Solution to Homework #11: ECE 461/661

Discrete-Time Compensator Design. Due Monday, November 19, 2018

Assume

$$G(s) = \left(\frac{200}{(s+2)(s+5)(s+10)(s+15)}\right)$$

Problem 1: Assume a sampling rate of 100ms.

Design a compensator, K(z), which results in

- 20% overshoot for a step input.
- No error for a step input, and
- A 2% settling time of 2 seconds

Translation:

- Make it a type-1 system
- Place the closed-loop dominant pole at s = -2 + j4

Pick K(z) to

- Add a pole at s = 0 (z = 1)
- Cancel the poles at s = -2 (z = -0.819)
- Cancel the pole at s = -5 (z = -0.607)
- Add a a pole (somewhere) to place the closed-loop dominant pole at s = -2 + j4

The net open-loop system is then:

$$G(s) \cdot K(z) \cdot \Delta = \left(\frac{200}{(s+2)(s+5)(s+10)(s+15)}\right) \cdot \left(\frac{(z-0.819)(z-0.607)}{(z-1)(z-a)}\right) \cdot e^{-0.05s}$$

Evaluate what you know at

- s = -2 + j4
- z = 0.7541 + j0.3188 $(z = e^{sT})$

$$\left(\left(\frac{200}{(s+2)(s+5)(s+10)(s+15)}\right) \cdot \left(\frac{(z-0.819)(z-0.607)}{(z-1)}\right) \cdot e^{-0.1s}\right)_{s=-2+j4} = 0.0258\angle -159.1^0$$

For the angles to add up to 180 degrees

$$\angle(z-a) = 20.9^0$$

$$a = 0.7541 - \frac{0.3188}{\tan\left(20.9^0\right)} = -0.0834$$

so

$$G(s) \cdot K(z) \cdot \Delta = \left(\frac{200}{(s+2)(s+5)(s+10)(s+15)}\right) \cdot \left(\frac{(z-0.819)(z-0.607)}{(z-1)(z-0.0834)}\right) \cdot e^{-0.05s}$$

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

$$\left(\left(\frac{200}{(s+2)(s+5)(s+10)(s+15)}\right) \cdot \left(\frac{(z-0.819)(z-0.607)}{(z-1)(z+0.0834)}\right) \cdot e^{-0.05s}\right)_{s=-2+j4} = 0.0288\angle 180^0$$

to make the gain one

$$k = \frac{1}{0.0288} = 34.7658$$

resulting in

$$K(z) = 34.7658\left(\frac{(z-0.819)(z-0.607)}{(z-1)(z+0.0834)}\right)$$

Write pseudo-code to implement K(z)

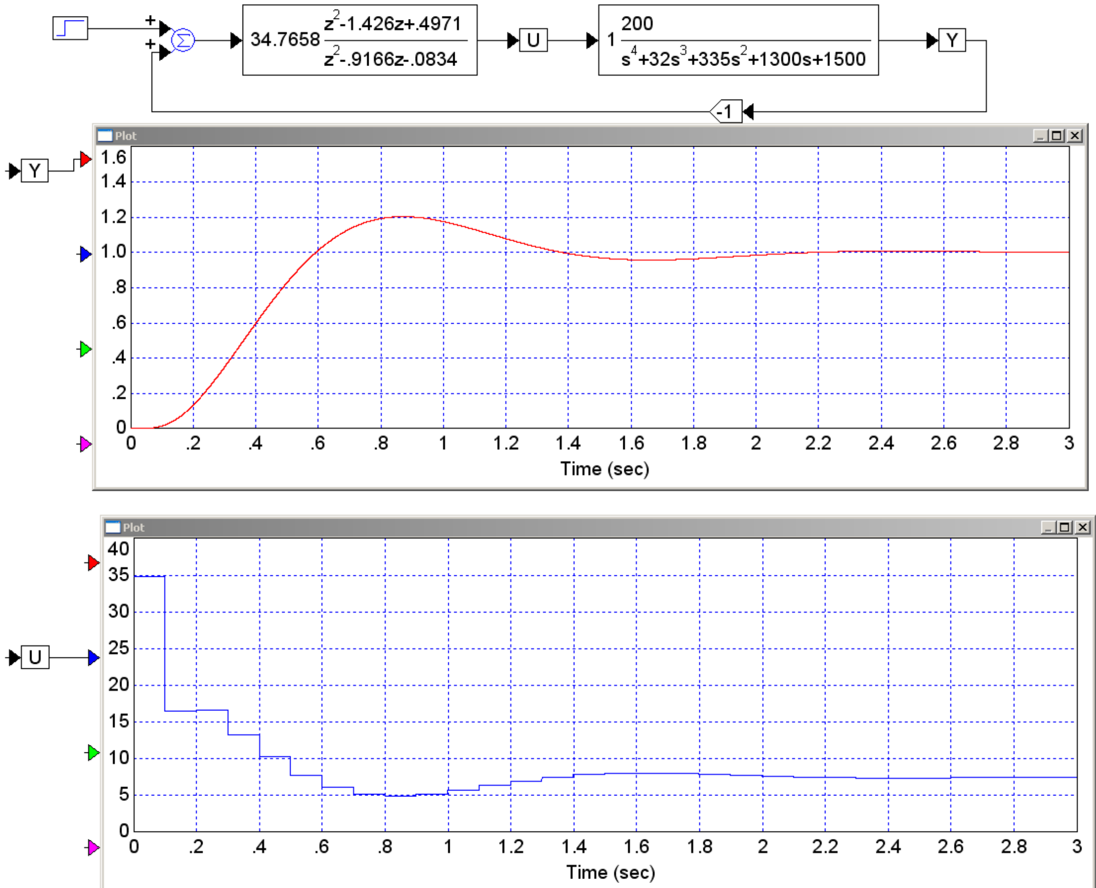
$$Y = 34.7658\left(\frac{(z-0.819)(z-0.607)}{(z-1)(z+0.0834)}\right)X$$

