if } s_a > 1 \textbf{ then } f_0 \textbf{ else}$$
$$\textbf{if } s_a < -1 \textbf{ then } -f_0 \textbf{ else } f_0 \cdot s_a$$

At zero velocity, these equations and the equations of the mechanical elements connected together with the friction element form a mixed continuous/discrete set of equations consisting of Real and Boolean unknowns, (Boolean unknowns are: $s_a > 1, s_a < -1$). This mixed system of equations has to be solved at event instants to determine whether a friction element remains in the stuck mode or starts to slide. Several other minor difficulties occur, which are discussed in more detail in [6].
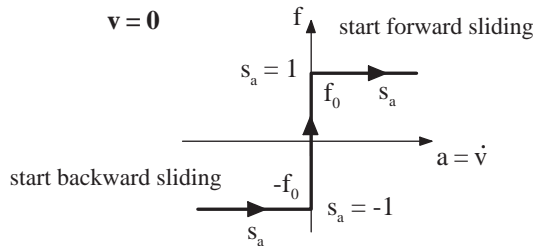

Figure 4: Friction characteristic at $v = 0$.

A simple example of dynamic coupling of friction elements is shown in figure 5 where two blocks are sliding on each other and on every surface friction is present which is described according to the discussed method. At event instants, a mixed continuous/discrete system of equations of the following structure has to be solved:

$$m_1 \dot{v}_1 = u_1 + f_2 - f_1$$
$$m_2 (\dot{v}_1 + \dot{v}_2) = u_2 - f_2$$
$$\dot{v}_1 = g_1 (s_1 > 1, \; s_1 < -1, \; s_1)$$
$$f_1 = g_2 (s_1 > 1, \; s_1 < -1, \; s_1)$$
$$\dot{v}_2 = g_3 (s_2 > 1, \; s_2 < -1, \; s_2)$$
$$f_2 = g_4 (s_2 > 1, \; s_2 < -1, \; s_2)$$

By applying appropriate time varying external forces $u_1(t), u_2(t)$ on the two blocks, a stick-slip like behaviour occurs. For this situation, a simulation was carried out with Dymola [2] leading to the simulation results of figure 6. The discontinuities in the friction forces when switching from stuck to sliding mode are due to the property that the maximum friction force f_{max} in the locked mode is a factor of 1.25 higher as the sliding friction force f_0 at zero velocity.

The process of finding the solution of the mixed system of equations, i.e., to determine a consistent set of relations and continuous variables, can be manually checked for demonstration purposes with the following settings ($m_1 = m_2 = 1, f_{max}/f_0 = 1$):

$$u_1 = 0.9 \cdot f_0$$
$$u_2 = \textbf{if } \text{time} < 0.1 \textbf{ then } 0 \textbf{ else } 1.1 \cdot f_0$$

At the beginning of the simulation the two blocks are stuck. At time = 0.1 s an event occurs and force f_2 jumps from 0 to $1.1 \cdot f_0$. After evaluating all equations under the assumption that both blocks are stuck, it turns out that both friction forces become larger than f_0. Therefore, it is natural to have the assumption that both elements start to slide in forward direction. Re-evaluating the corresponding mixed system of continuous/discrete equations leads to a relative acceleration of block 2 with respect to block 1 which is negative, i.e., block 2 cannot slide in forward direction. The assumption that block 1 slides and block 2 is stuck, finally leads to a consistent configuration.

