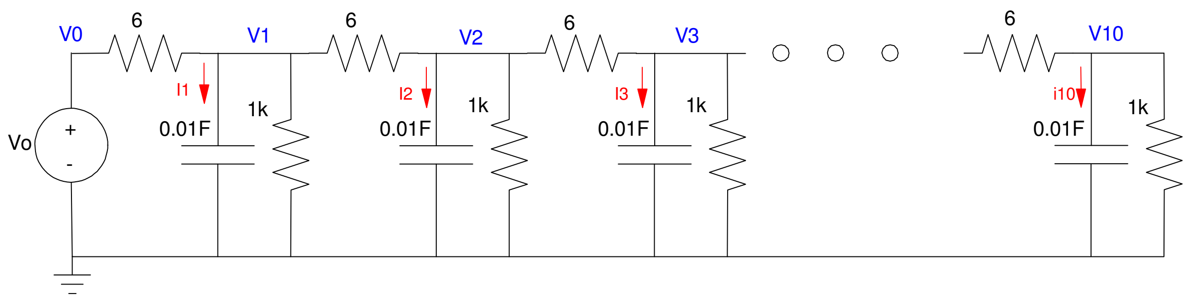
Homework #7: ECE 461/661

Gain, Lead, PID Compensation. Due Monday, October 18th

A 4th-order model for the following 10-stage RC filter is

$$G(s) = \left(\frac{170}{(s+0.47)(s+3.40)(s+9.00)(s+16.77)} \right)$$

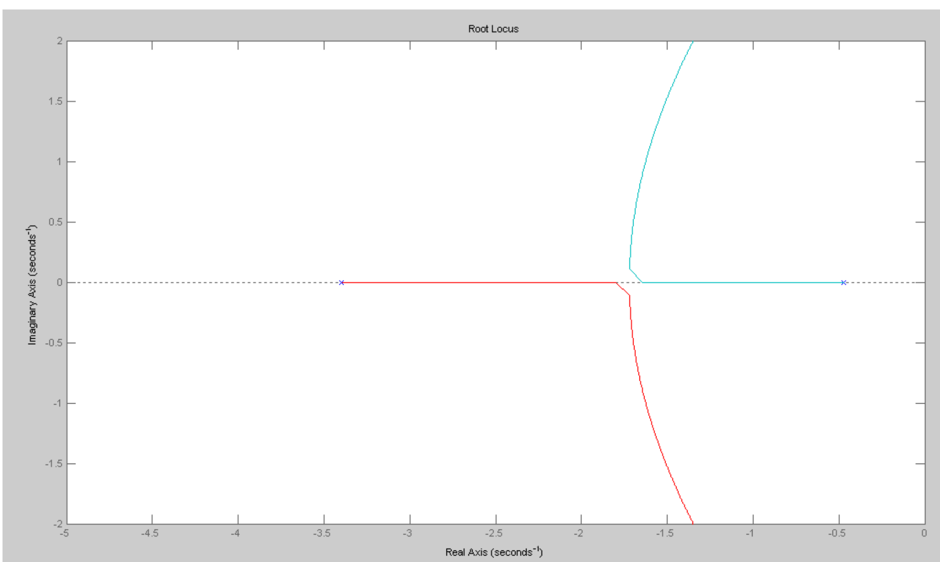


1) Design a gain compensator (K(s) = k) which results in

- The fastest system possible,
- With no overshoot for a step input (i.e. design for the breakaway point)

Step 1: Draw the root locus

```
>> G = zpk([],[-0.47,-3.40,-9.00,-16.77],170);
>> k = logspace(-2,2,1000)';
>> rlocus(G,k);
>> xlim([-5,0]);
>> ylim([-2,2]);
```



Step 2: Pick a point on the root locus. Since we want the fastest system with no overshoot, pick the breakaway point

s = -1.65 (approx)

Step 3: Find k at this point. At any point on the root locus

$$G \cdot k = -1$$

Pluging in numbers

$$\left(\frac{170}{(s+0.47)(s+3.40)(s+9.00)(s+16.77)}\right)_{s=-1.65} \cdot k = -1$$

In Matlab:

```
>> evalfr(G,-1.65)
   -0.7408

>> k = 1/abs(ans)

k =  1.3499
```

For this value of k, determine

a) The closed-loop dominant pole(s)

```
>> Gcl = minreal(G*k / (1 + G*k));
>> eig(Gcl)

   -1.6500
   -1.7929
   -9.5678
  -16.6293
```

