Nonlinear Control (FRTN05)

Introduction to Simulink

Last updated: Spring of 2014

Contents

The exercise is intended as an introduction into Simulink and the Control System Toolbox. It can be performed in Matlab and Simulink, either at the department or at any other computer that has Matlab with the Control System Toolbox, and Simulink. To find out if you have the control system toolbox write `help control`. (We will use the commands `ss`, `tf`, `bode`, `nyquist`, `evalfr`. Use `help` to find out how they work.) The introduction contains small examples of how Simulink works.

1. Introduction to Simulink

Simulink is a simulation program based upon Matlab. There are several ways to define a model. One can work graphically and connect block-diagrams with predefined blocks. Alternatively one can give the mathematical description in forms of differential equations in an m-file (the format for programs written in the Matlab programming language). Matlab/Simulink supports both these representations as well as combinations. Furthermore one can use descriptions that include a hierarchy of connected subsystems.

To understand how models are described and simulated using block diagrams, it is best to run small examples on a computer.

1.1 How to Start Simulink

Start Matlab. Then give the command `simulink` in Matlab. This gives a window with blocks as in Figure 1. Each block represents a library that contains several building blocks.

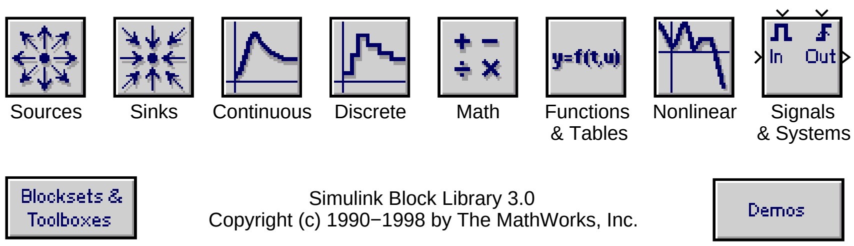

Figure 1 Available Simulink block diagram libraries

1.2 A Simple System

Click on File in the Simulink-window and choose New->Model. Click on the library Continuous and move a Transfer Fcn to the new window called "Untitled". Do the same with Sources->Step Fcn and Sinks->Scope. Draw arrows (left mouse button) and connect the ports on the block. You should now have a block diagram as in Figure 2.

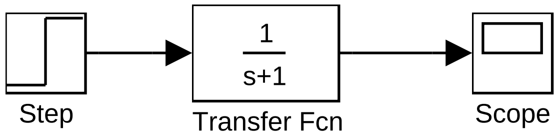

Figure 2 A simple Simulink system

Choose Simulation->Parameters in the window called "Untitled". Set Stop time to 5. Open a window for the Scope by double clicking on it. Start a simulation by Simulation->Start (or by pressing Ctrl-t in the window called "'Untitled").

How to Change a System To change the system to

$$\frac{1}{s^2 + 0.5s + 2}$$

you double-click on the block Transfer Fcn and change Denominator to [1 0.5 2]. Simulate the new system (Simulation->Start or Ctrl-t). Change parameters in the Simulation menu and the scales in the block Scope until you are satisfied.

How to Change an Input Signal To change the input signal, start with removing the block Step Fcn by clicking on it and delete it by pressing Delete (or using Edit->Cut or pressing Ctrl-x). Replace it by a Sources->Signal Generator block. Double-click on Signal Generator and select a wave form, amplitude and frequency. Also change Simulation->Parameters->Stop Time to 99999 and press Simulation->Start. This gives an "infinite" simulation that can be stopped by pressing Simulation->Stop (or Ctrl-t). Can the amplitude of the input signal be changed during simulation? Also try to change the block Transfer Fcn during simulation.

How to Use Matlab Variables in Blocks Note that variables defined in the Matlab environment can be used in Simulink. Define numerator and denominator by writing the following in the Matlab window.

```
num=[1 1]
den=[1 2 3 4]
```

Change Transfer Fcn->Numerator to num and Transfer Fcn->Denominator to den.

