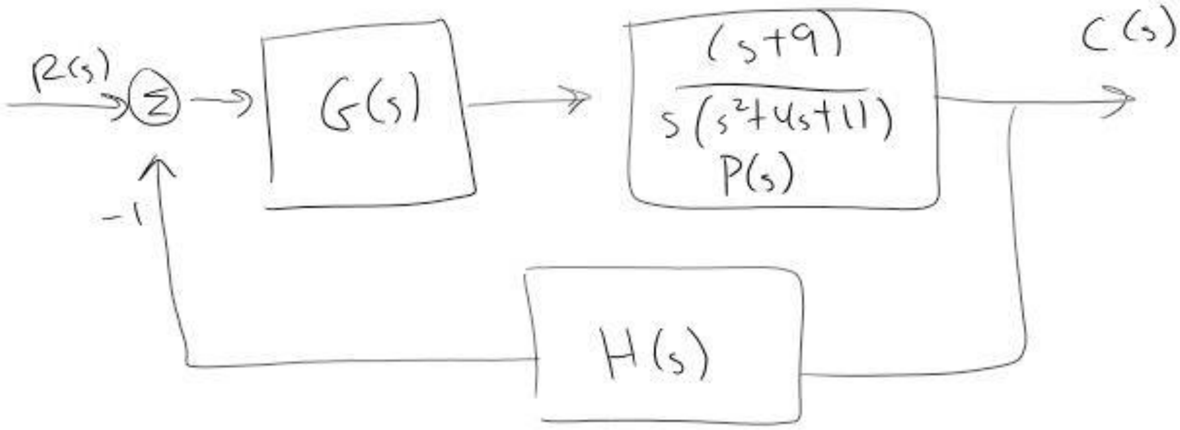
HW 7

John Sigmon
Js85773

1a)



First, find closed loop transfer function.

$TF_{CL} = \frac{G(s)P(s)}{1 + H(s)G(s)P(s)}$ We let $G(s) = K$, and $H(s) = 1$

$$TF_{CL} = \frac{K\,P(s)}{1 + K\,P(s)} = \frac{K\frac{(s+9)}{s(s^2+4s+11)}}{1 + K\frac{s+9}{s(s^2+4s+11)}} = \frac{k(s+9)}{s(s^2+4s+11) + k(s+9)}$$

$$T_{CL}(s) = \frac{k(s+9)}{s^3 + 4s^2 + (11+k)s + 9k}$$

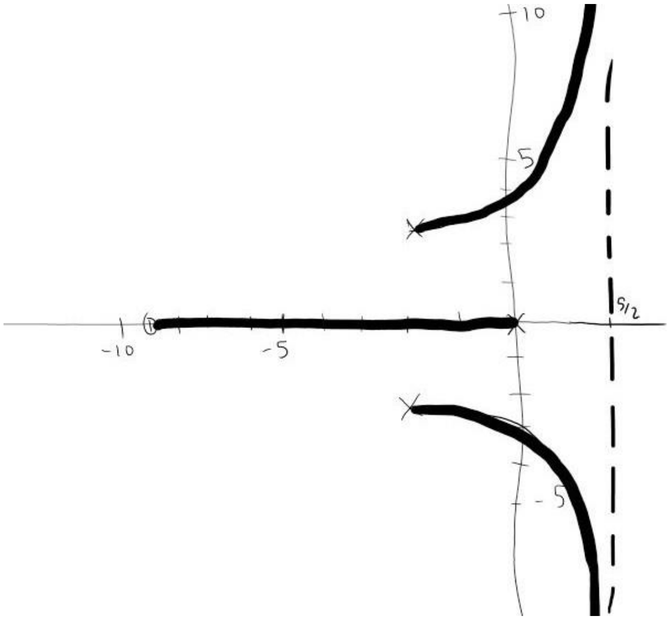
To sketch the RL, we need the open loop poles and zeroes

Zeroes: $\{-9\}$

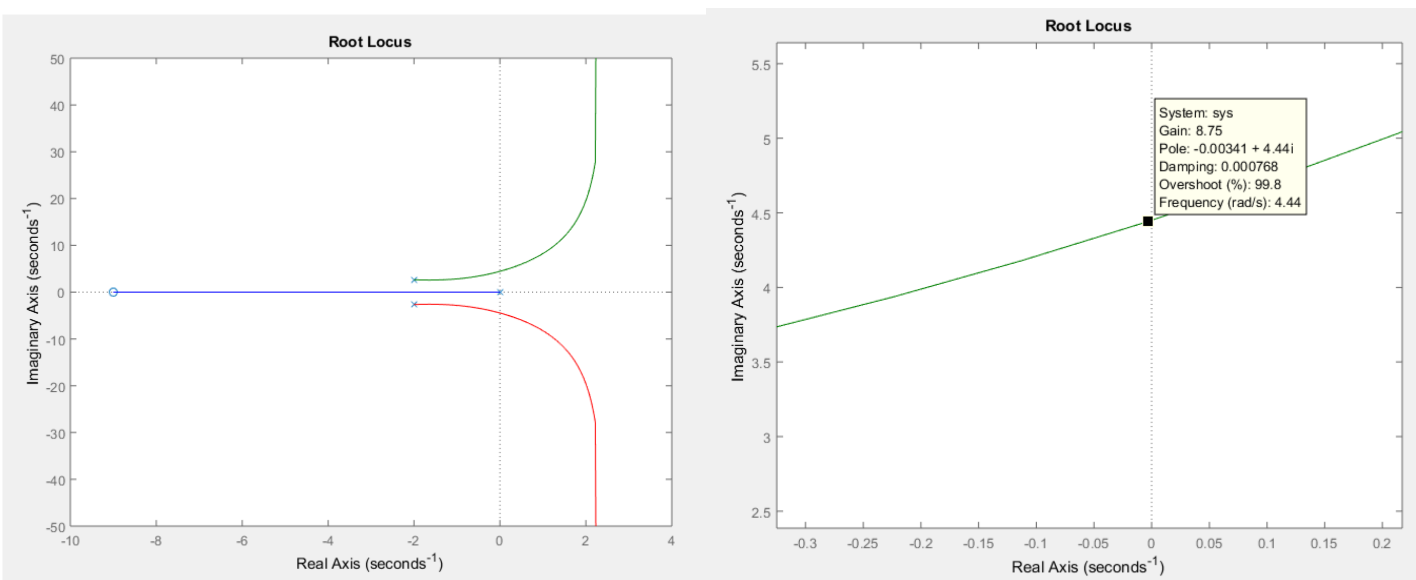
Poles: $\{0, -2+j\sqrt{7}, -2-j\sqrt{7}\}$ $\longrightarrow$ From $s(s^2+4s+11)$ and some algebra

