1)

Stability: $\dot{x_1} = 0;\ \dot{x_2} = 0;\ \dot{x_3} = 0$

Stable points: $\begin{cases} x_1^* = y^* \\ x_2^* = 0 \\ x_3^* = \sqrt{\frac{Mg}{c}\,(0.0072 - x_1^*)} \end{cases} ==> \begin{cases} x_1^* = 0.002 \\ x_2^* = 0 \\ x_3^* = \pm 0.000784257 \end{cases}$

$$A = \begin{pmatrix} 0 & 1 & 0 \\ \frac{c}{M}\frac{x_3^2}{(0.0072 - x_1)^2} & 0 & \frac{2c}{M}\frac{x_3}{0.0072 - x_1} \\ 0 & 0 & -\frac{R}{L} \end{pmatrix}_{x^*} = \begin{pmatrix} 0 & 1 & 0 \\ 1886.54 & 0 & \pm 25017.31 \\ 0 & 0 & -86.956 \end{pmatrix}$$

$$B = \begin{pmatrix} 0 \\ 0 \\ \frac{1}{L} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 8.696 \end{pmatrix}, C = (1 \quad 0 \quad 0), D = 0$$

Transfer Function: $G(s) = C(sI - A)^{-1}B + D$

$$G_1(s) = \frac{217600}{s^3 + 86.96s^2 - 1887s - 164000} \quad (for\ the\ positive\ value\ of\ x_3)$$

$$G_2(s) = \frac{-217600}{s^3 + 86.96s^2 - 1887s - 164000} \quad (for\ the\ negative\ value\ of\ x_3)$$

2)

Since zero error is demanded, we need a controller that has an integrator in it. So we use PID and the coefficients are going to be:

Proportional (P): 6.93550294613259

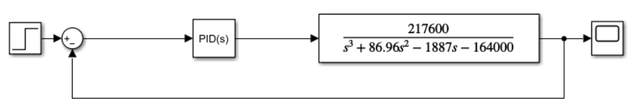
Integral (I): 13.9271022614454

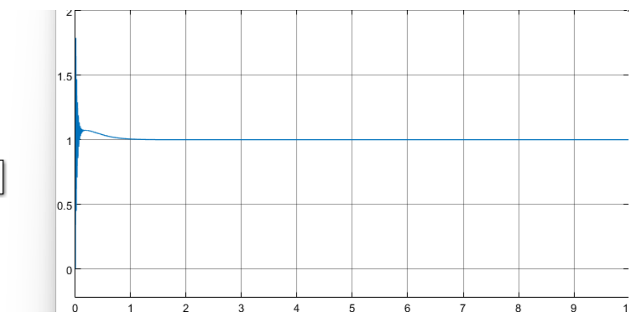
Derivative (D): 0.32804013853127

Use filtered derivative

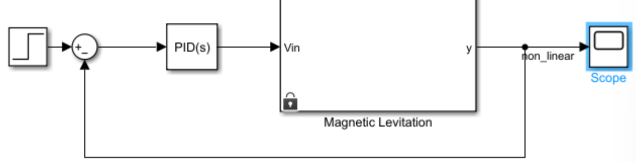
Filter coefficient (N): 29345.7140521187

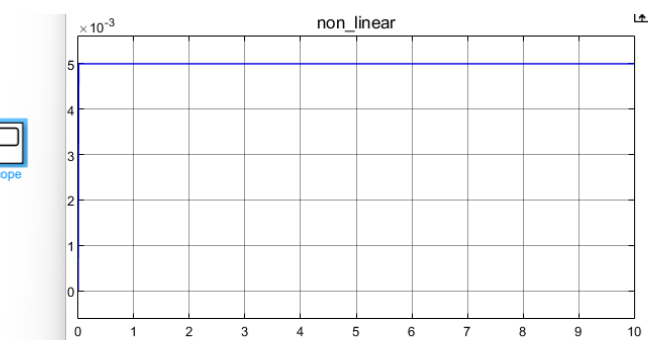
We've reached 1 after a few seconds.





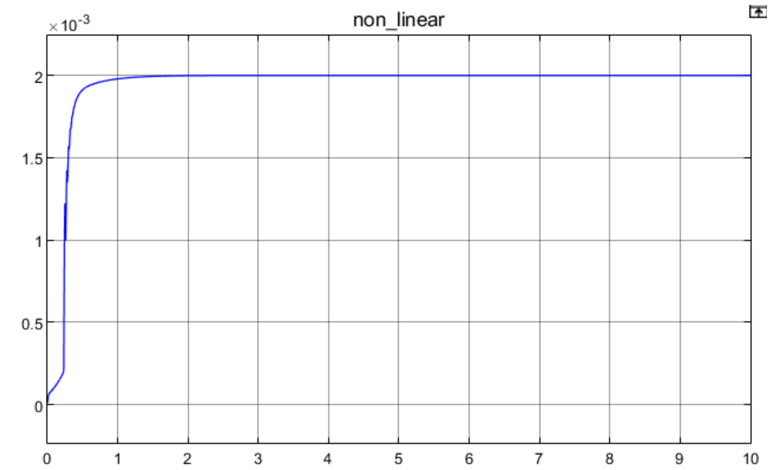
As we can see, we have a zero persistent error after less than a second. However, the output signal experienced major fluctuations and overshoot at the beginning which is not good in controlling a system.





It seems the designed controller can't control the system with input=1 because the non-linear system goes to a saturation state.

So we reduce the input gain(input = 2e-3):



We can see our continuous controller is able to control the main system.

3)

$$G_D(s) = \frac{(NK_D + K_p)s^2 + (K_I + K_P N)s + K_I N}{s(s+N)}$$

Controller in z space with “Tustin” method:

$$s \to \frac{2(1 - z^{-1})}{T(1 + z^{-1})}$$

- For T = 0.0001:

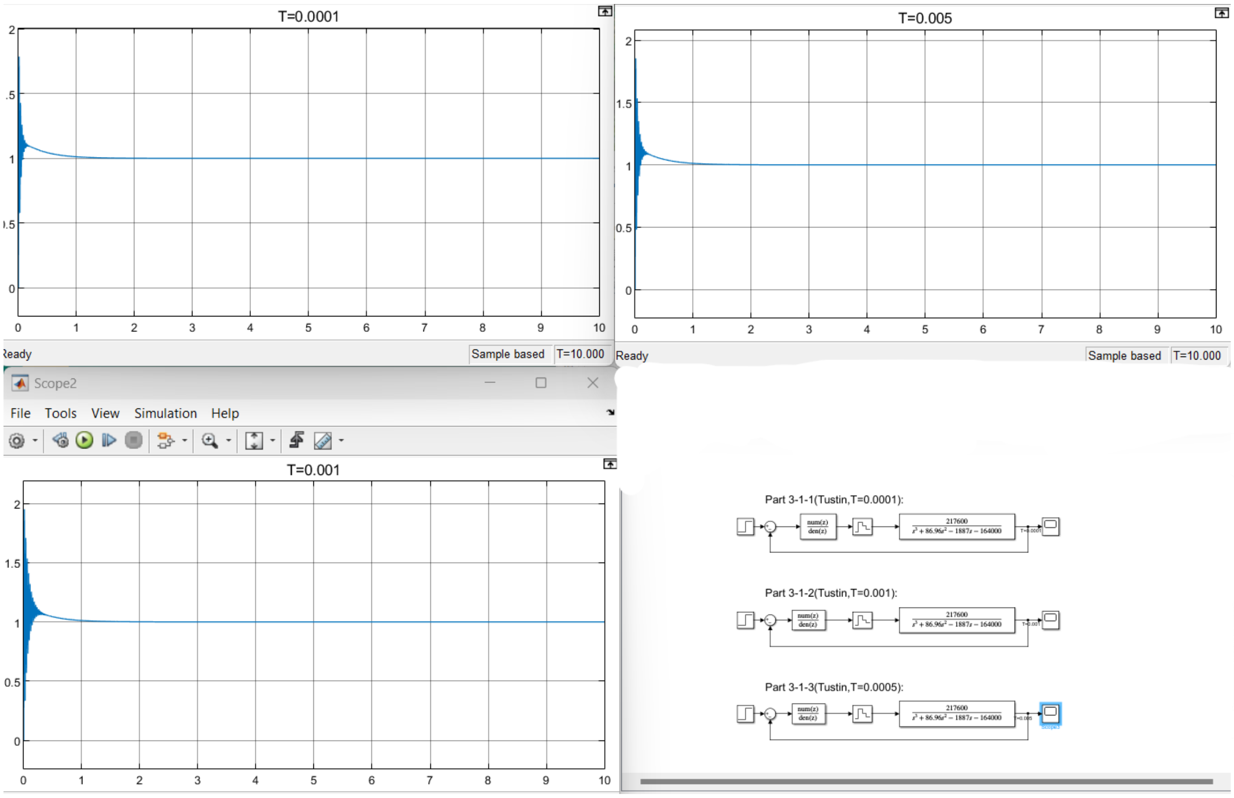
$$G_{D(z)} = \frac{38570z^2 - 77060z + 38490}{9.869z^2 - 8z - 1.869}$$