How to Save Results to Matlab Variables To save input and output, move two copies of the block `Sinks->To Workspace`. Connect these with the input and output to the block `Transfer Fcn`. Get a `Sources->Clock` and connect it to a `Sinks->To Workspace`. Double click on the "Workspace blocks" to be able to change the variable names to `u,y,` and `t` respectively. Also change `Save format` to the value `Array`. The window should look something like Fig. 3.

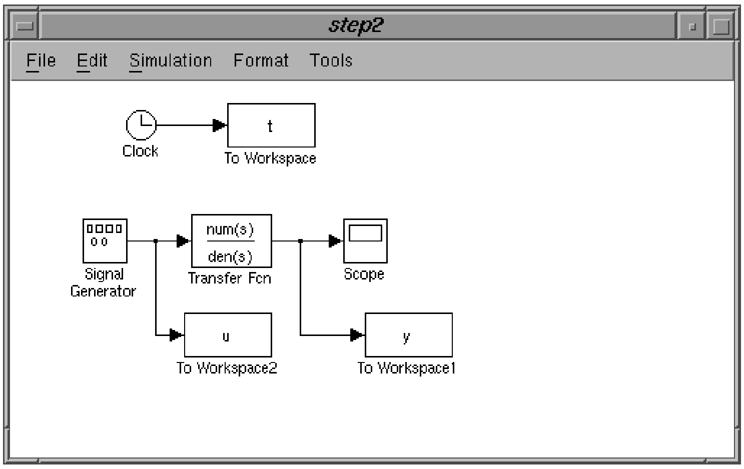

Figure 3 How to save results to Matlab variables in Simulink

How to Use Simulation Results in Matlab Calculations Let the input signal be a sinusoidal with frequency 0.1 rad/s and amplitude 1. Do a simulation that is long enough for the output to become stationary. Compute

```
n=length(y)
max(y(n/2:n))
```

and compare this with the theoretical value $|G(0.1i)|$.

```
>>g=tf(num,den)

Transfer function:
        s + 1
---------------------
s^3 + 2 s^2 + 3 s + 4

>> abs(evalfr(g,0.1*i))
ans =
    0.2518
```

How to Save Systems Use `File-Save As` or `File->Save`.

1.3 A Flow System

Consider a simple tank as in the basic control course

$$
\begin{aligned}
\dot{h} &= \frac{1}{A}(u - q) \\
q &= a\sqrt{2gh}.
\end{aligned}
$$

This can be implemented in Simulink as in Figure 4. The function $f(u)$

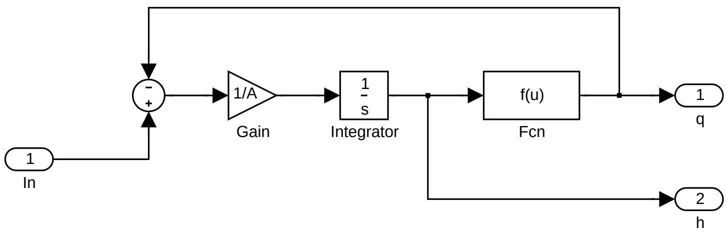

Figure 4 A tank system

has the value `a*sqrt(2*g*u[1])`. The summation block has been given two inputs with different signs by assigning the string "-+" to `Sum->List of Signs`. The summation and the `Gain` blocks are found in the `Math` library and the `Fcn` block is found in the `Functions & Tables` library. The small ellipses, that are contained in the `Signals & Systems` library, tell Simulink what should be considered inputs and outputs to this (sub)system. The block titles can be changed by clicking on them. Mark the entire system by holding the left mouse bottom pressed and drawing a rectangle around it. Then choose `Edit->Create Subsystem`. The result is that the system is represented by one block. Use `Edit->Copy` to create the following double-tank system. Use the commands `trim` and `linmod` to find a linearized model

