ME C134 / EE C128 Fall 2017 Prelab 4

Table of Contents

4.1 Position Controller Design

Amplitude will not affect ξ and ω_n, so it will not affect t_r and %OS.

4.1.1 Plant model

$G(s) = \frac{1.8097}{1.13s^2+8.1s+0.9886}$ has poles $s_1 = -0.12$, $s_2 = -7.04$. Both poles are in LHP, so the system is stable.

$M_p \leq 8\%$ yields $\xi \geq 0.627$

$t_r \leq 0.2s$ yields:

$\omega_n \geq 9.6$ at least;

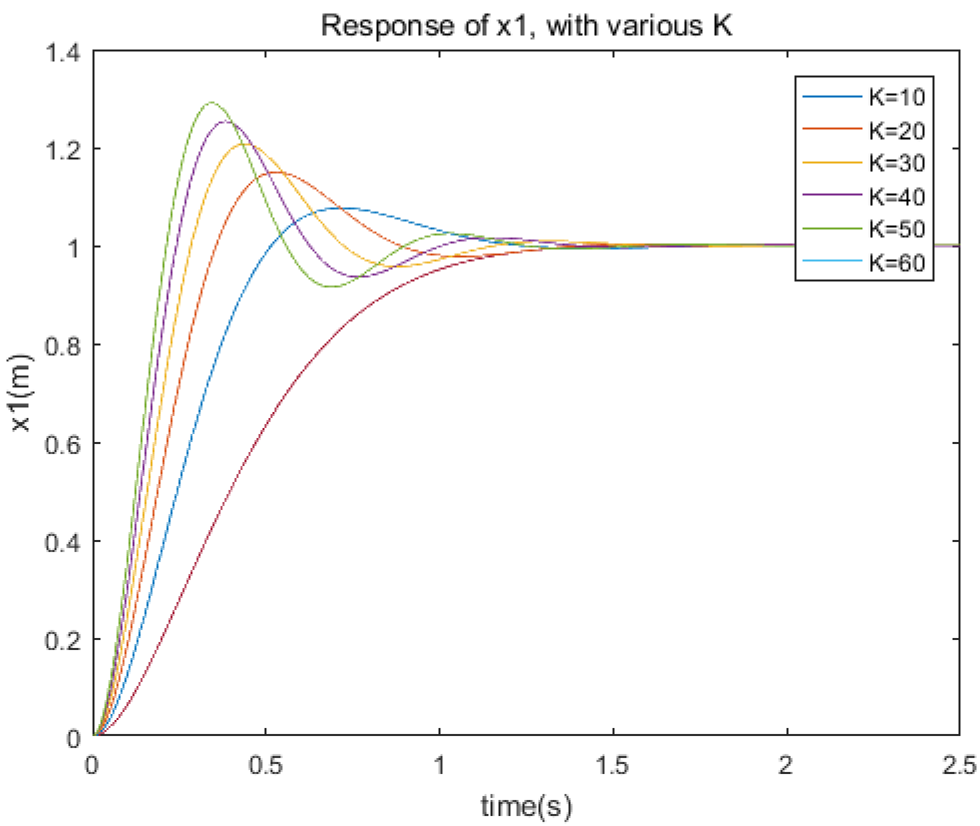
$\omega_n \geq 16.91$ will do for all ξ

4.1.2 Proportional Controller (a)

As K increases, the rise time will be shorter while the overshoot will become larger. Only with a P controller the desired performance cannot be achieved due to the "hard" tradeoff between rise time and overshoot within this kind of controller.

```
mc=0.94;
r=6.36e-3;
Rm=2.6;
Kt=7.67e-3;
Km=7.67e-3;
Kg=3.71;
Jm=3.9e-7;
t=10;
```

```
figure(1)
for i = 1:6
    K = i*10;
    sim('sim1.slx');
    plot(x1.time,x1.data)
    hold on
end
title('Response of x1, with various K')
xlim([0,2.5])
xlabel('time(s)')
ylabel('x1(m)')
legend('K=10','K=20','K=30','K=40','K=50','K=60')
```