$$(z^2 - 0.9166z - 0.0834)Y = 34.7658(z^2 - 1.4260z + 0.4971)X$$

Code:

```
while(1) {

   x2 = x1;           // x(k-2)
   x1 = x0;           // x(k-1)
   x0 = A2D_Read(0); // x(k)

   y2 = y1;           // y(k-2)
   y1 = y0;           // y(k-1)
   y0 = 0.9166*y1 +0.0834*y2 + 34.7658*(x0 - 1.4260*x1 + 0.4871*x2);

   Wait_100ms();

   }
```

Plot the step response of the closed-loop system



34.7658 z^2-1.426z+.4971 / z^2-.9166z-.0834
U
1 200 / s^4+32s^3+335s^2+1300s+1500
Y
-1
Plot
Y
1.6
1.4
1.2
1.0
.8
.6
.4
.2
0
0 .2 .4 .6 .8 1 1.2 1.4 1.6 1.8 2 2.2 2.4 2.6 2.8 3
Time (sec)
Plot
U
40
35
30
25
20
15
10
5
0
0 .2 .4 .6 .8 1 1.2 1.4 1.6 1.8 2 2.2 2.4 2.6 2.8 3
Time (sec)

Problem 2: Assume a sampling rate of 250ms.

- Design a compensator, K(z), which results in
  - 20% overshoot for a step input.
  - No error for a step input, and
  - A 2% settling time of 2 seconds
- Write pseudo-code to implement K(z)
- Plot the step response of the closed-loop system using VisSim or Simulink (or similar program)

First, compute the closed-loop pole location in the z-plane:

```
>> T = 0.25;
>> s = -2 + j*4;
>> z = exp(s*T)

z =   0.3277 + 0.5104i
```

Find

$$G(s) \cdot \Delta_{T/2} \cdot K(z) = 1\angle 180^0$$

$$\left(\frac{200}{(s+2)(s+5)(s+10)(s+15)}\right) \cdot e^{-0.125s} \cdot K(z) = 1\angle 180^0$$

Pick the zeros of K(z) to cancel the poles at s = -2 and s = -5

```
>> exp(-2*T)

ans =    0.6065

>> exp(-5*T)

ans =    0.2865
```

so

$$K(z) = k\left(\frac{(z-0.6065)(z-0.2865)}{(z-1)(z-a)}\right)$$

Evaluate what you know:

$$\left(\frac{200}{(s+2)(s+5)(s+10)(s+15)}\right) \cdot e^{-0.125s} \cdot \left(\frac{(z-0.6065)(z-0.2865)}{(z-1)}\right) = 0.0373\angle -154^0$$

To make the angles 180 degrees

$$\angle(z-a) = 25.7876^0$$

$$a = 0.3277 - \left(\frac{0.5104}{\tan\left(25.7876^0\right)}\right) = -0.7287$$

and

$$K(z) = k\left(\frac{(z-0.6065)(z-0.2865)}{(z-1)(z+0.7287)}\right)$$

To get the gain:

$$\left(\frac{200}{(s+2)(s+5)(s+10)(s+15)}\right) \cdot e^{-0.125s} \cdot \left(\frac{(z-0.6065)(z-0.2865)}{(z-1)(z+0.7287)}\right) = 0.0317\angle 180^0$$

$$k = \frac{1}{0.0317} = 31.5085$$

giving

$$K(z) = 31.5085\left(\frac{(z-0.6065)(z-0.2865)}{(z-1)(z+0.7287)}\right)$$

Write pseudo-code to implement K(z)

$$K(z) = 31.5085\left(\frac{z^2-0.893z+0.1738}{z^2-0.2713z-0.7287}\right)$$

```
while(1) {

   x2 = x1;            // x(k-2)
   x1 = x0;            // x(k-1)
   x0 = A2D_Read(0);  // x(k)

   y2 = y1;            // y(k-2)
   y1 = y0;            // y(k-1)
   y0 = 0.2713*y1 + 0.7287*y2 + 31.5085*(x0 - 0.893*x1 + 0.1738*x2);

   Wait_100ms();

   }
```

Checking in VisSim