HowTo, Matlab/Simulink for basic modeling

Nasser M. Abbasi

July 2, 2015 Compiled on September 7, 2023 at 11:01am

Contents

1 How to solve first order ode with zero initial conditions?

Solve $y'(t) + 3y(t) = e^{-t}$ with $y(0) = 0$

`models/model_1.slx`

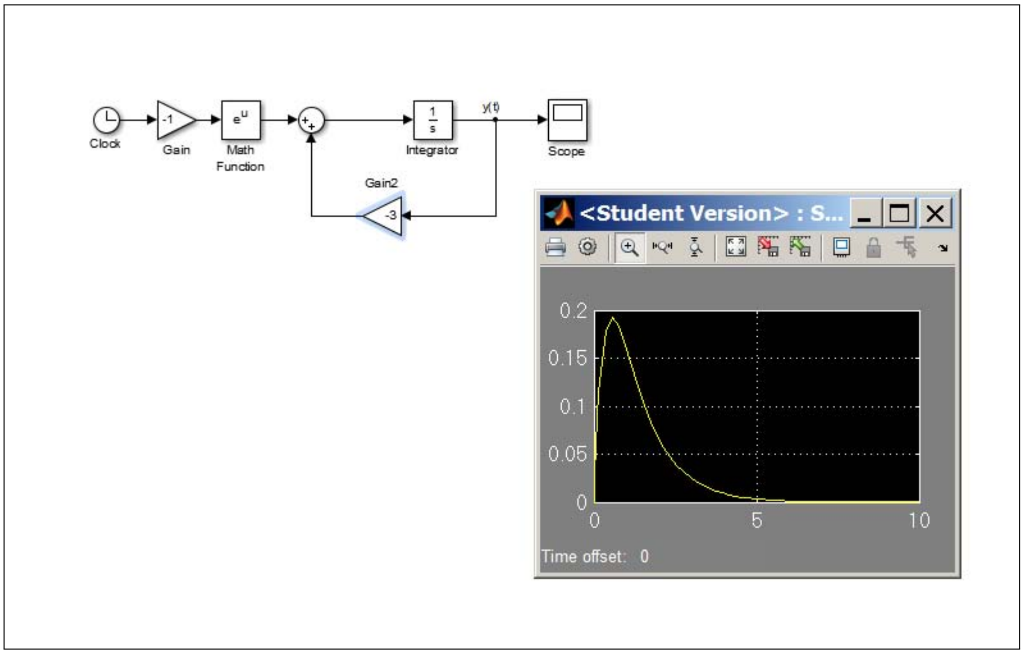

Figure 1: First example

2 How to solve first order ode with non-zero initial conditions?

Solve $y'(t) + 3y(t) = e^{-t}$ with $y(0) = 1$

`models/model_2.slx`

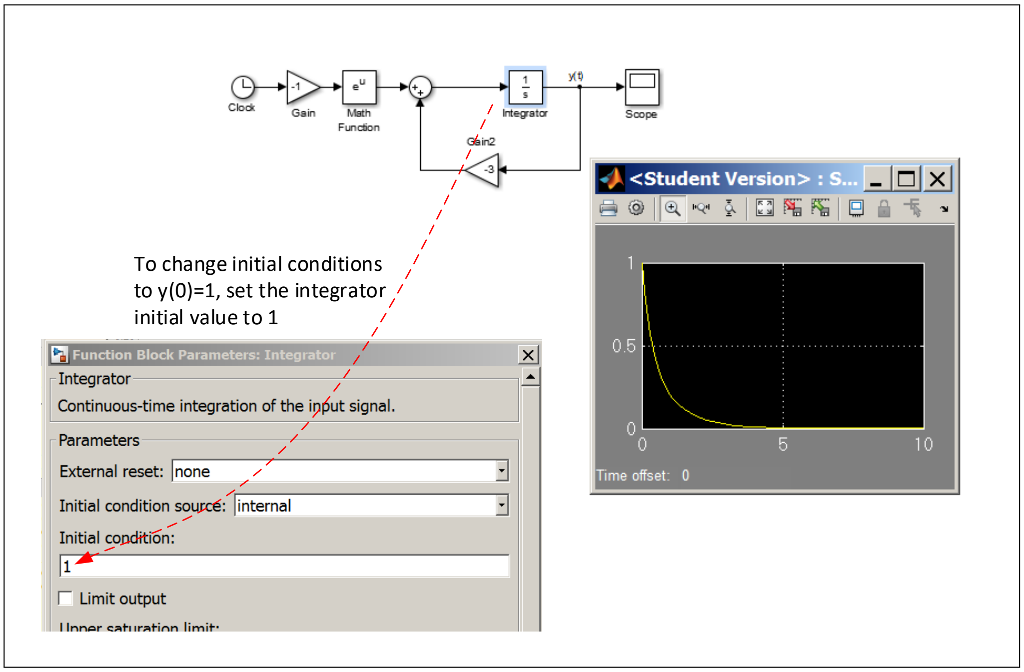

