SimDriveline 2.0

Model and simulate mechanical driveline systems

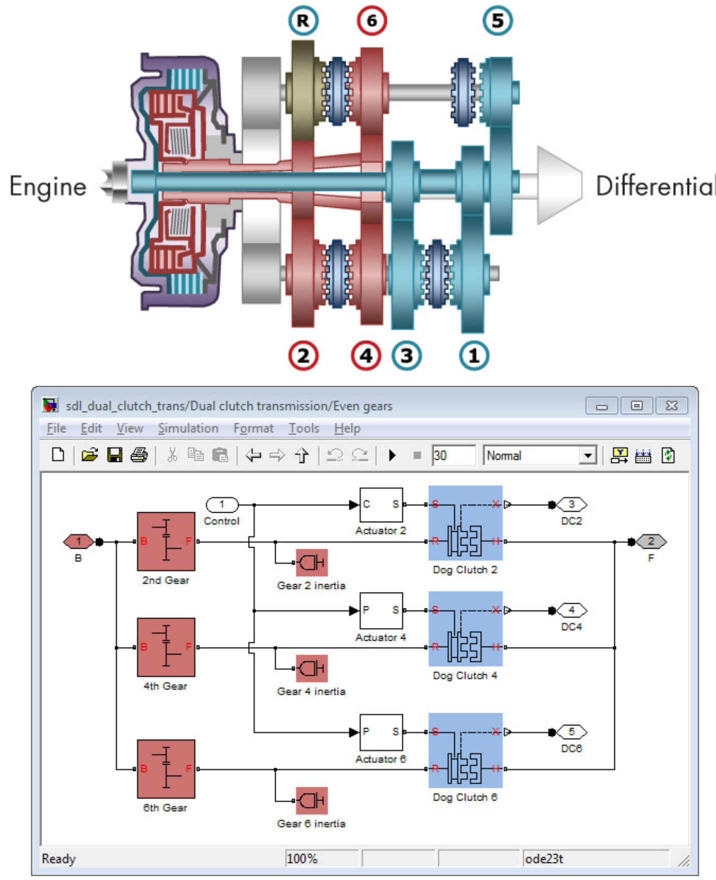
Introduction

SimDriveline™ provides component libraries for modeling and simulating one-dimensional mechanical systems. It includes models of rotational and translational components, such as worm gears, planetary gears, lead screws, and clutches. You can use these components to model the transmission of mechanical power in helicopter drivetrains, industrial machinery, vehicle powertrains, and other applications. Automotive components, such as engines, tires, transmissions, and torque converters, are also included. SimDriveline models can be converted into C code for real-time testing of controller hardware.

Learn more about physical modeling.

Key Features

- Common gear configuration models, including planetary, differential, and worm gears with meshing and viscous losses
- Clutch models, including cone, disk friction, unidirectional, and dog clutch
- Vehicle component models, including engine, tire, torque converter, and vehicle dynamics models
- Models of translational elements, including leadscrew, rack and pinion, and translational friction
- Ideal and nonideal model variants, enabling adjustment of model fidelity
- Ability to extend component libraries using the Simscape language
- Ability to specify units for parameters and variables, with automatic unit conversion
- Support for C-code generation from SimDriveline models (with Simulink Coder™)