All friction elements utilized in the Rotational and PowerTrain library are slight variations of the described basic friction element. In all cases, these elements lead to mixed systems of equations if they are
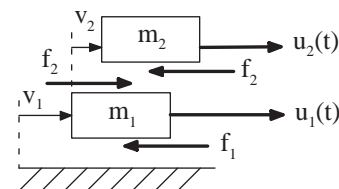
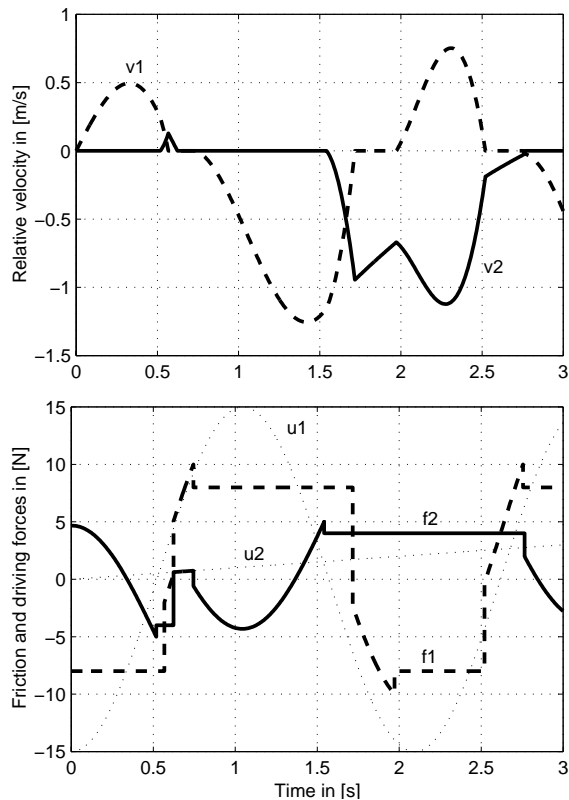

Figure 5: Two blocks with friction.

Figure 6: Simulation results of two block system.

dynamically coupled. Dymola [2] has appropriate numerical algorithms to solve these equations in a reliable and efficient way. For realtime applications, special adaptions of these algorithms are provided.

In the remaining part of this article, it is shown, how the PowerTrain library can be utilized in different applications.

6 Application 1: Fuel consumption

In the first application of the PowerTrain library, a vehicle model with an automatic gearbox is discussed which is suitable for carrying out drive cycle fuel consumption prediction work, see figure 7. Since the model is mainly used for predicting the energy losses in a power train, detailed component models, such as describing the detailed shifting behaviour of the automatic gearbox, are not needed.

The model in figure 7 contains an engine model which is based on steady-state engine maps but includes a simple warm-up model to account for the effects of cold-starts. To achieve this it incorporates a look-up table which determines the increased load that should be applied to the engine due to increased friction when the engine is cold. The additional friction load is dependant upon the predicted oil temperature. Engine auxiliaries such as power steering pump, alter-

nator, and others are lumped into one table as a speed dependant load on the engine. The individual torque loss curves for these components need to be adjusted to reflect the load on the crankshaft before entering them into the model.

The transmission, see lower part of figure 7, is a simplified model of an automatic transmission and incorporates a torque converter with a lock-up clutch. The gearbox itself is modelled as a series of fixed ratios which are jumped between according to the control system. 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 essentially a rigid model with no compliance in the drive shafts and no tyre-slip modelling. The wheel/tyre model does include a speed-dependant rolling radius for the tyre. This level of detail is sufficient for the drive cycle work that this model has been developed for.

The vehicle itself is modelled as a lumped mass and the resistance forces associated with the vehicle are modelled as a second order polynomial dependant upon vehicle speed. This curve can be determined from a vehicle coastdown and accounts for all resistance forces such as aerodynamic, rolling, transmission losses, etc.

The driver model is based around a PI controller which has external controls affecting its performance so that the driver behaviour is realistic. For example, the driver will only control one pedal at a time and will not jump rapidly from throttle to brake pedal. The change in pedal control is influenced purely by the desired speed-time profile.

In figure 8 typical simulation results of this model are shown. In the upper part of the figure the desired and actual speed of the vehicle is displayed. The desired speed is part of the European drive cycle. In the lower part, the consumption of the vehicle (in liter per 100 km) is shown. The consumption computation starts a little bit delayed, because at the beginning the car is not moving which would mean that the consumption in litres per 100 km is infinity.