This isn't *quite* the breakaway point since the other pole is at -1.7929 (not exactly equal to 1.65) but close

b) The 2% settling time,

$$T_s = \frac{4}{1.65} = 2.42 \text{ seconds}$$

c) The error constant, Kp, and

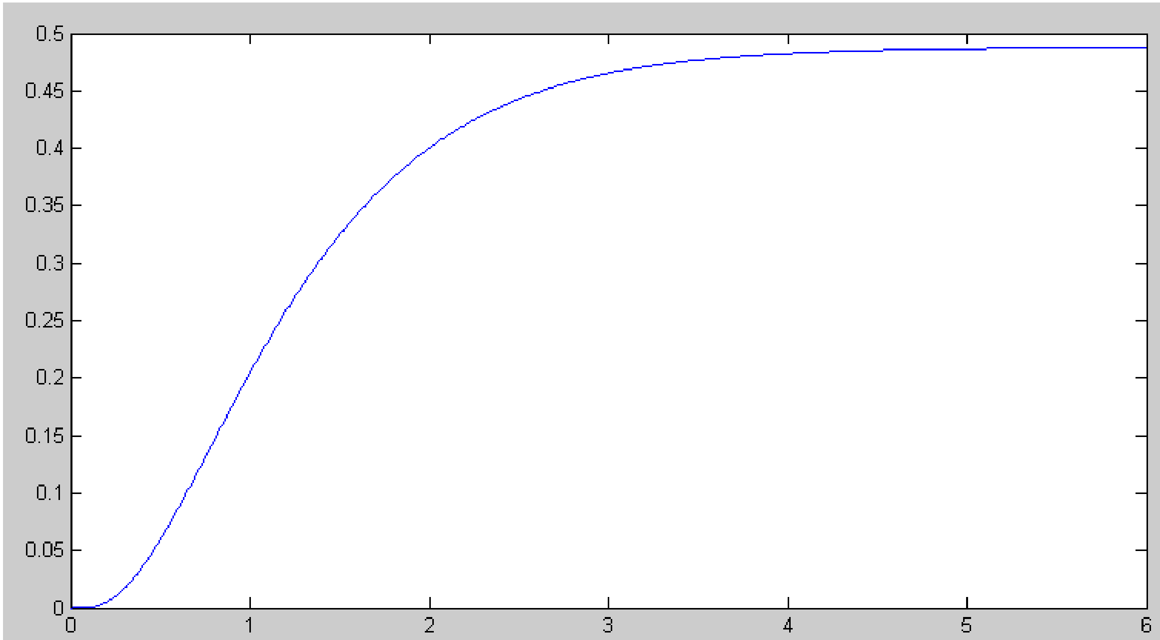
```
G = zpk([],[-0.47,-3.40,-9.00,-16.77],170);
>> k = 1 / abs(evalfr(G, -1.65))
k =    1.3499

>> Kp = evalfr(G,0) * k
Kp =    0.9515
```

The steady-state error for a step input.

```
>> Estep = 1 / (Kp + 1)
Estep =    0.5124
```

Check your design in Matlab or Simulink or VisSim

```
>> Gcl = minreal(G*k / (1+G*k));
>> t = [0:0.01:6]';
>> y = step(Gcl,t);
>> plot(t,y);;
```


0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
1
2
3
4
5
6

2) Design a gain compensator (K(s) = k) which results in 20% overshoot for a step input.

Step 1: Determine the damping line that corresponds to 20% overshoot

$$OS = 0.2 = \exp\left(\frac{-\pi\zeta}{\sqrt{1-\zeta^2}}\right)$$

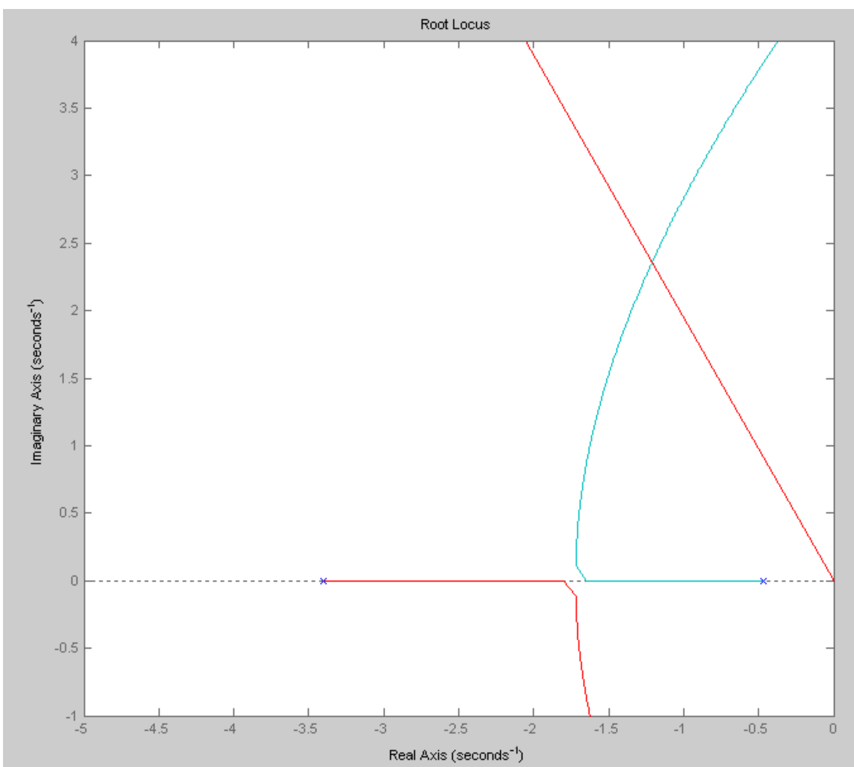
$$\zeta = 0.4560$$

$$\theta = \arccos(\zeta) = 62.61^0$$

Step 2: Sketch the root locus.