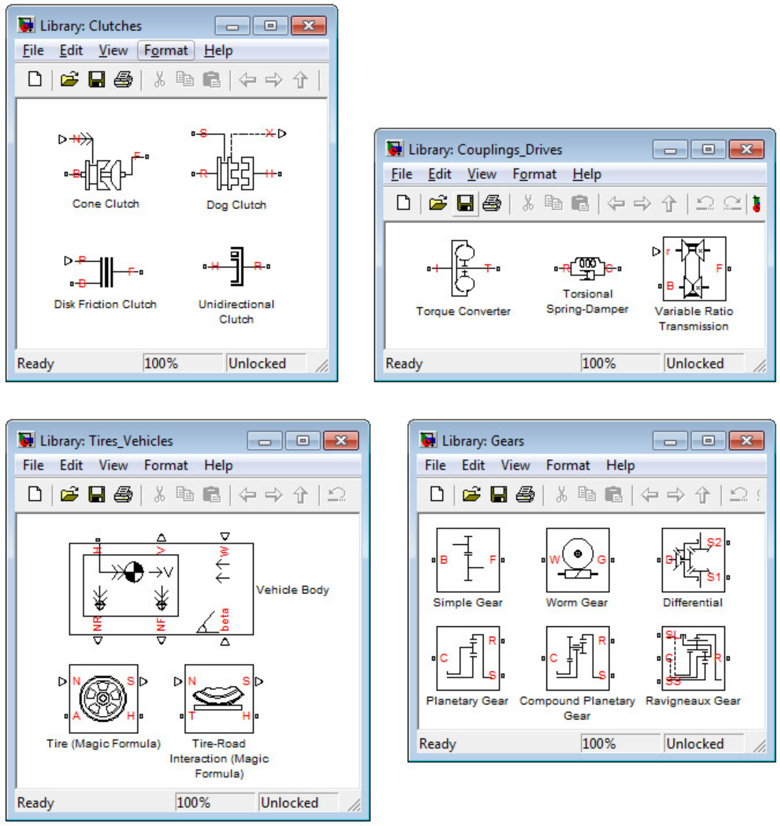


Cross-section of dual clutch transmission (top) and associated SimDriveline model. The colored blocks correspond to gears and the dog clutches that control gear selection.

Modeling Drivetrain Systems

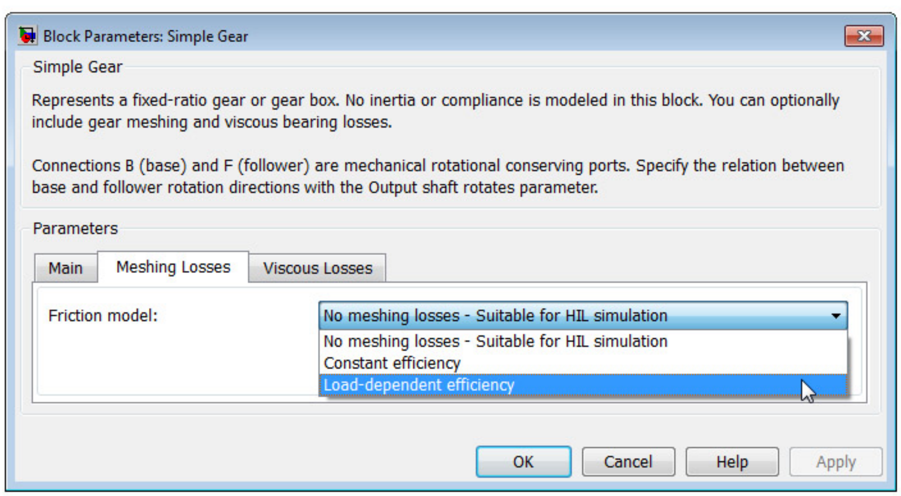
SimDriveline provides libraries of one-dimensional mechanical components. You can connect components, such as planetary gears, clutches, and brakes, to model your mechanical system. The models you create can be grouped into subsystems, making them easier to read and reuse.

In addition to the traditional input-output or signal flow connections used in Simulink®, SimDriveline uses physical connections that permit the flow of power in any direction. Models built using physical connections (also referred to as acausal models) closely resemble the physical system they represent, and are easier to understand and share with others.



SimDriveline libraries (clockwise from top left): clutches, couplings, gears, and tires.

Many of the component models in SimDriveline let you adjust the level of fidelity. You can choose to include or neglect certain effects, such as meshing and viscous losses, and as a result, balance the tradeoff between model fidelity and simulation speed.

