EE324 Control Systems Lab

Problem Sheet 9

Sheel Shah — 19D070052

October 18, 2021

1 Question 1

Code

```
s = poly(0, 's')

// part A

g = 10 / ((s) * (s/5 + 1) * (s/20 + 1))

G = syslin("c", g)

nyquist(G, 0.1, 500)

disp(g_margin(G)) // 7.9588002 dB

disp(p_margin(G)) // 22.535942 deg


// part B

scf()

c = (s + 3)/(s + 1)

nyquist(c * G, 0.1, 500)

disp(g_margin(c * G)) // 2.0762546
```

```
disp(p_margin(c * G)) // 4.0247332


// part C
scf()
c = (s + 1)/(s + 3)
nyquist(c * G, 0.1, 500)
disp(g_margin(c * G)) // 11.759539
disp(p_margin(c * G)) // 43.173118
```

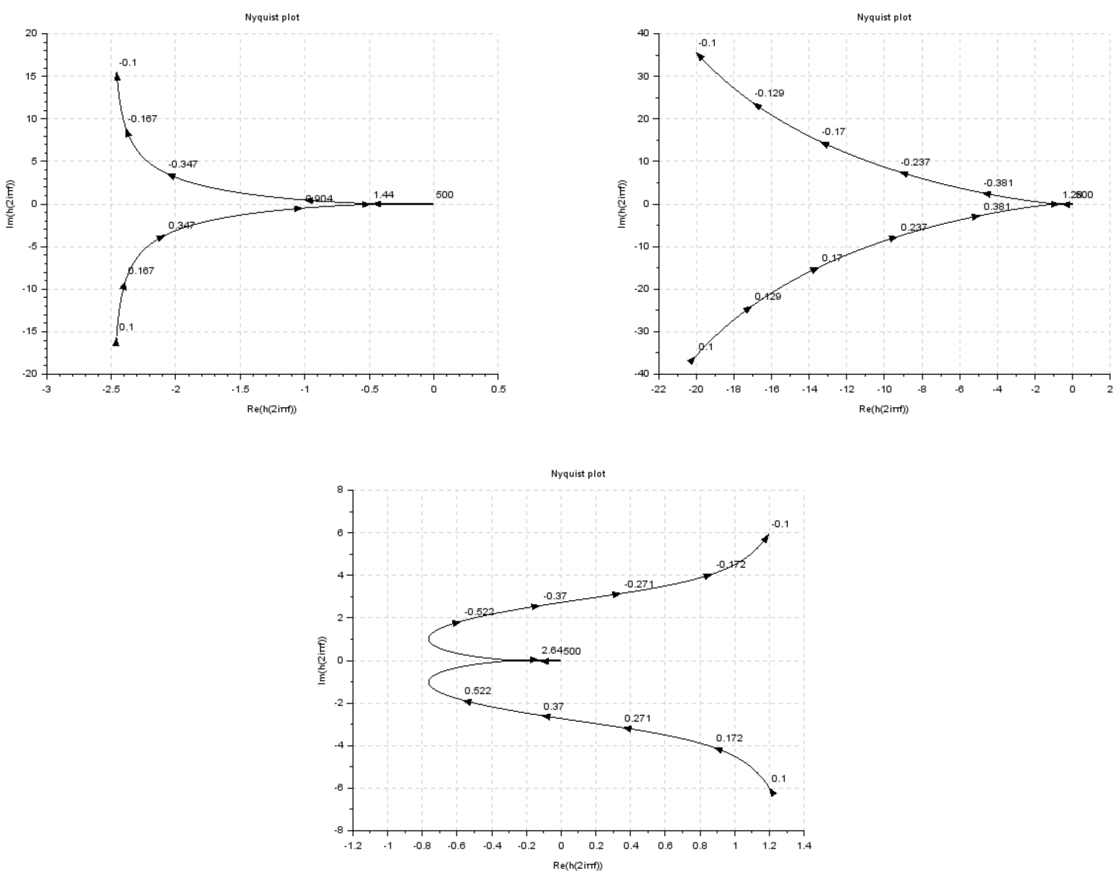

Figure 1: Nyquist plots for part a, b, c respectively

The values of the gain and phase margins are mentioned in the code.

