−90 −120

1: 30 Points

For the system $L(s) = \frac{100(0.5s+1)}{s(0.2s+1)(s+10)}$:

$\frac{50s+100}{s(0.2s^2+2s+s+10)} = \frac{50s+100}{0.2s^3+3s^2+10s}$

(a) Sketch the Bode plot using asymptotic approximations and the Nyquist plot based on your Bode plot.

(b) Find the gain margin and phase margin based on your plots, clearly indicating where these were measured. Compare the approximate results from your graphs to the values obtained using the *margin* command in Matlab.

(c) Based on the phase margin, calculate the delay margin of the system. Simulate the step response of the closed loop system.

2: 20 points

For the plant $G(s) = \frac{100}{s(s+1)(s+10)}$:

(a) Design a continuous time lead compensator to achieve a phase margin of 60 degrees at a frequency of 5 rad / s. Show that your design meets the specifications with the margin command, and plot the step response of the system.

(b) Find a state space realization for the plant $G(s)$ (can use Matlab). Design a state space output feedback controller (observer + state feedback) that achieves (roughly) 20% overshoot and a settling time of 6 s. Build the closed loop system in Matlab (the *reg* command is useful - remember positive feedback!) or Simulink and plot an initial condition response (set one of the plant states to 1 initially - the *initial* command will do this in Matlab).

3: 30 points

The file HDD_freqresp.mat includes a (fake) frequency response model of a single-stage hard disk drive. The model's lowest frequency mode is due to compliance in the bearing, whereas the higher frequency modes are due to flexibility in the structure between the actuator and the recording head. A "good" controller for this system will achieve an open loop crossover frequency of 1 kHz, 6283 rad/s a phase margin of at least 40 degrees, and a gain margin of at least 6 dB.

−140°

(a) Modern hard disk drives have relatively high sampling rates. Assuming the sampling rate is $F_s = 50$ kHz (you need to add this to your plant model with a delay approximation - it is not present in the measured data), design a (continuous time) feedback controller that includes integral action and meets the specifications. Use margin to demonstrate that the goals were acheived.

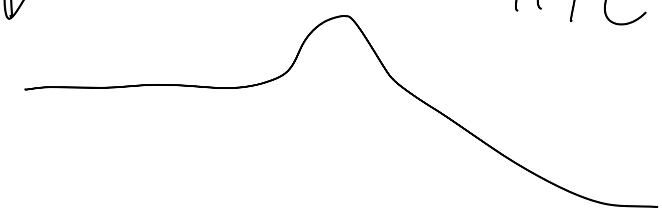
(b) Plot the sensitivity and complementary sensitivity function for your design. What are their peak values over frequency?

(c) FRD-based design can be quite effective in practice, but Matlab will not allow you to use an

FRD model in simulink or to simulate performance in Matlab. For simulation we will need to find a state space approximation to the data. Read about the command *fitfrd*, and use that command to fit an approximation to the complementary sensitivity function T. Plot the Bode plot of the FRD and its approximation on the same axes, and use the approximation to plot the step response.

L

Fit $T = \frac{PC}{1+PC}$



```
close all
clear all
clc

num = [50 100];
den = [0.2 3 10 0];
G = tf(num, den);

figure;
bode(G), grid
figure;
margin(G)
figure;
nyquist1(G)
figure;
step(feedback(G,1))
```

1(a) $L(s) = \dfrac{100(0.5s+1)}{s(0.2s+1)(s+10)}$: $= \dfrac{50(s+2)}{\frac{1}{5}s(s+5)(s+10)}$

