به نام خدا

دانشگاه تهران

سیستم های کنترل دیجیتال

پروژه نهایی

نام و نام خانوادگی	محمد مشرقی - محمد علی کیانی
شماره دانشجویی	810199492 - 810199480
تاریخ ارسال گزارش	

فهرست گزارش سوالات

1

در ابتدا سعی می کنیم نقاط تعادل سیستم را بیابیم:

* معادلات حالت سیستم

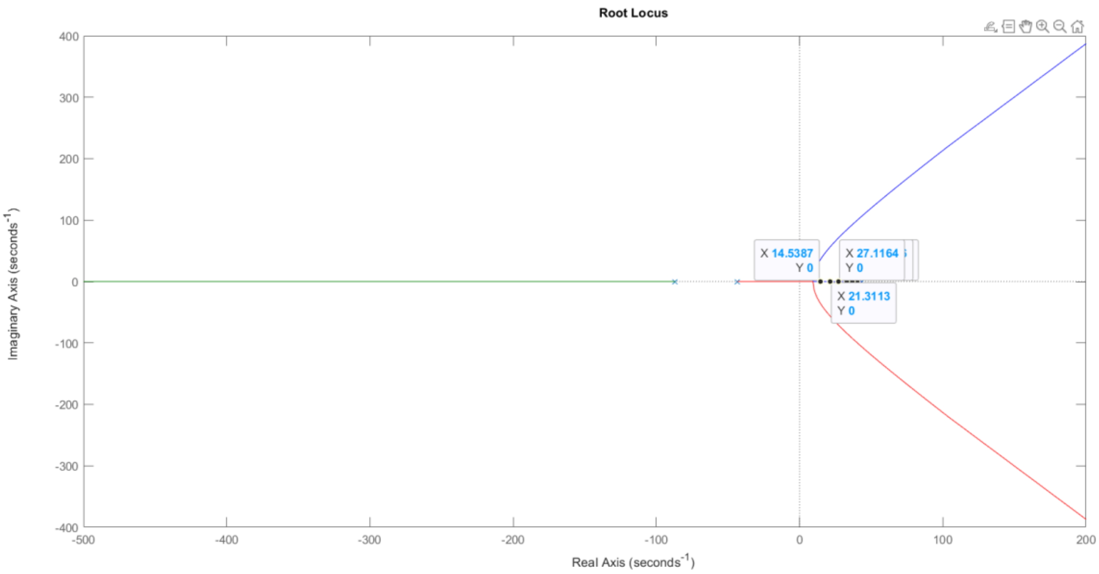
$$\begin{cases} \dot{x_1} = x_2 \\ \dot{x_2} = -g + \frac{c}{M}\frac{x_3^2}{0.0072 - x_1} \\ \dot{x_3} = \frac{1}{L}(-Rx_3 + u) \end{cases} \qquad \begin{cases} y = x_1 \end{cases}$$

تابع تبدیل:

```
                2.175e05
G(s) = ---------------------------------
        s^3 + 86.96 s^2 - 1887 s - 1.64e05
```

قطب های سیستم می شوند : 43.4343 , -86.9565 , -43.4343

مکان ریشه سیستم:



(1)

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -g + \dfrac{c}{m}\,\dfrac{x_3^2}{0.0072 - x_1} \\ \dot{x}_3 = \dfrac{1}{L}\,(-R x_3 + u) \\ y = x_1 \longrightarrow \hat{y} = 0.002 \end{cases}$$

$$\begin{cases} \dot{x}_1 = 0 \Rightarrow \boxed{x_2 = 0} \\ \dot{x}_3 = 0 \Rightarrow R x_3 = u \Rightarrow \boxed{x_3 = \dfrac{u}{R}} \\ \dot{x}_2 = 0 \Rightarrow \dfrac{mg}{c} = \dfrac{x_3^2 \to \frac{u^2}{R^2}}{0.0072 - x_1} \end{cases}$$

$$\Rightarrow \boxed{0.0072 - \frac{cu^2}{gmR^2} = x_1}$$

۱- نقطه تعادل $\Rightarrow$

$$\begin{cases} \hat{x}_1 = 0.0072 - \dfrac{cu^2}{gmR^2} = 0.002 \\ \hat{x}_2 = 0 \\ \hat{x}_3 = \dfrac{u}{R} = 7.84 \times 10^{-4} \end{cases}$$

$$\hat{x}_1 = \hat{y} = 0.002 = 0.0072 - \frac{cu^2}{gmR^2}$$

$$\Rightarrow \hat{u} = \sqrt{\frac{gmR^2}{c}(0.0052)} = 0.0078$$

(2)

خطی سازی :

$$A = \begin{bmatrix} 0 & 1 & 0 \\ \frac{\partial f_2}{\partial x_1} & 0 & \frac{\partial f_2}{\partial x_3} \\ 0 & 0 & \frac{-R}{L} \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{L} \end{bmatrix}$$

$$\frac{\partial f_2}{\partial x_1} = \frac{c}{m} \frac{x_3^2}{(0.0072 - x_1)^2}$$

$$\frac{\partial f_2}{\partial x_3} = \frac{c}{m} \frac{2x_3}{0.0072 - x_1}$$

$$A \cdot \begin{bmatrix} x_1 - \hat{x}_1 \\ x_2 - \hat{x}_2 \\ x_3 - \hat{x}_3 \end{bmatrix} = A \begin{bmatrix} x_2 \\ x_2 \\ x_3 \end{bmatrix} - A \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \end{bmatrix}$$

$$\dot{X} = AX - A\hat{X} + Bu$$

درنهایت به سه معادله زیر می رسیم :

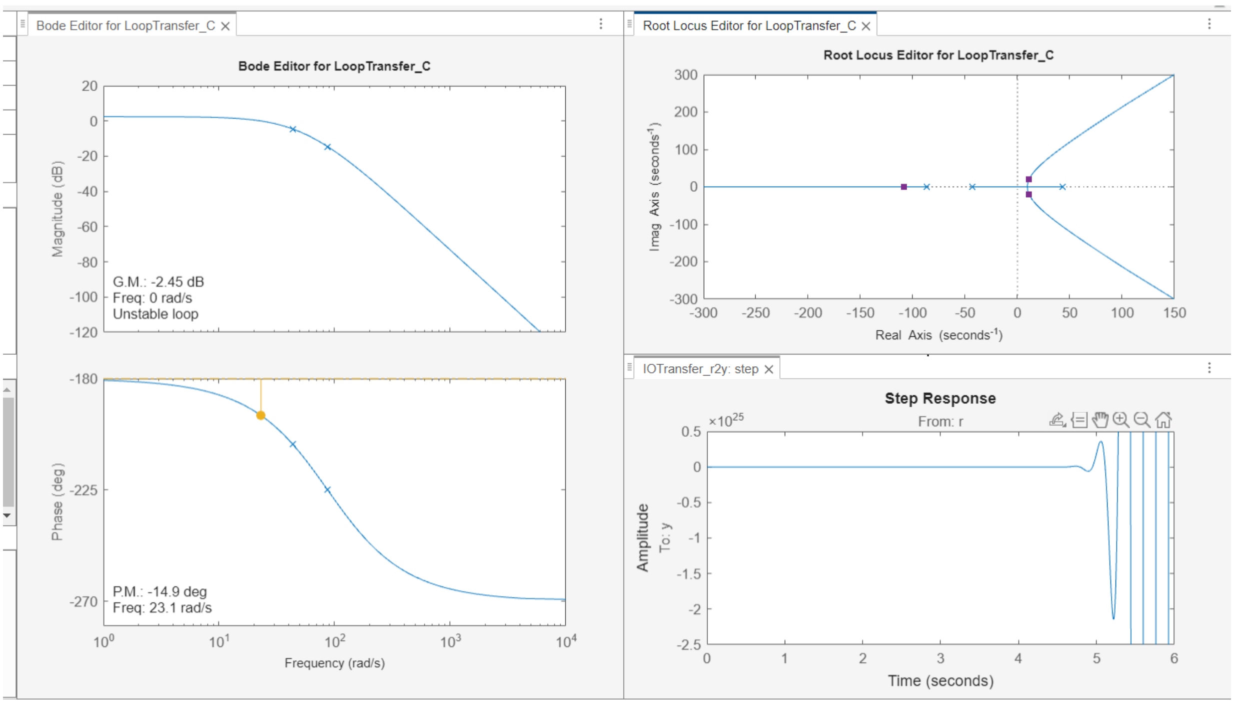
$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = \frac{g}{0.0052}(x_1 - 0.002) + \left(2\sqrt{\frac{gC}{m(0.0052)}}\right)\left(x_3 - \sqrt{\frac{mg(0.0052)}{c}}\right)$$