- For T = 0.0005:

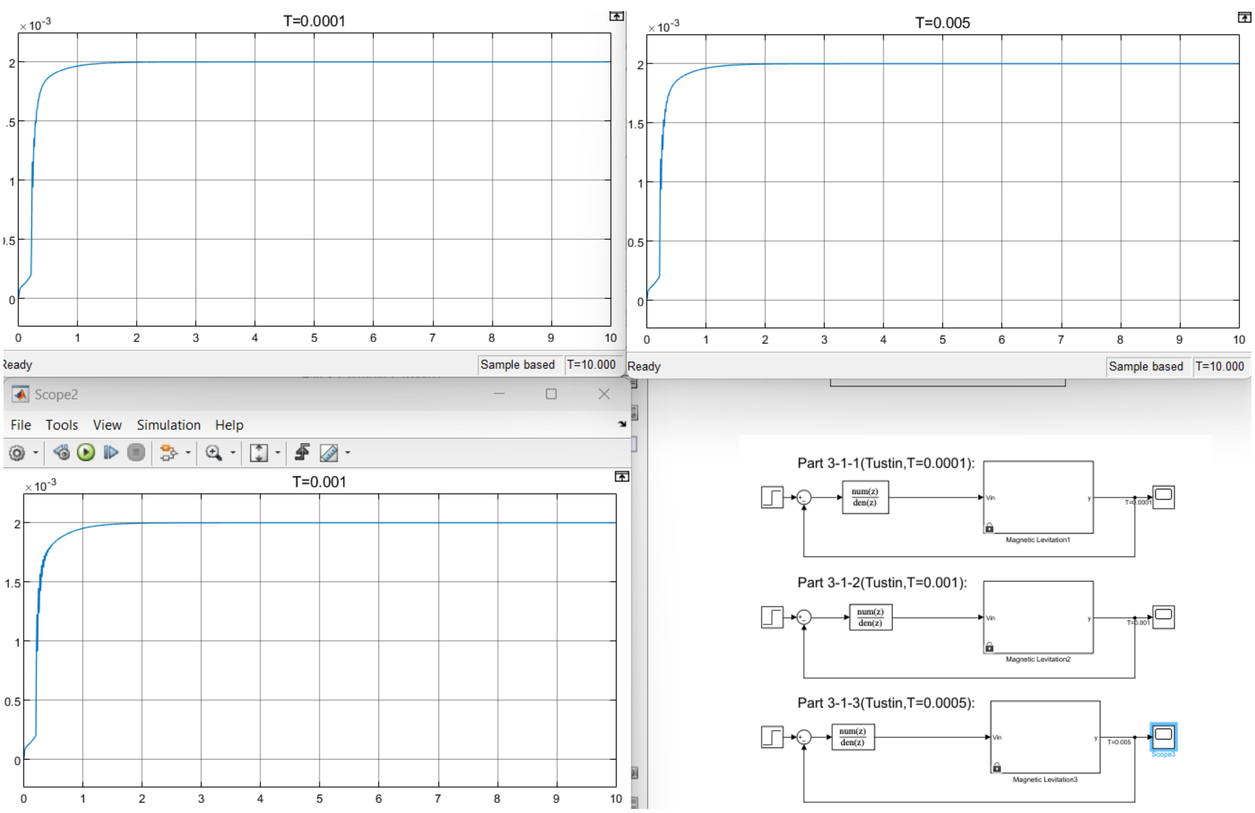
$$G_{D(z)} = \frac{38940z^2 - 77060z + 38120}{62.69z^2 - 8z - 54.69}$$

- For T = 0.001:

$$G_{D(z)} = \frac{38730z^2 - 77060z + 38330}{33.35z^2 - 8z - 25.35}$$



For the non-linear system: (input gain = 2e-3)



Controller in z space with "Corresponded Zero and Pole" method:

$$G_{D(z)} = K\frac{\left(z - e^{-T\left(-\frac{(K_I+K_PN)}{2(NK_D+K_p)}+i\sqrt{\frac{(K_I+K_PN)}{2(NK_D+K_p)}^{2}-\frac{K_IN}{(NK_D+K_p)}}\right)}\right)}{(z^2-1)(z-e^{NT})}$$

$$* \left(z - e^{-T\left(-\frac{(K_I+K_PN)}{2(NK_D+K_p)}-i\sqrt{\frac{(K_I+K_PN)}{2(NK_D+K_p)}^{2}-\frac{K_IN}{(NK_D+K_p)}}\right)}\right)$$

$$= \frac{K_D\left(\left(z-e^{-T(-5.3412+1.7806i)}\right)\left(z-e^{-T(-5.3412+17806i)}\right)\right)}{(z^2-1)(z-e^{-46503T})}; \lim_{s\to 0} G(s) = \lim_{z\to 1} G(z)$$

- For T = 0.0001:

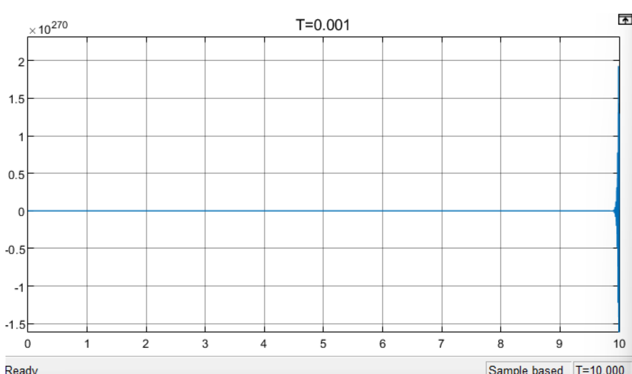
$$G_{D(z)} = K_D\frac{z^2-1.998z+0.9979}{z^2-1.053z+0.05315}; K_D \gg 1 \to K_D = 1000$$

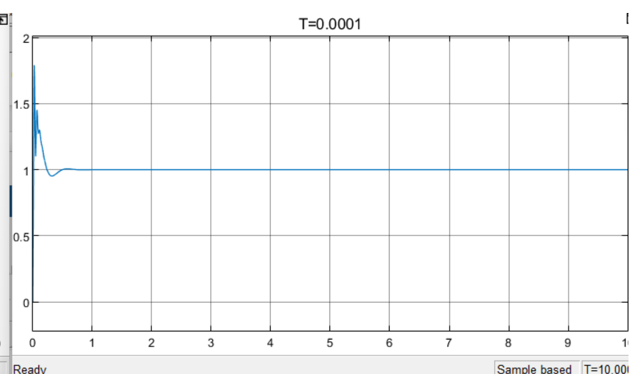
- For T = 0.0005:

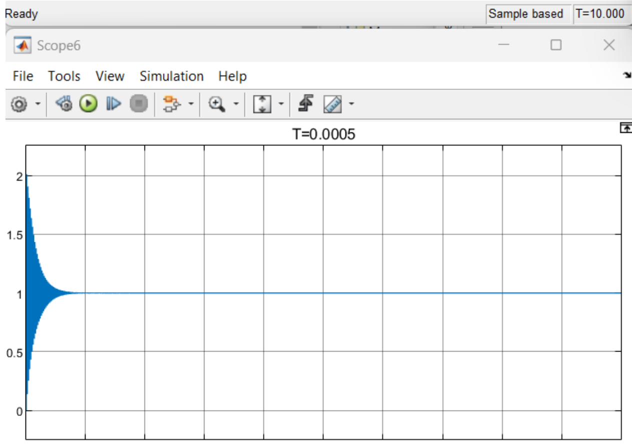
$$G_{D(z)} = K_D\frac{z^2-1.979z+0.9791}{z^2-z+1.8e-13}; K_D \gg 1 \to K_D = 1000$$

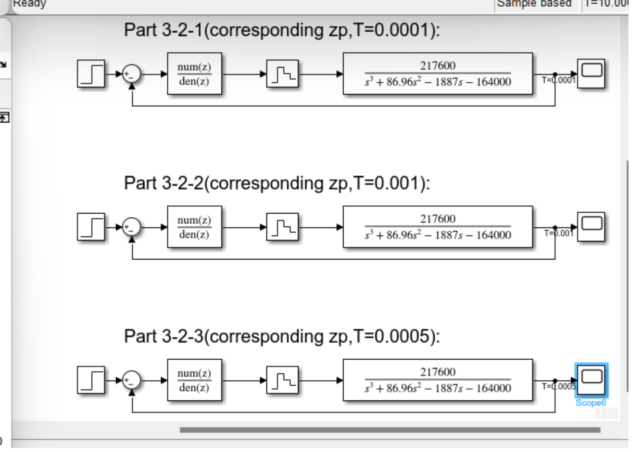
- For T = 0.001:

$$G_{D(z)} = K_D\frac{z^2-1.989z+0.989.5}{z^2-z+4.243e-7}; K_D \gg 1 \to K_D = 1000$$

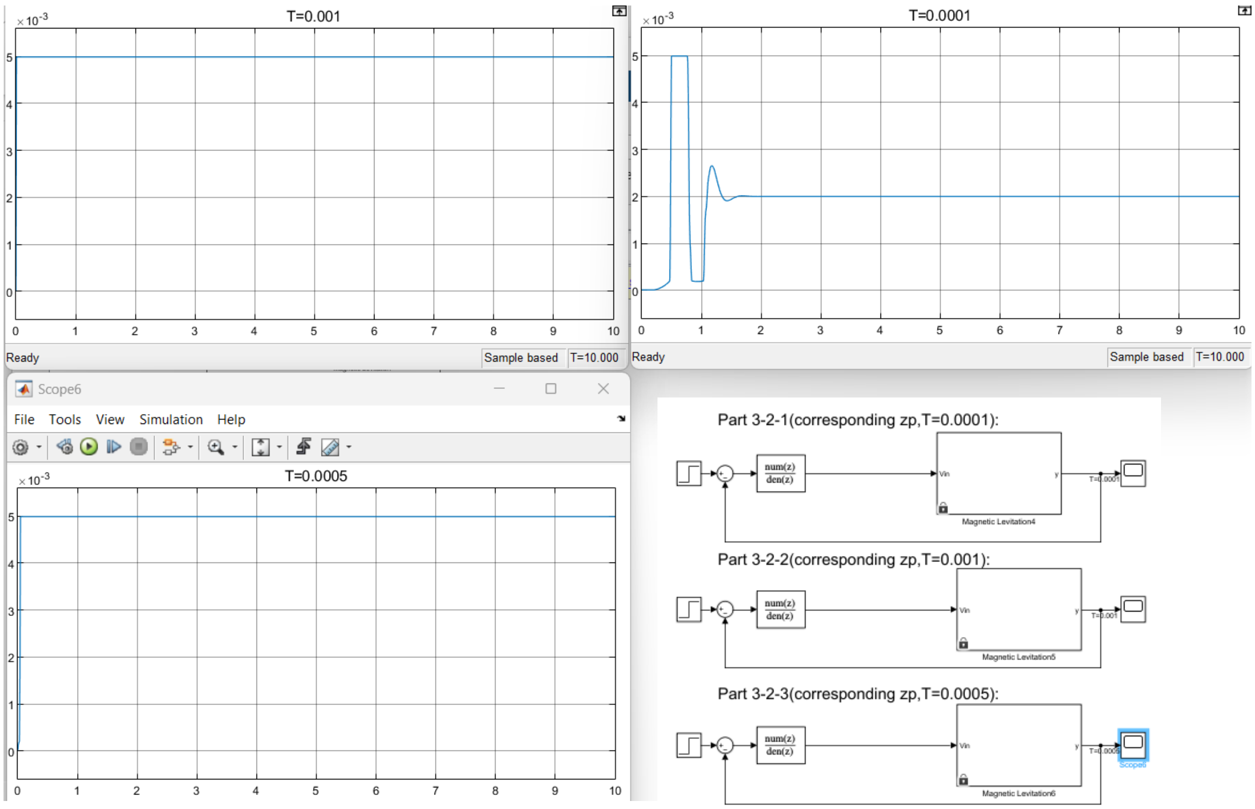








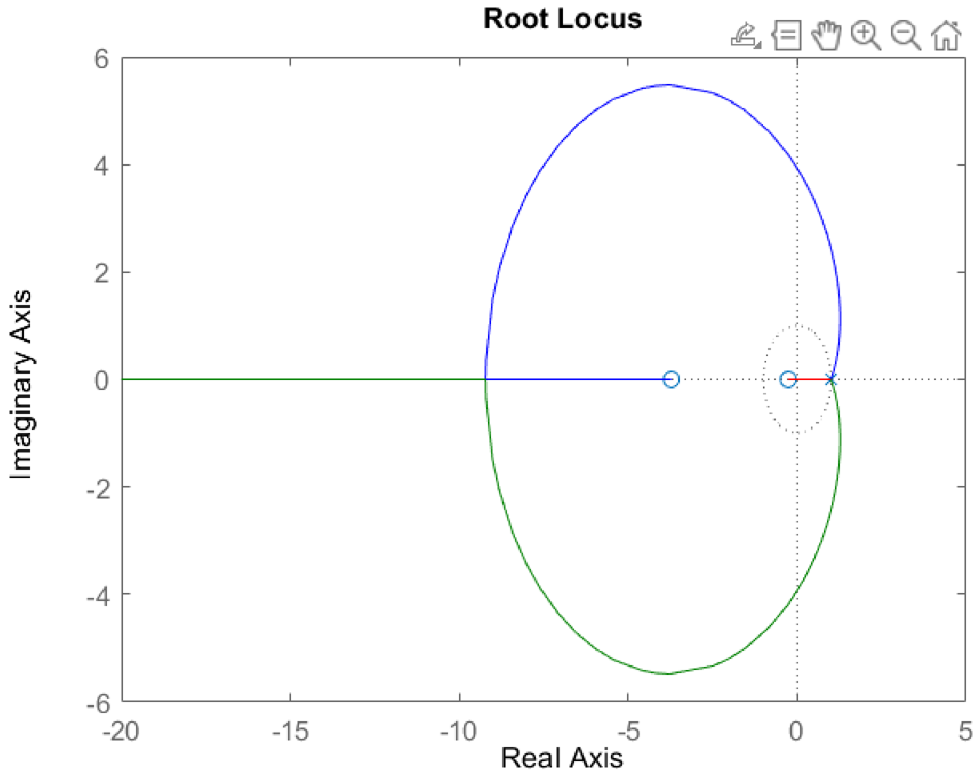
For the non-linear system: (input gain = 2e-3)

