ECE-320 Lab 7: State Variable Feedback Control Systems

<u>Overview</u>

In this lab you will be controlling your two one degree of freedom systems and your two degree of freedom system that you previously modeled.

You will need your model files for each system. You also need to download the Lab7 files from the class directory.

Design Specifications: *For each of your systems, you should try and adjust your parameters until you have achieved the following:*

<u>Torsional Systems (Model 205)</u>

- Settling time less than 0.5 seconds.
- Steady state error less than 2 degrees for a 15 degree step, and less than 1 degree for a 10 degree step (*the input to the Model 205 must be in radians!*)
- Percent Overshoot less than 10%

<u>Rectilinear Systems (Model 210)</u>

- Settling time less than 0.5 seconds.
- Steady state error less than 0.1 cm for a 1 cm step, and less than 0.05 cm for a 0.5 cm step
- Percent Overshoot less than 10

Your memo should include plots of the response of the real system and of the model for each of your systems. Specifically, you need to plot the position of both the real system and the position of the model on one graph, and the velocity of both the real system and the velocity of the model on one graph. Use subplot to put them both on the same page and be sure to use a legend command and different line styles so I can tell them apart. Don't depend on colors unless you print them on a color printer.

You should have plots for for pole placement and for the LQR design for each of your systems. Be sure to include the values of the closed loop poles and the gains for the LQR algorithm in the caption for each figure. Your memo should compare the difference between the predicted response (from the model) and the real response (from the real system) for each of the systems.

For the two degree of freedom system, you need to plot the positions and velocities of both carts (both predicted and measured) on one graph using the subplot command.

For each of your two 1 dof systems, you will need to go through the following steps:

Step 1: Set up the 1 dof system exactly the way it was when you determined its model parameters.

Step 2: Modify **Basic_1dof_State_Variable_Model_Driver.m** to read in the correct model file and to use the correct *saturation_level* for the system you are using.

Step 4: Pole Placement Design

- Design a state variable controller using pole placement to control the position of the cart or disk and meet the design specs. Use a **constant prefilter**. I would start with poles at around -10 and -15, and move them further from the origin if necessary.

- Simulate the system for 1.5 seconds. *Be sure to use radians for the Model 205 system!* If the design constrains are not met, or the control effort hits a limit, redesign your controller (you might also try a lower input signal)

- Compile the correct closed loop ECP Simulink driver, connect to the system, and run the simulation. For the model 205 this is **Model205_sv1.mdl** while for the model 210 use the file **Model210_sv1.mdl**. *You will probably need to change the base address of these systems, ask for help if you need it!*

- Use the **compare1.m** file (or a modification of it) to plot the results of both the simulation and the real system on two nice, neatly labeled graphs *using the subplot command.* **You need to compare both the position and the velocity of the cart/disk.** The results for the torsional systems must be displayed in degrees and degrees/sec. You need to include these graphs in your memo. *(Note: the steady state error is likely to be off. Placing the poles farther away can reduce this error for these systems.)*

Step 5: LQR Design

- Design a state variable controller using the LQR algorithm to control the position of the cart or disk and meet the design specs. Use a **constant prefilter**.

- Simulate the system for 1.5 seconds. *Be sure to use radians for the Model 205 system!* If the design constrains are not met, or the control effort hits a limit, redesign your controller (you might also try a lower input signal)

- Compile the correct closed loop ECP Simulink driver, connect to the system, and run the simulation. For the model 205 this **is Model205_sv1.mdl** while for the model 210 use the file **Model210_sv1.mdl**.

- Use the **compare1.m** file (or a modification of it) to plot the results of both the simulation and the real system on two nice, neatly labeled graphs *using the subplot command.* **<u>You need to compare both the position and the velocity of the cart/disk.</u>** The results for the torsional systems must be displayed in degrees and degrees/sec. You need to include these graphs in your memo.

<u>For your 2 dof system you will need to go through the following steps:</u>

<u>Step 1</u>: Set up the 2 dof system exactly the way it was when you determined its model parameters.

<u>Step 2:</u> Modify **Basic_2dof_State_Variable_Model_Driver.m** to read in the correct model file and to use the correct *saturation_level* for the system you are using.

<u>Step 4:</u> Pole Placement Design

- Design a state variable controller using pole placement to control the position of the ***<u>first cart</u>*** and meet the design specs. Use a **constant prefilter**. I would start with poles at around -10, -12, -14 and -16, and move them further from the origin if necessary. (If your system does not move, you definitely need to move the poles further from the imaginary axis.)

- Simulate the system for 1.5 seconds. If the design constrains are not met, or the control effort hits a limit, redesign your controller (you might also try a lower input signal)

- Compile the correct closed loop ECP Simulink driver, connect to the system, and run the simulation. For the model 210 use the file **Model210_sv2.mdl**. *You will probably need to change the base address of these systems, ask for help if you need it!*

- Use the **compare1.m** file (or a modification of it) to plot the results of both the simulation and the real system on two nice, neatly labeled graphs *using the subplot command.* **<u>You need to compare both the position and the velocity of both of the carts.</u>** *(Note: the steady state error is likely to be off. Placing the poles farther away can reduce this error for these systems.)*

- Repeat the above steps to design a state variable controller using pole placement to control the position of the ***<u>second cart</u>*** and meet the design specs.