2 Question 2

Code

```
s = poly(0, 's')

// we create poles just before 50Hz so that magnitude drops

// and then we add zeros at 50 Hz to bring the magnitude back

g = (s^2 + (100*%pi)^2) / (s^2 + s + (100*%pi)^2)

G = syslin("c", g)

gainplot(G, 10, 100)

scf()

g = (s^2 + (100*%pi)^2) / (s^2 + s + (100*%pi)^2)

G = syslin("c", g)

gainplot(G, 10, 100)

g = (s^2 + (100*%pi)^2) / (s^2 + 10*s + (100*%pi)^2)

G = syslin("c", g)

gainplot(G, 10, 100)
```

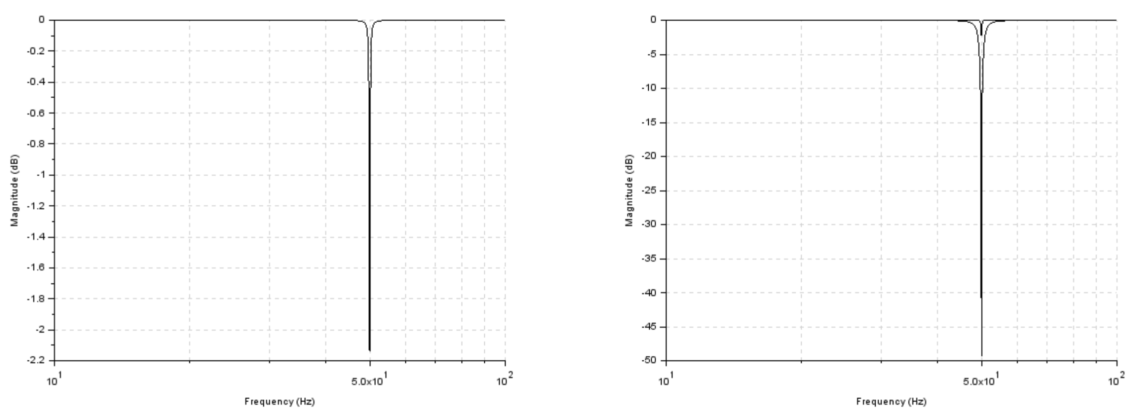

Figure 2: Increasing the coefficient of s significantly increases the drop in gain. This is because the real part of the poles moves farther away from the origin

3 Question 3

3.1 Code and calculation

```
s = poly(0, 's')


c = 100 / (s + 30)

C = syslin("c", c)

scf(0)

bode(C)

// [pm, gcf] = p_margin(C)

// T = 1/360 * pm / gcf

// disp(T)

T = 0.019660467


scf(1)

c = 100 / (s + 30)

num = 1 - 0.5*s*T + (1/9)*s^2*T**2 - (1/72)*(s*T)^3 + (1/1008)*(s*T)^4-(1/30240)*(s*T)^5

den = 1 + 0.5*s*T + (1/9)*s^2*T**2 + (1/72)*(s*T)^3 +(1/1008)*(s*T)^4+(1/30240)*(s*T)^5

cg = c * num / den

CG = syslin("c", cg)

bode(CG)

disp(p_margin(CG))

disp(g_margin(CG))
```

Plots

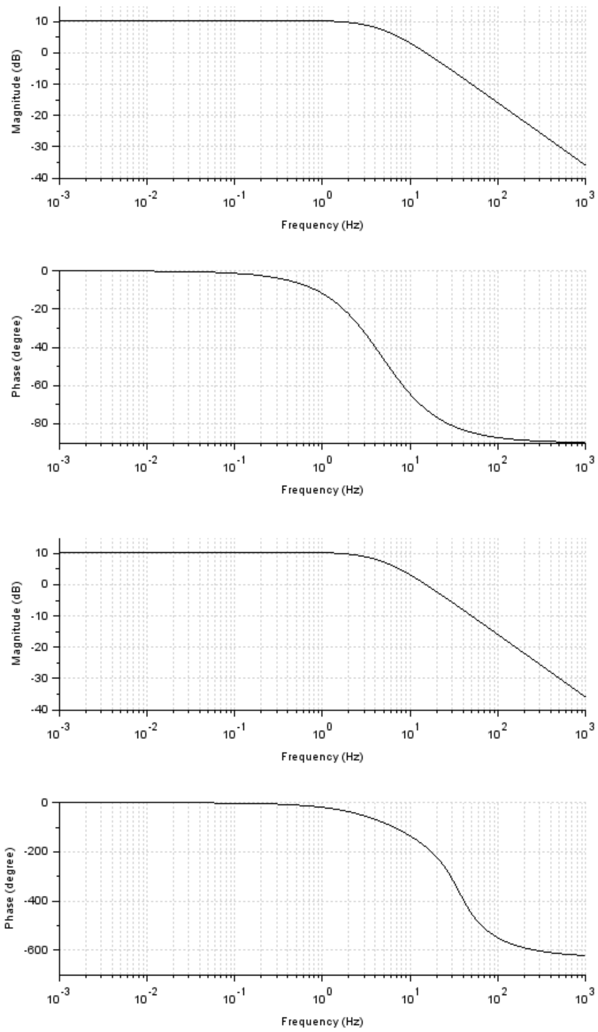

Figure 3: Bode plots before and after delay. The margins become almost zero after adding delay

