

SimDriveline Software

Martin Gažo – Stanislav Triaška *

Abstract:

This paper deals with the SimDriveline toolbox, which is used to simulate the mechanical components, primarily used in the automotive industry. Toolbox includes blocks, which can simulate the transmission, clutch, diesel or gasoline engine, and many other mechanical components. The article describes the individual blocks from toolbox and for illustration has been described a simple experiment with two speed transmission and brake.

Key words: Transmission, gear, break, engine, angular velocity

1 INTRODUCTION

SimDriveline software is a block diagram modeling environment for the engineering design and simulation of drivelines, or idealized powertrain systems. A driveline propels a vehicle or craft by transferring its engine torque and rotational power into vehicle kinetic energy and translational momentum. The vehicle moves through or on a medium (air, water, ground) which propels it by reaction forces and which acts as a load on the engine. Drivelines consist of bodies spinning around fixed axes and subject to Newton's laws of motion. The bodies can revolve about one axis, multiple parallel axes, or multiple nonparallel axes. Simple and complex gears constrain the bodies to revolve together and transfer torque up and down the driveline axes. Locking and unlocking clutches switch the driveline from one gear set to another. Gears and clutches make up transmissions. With SimDriveline software, you represent a driveline with a connected block diagram, like other Simulink models, and you can group blocks into hierarchical subsystems. You can initiate and maintain rotational motion in a driveline with actuators while measuring, via sensors, the motions of driveline elements and the torques acting on them. You can return sensor signals to the driveline via actuators, forming feedback loops and the basis for controls. The SimDriveline libraries offer blocks to represent rotating bodies; gear constraints among bodies; dynamic elements such as spring-damper forces, rotational stops, and clutches; transmissions; and sensors and actuators. You can also analyze linearized versions of your SimDriveline

models and generate code from them. Driveline Simulation and Physical Modeling: SimDriveline software is based on the Simscape environment, the platform product for the Simulink Physical Modeling family, encompassing the modeling and design of systems according to basic physical principles. Simscape software runs within the Simulink environment and interfaces seamlessly with the rest of Simulink and with MATLAB. Unlike other Simulink blocks, which represent mathematical operations or operate on signals, Simscape blocks represent physical components or relationships directly.

2 RUN SIMDRIVELINE

SimDriveline software is organized into a set of libraries of closely related blocks. This section shows you how to open these SimDriveline block libraries and explains the nature of each library. There are several ways to open the SimDriveline block library. You can access the blocks through the Simulink Library Browser. Open the browser by clicking the Simulink button . Expand the Simscape entry, then the SimDriveline subentry, in the contents tree.

* Institute of Control and Industrial Informatics, Slovak University of Technology, Faculty of Electrical Engineering and Information Technology, Ilkovičova 3, 812 19 Bratislava, Slovakia; martin.gazo@stuba.sk, Stanislav.triaska@stuba.sk

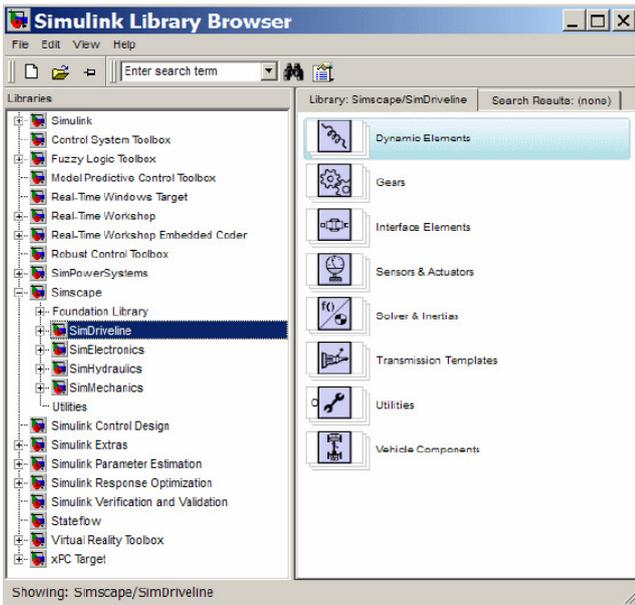


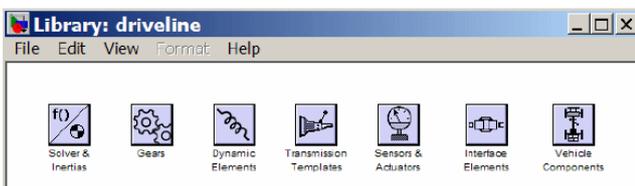
Figure 1 Overview of SimDriveline blocks