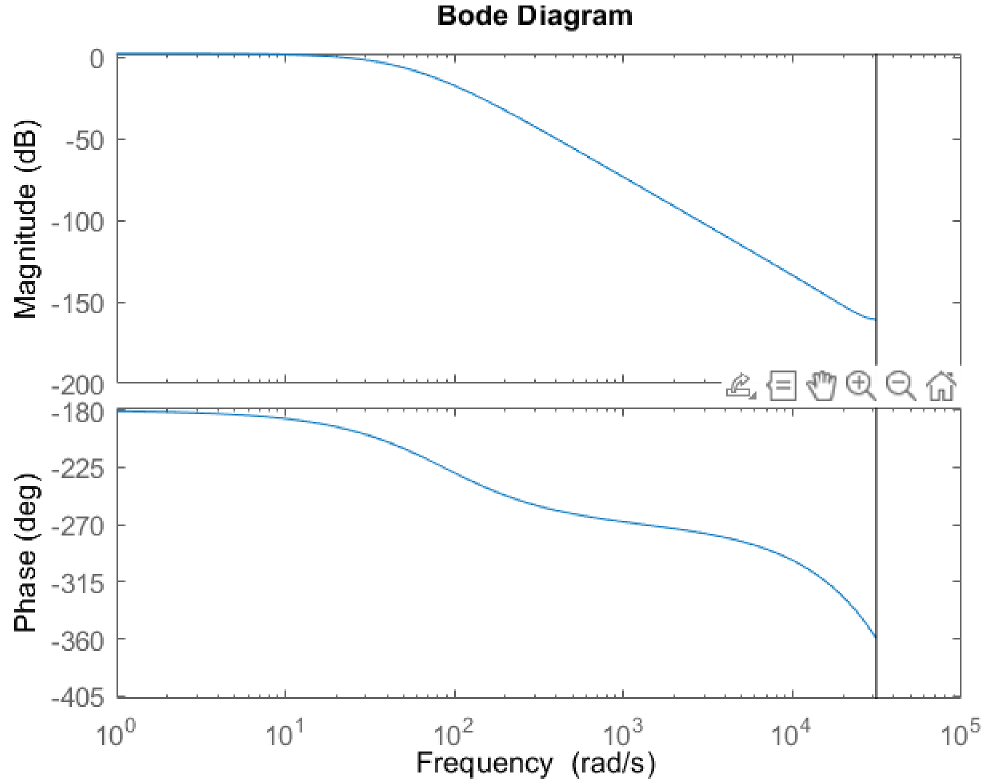


We can see the most reasonable sampling time is the first one with T=0.0001s which is the closest approximation to the continuous system. It has the lowest overshoot among other sampling times with a minimum fluctuation. It also seems the Tustin method is more robust through different sampling times and has a better performance comparing to the other method.

4)

$$G(z) = Z\left\{\frac{1-e^{-Ts}}{s}G_1(s)\right\} = \frac{z-1}{z}Z\left\{\frac{G1(s)}{s}\right\}$$
$$= \frac{3.603e-8z^{-3}+1.444e-7z^{-2}+3.619e-8z^{-1}}{1-0.9913z^{-3}+2.983z^{-2}-2.991z^{-1}}$$



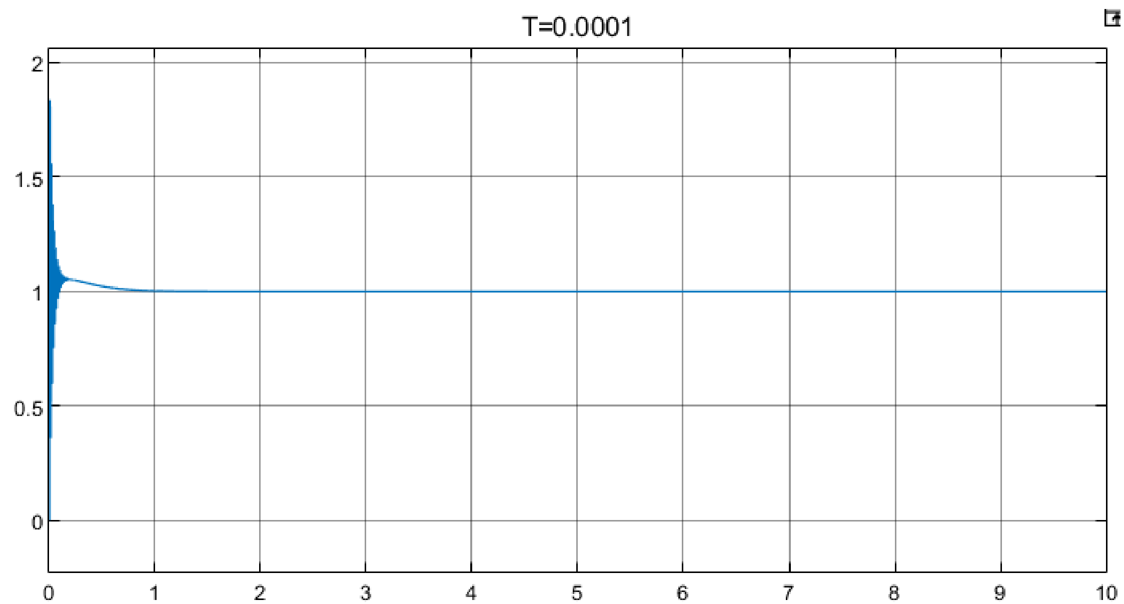



```
Gain Margin (dB):
   -2.4563
Phase Margin (degrees):
  -14.9352
Gain Crossover Frequency (rad/s):
     0
Phase Crossover Frequency (rad/s):
   23.0778
Bandwidth (rad/s):
   42.4750
```

Because our system is unstable we can't stabilize it by just gain, so we can't specify a range for stable gains.

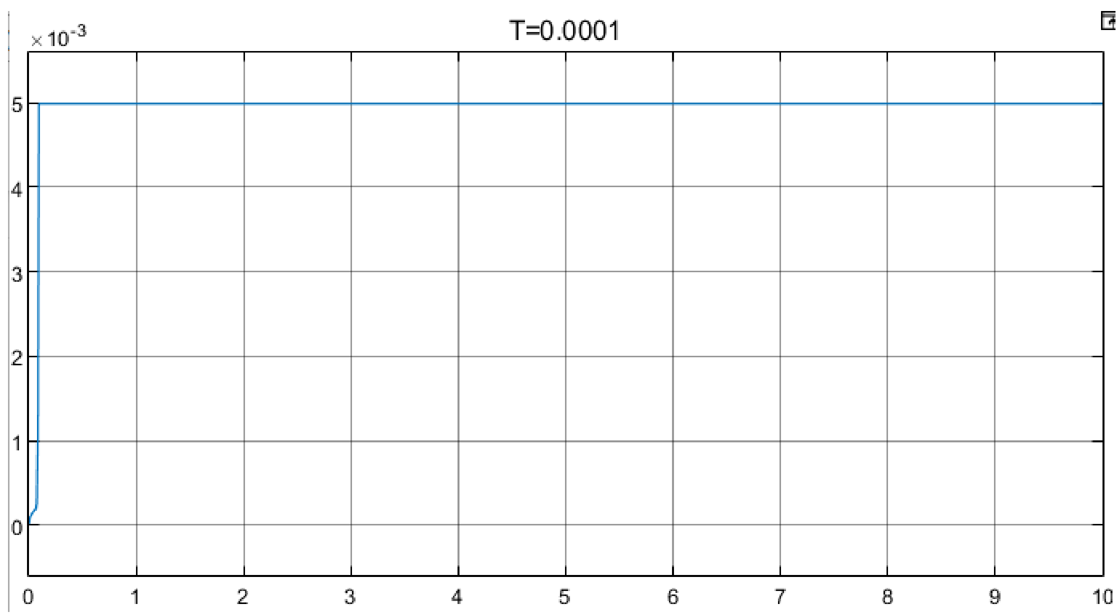
5)

Since we want the same attitude as the part 2 system, zero persistent error, and low settling time, we design a discrete PID here.

