ENAE432
PS9 Soln.

Question 1.

$u(t) = K_p e(t) + K_d \dot{e}(t) + K_I \int_0^t e(\tau) d\tau$

$\Rightarrow u(t) = K_p e(t) + K_d \dot{e}(t) + K_I x_1(t)$

$\dot{x}_1(t) = e(t)$

$$G(s) = \frac{4}{s(s+2)^2}$$

1.) Start with $H(s) = 1$ to determine K_u and T_u:

$L(s) = \frac{4}{s(s+2)^2}$

K_u = gain margin of $L(s)$

T_u = use phase crossover frequency of $L(s)$ to determine.

from Bode of $L(s)$:

$a = 28.6$ dB @ $\omega_r = 3 \frac{rad}{s}$ $\Rightarrow K_u = 27$

$T_u = \frac{2\pi}{\omega_r} = \frac{2\pi}{3} \approx 2.09s$

2.) Now use the given tuning equations to determine K_p, K_d, K_I:

$K_p = \frac{3K_u}{5}$ $\qquad K_I = \frac{2K_p}{T_u}$ $\qquad K_d = \frac{K_p T_u}{8}$

MATLAB gives:

$K_p = 16.20$
$K_I = 15.47$
$K_d = 4.24$

3.) Find $H(s)$ & $L(s)$ which results from Ziegler-Nichols tuning:

from $u(t)$, we get $H(s) = K_p + K_d s + \frac{K_I}{s}$.

$$\boxed{H(s) = 16.20 + 4.24s + \frac{15.47}{s} = \frac{4.24s^2 + 16.2s + 15.47}{s}}$$

$$\boxed{L(s) = \frac{4(4.24s^2 + 16.2s + 15.47)}{s^2 (s+2)^2}}$$

4) Quantify resulting crossovers & margins:

$\gamma = 25.6°$
no gain margin

see matlab plot.

*no gain margin because the phase plot never crosses $-180°$.

5) Compute resulting closed-loop poles:

matlab gives: $\boxed{\text{CLpoles} = \begin{cases} -0.6845 \pm 2.1181j \\ -2.3155 \pm 0.7244j \end{cases}}$

6.) Quantify features of unit step response of T(s)

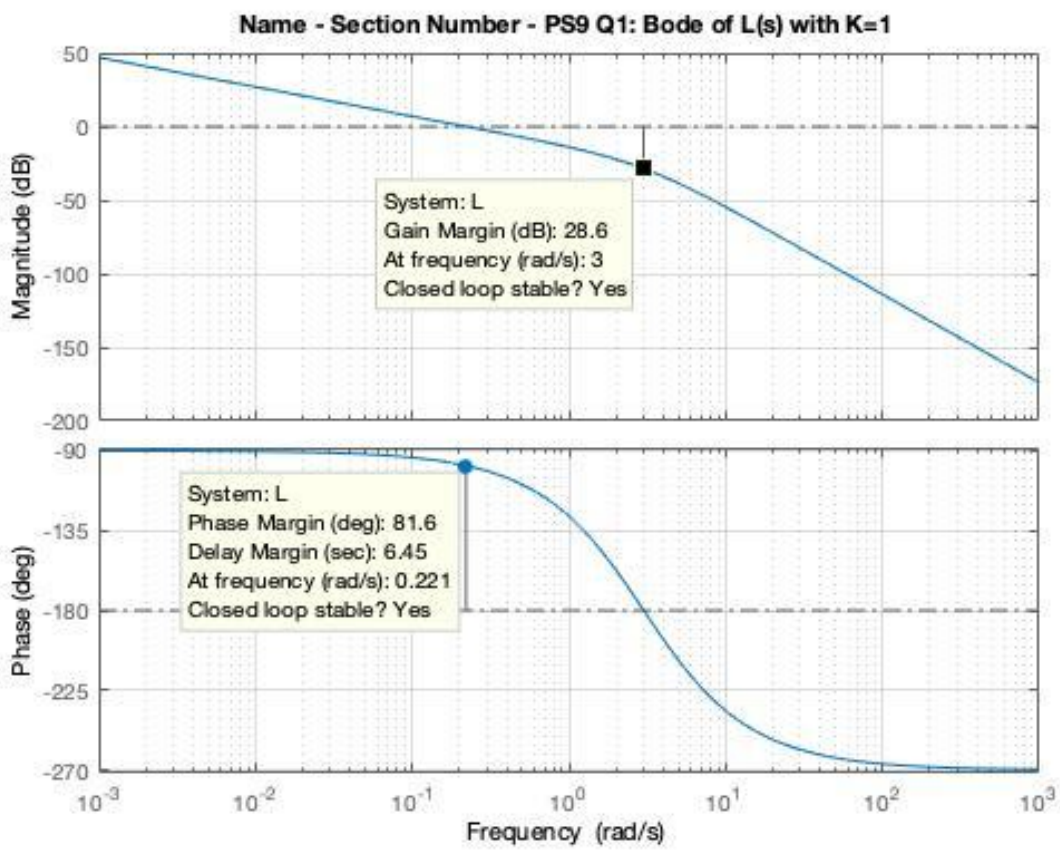
matlab gives:

% overshoot = 58.8%
$t_s = 6s$
$y_{ss} = 1$

7) Evaluate feedback loop design:

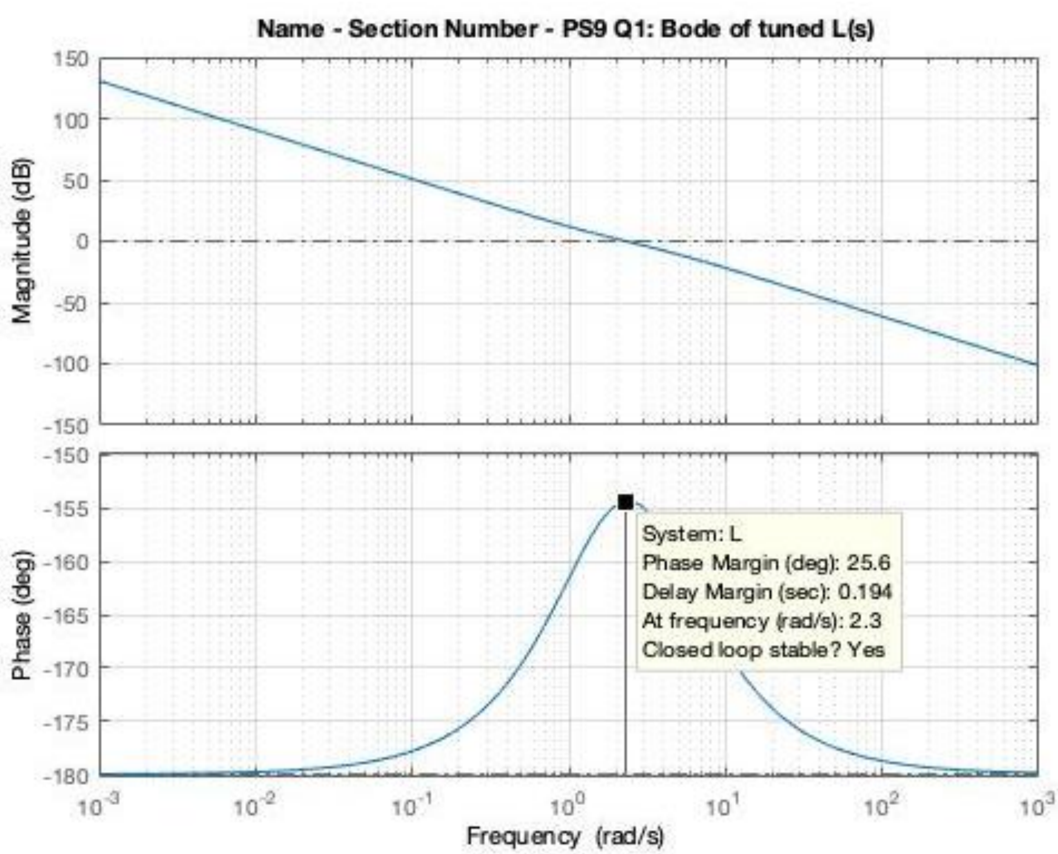
This is NOT a good design because 58.8% is a lot of overshoot. Plus, a settling time of 6s is too slow. Both of these are not ideal for most designs.

```
% ENAE 432, Spring 2019
% TA Solutions
% PS9, Question 1

s = tf('s');
w = logspace(-3,3,250000);
G = 2/(s*(s+3)^2);
H = 1;
L = minreal(G*H);
bode(L,w); grid on;
title('Name - Section Number - PS9 Q1: Bode of L(s) with K=1');
```