- Include the damping line
- Determine the point on the root locus that intersects the damping line

```
>> G = zpk([],[-0.47,-3.40,-9.00,-16.77],170);
>> k = logspace(-2,2,1000)';
>> rlocus(G,k);
>> hold on;
>> tan(acos(0.4560))
ans =     1.9517

>> plot([0,-3],[0,1.9517*3],'r')
>> xlim([-5,0]);
>> ylim([-1,4]);
```



From the graph (zooming in on the intersection)

```
>> s = -1.209 + j*2.36;
>> evalfr(G, -1.209 + j*2.36)
ans =  -0.1666 - 0.0000i
```

The complex part is almost zero. This is the correct point (or really close).

To find k, any point on the root locus satisfies

$$G \cdot k = -1$$

```
>> k = 1/abs(evalfr(G, -1.209 + j*2.36))

k =    6.0020
```

For this value of k, determine

a) The closed-loop dominant pole(s)

```
>> Gcl = minreal(G*k / (1+G*k));
>> eig(Gcl)

ans =

  -1.2092 + 2.3601i
  -1.2092 - 2.3601i
 -11.1897
 -16.0318
```

b)The 2% settling time,

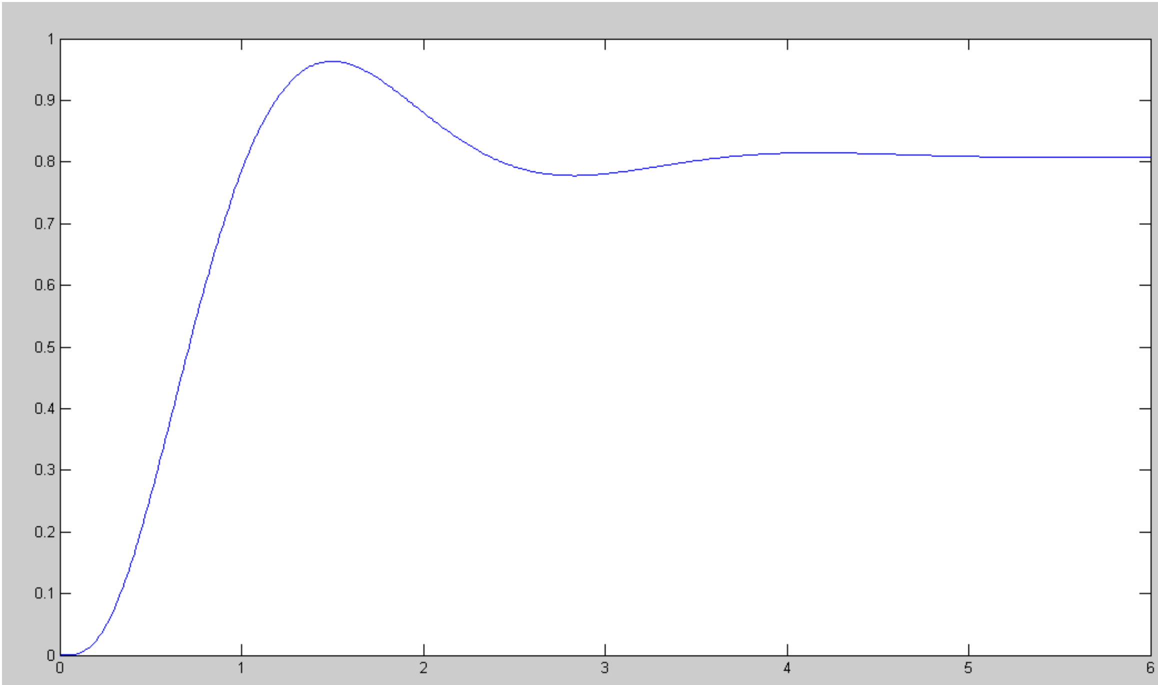
$$T_s = \frac{4}{1.2092} = 3.3080 \text{ seconds}$$

c)The error constant, Kp, and

```
>> Kp = evalfr(G,0) * k
Kp =    4.2305
```

d) The steady-state error for a step input.

```
>> Estep = 1 / (Kp + 1)
Estep =    0.1912

Check your design in Matlab or Simulink or VisSim>> t = [0:0.01:6]';
>> y = step(Gcl,t);
>> plot(t,y);
>> hold off
>> plot(t,y);
```



