ESE 406/505 & MEAM 513 - SPRING 2012
HOMEWORK #5
DUE 13-Feb-2013 (Monday, 18-Feb-2013 with late pass)

1. Sketch the unit step response of each of the following transfer functions. Use the graph paper that is included as the last page of this document. On your sketch, be sure to get the proper response at t=0+ (does the system have non-zero value, slope, curvature?), the proper frequency of any oscillation, qualitatively reasonable damping, and the correct steady value. Check your answers in Matlab. The purpose of this exercise is to solidify your understanding of the relationship of time responses to transfer functions--in particular, the locations of poles and zeros.

   a. $\dfrac{16}{s^2+4s+4}$

   b. $\dfrac{4s}{\left(s^2+s+4\right)}$

   c. $\dfrac{9}{\left(s^2-0.3s+9\right)}$

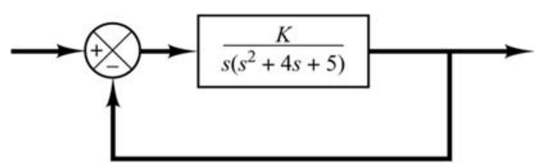
   d. $\dfrac{18s+9}{\left(s^2+s+9\right)}$

   Answers: Modify following code, which works for (c)

```
num = [9]; den = [1 -0.3 9]; G=tf(num,den);
t=[0:0.01:10]'; y=step(G,t); plot(t,y); grid on;
```

2. This is a modified version of problem B-6-14 in Ogata. You may use MATLAB for this problem. In particular, the `rlocus`, `rlocfind`, and `step` commands may be helpful. You might also want to do problem 4 first, to help solidify the root locus concept.

   a. Draw the root locus on K for the system shown in the figure at right. Determine the value of K such that the dominant closed-loop poles have a damping ratio of 0.5. Show this point on the locus.
   Answer: $K = 4.3$.



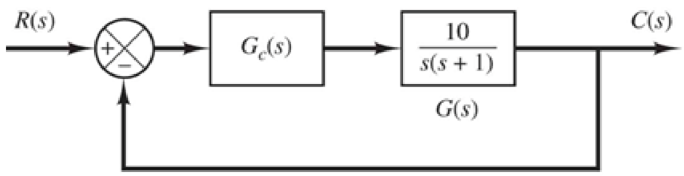
   b. What are the values of σ, ω_n, and ω_d for the second-order pole when K takes this value?
   Answer: $\sigma = 0.625$ rps, $\omega_d = 1.08$ rps, $\omega_n = 1.25$ rps.

   c. Where is the other pole when K takes this value? *Answer*: $p = -2.75$ rps.

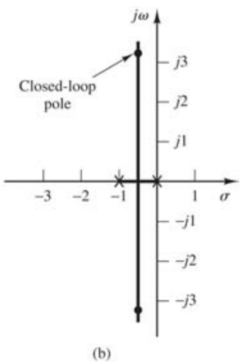
   d. Find the closed-loop unit step response. Do the rise time, peak overshoot, period of oscillation, and settling time match your expectation, based on the second-order pole location?

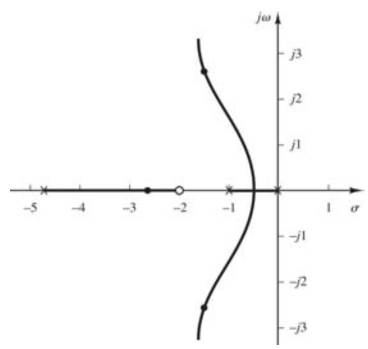
3. Read section 6-5. No, seriously, please read it. It is short and has some very good ideas concisely explained. Graduate students should also read section 6-6. "Lead" and "Lag" compensators are generalizations of PD and

PI controllers that we will talk about next week. (A "lead compensator" is just PD feedback that is passed through a low-pass filter.)



Now look at Example 6-6 in Ogata, the block diagram of which is shown above. The example considers a system that could represent a DC motor used as a position control device--see Problem A-3-9 or the DC motor equations in our lecture notes. For our purposes, the important thing in this example is that proportional feedback doesn't generate enough damping, so we add some dynamics to the compensator to get better damping, as is readily seen by comparing the root loci of Figure 6-39(b) and Figure 6-43:





We will consider the compensator designed by **Method 1**, which is approximately $G_C(s) \approx K\left(\frac{s+2}{s+5}\right)$. Use the `rlocus` and `rlocfind` to verify that $K \approx 1.25$ yields closed-loop poles with damping ratio of about 0.5. Notice that while the damping ratio decreases as *K* increases, the system is stable for *any* positive value of the gain *K*. That is, the poles never cross into the right-half plane, regardless of how large the gain is.