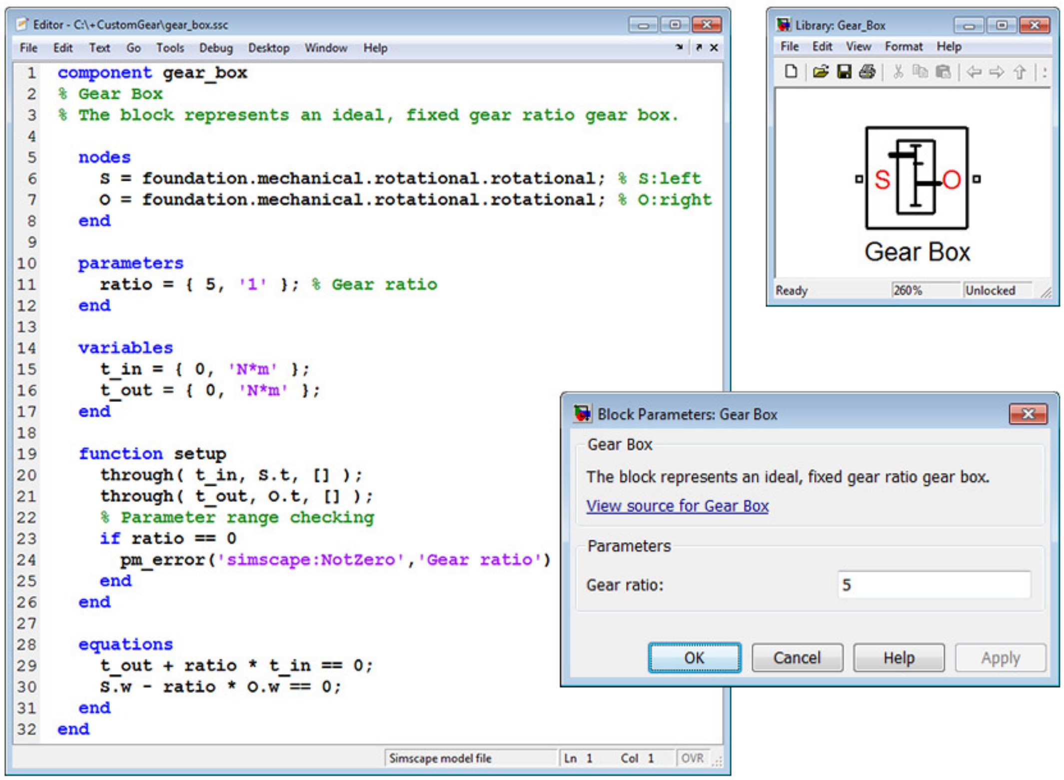


Dialog box for SimDriveline simple gear model. You can select the fidelity level of the friction model.

Creating Custom Components

You can add components from other physical modeling products to your SimDriveline model. The foundation library in Simscape™ contains blocks in other physical domains, such as electrical, hydraulic, and thermal. Integrating these domains into your SimDriveline model using physical connections helps expand your model's range of effects.

The Simscape language, an object-oriented language that is based on MATLAB®, enables you to create your own physical modeling components and libraries. You can define custom components, complete with parameterization, physical connections, and equations represented as acausal implicit differential algebraic equations (DAEs). Within your component's Simscape language file you can use MATLAB to analyze parameter values, perform preliminary computations, and initialize system variables. The Simulink block and dialog box for your custom component are automatically created from the file.



Simscape language file (left) used to create a custom gear box. The file automatically generates a Simulink Gear Box block and dialog box.

Simulating Drivetrain Systems

You can combine SimDriveline models with Simulink control system models for dynamic simulation. The simulations can be run on your desktop (variable step) or in a real-time environment (fixed step). Every aspect of your simulation can be automated using scripts in MATLAB, including configuring the model, entering simulation settings, and arranging sets of simulations.