$L(s) = 250 \dfrac{s+2}{s(s+5)(s+10)}$

$K = 250$

$z = -2$

$p = 0, -5, -10$

$G(s) = 250 \dfrac{|jw+2|}{|jw| \cdot |jw+5| \cdot |jw \cdot 10|}$

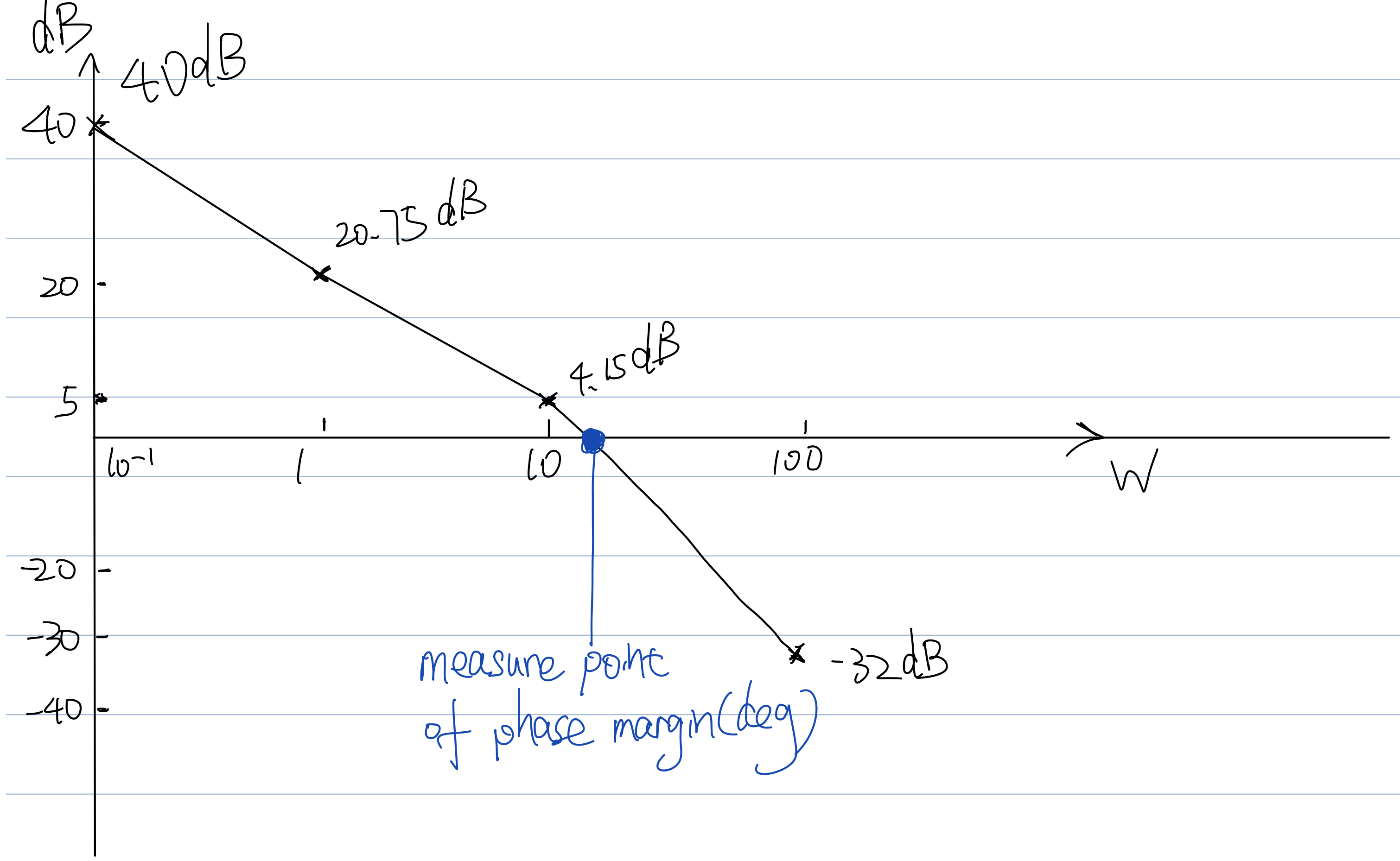
$G(s) = 250 \sqrt{\dfrac{w^2+4}{w^2 \cdot (w^2+25) \cdot (w^2+100)}}$

$G(s) = 250 \sqrt{\dfrac{w^2+4}{w6+125w^4+2500w^2}}$

W	0.1	1	10
dB	$G=\sqrt{\frac{4}{25}}\cdot 250$ $G=99.97$ $20\log(G)=40\text{ dB}$	$G=\frac{\sqrt{5}}{\sqrt{2626}}\cdot 250$ $G=10.91$ $20\log(G)=20.75\text{ dB}$	$G=\sqrt{\frac{104}{2500000}}\cdot 250$ $G=1.61$ $20\log(G)=4.15\text{ dB}$

