# EE324 CONTROL SYSTEMS LAB
# Problem Sheet 7

Yatish Vaman Patil | 190070076

## Question 1: Gain and Phase Margin

$$\mathrm{Gs} = \frac{1}{\mathrm{s}(s^2 + 4s + 8)}$$

### Part A)

In this part, we have to find gain K such that the Gain margin and phase margin of the closed-loop characteristic function of Gs is 0

We have closed-loop characteristics function as

Ps= K*Gs

$$\mathrm{Ps} = \frac{\mathrm{k}}{\mathrm{s}(s^2 + 4s + 8)}$$

$$\mathrm{Ps} = \frac{\mathrm{k}}{s^3 + 4s^2 + 8s}$$

S=jw

$$\mathrm{Ps} = \frac{\mathrm{k}}{(-4w^2) + j(8w - w^3)}$$

For GM and PM equal to 0, $w_{gcf}=w_{pcf}=w$

The imaginary part will be 0

$w^2=8$

f=0.45 Hz

magnitude pf Ps should be one and phase should be $-180^0$

hence Ps=-1

$$\mathrm{Ps} = \frac{\mathrm{k}}{-4w^2} = -1$$

$k=4w^2$

**k=32**

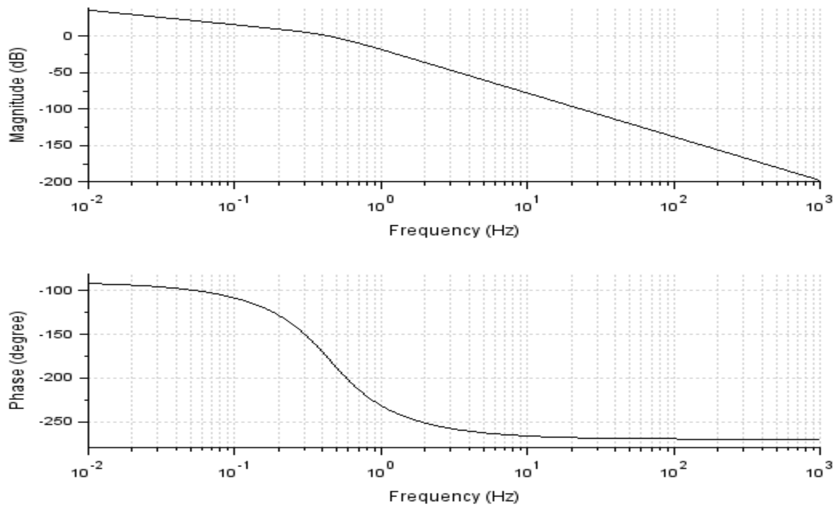



Fig. Bode Plot of Characteristic function

**Code for the same:**

```
s=%s;
Gs=syslin('c',1/(s*(s^2+4*s+8)));
CharEq=32*Gs;
bode(CharEq,0.01,1000);
```

## Part B)

It is not possible to have a non zero gain margin and zero phase margin or vice versa. If the gain margin is zero, then from its definition, we know that at $-180^{0}$, we have a gain equal to 0 dB. But this statement also implies a phase margin equal to zero.
We also know that GM and PM decrease monotonically for increasing value of k. Both of them intersect at a point where both are zero. Hence mathematically, it won't be possible to have PM zero and GM non zero.