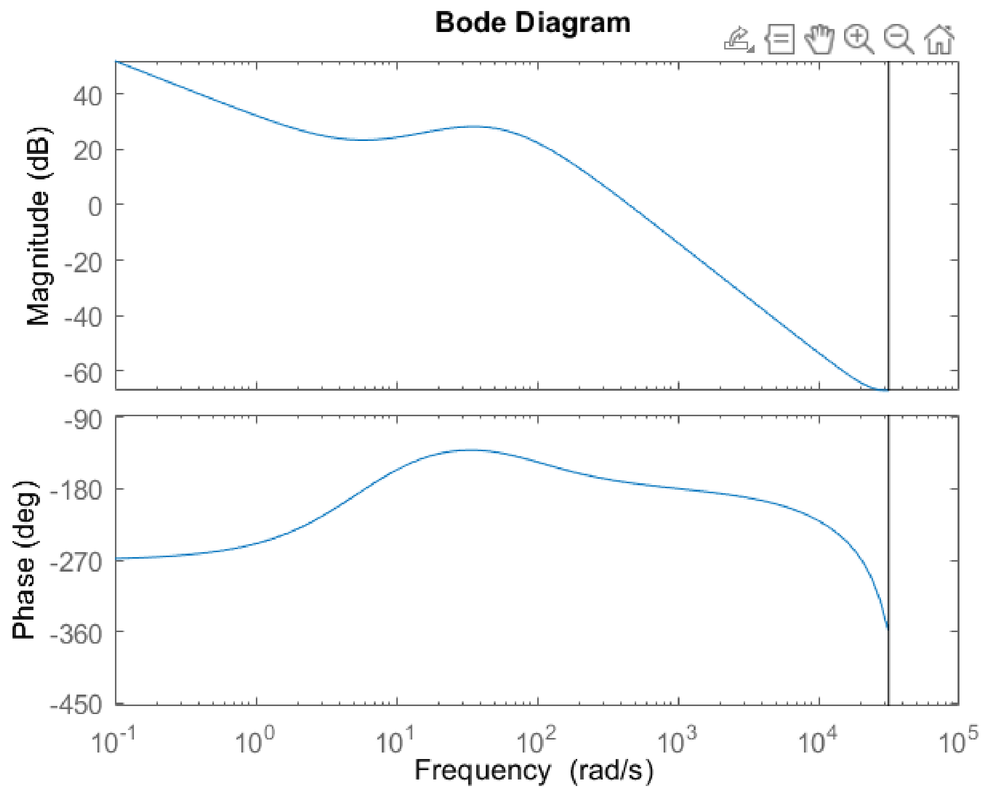


We can see the output signal is similar to part 2's output and we have reached our desired controller in discrete space.

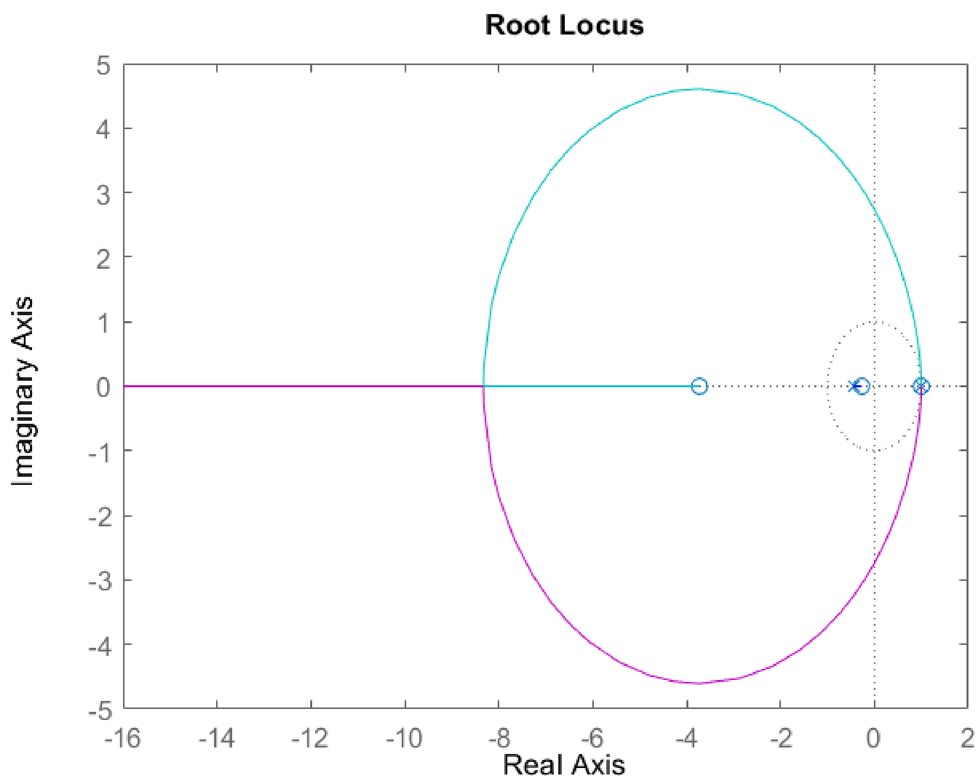
Non-linear response: (input gain = 2e-3)



6)



Bode Diagram
Magnitude (dB)
40
20
0
-20
-40
-60
Phase (deg)
-90
-180
-270
-360
-450
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
Frequency (rad/s)



Root Locus
Imaginary Axis
5
4
3
2
1
0
-1
-2
-3
-4
-5
-16
-14
-12
-10
-8
-6
-4
-2
0
2
Real Axis

```
Gain Margin (dB):
   14.3288
Phase Margin (degrees):
    7.6355
Gain Crossover Frequency (rad/s):
   1.0389e+03
Phase Crossover Frequency (rad/s):
  450.2725
Bandwidth (rad/s):
  619.4085
```

```
Stable gain range:
K_min = 0.19605
K_max = 5.0834
```

We can see phase and gain margins have been increased compared to the uncontrolled system which means that our system has more penalty before unstability. Also, our bandwidth becomes larger, and we can find a gain range to remain stable.

7)

$$F(z) = f_1 z^{-1} + f_2 z^{-2} + f_3 z^{-3}$$

We have one out-of-unit circle zero, our unstable zero: $z = -3.72$

$$F(z) = (3.72z^{-1} + 1)M(z) \to M(z) = m_0 + m_1 z^{-1} + m_2 z^{-2}$$

$$m_0 = 0; m_1 = f_1;\ m_2 + 3.72m_1 = f_2; 3.72m_2 = f_3$$

$$f_3 + 3.72^2 f_1 = 3.72 f_2$$

We have two out-of-unit circle poles, our unstable poles: $p = 1.0395 \pm 0.08i$

$$1 - F(z) = \big((1.0395 + 0.08i)z^{-1} + 1\big)\big((1.0395 - 0.08i)z^{-1} + 1\big)N(z)$$

$$\to N(z) = n_0 + n_1 z^{-1} + n_2 z^{-2}$$

$$n_0 = 1; n_1 + 2.079n_0 = -f_1; n_2 + 2.079n_1 + 1.074n_0 = -f_2; 2.079n_2 + 1.074n_1 = -f_3$$

$$2.079(-f_2 - 2.079(-f_1 - 2.079) - 1.074) + 1.074(-f_1 - 2.079) = -f_3$$

We have step input:

$$1 - F(z) = (1 - z^{-1})Q(z) \to Q(z) = q_0 + q_1 z^{-1} + q_2 z^{-2}$$

$$1 = q_0;\ -f_1 = -q_0 + q_1;\ -f_2 = -q_1 + q_2;\ -f_3 = -q_2$$

$$-f_3 = -\big(-f_2 + (-f_1 + 1)\big)$$

By solving the highlighted equations, we can find F(z) properly.