You can also access the blocks directly inside the SimDriveline library in several ways:

- In the Simulink Library Browser, right-click the SimDriveline subentry under Simscape and select **Open SimDriveline Library**. The library appears.
- Click the **Start** button in the lower-left corner of your MATLAB desktop. In the pop-up menu, select **Simulink**, then **SimDriveline**, then **Block Library**.
- Enter drivelib at the MATLAB command line.

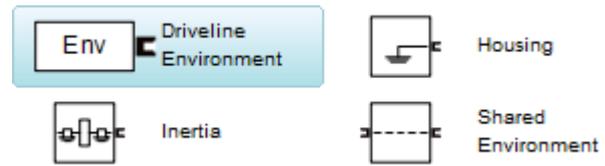
3 SIMDRIVELINE LIBRARY

Once you perform one of these steps, the SimDriveline library opens. This library displays seven top-level block groups. You can expand each library by double-clicking its icon.



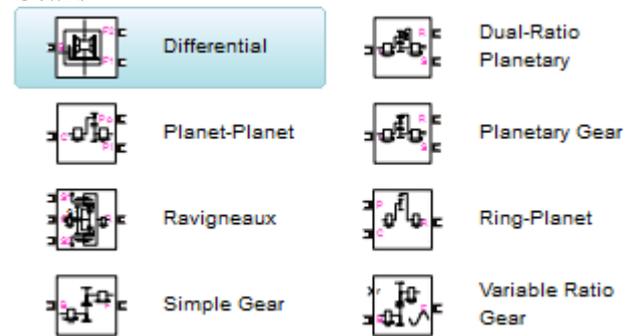
The next section summarizes the blocks of each library and their use. For explanations of individual blocks, consult the SimDriveline “Block Reference” reference. The SimDriveline block library is organized into separate libraries, each with a different type of driveline block.

Solver a Inertias



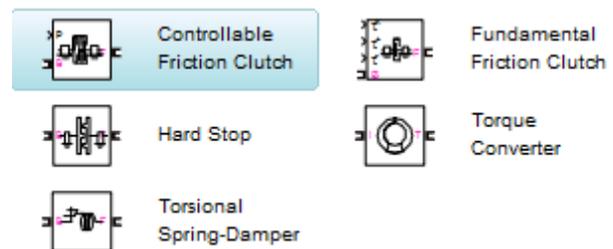
The Solvers & Inertias library provides the Inertia block, which represents a rotating body specified by its moment of inertia, the fundamental unit of driveline modeling. It also contains the Housing block, which represents an immobile rotational ground. Finally, the library contains the Driveline Environment block, which configures the driveline settings of a SimDriveline block diagram, and the Shared Environment block, which allows you to connect two driveline block diagrams in a nonphysical way so that they share the same driveline environment settings.

Gears



The Gears library contains blocks that represent simple and complex gears, driveline elements that couple distinct driveline axes and constrain their relative motions. The Gear blocks range from simple two-wheel gear couplings with fixed and variable gear ratios, to complex multiwheel and multiaxis gears such as planetary and differential gears.

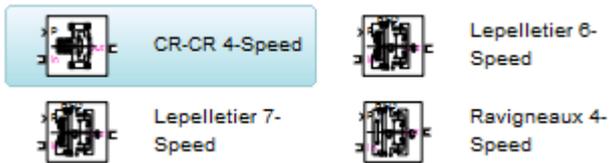
Dynamic Elements



The Dynamic Elements library contains blocks that model such critical drivetrain components as clutches, torque converters, damped springs, and stops. Dynamic elements generate internal driveline torques.