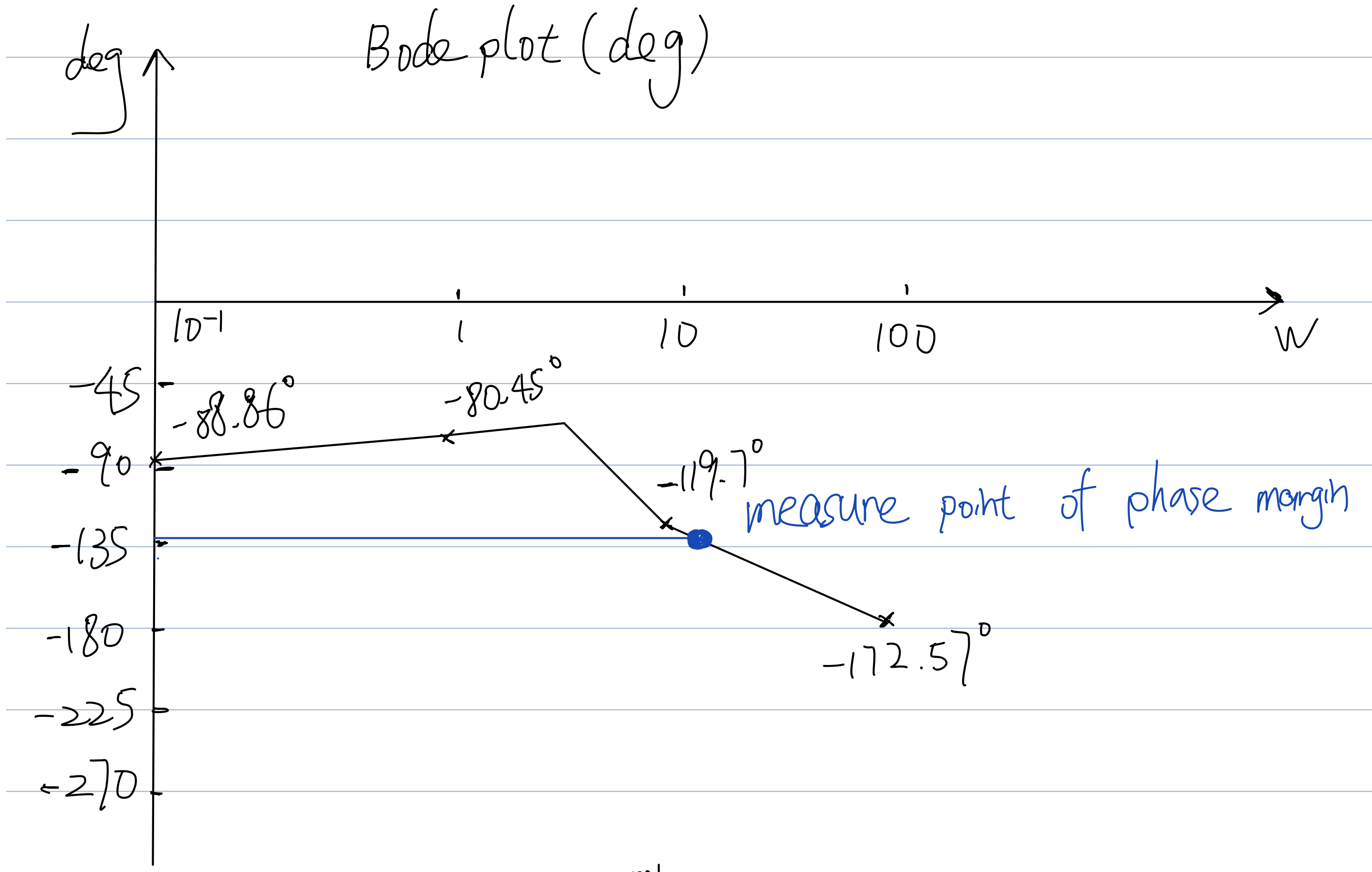
W	100
dB	$G=\sqrt{\frac{10004}{1.013\times 10^{12}}}\cdot 250 = 0.025$ $20\log(G)=-32\text{ dB}$

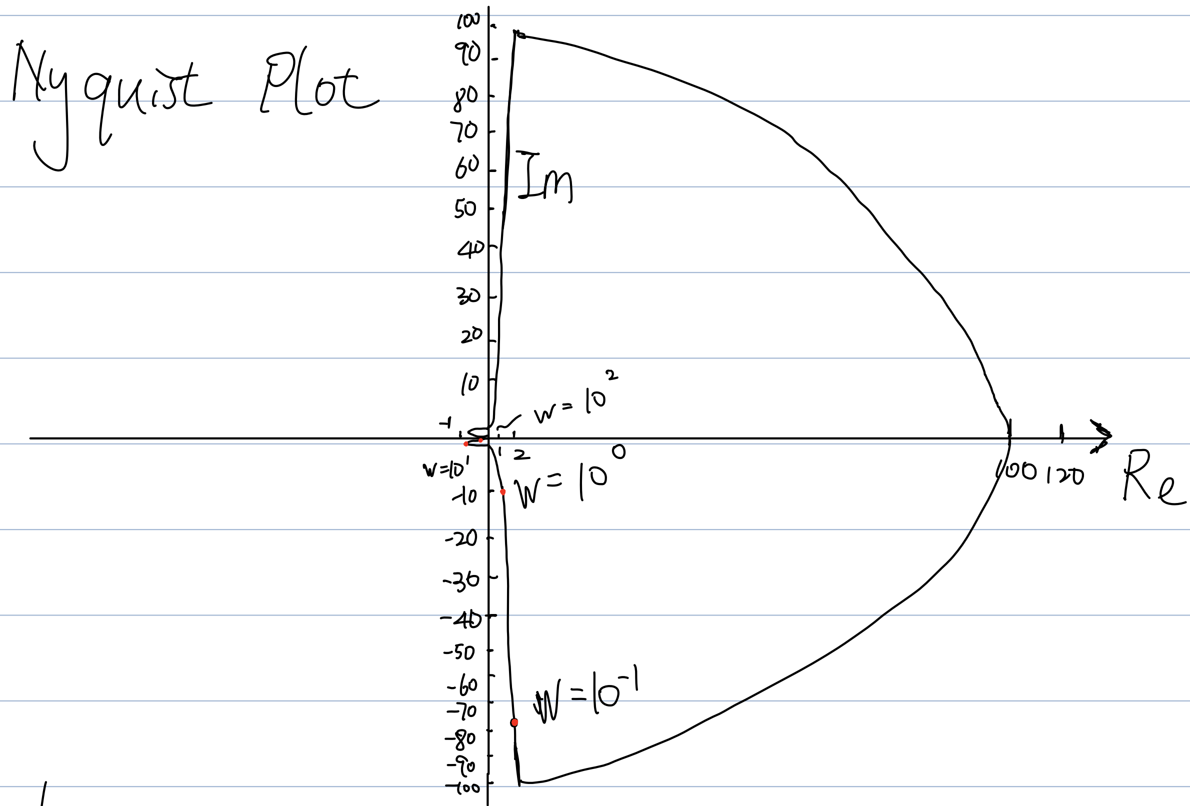
Bode plot (dB)



$$\Phi(w) = \tan^{-1}\left(\frac{w}{2}\right) - 90^{\circ} - \tan^{-1}\left(\frac{w}{5}\right) - \tan^{-1}\left(\frac{w}{10}\right)$$