Figure 2: Second example

3 How to solve second order ode with zero initial conditions?

Solve $y''(t) + 5y'(t) + 4y(t) = 5\cos(2t)$ with $y(0) = 0, y'(0) = 0$

`models/model_3.slx`

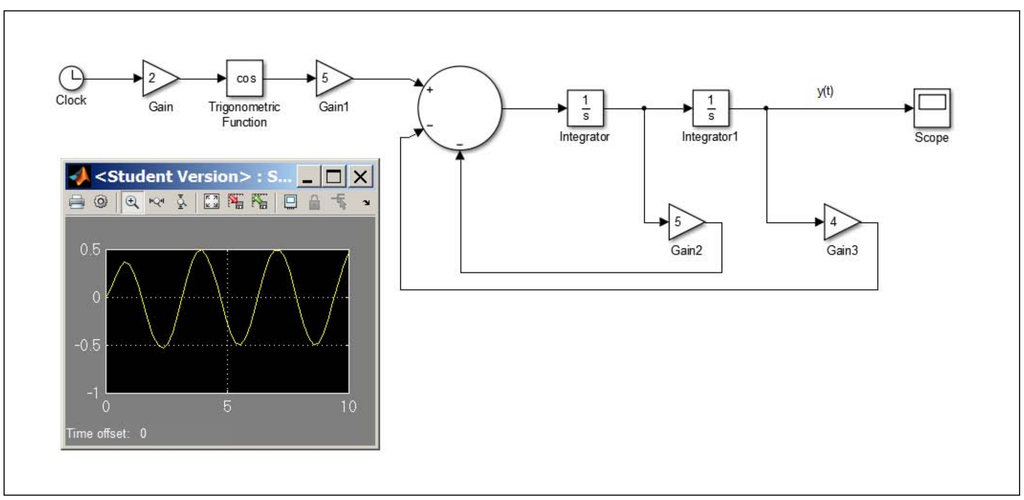

Figure 3: Third example

4 How to solve second order ode with non-zero initial conditions?

Solve $y''(t) + 5y'(t) + 4y(t) = 5\cos(2t)$ with $y(0) = 1, y'(0) = 0.5$

The initial conditions are set up by modifying the integrator $\frac{1}{s}$ block as shown below.