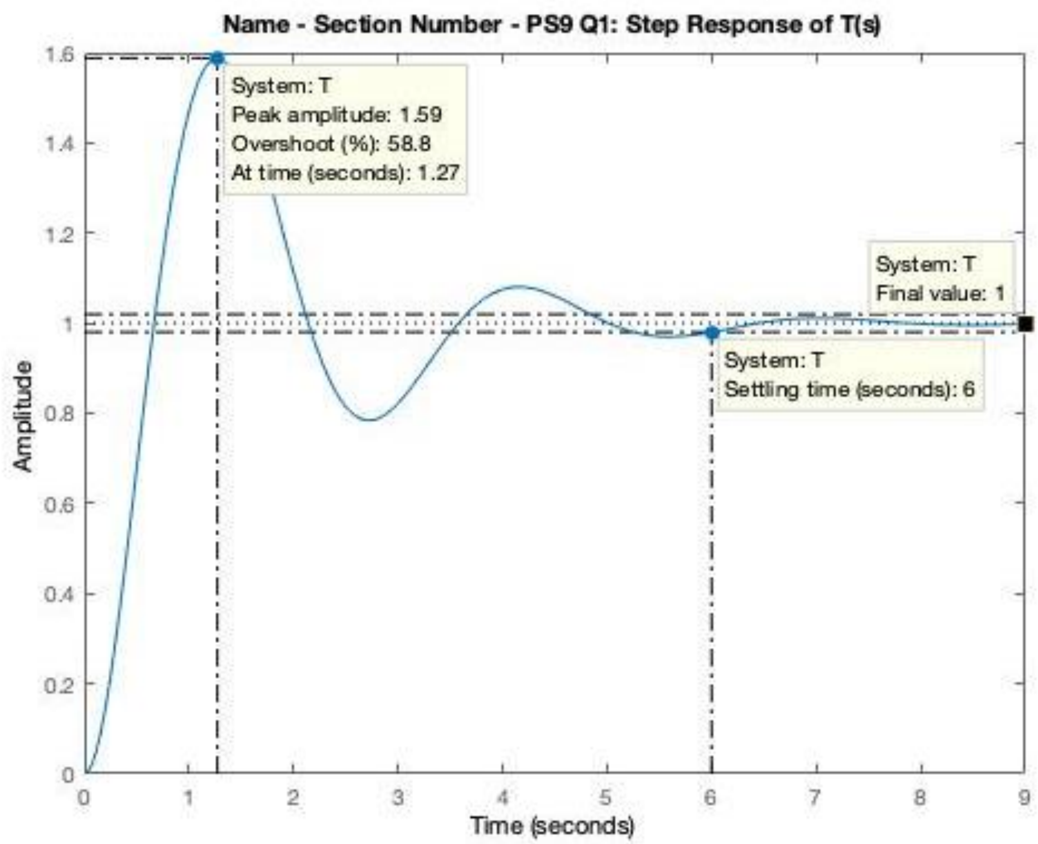

```
Ku = 27; Tu=2*pi/3;      % determined from Bode of L(s)
Kp = 3*Ku/5              % tuning equations
Ki = 2*Kp/Tu
Kd = Kp*Tu/8


Kp =

   16.2000
```

```
Ki =

   15.4699


Kd =

     4.2412


H = Kp+Kd*s+Ki/s         % new H(s) with Kp, Kd, Ki values
L = minreal(G*H);        % L(s) which results from Ziegler-Nichols
tuning
bode(L,w); grid on;
title('Name - Section Number - PS9 Q1: Bode of tuned L(s)');

H =

  4.241 s^2 + 16.2 s + 15.47
  --------------------------
              s

Continuous-time transfer function.
```



```
T = feedback(L,1)
CLpoles = pole(T)
step(T); title('Name - Section Number - PS9 Q1: Step Response of
T(s)');
```

```
T =

        8.482 s^2 + 32.4 s + 30.94
  ----------------------------------------
  s^4 + 6 s^3 + 17.48 s^2 + 32.4 s + 30.94

Continuous-time transfer function.


CLpoles =

  -0.6845 + 2.1881i
  -0.6845 - 2.1881i
  -2.3155 + 0.7244i
  -2.3155 - 0.7244i
```



PS9

Q2. PID controller: $u(t) = K_p\, e(t) + K_d\, \dot{e}(t) + K_I \int_0^t e(\tau)\,d\tau$

$$H(s) = K_p + K_d s + \frac{K_I}{s}$$

More generally, $H(s) = K\dfrac{(s-z_1)(s-z_2)}{s}$

Now the goal is to tune the compensator so the loop has a phase margin of 50° at a crossover of 2 rad/s.

- Determine the equivalent PID gains to achieve this target
- Generate step response of $T(s)$ compare with Q#1.