Separate Loci	Find asymptotes:	Find asymptote angle:
$N_p - N_z = \lvert SL \rvert = 2$	$\sigma_A = \frac{\Sigma p_i - \Sigma z_j}{n_p - n_z} = \frac{(-2-2) - (-9)}{2} = \frac{5}{2}$	$\phi_A = \frac{2q+1}{2} = \frac{1}{2}(180°) = 90°, 270°$

Root locus sketch

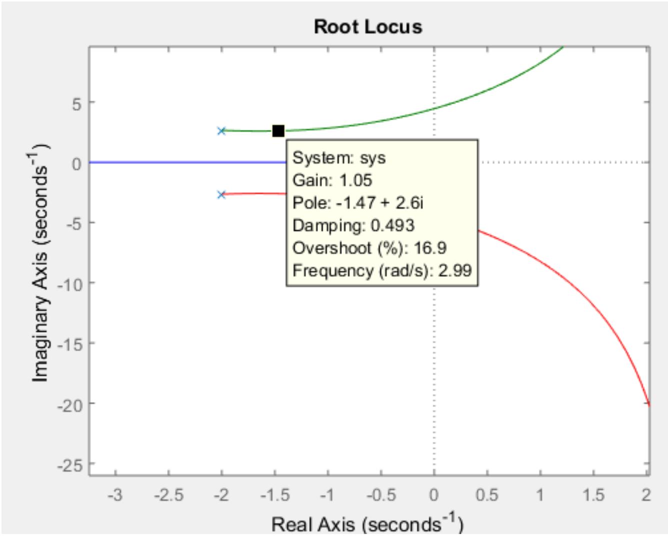


1b) Root locus plot by MATLAB



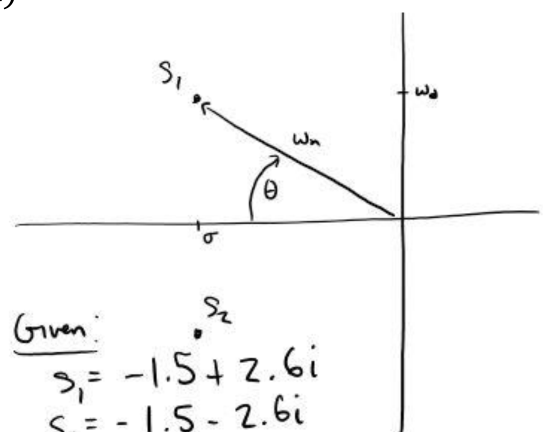
A neutrally stable system requires a pole to be on the imaginary axis (damping coefficient equal to zero). MATLAB's data cursor allowed me to estimate a gain that would place a pole on the imaginary axis (by the right hand photo above). When testing the minors of the Hurwitz matrix, at least one inequality in K will result. Since the resulting inequality will tell you when K has negative real parts, one of the bounds of the inequality will also give the point at which the real part crosses the imaginary axis, which is your desired gain.

1c)



The data cursor in the root locus plot shows the point where the damping ratio is approximately equal to 0.5. The resulting K value is 1.05.

1d)



Given:

$s_1 = -1.5 + 2.6i$

$s_2 = -1.5 - 2.6i$

By allowing our C.E. to be written as $\Delta s = s^2 + 2\zeta\omega_n s + \omega_n^2 = 0$

We can say:

$$s_1, s_2 = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2} = \sigma + j\omega_d$$

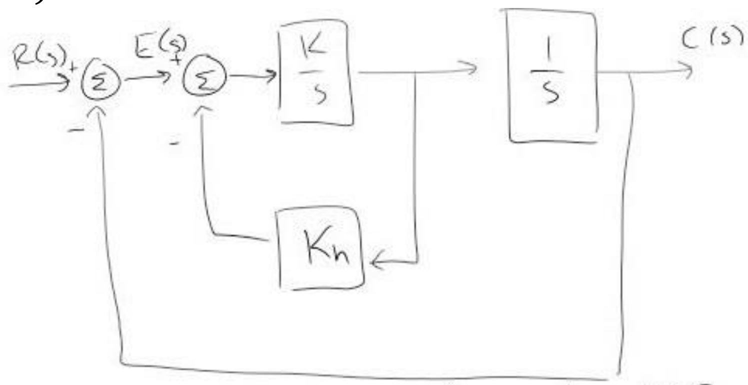
$$\omega_n = \|s_i\| = \sqrt{1.5^2 + 2.6^2} \approx 3$$