`models/model_4.slx`

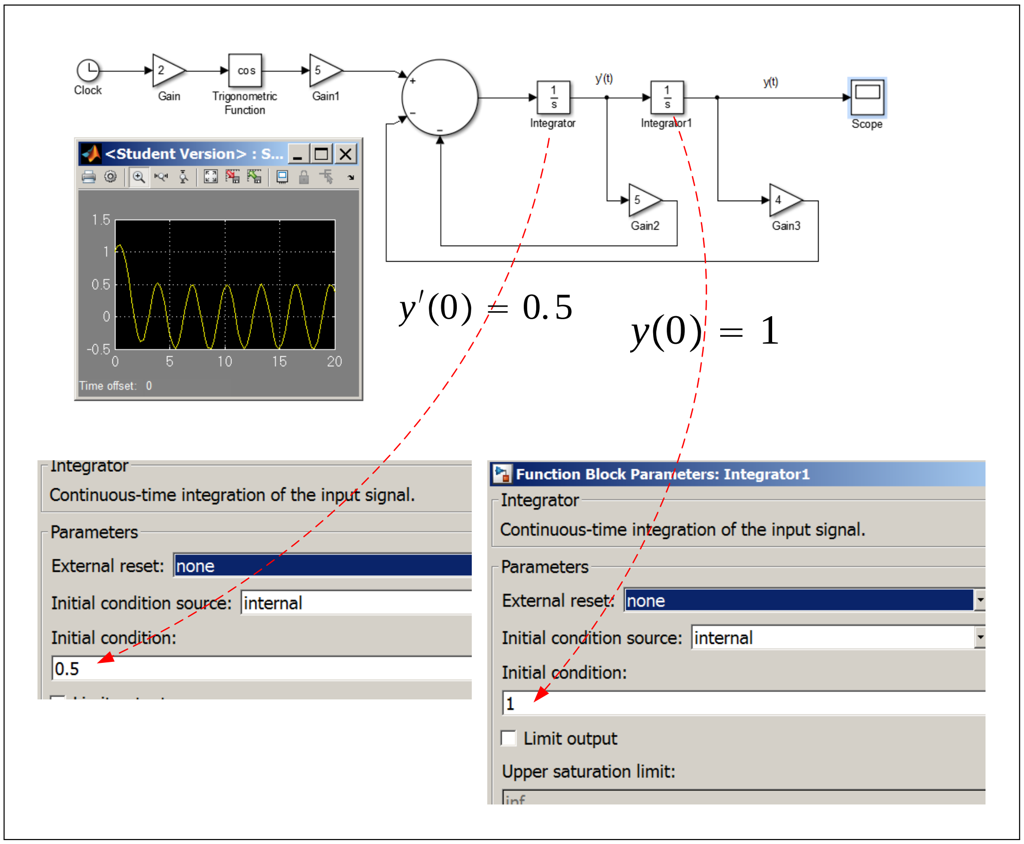

Figure 4: Example 4

5 How to solve mass/spring system with unit step input?

`models/model_5.slx`

From HW problem I did for ECE 717, to be solved using Simulink

For the mass-spring system depicted on the next page, the input $u(t)$ is taken to be the displacement of the supporting platform.

(a) Apply Newton's Laws to obtain the two governing differential equations of motion in s_1, s_2 and u.

(b) With states taken to be $x_1 = s_1, x_2 = s_2, x_3 = \frac{ds_1}{dt}, x_4 = \frac{ds_2}{dt}$ and outputs $y_1 = s_1, y_2 = s_2$, obtain linear time-invariant state equations in the matrix form $\dot{x} = Ax + Bu,\ y = Cx + Du$.

(c) Use Simulink to obtain the unit step response for y_1 and y_2 using normalized parameter values $k_1 = k_2 = 0.5, m_1 = 1, m_2 = 2$. Assume the system is initially at rest; i.e., $x(0) = 0$.

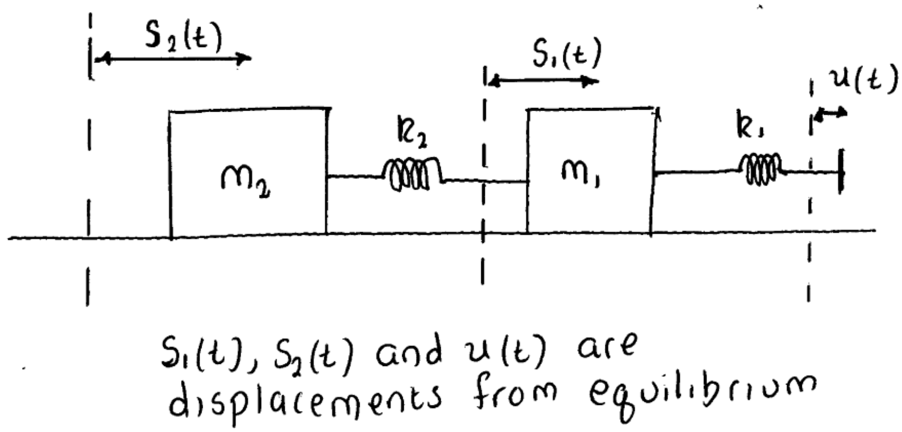

Figure 5: Example 5

Starting with the assumption that the ground surface is smooth and there is no friction. Assuming that all parts are moving in the positive direction to the right. Taking a snap shot when $s_2 > s_1$ so that the spring k_2 is in compression. Spring k_1 is in compression by also assuming that $s_1 > u$ at this instance.