1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
1
2
3
4
5
6

3) Design a lead compensator, $K(s) = k\left(\frac{s+a}{s+10a}\right)$, which results in 20% overshoot for a step input.

$$G(s) = \left(\frac{170}{(s+0.47)(s+3.40)(s+9.00)(s+16.77)}\right)$$

Keep the pole at s = -0.47. This makes the system sort-of type-1. Cancel the next slowest pole

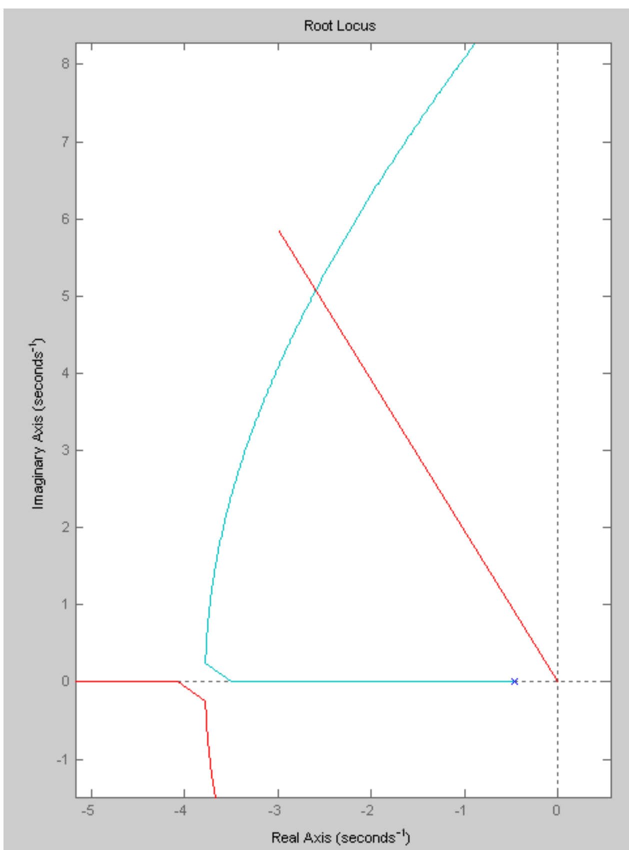
$$K(s) = k\left(\frac{s+3.40}{s+34}\right)$$

resulting in

$$GK = \left(\frac{170}{(s+0.47)(s+9.00)(s+16.77)(s+34)}\right)$$

Sketch the root locus along with the damping line

```
>> GK = zpk([],[-0.47,-34,-9.00,-16.77],170);
>> k = logspace(-2,2,1000)';
>> rlocus(GK,k);
>> k = logspace(-1,3,1000)';
>> rlocus(GK,k);
>> hold on
>> plot([0,-3],[0,1.9517*3],'r')
```



Zoom in to find the spot on the root locus

s = -2.5978 + j5.0701

Checking

```
>> s = -2.5978 + j*5.0701;
>> evalfr(GK,s)

ans =  -0.0079 - 0.0000i
```

The complex part is almost zero, so this point is on the root locus (or really close)

```
>> k = 1/abs(ans)

k =  126.4665
```

For this K(s), determine

a) The closed-loop dominant pole(s)

```
>> GKcl = minreal( (GK*k) / (1 + GK*k) );
>> eig(GKcl)

  -2.5978 + 5.0701i
 -22.9710
 -32.0734
  -2.5978 - 5.0701i
```