Then, $\sigma = -\zeta\omega_n \rightarrow \zeta = -\frac{\sigma}{\omega_n}$

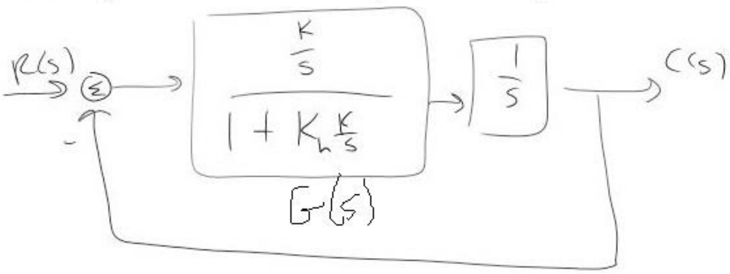
$$= -\frac{-1.5}{3} = 0.5$$

And so we confirmed that s_1, s_2 satisfy the condition $\zeta = \frac{1}{2}$

2a)



Choose K_h and K s.t. system has poles $-1 \pm j\sqrt{3}$



$$R(s) \longrightarrow \boxed{\frac{G(s)\frac{1}{s}}{1+G(s)\frac{1}{s}}} \longrightarrow C(s) \qquad = \qquad \frac{\frac{\frac{K}{s}}{1+K_h\frac{K}{s}}\cdot\frac{1}{s}}{1+\left(\frac{K/s}{1+K_h\frac{K}{s}}\right)\frac{1}{s}}$$

$$\frac{C(s)}{R(s)} = \frac{\frac{\frac{K}{s}}{1+K_h\frac{K}{s}}\cdot\cancel{\frac{1}{s}}\cdot s}{1+\left(\frac{\frac{K}{s}}{1+K_h\frac{K}{s}}\cdot\cancel{\frac{1}{s}}\right)\cdot s} = \frac{\frac{\frac{K}{s}}{1+K_h\frac{K}{s}}}{s+\left(\frac{\frac{K}{s}}{1+K_h\frac{K}{s}}\right)}\frac{\frac{s}{s}}{\frac{s}{s}} = \frac{\frac{K}{s+K_hK}}{s+\frac{K}{s+K_hK}}\cdot\frac{s+K_hK}{s+K_hK} = \frac{K}{s(s+K_hK)+K} = \boxed{\frac{K}{s^2+K_hKs+K} \quad \text{* Closed Loop Transfer Function}}$$

The question becomes, choose K_h, K s.t. $\Delta s = s^2 + K_hKs + K = 0$

$$\Delta s\big|_{-1+j\sqrt{3}} = (-1+j\sqrt{3})^2 + K_hK(-1+j\sqrt{3}) + K = 0$$

$$1 - j2\sqrt{3} + j^2 3 - K_hK + K_hKj\sqrt{3} + K = 0 \quad \text{separate Re, Im}$$

$$2 + j2\sqrt{3} = K - KK_h + j(K_hK)\sqrt{3}$$

We have: $K_hK = 2$ and $K - KK_h = 2 \rightarrow K(1-K_h) = 2$

$$K_h\frac{2}{1-K_h} = 2$$

$$2K_h = 2 - 2K_h$$

$$+2K_h \quad +2K_h$$

$$2 = 4K_h \rightarrow K_h = \tfrac{1}{2}$$

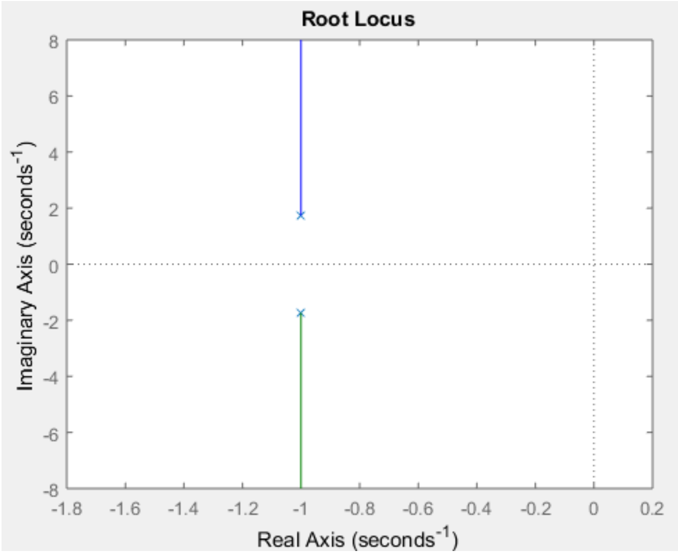
$$K - K\tfrac{1}{2} = 2$$

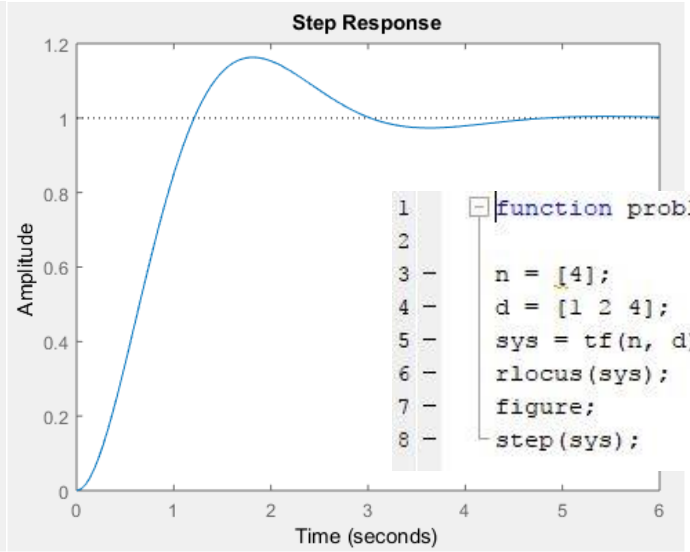
$$\tfrac{1}{2}K = 2$$

$$K = 4$$

$$\boxed{\begin{array}{l} K = 4 \\ K_h = \tfrac{1}{2} \end{array}}$$

These K values are verified by the RL plot of $H(s) = \frac{4}{s^2+2s+4}$





2b)