$$\dot{x}_3 = \frac{1}{L}(-R x_3 + u)$$

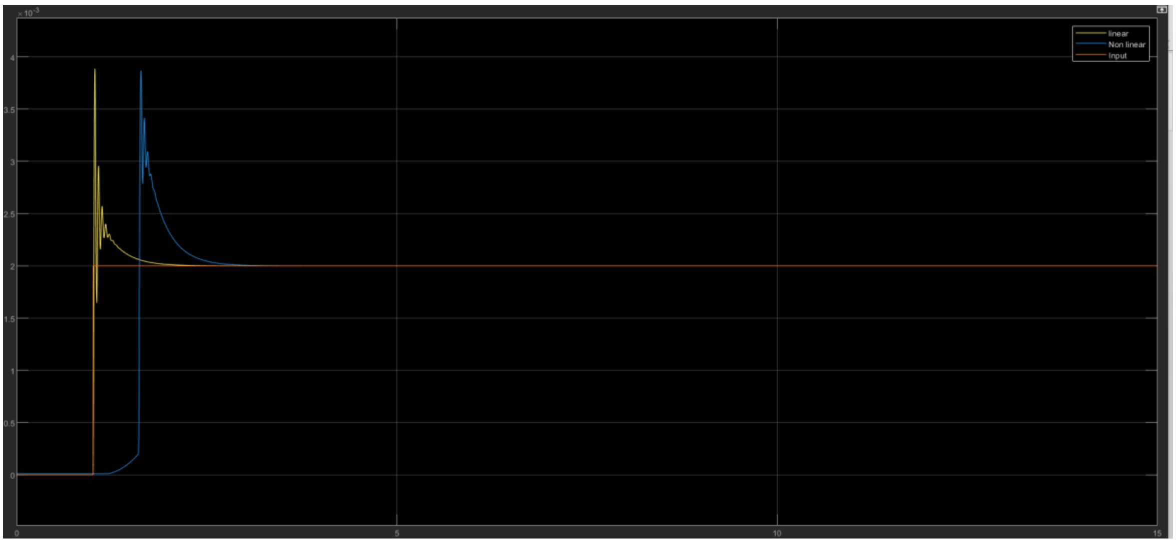
2

Main system(unstable):



Gc= 370(s+3)(s+33)/ s(s+3650)

The unlinear system takes a lot of time to converge but linear system convege as fast as it can.



System info of Gcpid with linear system:

```
ans = struct with fields:
          RiseTime: 0.0080
     TransientTime: 0.7058
      SettlingTime: 0.7058
       SettlingMin: 0.8419
```

```
       SettlingMax: 1.9358
         Overshoot: 93.5818
        Undershoot: 0
              Peak: 1.9358
          PeakTime: 0.0237
```

For unlinear system belike:

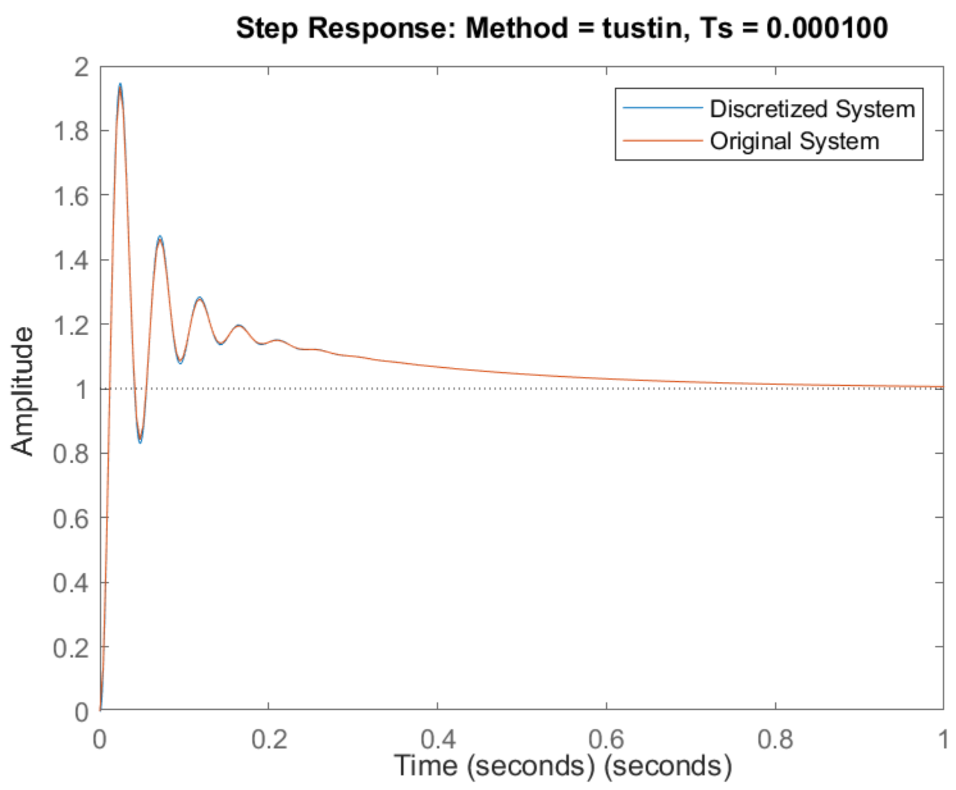
```
ans = struct with fields:
          RiseTime: 16.5573
     TransientTime: 256.2058
      SettlingTime: 256.1924
       SettlingMin: 0.0018
       SettlingMax: 0.0038
         Overshoot: 90.5889
        Undershoot: 0
              Peak: 0.0038
          PeakTime: 167
```

Which should devided with 1000

3

According to Matlab code:we caculate error for linear system as it follows:

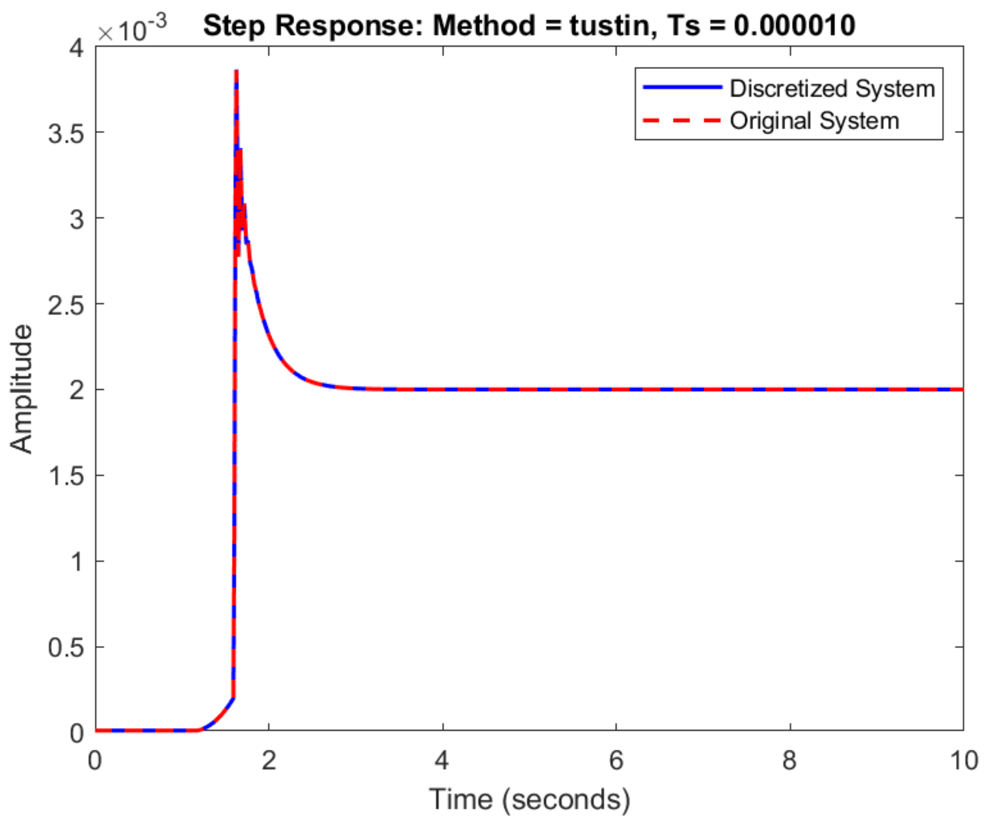
```
Method: matched, Ts: 0.000001, Error: 4796926946189790481334927 36.0000
Method: matched, Ts: 0.000010, Error: 27434.3974
Method: matched, Ts: 0.000100, Error: 9.1840
Method: matched, Ts: 0.001000, Error: 18.7841
Method: matched, Ts: 0.010000, Error: 6759966820938423484809216 0.0000
Method: tustin, Ts: 0.000001, Error: 4100876986198260711088455 68.0000
Method: tustin, Ts: 0.000010, Error: 31451.1403
Method: tustin, Ts: 0.000100, Error: 8.5147
Method: tustin, Ts: 0.001000, Error: 10.7554
Method: tustin, Ts: 0.010000, Error: 7714264555269668253938483 2.0000
Best fit: Method = tustin, Ts = 0.000100, Error = 8.5147
```