$$f_1 = 0.7945; f_2 = 2.373; f_3 = -2.1676$$

Since this controller can't stabilize the system, we changed our function to:

$$F = f_1 z^{-1} + f_2 z^{-2} + f_3 z^{-3} + f_4 z^{-4}$$

We have one out-of-unit circle zero, our unstable zero: $z = -3.72$

$$F(z) = (3.72z^{-1} + 1)M(z) \to M(z) = m_0 + m_1 z^{-1} + m_2 z^{-2} + m_3 z^{-3}$$

$$m_0 = 0; m_1 = f_1;\ m_2 + 3.72m_1 = f_2; m_3 + 3.72m_2 = f_3; 3.72m_3 = f_4$$

$$f_4 = 3.72\big(f_3 - 3.72(f_2 - 3.72f_1)\big)$$

We have two out-of-unit circle poles, our unstable poles: $p = 1.0044$

$$1 - F(z) = (-1.0044z^{-1} + 1)N(z)$$

$$\to N(z) = n_0 + n_1 z^{-1} + n_2 z^{-2} + n_3 z^{-3}$$

$$n_0 = 1; n_1 + 1.0044n_0 = -f_1;\ n_2 + 1.0044n_1 = -f_2; n_3 + 1.0044n_2 = -f_3; 1.0044n_3 = -f_4$$

$$-f_4 = 1.0044\left(-f_3 - 1.0044\big(-f_2 - 1.0044(-f_1 - 1.0044)\big)\right)$$

We have step input:

$$1 - F(z) = (1 - z^{-1})Q(z) \to Q(z) = q_0 + q_1 z^{-1} + q_2 z^{-2} + q_3 z^{-3}$$

$$1 = q_0;\ -f_1 = -q_0 + q_1;\ -f_2 = -q_1 + q_2;\ -f_3 = -q_2 + q_3; -f_4 = -q_3$$

$$-f_4 = -\Big(-f_3 + \big(-f_2 + (-f_1 + 1)\big)\Big)$$

Not having fluctuations in sampling:

$$\frac{F(z)}{G(z)}R(z) = U(z) \to U(z) = u_0 + u_1 z^{-1} + u_2 z^{-2} + u_3 z^{-3}$$

$$\frac{(f_1 z^{-1} + f_2 z^{-2} + f_3 z^{-3} + f_4 z^{-4})(1 - 0.9913z^{-3} + 2.983z^{-2} - 2.991z^{-1})}{(1 - z^{-1})(3.603e - 8z^{-3} + 1.444e - 7z^{-2} + 3.619e - 8z^{-1})} = u_0 + u_1 z^{-1} + u_2 z^{-2} + u_3 z^{-3}$$

$$
\begin{aligned}
f_1 z^{-1} + (f_2 - 2.991 f_1) z^{-2} + (f_3 - 2.991 f_2 + 2.983 f_1) z^{-3} \\
+ (f_4 - 2.991 f_3 + 2.983 f_2 - 0.9913 f_1) z^{-4} \\
+ (-2.991 f_4 + 2.983 f_3 - 0.9913 f_2) z^{-5} + (2.983 f_4 - 0.9913 f_3) z^{-6} \\
- 0.9913 f_4 z^{-7} \\
= (3.619 \times 10^{-8} z^{-1} + 1.082 \times 10^{-7} z^{-2} - 1.0837 \times 10^{-7} z^{-3} \\
- 3.603 \times 10^{-8} z^{-4})(u_0 + u_1 z^{-1} + u_2 z^{-2} + u_3 z^{-3})
\end{aligned}
$$

$$3.619 \times 10^{-8} u_0 = f_1;\ 1.082 \times 10^{-7} u_0 + 3.619 \times 10^{-8} u_1 = f_2 - 2.991 f_1;$$

$$-1.0837 \times 10^{-7} u_0 + 1.082 \times 10^{-7} u_1 + 3.619 \times 10^{-8} u_2 = f_3 - 2.991 f_2 + 2.983 f_1;$$

$$
\begin{aligned}
-1.0837 \times 10^{-7} u_1 + 1.082 \times 10^{-7} u_2 + 3.619 \times 10^{-8} u_3 - 3.603 \times 10^{-8} u_0 \\
= f_4 - 2.991 f_3 + 2.983 f_2 - 0.9913 f_1;
\end{aligned}
$$

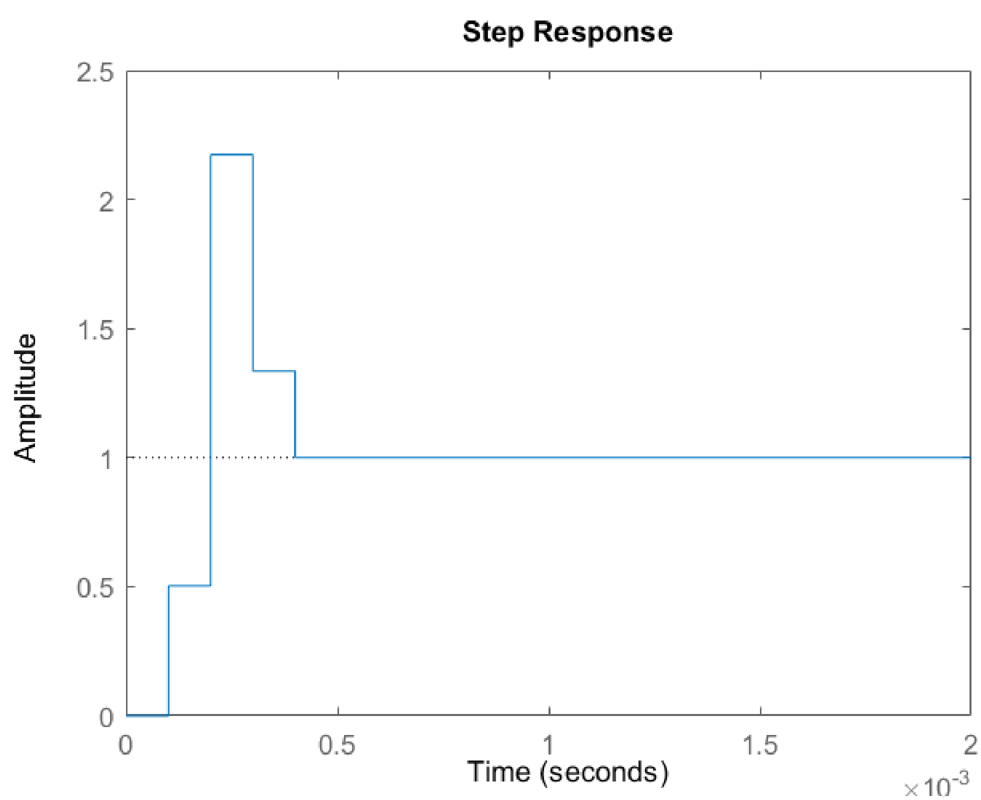
$$
\begin{aligned}
-3.603 \times 10^{-8} u_1 - 1.0837 \times 10^{-7} u_2 + 1.082 \times 10^{-7} u_3 \\
= -2.991 f_4 + 2.983 f_3 - 0.9913 f_2;
\end{aligned}
$$

$$-3.603 \times 10^{-8} u_2 - 1.0837 \times 10^{-7} u_3 = 2.983 f_4 - 0.9913 f_3$$

$$-3.603 \times 10^{-8} u_3 = -0.9913 f_4$$

By solving the highlighted equations, we can find F(z) properly.