Re-stating, the closed loop transfer function is;

$$G_{cl}(s) = \frac{K}{s^2 + K_h K s + K}$$

Comparing with canonical form, we get the following:

$$\boxed{\begin{aligned} K &= \omega_n^2 \\ 2\zeta\omega_n &= K_h K \end{aligned}}$$

So, $\omega_n = \sqrt{K}$

Inserting into the second equation, $K_h = \frac{2\zeta}{\omega_n}$

We can consider our poles (s_1, s_2) to be given by the following:

$$\boxed{s_1, s_2 = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2}}$$

Now, our goal of no overshoot is accomplished when $Im\{s_1, s_2\} = 0$

$$\boxed{\omega_n\sqrt{1-\zeta^2} = 0}$$

Or, $\boxed{\zeta \geq 1}$, resulting in $\omega_n\sqrt{1-\zeta^2}$ being multiplied by an additional factor of j.

Rearranging our earlier equations $\longrightarrow$ $\omega_n = \sqrt{K}$ and $K_h = \frac{2\zeta}{\omega_n}$

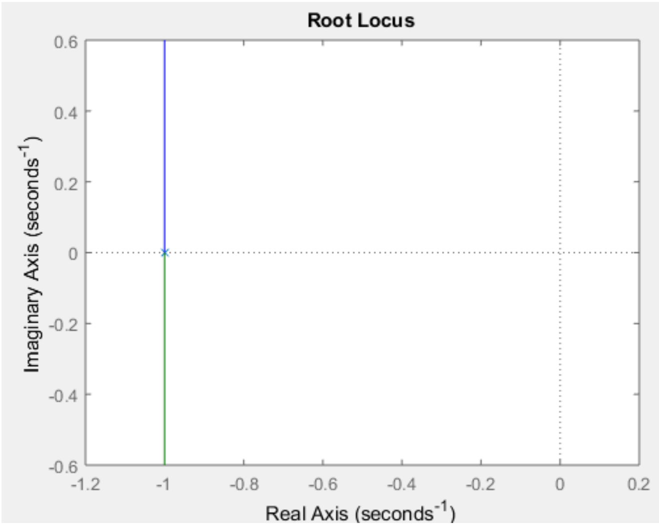
$$\zeta = \frac{K_h\sqrt{K}}{2} \geq 1 \rightarrow K_h\sqrt{K} \geq 2$$

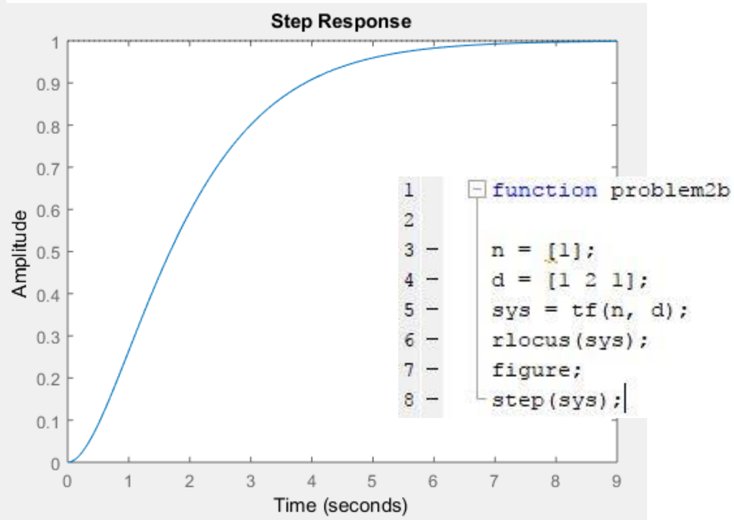
For simplicity, we choose $K_h = 2$ and $K = 1$.

Then, the closed loop poles are:

$$\boxed{s_1, s_2 = \frac{-2 \pm \sqrt{2^2 - 4}}{2} = -1}$$

The MATLAB Plot confirms that these poles result in no overshoot.





3a)

Determine stability only if necessary, but not sufficient conditions are met.

Part a.) $G(s)=\dfrac{s-5}{s^4+10s^3+100s^2+15}$ (Hurwitz Criterion)

The following conditions must hold:
1.) All coefficients must have same sign (TRUE)
2.) None of the coefficients vanish (FALSE)

~~Therefore, the necessary conditions do not hold~~ Edit - per rubric see below for determinant calculations.

$a=\{15, 0, 100, 10, 1\}$

The matrix of Hurwitz determinants is:

$$\begin{bmatrix} 10 & 1 & 0 & 0 \\ 0 & 100 & 10 & 1 \\ 0 & 15 & 0 & 100 \\ 0 & 0 & 0 & 15 \end{bmatrix}$$

$D_1 = |10| \rightarrow D_1 > 0$

$D_2 = \left|\begin{bmatrix} 10 & 1 \\ 0 & 100 \end{bmatrix}\right| = 1{,}000 \rightarrow D_2 > 0$

$D_3 = \left|\begin{bmatrix} 10 & 1 & 0 \\ 0 & 100 & 10 \\ 0 & 15 & 0 \end{bmatrix}\right| = 10\begin{vmatrix} 100 & 10 \\ 15 & 0 \end{vmatrix} - 0 + 0 = -1{,}500 \rightarrow D_3 \not> 0$

It is not necessary to check D_4.