We are interested in the effects of "transport lag" or "time delay" on the closed-loop roots. Suppose that the position control system contains a time delay of 0.2 seconds, so that the actual plant transfer function is

$$G(s) = \frac{10e^{-0.2s}}{s\left(s+1\right)}$$

We want to make a new root locus that shows how the closed-loop poles move with K. The problem is that exact closed-loop transfer function is no longer a ratio of polynomials: we have terms involving $e^{-0.2s}$.

We need to make a polynomial to approximation[1] to e^{-Ts}, so that we can continue to use the standard root locus tools. The most common approach is to use what is called a Pade Approximant[2], such as this:

[1] When we do frequency response analysis, we will be able to represent the effects of delay exactly, with no approximation. But the root locus tool is so useful, and time delay is so ubiquitous, that we need to know how to approximate its effects here.
[2] http://en.wikipedia.org/wiki/Pad%C3%A9_approximant

$$e^{-Ts} \approx \frac{-\frac{T}{2}s+1}{\frac{T}{2}s+1}$$

With this approximation, our plant transfer function becomes

$$G(s) = \frac{10\left(-0.1s+1\right)}{s\left(s+1\right)\left(0.1s+1\right)}$$

With this approximation, we can again use the root locus tools to determine the effects of feedback on the poles.

MATLAB can handle time delay, including Pade approximations, very easily. Here is some code that shows how to generate the root locus of interest in this problem:

```
G = tf(10,[1 1 0]);           % create a tf object G(s)=10/[s(s+1)]
set(G,'InputDelay',0.2);      % add 0.2 seconds transport lag
Gpade = pade(G,1);            % version of G with first-order Pade approximation of delay
Gc = tf([1 2],[1 5]);         % compensator Gc(s) = (s+2)/(s+5)
rlocus(Gc*Gpade);             % make the root locus
axis([-12 8 -12 12]);         % see the important stuff
axis('equal');                % don't distort the x and y axes
```

Some questions:

a. What value of K yields a closed-loop damping ratio of 0.5?

b. What is the damping ratio of this system if the original gain ($K=1.25$) is used?

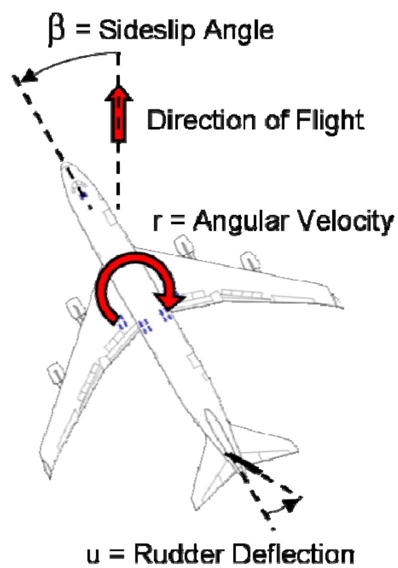
c. What value of K will result in neutral closed-loop stability (pole on the imaginary axis)?

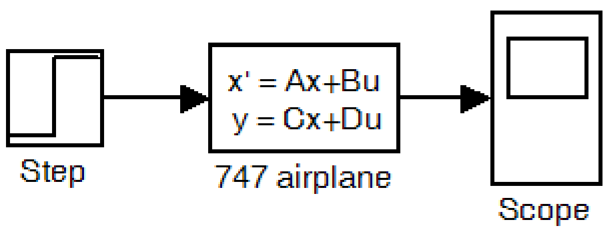
d. Make a graph that shows the closed-loop unit step response. Include one line for the nominal system and a second line for the system with time delay. For both systems, use the gain required to get a closed-loop damping ratio of 0.5.

Problem 4 In this problem, we will study the development of a yaw damper for the 747 aircraft in high-altitude, high-speed cruise[3]. By walking through a real design problem in considerable detail, you will see how the vocabulary and analytical tools you have learned so far enable you to understand almost all of the key considerations in the design. The following specify the state-space linearized model of the dynamics:

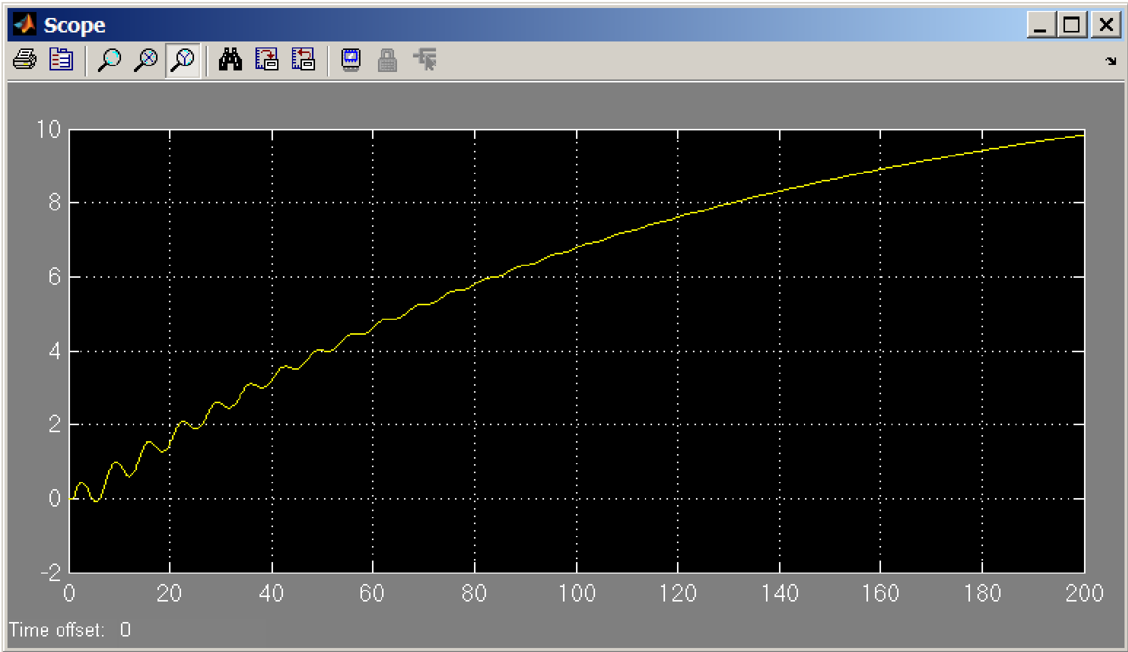
$$A = \begin{bmatrix} -0.05 & -1 & 0 & 0.04 \\ 0.84 & -0.15 & -0.01 & 0 \\ -3.00 & 0.41 & -0.43 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad B = \begin{bmatrix} -0.01 \\ 0.46 \\ -0.11 \\ 0 \end{bmatrix} \quad C = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} \quad D = \begin{bmatrix} 0 \end{bmatrix}$$

The control input is the deflection of a small flap at the back of the vertical tail, called the rudder, in radians. The state vector is $\underline{x} = \begin{bmatrix} \beta \\ r \\ p \\ \phi \end{bmatrix}$, where β is the sideslip angle, r is yaw rate, p is roll rate, and ϕ is roll angle. The scalar output is the yaw rate, in radians/second. Let's set up a SIMULINK model and see what the step response looks like (the "747 airplane" is a "state space" block from the "Continuous" library; the "step" is in the "sources" library; the "scope" is in the "sinks" library):





[3] Etkin and Reid, *Dynamics of Flight*, 3rd Edition, 1996. Numbers simplified slightly from original values. A much more comprehensive discussion of the 747 flight dynamics is given in Heffley, R.K., *Aircraft Handling Qualities Data*, NASA CR-2144, December 1972. Page 215 of this document contains a block diagram of the yaw SAS ("Stability Augmentation System") of the 747. The document is available online:
http://www.robertheffley.com/docs/Data/NASA%20CR-2144--Heffley--Aircraft%20Handling%20Qualities%20Data.pdf

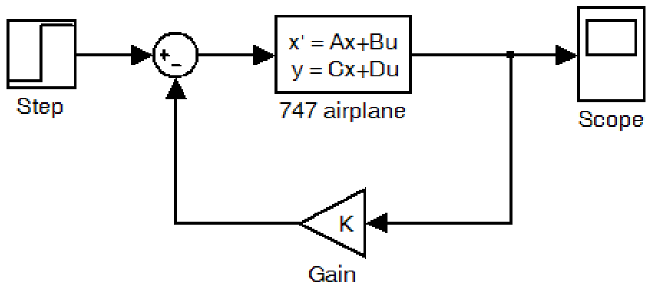


We know our 4-state system has to have 4 poles. In this response, it looks like we can see 3 poles: one very slow first-order mode and one very lightly damped oscillatory mode (2 poles, called the "dutch roll") with a period of about 6 or 7 seconds (just count cycles between 20-second grid intervals). Evidently, the 4^{th} pole has a nearby zero in the yaw-rate transfer function, making it difficult to see in the response. While you aren't expected to have an intuition for the magnitudes or character of the response for this complicated example, you should recognize that it is not a simple second-order system. Use the results shown above to confirm that your SIMULINK model is implemented correctly.

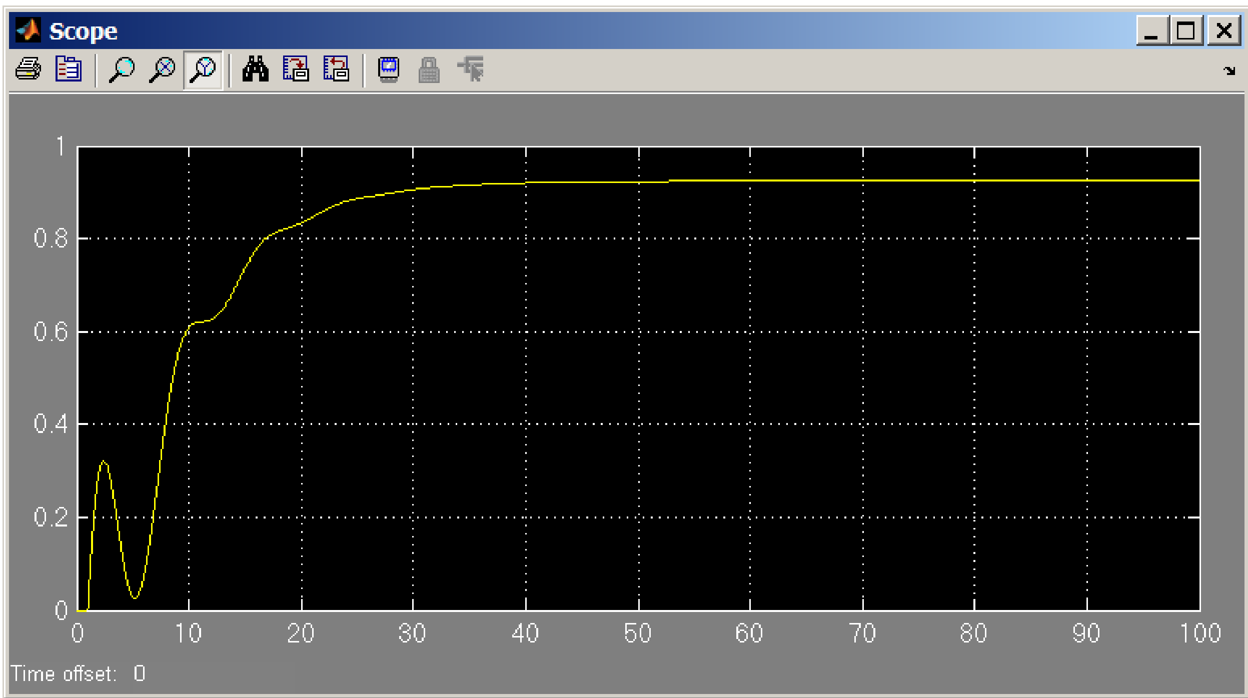
Virtually all airplanes and helicopters have modes that are qualitatively very much like those seen here. The very slow mode, called the "spiral mode" turns out to be not very important to the pilot—it is quite easy for her to compensate for that one herself[4]. But the oscillatory mode is very objectionable and would create serious ride comfort issues for passengers, as well as problems with control in crosswinds. As a result, perhaps the highest priority element of a "stability and control augmentation system" (SCAS) is a "yaw damper" that will improve the damping of this "dutch roll" mode.

The simplest yaw damper uses a measurement of the yaw rate (using a device called a "rate gyro" which is inexpensive and quite reliable) to generate a proportional rudder command in the opposite direction, which is summed with the pilot input, $u = u_{PILOT} - Kr$. Let's go back to our SIMULINK model and add the feedback path:

[4] The pilot can compensate for the slow "spiral" mode when her attention is properly focused on flying the airplane. This mode can be dangerous if the pilot is distracted or flying in conditions with poor visibility. An inadvertent rudder input was evidently the primary cause of the crash of the private airplane being flown by singer John Denver that resulted in his death. (`http://www.check-six.com/Crash_Sites/John-Denver-N555JD.htm`)



And now the step response looks something like this (depending on the value of K)



Note that the steady-state response to the step input is much smaller than it was before. This isn't good, because it suggests that our damper is taking away the pilot's steady-state control authority. Also, when the pilot wants the airplane to be turning, our yaw rate feedback will be force the pilot to hold a large input, which will be uncomfortable. (The pilot controls the rudder with pedals at her feet; the forces required to move the pedals can be quite large, by design, to prevent unintended inputs.) We need to do something other than just proportional feedback.

Solving this problem is straight-forward: we need a high-pass, or washout, filter. It turns out that we also want to use a low-pass filter so that high-frequency noise and aircraft structural vibrations[5] don't get into the feedback signal. Therefore, the actual 747 SCAS used a "yaw damper" that looked like this (once we include dynamics in our feedback, it is much easier to describe the design using transfer functions):

$$U(s) = U_{PILOT}(s) - KG_F(s)R(s)$$

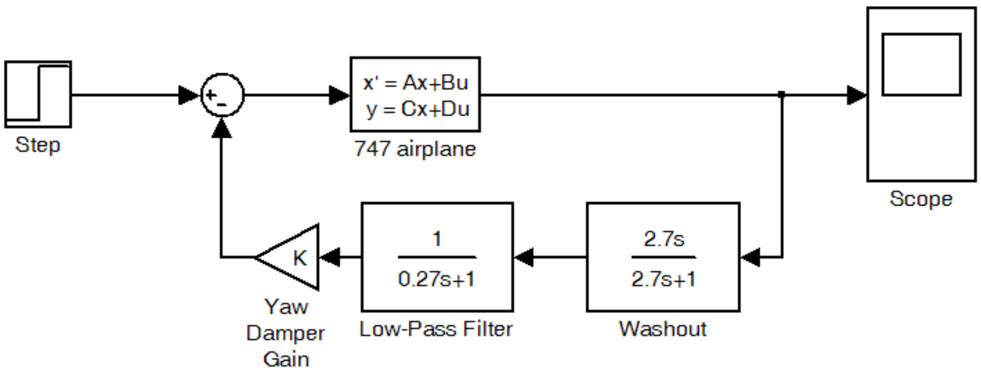
where the filter transfer function is

[5] Structural vibrations have lower frequencies on bigger airplanes. On a very large airplane, the fuselage might have bending modes in the range of 1 to 2 Hz, so it might be easy for the pilot or the feedback to excite these modes.

$$G_F(s) = \frac{\tau_1 s}{\tau_1 s + 1} \frac{1}{\tau_2 s + 1}$$

In the filter, the washout time constant, τ_1, is 2.7 seconds and the structural filter time constant, τ_2 is 0.27 seconds. To better understand the effects of this filter, we can use the frequency response analysis that we will learn after spring break. For now, we will just be content with knowing that the high-pass filter (τ_1) gets rid of the feedback in steady state and the low-pass filter (τ_2) prevents problems with the structural modes.

In SIMULINK, our system now looks like this (the filters use the "transfer function" block from the "continuous" library):



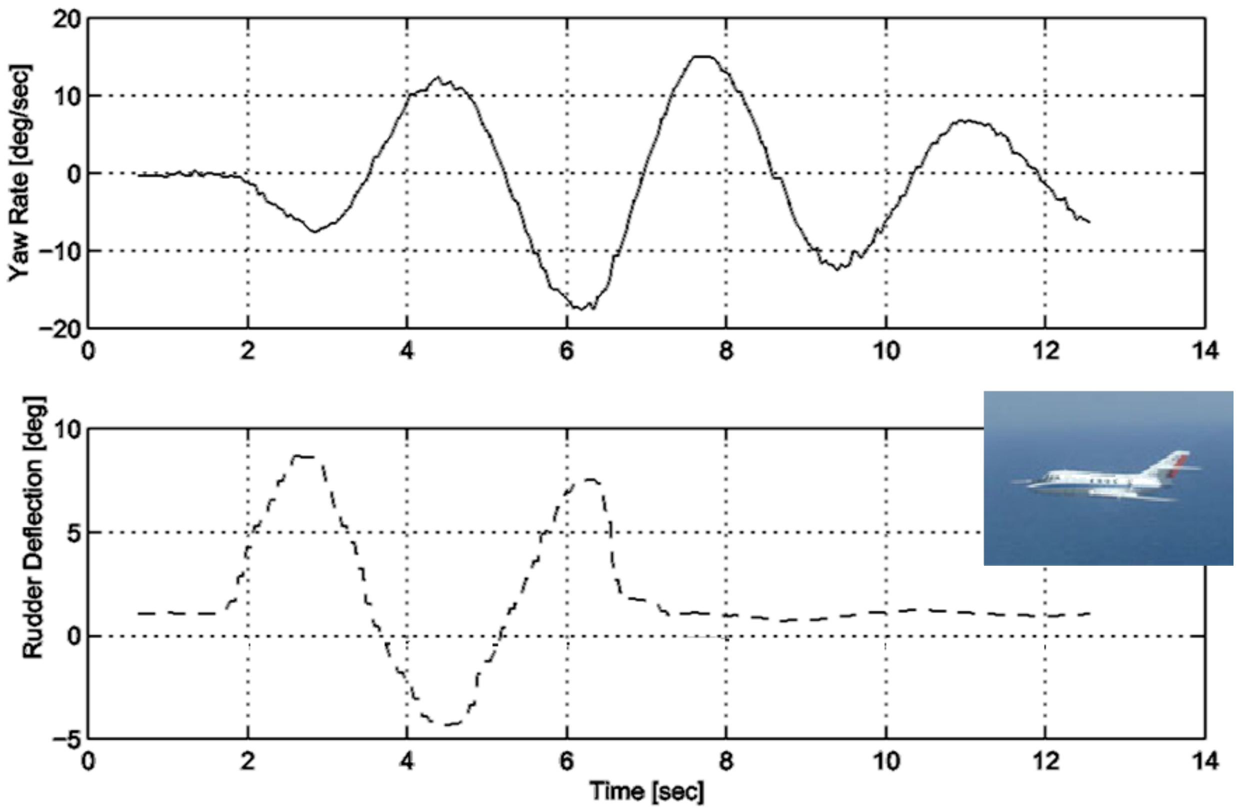
You should look at the step response and confirm that we have solved the steady-state response problem—we get the same steady-state response that we got with the open-loop system. Our yaw rate feedback certainly improved the damping of the mode. But it is difficult to quantify how much, because of the slow first-order response of the spiral mode that is super-imposed. We could update our state-space equations to include the effects of feedback, but for now let's pretend that we are doing flight tests, and we need to confirm the increased damping from our "test data." If we want to get a quantitative measure of dutch roll damping, we are going to have to do something more clever with the input than a step.

We need to figure out how to excite primarily the dutch roll mode, without exciting the slower mode, so that we can get accurate quantitative estimates of how the pole is moving as we vary the gain. Our closed-loop system can be written as $R(s) = H(s)U_{PILOT}(s)$. $H(s)$ will now have 6 poles, because we have added two poles in the feedback path. For any $U_{PILOT}(s)$, we could do a partial-fraction expansion on $R(s)$ and we would find terms for each mode in $H(s)$. We want to pick a $U_{PILOT}(s)$ that will be very large when s is equal to the dutch roll root, but small at the small value of s that corresponds to the slow mode. A typical input to choose is called a "doublet", which involves step inputs in opposite directions, separated by a time delay of approximately half the period of oscillation of the mode.

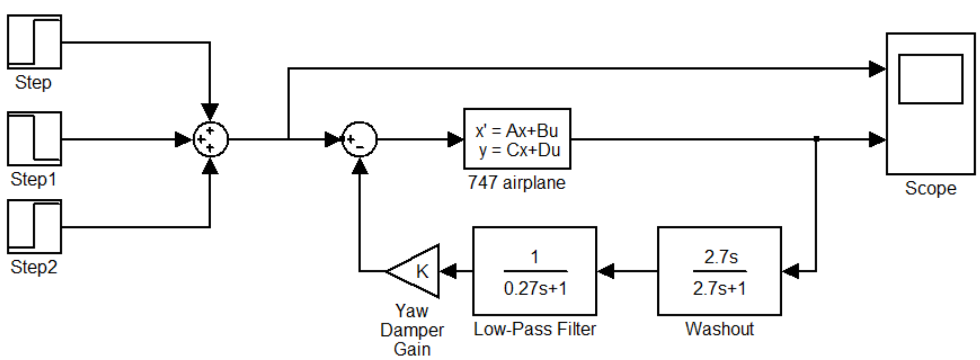
A real dutch roll flight test event showing excitation of the dutch roll mode by a "triplet" type input to the rudder is shown below[6]. Note that the pedals are not moving for $t \geq 7$, and the lightly damped dutch-roll mode is clearly evident. It is a shame that they cut off the data record so early.

---

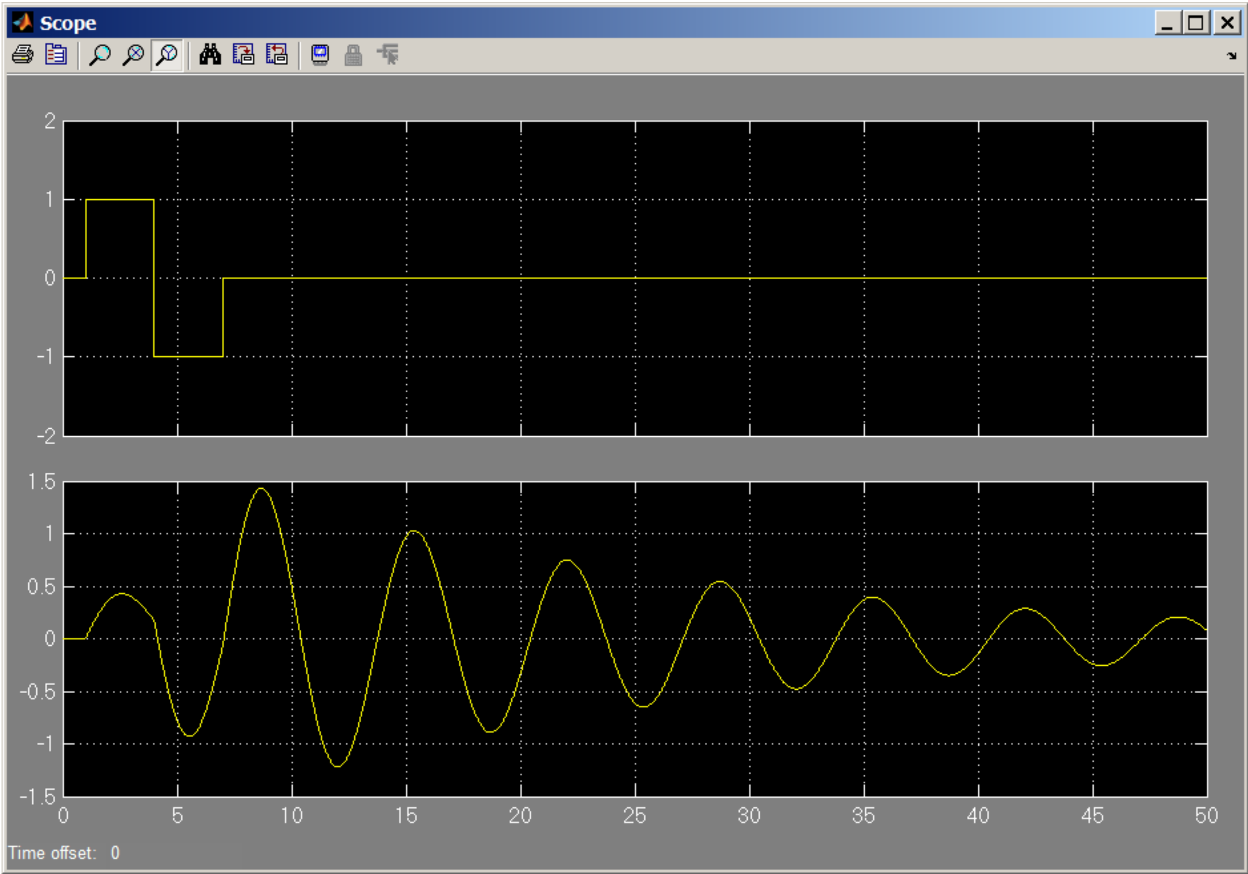
[6] I downloaded these data from the Technical University of Delft about 10 years ago. I lost the link and reference, but that school does still maintain an experimental flight test program.



We can build an idealized doublet-type input in SIMULINK by summing 3 step inputs (a step of +1 at t=1 followed by a step of -2 at t=4 followed by a step of +1 at t=7). Our model now looks like this:



For a small value of the feedback gain, the input and the system response looks like this:



Notice that the doublet input has excited a strong dutch roll response, but almost none of the slow spiral mode. Now we just have to figure out how to get an accurate quantitative measure of frequency and damping ratio from this time response. Determining the damped natural frequency, ω_d, is very straight-forward: $\omega_d = \frac{2\pi}{T}$, where T is the period of the oscillation. Of course, we want to measure the period after the doublet has ended, as that best reflects the system modes. The damping ratio can be found from a technique called "logarithmic decrement" which is described on Wikipedia as follows[7]:

[7] The math behind the logarithmic decrement is not difficult. You are encouraged to reconcile the time response of a second-order system with the equations given here. Searching for "logarithmic decrement" on google yields thousands of hits, too...

Logarithmic decrement, δ, is used to find the damping ratio of an underdamped system in the time domain. The logarithmic decrement is the natural log of the amplitudes of any two peaks:

$$\delta = \frac{1}{n} \ln \frac{x_0}{x_n},$$

where x_0 is the greater of the two amplitudes and x_n is the amplitude of a peak n periods away. The damping ratio is then found from the logarithmic decrement:

$$\zeta = \frac{1}{\sqrt{1 + (\frac{2\pi}{\delta})^2}}.$$

The **damping ratio** can then be used to find the undamped natural frequency ω_n of vibration of the system from the damped natural frequency ω_d:

$$\omega_d = \frac{2\pi}{T},$$

$$\omega_n = \frac{\omega_d}{\sqrt{1 - \zeta^2}},$$

where T, the period of the waveform, is the time between two successive amplitude peaks.