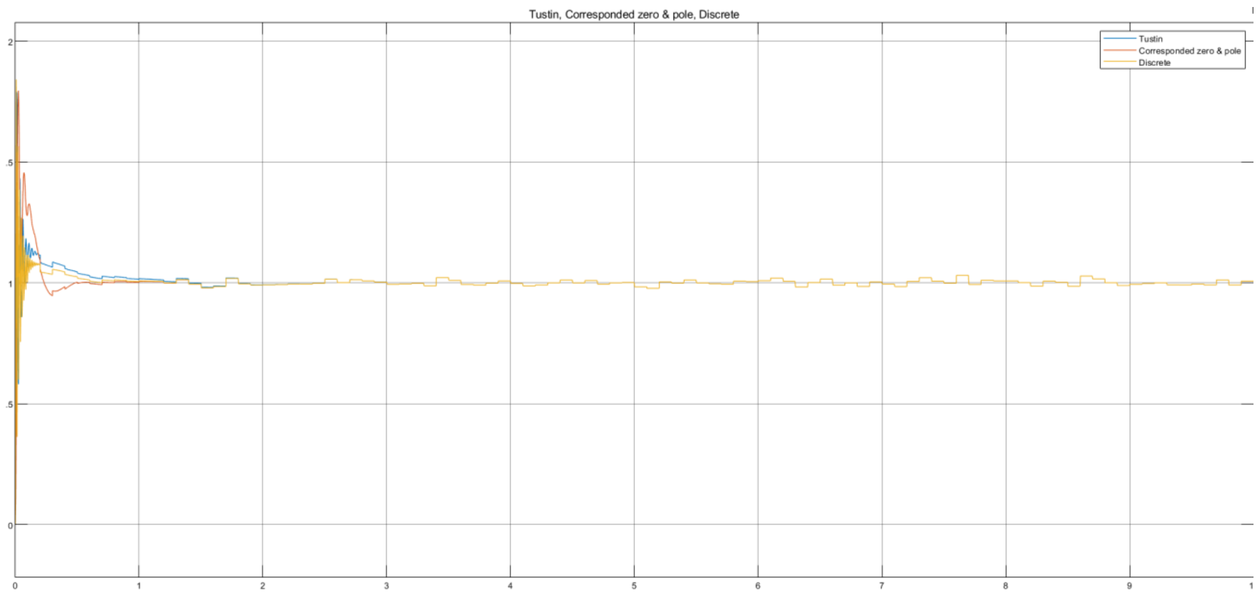
$$f_1 = 0.5032;\ f_2 = 1.6707;\ f_3 = -0.8384;\ f_4 = -0.3354$$



We succeed in controlling the system after 4 steps, at 0.0004 seconds (4*T1).

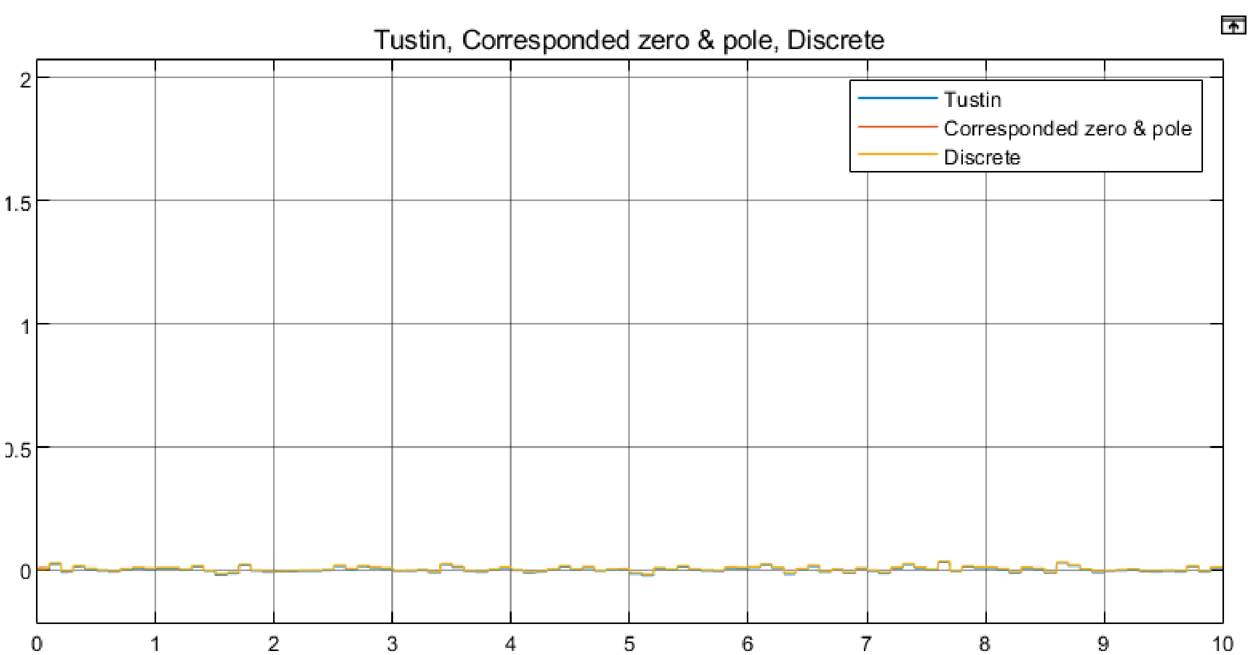
Since we use some approximations in the process of designing the deadbeat, implementing it in the non-linear system isn't suitable.

8)



All three controllers effectively manage the white noise and produce equal outputs. However, the controller with the "corresponding zero and pole" method can control the noise more quickly than the other two.

For non-linear system: (input gain = 2e-3)



All three controllers have the same attitude when confronting the white noise.

9) To convert the continuous-time state-space model to a discrete-time state-space model, we need to use a suitable sampling rate. Let's denote the sampling time by T=0.0001. The discrete-time state-space equations are given by:

$$x[k+1] = A_d x[k] + B_d u[k]; y[k] = C_d x[k] + D_d u[k]$$

By using a Matlab code we have:

```
A_d = 3x3
      2.9914   -2.9827    0.9913
      1.0000         0         0
           0    1.0000         0

B_d = 3x1
       1
       0
       0

C_d = 1x3
10^-6  x
      0.0362    0.1444    0.0360

D_d = 0
```

The system is observable and controllable since both observability and controllability matrices are full rank.

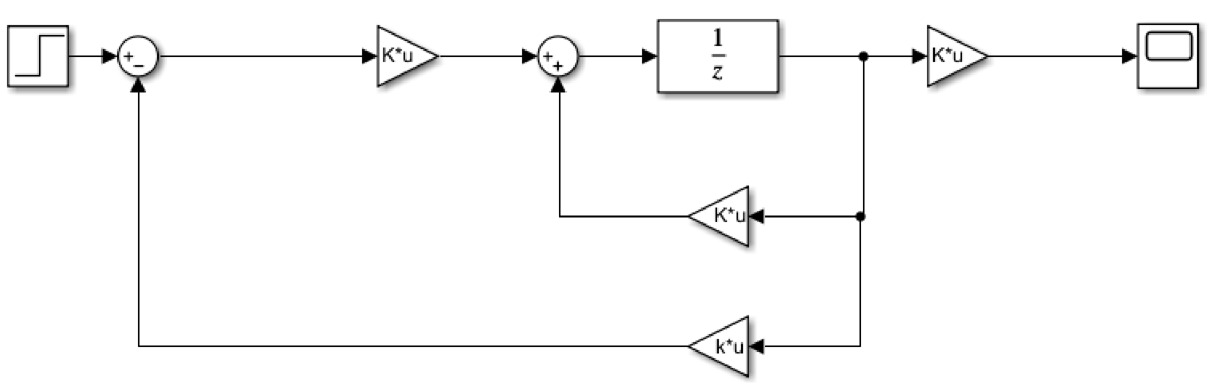
10) For a deadbeat controller, we should put our desired poles in the origin.

$$k = (\hat{a} - a)T^{-1} = -aT^{-1}$$

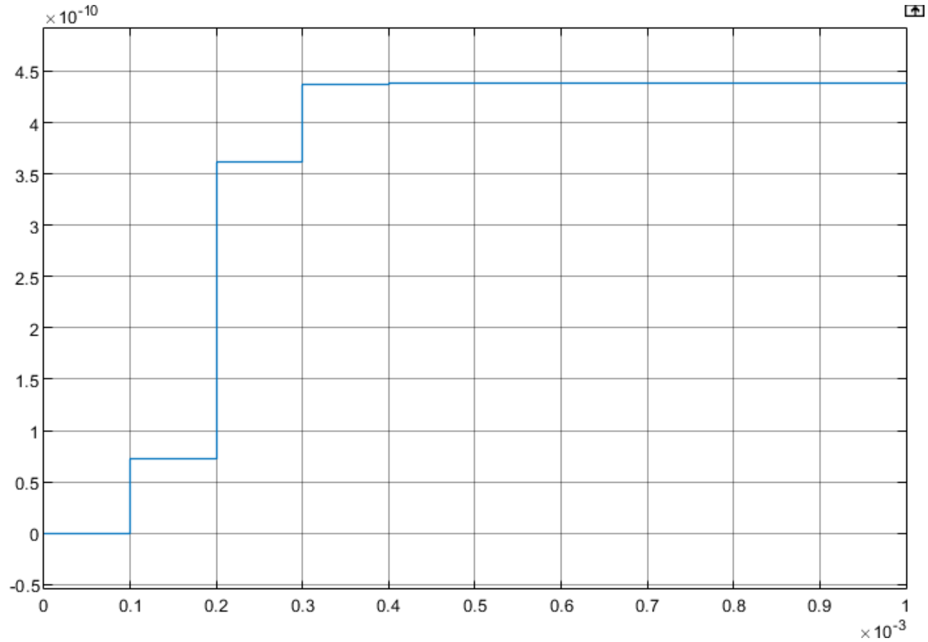
$$|\lambda I - G| = z^3 - 2.991z^2 + 2.983z - 0.9913 \rightarrow a = (-0.9913 \quad 2.983 \quad -2.991)$$

$$T = MW = (H \quad GH \quad G^2H)\begin{pmatrix} 2.983 & -2.991 & 1 \\ -2.991 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

