Digital PID Control

ECE 461/661 Controls Systems

Jake Glower - Lecture #32

Please visit Bison Academy for corresponding
lecture notes, homework sets, and solutions

Digital PID Control

Similar to analog PID control except

- P: Proportional
- I: Integral
- D: Delay

P:

$$K(z) = P$$

PI:

$$K(z) = P + I\left(\frac{z}{z-1}\right) = k\left(\frac{z-a}{z-1}\right)$$

PID:

$$K(z) = P + I\left(\frac{z}{z-1}\right) + D\left(\frac{1}{z}\right) = k\left(\frac{(z-a)(z-b)}{z(z-1)}\right)$$

Example: $G(s) = \left(\frac{1000}{(s+2)(s+4)(s+6)(s+8)}\right)$

- Design P, PI, PID
- 20% Overshoot
- T = 50ms

Step 1: Convert G(s) to the z-Domain

s = -2	z = 0.9048
s = -4	z = 0.8187
s = -6	z = 0.7408
s = -8	z = 0.6703

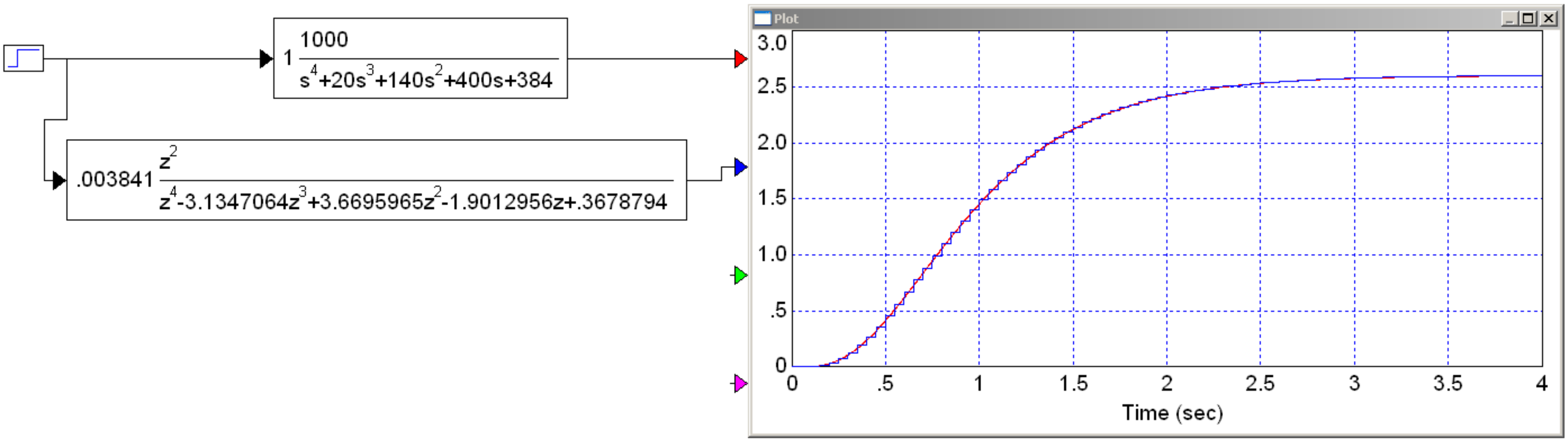
$$G(z) \approx \left(\frac{kz^2}{(z-0.9048)(z-0.8187)(z-0.7408)(z-0.6703)}\right)$$

Matching the DC gain:

$$G_s(s=0) = G_z(z=1) = 2.6042$$

$$G(z) \approx \left(\frac{0.003841 z^2}{(z-0.9048)(z-0.8187)(z-0.7408)(z-0.6703)} \right)$$

Add two zeros at z = 0 to match the delay



P Compensation: K(z) = P = k.

Method #1: Analyze the system in the z-pla

$$G(z) \approx \left(\frac{0.003841 z^2}{(z-0.9048)(z-0.8187)(z-0.7408)(z-0.6703)} \right)$$

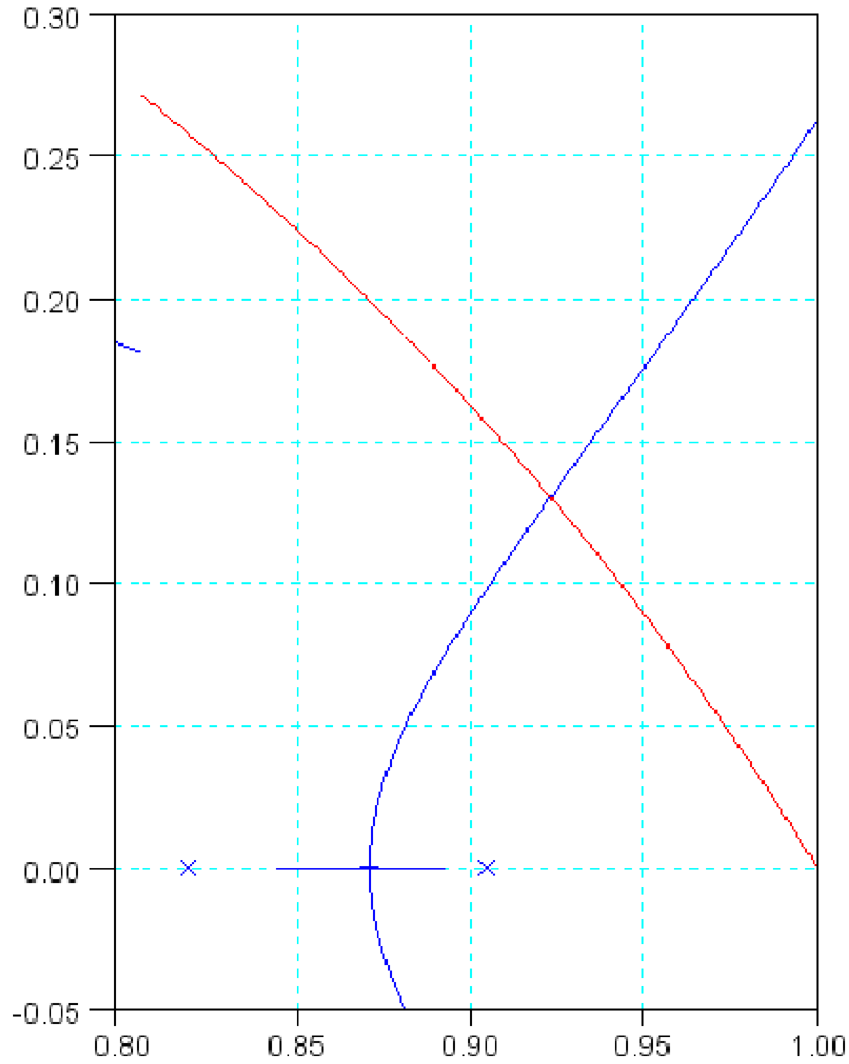
Sketch the root locus

- Find where the damping is 0.4559

```
G = zpk([0,0],[0.9048,0.8187,0.7408,0.6703],0.003841
k = logspace(-2,2,300)';
R = rlocus(G,k);

% add in the damping line

s = [0:0.01:10]' * (-1+j*2);
T = 0.05;
z = exp(s*T);
plot(real(R'), imag(R'), 'b', real(z),imag(z),'r');
```