## Part C)

We have a system transfer function for k=32

$$\text{Ts} = \frac{32}{s^3 + 4s^2 + 8s + 32}$$

Poles of this transfer function are

**-4**

**+ j2.83**

**– j2.83**

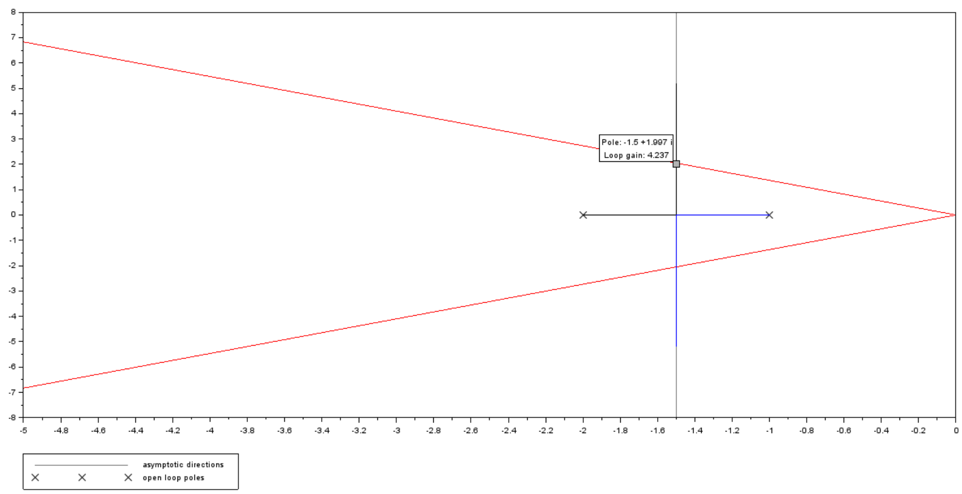
As we can see, the dominant poles of the System are on an imaginary axis; the System becomes unstable

# Question 2: Lag Compensator

## Part A)

$$Gs = \frac{1}{s^2 + 3s + 2}$$

For this open-loop transfer function Gs, we have to find gain K such that the %OS=10. We can solve this using the Root Locus Method in Scilab




The desired value of gain K=4.297

## Code for the same:

```
s=%s;
Gs=syslin('c',1,s^2+3*s+2);
OS=0.11;
slope=%pi/2.30;
x=-5:0.01:0;
OS_line1=slope.*x;
OS_line2=-1*slope.*x;
plot(x,OS_line1,'r');
plot(x,OS_line2,'r');
evans(Gs,kpure(Gs));
```

## Part B)

Steady state error for unity feedback loop is

$$e(\infty) = \frac{1}{1 + kG(0)}$$

$$e(\infty) = \frac{1}{1 + \frac{4.297}{2}}$$

$$e(\infty) = 0.317$$

Now we will introduce the Lag compensator of z/p ratio 20

Let z=-0.2 and p=-0.01

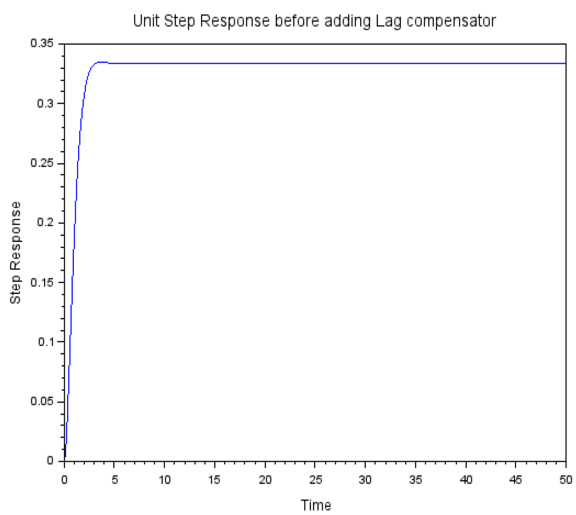
Our new Plant becomes

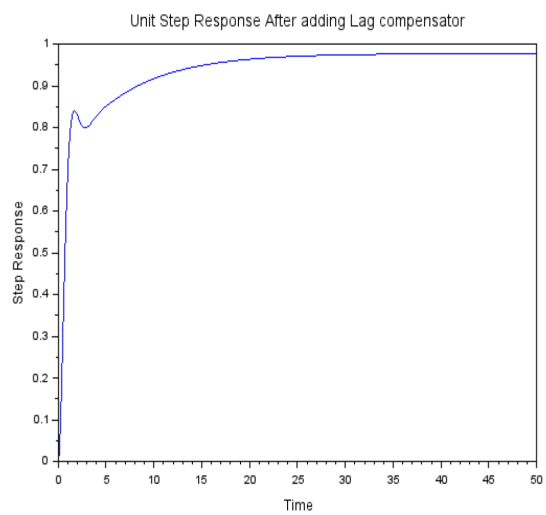
$$Gs' = \frac{s + 0.2}{(s + 0.01)(s^2 + 3s + 2)}$$