W	0.1	1	10	100
Φ	-88.86	-80.45	-119.7	-172.57





Gain	99.97	10.91	1.61	0.025
dB	40dB	20.75dB	4.15dB	-32dB
Deg	-88.86°	-80.45°	-119.7°	-172.57°

(b)

Phase margin is the amount of phase shift when the gain passes through 0dB

Gain margin is done by calculating the vertical distance between the magnitude curve (on the Bode magnitude plot) and the x-axis at the frequency where the Bode phase plot = 180°.

Gain margin : Infinite

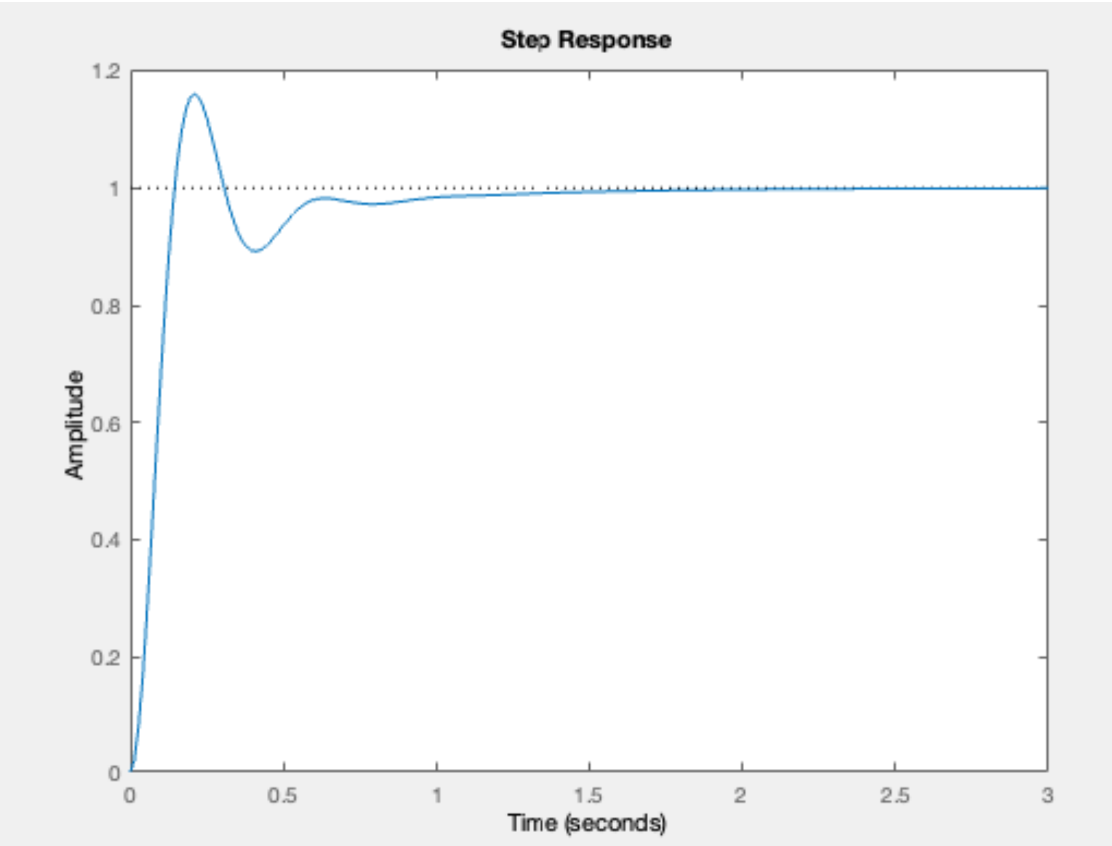
Phase margin : $-140 + 180 = 40°$

In matlab, phase margin : $47.4°$

gain margin infinite

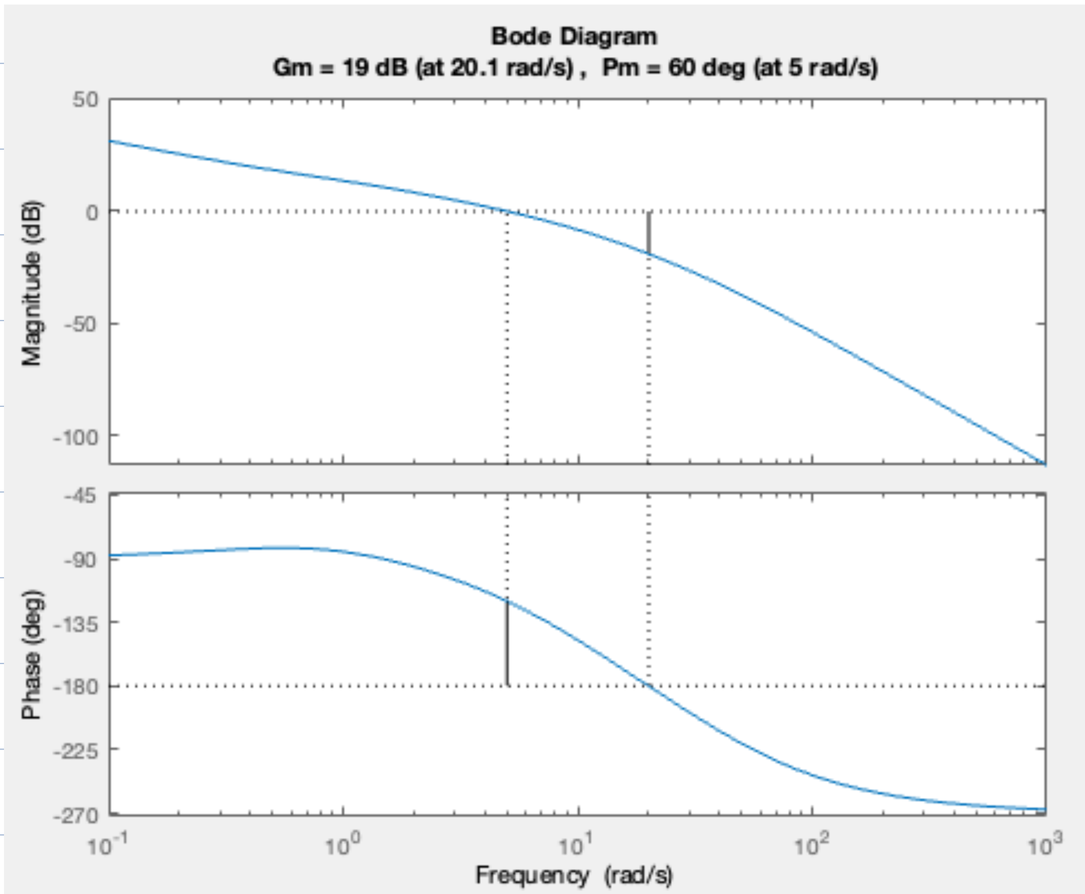
(c) $T_{dm} = \frac{\text{Phase Margin}}{W_{gc}} \cdot \frac{\pi}{180} = \frac{47.4°}{14.2} \cdot \frac{\pi}{180°} = 0.0583$

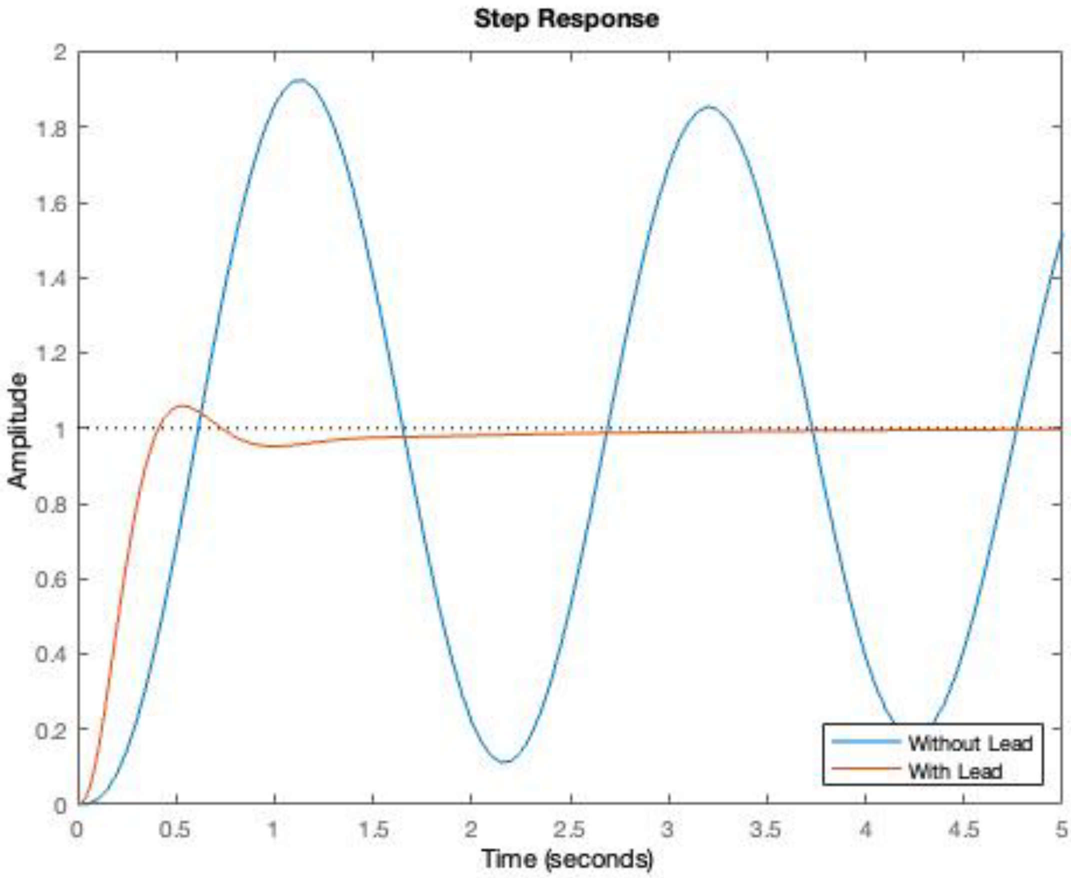
From MatLab, Delay margin $= 0.0595$, very close to our approximation