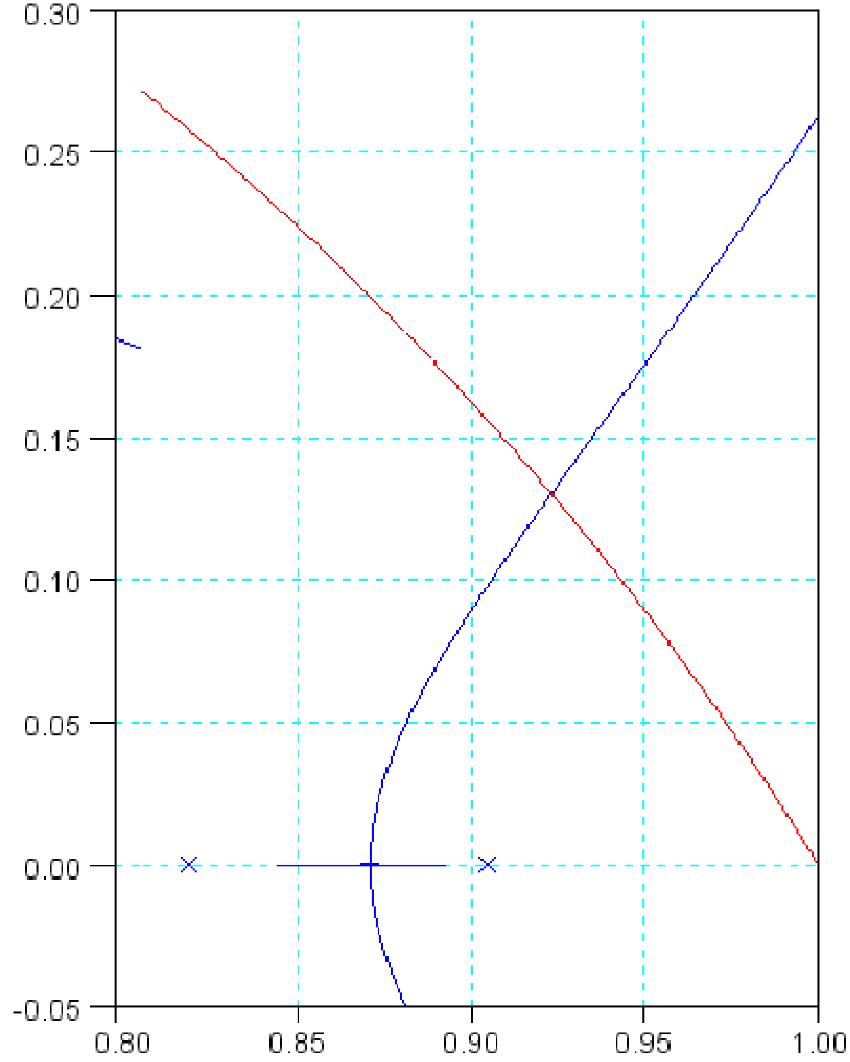
This gives

$$z = 0.9224 + j0.1289$$

and

$$G(z) = -2.4547k = -1$$

$$k = 0.4047$$



Method #2: Model the sample and hold with a 1/2 sample delay:

$$G(s) = \left(\frac{1000}{(s+2)(s+4)(s+6)(s+8)}\right)(e^{-0.025s})$$

Search along the damping ratio of 0.4559

$$s = \alpha\angle 117.1229^0$$

Iterate until the angles add up to 180 degrees

s = -1.3887 + j2.7111

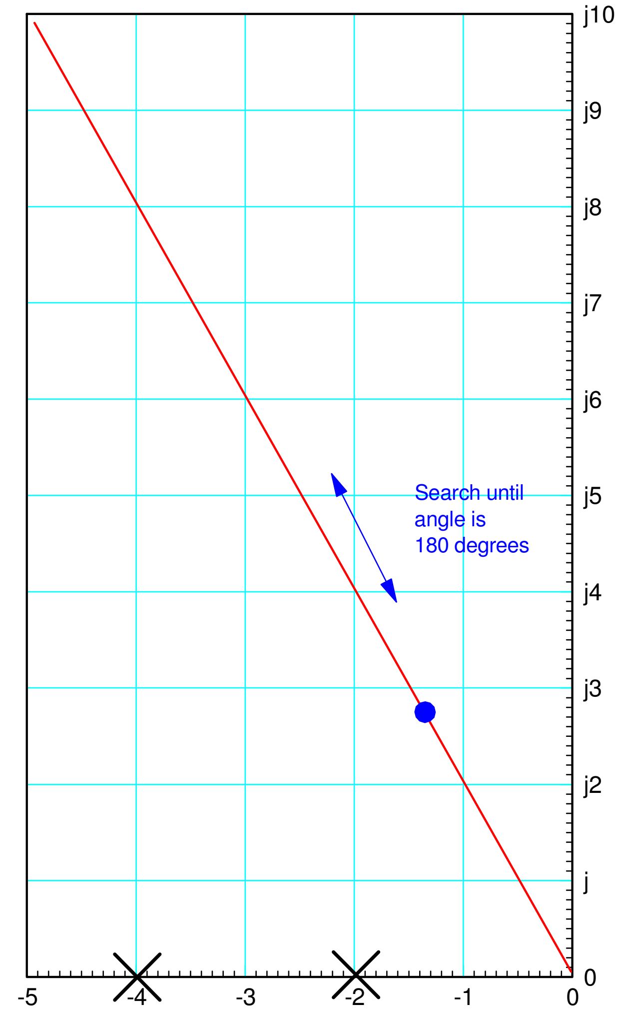
z = 0.9244 + j0.1261

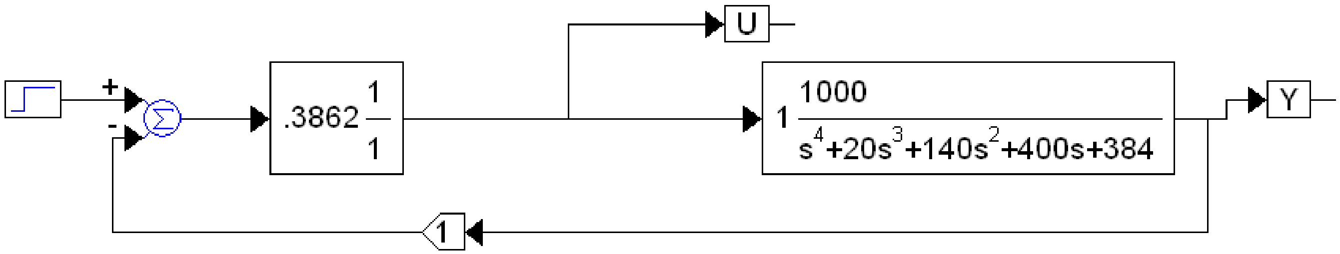
At any point on the root locus, GK = -1

G(s) = -2.5892

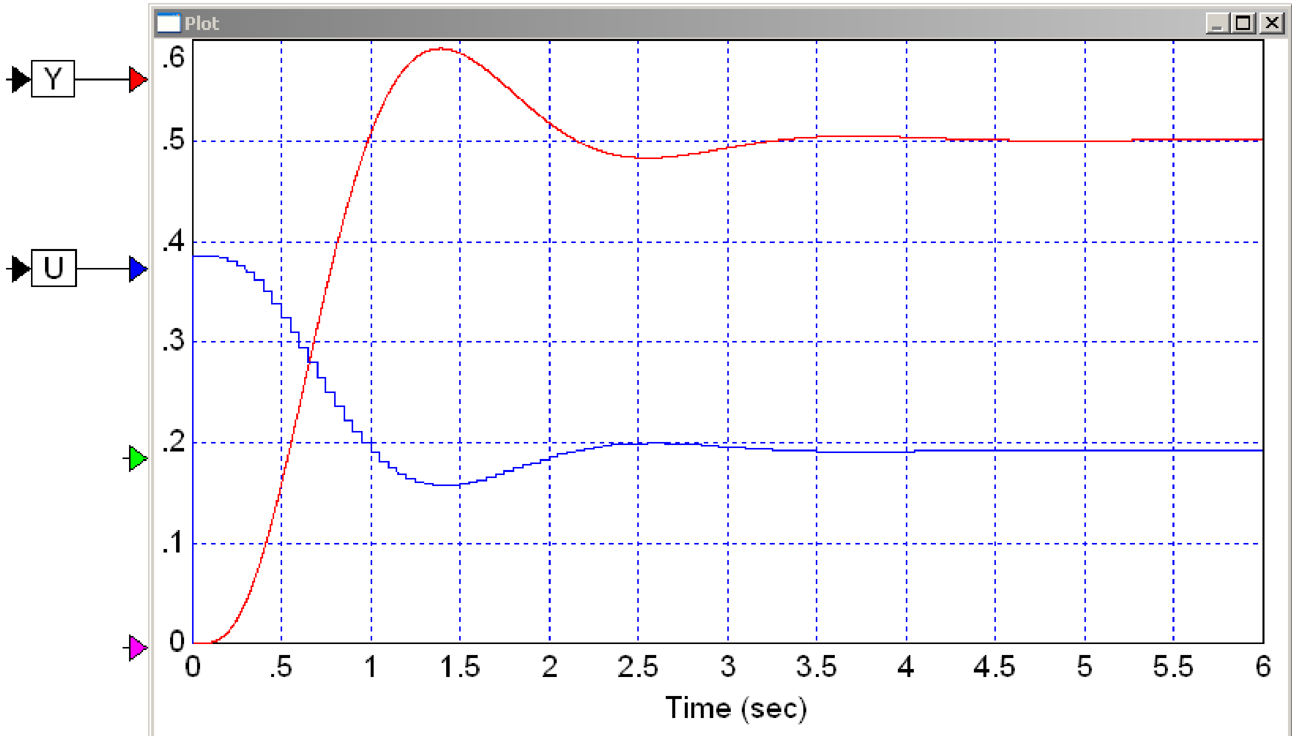
so

k = 0.3862





U
+
-
.3862 1/1
1 1000/(s^4+20s^3+140s^2+400s+384)
Y
1



Plot
Y
U
.6
.5
.4
.3
.2
.1
0
0 .5 1 1.5 2 2.5 3 3.5 4 4.5 5 5.5 6
Time (sec)

PI Compensation: $K(z) = k\left(\frac{z-a}{z-1}\right)$

- Add a zero at s = 0 (z = 1)

  Makes it a Type-1 system

Method #1: Design in the z-plane.

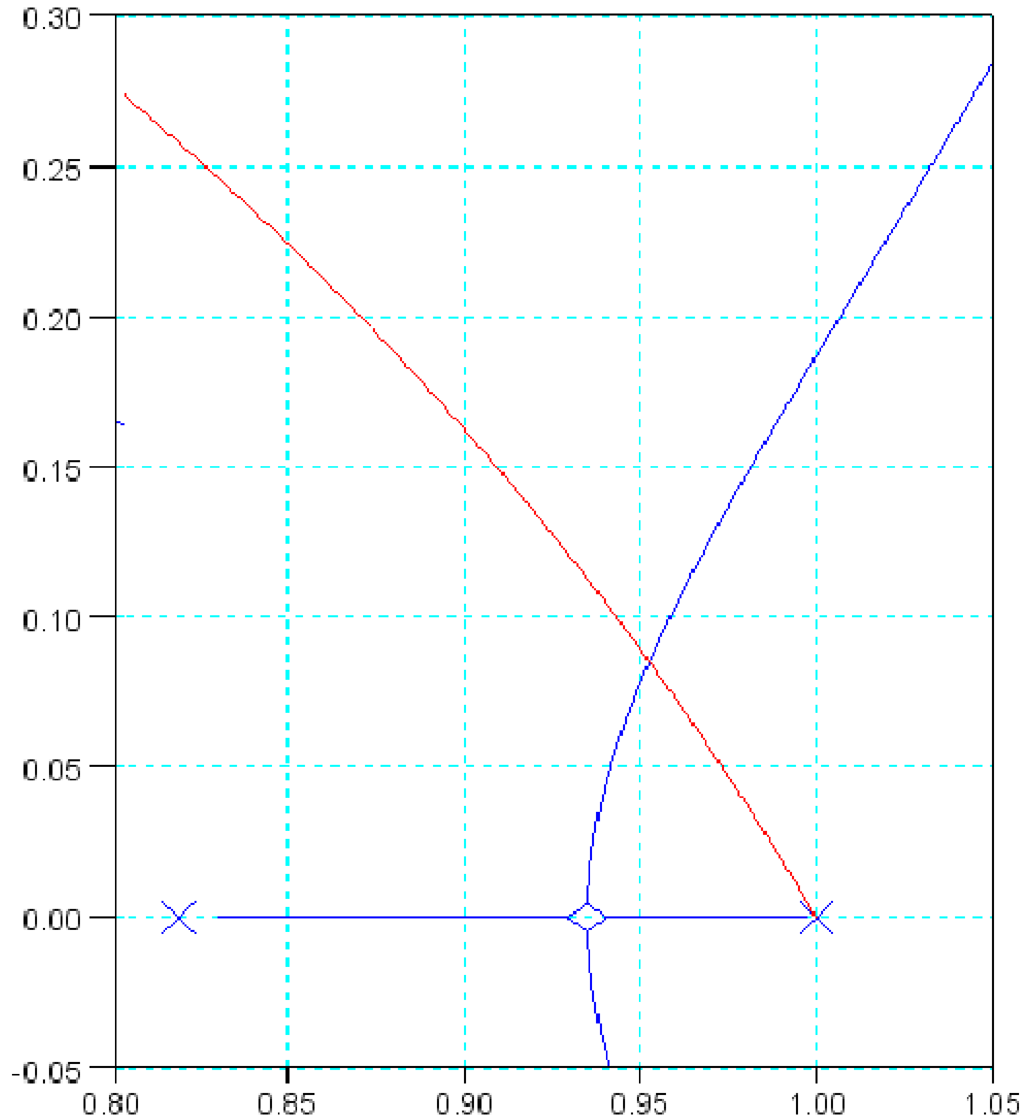
$$G(z) \approx \left(\frac{0.003841z^2}{(z-0.9048)(z-0.8187)(z-0.7408)(z-0.6703)}\right)$$