SOLUTION:

Since $\gamma = 50^\circ$, $\gamma = 180^\circ + \measuredangle L(j\omega_\gamma) \Rightarrow \measuredangle L(j\omega_\gamma) = \underline{-130^\circ}$, $\omega_\gamma = \underline{2 \text{ rad/s}}$

$$G(s) = \frac{4}{s(s+2)^2};$$

If we place the zeros in $H(s)$ at the same location, then $H(s) = \dfrac{K(s-z)^2}{s}$

Therefore $L(s) = \dfrac{4K(s-z)^2}{s^2(s+2)^2}$

$$\measuredangle L(j\omega_\gamma) = \measuredangle 4K + 2\measuredangle(\omega_\gamma j - z) - 2\measuredangle \omega_\gamma j - 2\measuredangle(\omega_\gamma j + 2)$$

$$-130^\circ = 0^\circ + 2\measuredangle(\omega_\gamma j - z) - 2(90^\circ) - 2(2j+2)$$

$$-130^\circ = 0^\circ + 2\measuredangle(2j - z) - 180^\circ - 90^\circ$$

$$140^\circ = 2\measuredangle(2j - z)$$

$$70^\circ = \pi - \tan^{-1}\left(\frac{2}{z}\right) \qquad \underline{z = -0.7279}$$

Next solve for K

$$|L(j\omega_\gamma)| = 1 = \frac{|4K|\,|2j + z|^2}{|2j|^2|2j+2|^2} \qquad \underline{K = 1.7660}$$

Therefore $H(s) = \dfrac{K(s-z)^2}{s} = \dfrac{1.7660(s+0.7279)^2}{s}$

$$H(s) = 1.766s + 2.571 + \frac{0.9358}{s}$$

Therefore $\boxed{\begin{aligned} K_p &= 2.571 \\ K_d &= 1.766 \\ K_I &= 0.9358 \end{aligned}}$

- The MATLAB graph attached shows the phase margin for $L(s)$ confirming that γ is 50° at 2 rad/s

- Step response of $T(s)$ shows that % overshoot = 25.7 %

  t_s = 5.55 seconds

  Step response of $T(s)$ from Q#1 showed that % OS = 58.8%

  t_s = 6 seconds

  Therefore we can see that the PID controller from Question #2 produces more desirable metrics in the closed loop response.

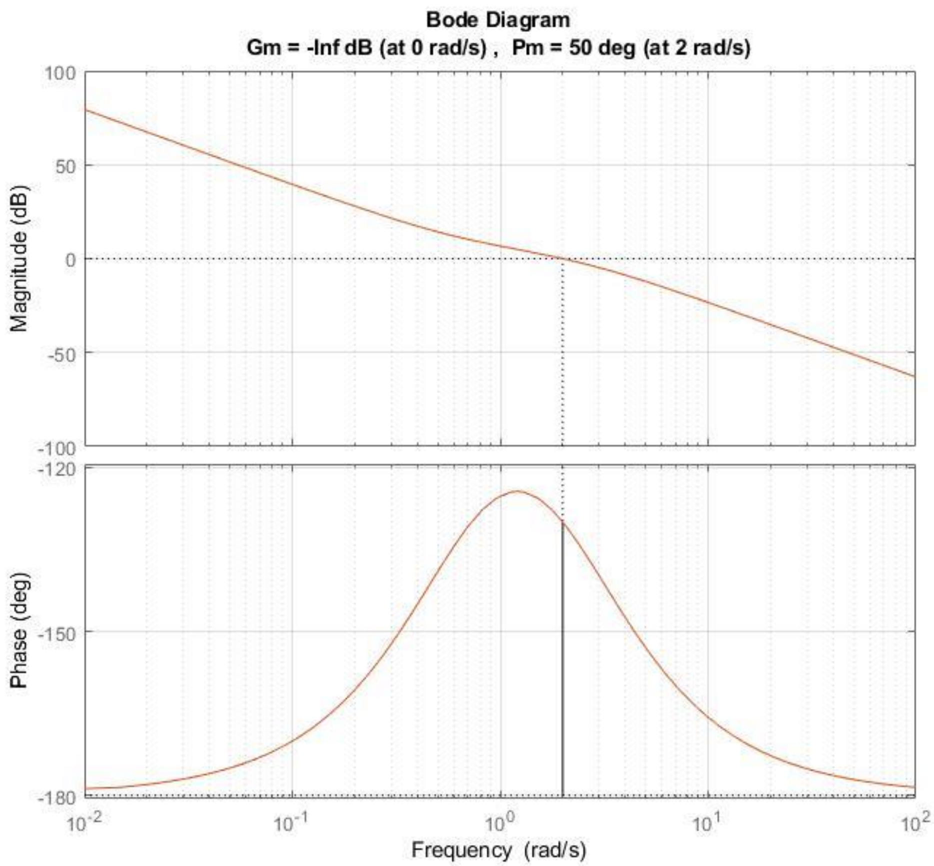
- An alternative approach would have been to split the phase contribution to 2 separate zeros in $H(s)$. i.e. $H(s) = \dfrac{K(s-z_1)(s-z_2)}{s}$