2(a)

```
num = [100];
den = [1 11 10 0];
G_before = tf(num, den);
[~,phase] = bode(G_before,5);

ang = -180+60-phase; %how much need to compensate
f = 5;
C = lead(ang,f);
G_after = C * G_before;

[mag,~] = bode(G_after,5);
K = 1/mag;

figure
margin(K * G_after)

sys1 = feedback(G_before,1);
sys2 = feedback(K * G_after,1);
figure
step(sys1, sys2, 5)
legend('Without Lead','With Lead','Location','SouthEast')
```





2.(b) $T_s = 6 = \frac{4}{\xi w_n}$ $\quad \xi w_n = \frac{2}{3}$

$OS\% = 20\%$

$= 100 e^{-\frac{\xi \pi}{\sqrt{1-\xi^2}}} \leq 20$

$\xi = 0.456$

$\therefore w_n = 1.462$

Desired CL system $= \dfrac{1.462^2}{s^2 + 2 \cdot \frac{2}{3} s + 1.462^2}$

$$= \frac{2.137}{s^2 + 1.33s + 2.137}$$

TF to SS :
$$A = \begin{bmatrix} -11 & -10 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 100 \end{bmatrix}$$

Use pole placement, want pole at

$$\frac{-1.33 \pm \sqrt{1.33^2 - 4 \times 2.137}}{2} = \frac{-1.33 \pm \sqrt{-6.78}}{2}$$

$$= -0.665 \pm 1.3i$$

Place third pole : -3.4

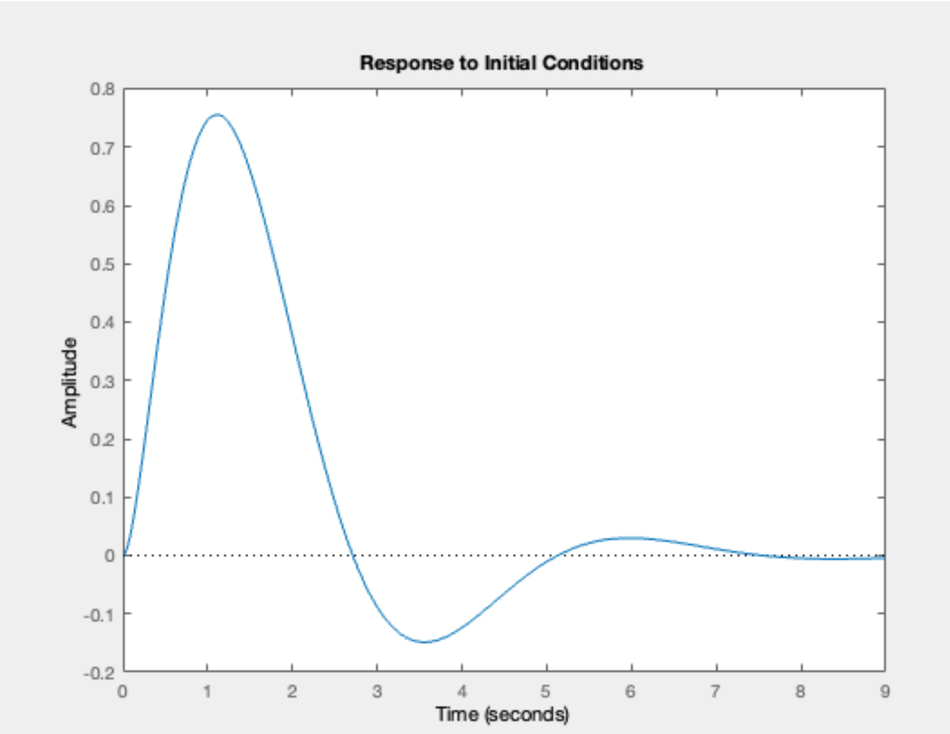
$$\therefore K = [-6.27 \quad -3.35 \quad 7.25]$$ State feedback

Design observer, in order to have faster than the system, place at least five time further from poles.