Therefore the system is not stable, by the Hurwitz matrix and the Hurwitz criteria.

3b)

Part b.) $G(s) = \frac{s+5}{10s^4+10s^3+20s^2+Ks+1}$ For the Routh Tabulation, we make a table.

s^4	10	20	1
s^3	10	K	0
s^2	20-K	1	0
s^1	$K-\frac{10}{20-K}$	0	0
s^0	1	0	0

$b_3 = \frac{10\cdot 20 - K10}{10} = 20-K$

$b_2 = \frac{10\cdot 1 - 0\cdot 10}{10} = 1$

$c_3 = \frac{(20-K)(K) - 10\cdot 1}{20-K} = K - \frac{10}{20-K}$

$c_2 = 0$

We require the first column to have the same sign.

Therefore, $20-K > 0$, $K - \frac{10}{20-K} > 0$

Condition ① gives $\boxed{K<20}$ However, our second condition turns out to be stricter.

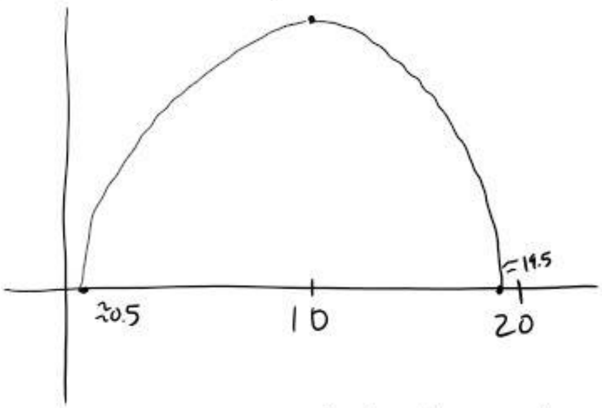
From the second condition, $K(20-K) - 10 > 0 \rightarrow \underbrace{-K^2+20K-10}_{p(K)} > 0$

$\frac{dp(K)}{dK} = -2K+20 \rightarrow$ zero when $K=10$

Roots of $p(K)$:

$K = \frac{-20 \pm \sqrt{400-4(-1)(-10)}}{-2} = 10 \pm \frac{\sqrt{400-40}}{-2}$

$= 10 \pm \frac{\sqrt{10}\sqrt{36}}{-2} = 10+3\sqrt{10}, 10-3\sqrt{10}$



We can conclude that the roots lie in the left hand side of the s-plane when $\boxed{10-3\sqrt{10} \le K \le 10+3\sqrt{10}}$

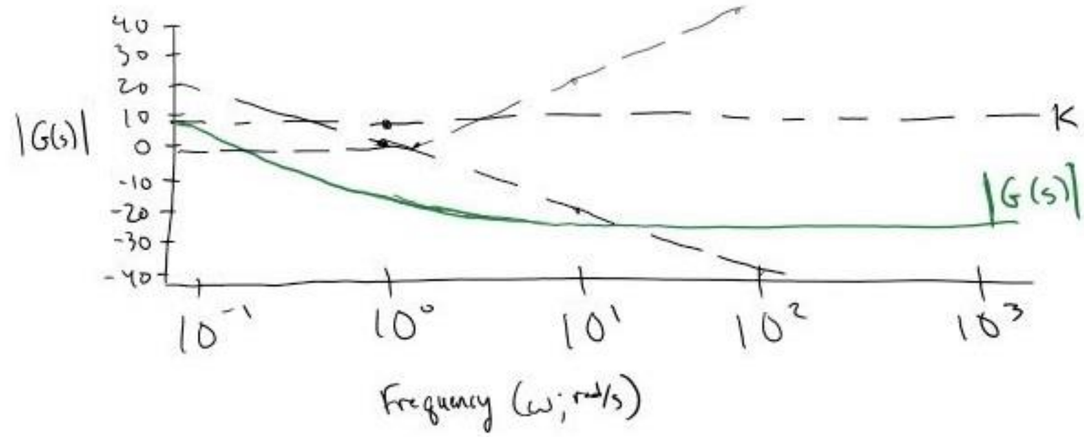
4a)

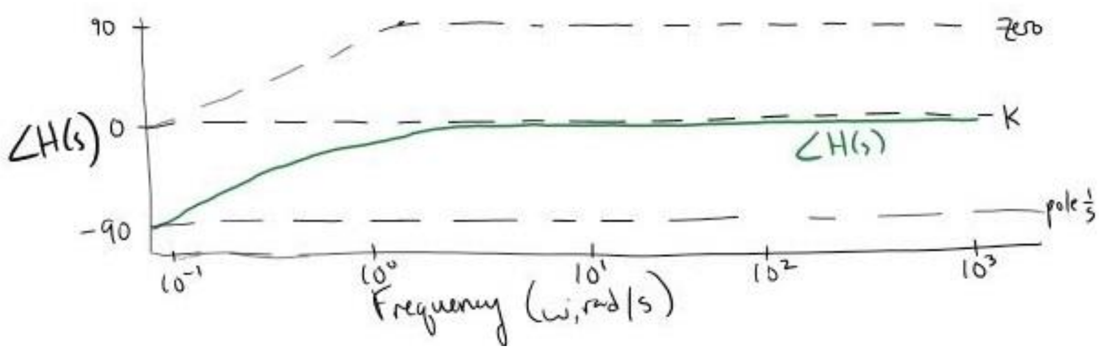
$$G_c(s) = 5\left(1+\frac{1}{2s}\right) \rightarrow 5\left(\frac{2s+1}{2s}\right) \rightarrow \frac{5}{2}\left(\frac{2s+1}{s}\right)$$