7 Application 2: Automatic gearbox

In the second application of the PowerTrain library, the detailed gearshift transition dynamics of an automatic gearbox are investigated. For a demonstration of the modelling and simulation methodology using Modelica the automatic gearbox ZF 4HP22 is choosen [4]. A crossectional view of this gearbox is shown in figure 9 and the schematic in the lower part of fig-
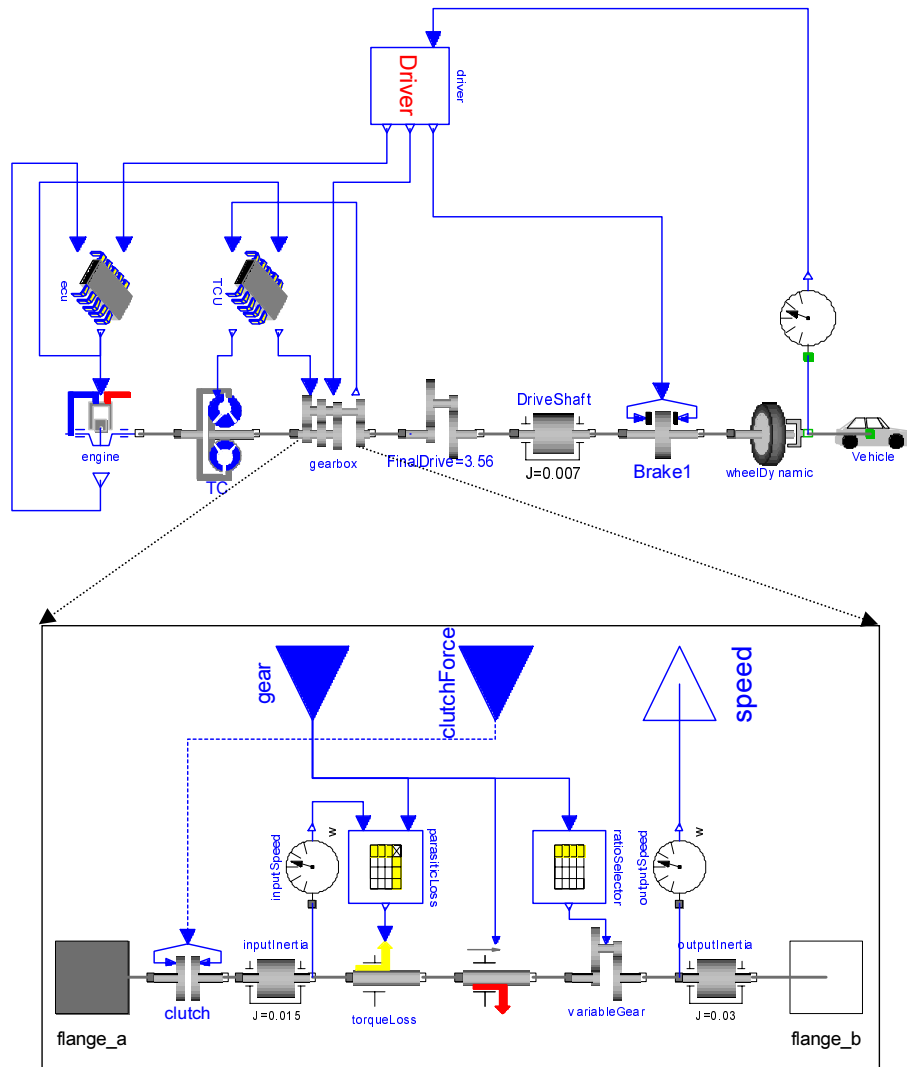
Figure 7: Modelica model of power train for drive cycle fuel consumption prediction.

ure 10. It consists of a hydrodynamic torque converter and three planetary gear sets, which are connected via two clutches, four brakes and two free wheels. These switching elements enable or disable the relative motion between the wheels of the planetary gears in order to achieve the desired transmission ratios. In the upper part of figure 10 the gear shift scheme is shown, describing which clutches have to be enganged or disenganged to activate the desired gear.

