EEE 342 **Spring 2019**

Lab 2 - PI Controller Design by Using Root Locus

As we have done in first lab, we will identify our systems, but this time, step responses of open loop DC motors will be obtained, and a first order approximation will be used. As in first lab, initially, Simulink files and hardware communication need to be configured. You can check manual of the first lab to configure Simulink files of this lab (sampling time, target hardware, serial baud rate etc.).

Part-1: System Identification in Time Domain

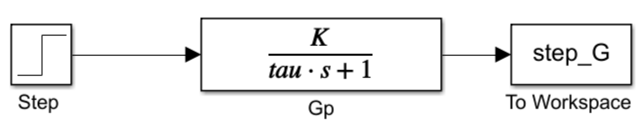
1. Download `lab2_step_resp.slx` and `lab2_step_read.slx` from moodle. Configure these files as you did in first lab (you can use manual of first lab).

2. Adjust the input as $r(t) = 4u(t)$ and obtain the response from the hardware. To receive data you need to follow same steps as in first lab (build `lab2_step_resp.slx` file and run `lab2_step_read.slx` file) Save this result as `step_4.mat` from workspace.

3. Fit a first order approximation on the received data. To do this, you will need to solve the following equation (there are 2 unknowns, so 2 samples would be enough)

$$Y(s) = \frac{K_{step}}{s}\frac{A}{s+B} \rightarrow y(t) = \int_{-\infty}^{t} K_{step} u(\tau) A e^{-B(t-\tau)} d\tau$$

4. Implement the estimated first order approximation ($G_p(s) = A_1/(s+B_1) = K_1/(\tau_1 s + 1)$) in Simulink, and use a step input with an amplitude of 4.



5. Adjust the input as $r(t) = 5u(t)$ and obtain the response from the hardware. Save this result as `step_5.mat` from workspace.

6. Fit another first order approximation as you did in third step.

7. Implement the estimated first order approximation ($G_p(s) = A_2/(s+B_2) = K_2/(\tau_2 s + 1)$) in Simulink, and use a step input with an amplitude of 5.

8. As you did in preliminary work (differently, τ is calculated by using geometric mean), model your system as

$$G_p(s) = \frac{K}{\tau s + 1} \times \frac{1 - 0.005s}{1 + 0.005s}, \quad K = \frac{K_1 + K_2}{2}, \quad \tau = \sqrt{\tau_1 \tau_2}$$

Check-1 Plot received data and response of first order approximations on top of each other (One figure for $r(t) = 4u(t)$, another for $r(t) = 5u(t)$) and show the figures to one of the TAs. Also, show your resulting transfer function ($G_p(s)$) for your plant to TA.

Report In your reports, comment on difference between those two approximations. Why are they different? How does this difference affect system modelling and controller design? Also, compare your resulting plant with the estimated plants in lab-1. Comment on differences. Which one is more accurate?

Part-2: PI Controller Design by Using Root Locus

As you have done in your preliminary work, you will need to design 3 different PI controllers by using the same procedure.

1. Define your plant ($G_p(s)$) in Matlab by using `tf()` command.
2. Since the controller is PI (i.e. $G_c(s) = K_c(-s/z+1)/s$), you will need to check root loci for z values in the following range: `z=-100:0.1:-1`.

   (i) By using `rlocus()` command in Matlab for each z value, save minimum of maximum real parts of closed loop poles and corresponding K values in an array.

   (ii) Plot the minimum of maximum real parts of closed loop poles with respect to z values. Choose the z_1 value at global minimum of this graph.

   (iii) Plot the root locus for chosen z_1 value. Show the point where the closest poles to imaginary axis have minimum real parts (it should also show the corresponding gain (K_c)). Choose $z_2 = z_1/2$ and $z_3 = z_1/3$.

Check-2 Show the root locus for $z = z_1$ and the graph you obtained in step-(ii) to one of the TAs.

3. Design the closed velocity loop in Simulink. Use $G_c = K_c(-s/z_i + 1)/s$ as your controllers ($i = \{1, 2, 3\}$) and G_p as your plant. Show the block diagram in your report.
4. Use Step block to provide the following input: $r(t) = 50u(t)$. Plot the step responses of 3 various systems on same figure. Note that you will need to use Simulink for your results. The results obtained from `step` command of Matlab will not be accepted. Use legends and labels to show your results clearly. Comment on your plot.

Check-3 Show the plot you obtained in step-4 to TAs to receive your third check for this lab.

Report Since this part is just repetition of the preliminary work, you can just show the plots in your report (Indicate samples you used on root loci for each z_i to show how you found K_c values). Also, show the PI controllers you designed.

Part-3: Controller Implementation on Hardware

1. Download `lab2_CL.slx` from moodle. Configure this file as you did in first lab (you can use manual of first lab). Implement the PI controller for three various cases ($z_1,\ z_2,\ z_3$).

2. Use `lab2_step_read.slx` to receive data from hardware for closed velocity loop step response.

3. Save each received data (totally 3) as **`result_z1, result_z2, result_z3`** respectively.

4. Download `lab2_LPF.slx` from moodle. This Simulink file includes a low-pass filter to filter raw data to provide visually better responses. Use **`result_zi`** as inputs to those LPF and plot outputs (f_1, f_2, f_3) in one figure:

$$\texttt{result\_zi} \rightarrow LPF \rightarrow f_i$$

Check-4 Show the last figure to one of the TAs.

Report Show both raw data and filtered data in your reports. Comment on the final figure. Does your result match with the claim we used in preliminary work (about settling time)? Also, compare hardware results with simulation results.

In your report, you are expected to explain the work done in order. It needs to include all plots you drew, all mathematical equations you did (handwritten results will not be accepted), and all the results you obtained in the lab. You also need to comment on each result you obtained between lab checks. All Matlab code should be included in your reports. Do not forget to use report template and write introduction and conclusion parts.