The new value of steady-state error is

$$e(\infty) = \frac{1}{1 + \frac{4.297 * 20}{2}}$$

$$e(\infty) = 0.0227$$




## Code for the same:

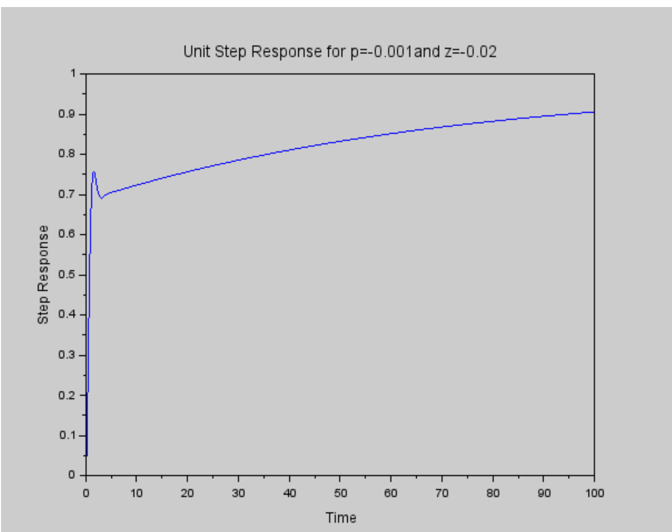
```
s=%s;
k=4.297;
Gs1=syslin('c',1,s^2+3*s+2);
Gs2=syslin('c',(s+0.2),(s+0.01)*(s^2+3*s+2));
time=0:0.01:100;
```

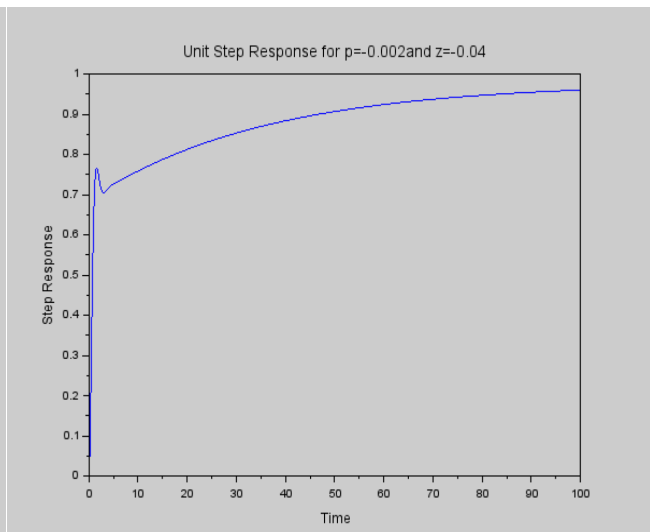
```
plot(time,csim('step',time,syslin('c',Gs1/(1+Gs1))));
xlabel("Time");
ylabel("Step Response");
xtitle("Unit Step Response before adding Lag compensator");
scf(2);
plot(time,csim('step',time,syslin('c',k*Gs2/(1+k*Gs2))));
xlabel("Time");
ylabel("Step Response");
xtitle("Unit Step Response After adding Lag compensator");
```