We have: $K = 20\log\frac{5}{2}$ (Gain ≈ 7.96)

$\frac{s}{1/2}+1$ (zero at $-\frac{1}{2}$)

$\frac{1}{s}$ (pole at origin)





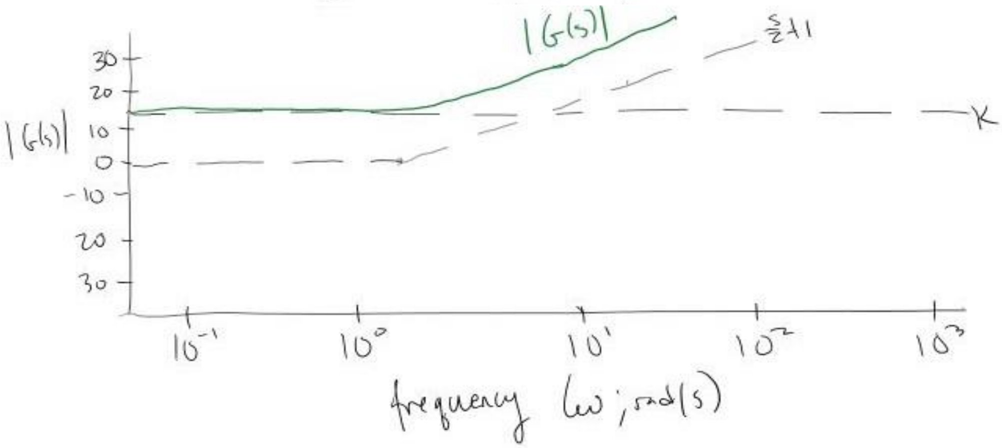
4b)

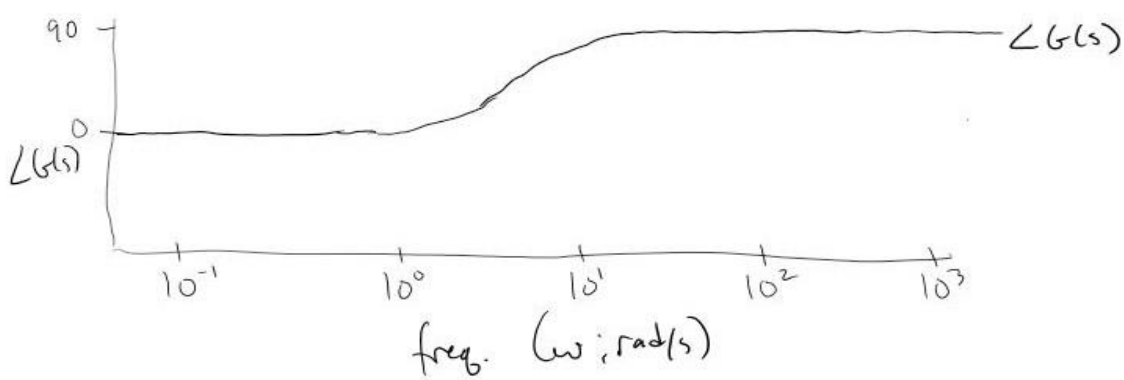
$$G_c(s) = 5\left(1+\tfrac{1}{2}s\right) \rightarrow 5\left(\tfrac{s}{2}+1\right)$$

Two terms:

Gain: $20\log(5) \approx 13.98$

Zero: $\frac{s}{2}+1$





4c)

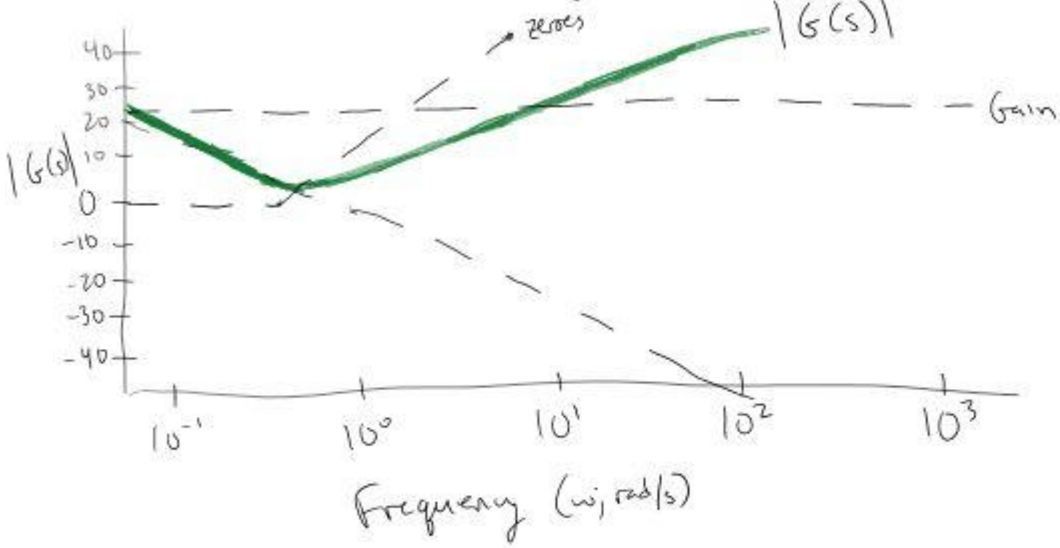
$$G_c(s) = 30.325\left(\frac{(s+.65)^2}{s}\right) \rightarrow 30.325\,\frac{(s+0.65)(s+0.65)}{s} = (30.325)(0.65)(0.65)\,\frac{\left(\frac{s}{0.65}+1\right)\left(\frac{s}{0.65}+1\right)}{s}$$