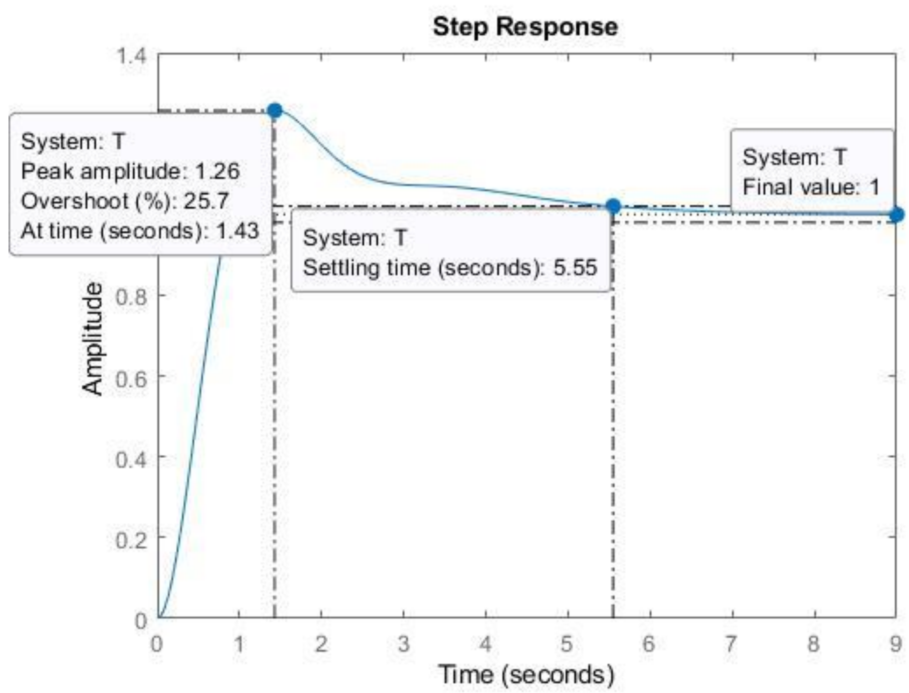
- So If we say: 140° is split 80°, 60° between both zeros

  $z_1 = \dfrac{\omega_\gamma}{\tan(80^\circ)}$, $z_2 = \dfrac{\omega_\gamma}{\tan(60^\circ)}$. The analysis for this case is shown in the MATLAB portion.

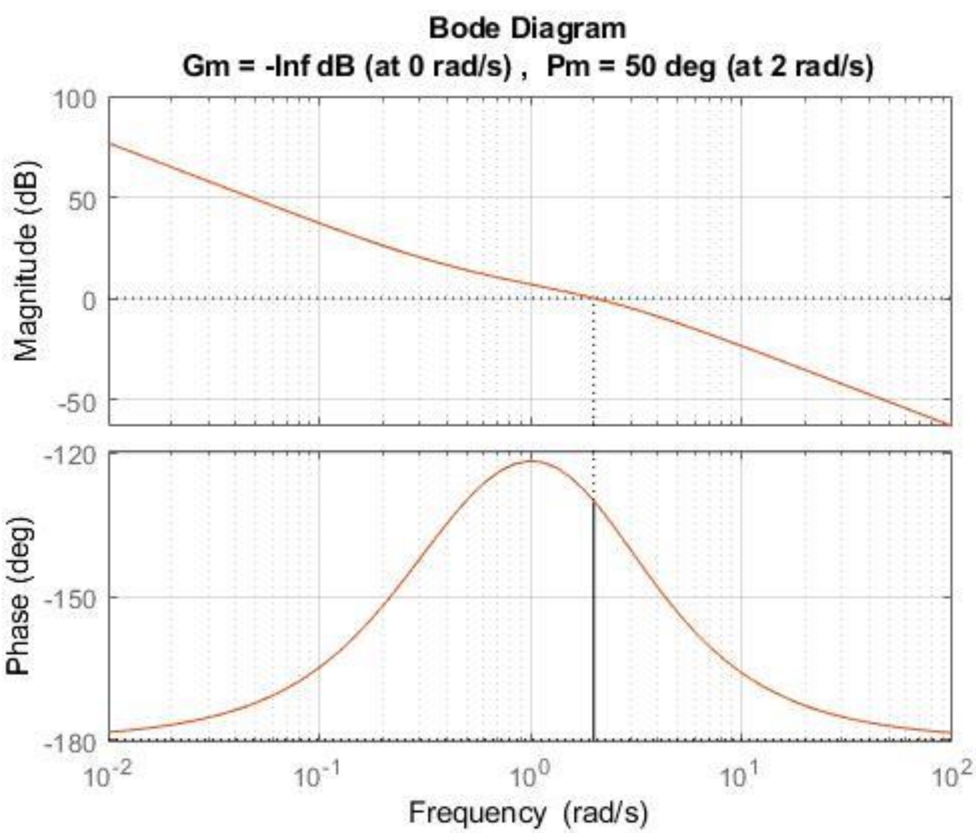
ENAE 432 PS9

```
% QUESTION 2
close all
clear all
clc
s = tf('s');
G = 4/(s*(s+2)^2);
PM = 50; wdes=2;
z = wdes/(tand(70));
Ho = (s+z)^2/s;
Lo = G*Ho;
magi = abs(evalfr(Lo,2j));
K = 1/magi
H = K*Ho
L = G*H;
figure(1)
bode(L)
grid on
hold on
margin(L)
hold off
T = minreal(L /(1+L));
step(T)
%  2nd scenario would be to split the phase contribution among 2 separate
%  zeros, say an 80:60 ratio
z1 = 2/(tand(80));
z2 = 2/(tand(60));
Ho = (s+z1)*(s+z2)/s;
Lo = G*Ho;
magnitude = abs(evalfr(Lo,2j));
K = 1/magnitude
H = K*Ho
L = G*H;
figure(2)
bode(L)
grid on
hold on
margin(L)
hold off
T1 = minreal(L /(1+L));
step(T1)
```