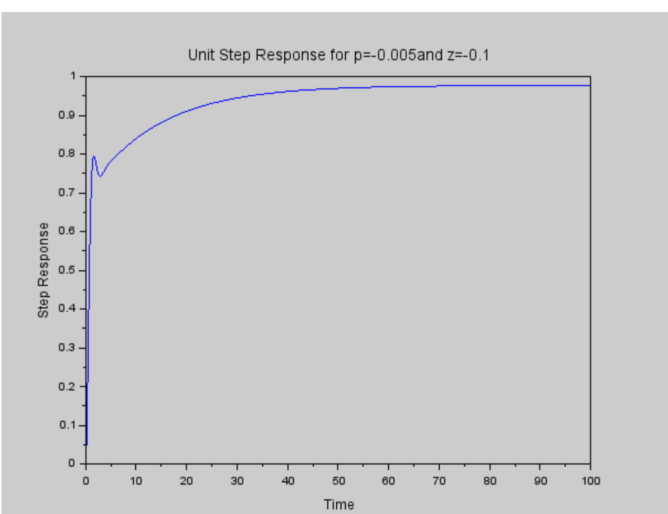
## Part C)

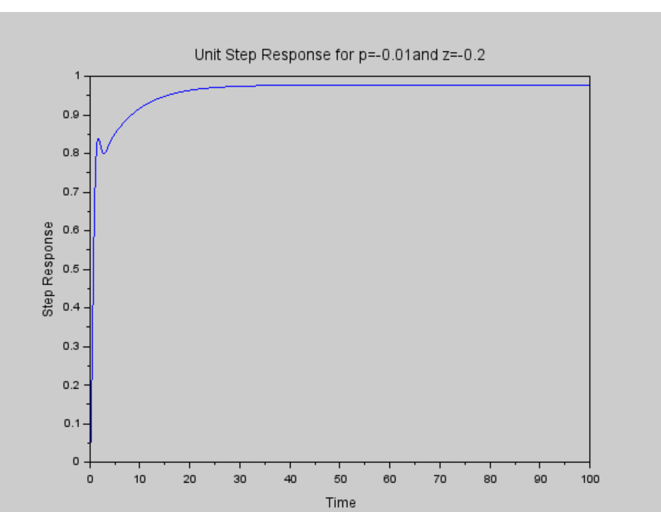
In this part, we have to vary the pole-zero Location by keeping the z/p ratio constant to 20

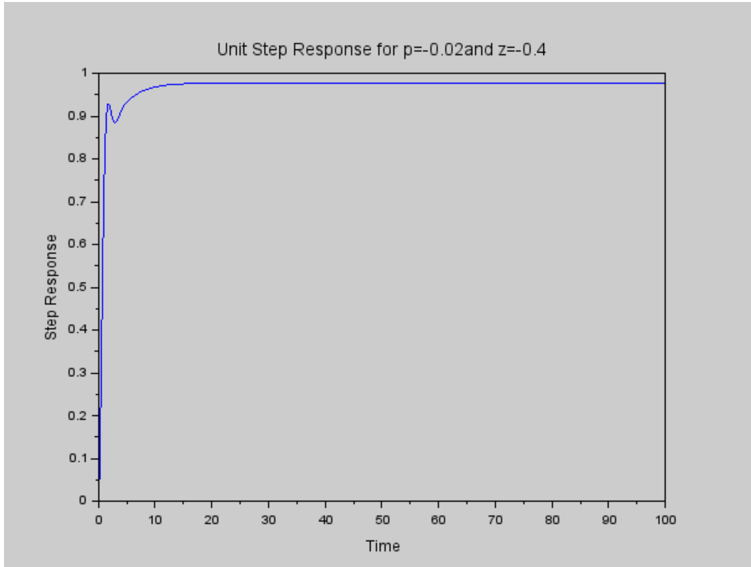
Using Scilab we get following Step response for the pole value=[0.001,0.002,0.005,0.01,0.02]