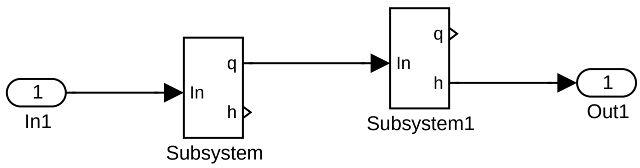

Figure 5 Two tanks and some connections

of the double tank around $h_1^0 = h_2^0 = 0.1$. Use the parameters $A_1 = A_2 = 2.7 \times 10^{-3}$, $a_1 = a_2 = 7.0 \times 10^{-6}$, $g = 9.8$. Plot the Nyquist curve using the command `nyquist`.

```
>> A=2.7e-3;a=7e-6;g=9.8;
>> [x0,u0,y0]=trim('flow',[0.1 0.1]',[],0.1)
 Warning: Output port 2 of block 'twotank/Subsystem' is not connected.
 Warning: Output port 1 of block 'twotank/Subsystem1' is not connected.
```

```
x0 =
    0.1000
    0.1000
u0 =
   9.7995e-06
y0 =
    0.1000
>> [aa,bb,cc,dd]=linmod('flow',x0,u0);
>> sys=ss(aa,bb,cc,dd);
>> bode(sys)
```

Alternative: Linearization in Simulink;
By right-clicking on a signal connector in a Simulink model you can add “Linearization points” (inputs and/or outputs). Use this for the two water tanks.
Start a “Control and Estimation Tool Manager” by
`Tools -> Control Design ->Linear analysis ...`, see Fig. 6. Here you can set the desired operating points, export linearized model to Workspace (Model-> Export to Workspace) and much more.

Repeat the linearization of the system at the same equilibrium point as above.

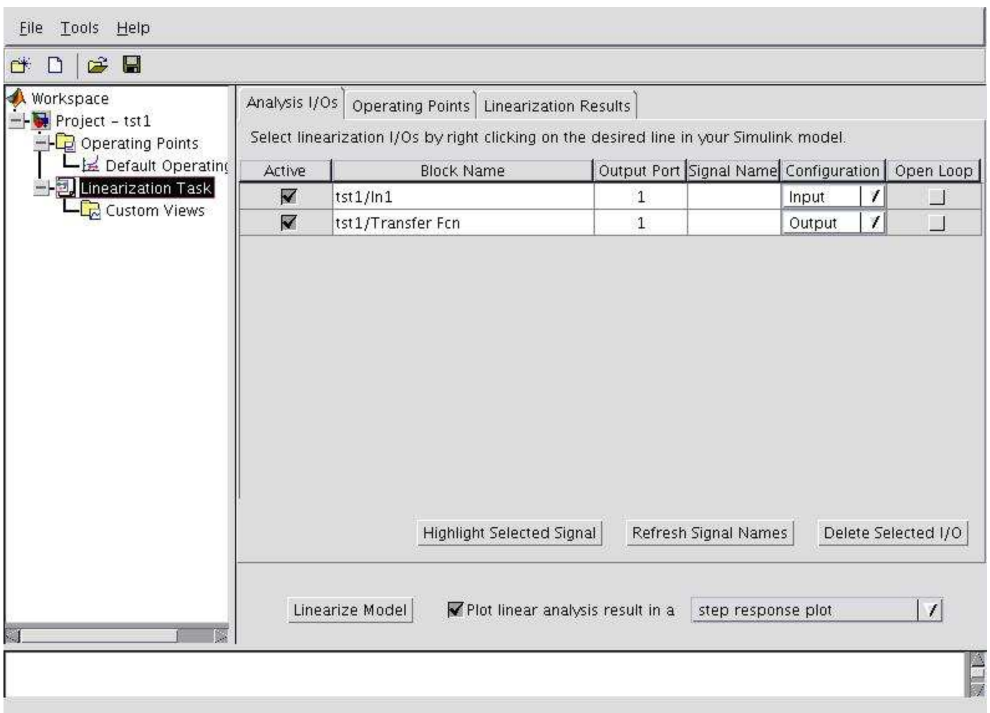

Figure 6 View of "Control and Estimation Tool Manager" in Matlab/Simulink.