ESE 505 & MEAM 513 - SPRING 2015
HOMEWORK #5
DUE Wednesday 2015-04-29

1. This is a textbook problem to give us some practice with lead-lag compensators. If you want something like a cookbook approach, google can get you lots of good stuff, like this:

   `lead compensator design using frequency response`

   for which the first hit is pretty good:

   `http://faculty.uml.edu/ahmed_abuhajar/documents/FrequencyResponseLead-LagCompensator.pdf`

   Now here is the problem. For the system shown below design a lead-lag compensator,

   $G_C(s) = K \dfrac{s+z}{s+p} \dfrac{T_1 s + 1}{T_2 s + 1}$ (with $T_1 > T_2$ and $z > p$), that will yield the following properties:

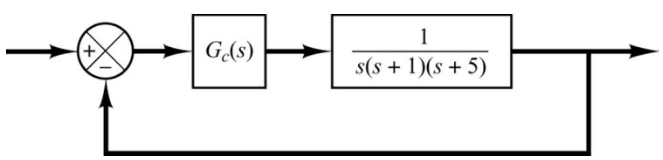
   i. Phase margin of about 60 deg.

   ii. Gain margin of about 8 dB.

   iii. Crossover frequency between 2.0 and 3.0 rps.

   iv. Velocity error constant of at least 20.0 sec^{-1}.

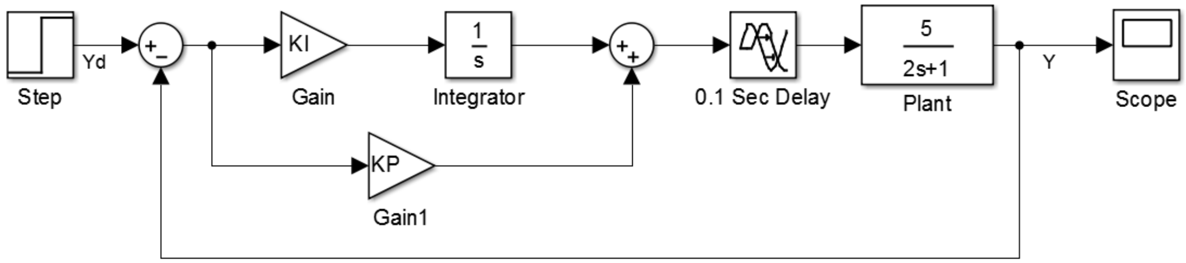
   Submit a bode plot of the loop transfer function that includes your final design. Label the margins. Also include a closed-loop step response.



   Answers: There are lots of possible answers, e.g., $G_C(s) \approx 5 \dfrac{s+0.2}{s+0.01} \dfrac{2.7s+1}{0.13s+1}$. Notice that to achieve the very generous phase margin, we have to use a very high lead ratio, here just over 20 (2.7/0.13 > 20). This much lead is often unrealistic, as it can cause actuator saturation for modest inputs and amplify noise too much. Practical limits on the maximum lead are very system dependent, but they are often the determining factor in the achievable cross-over frequency.

2. Up to now, we have almost exclusively discussed the open-loop bode plot, without much thought for the closed-loop bode. Let's shift gears now to think about the closed-loop frequency response. Consider a simple first-order system, such as shown below. This might represent a temperature-control, such as the glasshouse example we discussed in class, or a vehicle speed control.

   a. Design a P+I controller for this system with a gain margin of 6dB and a velocity error constant[1] of 20. Make a note of the values of phase margin and ω_C and ω_{180}.



   Answer: KI=4 (to achieve desired velocity error constant) and KP=3 (for desired gain margin). The resulting phase margin is 40 degrees. $\omega_C = 7.6rps$ and $\omega_{180} = 15.2rps$.

   b. Now compute the closed-loop transfer function, $G_{CL} = \frac{Y}{Y_d}$. Make a frequency response plot[2] of the closed-loop transfer function, G_{CL}. Note that at low frequency, the frequency response has magnitude 0dB, which implies perfect tracking as desired. As the frequency gets larger, the gain drops off, indicating that we cannot follow very fast commands. The frequency above which we no longer follow commands is typically called the *bandwidth*. The definition of bandwidth is somewhat arbitrary, but the most widely used definition[3] is the frequency at which the closed-loop magnitude first reaches -3dB. What is the -3dB bandwidth for this system?

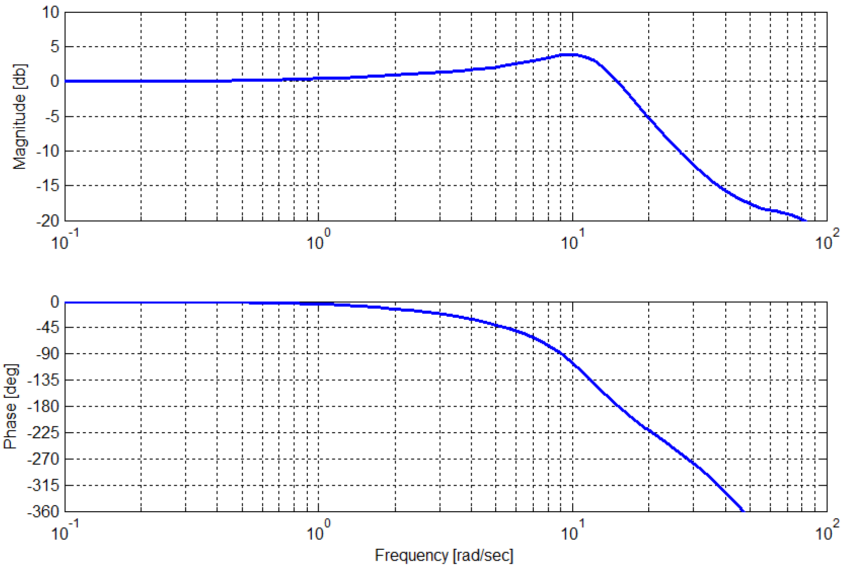
   Answer: As shown on the bode plot below, the bandwidth is about 17.74 rps.

---

[1] The velocity error constant is defined as $K_V = \lim_{s \to 0} sG(s)$, where $G(s)$ is the loop transfer function. Specifying K_V is a way of saying "how high" the low frequency gain must be to ensure adequate tracking and disturbance rejection. Although it is common in textbooks, I have never personally worked on a system with a velocity error constant specification. But it seems like a reasonable way to constrain the design to achieve the low-frequency objectives, particularly for simple architectures of the type contemplated here.

[2] Note that to use MATLAB's `bode` command for the closed-loop transfer function, you may have to use a Pade approximation for the delay. We want an accurate approximation, so use `pade(0.1,4)` to get a 4th-order approximation of the 0.1-second delay.

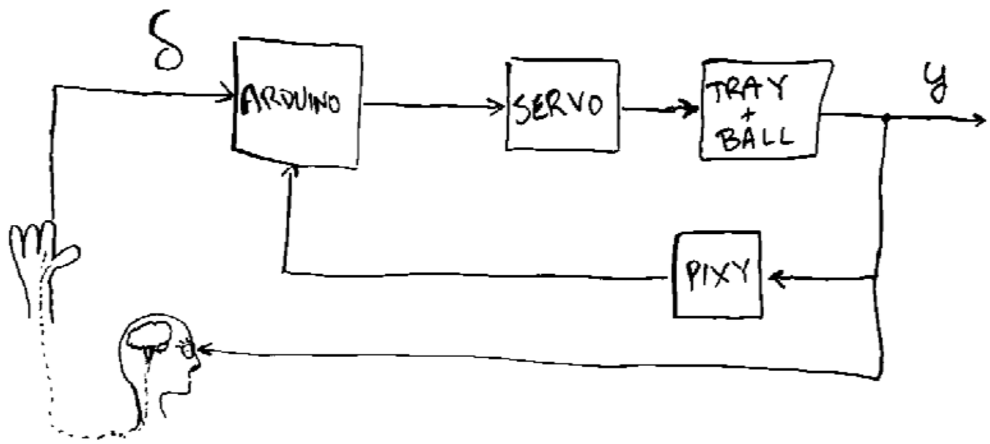
[3] This definition of bandwidth was originally devised for signal-processing (amplifier) applications, but has become quite ubiquitous.



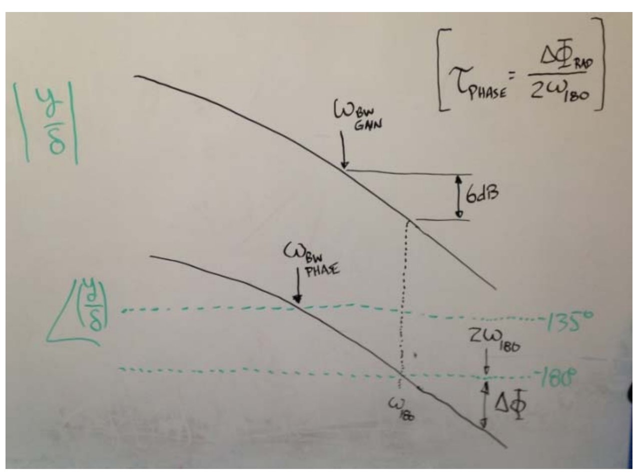
c. Let's take a more critical look at the notion of bandwidth and its relation to ω_C and ω_{180}. In this problem, we found that the usual definition of bandwidth returned a frequency that was larger than either of ω_C or ω_{180}. Does this make sense? Consider the closed-loop response for input frequency equal to ω_{180}. What are the magnitude and phase of G_{CL} at this frequency? You should be sure that you see how these values are related to the open-loop bode plot (from which the margins are computed.) Do you think it makes sense to say that the output is accurately tracking the input at this frequency? Briefly explain.

Answer: When the gain margin is 6dB, we find that $G_{CL}(j\omega_{180}) = -1$. This means that the output has the desired amplitude, but is *exactly out of phase with the input*! This might be fine for an electrical signal processing application, but it is not a useful measure of bandwidth for something like an aircraft roll response--when we want to roll right, the airplane is rolling left!

3. Now let's do some closed-loop frequency response analysis of the One-Two-Red-Blue project, and see if we can get some insight into how easy or difficult it is for a human operator to control the ball position precisely[4]. Once we have closed the loop using measured ball position, we have a system that looks like this:



Here, the Arduino closes the "inner loop" such that $\frac{y}{\delta} = G_{CL}(s)$. The human closes the "outer loop". A very simple model for the human activity is proportional feedback: $\delta = K\left(y_d - y\right)$, where y_d is the desired position that exists in the human's mind. Humans are very adaptable, so we might assume that the human increases the gain, K, until the "outer loop" has either a gain margin of 6dB or a phase margin of 45°. The gain that results in a gain margin of 6dB will cause the 0dB outer-loop crossover frequency, called the "gain bandwidth" to be $\omega_C = \omega_{BW,GAIN}$, while the gain that results in a phase margin of 45° results in a loop crossover frequency, called the "phase bandwidth" of $\omega_C = \omega_{BW,PHASE}$. These quantities are shown below. Notice that the absolute magnitude of $\frac{y}{\delta} = G_{CL}$ is irrelevant (no scale on the y-axis), because we are assuming that the human operator adjusts the gain as necessary to reach the indicated margins.



The higher these crossover frequencies are, the more quickly and precisely we expect the human control to be. In cases where the gain bandwidth is lower than the phase bandwidth, there is some evidence that the

[4] These ideas are based on the Army's ADS-33E specification for precision attitude control of a helicopter in near-hover flight. That these ideas might also apply to the simpler position-control activity considered here is mere speculation. But hopefully it is interesting speculation.

system will be more susceptible to PIO, pilot-induced oscillation. Another PIO indicator is large "phase delay", which is a measure of how quickly the phase decreases at frequencies above the 180-crossover frequency as shown in the figure. Phase delay in excess of about 0.25 seconds is probably undesirable.

Do your best to apply these ideas to your One-Two-Red-Blue system. What are the gain and phase bandwidths and phase delays of your system? Do you think your system was "PIO-prone"?

4. Finally, let's apply closed-loop frequency response analysis to our train project. We are interested in a property known as "string stability", which means that the amplitude of a disturbance at any frequency in the lead train will not grow larger as we move back in the string. One way to analyze string stability is to number the trains, so V_i is the speed of the ith train, and V_0 is the speed of the lead train. Our block diagram analysis should allow us to compute a closed-loop transfer function:

$$H(s) = \frac{V_i(s)}{V_{i-1}(s)}$$

And we will have string stability if the bode plot of $|H(j\omega)|$ is less than 0dB at all frequencies.

You should spend some time to convince yourself that our simple approach makes it almost impossible[5] to meet this criterion! But we should be able to use our analysis to compute the frequency at which $|H(j\omega)|$ has a maximum value, and if we get lucky, we might expect to see that frequency emerge in experiments, at the tail of a long string of trains.

If we want to try to improve things so that we can satisfy the string stability criterion, we probably need to change something about our control approach. Posted to Canvas with this assignment is a technical paper[6] that discusses this problem in some detail. They suggest that the following distance be increased in proportion to vehicle speed, like this (equation 5 in the paper):

$$X_{i,des} = X_{i-1} - L - hV_i$$

where L is the nominal following distance and h is called the "time gap". With this change, it is evidently possible to achieve string stability if the vehicle acceleration is commanded based on PD feedback of position error.

We cannot directly implement these ideas, because we don't have a measurement of vehicle speed, and our control (motor PWM) results in a change in speed, not acceleration. But maybe PD feedback of position to acceleration is like PI feedback of position to speed, and maybe we can increase the desired following distance using an estimate vehicle speed from our PWM, and maybe somehow if we do all that we will get lucky and achieve string stability?

Do your best to apply some of these ideas to the train problem. Videos of many trains on the track at the same time are especially encouraged, particularly if there are oscillations that are related in some way to $\max|H(j\omega)|$!

---

[5] I think (really more like a conjecture) that the string stability criterion amounts to requiring that the Nyquist plot of the loop transfer function stays out of the unit circle centered at the -1 point. Stated another way, the Sensitivity Function must have magnitude less than 0dB at all frequencies. The insight about the fundamental difficulty of achieving string stability is a major payoff for the trouble of doing linear analysis!

[6] R. Rajamani and C. Zhu, "Semi-Autonomous Adaptive Cruise Control Systems", IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, VOL. 51, NO. 5, SEPTEMBER 2002.