Bode Diagram
Gm = -Inf dB (at 0 rad/s) , Pm = 50 deg (at 2 rad/s)
Magnitude (dB)
Phase (deg)
Frequency (rad/s)
100
50
0
-50
-100
-120
-150
-180
10^-2
10^-1
10^0
10^1
10^2

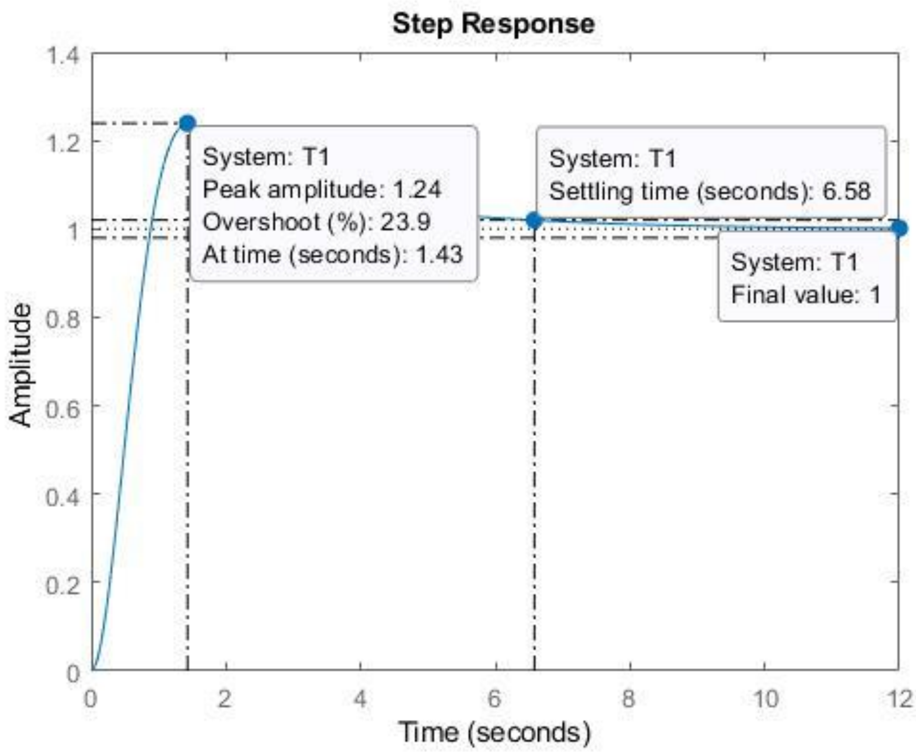

Step Response
System: T
Peak amplitude: 1.26
Overshoot (%): 25.7
At time (seconds): 1.43
System: T
Settling time (seconds): 5.55
System: T
Final value: 1
Amplitude
Time (seconds)
1.4
0.8
0.6
0.4
0.2
0
0
1
2
3
4
5
6
7
8
9

```
K =

    1.7660


H =

  1.766 s^2 + 2.571 s + 0.9358
  ----------------------------
               s

Continuous-time transfer function.
```




```
K =

    1.7057


H =

  1.706 s^2 + 2.571 s + 0.6946
```

```
  ----------------------------
               s

Continuous-time transfer function.
```



Published with MATLAB® R2018b

PS9 Q3 Solns

a) $\tau_s = \frac{\gamma_0}{\omega_\gamma} = \frac{50\left(\frac{\pi}{180}\right)}{2} = \underline{0.436\,sec}$; loop rate $\frac{1}{\tau_s} = \underline{2.29\ Hz}$

b) $\tau_s = \frac{1}{10} = 0.1\,sec$

$\gamma_{eff} = \gamma_0 - \omega_\gamma \tau_s = 50 - 2(0.1)\left(\frac{180}{\pi}\right) = 38.5^\circ$

- for a_{eff}, time delay can be represented as a transfer fxn as: $e^{-s\tau_s}$

so that $L_{eff} = e^{-s\tau_s} L(s) = e^{-s\tau_s}\left(\frac{7.06(s+0.7279)^2}{s^2(s+2)^2}\right)$

we use matlab's margin command to find the gain margin a_{eff} of L_{eff} to be $\underline{a_{eff} = 11.6\,dB}$ (see attached)

c) $y_d(t) = a_0 + a_1 t + a_2 t^2$. Per the 2 poles in the origin of $L(s)$, we can perfectly track up to type $p=1$ polynomials: the only term contributing to steady-state error is $a_2 t^2$.

- for $y_{d2}(t) = a_2 t^2 = \left(\frac{A_2}{2!}\right)t^2$, ess for the $(N=P=2)$ case is

$C_0 = \frac{A_2}{K_{BL}}$, In our example, $A_2 = a_2(2)! = 2a_2$, and $K_{BL} = 7.06$

$\therefore\ ess(t) = \frac{2a_2}{7.06} = \underline{0.283\,a_2}$

d) $d(t) = b_0 + b_1 t$. Given 1 pole @ origin for $H(s)$ we can perfectly track b_0, but not $b_1 t$.

$S_i(s) = \frac{G(s)}{1+L(s)} = \frac{4/s(s+2)^2}{1+7.06(s+.7279)^2/s^2(s+2)^2} = \frac{4s}{s^2(s+2)^2 + 7.06(s+.7279)^2}$

$D_1(s) = \frac{b_1}{s^2}$

then, $e_{iss}(t)$ will be the steady-state terms from $\mathcal{L}^{-1}\{S_i(s)D_1(s)\}$,

$Z = S_i(s)D(s) = \frac{4b_1}{s(s^2(s+2)^2 + 7.06(s+.7279)^2)} \rightarrow$ residue formula $[sZ]_{s=0} = \frac{4b_1}{7.06(.7279)^2}$

thus, $e_{iss}(t) = \frac{4b_1}{7.06(.7279)^2} = \underline{1.07\,b_1}$

e) Per the problem specifications $|S_i(j\omega)| \leq .01$, or $|S_i(j\omega)|_{dB} \leq -40\,dB$

we use matlab to find ω_D for which $|S_i(j\omega_D)| = -40\,dB$ (attached)

range $\underline{0 \leq \omega_D \leq .0093\ rad/s}$

(see matlab)

↑

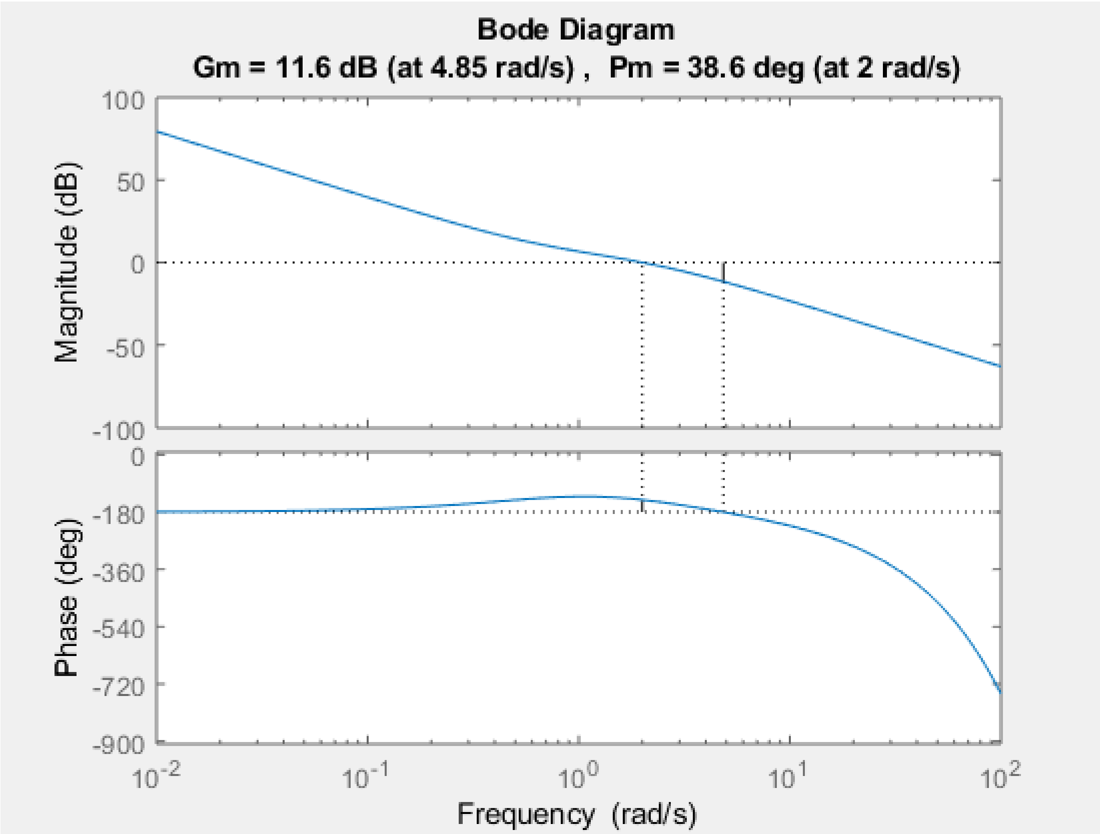
largest ss error corresponds to $|S_i(j\omega)|_{max}$; this max occurs at $\underline{\omega_p = 0.74\ rad/s}$