And for Unlinear system we have:

```
Method: matched, Ts: 0.000010, Error: 0.0612
Method: matched, Ts: 0.000100, Error: 0.0658
Method: matched, Ts: 0.001000, Error: 0.1447
Method: matched, Ts: 0.010000, Error: 5.2725
Method: tustin, Ts: 0.000010, Error: 0.0608
Method: tustin, Ts: 0.000100, Error: 0.0619
Method: tustin, Ts: 0.001000, Error: 0.0931
Method: tustin, Ts: 0.010000, Error: 5.4462
```

```
Best fit nonlinear: Method = tustin, Ts = 0.000010, Error = 0.0608
```

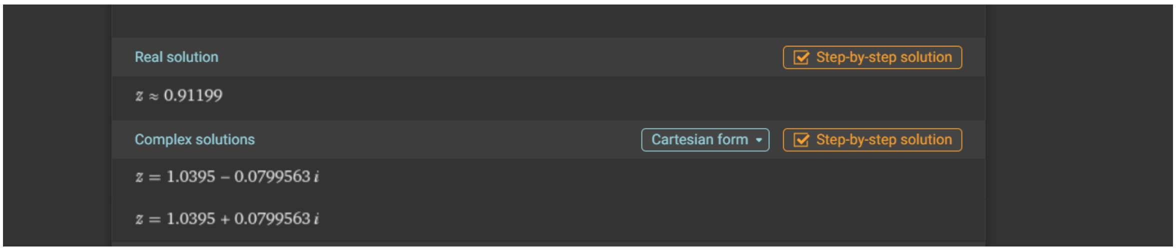


4

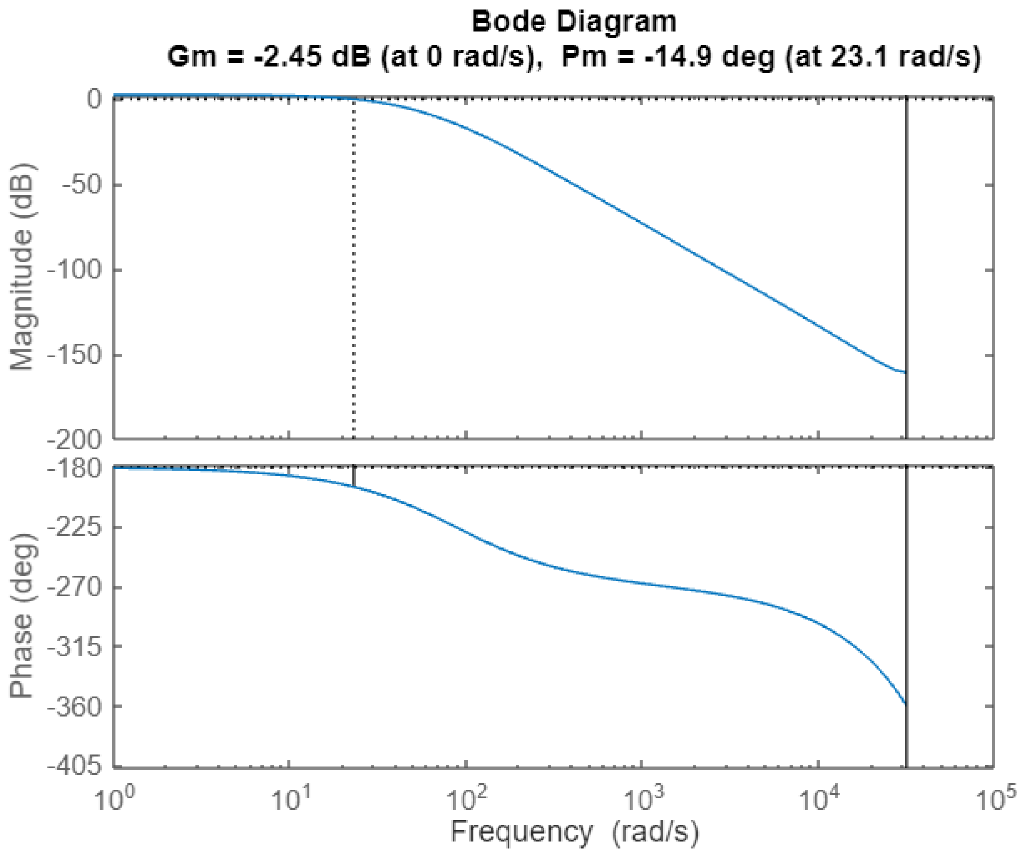
The system in discrete time:

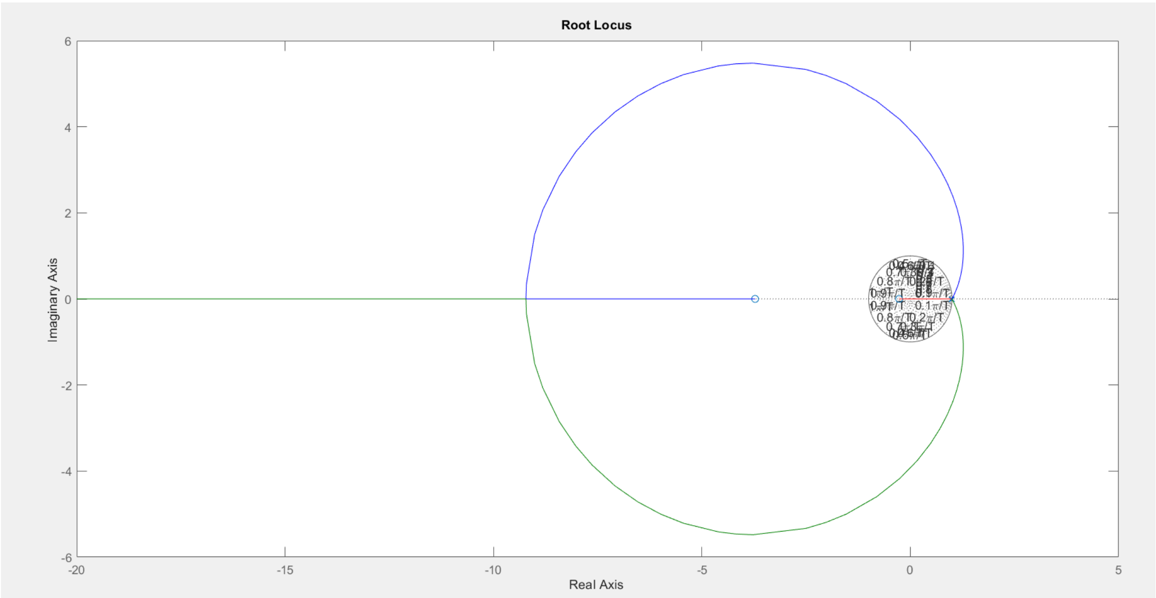
```
G_z =

  3.618e-08 z^2 + 1.444e-07 z + 3.602e-08
  ---------------------------------------
    z^3 - 2.991 z^2 + 2.983 z - 0.9913

Sample time: 0.0001 seconds
Discrete-time transfer function.
Model Properties


poles_G_z = 3×1
    0.9957
    1.0044
    0.9913
zeros_G_z = 2×1
   -3.7240
   -0.2674
```