The clutches and brakes are controlled by electro-hydraulic actuators which are in turn controlled by an electronic control unit (ECU). The control scheme for the clutches and brakes has to be designed carefully to achieve a good compromise between gearshift comfort, wear and driving performance. Depending on the goals of the actual investigation, the dynamic effects of the hydraulic force generation may be modelled globaly by a first order time delay, or in detail using the Modelica Hydraulics library [1], or by using any particular modelling level.

To simulate the gearshift transitions accurately, the input and output torques acting on the gearbox are required. Therefore the whole driveline of engine and torque converter (for details see [4]), differential gearbox and longitudinal vehicle dynamics (vehicle inertia and driving resistances) has to be included in the model. For offline-simulations the essential functions of the engine and gearbox ECUs have to be modelled too. In a hardware-in-the-loop simulation they are replaced by the actual ECU hardware [8].

These components are shown in the upper part of figure 11 together with a driver model which uses the desired driving maneuvre as input. The model is set up using the PowerTrain library and the Modelica Standard library. In the middle part of the figure, the connection of the switching logic, the clutch control and the gearbox mechanics are shown. Finally, in the lower part of figure 11 the mechanical components of the gearbox, i.e., the 3 planetary gear sets, the clutches, brakes and free wheels can be seen. The switching el-
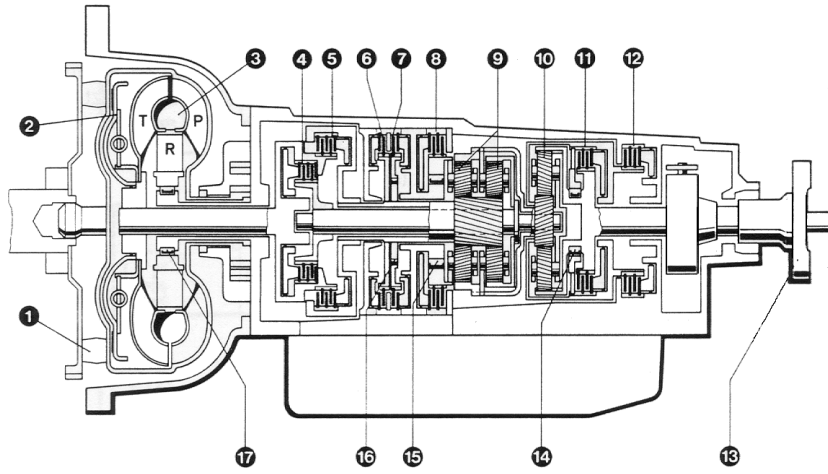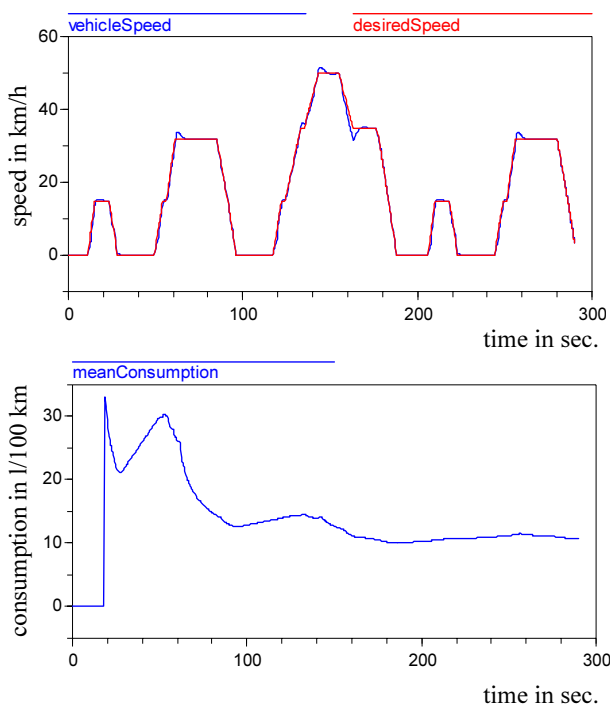
Figure 9: Crossectional view of gearbox ZF 4HP22.