f) minimum distance to -1 point $\Rightarrow$ $d_{min} = |S(j\omega)|^{-1}_{max}$, where $S(s) = \frac{1}{1+L(s)}$

the distance d_{min} is $\frac{1}{|S(j\omega)|_{max}} = \frac{1}{10^{3.46/20}} = \underline{0.671}$

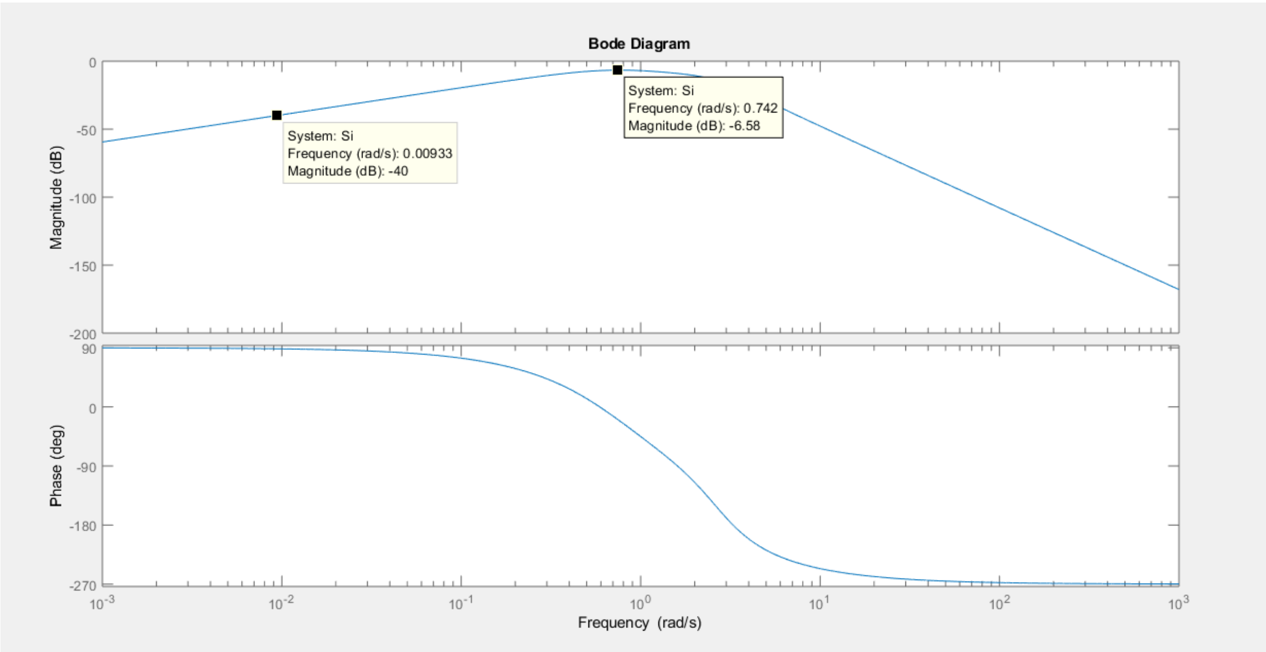
PS9Q3 part B

```
s = tf('s');
L_q2 = 7.06*(s+0.7279)^2/(s^2*(s+2)^2);
timedelay = exp(-s*0.1);
Leff = L_q2*timedelay;
margin(Leff)
```



PS9Q3 part E

```
s = tf('s');
Si = 4*s/(7.06*(s+0.7279)^2+(s^2*(s+2)^2));
w = logspace(-3,3,10000);
bode(Si,w)
```



PS9Q3 part F

```
s = tf('s');
L_q2 = 7.06*(s+0.7279)^2/(s^2*(s+2)^2);
S = feedback(1,L_q2);
w = logspace(-3,3,10000);
bode(S,w)
```