**Code for the same:**

```
s=%s;
pole_value=[0.001,0.002,0.005,0.01,0.02];
time=0:0.01:100;
k=4.297
for p=pole_value
    z=p*20;
    Gs1=syslin('c',(s+z),(s+p)*(s^2+3*s+2));
    Ts1=syslin('c',k*Gs1/(1+k*Gs1));
    figure,plot(time,csim('step',time,Ts1));
    xlabel("Time");
    ylabel("Step Response");
    xtitle("Unit Step Response for p="+string(-p)+"and z="+string(-
z));
end;
```

# Question 3: Lead Compensator

## Part A)

In this part, we have to design a lead compensator such that Closed-loop step response has 10% OS and half the settling time than non compensated System

$$Gs = \frac{1}{s^2 + 3s + 2}$$

Pole location for 10% OS calculated using intersection 10%OS line with RL

Pole location➔-1.5 $\pm$ j2.01

$\zeta w_n$=1.5

Settling time = $T_s$= $\frac{4}{\zeta w}$ = 2.66 sec

Now we have to design the PD controller such that settling time becomes half

The desired value of $T_s$=1.33 sec

$\zeta w_n$=3

The real part of poles = -3

Our %OS is the same, so by extrapolating, we get desired root locations as

Pole Location➔-3 $\pm$ j4.02

Original Poles of Gs are -1 and -2

**Let zero of lead compensator be at -4**

Now after summing angle extended at pole location by poles and zero, we get

Sum of angles = -144.39$^0$

Angle extended by the pole of the lead compensator

= 180-144.39

= 35.61$^0$

Pole location of lead compensator = -3 - $\frac{4.02}{\tan(35.61)}$

**= -8.613**

Our new Plant Gs' becomes

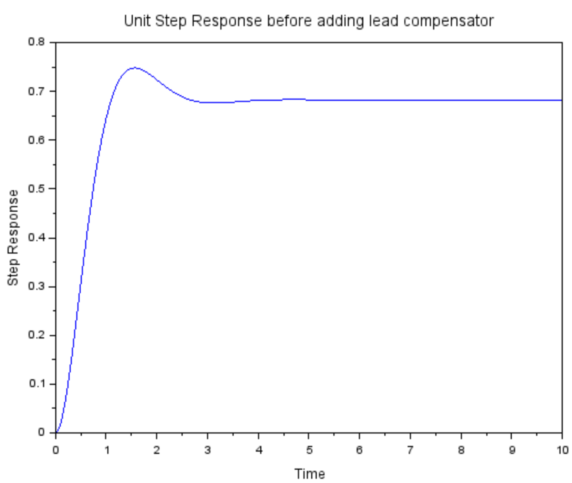
$$Gs' = \frac{k(s+4)}{(s+8.613)(s^2+3s+2)}$$