4 Question 4

Code

```
s = poly(0, 's')


g = 1 / (s^3 + 3*s^2 + 2*s)

G = syslin("c", g)

scf(0)

evans(G)


scf(1)

nyquist(G)


scf(2)

show_margins(G)

bode_asymp(G)


scf(3)

show_margins(G)

bode(G)
```

Plots

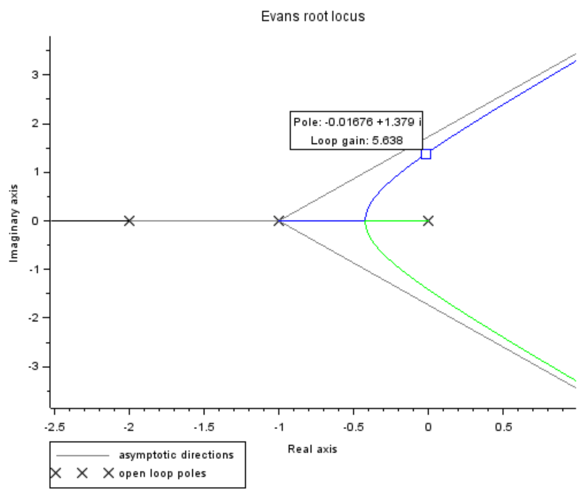

Figure 4: Root Locus. gain margin = 20 log (crossing$_{g}ain$) = 15.02

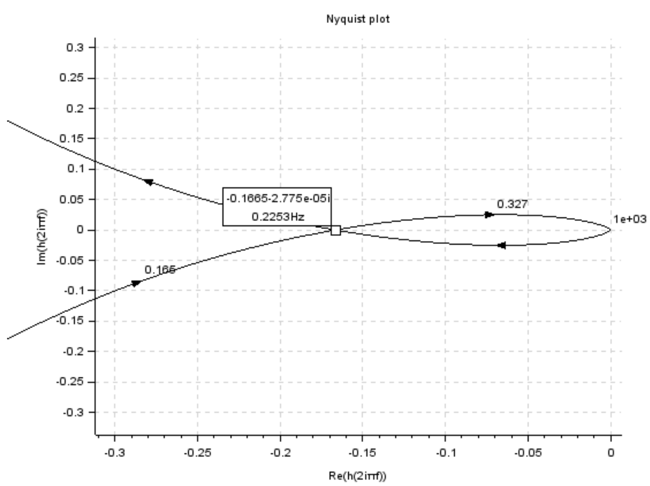

Figure 5: Nyquist. gain margin = 1 / —zero-crossing— = 6

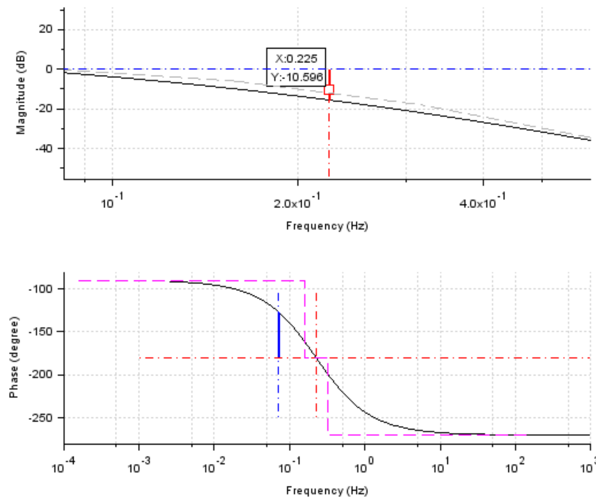

Figure 6: Asymptotic Bode plot. gain margin = 10.696

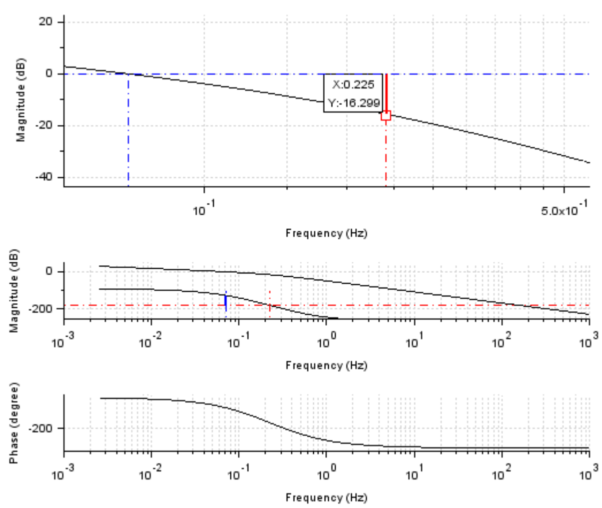

Figure 7: Bode plot. gain margin = 16.299

5 Question 5

5.1 Part A

```
s = poly(0, 's')

g = (10*s + 2000) / (s^3 + 202*s^2 + 490*s + 18001)

G = syslin("c", g)

bode(G)

[gm, pcf] = g_margin(G)

[pm, gcf] = p_margin(G)

disp(pcf, pm, gcf, gm)
```