We also got different answer for loc of poles in the system :

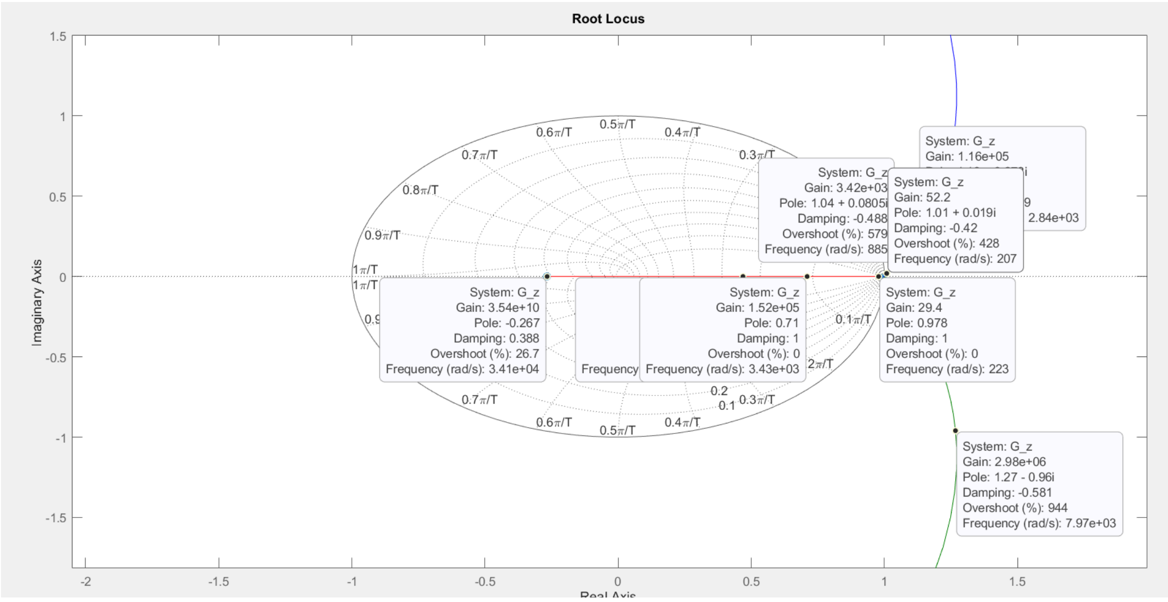


Gm = 0.75 in Wcg = 0hz , , Pm = -15.2 in Wcp = 23.0 , fb = 25

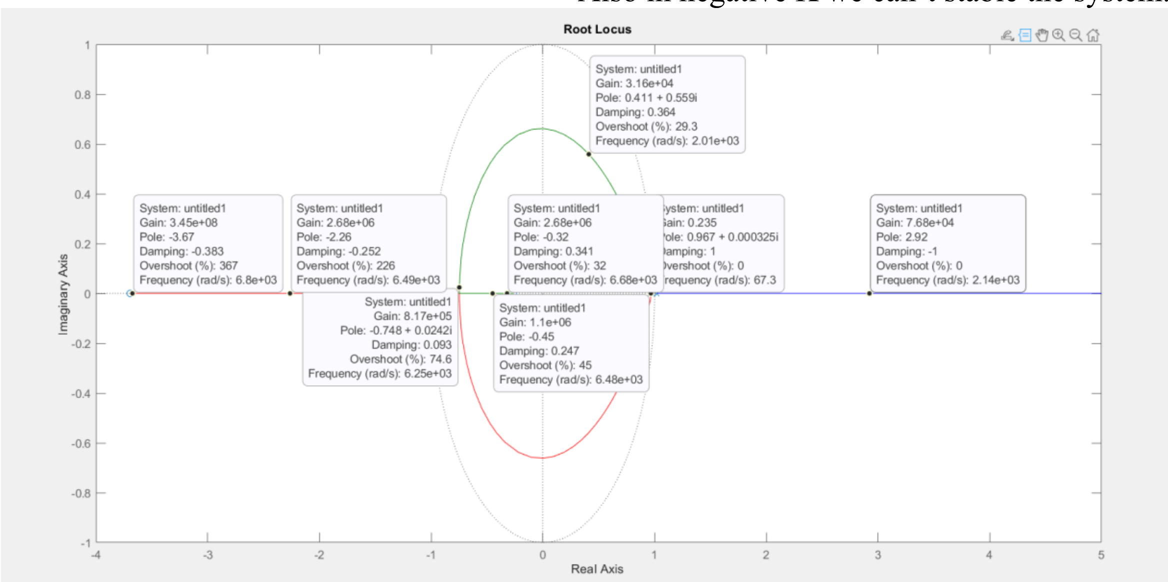

Bode Diagram
Gm = -2.45 dB (at 0 rad/s), Pm = -14.9 deg (at 23.1 rad/s)
Magnitude (dB)
Phase (deg)
Frequency (rad/s)


Root Locus
Imaginary Axis
Real Axis

With each k the system is unstable and we can’t stable it with changing the value of k:



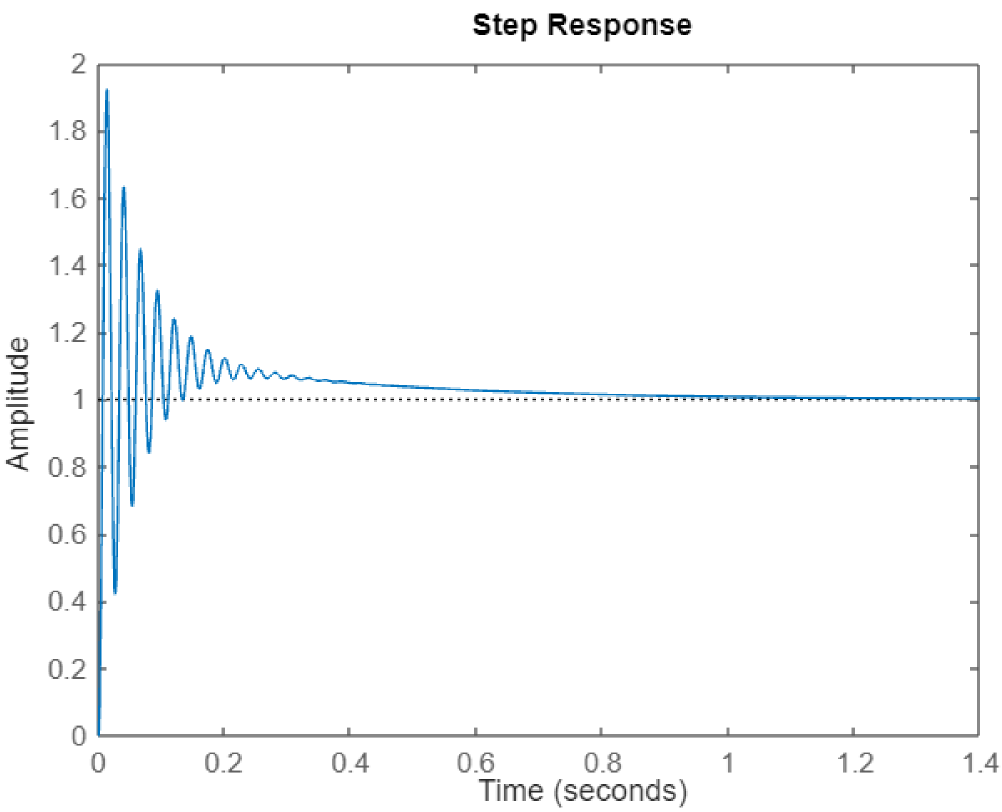
Also in negative K we can’t stable the system.