Figure 8: Simulation results of drive cycle model.

reuse of model components in a simulation model of an other gearbox without any modification, whitch is not possible using other modelling methods. Using an appropriate simulation tool [2] the described Modelica gearbox model is also very efficient. Therefore it can be used in realtime applications too [3, 8].

Figure 12 gives some simulation results: On the lefthand side engine respectively torque converter speed and car acceleration during a fullspeed upshift through all four gears are shown. The plots on the righthand side are zoomed in time, showing details of clutch / freewheel C11 opening during upshift from 3rd to 4th gear.

Because clutch C12 starts to close at T=18 the torque transmitted in C11 increases. C11 starts to open at T=18 (control input decreasing), but is still strong enough to stay stuck. At T=18.53 clutch C11 starts to slide (state lC11 changes from 1 to 0). An increasing relative speed can be observed. Clutch C11 can now

ements are controlled by signals which are generated in the clutch control component.

The Modelica model of the automatic gearbox has a variable structure, since the 6 clutches and brakes C_i allow $2^6 = 64$ different structures. Although, basically only the 5 different structures according to the gears from figure 10 occur, during a gear shift every of the other 59 structures may show up, depending on the actual clutch control signals and the clutch actuators performances.

The components of the gearbox as well as the connections of the components are identical in the Modelica model of the gearbox (= lower part of figure 11) and in the gearbox schematic in figure 10. This enables

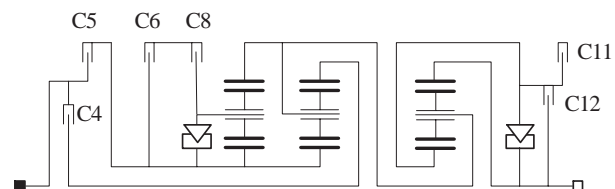| Gear	C4	C5	C6	C8	C11	C12
1	x				x	
2	x		x		x	
3	x	x			x	
4	x	x				x
R		x		x	x	