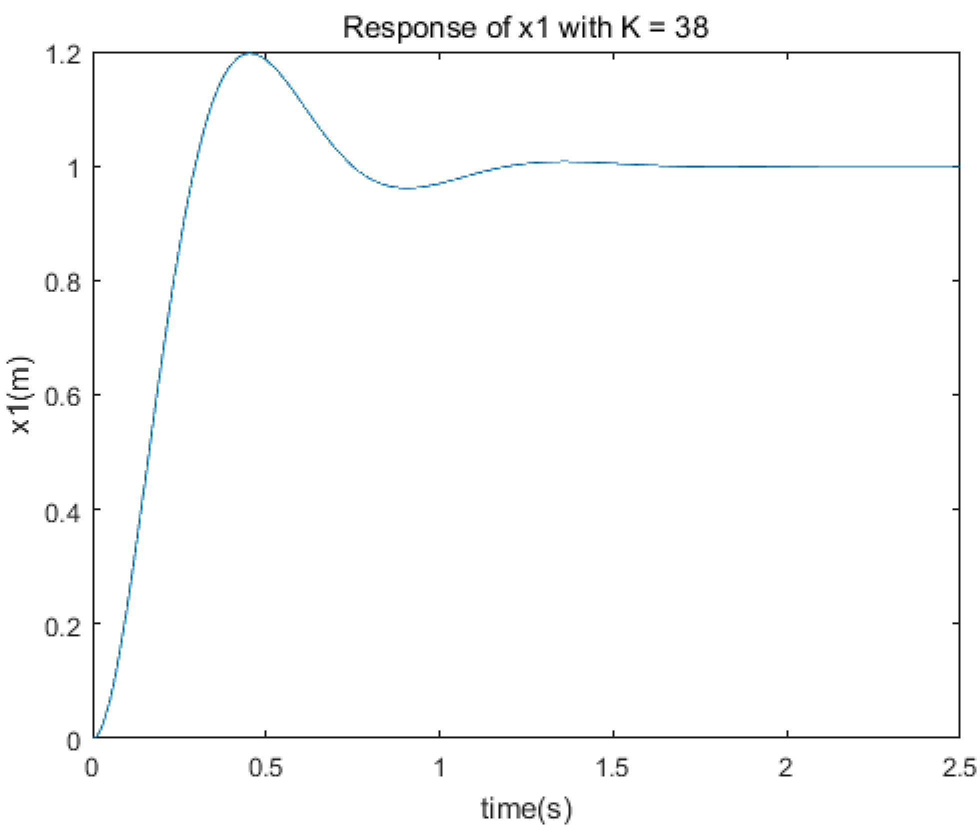
4.1.2 Proportional Controller (b)

The smallest integer value of K is 38 for which the rise time performance specification is met. However the overshoot percentage is 19.7% under this circumstance. Therefore the overshoot specification has not been met.

```
K = 38;
sim('sim1');
x1data = x1.data;
x1time = x1.time;
x1final = x1data(end);
x1_01 = x1time(find(x1data>=0.1*x1final,1));
x1_09 = x1time(find(x1data>=0.9*x1final,1));
```

```
Tr = x1_09-x1_01
overshoot_percentage = (norm(x1data,inf)-x1data(end))/x1data(end)*100
figure(2)
plot(x1)
xlim([0,2.5])
title('Response of x1 with K = 38')
xlabel('time(s)')
ylabel('x1(m)')
```