With using ControlSystemDesigner, we try to define a new PID controller in discrete time that system have the same respond like part 2:

```
Gc_z_designed =

  550 z^2 - 1093 z + 543.4
  ------------------------
          z^2 - z

Sample time: 0.0005 seconds
Discrete-time transfer function.
Model Properties
```

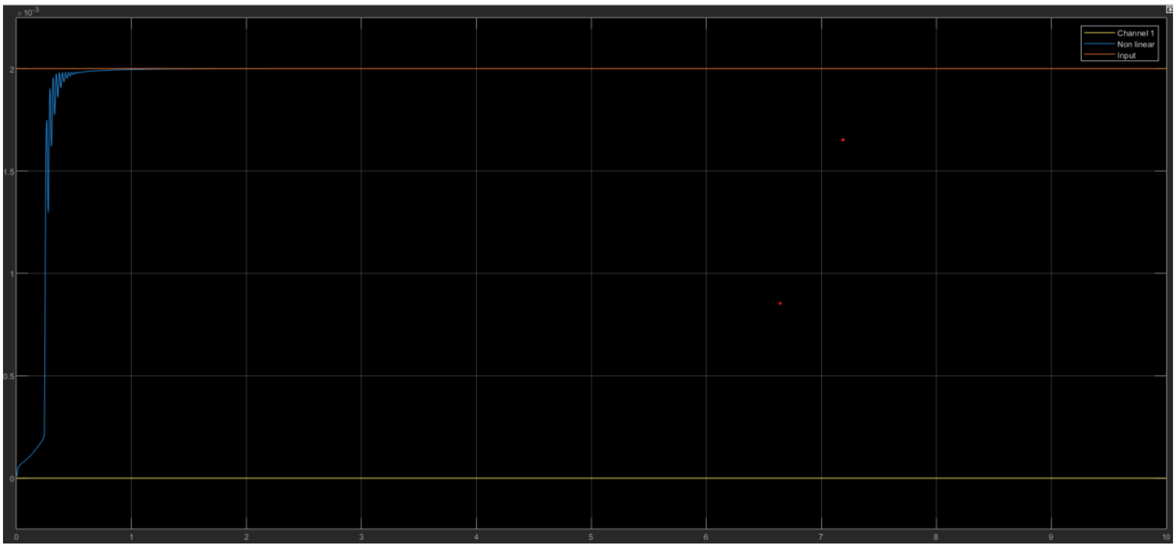
Linear system:



Stepinfo:

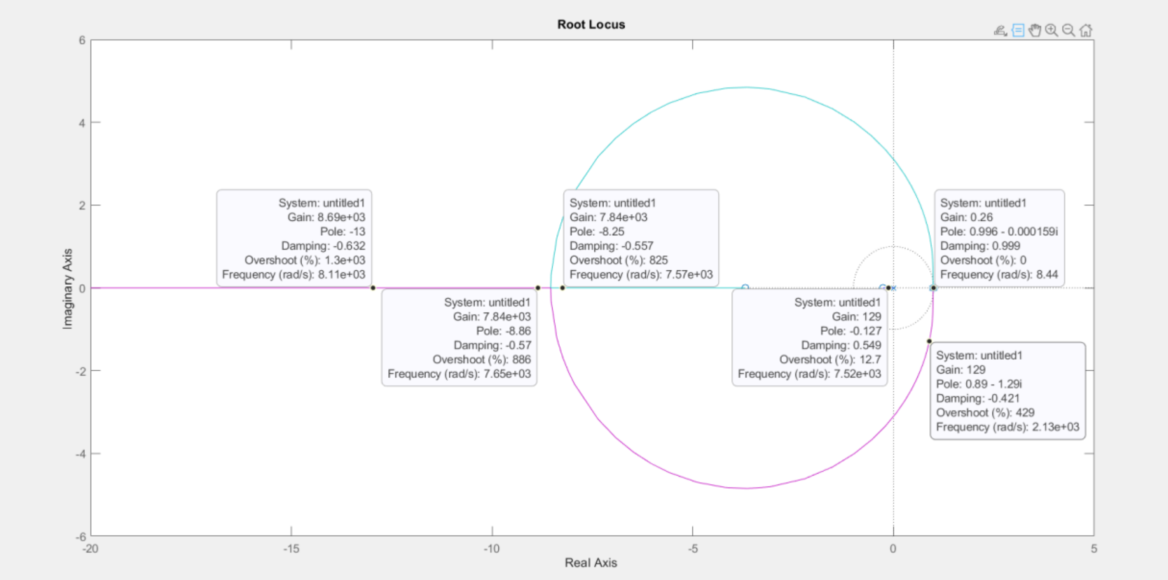
```
ans = struct with fields:
         RiseTime: 0.0045
    TransientTime: 0.7025
     SettlingTime: 0.7025
      SettlingMin: 0.4210
      SettlingMax: 1.9227
        Overshoot: 92.2683
       Undershoot: 0
             Peak: 1.9227
         PeakTime: 0.0135
```

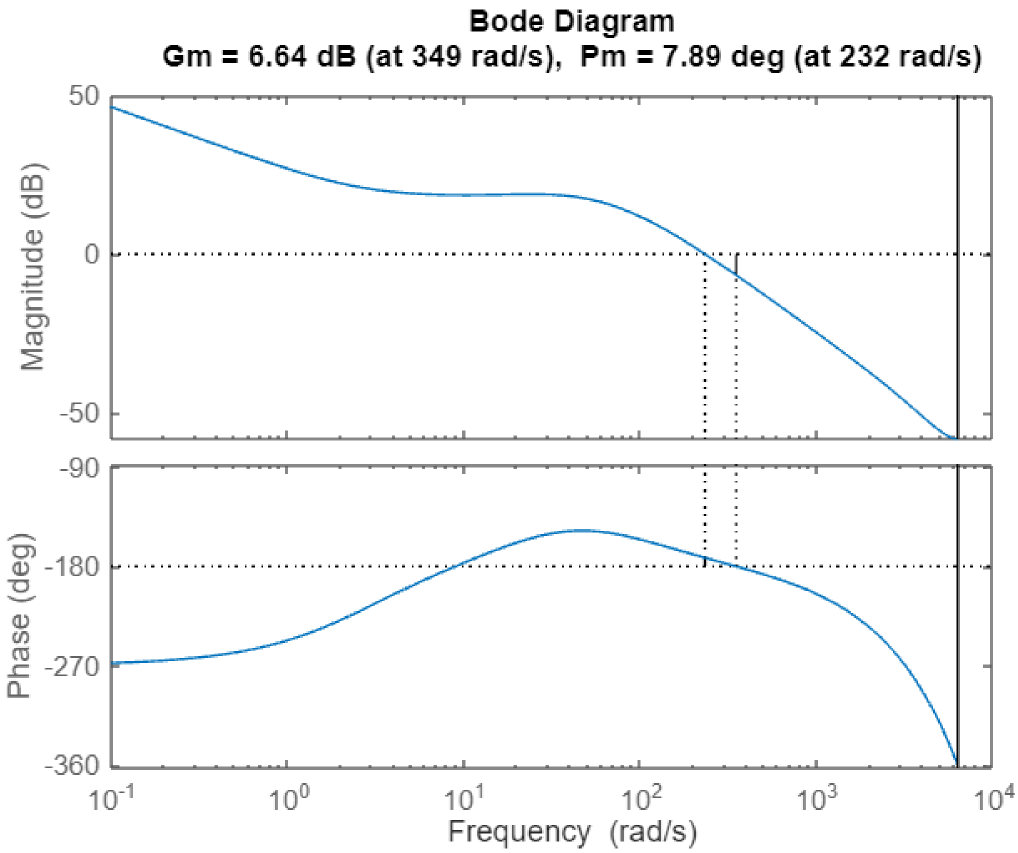
Unlinear System:



```
ans = struct with fields:
         RiseTime: 94.6917
    TransientTime: 2.8255e+03
     SettlingTime: 2.8253e+03
      SettlingMin: 0.0016
      SettlingMax: 0.0020
        Overshoot: 1.3987e-10
       Undershoot: 0
             Peak: 0.0020
         PeakTime: 19851
```

Which needs to divide by 2000


Root Locus
System: untitled1
Gain: 8.69e+03
Pole: -13
Damping: -0.632
Overshoot (%): 1.3e+03
Frequency (rad/s): 8.11e+03
System: untitled1
Gain: 7.84e+03
Pole: -8.25
Damping: -0.557
Overshoot (%): 825
Frequency (rad/s): 7.57e+03
System: untitled1
Gain: 0.26
Pole: 0.996 - 0.000159i
Damping: 0.999
Overshoot (%): 0
Frequency (rad/s): 8.44
System: untitled1
Gain: 7.84e+03
Pole: -8.86
Damping: -0.57
Overshoot (%): 886
Frequency (rad/s): 7.65e+03
System: untitled1
Gain: 129
Pole: -0.127
Damping: 0.549
Overshoot (%): 12.7
Frequency (rad/s): 7.52e+03
System: untitled1
Gain: 129
Pole: 0.89 - 1.29i
Damping: -0.421
Overshoot (%): 429
Frequency (rad/s): 2.13e+03
Imaginary Axis
Real Axis



As you can see we have improvement in Gm from -2.45 become 6.64 and Wcg changed from 0 to 349 also Pm changed from -15.2 to 7.9 and Wcp become 232.