Any other assumptions will also lead to the same set of equations as long as they are used in consistent way when finding the forces in the springs.

Starting with drawing a free body diagram of each body showing all forces acting on them based on the above assumption, and then using $F = ma$ to find the equation of motion of each body m_1, m_2. The free body diagrams is shown below

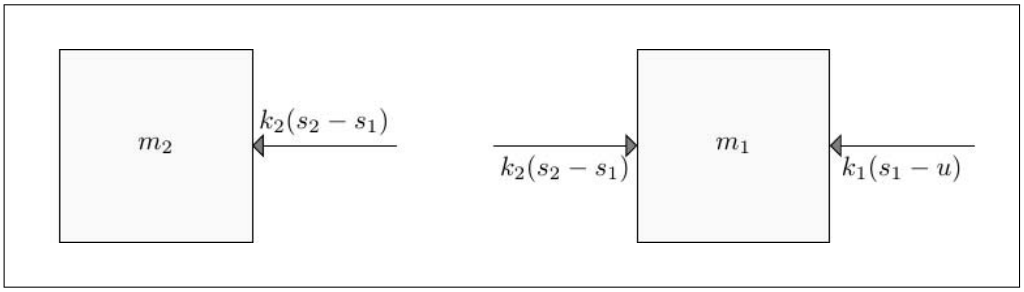

Figure 6: Free body diagram

Now $F = ma$ is applied to each body to obtain the equation of motions. For mass m_2

$$m_2 s_2'' = -k_2(s_2 - s_1)$$
$$s_2'' = -\frac{k_2}{m_2}(s_2 - s_1)$$

And for mass m_1

$$m_1 s_1'' = k_2(s_2 - s_1) - k_1(s_1 - u)$$
$$s_1'' = \frac{k_2}{m_1}(s_2 - s_1) - \frac{k_1}{m_1}(s_1 - u)$$

Now the state space equations are found.

$$\begin{pmatrix} x_1 = s_1 \\ x_2 = s_2 \\ x_3 = s_1' \\ x_4 = s_2' \end{pmatrix} \overset{\frac{d}{dt}}{\Longrightarrow} \begin{pmatrix} x_1' = s_1' = x_3 \\ x_2' = s_2' = x_4 \\ x_3' = s_1'' = \frac{k_2}{m_1}(s_2 - s_1) - \frac{k_1}{m_1}(s_1 - u) = \frac{k_2}{m_1}(x_2 - x_1) - \frac{k_1}{m_1}(x_1 - u) \\ x_4' = s_2'' = -\frac{k_2}{m_2}(s_2 - s_1) = -\frac{k_2}{m_2}(x_2 - x_1) \end{pmatrix}$$
$$= \begin{pmatrix} x_3 \\ x_4 \\ x_1\left(-\frac{k_2}{m_1} - \frac{k_1}{m_1}\right) + \frac{k_2}{m_1}x_2 + \frac{k_1}{m_1}u \\ \frac{k_2}{m_2}x_1 - \frac{k_2}{m_2}x_2 \end{pmatrix}$$

Hence

$$\begin{pmatrix} x_1' \\ x_2' \\ x_3' \\ x_4' \end{pmatrix} = \overbrace{\begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\left(\frac{k_2}{m_1}+\frac{k_1}{m_1}\right) & \frac{k_2}{m_1} & 0 & 0 \\ \frac{k_2}{m_2} & -\frac{k_2}{m_2} & 0 & 0 \end{pmatrix}}^{A(n\times n)} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} + \overbrace{\begin{pmatrix} 0 \\ 0 \\ \frac{k_1}{m_1} \\ 0 \end{pmatrix}}^{B(n\times m)} u(t)$$

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \overbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}}^{C(r\times n)} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} + \overbrace{\begin{pmatrix} 0 \\ 0 \end{pmatrix}}^{D(r\times m)} u(t)$$

The above is in the form of $x' = Ax + Bu$ and $y = Cx + Du$ where $r = 2$ is number of outputs, $m = 1$ is the number of input and $n = 4$ is the number of states.