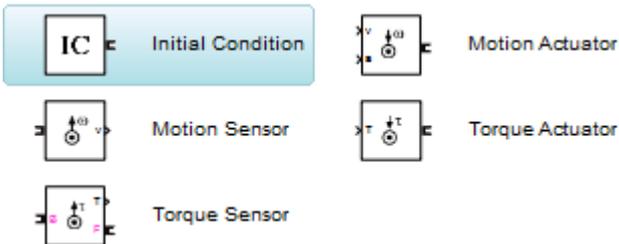
The blocks of this library serve as suggestions for developing variant or entirely new models to simulate the same components. Look under the block mask, break the block's library link before modifying it, and create your own version.

Transmission Templates



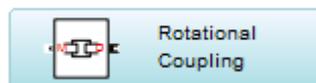
The Transmission Templates are a set of predesigned transmission examples constructed from gears, clutches, and inertias. You can copy and use these examples in your drivetrain models. Transmission templates copied into your model are not linked to the block library. You can modify and rebuild these template copies at will.

Sensors & Actuators



The Sensors & Actuators library provides blocks for sensing and initiating the motions of driveline axes and applying and sensing torques along those axes.

Interface Elements



The Interface Elements library enables connections between SimDriveline driveshaft connection lines and Simscape mechanical rotational motion.

Vehicle Components



The Vehicle Components library contains blocks that represent components of a full vehicle beyond the drivetrain itself. It includes models of engines, wheeled vehicles, and tires in contact with the ground. The blocks of this library serve as suggestions for

developing variant or entirely new models to simulate the same components. Look under the block mask, break the block's library link before modifying it, and create your own version.

Essential Steps to Building a Driveline Model

The key rules for connecting driveline blocks to each other and the dual roles of driveline connection lines is transferring torque and enforcing angular velocity constraints. You should review these rules before building and running the tutorial models of this chapter.

- Driveline blocks, in general, feature both driveline connector ports  and regular Simulink inports and outports $>$. You connect connector ports to one another and Simulink ports to one another. But you cannot connect a driveline port to a Simulink port.
- The driveline connection lines interconnecting driveline connector ports  represent driveline axes and enforce physical relationships. Unlike Simulink lines, they do not represent signals or mathematical operations, and they have no inherent directionality.
- A driveline connection line represents an idealized massless and perfectly rigid spinning shaft. A driveline connection line between two ports enforces the constraint that the two driveline components so connected rotate at the same angular velocity. The connection line also transfers any torque applied to a driveline component at one end to the component at the other end.
- You can branch driveline connection lines. You must connect the end of any branch of a driveline connection line to a driveline connector port .
- Branching a driveline connection line modifies the physical constraints that it represents. All driveline components connected to the ends of a set of branched lines rotate at the same angular velocity. The torque transferred along the input driveline axis is split up among the branches. How the torque is split depends on the dynamical details of the system that you are modeling.
- Driveline connection lines satisfying the angular velocity constraint must have the same initial angular velocities.

4 SIMULATION EXPERIMENTS

We have simulated a car drive. This simulation includes engine and transmission models and a simplified model of the drivetrain-wheel-road coupling. The engine and transmission are coupled with a torque converter. Programmed clutch control steps the transmission through four gears and neutral before a braking torque is applied. This section explains these features, subsystems, and their

relationship and purposes, leading you to actual simulation.

Time Range (s)	CR-CR Gear Setting	Brake Torque (N-m)
0 - 10	1	0
10 - 20	2	0
20 - 30	3	0
30 - 40	4	0
40 - 44	Neutral	0
44 - 50	Neutral	0 - 10,000

Table 1 Switching table

The main driveline subsystems are

- Engine
- Torque converter
- Transmission

The large subsystem to the right represents the final part of the drivetrain: the vehicle inertia, the wheels, their coupling to the road, and braking. All the other subsystems in the model represent inputs that control the drivetrain or outputs that measure its behavior.

