Massachusetts Institute of Technology
Department of Aeronautics and Astronautics
16.06 Principles of Automatic Control
Fall 2003

Lab #2

Issued: Thursday, October 30
Lab work complete: Friday, November 6
Report Due: Thursday, November 13

Please note that there are no formal scheduled lab sessions. The Quanser lab will be open to you the following times. Please complete the lab work by Friday, November 6.
Mon-Fri: 10am-5pm
Wednesday, November 5: 4-7pm (a TA will be present)
Thursday, November 6: 5-8pm (a TA will be present)

Objective

The objective of this lab is to design and implement a pitch controller for the Quanser.

Plant Model

The dominant poles of the plant are given by the second-order model you derived in Lab 1. In addition, we will consider a simple lag pole that is due to the motor. This pole represents a delay in the system – when you change the input voltage, it takes some time for the motor propellers to spin up to their new speed. The motor poles for the Quansers are located as follows:
Quanser 1: $s = -10$
Quanser 2: $s = -7$
Quanser 3: $s = -6.5$
Quanser 4: $s = -9$
Quanser 5: $s = -6$
Quanser 6: $s = -6$

The open-loop system is therefore of the form

$$\frac{\Theta(s)}{V(s)} = K \frac{\omega_n^2}{\left(s^2 + 2\zeta\omega_n s + \omega_n^2\right)\left(sT_m + 1\right)}$$

Note that you must introduce the motor pole in such a way that you do not upset the system gain.

Please note that Quanser 5 is very badly behaved, and we have not yet managed to find a good controller for this machine! If you used Quanser 5 in the first lab, you may switch machines (and use someone else's complex conjugate pair result from Lab 1). Or you may continue with a more challenging design task!

Controller Design

We will carry out the controller design by specifying the closed-loop positions of the complex conjugate poles. Note, however, that these poles might not be dominant in the closed-loop system! We will need to check this once the design is completed.

1. *Preliminary controller design (paper design)*
   Using root locus techniques, design a compensator that places the complex conjugate poles with a damping ratio of $\zeta = 0.6$ and an undamped natural frequency of approximately 2.5 rad/s. You may try any controller you want, but we suggest that you try a PI + phase-lead as follows:

   $$G_c(s) = K'\frac{(s+z_1)}{(s+p_1)}\left(K_p + \frac{K_I}{s}\right) = K_c\frac{(s+z_1)(s+z_2)}{s(s+p_1)}$$

   To help you with the design, we suggest the following placements for p_1 and z_2:

   For all Quansers, $p_1 = -20$
   Quanser 1: $z_2 = -1/2.5$
   Quanser 2: $z_2 = -1/2.5$
   Quanser 3: $z_2 = -1/3.33$
   Quanser 4: $z_2 = -1/2.5$
   Quanser 5: z_2 ?
   Quanser 6: $z_2 = -1/2$

   Using root locus, you should determine the position of the zero z_1 to meet the design specifications.

2. *Controller implementation and re-design*
   Once you have completed your initial design, you should go into the lab and try out your controller. REMEMBER TO BE READY WITH THE KILL SWITCH IN CASE YOUR CONTROLLER DOES NOT HAVE THE DESIRED EFFECT!!!

   Use the file C:/MATLAB6p1/work/16.060/base/lab2.mdl – make sure you copy this file to your own directory and work there. You will need to add the controller block and the feedback path then build you model as we did in Lab 1.

   Adjust the controller gain and/or controller zeroes as necessary to achieve a satisfactory response to a 20 degree pitch angle step input. Record your final controller and save a plot of the closed-loop step response. Please feel free to play around with other controllers (see below for a nice Matlab tool that can help with this).

3. *Closed-loop system analysis*
   You should determine the final positions of all the closed-loop system poles (that is, the theoretical positions according to the model and the final controller design). While you are adjusting your controller, you may find the Matlab SISO Design Tool useful. To access this tool, type "sisotool" at the Matlab prompt and take the following steps.
   a) Define all your transfer functions in Matlab. A useful trick here is to first type "s=tf('s');". This defines the variable 's', then you can enter your transfer functions the way we write them, e.g. "G=1/(s+10)" etc.
   b) In the SISO tool window, click on File/Import. You should now see a list of available transfer functions and a standard block diagram.
   c) You can select appropriate transfer functions for the plant and controller by choosing the Matlab variable and then clicking on the arrow to put it into G, F, H, or C.
   d) The View menu now lets you choose to look at various system plots, including a root locus diagram. The neat thing about the root locus plot is that it lets you click on the controller zeroes and move them around. It also lets you change the controller gain and tells you the corresponding closed-loop pole positions (the pink squares). You can play around with the various options to get the system data that you need.

Lab Write-Up

Write a lab report as for Lab 1, including an Abstract, Introduction, Results, Conclusion and Appendix (Apparatus/Procedure is not necessary). Please include your original controller design working in an appendix. In the main body of the report, be sure to present your final controller design, the final theoretical closed-loop pole positions (for all system poles), the closed-loop dominant modes, and the final closed-loop step response. You should discuss the characteristics of your controlled system, including transient and steady-state behavior. You should also discuss the design trade-offs you had to make.

Each member of your group should do a write-up. The lab should be performed in the same team as for Lab 1, but the analysis and write-up should be your own work.