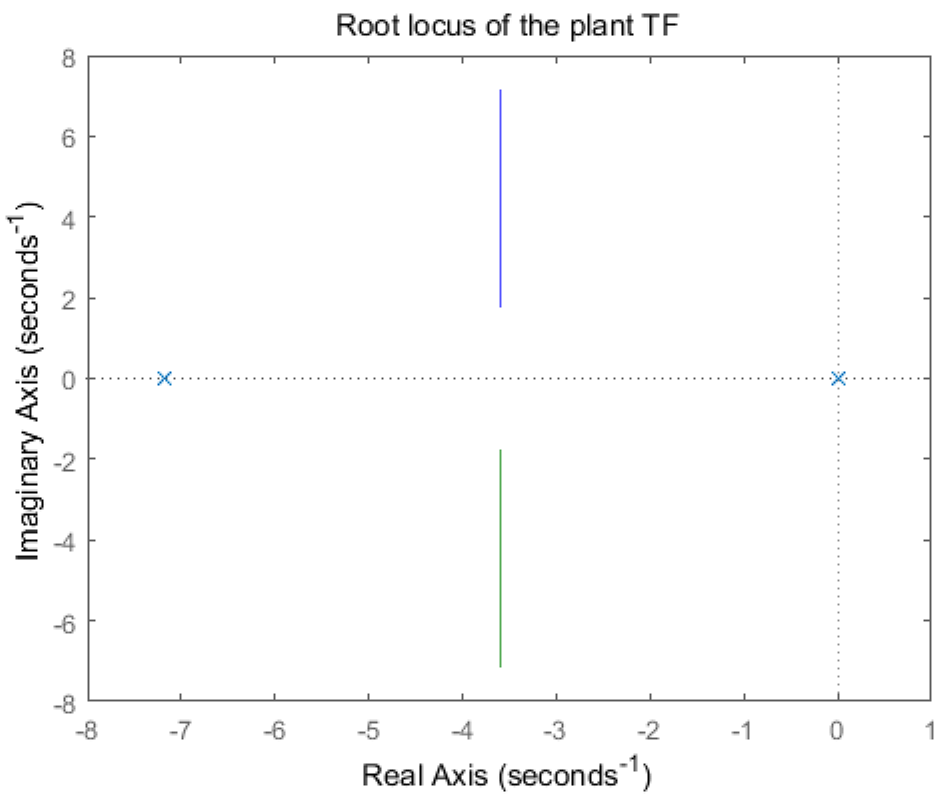
```
Tr =

    0.2000


overshoot_percentage =

   19.6721
```