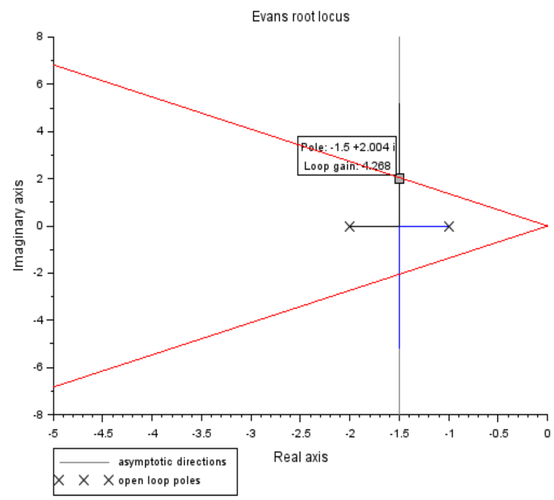
Using Scilab, we get

K = 32.41

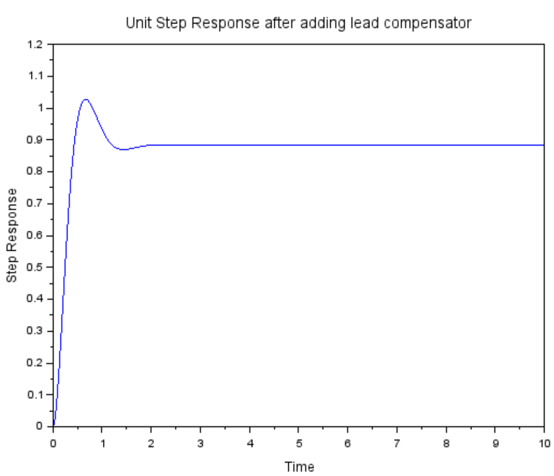
$$Gs' = \frac{32.41(s+4)}{(s+8.613)(s^2+3s+2)}$$

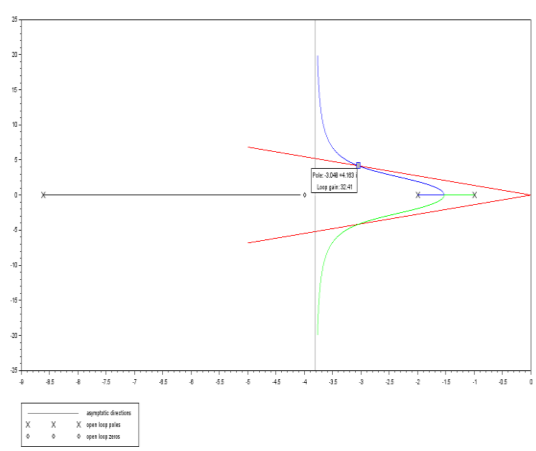
Before adding Lead Compensator




After adding Lead Compensator



**Code for the same:**

```
s=%s;
k=32.41;
Gs1=syslin('c',1,s^2+3*s+2);
Gs2=syslin('c',(s+4),(s+8.613)*(s^2+3*s+2));
time=0:0.01:10;
plot(time,csim('step',time,syslin('c',4.297*Gs1/(1+4.297*Gs1))));
xlabel("Time");
ylabel("Step Response");
xtitle("Unit Step Response before adding PD controller");
scf(2);
plot(time,csim('step',time,syslin('c',k*Gs2/(1+k*Gs2))));
xlabel("Time");
ylabel("Step Response");
xtitle("Unit Step Response after adding PD controller");
scf(3);
evans(Gs1,kpure(Gs1));
slope=%pi/2.30;
x=-5:0.01:0;
OS_line1=slope.*x;
OS_line2=-1*slope.*x;
plot(x,OS_line1,'r');
plot(x,OS_line2,'r');
scf(4);
slope=%pi/2.30;
x=-5:0.01:0;
OS_line1=slope.*x;
OS_line2=-1*slope.*x;
plot(x,OS_line1,'r');
plot(x,OS_line2,'r');
evans(Gs2,kpure(Gs2));
```