b) The 2% settling time,

$$T_s = \frac{4}{2.5978} = 1.5398 \text{ seconds}$$

c) The error constant, Kp, and

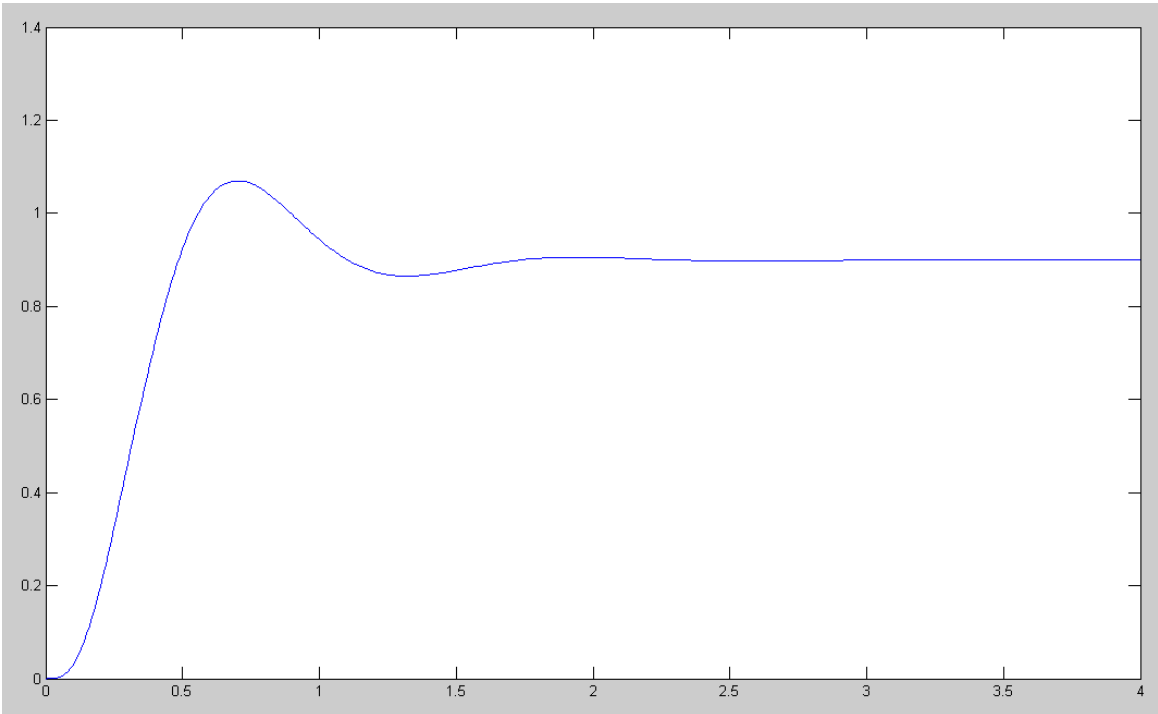
```
>> Kp = evalfr(GK,0) * k

Kp =    8.9140
```

d) The steady-state error for a step input.

```
>> Estep = 1 / (Kp + 1)

Estep =    0.1009
```

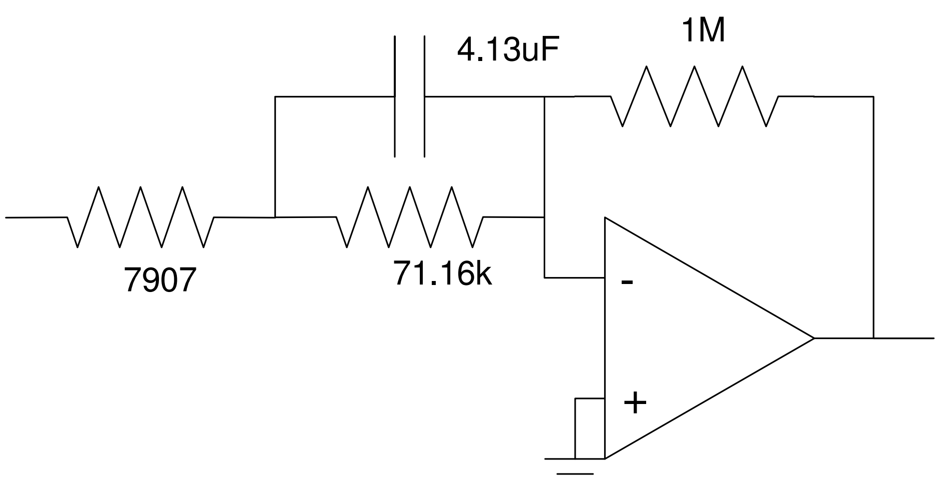
e) Check your design in Matlab or Simulink or VisSim

```
>> GKcl = minreal( (GK*k) / (1 + GK*k) );
>> t = [0:0.01:4]';
>> y = step(GKcl,t);
>> hold off
>> plot(t,y);
```



f) Give an op-amp circuit to implement K(s)

$$K(s) = 126.46\left(\frac{s+3.4}{s+34}\right)$$



I Compensation

4) Design an I compensator, $K(s) = \frac{I}{s}$, which results in 20% overshoot for a step input.