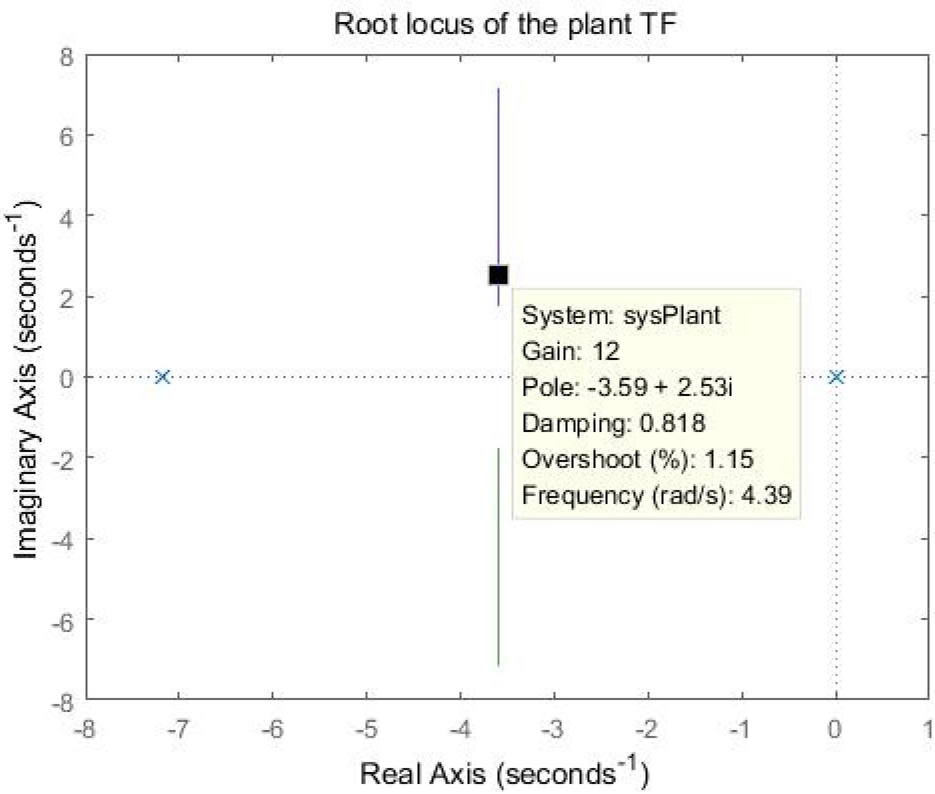


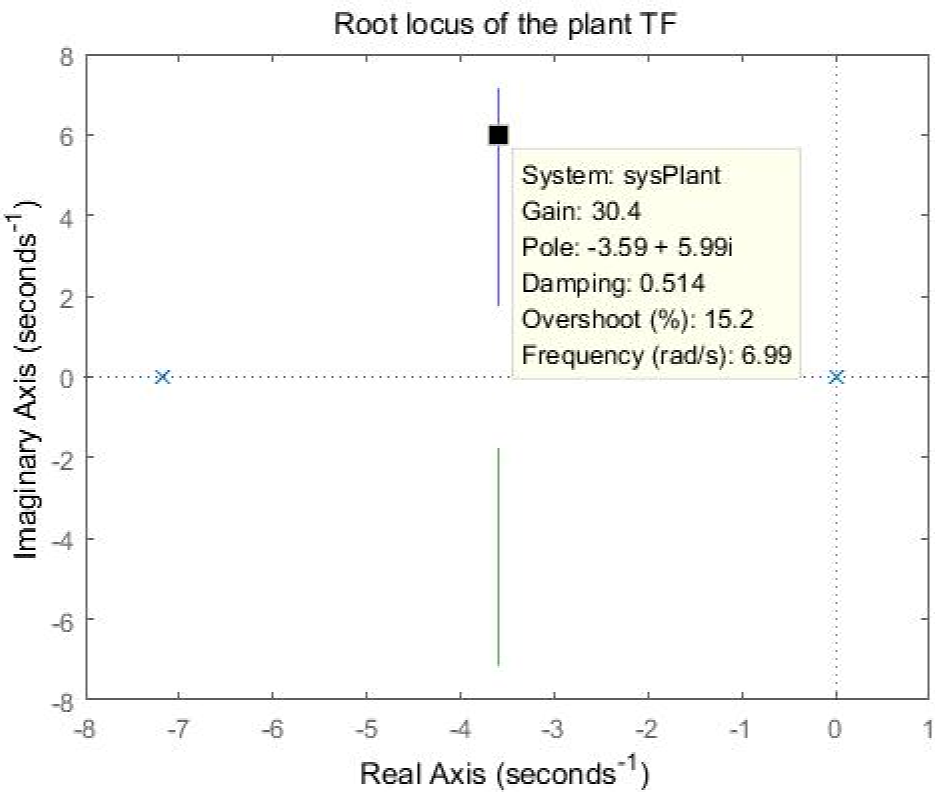
4.1.2 Proportional Controller (c)

For complex poles of the closed loop transfer function, as K increases the radial distance and angle θ both get larger. ξ gets smaller with the increase of θ while ω_n gets larger with the increase of the radial distance. These are shown in the following figures with different gains.

```
figure(3)
sysPlant = tf([r*Kt*Kg],[mc*r^2*Rm+Rm*Kg^2*Jm Kt*Km*Kg^2 0]);
rlocus(sysPlant,[10:40])
title('Root locus of the plant TF')
```




Root locus of the plant TF
Imaginary Axis (seconds^{-1})
Real Axis (seconds^{-1})
8
6
4
2
0
-2
-4
-6
-8
-8
-7
-6
-5
-4
-3
-2
-1
0
1
System: sysPlant
Gain: 12
Pole: -3.59 + 2.53i
Damping: 0.818
Overshoot (%): 1.15
Frequency (rad/s): 4.39