Also in previous section we couldn't stable system with changing k, but in this system we can change k.

7

$\alpha = 1.044$ $\beta = 0.652$

rule 1: $\underline{\sum f_i = 1}$

rule 2: $F(z) = (1 - \alpha z^{-1})\, Q(z)$

rule 3: $1 - F(z) = (1 + \beta)\, z^{-1})\, P(z)$

rule 4: we start we with set step

$1 - F(z) = (1 - \alpha z^{-1})\, Q'(z)$

$$(1-\alpha z^{-1})(q_1 z^{-1} + q_2 z^{-1} + q_3 z^{-1}) =$$

$$q_1 z^{-1} + (q_2 - \alpha q_1) z^{-2} + (q_3 - \alpha q_2) z^{-3} + (\alpha q_3) z^{-4}$$

$$\Rightarrow \boxed{\alpha^3 f_1 + \alpha^2 f_2 + \alpha f_1 z_7 + f_4 = 0} \quad (1)$$

$$(1 - f_1 z^{-1} - f_2 z^{-2} - f_3 z^{-3} - f_4 z^{-4})$$

$$= (1+\beta z^{-1})(1 + p_1 z^{-1} + p_2 z^{-2} + p_3 z^{-3})$$

$$\Rightarrow 1 + (p_1+\beta) z^{-1} + (p_2 + \beta p_1) z^{-2} + (p_3 + \beta p_2) z^{-3} + \beta p_3 z^{-4}$$

$$\Rightarrow \boxed{\beta^3 F_1 + \beta^2 F_2 + \beta f_1 + f_4 - \beta^4 = 0} \quad (2)$$

$$\sum f_i = q \Rightarrow \boxed{f_1 + f_2 + f_3 + f_4 = 1} \quad (3)$$

2

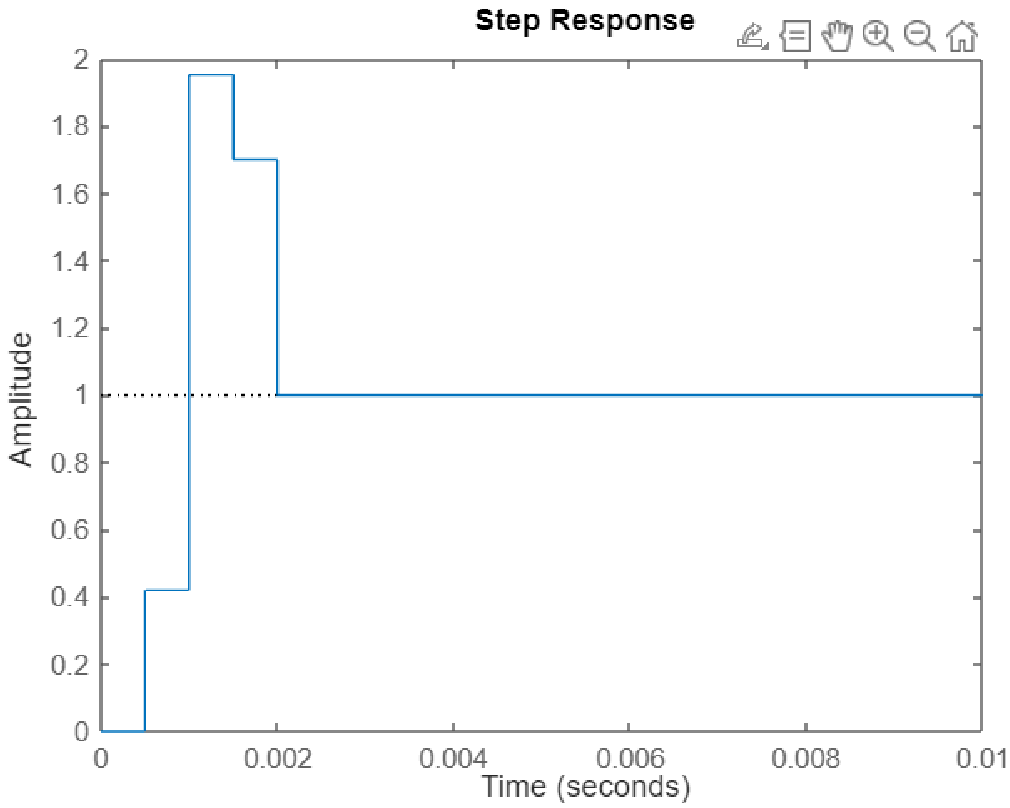
در اینجا که 4 مجهول و سه تا معادله داریم و یک درجه آزادی