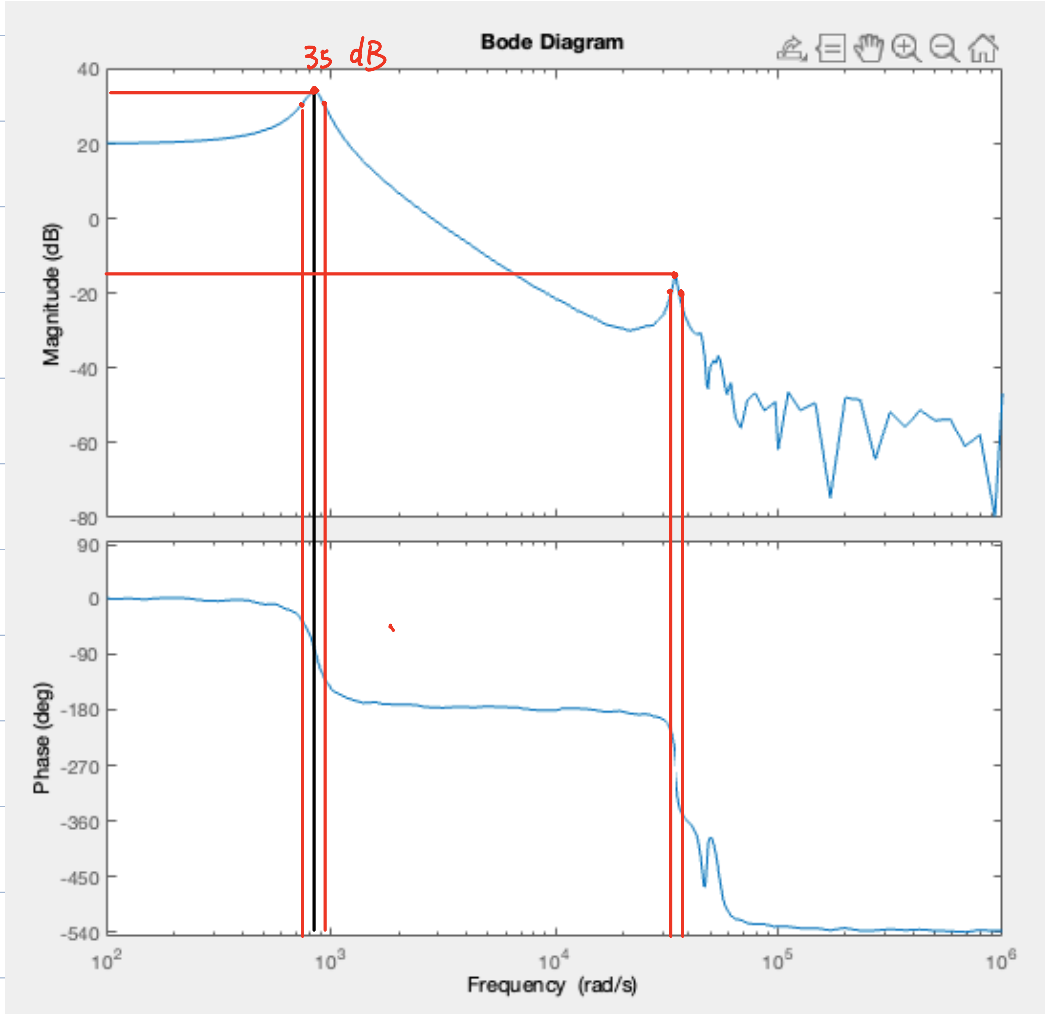
Place observer poles $= [-30.5 \quad -30.6 \quad -30.7]$

$$\therefore L = \begin{bmatrix} 68.31 \\ 19.1 \\ 0.808 \end{bmatrix} \text{ observer}$$

```
num = [100];
den = [1 11 10 0];
G = tf(num,den);
[A,B,C,D] = tf2ss(num,den);
sys = ss(A,B,C,D);
state_poles = [-0.665+1.3i -0.665-1.3i -3.4];
K = place(A, B, state_poles);
observer_poles = [-30.5 -30.6 -30.7];
L = place(A', C', observer_poles)';
rsys = reg(sys, K, L);
sys = feedback(G * rsys, -1);
initial(sys,[1; 0; 0; 0; 0; 0])
```



3.(a)



