

MCE 503: Modeling and Simulation of Mechatronic Systems

Overview of the 20-Sim Software

October 16, 2006

1 Background

20-Sim is a general purpose modeling tool. It understands bond graphs as well as conventional icon-based modeling. The package is able to simplify and analyze a bond graph, and, more importantly, it can generate state equations in symbolic form. In addition, 20-sim is capable of generating ready-to-run code for Matlab/Simulink. 20-sim was developed at the Control Laboratory of the University of Twente, in the Netherlands. We currently have a demo which is fully functional except for saving the models. We will export our models into Matlab/Simulink, where they can be saved.

2 Project 1: RLC Circuit

We will use the simple, linear RLC circuit shown in Fig. 1 as an example of the basic way to work in 20-sim.

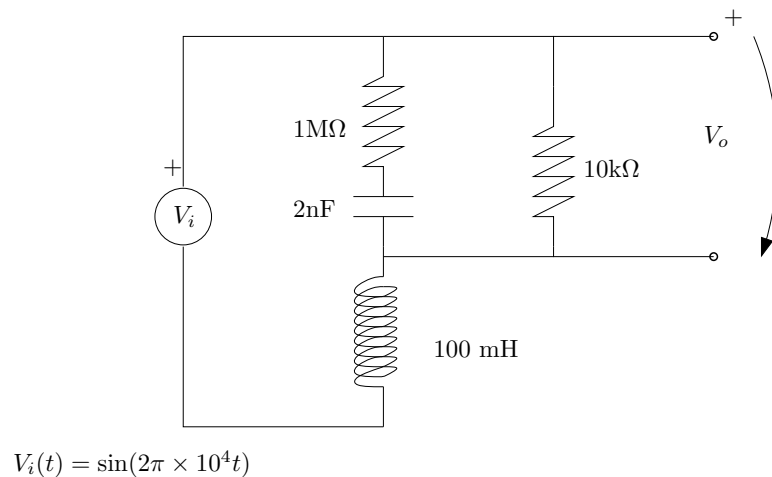


Figure 1: Example System

The objective is to simulate the circuit in a flexible and quick way. As you will see, it is preferable to export the model to Matlab and perform the simulation there as opposed to using 20-sim's built-in simulator. Do the following:

1. Start a blank model in 20-sim. Follow the construction method for electrical systems directly on the screen by dragging the appropriate elements from the Bond Graph library. Do not perform any manual simplification.
2. Use a modulated source of effort instead of a plain SE. The SE in 20-sim is only for constants. The MSE will require an input. Find the Waveform Sine block under the sources menu.
3. Observe how causality is automatically and progressively assigned as you build the model. Power directions result from the order in which you connect two elements. You can reverse power directions at will.
4. Now instruct 20-sim to simplify the model. The program will run a verification of the model.
5. Compare the result to a bond graph obtained by inspection. If simplification was not complete, it must be that you have inconvenient power directions. Reverse some power directions to obtain a thru-power form. This will result in maximum simplification.
6. Enter numerical values for all elements and for the Waveform Sine block, according to Fig. 1.
7. Start the simulator and change the simulation time to 1 sec. Plot the input vs. time and the effort at the output vs. time.
8. Now use the Model/Show Equations function to see system equations in 20-sim syntax.
9. Now go to Tools/Model Linearization. Although this is a linear model, this is where the symbolic functions are found. Choose the source effort to be the input and the effort at the flow source to be the output. A symbolic set of state-space matrices A,B,C and D is generated.
10. Generate a Matlab state-space system (numerical only)
11. Export the model to Matlab (symbolic with numerical parameter declarations). Observe that a lot of overhead is generated alongside the equations. If we are interested in a ready-to-run model, we keep the code. If we are interested only in the equations, we extract that section of the code and eliminate the rest.
12. Export the model to Simulink, but delete the sources first. This will create a ported Simulink block that will allow is to change the inputs and store the output.

3 Project 2: Nonlinear Elements

Nonlinear constitutive laws can be easily introduced via code. Create a bond graph consisting of a MSE driving an R. Use the Waveform Sine block to drive the MSE. Once the model is created and verified, look in the model hierarchy tree on the left and find the R element. The constitutive law will be displayed in an easy-to-understand syntax. Change it to the constitutive law of a diode. That is, use:

$$f = \begin{cases} \frac{e}{R_h} & \text{if } e < 0.7 \\ \frac{e}{R_l} & \text{otherwise} \end{cases}$$

where R_h is a high resistance (use 1000) and R_l is a low resistance (use 0.001). Do this following the 20-sim syntax and aided by the predefined program structures found on the left. Simulate the circuit and verify that the current on the diode is rectified (negative portions chopped off).