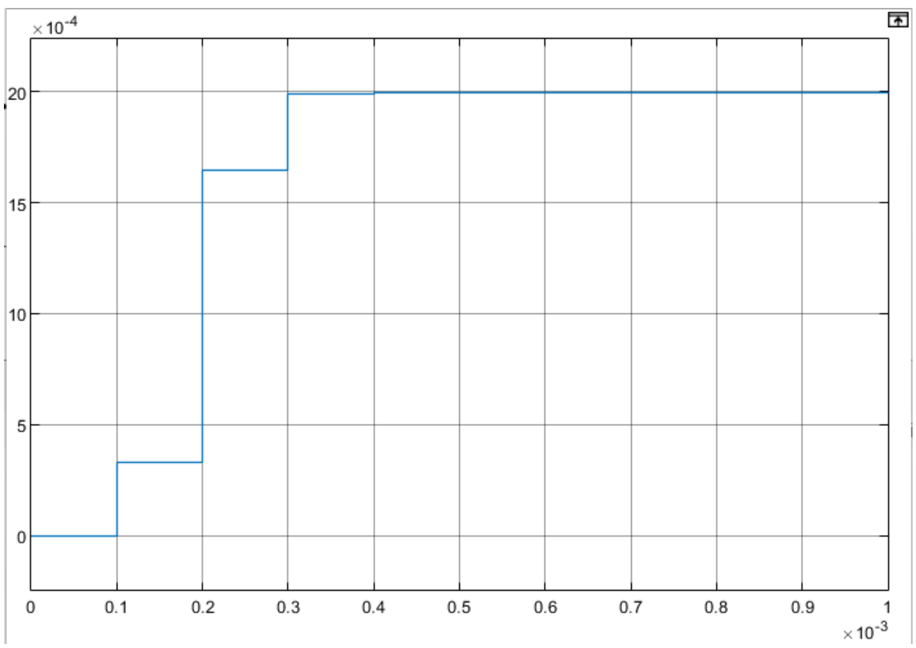
But we can use the place function in Matlab for this. But since we can’t have identical poles in place function, we set 0, 0.001, and 0.01 poles to design our controller.



For the step input, we have:



We can see the state feedback system change our gain. So we should put an appropriate gain to solve this issue:



The system reached the stable point in 4 steps.

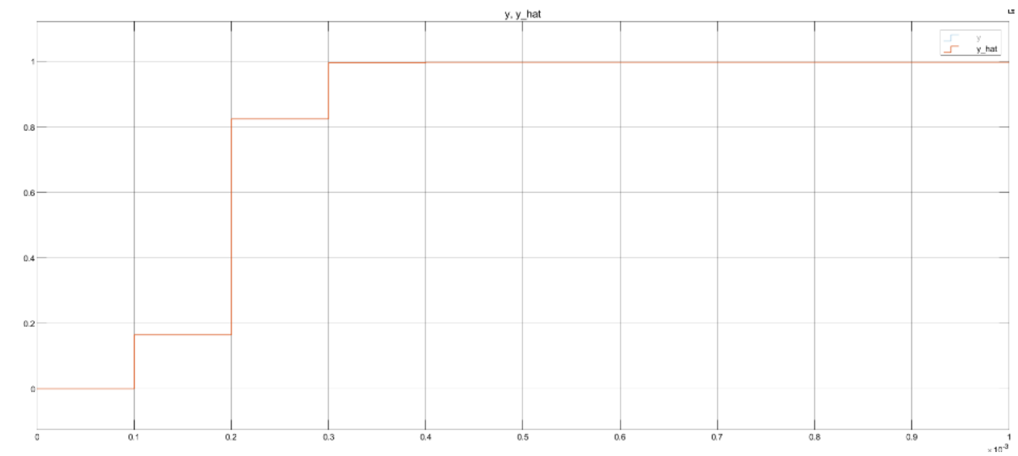
11)

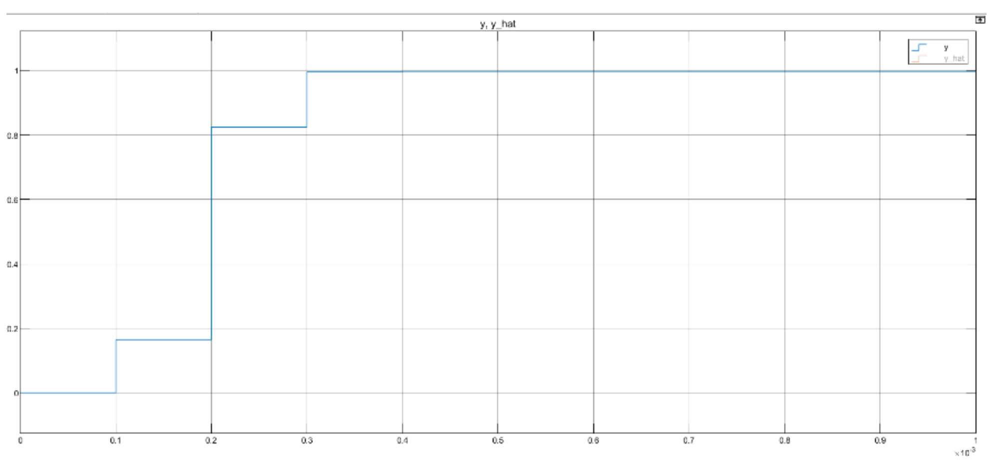
We start with a discrete-time linear system represented by the state-space matrices A_d and C_d. The Ackermann's method is employed to place the observer poles at desired locations. The chosen poles for the observer are: −0.6,0.5,−0.1

The polynomial associated with these desired poles is (z+0.6)(z−0.5)(z+0.1)

From these poles, we derive the desired polynomial coefficients using the poly function and these coefficients are then used to compute the observer gain L.

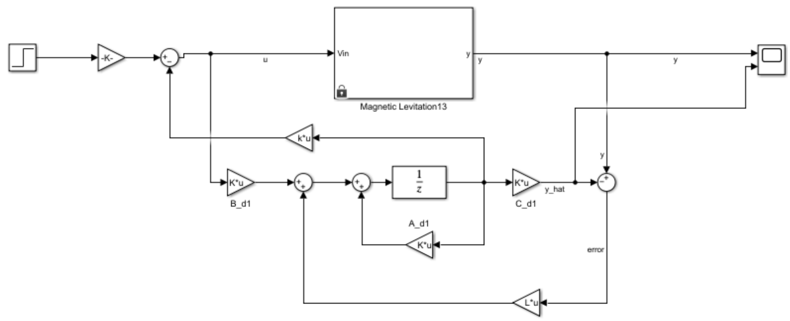
We can see our designed observer works well and y and y_hat are the same in the linearized system.





But for the actual system, while the state estimate y_hat was not stable, the output y approached and stayed at a small value around 5* 10^-3.

Schematic of the system is like this:



Y and y_hat of the actual system: