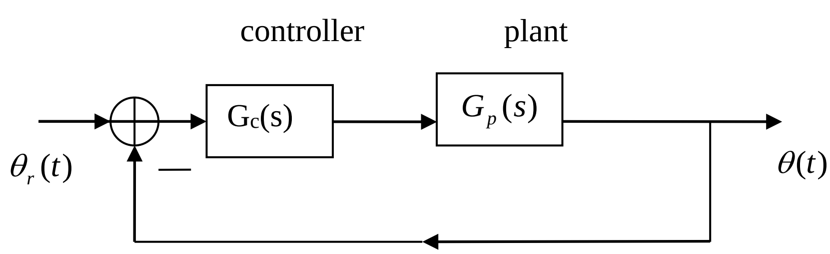
Lecture 8 **Class Example of Pole Placement Using a Lead/Lag Controller**

In this lecture we will complete our time domain approach to controller design. Specifically, we will address two topics:

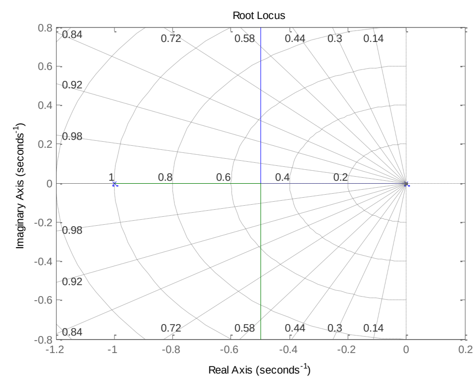
1.Using the *bisector method* of designing a *lead controller*, as opposed to a PD controller.

2.Using a *lag controller* to achieve the steady state error specification without altering the placed closed loop pole.

Consider the command feedback control system shown below.

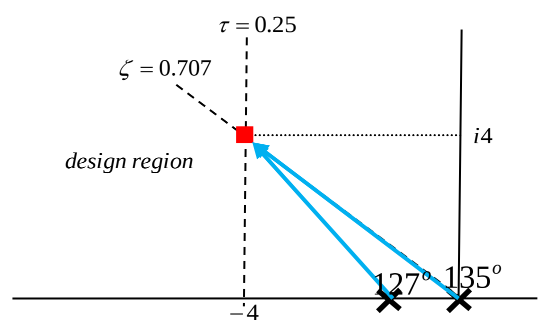


The plant is a DC motor with $G_p(s) = \dfrac{100}{s(s+1)}$ [Degrees/volt].



PROBLEM: Design a controller that will satisfy the following closed loop specifications:
(S1): All time constants less than 0.25 sec.
(S2): All damping ratios greater than 0.707.
(S3) Steady state error for a ramp input $\theta_r(t) = 0.1t$ [degrees/sec.] must be no more than 0.01^o.

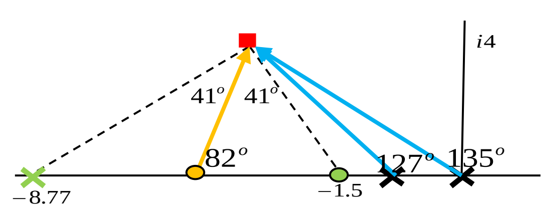
(a) *How about using* $G_c(s) = K$?
Answer: The root locus at the right shows that (S1) cannot be satisfied for any *K*.



(b) The design region associated with (S1) and (S2) is shown at the right. We will design a lead compensator that will pull the root locus in (a) to the left, so that it passes through the closed loop pole location shown in red.

To determine how much angle must be added to satisfy the root locus *angle criterion*: $\sum \phi_z - \sum \phi_p = 0^0 - (135^o + 127^o) = -262^o$.

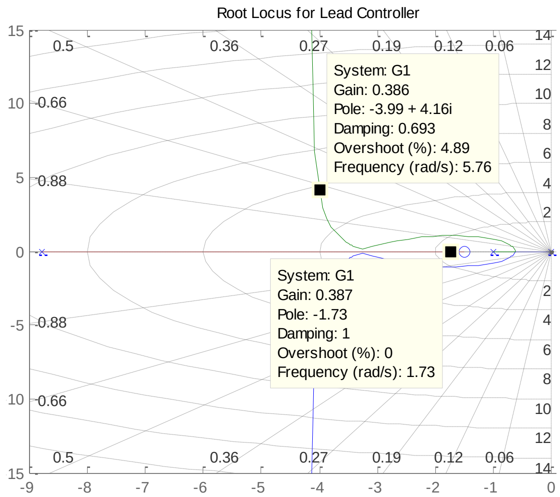
Hence, the controller must add 82^o. Were it to entail to add only an open loop zero, it would need to be located as shown by the orange zero at the right. This would be a PD controller. We (as well as the authors) have pointed out the drawbacks of PD control; namely (i) sensitivity to high frequency noise, and (ii) a high power requirement.

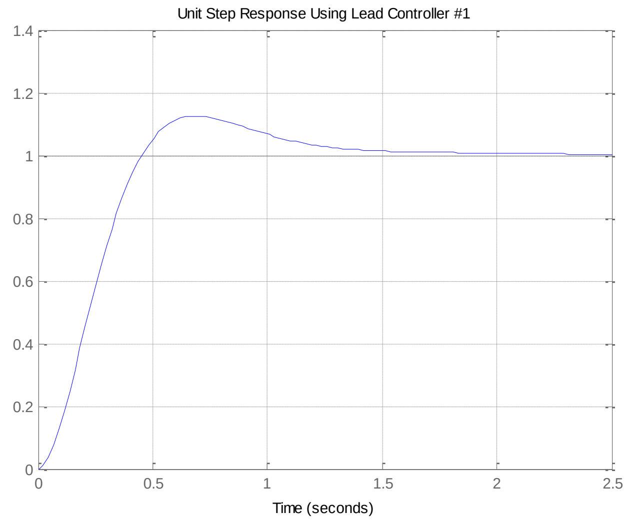


Using the ***bisector method,*** we split the 82^o as shown in green.

This results in: $G_c(s) = 0.39\left(\dfrac{s+1.5}{s+8.77}\right)$. This is called a ***lead controller*** (or compensator). It is because the zero adds more root locus angle than the pole. One could also say it is because the controller zero is closer to the origin than the pole.

The root locus for $G_c(s) = K\left(\dfrac{s+1.5}{s+8.77}\right)$ is shown below. It shows that we will not be able to satisfy (S1) completely. The complex pole satisfies it, but the real pole at -1.73 does not. [NOTE: The root locus is interesting!]





However, the closed loop unit step response shown above suggests that this time constant (1.58sec) does not have much 'presence'. And so, we will go with this lead controller. [**NOTE**: An alternative would be to not use the *bisector method*, and, instead, place the controller zero at -4. There are an infinite number of possibilities ☺]

Before proceeding, it is worth noting that the step response also has significantly more overshoot than one might expect a complex pole with $\zeta = 0.707$ to give. This is due to the presence of the controller zero. Such overshoot can wreak havoc, since it could generate oscillations of the load (in this case, the tracking antenna) that would lead to rapid material fatigue.

***Specification* (S3):** Notice that the closed loop static gain equals 1 (as evidenced in the step response). It follows that the response to a unit ramp input will be finite. To see this, we will now develop the notion of a system *type number* for a *unity feedback* command control system $W(s)$. Suppose that the system input is $\theta_r(t) = t^m$, with Laplace transform $\Theta_r(s) = m!/s^{m+1}$. Then the steady state error is (for <u>$m > 0$</u>):

$$e_{ss} = \lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s[\Theta_r(s) - \Theta(s)] = \lim_{s\to 0} s[1 - W(s)]\Theta_r(s) = \lim_{s\to 0}[1 - W(s)]m!/s^m$$

$$= \lim_{s\to 0}\left[\frac{1}{1+G(s)}\right]m!/s^m = \lim_{s\to 0}\left[\frac{m!}{s^m + s^m G(s)}\right] = \frac{m!}{\lim_{s\to 0} s^m G(s)}$$

In order for e_{ss} to be finite, the open loop $G(s)$ must have at least m poles at the origin. If it has more, then this error will be zero (assuming the CL is still stable). If it has exactly m poles at the origin, then this error will be nonzero but finite. In this case, the closed loop system is said to be a *Type m* system. We will now apply this to specification (S3): For $\theta_r(t) = 0.1t$, we have

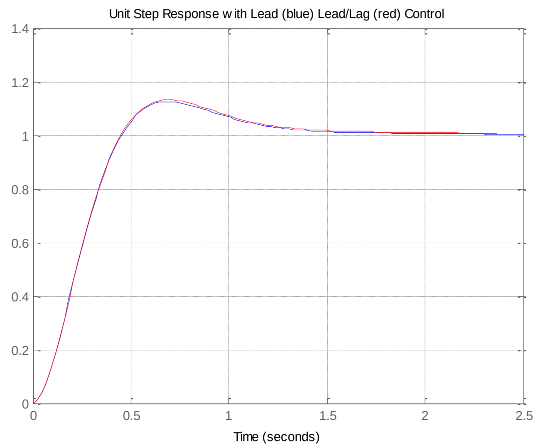
$$e_{ss} = \frac{0.1}{\lim_{s\to 0} sG(s)} = \frac{0.1}{\lim_{s\to 0} sG_c(s)G_p(s)} = \frac{0.1}{\lim_{s\to 0} s\left[\dfrac{.39(s+1.5)}{s+8.77}\right]\left[\dfrac{100}{s(s+1)}\right]} = 0.015^o \cdot$$

To reduce this error to 0.01^o, without altering the closed loop pole location at $-4+i4$, we will incorporate a second pole/zero controller, whose pole and zero as sufficiently close to zero that they contribute essentially a net angle of 0^o to the root locus angle criterion.

Specifically, let $G_{c_2}(s) = \dfrac{s-z}{s-p}$. Then $G_{c_2}(0) = \dfrac{z}{p}$, and the steady state error becomes

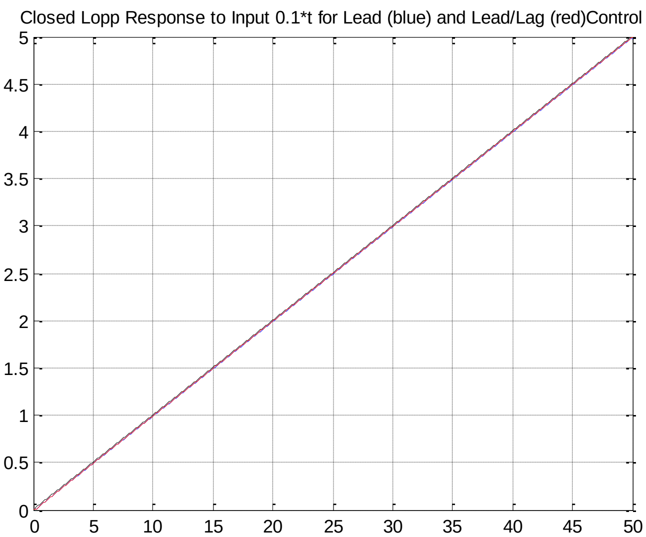
$$e_{ss} = \frac{0.1}{\lim_{s\to 0} sG(s)} = \frac{0.1}{\lim_{s\to 0} sG_c(s)G_{c_2}(s)G_p(s)} = \frac{0.1}{\lim_{s\to 0} s\left[\frac{.39(s+1.5)}{s+8.77}\right]\left[\frac{z}{p}\right]\left[\frac{100}{s(s+1)}\right]} = 0.015(p/z) \cdot$$

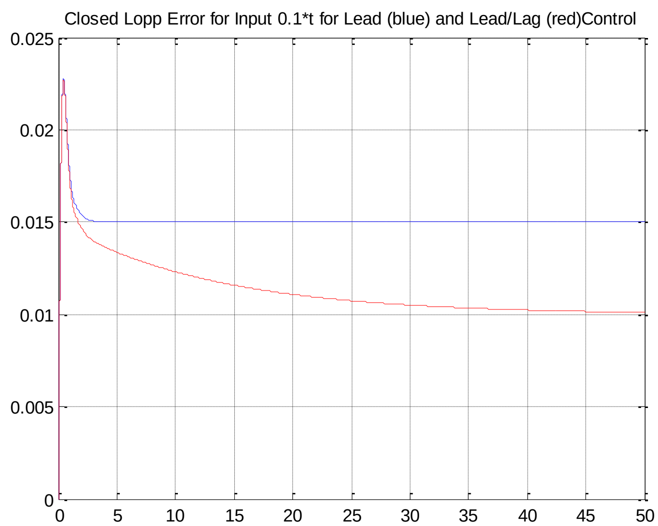
And so, (S3) results in $e_{ss} = 0.01 = 0.015(p/z)$, or $z = 1.5p$. If we choose p=-.05, then z=-.075, and so $G_{c_2}(s) = \frac{s+.075}{s+.05}$. For this controller the pole is closer to the real axis than the zero. Hence, it is called a ***lag controller*** (or compensator). The resulting closed loop unit step response is shown at the right, in comparison to that associated with only the lead controller. There is essentially no difference. This is because the lag controller pole and zero are so close to each other that they nearly cancel out.



Closed Loop response to the Ramp-

The closed loop response to $\theta_r(t) = 0.1t$ is shown below at the left, for both the lead and lead/lag controllers. From that figure it is difficult to see any difference between the two controllers. The figure at the lower right shows the error for each controller.





We see that the addition of the lag controller did, indeed, result in satisfying (S3). However, the steady state error is not achieved until ~50 seconds! This leads to the following question:

What was the intent behind the specification (S1)? If it was in relation to the time to achieve a fixed angle, then the above step response indicates that it was satisfied. If it was in relation to the time to achieve a steady state error for a ramp input, then it was not *at all* achieved. □

Conclusion: A lead controller is used to address CL dynamics, while a lag controller is generally used to address steady state performance. Once again, we see that controller design is as much an art as it is a science. We have a lot of opportunity to experiment with various controller pole/zero arrangements. It should also be reiterated that design specifications can include potential ambiguities. For example, is the specified response time chosen in relation to position or to ramp tracking error.

***Matlab* Code**

```
% PROGRAM NAME: LEC9_leadlag.m
Gp = tf(100,[1 1 0]);
% PART (a):
figure(1)
rlocus(Gp)
grid
title('Root Locus for Gc = K')
% Incorporate Lead Controller
Gc = tf([1 1.5],[1 8.77]);
G1 = Gp*Gc;
figure(2)
rlocus(G1)
grid
title('Root Locus for Lead Controller #1')
G1 = 0.39*G1; % Incorporate Controller K-value
H = tf(1,1);
W1 = feedback(G1,H);
figure(3)
step(W1)
grid
title('Unit Step Response Using Lead Controller #1')
% Incorporate Lag Controller
Gc2 = tf([1 .075],[1 .05]);
G2 = G1*Gc2;
W2 = feedback(G2,H);
hold on
[th,t]=step(W2);
plot(t,th,'r')
title('Unit Step Response with Lead (blue) Lead/Lag (red) Control')
figure(4)
% Compute Response to the Ramp Input
t = 0:.001:50;
thr = 0.1*t;
th1 = lsim(W1,thr,t)';
th2 = lsim(W2,thr,t)';
plot(t,thr,'k',t,th1,'b',t,th2,'r')
grid
title('Closed Lopp Response to Input 0.1*t for Lead (blue) and Lead/Lag (red)Control')
figure(5)
err1 = thr - th1;
err2 = thr - th2;
plot(t,err1,'b',t,err2,'r')
grid
title('Closed Loop Error for Input 0.1*t for Lead (blue) and Lead/Lag (red)Control')
```