ESE 406 - SPRING 2011
HOMEWORK #4
DUE 16-Feb-2011 (21-Feb-2011 with late pass)

Problem 1 Submit solutions to the following problems from the textbook:

i. Problem 3.28. Answers:

a. $2.6 \leq \omega_n \leq 4.6$, $0.6 \leq \zeta \leq 0.9$

b. $K = 2$, $K_I = \frac{13}{4}$

c. Make a simple argument using the closed-loop denominator.

ii. Problem 3.31. Answers:

a. $$\frac{\Theta(s)}{T_c(s)} = \frac{1.667 \times 10^{-6}}{s(s + \frac{1}{30})}$$

b. $$\frac{\Theta(s)}{\Theta_r(s)} = \frac{1.667K \times 10^{-6}}{s^2 + \frac{1}{30}s + 1.667K \times 10^{-6}}$$

c. $K < 477$

d. $K \geq 304$

e. Calculations seem to be pretty good.

Problem 2 In hovering flight, the response of the vertical velocity, w(t) [feet per second], of a helicopter to pilot "collective stick" input, u(t) [inches], is reasonably modeled by a first-order transfer function:

$$\frac{W(s)}{U(s)} = \frac{A}{\tau s + 1}$$

The pilot inputs and vertical velocity response of a Bell 206 Jetranger (shown below) were recorded in flight[1] and are provided in the Matlab file `HW05.mat`. These data are loaded into the Matlab workspace with the following command:

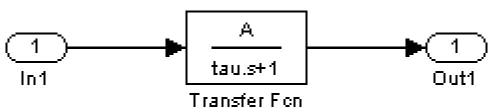
```
>> load HW04.mat
```

The workspace now contains vectors `t`, `u`, and `w`.

[1] Actually, the data were recorded from Microsoft's Flight Simulator (Version X), using the flight data recorder available online at www.fs-recorder.net. The data were gathered at an airfield near Seattle, with Mount Rainier in the background.

a. In this part of the problem, you will determine the best values of A and τ to match the flight data. A simple theory suggests that reasonable guesses would be A=20 [fps/inch] and τ=4 [sec]. You may use these as starting guesses.

One way to determine A and τ is to create a simulink model that looks like this:



Unlike our previous Simulink models, this one receives external inputs and produces external outputs. After setting the values of A and τ in the Matlab workspace, we can run this model from the Matlab command prompt:

```
>> [tsim,xsim,ysim]=sim('HW04a',[],[],[t u]);
```

The first argument of `sim` is the name of the model. The next two arguments are optional arguments that we won't use. The last argument defines the input. You should use Matlab's `help` function to learn more about the `sim` command[2]. You can compare the model to the data by plotting:

```
>> plot(tsim, ysim, '-k', t, w, 'or');
```

You should experiment with different values of A and τ to get a reasonably good fit. You will probably find that your "best fit" values are about half of the simple theoretical values. We'll take a guess at why this is so as a comment to part (c) of this problem.

When you have final values of A and τ, make a pretty graph comparing your model (a solid line) to the data (open circles). Be sure to label the axes clearly, include a legend, and include the values of A and τ on the figure.

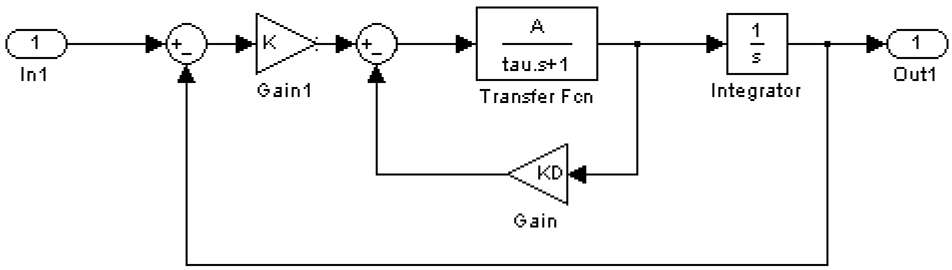
b. Because our model of the system is a linear model, we could also have generated our model of the response without using Simulink at all. The following commands achieve the desired result:

[2] The syntax shown here is outdated. Matlab often updates commands to add additional functions and flexibility. This reflects a strong commitment to increasing the capability of the tool, but also adds a burden of continuous learning to Matlab users.

```
>> y = lsim([A],[tau 1],u,t);
>> plot(t, y, '-k', t, w, 'or');
```

Note that in this case, the output is generated at times corresponding to the input time vector, so we have only one time variable. You don't have to submit anything for part b. This is just for your information.

c. Now let's design an altitude hold system for the helicopter. The block diagram should look like this:



Pilots generally strongly dislike overshoot, so let's go for less than 10% peak overshoot. Rise time of 1 second is a good target for us to design for, though in practice this might be too sporty. Find the values of KD and K that meet these specifications. Submit a graph of the closed-loop step response.

A couple of comments on this system:

1. Notice that the effect of the KD path is to replace $\frac{A}{\tau s+1}$ with $\frac{A}{\tau s+1+AK_D}$, which can be rewritten as $\frac{\hat{A}}{\hat{\tau}s+1}$, where $\hat{A}=\frac{A}{1+AK_D}$ and $\hat{\tau}=\frac{\tau}{1+AK_D}$. In other words, both of these parameters are effectively reduced from their open-loop values. The fact that the identified parameter values were lower than expected suggests that a "velocity damper" is included in the FSX model for the vertical velocity. Low "heave damping" (large time constant) is a very common problem on rotorcraft and adds considerably to the difficulty of the hover task. An "artificial damper" in heave is a common feature of many helicopter stability augmentation systems.

2. It is important that the proportional gain be in the feed-forward path to ensure zero steady error. If we used the architecture shown below, we could still meet the rise-time and overshoot requirements (which only tell us where the closed-loop poles will be). But the steady gain would not be one, as desired.