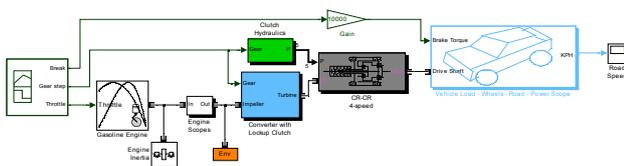


Figure 1 Full car model (Matlab simulink)

Running the Model, Switching Gears, and Braking

1 After we have built up a model, we have simulated a several actions.

2 We have changed the gears according to switching table (Table 1), and observed ground speed of a vehicle.

3 At the end of simulation we have applied the Braking torque at time 44s. We have observed all important signals of a model. (Ground speed, engine speed and torque, driven ratio, clutch and transmission)

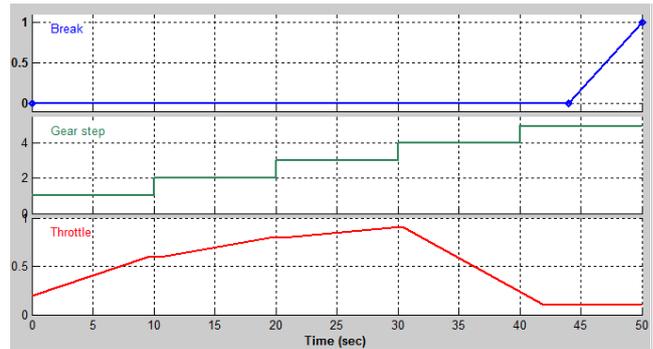


Figure 2 Signal generator for Break, Gear and Throttle signals

The brake torque signal is designed not to be applied until after the CR-CR 4-speed transmission is set into neutral and the lockup clutch is triggered. The signal is 0 Nm until 44 seconds of simulation time. It then rises to a large value at the end of the simulation at 50 seconds. The brake torque is strong enough to stop the driveline completely before 50 seconds. Close the Programmed Brake Torque subsystem and related windows. The gear signals shows how the gears are set in each time period. The throttle signal is programmed to produce a realistic acceleration.

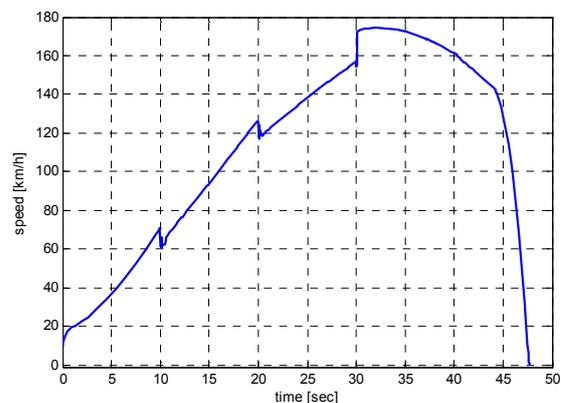


Figure 3 Gear and break switching diagram

The Road Speed scope displays the vehicle's linear velocity in kilometers per hour (km/h). Note that the braking stops the driveline and the vehicle before 50 seconds — to be exact, at 47.5 seconds.

6 CONCLUSION

This toolbox is designed for simulation and modelling of mechanical components primarily use for the automotive industry. In Toolbox there are modelled various components such as gearboxes, couplings, clutches, sensors, engines, wheels and various external environments. These components were used to create simple two speed transmission model, which was transferred to a series of experiments. The simulation schema and the results of experiments are shown in the figures and graphs in the article.

Acknowledgement

Since this work has been supported by the Slovak Research and Development Agency project reference number is: VMSP-P-0155-09. Therefore the authors of the article would like to give thanks for the support.

REFERENCES

1. <http://www.mathworks.com/access/helpdesk/help/toolbox/physmod/drive/> 10.07.2010
2. http://www.kxcad.net/cae_MATLAB/toolbox/physmod/drive/bqbdaga-1.html 10.07.2010
3. <http://www.mathworks.de/products/simdriv e/demos.html> 10.07.2010