Cancel the slowest pole

$$K(z) = k\left(\frac{z-0.9048}{z-1}\right)$$

Sketch the resulting root locus:

```
G = zpk([0,0],[1,0.8187,0.7408,0.67
k = logspace(-2,2,1000)';
R = rlocus(G,k,0.4559)';
// damping lines from before
plot(real(R), imag(R), 'b', real(z),
```



Find z:

$$z = 0.9522 + j0.0840$$

This gives

$$G(z) = -3.4289$$

$$k = 0.2908$$

$$K(z) = 0.2908\left(\frac{z-0.9048}{z-1}\right)$$

Method #2: G(s) * Sample & Hold * K(z):

$$G(s) = \left(\frac{1000}{(s+2)(s+4)(s+6)(s+8)}\right) \cdot (e^{-0.025s}) \cdot k\left(\frac{z-0.9048}{z-1}\right)$$

Search in the s (and corresponding z) plane

s = 0.8734 + j1.7050

z = 0.9538 + j0.0815

At any point on the root locus, GK = -1

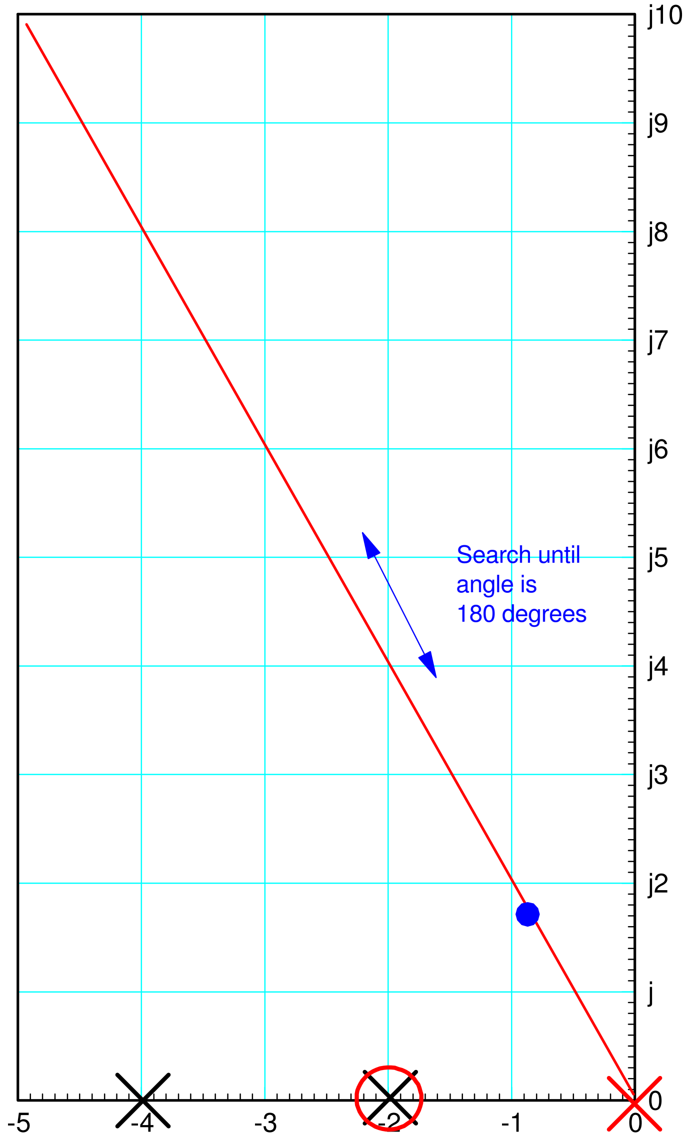
GK = -3.6004

k = 0.2777

so

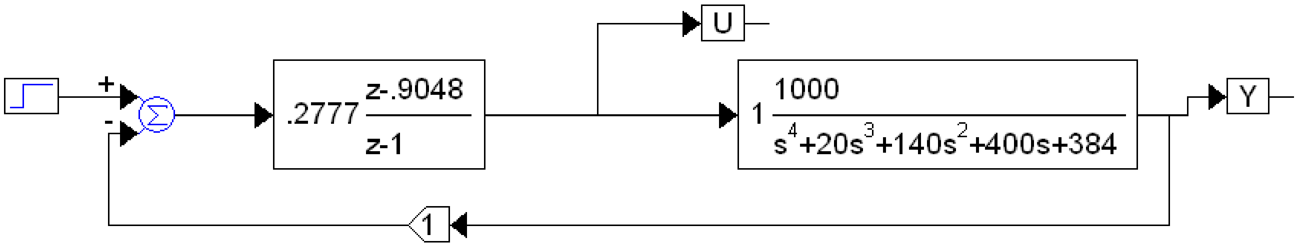
$$K(z) = 0.2777\left(\frac{z-0.9048}{z-1}\right)$$

Verify your design in VisSim.



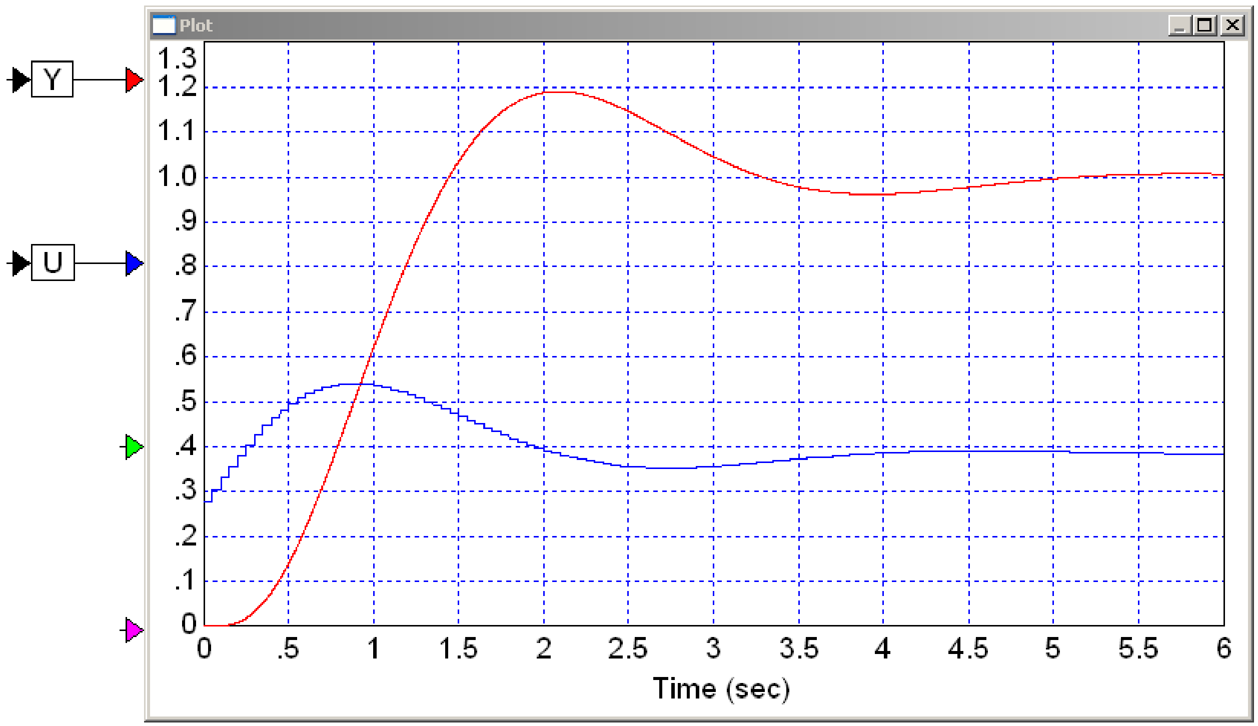
Note:

- Almost the same result
- Latter method is more accurate



I:

- Initial guess for U is zero
- Pole at s = -2 slows down the system



PI:

- Initial guess for U is 0.2777

PID Compensation: $K(z) = k\left(\frac{(z-a)(z-b)}{z(z-1)}\right)$

Method #1: Convert to the z-plane

$$G(z) \approx \left(\frac{0.003841z^2}{(z-0.9048)(z-0.8187)(z-0.7408)(z-0.6703)}\right)$$

Cancel two poles

$$K(z) = k\left(\frac{(z-0.9048)(s-0.8187)}{z(z-1)}\right)$$

Sketch the resulting root locus:

```
G = zpk([0,0],[1,0,0.7408,0.6703],0.003841);
k = logspace(-2,2,1000)';
R = zlocus(G,k,0.4559)';
// add damping line from before
plot(real(R), imag(R), real(z),imag(z));
```

Find z:

$$z = 0.9164 + j0.1373$$

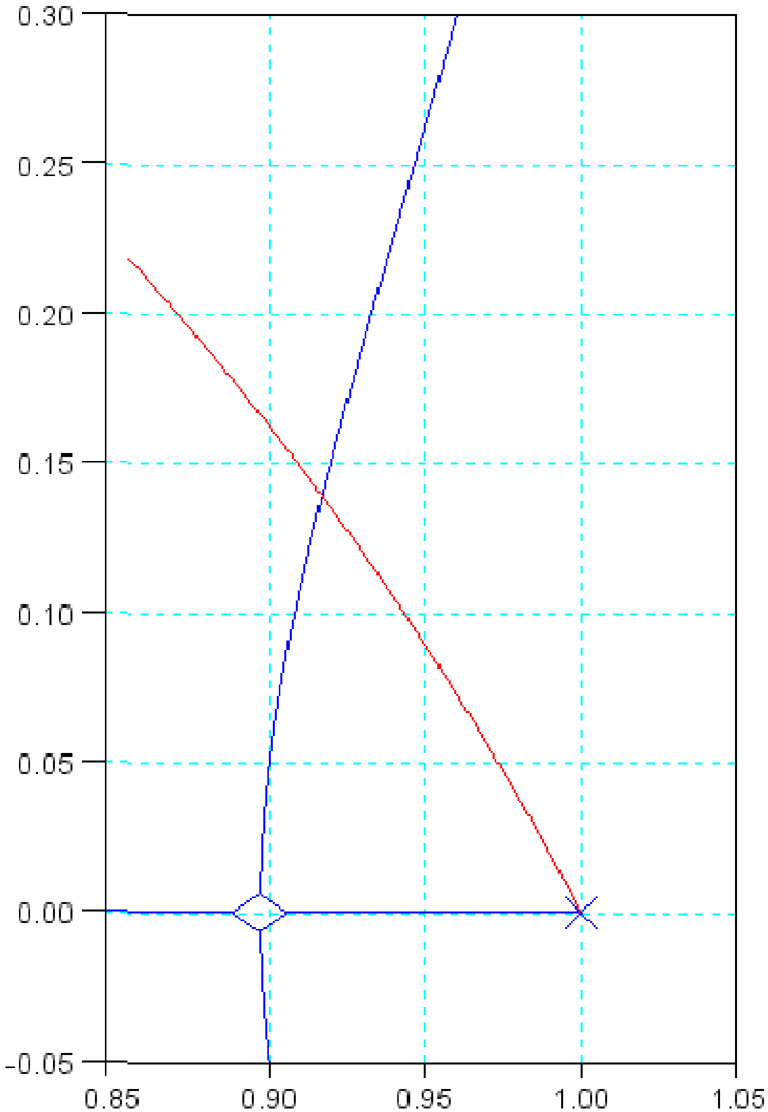
At this point

$$GK = -0.3524k = -1$$

$$k = 2.8381$$

and

$$K(z) = 2.8381\left(\frac{(z-0.9048)(s-0.8187)}{z(z-1)}\right)$$



Method #2: G(s) * Sample & Hold * K(z) is

$$G(s) = \left(\frac{1000}{(s+2)(s+4)(s+6)(s+8)}\right) \cdot (e^{-0.025s}) \cdot k\left(\frac{(z-0.9048)(s-0.8187)}{z(z-1)}\right)$$