داریم برای همین ~~$f_4 = 0.9$~~ ~~f_4~~ رو فرض می کنیم

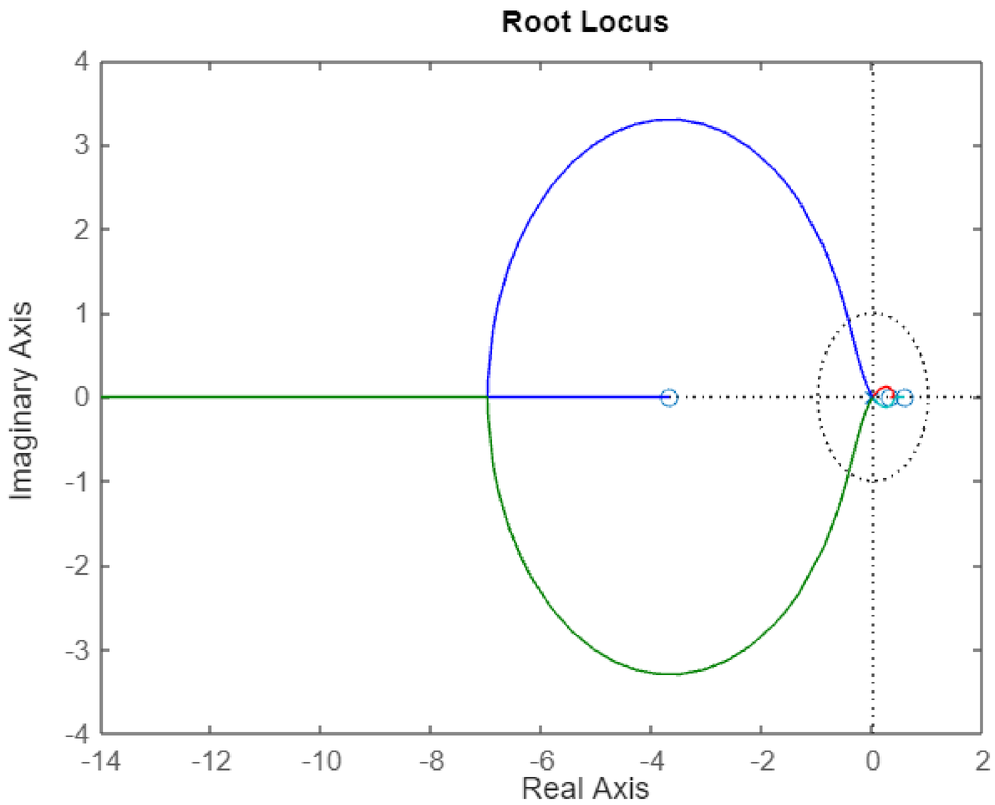
$\boxed{f_4 2+0{,}5}$

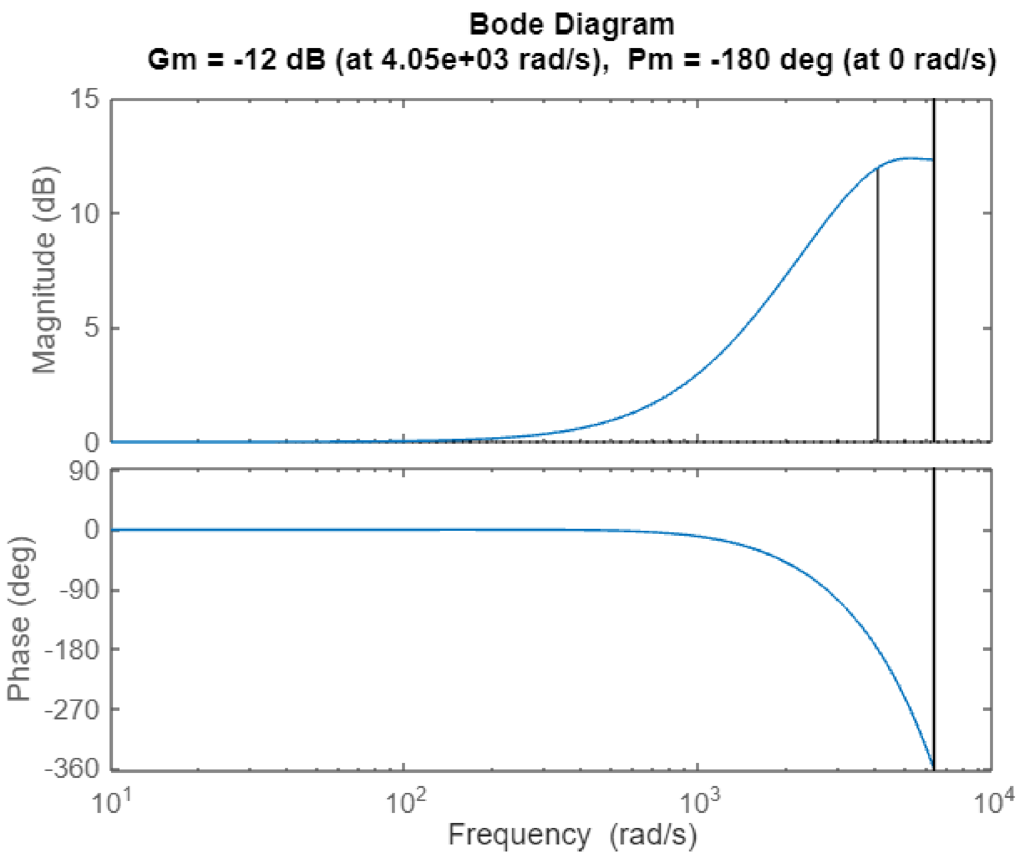
با حل معادله داریم

In linear system as u can see with 4 steps we converge to the final amount.

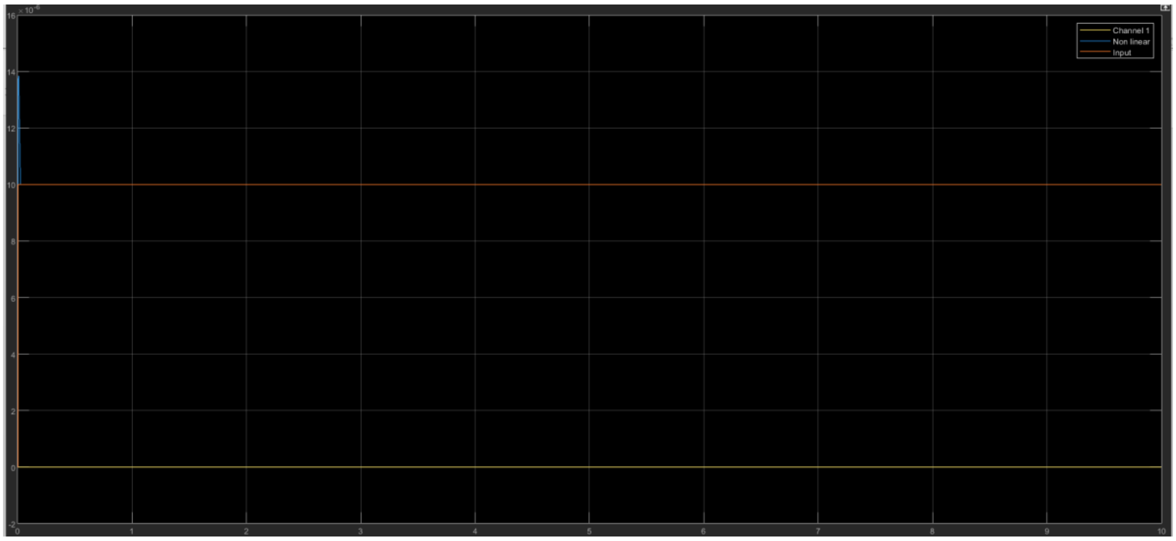



```
ans = struct with fields:
         RiseTime: 5.0000e-04
    TransientTime: 0.0020
     SettlingTime: 0.0020
      SettlingMin: 1.0000
      SettlingMax: 1.9524
        Overshoot: 95.2421
       Undershoot: 0
             Peak: 1.9524
         PeakTime: 1.0000e-03
```





Also let's take look to nonlinear system:



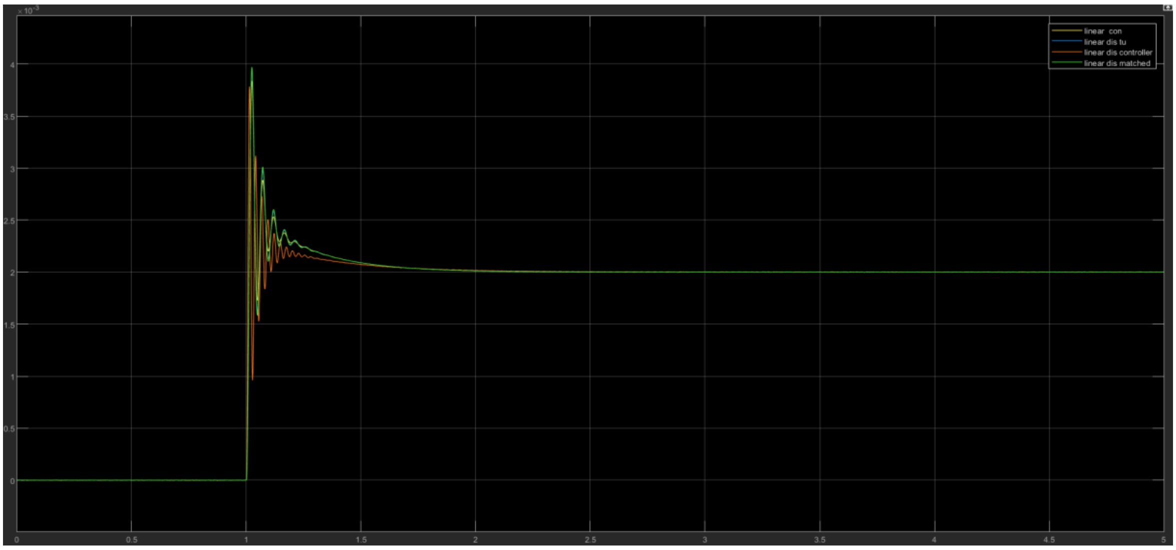

```
ans = struct with fields:
         RiseTime: 0
    TransientTime: 298.0374
     SettlingTime: 293.5705
      SettlingMin: 1.0000e-05
      SettlingMax: 1.4328e-05
        Overshoot: 43.2813
       Undershoot: 0
             Peak: 1.4328e-05
         PeakTime: 104
```