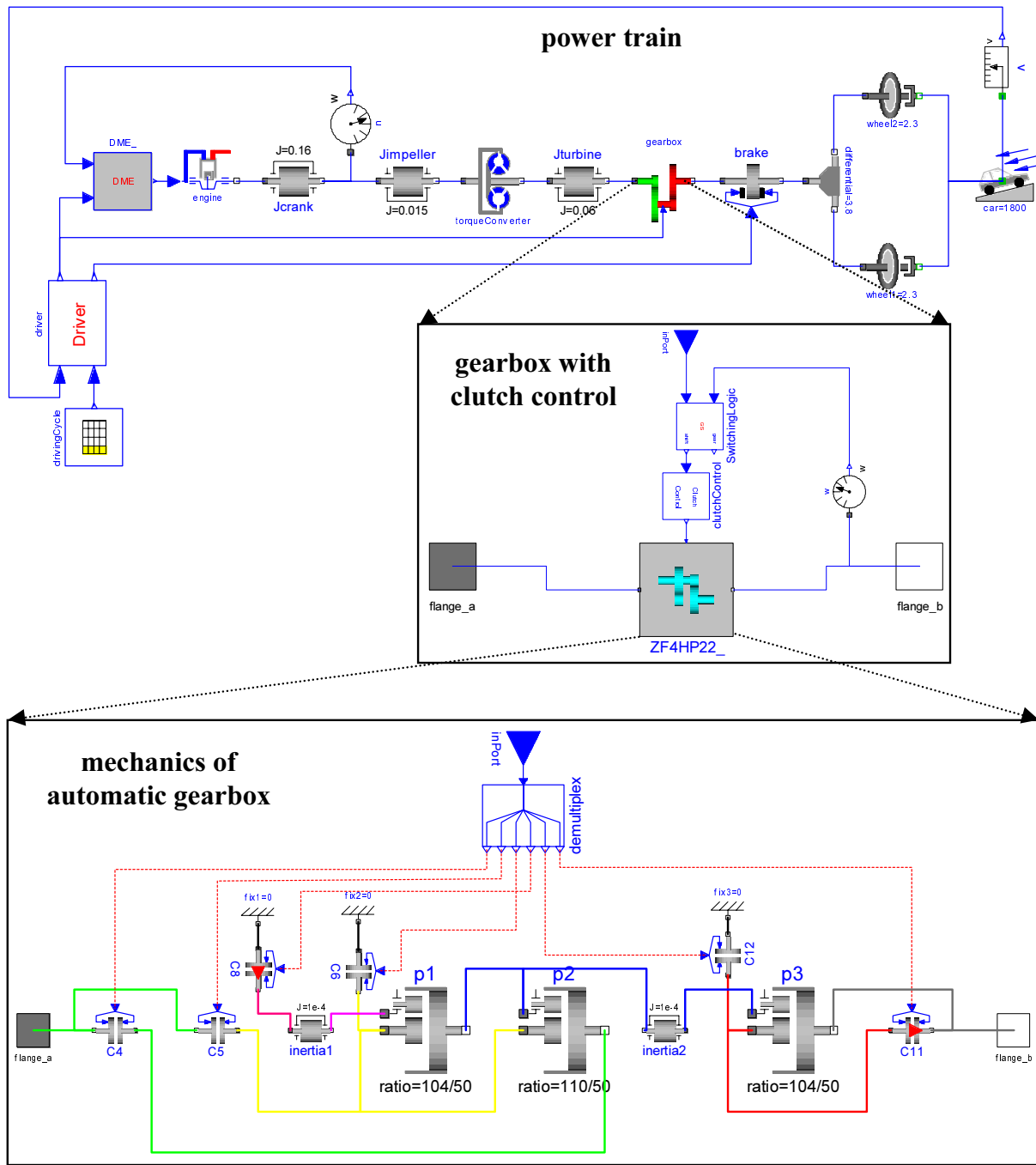Figure 10: Shift scheme of gear ZF 4HP22.

Figure 11: Modelica model of power train with automatic gearbox ZF 4HP22.

no more transmitt full torque, it breaks down a bit and decreases as the clutch is opened more and more. At T=18.65, before the clutch is opened completely, full relative speed between the disks of clutch C11 can be observed. At T=18.7 C11 is fully opened (control input 0).

Optimal clutch control would avoid the change of sign of the transmitted torque in clutch C11 (and other clutches involved) and therefore avoid the double peak in the car acceleration.

8   Conclusion

The predecessor of the PowerTrain library was realized with the Dymola language, an object-oriented modeling language which is superseded by the Modelica language (and will not be developed further by Dynasim). It has been used successfully to model automatic gearboxes, similiar to the one shown in the lower part of figure 11, for various hardware-in-the-loop simulator projects in the automotive industry.