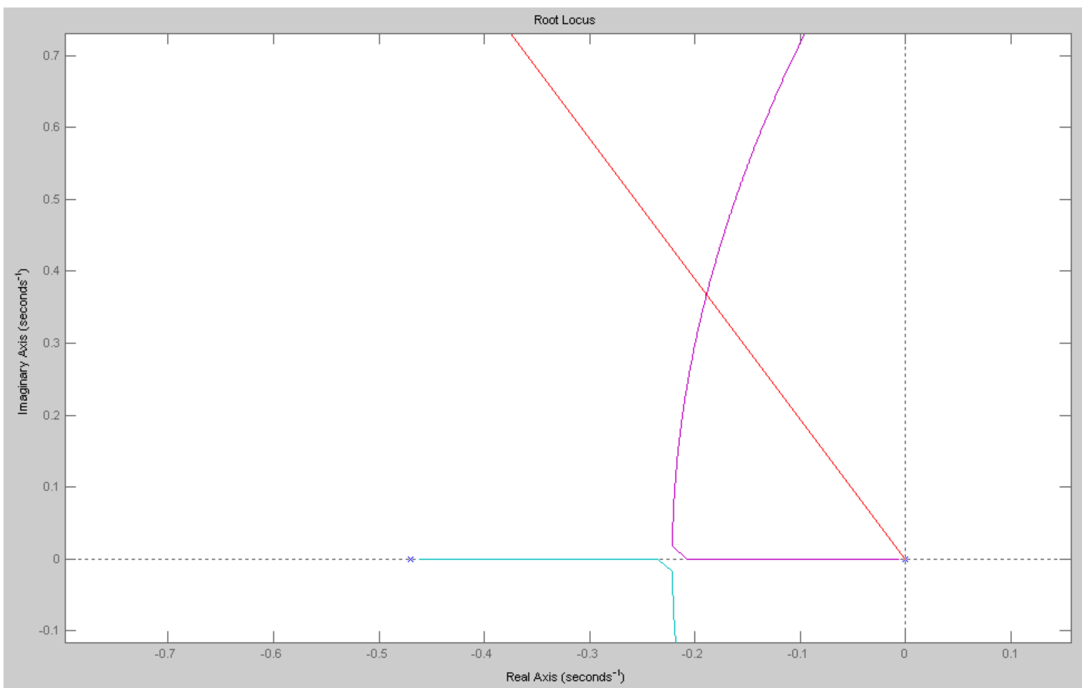
$$G(s) = \left(\frac{170}{(s+0.47)(s+3.40)(s+9.00)(s+16.77)}\right)$$

$$K(s) = \left(\frac{k}{s}\right)$$

$$GK = \left(\frac{170k}{s(s+0.47)(s+3.40)(s+9.00)(s+16.77)}\right)$$

Sketch the root locus

```
>> GK = zpk([],[0,-0.47,-3.4,-9.00,-16.77],170);
>> k = logspace(-2,2,1000)';
>> rlocus(GK,k);
>> hold on
>> plot([0,-3],[0,1.9517*3],'r')
```



Zoom in to find the crossing point

```
>> s = -0.1888 + j0.3685

>> evalfr(GK,s)

ans =  -1.8736 - 0.0001i

>> k = 1/abs(ans)

k =    0.5337
```

For this K(s), determine

The closed-loop dominant pole(s)

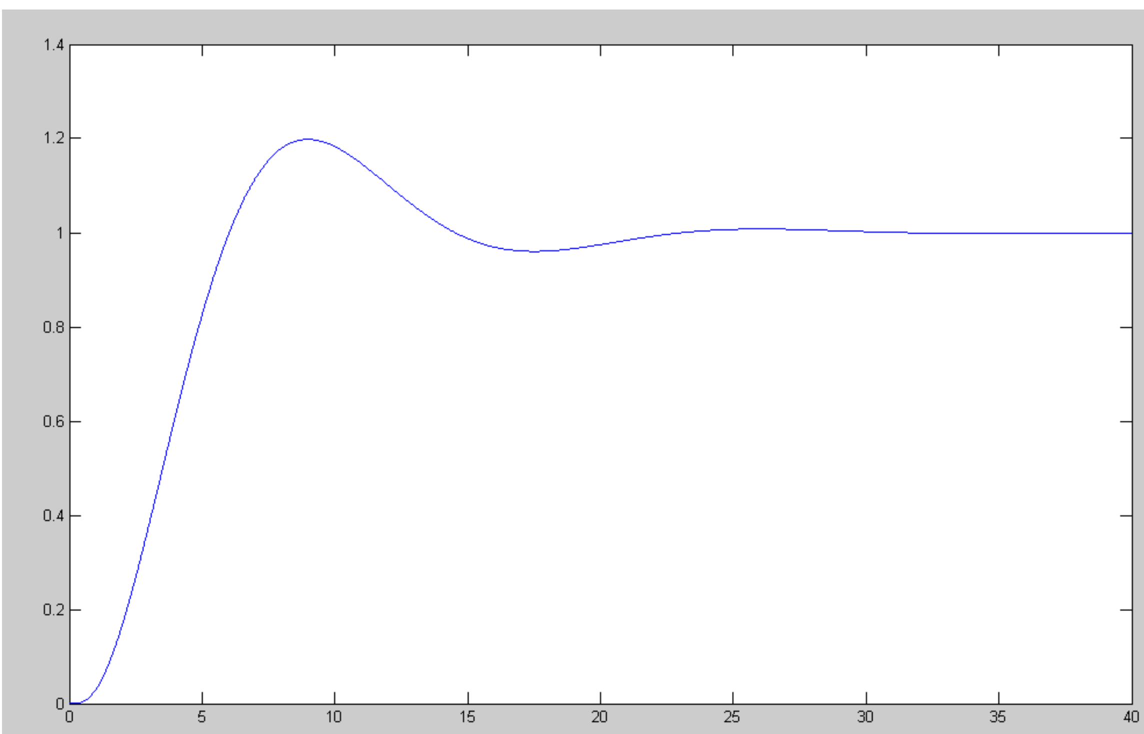
```
>> GKcl = minreal( (GK*k) / (1 + GK*k) );
>> eig(GKcl)

  -0.1888 + 0.3685i
  -0.1888 - 0.3685i
  -3.5165
  -8.9726
 -16.7732
```