Applying this technique to the simulation results shown above, taking the first peak to be slightly larger than 1.0 at about 15.5 seconds (***after the input has ended***), the amplitude 4 peaks later, at about 42.5 seconds, is about 0.25. Thus, we have

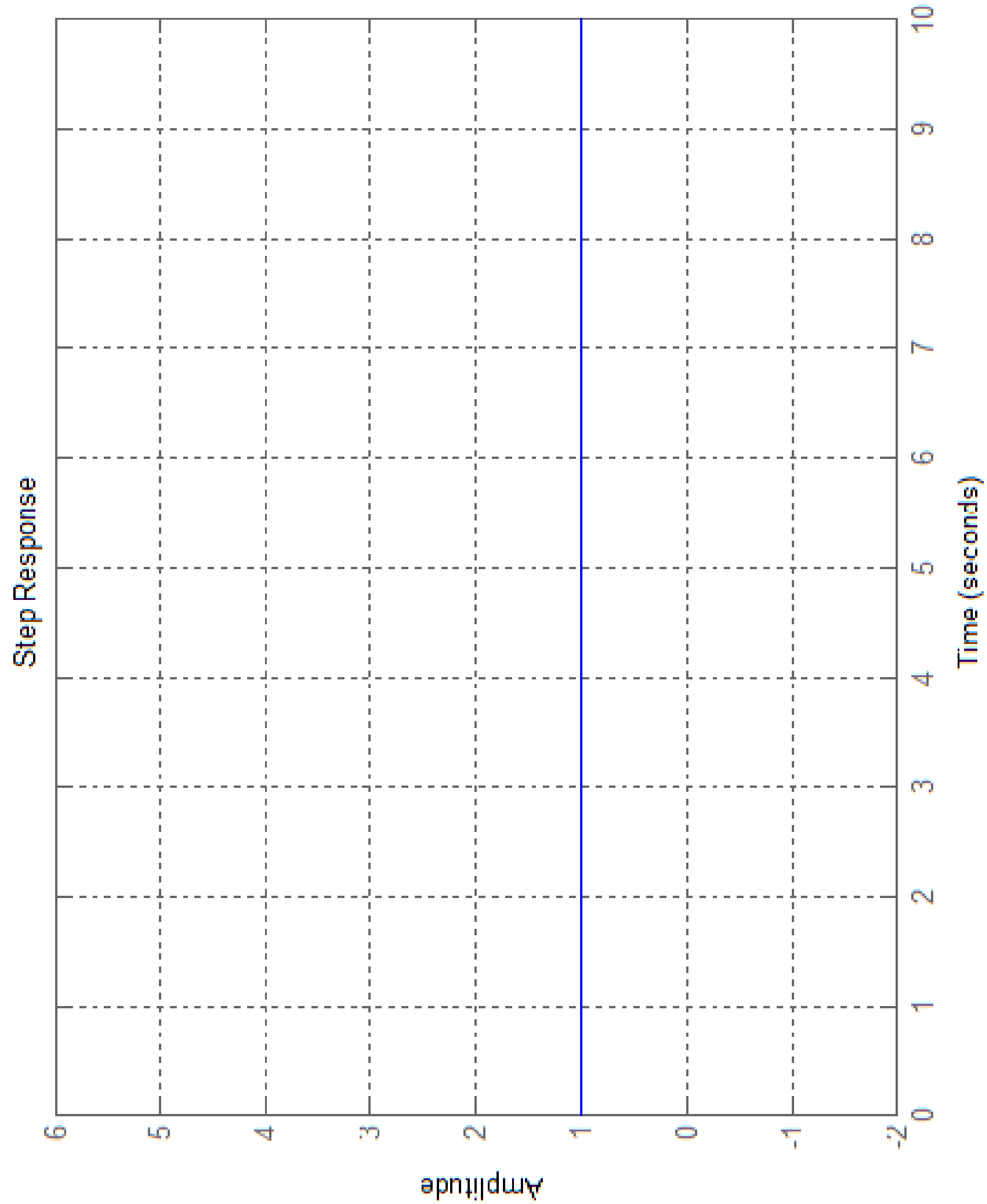
$$\delta = \frac{1}{4} \ln \frac{1}{0.25} = 0.3466$$

$$\varsigma = \frac{1}{\sqrt{1 + \left(\frac{2\pi}{.3466} \right)^2}} = 0.055$$

$$\omega_n = \frac{2\pi}{T\sqrt{1 - \varsigma^2}} = 0.932 rps$$

Natural frequency of 0.932 rps and damping ratio of 0.055 correspond to a pole at $-0.051 \pm 0.931 j$. The exact answer for this configuration is $-0.047 \pm 0.941 j$. You should make somewhat more accurate measurements of the peak amplitudes and the modal period (use the X and Y zoom tools in the Scope window).

***<u>SUBMIT</u>*: Repeat the pole estimation process for each of the following values of K: 0.0 , 0.5, 1.0, 1.5, 2.0, 2.5, 3.0. Make a graph (a "root locus") showing how the dutch roll mode moves in the complex plane as the gain K is increased. The actual value of the gain on the 747 yaw damper is about 1.4. Do you see why?**



Make your own graph paper using Matlab:

```
step(tf(1,1)); grid on; axis([0 10 -2 6]); set(gcf,'Color','w')
```