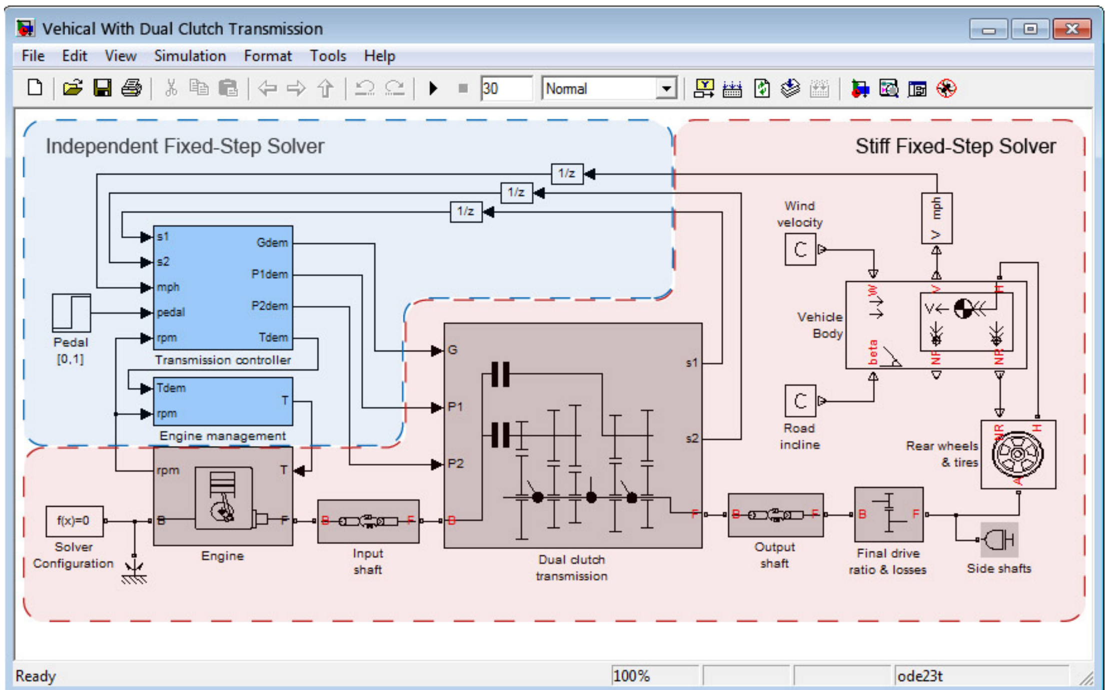
You can use optimization algorithms to automatically tune parameters in simulation. This approach enables you, for example, to find designs that minimize weight or minimize fuel consumption. To accelerate optimization tasks and other design studies that require many simulations, you can use Parallel Computing Toolbox™ to distribute your SimDriveline simulations across multiple cores or a cluster of computers.

Analyzing Drivetrain Systems

All of the data from your SimDriveline model can be saved automatically to the MATLAB workspace. Using MATLAB, the results of your simulation can be analyzed, plotted, animated, and saved into many different file formats. You can perform tasks such as analyzing the frequency response of the powertrain, comparing simulation runs to improve fuel economy, and verifying the timing of clutch events during the simulation. When combined with Simulink Report Generator™, the results of SimDriveline simulations can be automatically saved in a report, along with screenshots of the model, plots, and other information.

Performing Hardware-in-the-Loop (HIL) Simulations

SimDriveline models can be configured specifically for real-time simulation and converted to C code, enabling you to perform HIL tests. Many components in SimDriveline can be configured to use abstracted behavioral models, ideal for real-time simulation. Using Simscape local solvers, you can speed up your simulation by using a fixed-step solver for your physical system and independently choosing a different solver for the rest of your model.



Solver configuration for dual-clutch transmission model. A stiff fixed-step solver is used for the physical system (shaded pink) and an independent fixed-step solver is used for the rest the model (shaded blue).

You can generate C code from your SimDriveline models using Simulink Coder. The generated code can be used to run HIL simulations on real-time processors that interface directly with hardware. This enables you to test your control algorithms without relying on hardware prototypes.

Deploying Drivetrain Models

You can deploy SimDriveline models using code generated with Simulink Coder. The generated code lets you:

- Build stand-alone executables of SimDriveline models that can be integrated into C programs or other MATLAB and Simulink models
- Run HIL simulations by deploying SimDriveline plant models onto real-time processors that interface directly with hardware
- Improve simulation speed by compiling the C code

- Share models without exposing your intellectual property

Sharing Models

You can share SimDriveline models with Simscape users who have not purchased SimDriveline. Simscape users can view, simulate, and change parameter values in SimDriveline models by leveraging the Simscape Editing Mode. As a result, your team can share SimDriveline models with a larger group of engineers who use Simscape.

Working with SimDriveline Models		
Task	**Model Developer** (Purchases Simscape and SimDriveline)	**Model User** (Purchases Simscape)
Simulate	√	√
Log data or change visualization	√	√
Change numerical parameters	√	√
Generate code with Simulink Coder	√	√
Change block parameterization options	√	X
Make or break physical connections	√	X

Resources

Product Details, Demos, and System Requirements
www.mathworks.com/products/simdrive

Trial Software
www.mathworks.com/trialrequest

Sales
www.mathworks.com/contactsales

Technical Support
www.mathworks.com/support

Online User Community
www.mathworks.com/matlabcentral

Training Services
www.mathworks.com/training

Third-Party Products and Services
www.mathworks.com/connections

Worldwide Contacts
www.mathworks.com/contact