Using $k_1 = k_2 = 0.5, m_1 = 1, m_2 = 2$ and $x(0) = 0$ now the unit step response for y_1, y_2 is found using Simulink. With the above values the system becomes

$$\begin{pmatrix} x_1' \\ x_2' \\ x_3' \\ x_4' \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0.5 & 0 & 0 \\ 0.25 & -0.25 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0.5 \\ 0 \end{pmatrix} u(t)$$

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix} u(t)$$

Using simulink, state space block was used to implement the above. A step input source was used. Demux was used to send the y_1 and y_2 responses to two different time scopes. Simulation was set for 40 seconds to obtain long enough view of the response. The following figure shows the step response and the model used.

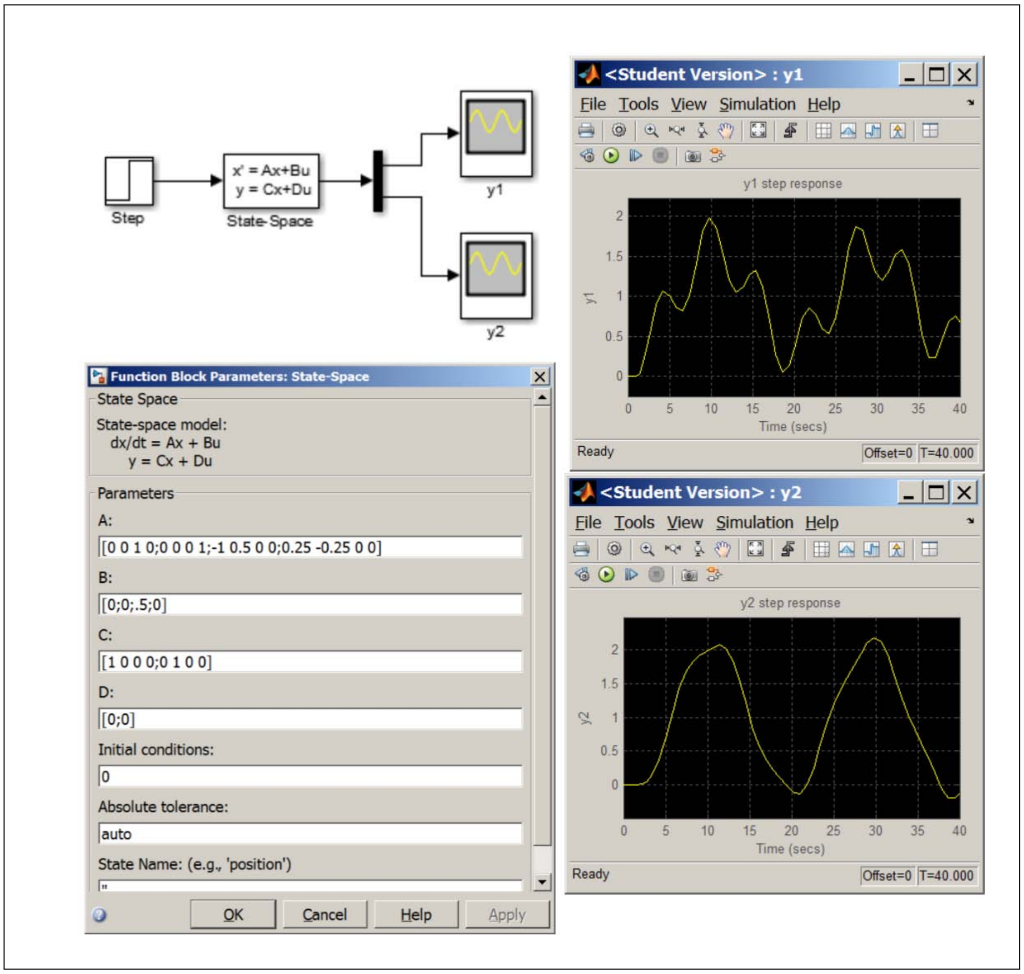

Figure 7: the step response and the model used