The Modelica PowerTrain library as successor of the previous Dymola library offers several bene-
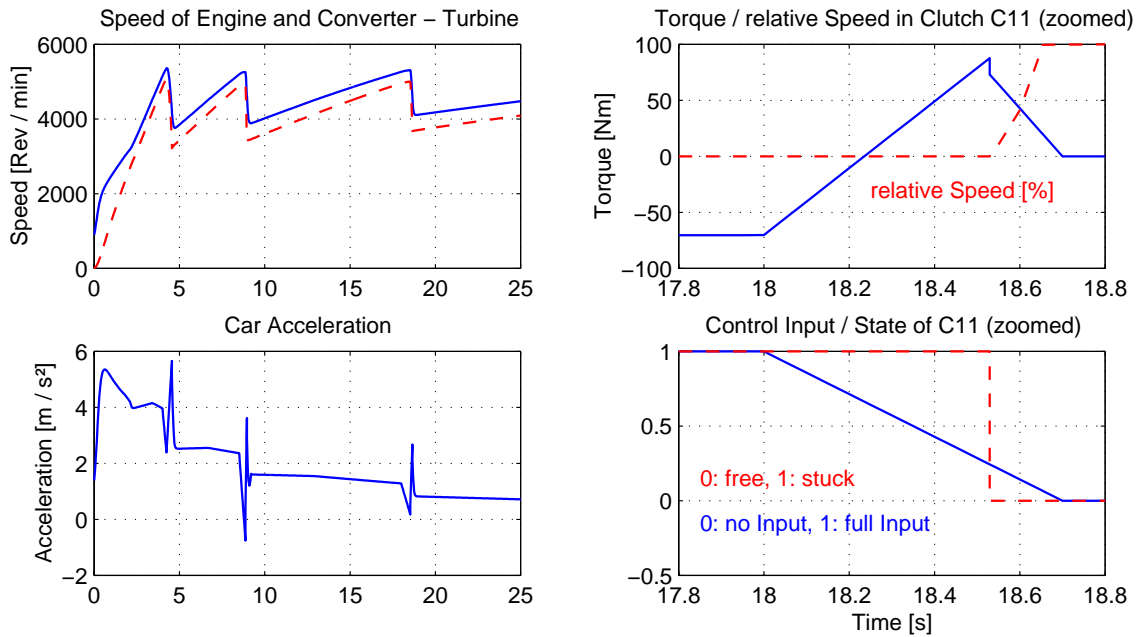
Figure 12: Gearbox simulation results

fits: The friction elements, such as bearing friction, clutches and brakes are realized differently, based on the theory scetched in section 5. This leads to more robust simulations and also to higher efficiency, because 1 or 2 local iterations over the set of dynamically coupled friction elements are sufficient (previously a global iteration over the whole model had to be performed). A lot of new elements are included, such as torque losses and a warm-up model of the engine. Finally, all other available Modelica libraries can be easily utilized, e.g., to model the hydraulics or electronic components.

References

[1] *Beater P.:* HyLib. A Modelica Hydraulics Library. http://www.dynasim.se/

[2] *Dymola.* Homepage: http://www.dynasim.se/

[3] *Elmqvist H., M. Otter,* and *C. Schlegel:* Physical Modeling with Modelica and Dymola and Realtime Simulation with Simulink and Real Time Workshop. 1997 Matlab User Conference, San Jose, 6.-8. Okt., 1997. (http://www.modelica.org/papers/mlconf.ps)

[4] *Förster, H.J.:* Automatische Fahrzeuggetriebe. Springer, 1991.

[5] *Laschet A.:* Simulation von Antriebssystemen. Springer, 1988.

[6] *Otter M., H. Elmqvist,* and *S. E. Mattsson:* Hybrid modeling in Modelica based on the synchronous data flow principle. In *Proceedings of the 1999 IEEE Symposium on Computer-Aided Control System Design, CACSD'99*, Hawaii, USA, August 1999. IEEE Control Systems Society.

[7] *Rubin Z.J., S.A. Munns,* and *J.J. Moskwa:* The Development of Vehicular Powertrain System Modeling Methodologies: Philosophy and Implementation. 1997 SAE Congress and Exposition, Detroit, SAE paper No. 971089.

[8] *Schlegel C.,* and *F. Kessler:* Modellierung und Hardware-in-the-Loop Simulation von Automatikgetrieben. "Hardware-in-the-Loop Simulation in der Fahrzeugentwicklung", Haus der Technik e.V., Essen, 1998