The 2% settling time, the error constant, Kp, and the steady-state error for a step input.

```
>> Ts = 4/0.1888

Ts =   21.1864

>> Kp = evalfr(GK,0) * k

Kp =   Inf

>> Estep = 1 / (Kp + 1)

Estep =     0
```

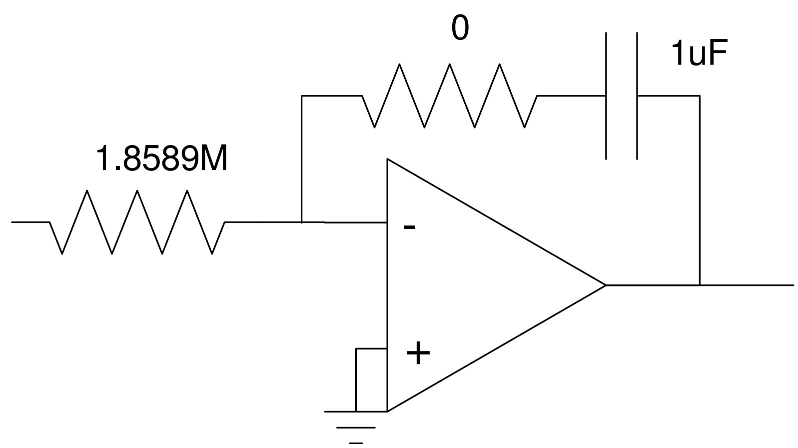
Check your design in Matlab or Simulink or VisSim

```
>> t = [0:0.01:40]';
>> y = step(GKcl,t);
>> hold off
>> plot(t,y);
```



Give an op-amp circuit to implement K(s)

$$K(s) = \left(\frac{0.5377}{s}\right)$$



PI Compensation

5) Design a PI compensator, $K(s) = k(\frac{s+a}{s})$, which results in 20% overshoot for a step input.

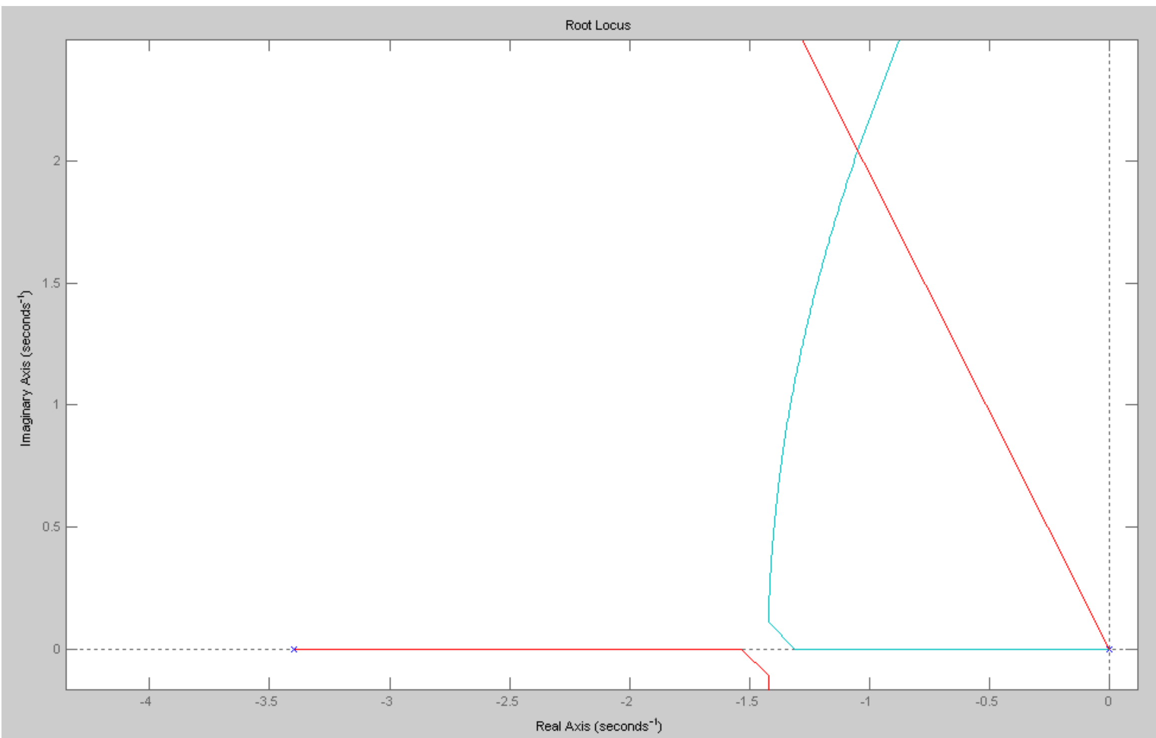
$$G(s) = \left(\frac{170}{(s+0.47)(s+3.40)(s+9.00)(s+16.77)} \right)$$