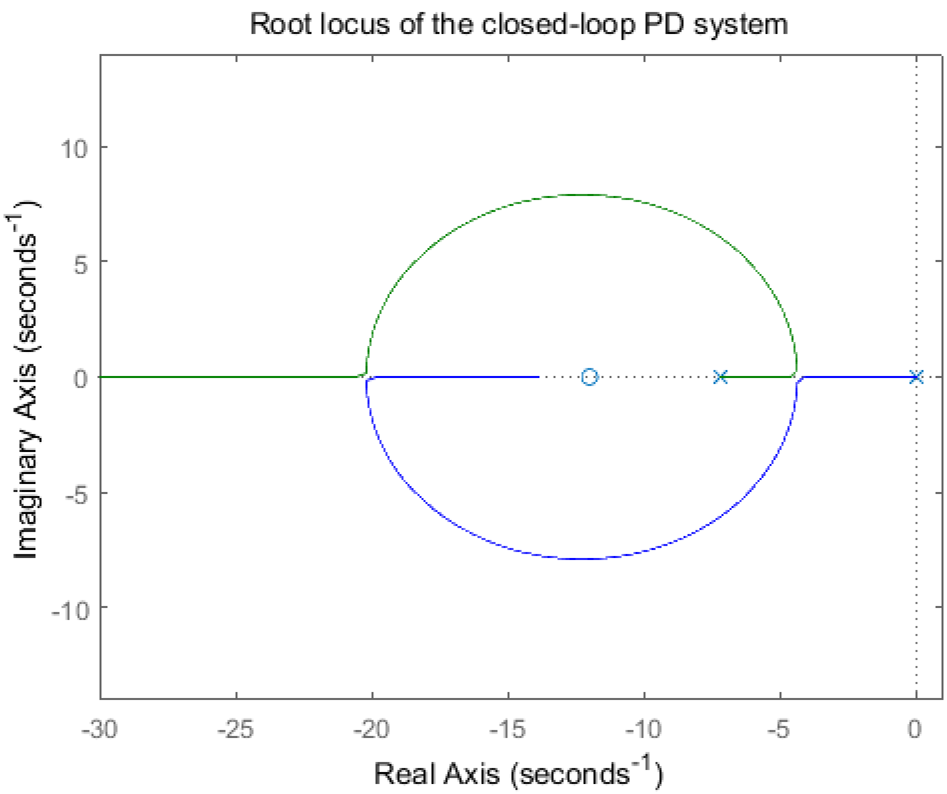
4.1.3 PD Controller (a)

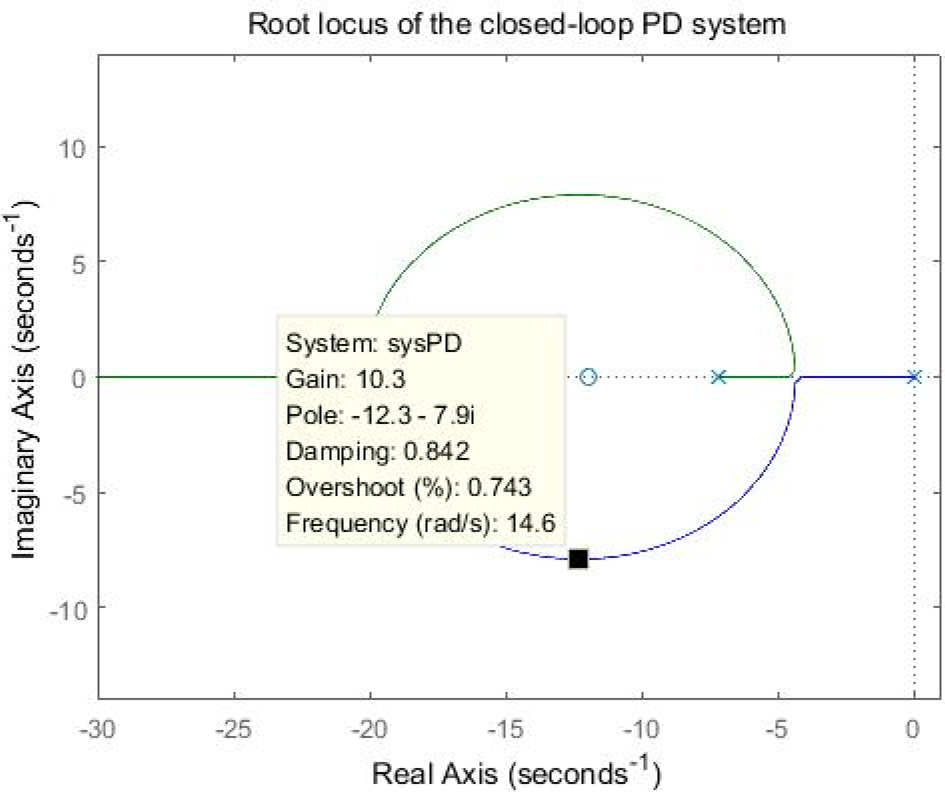
Reason why we should not introduce a pure zero: Because pure zero implies that either $K_P \to 0$ or $K_D \to \inf$, which yield numerical problems and are disastrous for practical implementation.