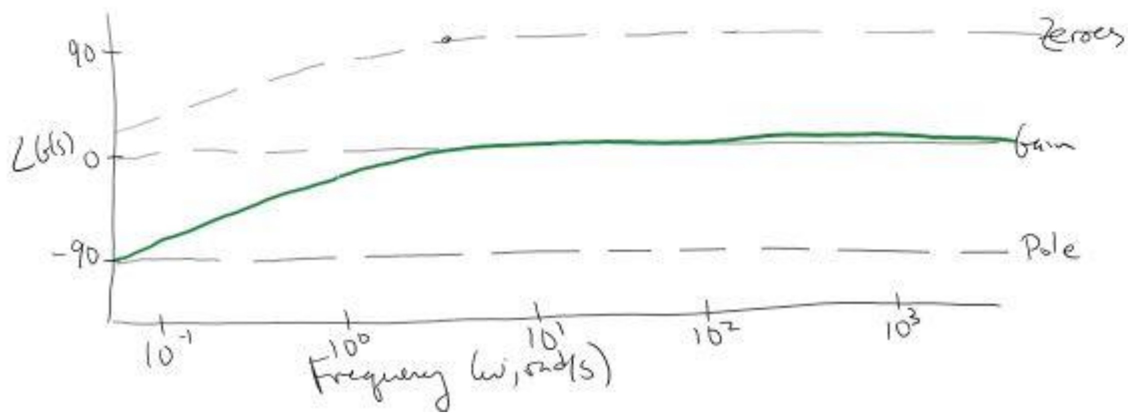
$$= 12.81\,\frac{\left(\frac{s}{0.65}+1\right)\left(\frac{s}{0.65}+1\right)}{s}$$

We have 4 terms: Two zeroes $\left(\frac{s}{0.65}+1\right)$

Gain $K = 20\log(12.81) \approx 22.15$

Pole at origin $\frac{1}{s}$





5a)

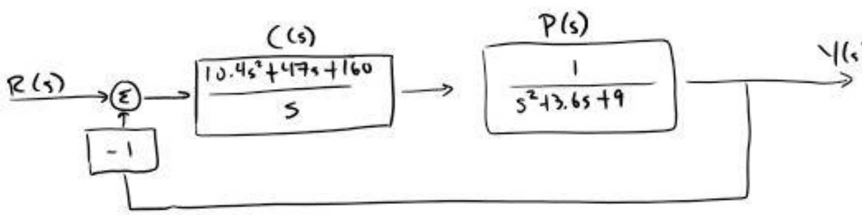
We see that $G_c(s) = \dfrac{s^2 + 4.5192s + 15.385}{s}$, which is then multiplied by another gain K.

The complete controller is given by:

$$C(s) = \frac{(10.4)s^2 + (4.5192)(10.4)s + (15.385)(10.4)}{s} \longrightarrow \begin{array}{l} K_i = 15.385 \cdot 10.4 = 160 \\ K_d = 10.4 \\ K_p = 4.5192 \cdot 10.4 = 47 \end{array}$$

Assuming $N(s) = D(s) = 0$

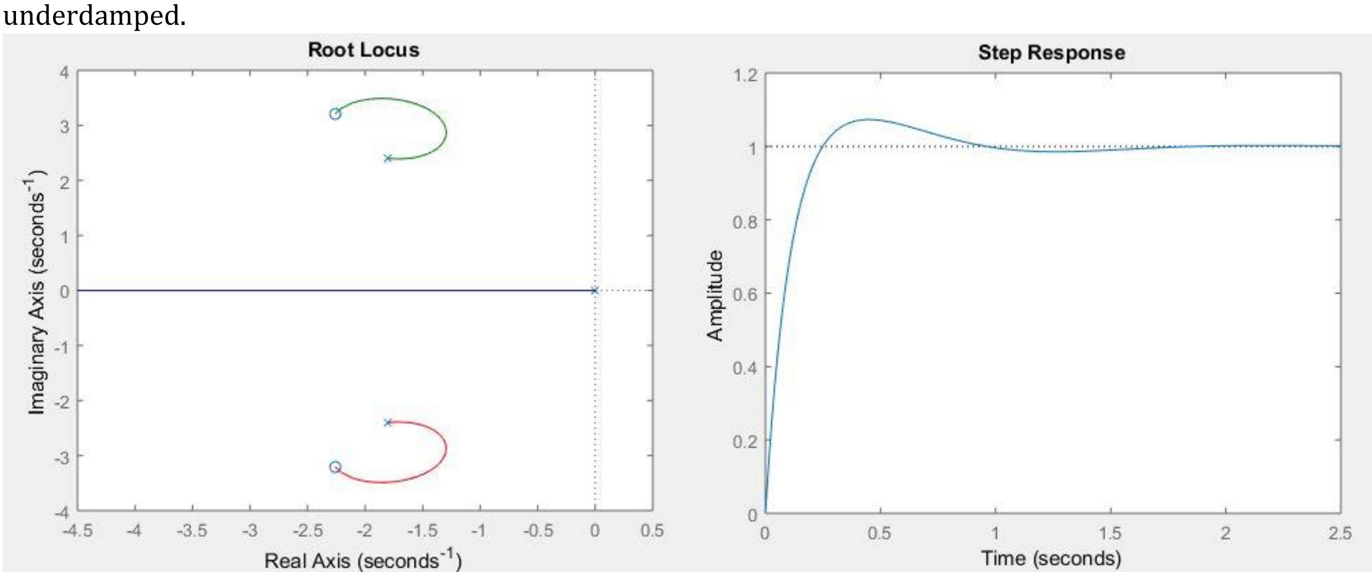
$$C(s) = \frac{10.4s^2 + 47s + 160}{s}$$



$$H(s) = \frac{C(s)P(s)}{1 + C(s)P(s)}$$

5b)

An inspection of the root locus plot for the system helps to show the controllers limitations. By inspection, there is no gain value for which the system can be made overdamped. The complex conjugate poles and zeros are specifically what limit the system performance in this case. The symmetric loci travel a very short distance in the s-plane, meaning the gain values allow a very short range of damping ratios to be selected, and they are all underdamped.



