ME360: FUNDAMENTALS OF SIGNAL PROCESSING, INSTRUMENTATION AND CONTROL

Laboratory No. 6 − PID Control and the Parker-Hannifin Hydraulic Station

The format of the lab is quite a bit different from your previous labs. Not as many details are given to you. Think of this lab as testing some of the knowledge and software experience you received in the previous lab. In Lab 5 you implemented a PI control for the DC motor in the Control Systems Lab. This week you will add derivative control and implement a PID controller for a hydraulic cylinder in the MechSE hydraulics lab.

Hydraulic Station Precautions

***EXTREME CAUTION** should be followed when working with the Station. Common sense safety practices should be followed and close attention paid to the instructor at all times.*

***READ** the following safety guidelines before attempting the procedure.*

- **Your TA will energize the station after checking the hydraulic circuitry!**
- Check all connections to valves and actuators, including quick connect couplings. Verify that they are securely connected and that the safety lock on the quick couplings is activated.
- Do not rub your eyes with even a little of the hydraulic fluid on your hands. Use the towels in the lab to remove the fluid from your hands and then wash your hands when you are done hooking up the station.
- Do not exceed system pressure of 300 psi for this experiment.
- Do not disconnect any hoses while the system is running.
  - Routinely check the system for leaks. Shut-down the system immediately if oil leaks occur.

PID control of a hydraulic piston-cylinder assembly

Loading real-time SIMULINK model

- Login to your station's PC, start MATLAB and enter the "`simulink`" command at the MATLAB prompt. The SIMULINK library window should appear after a number of seconds.
- Using Windows Fille Explorer, browse to the folder C:\matlab\ME360. Create a new folder inside C:\matlab\ME360 and name it your netID. Then browse to the folder N:\HydraulicsLab\ME360. Copy the folder "Cylinder" to your NetId folder you created on the C: drive.
- Inside Matlab, change Matlab's current directory to the "Cylinder" folder you just copied. Run the Simulink model by typing "cylPIDshell".
- Add the correct blocks to this Simulink diagram to create a PID controller for the piston system. First start with the Ki and Kd gains zero. As you look at the block diagram given to you, determine what signal is the feedback and what signal is the control input. Lab 5's lab document should help you with this.

Setting the reference command signal

- Double-click the signal generator block in the model. This block sets the reference input R(s) to the system. The reference command signal is the desired system response, or in our case the desired position of the piston. Verify that the reference signal is a square wave with amplitude 1.0 inch and a frequency of 0.1. When your controller implementation is running this reference will cause the piston to step back and forth from -1 inch to 1 inch every 5 seconds.

Setting the A/D and D/A sampling rate

- Choose "Model Configuration Parameters" under the "Simulation" drop-down menu.
- Enter/(Verify) "0.001" seconds in the "Fixed step size" text box. Then click "OK". This sets a sampling rate of 1 / 0.001 s = 1000 samples per second.

Implement a PID controller to regulate the position of the hydraulic cylinder. Here you will need to add the necessary Simulink blocks to implement the controller. As in the prelab, implement the derivative with the transfer function 100s/(s+100). Make sure to add at least two “Scope” blocks to monitor the position or height of the cylinder and the desired position input also call the reference input.

Building and Running your Simulink model

- Select the Build button at the top right corner of your Simulink model. This will start the build process of your Simulink model. It can take up to 45 seconds or so for the model to build. When the pop up build status windows says Code Generation Complete, click OK. Then in the Simulink model press the “Connect to Target” button. When you see data plotting to the Simulink Scopes, click the “Run” pushbutton in your Simulink model. This will start you controller running. When you want to stop your controller you can click the “Halt” button and disconnect from target.

Connecting the PC A/D and D/A hardware to the EHCB

(This should already be done for you. This is here for students doing the lab at a make-up time)

- Locate the MW2000 signal conditioner with analog and digital inputs and outputs. Connect a coaxial cable from Analog Output Channel 1 to the electrohydraulic valve EHV input located on the bench top close to the piston. This allows us to send a control signal to the valve.
- Connect another cable from the pot connector located on the left panel of the EHCB to Analog Input Channel 6. This allows us to sample and feedback the current position of the piston.

Controlling the piston-cylinder assembly

- Start the hydraulic Station and set the oil pressure to 300 psi. Ask your TA for help here.
- Start your K_p gain at 1 and K_i and K_d at 0.
- Start your real-time controller as stated above in the “Building and Running” section abovoe. Open your Scope blocks and observe the response. Does the system response track the input? Is there overshoot and/or steady-state error in the response?
- Slowly increase the proportional gain (steps of 1 at first then steps of 5). How does the proportional gain effect the rise time, settling time, overshoot and steady-state error in the transient response? Is there an upper limit on K_p above which the control has an adverse effect on the system's response? If yes, what is the upper limit value?
- Now take some time and experiment with different values of K_p, K_d and K_i. Use the following limits as a guide $0 < K_p < 50$, $0 < K_d < 5$, $0 < K_i < 5$. Try just a PD controller by zeroing K_i. Note effects of increasing K_p and K_d. Try just a PI control by zeroing K_d. (You will find that with integral control added you will also need a larger K_p gain to allow the system to settle in less than 5 seconds.) Note observations.
- Tune the PID control gains to achieve $e_{ss} = 0$, %OS = 0%, $t_r \approx 1.5$ s. Record the gain values that best meet these performance characteristics. You may find that some the PID gains can be kept at zero and still achieve these specifications.
- Tune the PID control gains to achieve $e_{ss} = 0$, %OS = 1%, $3.0 \le t_r \le 4.0$ s.

- Draw conclusions about the effect of K_p, K_i, and K_d on rise time, settling time, overshoot and steady-state error. Make sure to note these in your datasheet.