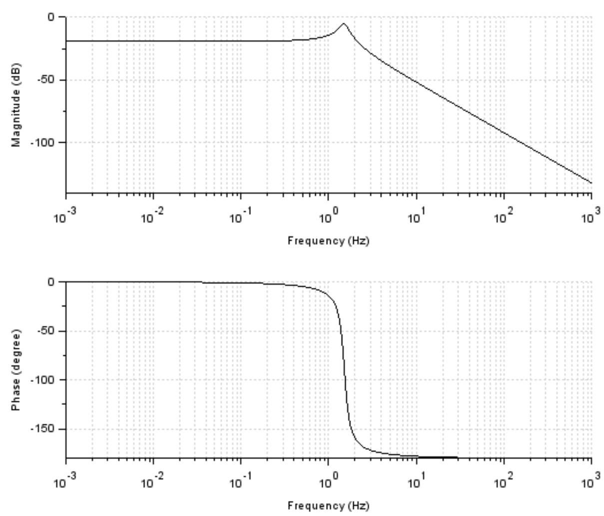

Figure 8: Bode plot. PM: -, GM: infinity, GCF: -, PCF: -

5.2 Parts B, C

TF = KG / (1 + KG)

sse = 1 / (1 + KG(0)) = 0.1

KG(0) = 9

K = 81.0045

```
s = poly(0, 's')

g = (10*s + 2000) / (s^3 + 202*s^2 + 490*s + 18001)

G = syslin("c", g)

bode(G)

[gm, pcf] = g_margin(G)

[pm, gcf] = p_margin(G)

disp(pcf, pm, gcf, gm)
```

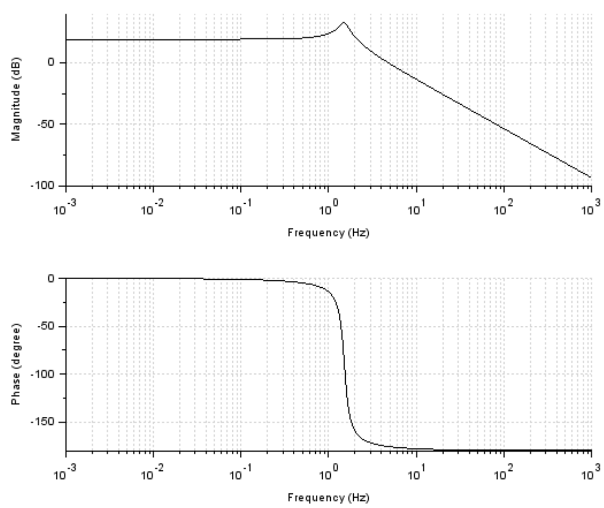

Figure 9: Bode plot. PM: 4.24, GM: infinity, GCF: 4.76, PCF: -

5.3 Parts D, E

We choose zero at -2, by trial and error

```
s = poly(0, 's')

g = (1 + s/2)*(10*s + 2000) / (s^3 + 202*s^2 + 490*s + 18001)

G = syslin("c", g)

K = 81.0045

G = (K*G)

disp(roots(G.den))

bode(G)

[gm, pcf] = g_margin(G)

[pm, gcf] = p_margin(G)

disp(pcf, pm, gcf, gm)
```

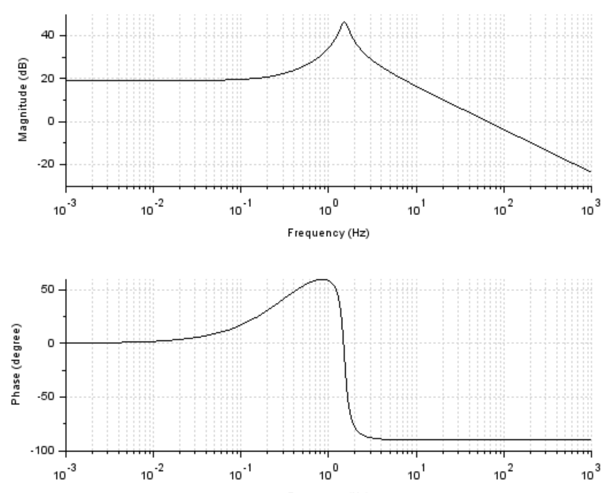

Figure 10: Bode plot. PM: 90.0001, GM: infinity, GCF: 64.5, PCF: -