## Part B)

In this part, we have to design a PD controller

First of all, an uncompensated system is

$$\mathrm{Gs} = \frac{1}{s^2 + 3s + 2}$$

Pole location for 10% OS calculated using intersection 10%OS line with RL

Pole location➔-1.5 $\pm$ j2.01

$\zeta w_n = 1.5$

Settling time = $T_s = \frac{4}{\zeta w}$ = 2.66 sec

Now we have to design the PD controller such that settling time becomes half

The desired value of $T_s$=1.33 sec

$\zeta w_n = 3$

The real part of poles = -3

Our %OS is the same, so by extrapolating, we get desired root locations as

Pole Location➔-3 $\pm$ j4.02

Original Poles of Gs are -1 and -2

We will now measure the angle extended by these poles at the new pole location

Sum of angles = -220.45$^0$

Angle contribution from zero = 220.45$^0$ – 180$^0$

= 40.45$^0$

Now Location of zero = $-3 - (\frac{4.02}{\tan(40.45)})$

**= -7.83**

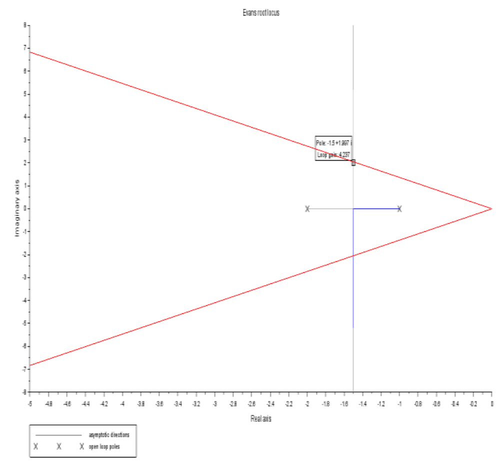
Our new Plant becomes

$$Gs' = \frac{k(s + 7.83)}{s^2 + 3s + 2}$$

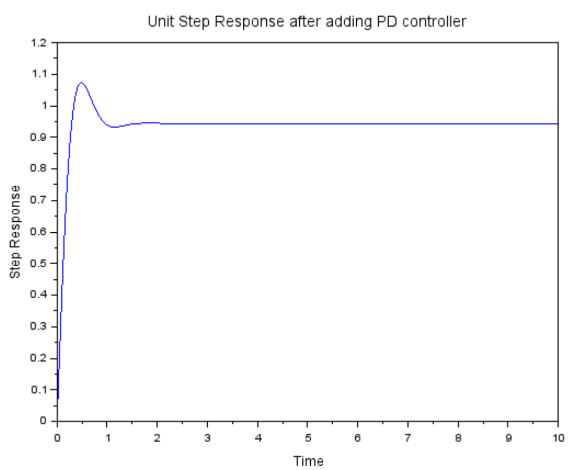
Using Scilab, we get k = 3.008

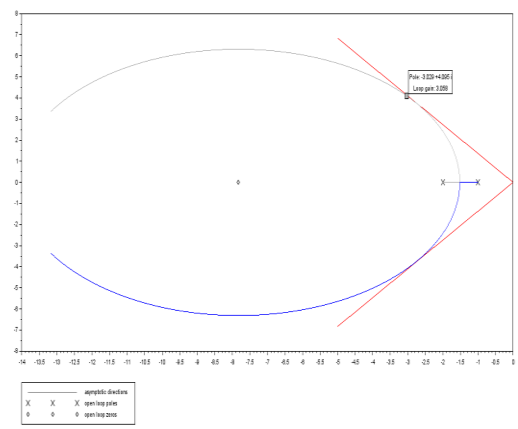
$$Gs' = \frac{3(s + 7.83)}{s^2 + 3s + 2}$$

Before Adding PD controller

After Adding PD controller



## Code for the same:

```
s=%s;
k=3;
Gs1=syslin('c',1,s^2+3*s+2);
Gs2=syslin('c',(s+7.83),(s^2+3*s+2));
time=0:0.01:10;
plot(time,csim('step',time,syslin('c',4.297*Gs1/(1+4.297*Gs1))));
xlabel("Time");
ylabel("Step Response");
xtitle("Unit Step Response before adding PD controller");
scf(2);
plot(time,csim('step',time,syslin('c',k*Gs2/(1+k*Gs2))));
xlabel("Time");
ylabel("Step Response");
```

```
xtitle("Unit Step Response after adding PD controller");
scf(3);
evans(Gs1,kpure(Gs1));
slope=%pi/2.30;
x=-5:0.01:0;
OS_line1=slope.*x;
OS_line2=-1*slope.*x;
plot(x,OS_line1,'r');
plot(x,OS_line2,'r');
scf(4);
slope=%pi/2.30;
x=-5:0.01:0;
OS_line1=slope.*x;
OS_line2=-1*slope.*x;
plot(x,OS_line1,'r');
plot(x,OS_line2,'r');
evans(Gs2,kpure(Gs2));
```