Search along the damping line until angles add up to 180 degrees

s = -1.4422 + j2.8155

z = 0.9212 + j0.1305

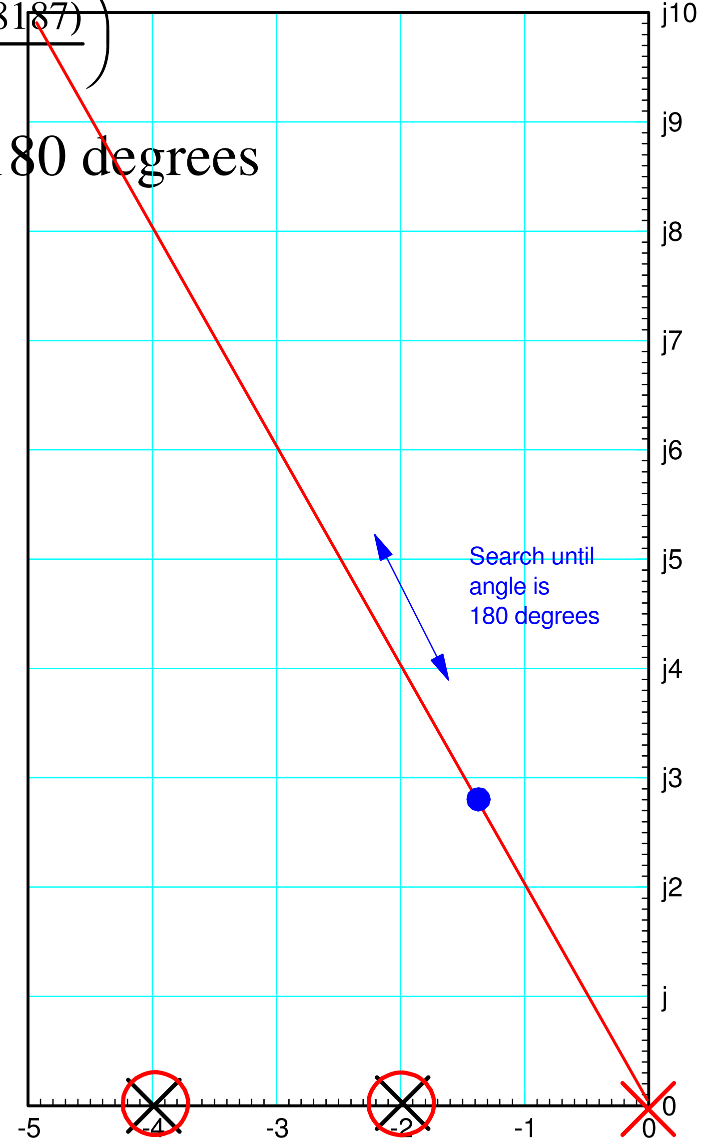
At this point, GK = -1

GK = -0.3824k = -1

k = 2.6153

so

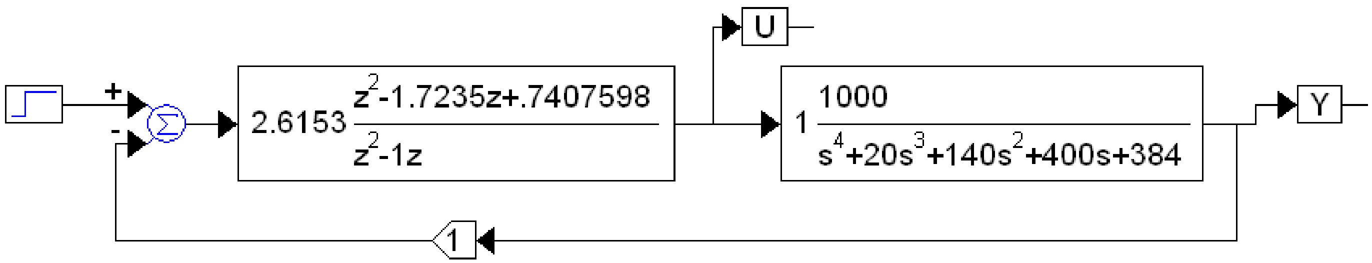
$$K(z) = 2.6153\left(\frac{(z-0.9048)(s-0.8187)}{z(z-1)}\right)$$

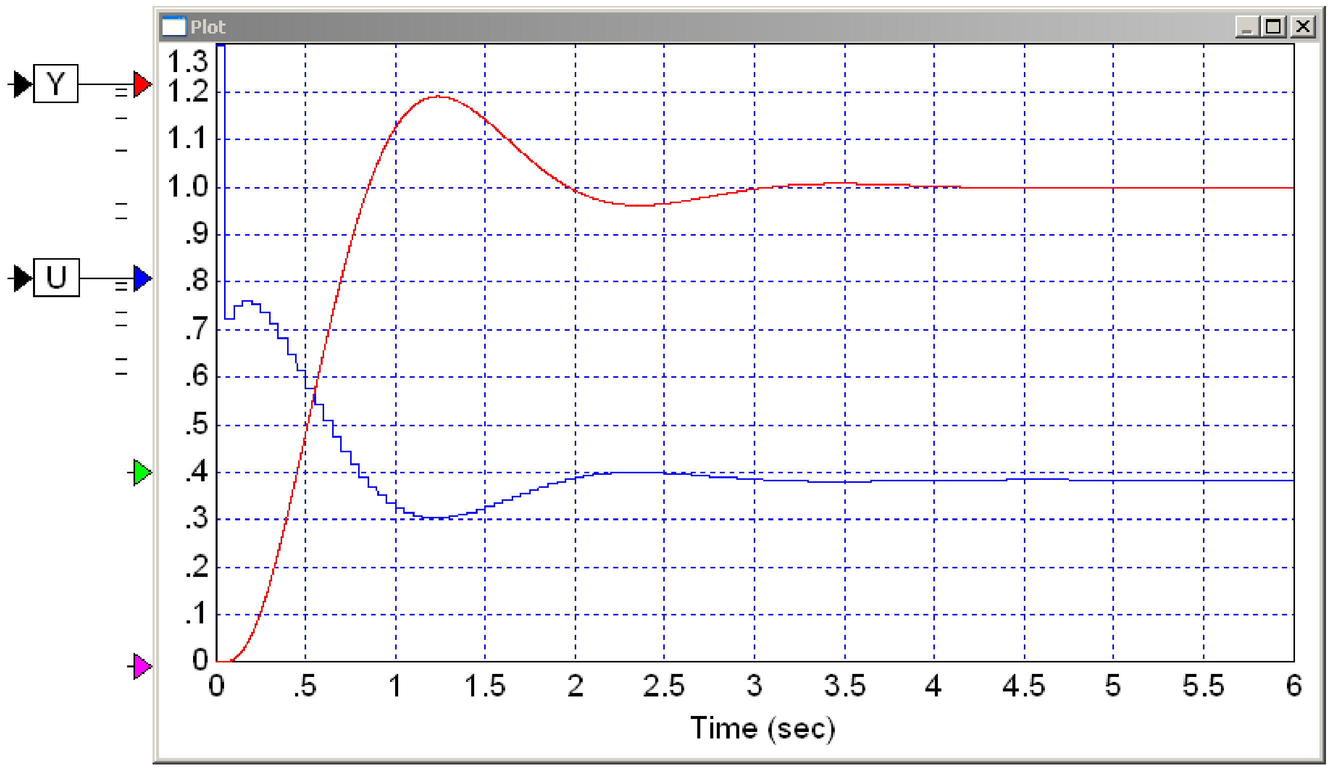


Note:

- Two zeros allow you to speed up the system

You get an impulse at k=1

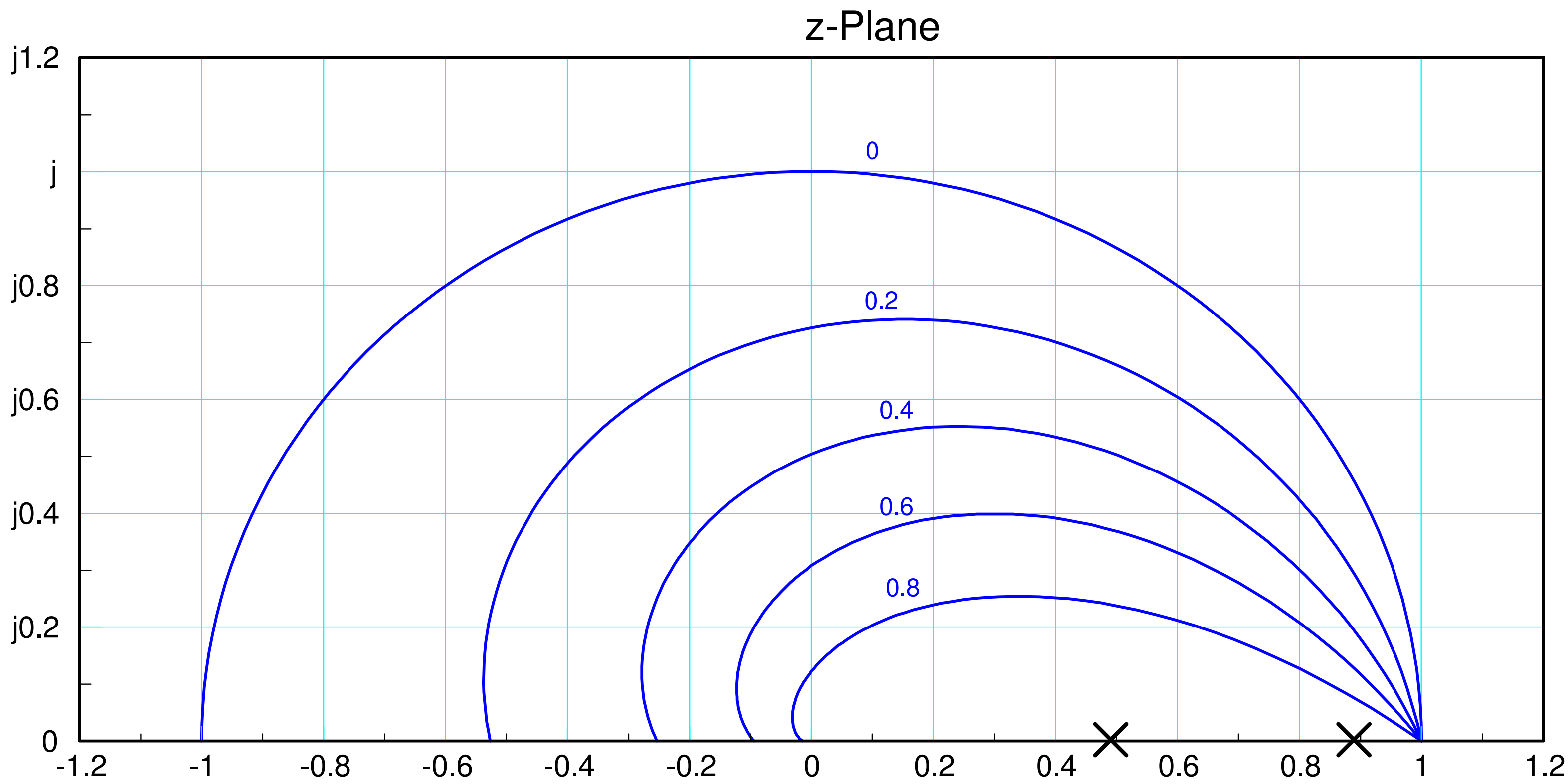




Handout: Design a PI compensator for

$$G(z) = \left(\frac{0.1}{(z-0.9)(z-0.5)}\right)$$

that results in a damping ratio of 0.4



Changing the Sampling Rate

The 2% settling time is 3 seconds

T = 50ms is too fast

- Gives 60 samples
- Results in an impulse for PID

  *Energy is width * height*

  Needs a height of 2.6153 (off graph) to provide enough energy

T = 200ms is more reasonable

- 15 samples in 3 seconds
- More width means less height for U at k=0

PID with T = 200ms

Plant * Sample & Hold * K(z)

$$\left(\frac{1000}{(s+2)(s+4)(s+6)(s+8)}\right)\cdot e^{-0.1s}\cdot k\left(\frac{(z-0.6703)(z-0.4493)}{z(z-1)}\right)$$

Note: Zeros move in the z-plane as $z = e^{sT}$

Search along the damping line:

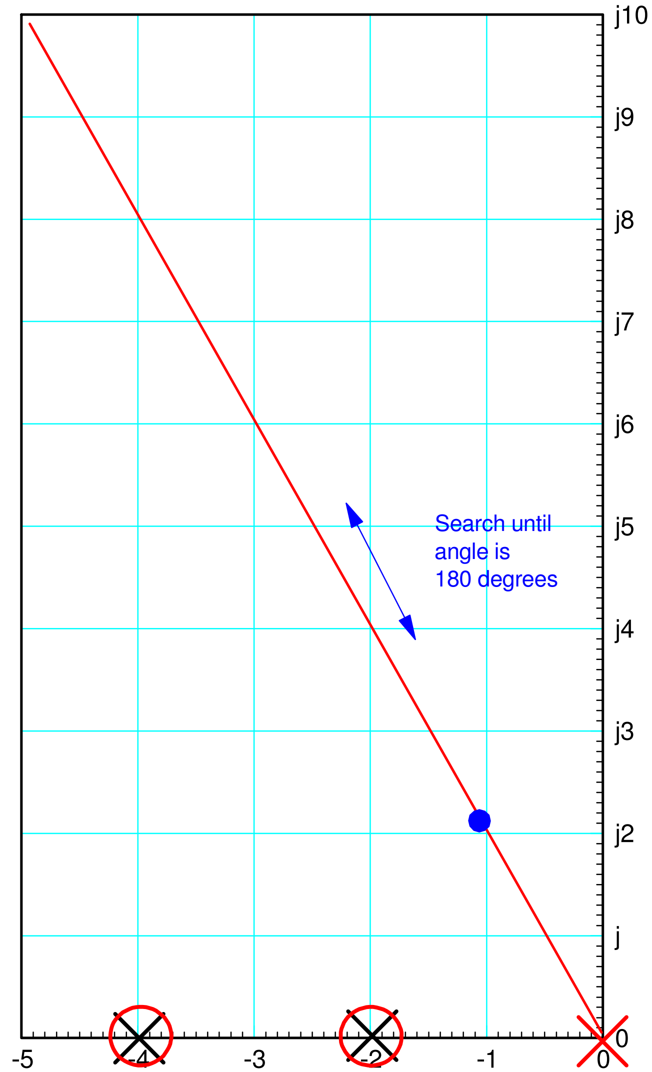
s = -1.0747 + j 2.1494

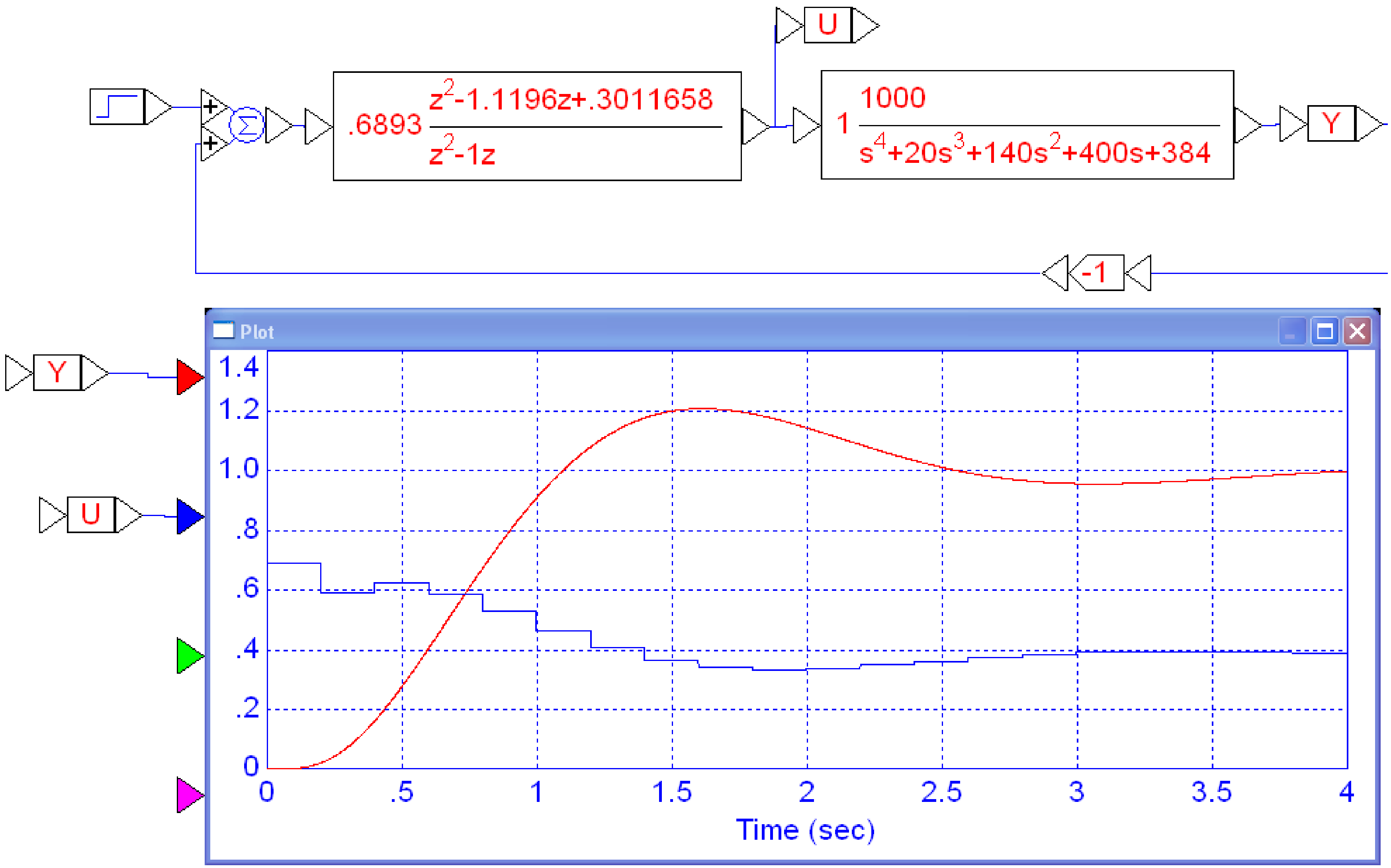
z = 0.7332 + j 0.3362

k = 0.6893

and

$$K(z) = 0.6893\left(\frac{(z-0.6703)(z-0.4493)}{z(z-1)}\right)$$



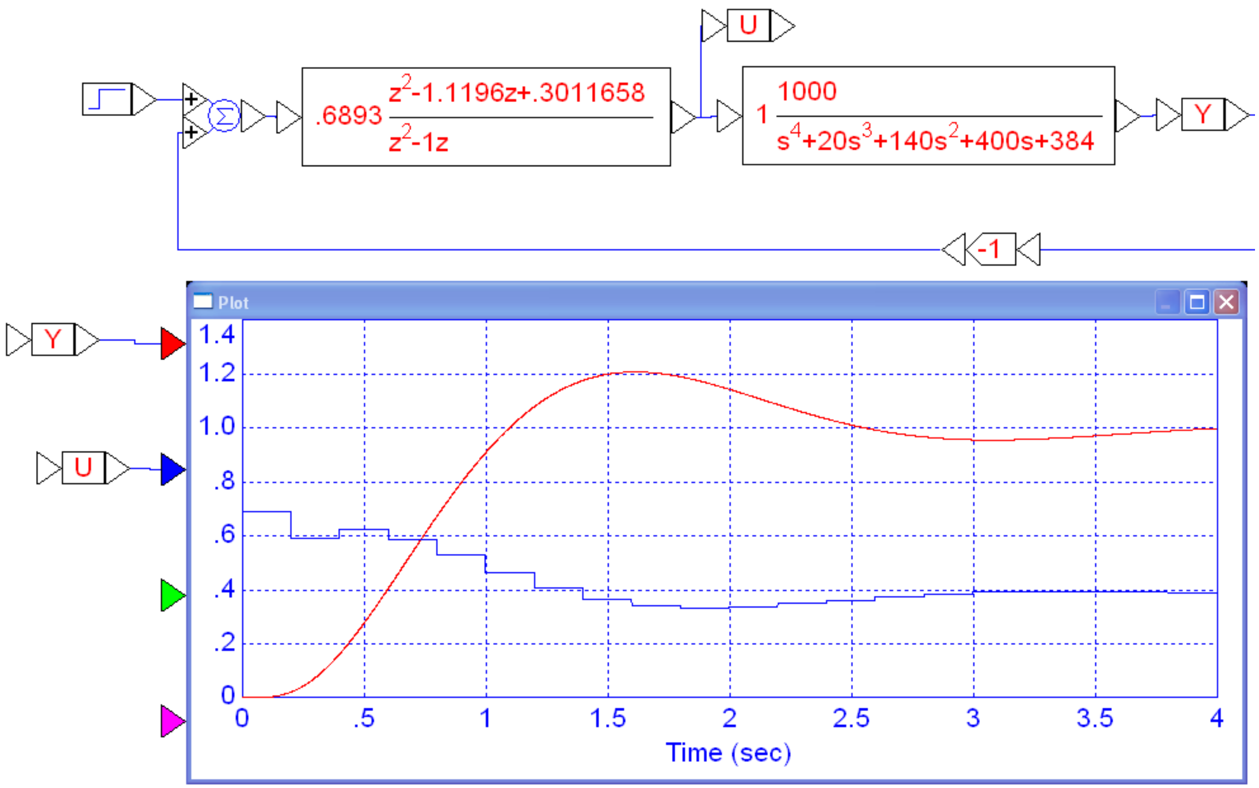


U
.6893
z2-1.1196z+.3011658
z2-1z
1
1000
s4+20s3+140s2+400s+384
Y
-1
Y
U
Plot
1.4
1.2
1.0
.8
.6
.4
.2
0
0
.5
1
1.5
2
2.5
3
3.5
4
Time (sec)

This is about the same response as we had before, only with

- A much more reasonable input at t = 0 (0.689 vs. 2.615)
- A much slower sampling rate (200ms vs 50ms)

Faster sampling rates are not always good. They can actually cause problems.



Summary:

Digital PID control is similar to analog PID

- P: Gain compensation
- I: Add an integrator (make the system type-1)
- PI: Add an integrator and one zero. Cancel one pole with the zero
- PID: Add an integrator and two zeros. Cancel two poles

The main difference is 'D' stands for *delay* rather than *derivative*