Additionally, the open loop zeros are at $-2.26 \pm 3.21i$.
The open loop poles are at the origin, and at $-1.8 \pm 2.4i$.
This information was verified via MATLAB's data cursor and the Root Locus plot above.

5c)

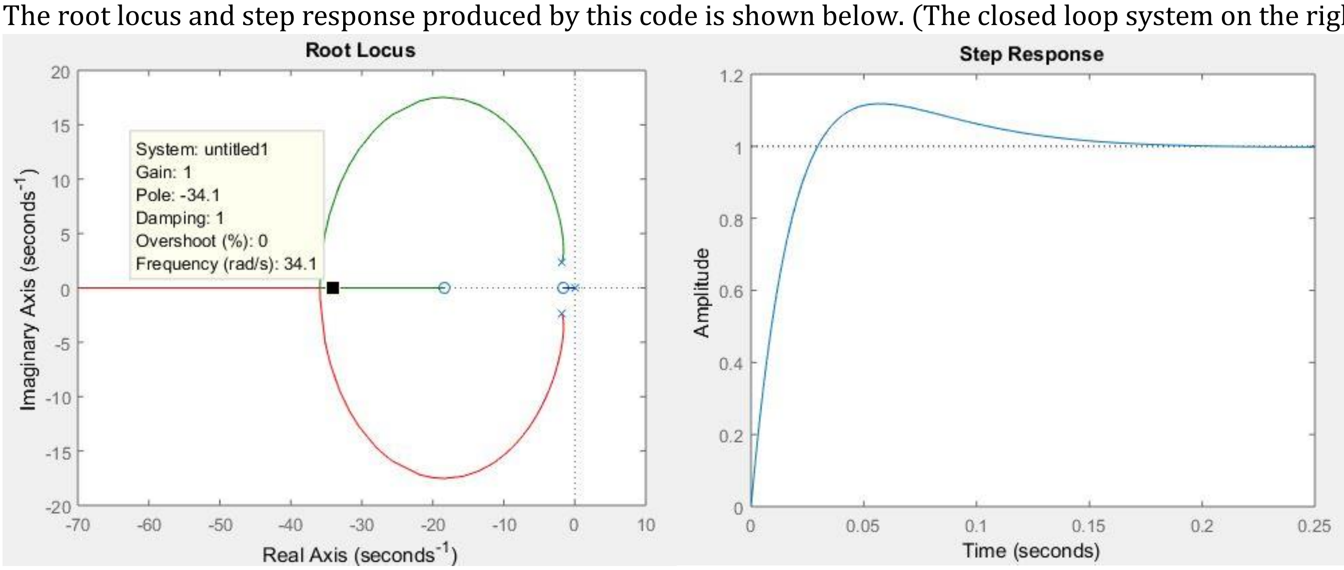
As we discussed in office hours, I had some weird MATLAB results. The correct procedure is to use the PID controller to place the zeros on the real line. Then MATLAB can be used to identify values of K for which all loci are on the real line, thus insuring an overdamped system (i.e. no overshoot). The numerator of the controller's transfer function is quadratic, so some simple algebra can be used to determine when its discriminant is greater than zero, which results in real roots. Once the zeros are placed on the real line, a gain value just after the break-away points on the real line will be sufficient to determine where the system has all three poles on the real line. This should result in an overdamped system. My trouble came at this point. After verifying all three poles are indeed negative and real, the step response still has overshoot.

Here is the full code, along with the controller gains used in line 5. The roots of the closed loop system were verified to be real and negative.

```
1      % function [CLsys] =openloop
2  -   clear all;
3  -   close all;
4
5  -   n1 = 6.72*10.4*[1 20 30];
6  -   d1 = [1 0];
7  -   C = tf(n1, d1);
8  -   n2 = [1];
9  -   d2 = [1 3.6 9];
10 -   P = tf(n2, d2);
11 -   OLsys = C*P;
12
13 -   CLsys = feedback(C*P, 1);
14 -   step(P/(1+P), 'r--', CLsys, 'b:');
15 -   legend("Process", "Process with Controller");
16 -   figure;
17 -   rlocus(C*P);
18
19 -   figure;
20 -   step(CLsys);
21
```

```
>> openloop
>> CLsys

CLsys =

     69.89 s^2 + 1398 s + 2097
  -------------------------------
  s^3 + 73.49 s^2 + 1407 s + 2097

Continuous-time transfer function.

>> roots([1 73.49 1407 2097])

ans =

  -36.9001
  -34.9645
   -1.6253
```

The root locus and step response produced by this code is shown below. (The closed loop system on the right)



Clearly something strange is happening. The gain shown on the root locus plot verifies the poles shown in the command window. I hope this is sufficient to demonstrate that I understand the argument needed, how to use a PID controller to move the zeros of the closed loop system, how that affects the geometry and loci of the root locus plot, and how these methods can be used to adjust the parameters of the system.