$$K(s) = k\left(\frac{s+0.47}{s} \right)$$

$$GK = \left(\frac{170k}{s(s+3.40)(s+9.00)(s+16.77)} \right)$$

Sketch the root locus along with the damping line

```
>> GK = zpk([],[0,-3.4,-9.00,-16.77],170);
>> k = logspace(-2,2,1000)';
>> rlocus(GK,k);
>> hold on
>> plot([0,-3],[0,1.9517*3],'r')
```



For this K(s), determine

- The closed-loop dominant pole(s)
- The 2% settling time,
- The error constant, Kp, and
- The steady-state error for a step input.

```
>> s = -1.0484 + j*2.0462;
>> evalfr(GK,s)

ans =  -0.1822 + 0.0000i
```

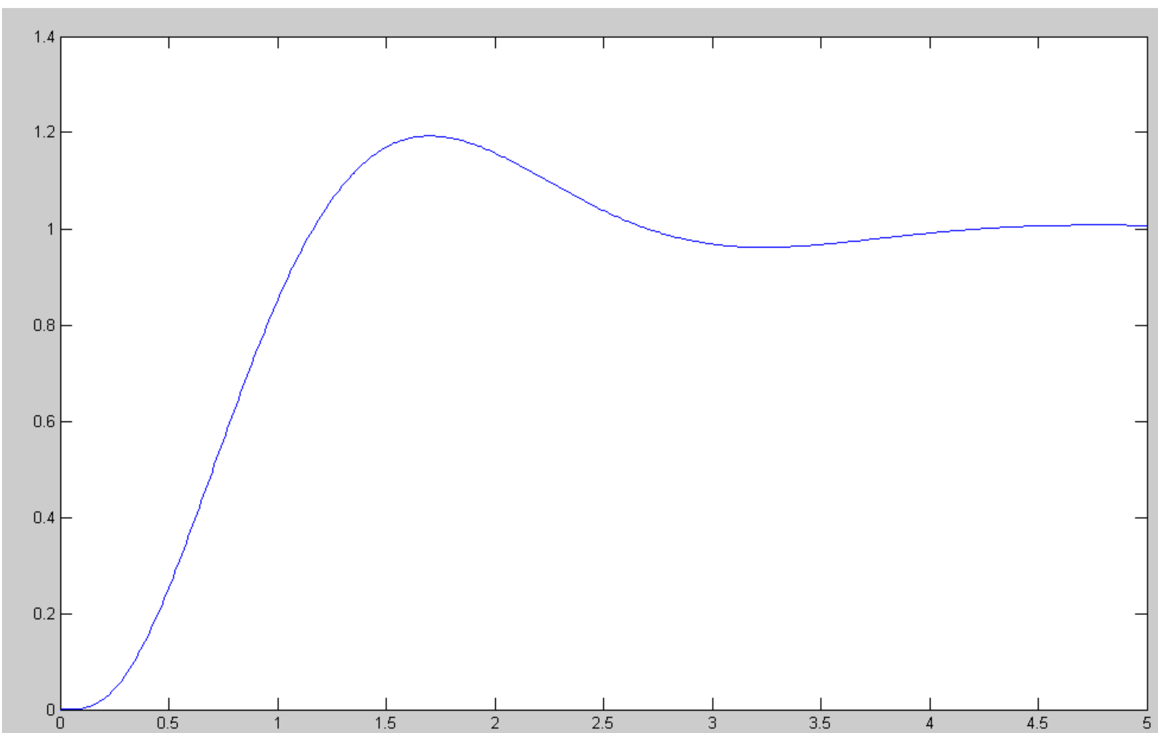
```
>> k = 1/abs(ans)

k =    5.4879

>> GKcl = minreal( (GK*k) / (1 + GK*k) );
>> eig(GKcl)

  -1.0483 + 2.0462i
  -1.0483 - 2.0462i
 -10.9400
 -16.1334

>> Ts = 4 / 1.0483

Ts =    3.8157
```

Type-1 system

- No error for a step input

Check your design in Matlab or Simulink or VisSim

```
>> hold off;
>> t = [0:0.01:5]';
>> y = step(GKcl,t);
>> plot([0,-3],[0,1.9517*3],'r')
>> plot(t,y);
```



Give an op-amp circuit to implement K(s)

$$K(s) = 5.4879\left(\frac{s+0.47}{s}\right)$$