Also it needs to divided by 2000

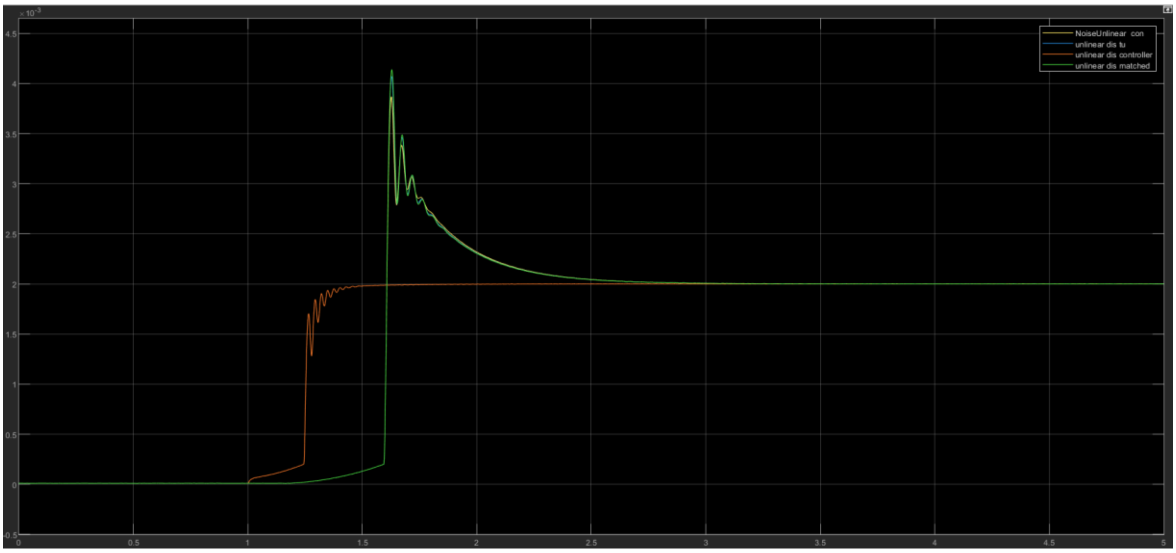
So here's we add some noise, to see what happen to are system,

It depend on when we noice power and sample per time.
with out setting in matlab all system work well but if change the noice the discrete signals will be unstable.

Linear system:



Unlinear system:



9

At first we try to caculate H, G and after that check that our matrix is controlable or observable on not.

Q9

```
z = tf("z" , best_fit_Ts);


G = expm(A*best_fit_Ts)
syms x
H = double(int(expm(A*x)*B, x, [0 best_fit_Ts]))

rank(ctrb(G,H))

rank(obsv(G,C))
```

```
G = 3×3
    1.0002    0.0005    0.0031
    0.9433    1.0002   12.2416
         0         0    0.9575

H = 3×1
    0.0000
    0.0268
    0.0043

ans = 3

ans = 3
```

The system is controllable and observable for Calculate the feedback gain we use formalua

$$T = \underbrace{\left[H \;\middle|\; GH \;\middle|\; \cdots \;\middle|\; G^{n-1}H \right]}_{M} \underbrace{\begin{bmatrix} a_{n-1} & a_{n-2} & \cdots & a_1 & 1 \\ a_{n-2} & \vdots & & 1 & \\ \vdots & \vdots & 1 & & \\ a_1 & 1 & & & 0 \\ 1 & & & & \end{bmatrix}}_{W}$$

$$T = M \times W$$

حتماً معکوس پذیر است

Q10

```
% Extract the Denominator values from the system G and store i
a = poly(G);
a = a(2:end);

% Construct matrix W using elements of 'a'
W = [a(2) a(1) 1;
     a(1) 1 0;
     1 0 0];
Tinv = inv(ctrb(G, H) * W);
```

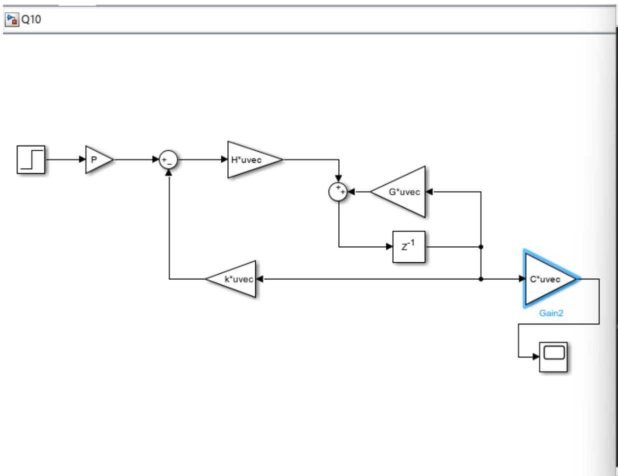
$$u[k] = -(\hat{a} - a)\, T^{-1} x[k]$$

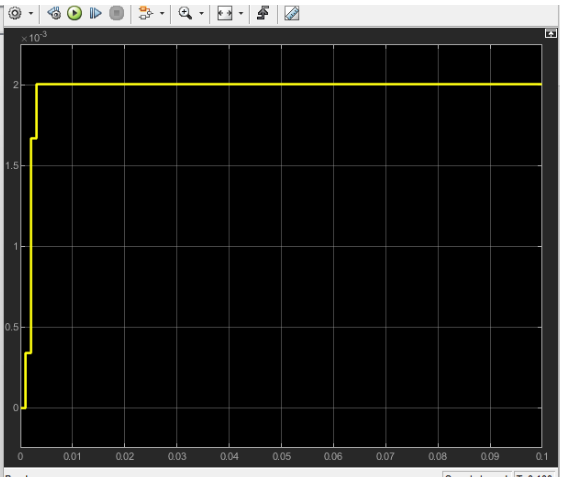
```
% Initialize alpha for Dead beat control design
alpha = zeros(1, 3);

% Calculate the feedback gain 'k'
k = flip(alpha - a) * Tinv
P = inv(C*inv(eye(3) - (G-H*k))*H);
```

k = 1×3

$10^4 \times$

3.7611   0.0038   0.0419





	Q11

for Calculate the observer gain we use c’ and g’ as new system and find controller

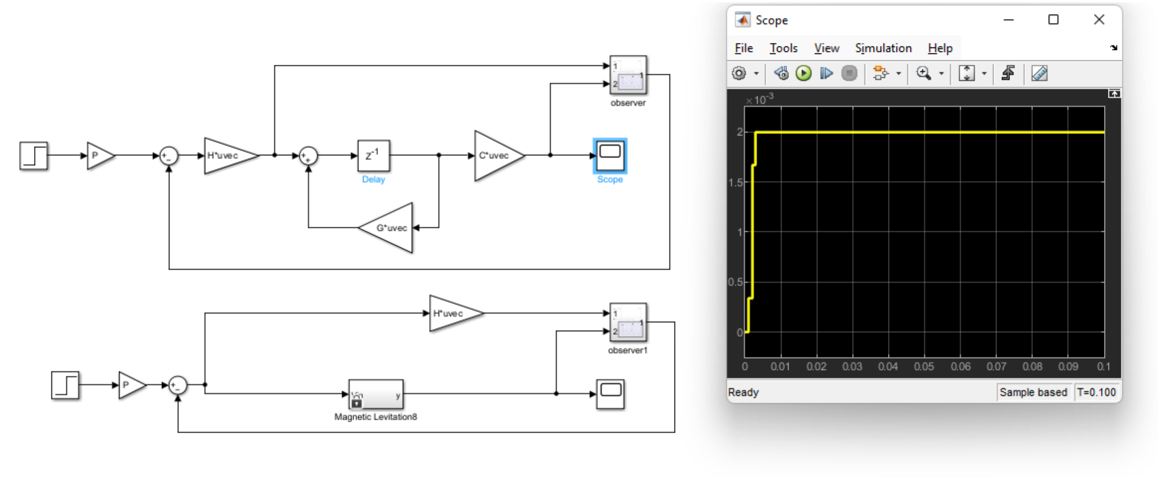
Q11

```
G_prime = G.';
H_prime = C.';
Tinv = inv(ctrb(G_prime, H_prime)*W);

alpha = [0, 0.1, 0.01];
L = flip(alpha - a)*Tinv
```

L = 1×3

10^3 ×

0.0030   4.9380   0.1642



we can control linear system

but we cant control non linear system