Original system

By eye inspection: Notch 1 : Width: 924 − 741 = 183 rad/s

Depth : 12.2 dB

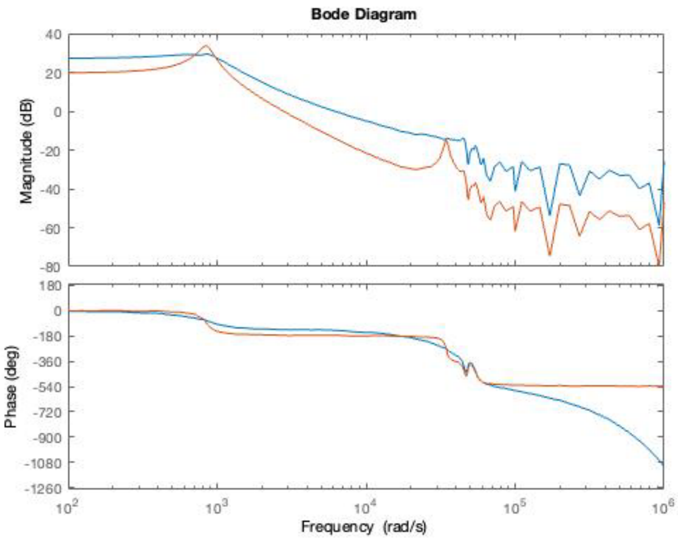
freq : 834 rad/s

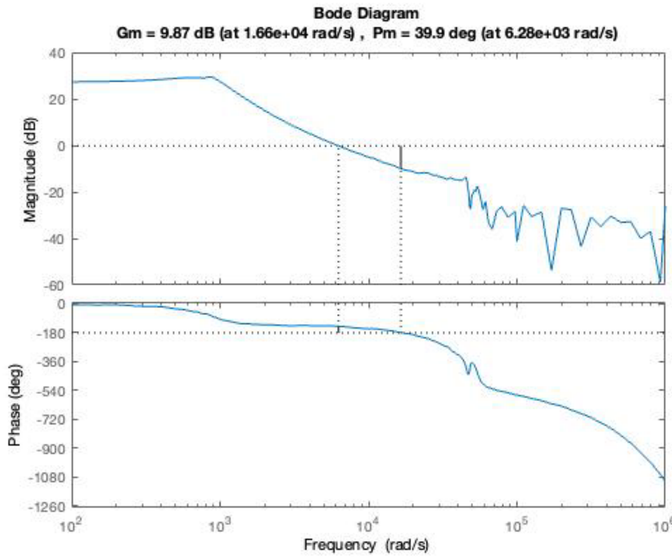
Notch 2 : Width : 35600 − 33400 = 1200 rad/s

Depth 14.7 dB

freq : 34400 rad/s

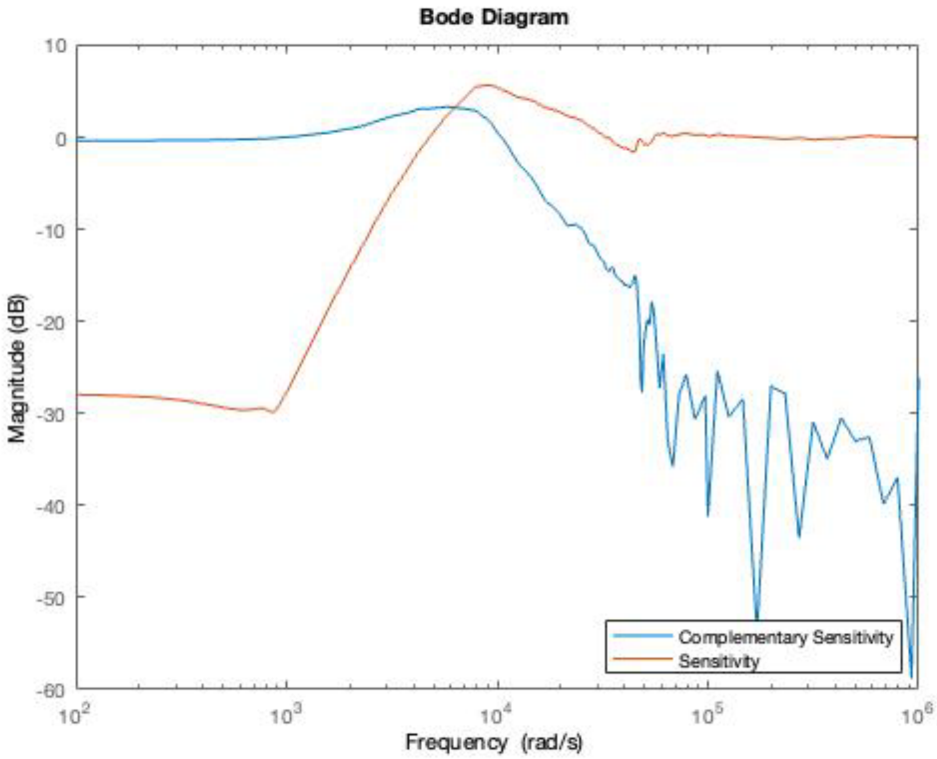
```
close all
clear all
clc

load('HDD_freqresp.mat');
s = tf("s");
sampling_freq = 50000;
delay = exp(-1/2/sampling_freq * s); %approximated delay
Wc = 1000 * 2 * pi;
PM = 40;

sys = HDD_freqresp;
%In the bode plot, we found 2 notches need to be anti-notched
notch1 = notch(12.2, 183, 834);
notch2 = notch(20, 2100, 34400);

G = HDD_freqresp * delay;
[~, phase] = bode(G * notch1 * notch2, Wc); %after eliminating the notch, we use lead compensator to tune it

ang = -180+PM-phase; %how much need to compensate

C = lead(ang,Wc);
G_after = C * G * notch1 * notch2;
[mag, ~] = bode(G_after, Wc);
K = 1/mag;
figure;
bode(K * G_after, sys)
figure;
margin(K * G_after)
```





(b)

```
%% q3b

figure;
bodemag(feedback(K * G_after, 1), 1 - feedback(K * G_after, 1))%sensitivity function
legend('Complementary Sensitivity','Sensitivity','Location','SouthEast')

[T_peak,T_freq_peak] = getPeakGain(feedback(K * G_after, 1));
[S_peak,S_freq_peak] = getPeakGain(1 - feedback(K * G_after, 1));
T_db_peak = 20 * log10(T_peak)
T_freq_peak
S_db_peak = 20 * log10(S_peak)
S_freq_peak
```




```
T_db_peak =

    3.3326


T_freq_peak =

   5.8195e+03


S_db_peak =

    5.6868


S_freq_peak =

   9.2266e+03
```

(c)

```
%% q3c
% K * G_after is a frequency-response model
omeg = logspace(2, 6);
approx = fitfrd(K * G_after, 8); %approximation want to fit
approxg = frd(approx, omeg); % change to frequency response model

figure;
bode(K * G_after,'r-',approxg,'b:')
figure;
step(feedback(approx, 1))
```