4.1.3 PD Controller (b)

The root locus plot of the closed-loop system and one example of a suitable gain equal to 10.3 that satisfies the performance specifications.

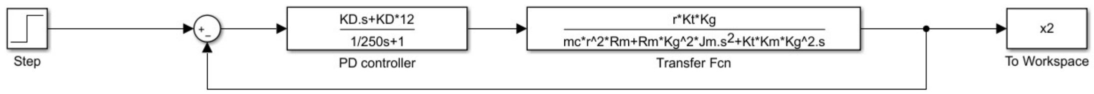
```
figure(4)
num = conv([1 12],[r*Kt*Kg]);
den = conv([1/250 1],[mc*r^2*Rm+Rm*Kg^2*Jm Kt*Km*Kg^2 0]);

sysPD = tf(num,den);
rlocus(sysPD,[0:0.01:30])
xlim([-30,1])
title('Root locus of the closed-loop PD system')
```


Root locus of the closed-loop PD system
Imaginary Axis (seconds^{-1})
Real Axis (seconds^{-1})
10
5
0
-5
-10
-30
-25
-20
-15
-10
-5
0



4.1.3 PD Controller (c)

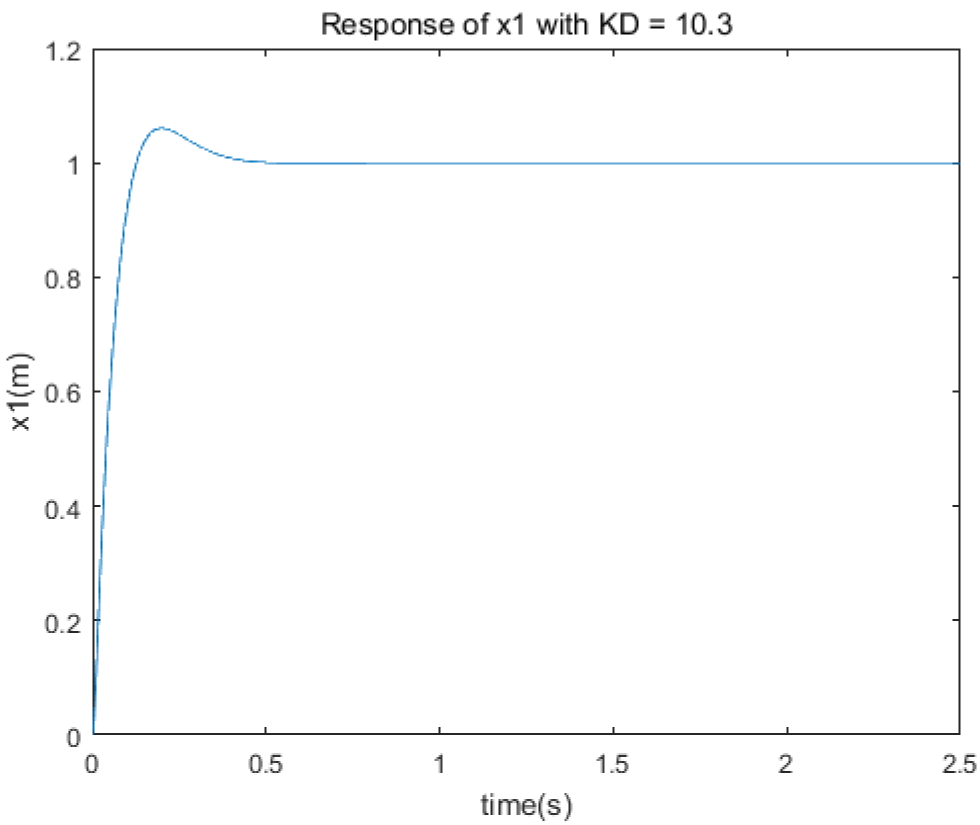
Following shows the simulation results of KD = 10.3 sugguested by root locus. The overshoot is 6% and satisfies the design specifications.




```
KD = 10.3;
sim('sim2');
x2data = x2.data;
x2time = x2.time;
x2final = x2data(end);
x2_01 = x2time(find(x2data>=0.1*x2final,1));
x2_09 = x2time(find(x2data>=0.9*x2final,1));
Tr = x2_09-x2_01
overshoot_percentage = (norm(x2data,inf)-x2data(end))/x2data(end)*100
figure(5)
plot(x2)
xlim([0,2.5])
title('Response of x1 with KD = 10.3')
xlabel('time(s)')
```

```
ylabel('x1(m)')
```

```
Tr =

    0.0860


overshoot_percentage =

    6.0336
```

