**Lesson 8**
**BME 444 - Control Systems**

By the end of this lecture students will be able to:
- Use MATLAB commands or Simulink to simulate the response of a system to inputs
- Simulate systems using transfer functions, state-space representations, or integrator-blocks

I. Overview
   A. We've practiced representing systems as equations
   B. Now it's time to learn how to give the systems inputs and visualize the outputs
   C. We will practice two methods of simulating systems
      1. Matlab commands
      2. Simulink
   D. Within Simulink, we will explore three different methods of building models
      1. Transfer functions (TF)
      2. State-space representation (SSR)
      3. Integrator blocks (IB)
   E. Each approach has advantages and disadvantages, as summarized in the table below

| | TF | SSR | IB |
|---|---|---|---|
| **Allows initial conditions?** | N | Y | Y |
| **Linear/Non-linear** | Linear | Linear | Linear or Non-linear |
| **Insight** | Y | N | N |

II. Command-line simulations
   A. We defined four input types last time. In MATLAB they are:
```
1. u = eq(t,0); % impulse input
2. u = heaviside(t); % unit step
3. u = t; % ramp
4. u = sin(2*t); % sin(wt)
```
   B. Shortcuts: you can use `step()` and `impulse()` but these assume all initial conditions in the system are = 0

C. The function `tf()` creates a transfer function object in MATLAB

D. The function `lsim()` can
   1. Simulate transfer functions or state space
   2. Use any input function
   3. Handle initial conditions if state-space is used

E. **Example 1.** Plot the unit impulse response and step response of the DE shown below

$$\ddot{y} + 3\dot{y} + 12y = 0.8u$$

F. **Student example 1.** Given the RL circuit shown below, use MATLAB to determine the current in the loop and the voltage $v_0(t)$. Assume $v_{in}(t) = u(t)$, zero initial conditions, and $R = 1.6\,\Omega$ and $L = 0.1$ H.

G. **Example 2.** Implement a state-space representation with `lsim()`.

```
A = [0 1; -12 -3];
B = [0; 0.8];
C = [1 0];
D = 0;
sys = ss(A,B,C,D);
```

Then everything else the same as before

H. Define initial conditions in state-space like this:

```
[y,t] = lsim(sys, u, t, x0);
```

Where $x0$ is the initial state vector (the starting values of each state variable)

III. Simulink - Transfer functions

A. Simulink is a graphical way to simulate systems

B. Behind the scenes, it is numerically solving the differential equations; in other words, it is computing actual values from the equations themselves

C. **Example 3.** Implement the RL circuit example in Simulink. Plots outputs of the current and $v_0(t)$. Under Modeling>Model Settings choose ode4 and fixed time step of $10^{-3}$. For convenience, the transfer function is

$$\frac{I(s)}{V_{in}(s)} = \frac{1}{Ls + R}$$

D. Write output to workspace if you want to analyze it in MATLAB
E. Use Scope for output if you want to see the graph
**F. Student example 2.** Use Simulink to model the following differential equation of a mechanical system as a transfer function. Assume a step input of magnitude 12 N is applied at t = 0.02 sec. Run the simulation to 0.1 sec.

$$0.04\ddot{y} + 16\dot{y} + 7000y = u$$

IV. Simulink - State Space
A. State space works essentially the same way as with transfer functions
B. Just choose the state space box instead of the transfer function box
C. Be sure that the dimensions on your **D** vector are correct (down x across)
1. **A** matrix: $n \times n$
2. **B** matrix: $n \times r$
3. **C** matrix: $m \times n$
4. **D** matrix: $m \times r$
D. Define the initial conditions for each state space variable as appropriate
E. **Example 4.** Implement the following state space representation in Simulink. Assume the input is a step input with magnitude 12 N applied at $t = 0$ sec. Run the simulation to $t = 0.1$ sec.

$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -175{,}000 & -400 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 25 \end{bmatrix} u(t) \qquad y = [1 \quad 0] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u$$

V. Simulink - Integrator block method
A. This method is the easiest to implement because you are basically re-creating the equation using the blocks
B. **Example 5.** Implement the following DE in Simulink using the integrator block method. Assume a step input applied at $t = 0$ sec with a

magnitude of 12 N. Run the simulation to $t = 0.1$ sec.

$$0.04\ddot{y} + 16\dot{y} + 7000y = u$$

C. **Student example 3.** Use Simulink to simulate a non-linear model of the hydraulic tank system shown below. Assume the equation is

$$C\dot{P} = Q_{in} - K_T\sqrt{P - P_{atm}}$$

The input is $Q_{in}$ and the output is $P$, tank base pressure

Assume the following:
$C = 0.0002$ m$^3$/Pa
$K_T = 4 \times 10^{-4}$ m$^{3.5}$/kg$^{0.5}$
$Q_{in} = 0.052$ m$^3$/s
$P_{atm} = 1.0133 \times 10^5$ N/m$^2$
$P_0 = 1.15 \times 10^5$ N/m$^2$

Note the $P_0$, which is the initial pressure in the tank.

Run the simulation to $t = 1400$ sec