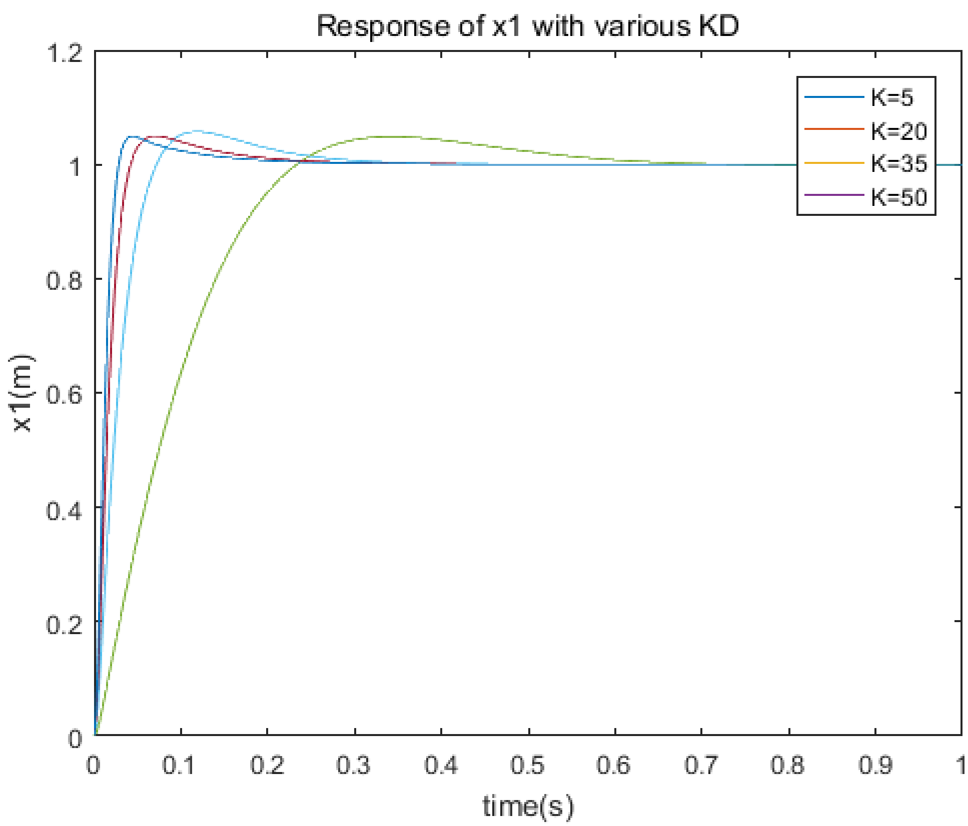


4.1.3 PD Controller (d)

The range where the specifications are met is $K_D \geq 4.0$

```
figure(6)
KD = 5;
sim('sim2');
plot(x2)
hold on
KD = 20;
sim('sim2');
plot(x2)
hold on
```

```
KD = 35;
sim('sim2');
plot(x2)
hold on
KD = 50;
sim('sim2');
plot(x2)
hold on
xlim([0,1])
title('Response of x1 with various KD')
xlabel('time(s)')
ylabel('x1(m)')
legend('K=5','K=20','K=35','K=50')
```



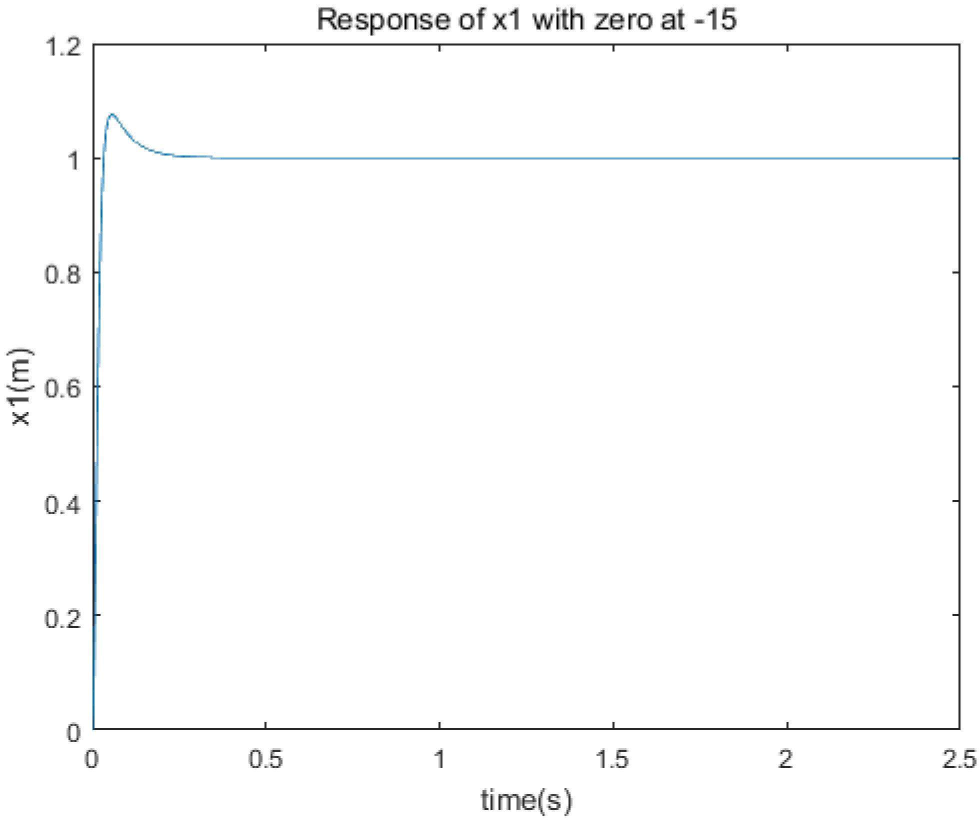
4.1.3 PD Controller (e)

When the zero is placed at -15, the range for KD which meets the specifications is $K_D \in [3.1, 4.2] \cup [33, 50]$

```
KD = 40;
sim('sim3');
x3data = x3.data;
x3time = x3.time;
x3final = x3data(end);
x3_01 = x3time(find(x3data>=0.1*x3final,1));
x3_09 = x3time(find(x3data>=0.9*x3final,1));
```

```
Tr = x3_09-x3_01
overshoot_percentage = (norm(x3data,inf)-x3data(end))/x3data(end)*100
figure(7)
plot(x3)
xlim([0,2.5])
title('Response of x1 with zero at -15')
xlabel('time(s)')
ylabel('x1(m)')
```

```
Tr =

    0.0200


overshoot_percentage =

    7.5774
```



Attribution

Rui Wang, Haimin Hu, Yujian An

Published with MATLAB® R2016b