Spring Break Project

Table of Contents

Daniel Lopez Villa - - dlopezvilla99@gmail.com - - Stabilization of a Mobile Inverted Pendulum

MiP Hardware

```
clear; close all; clc;
Vb = 7.4; %V Battery Voltage
wf = 1643.5; %rad/s motor shaft free speed
sbar = .0036021; %Nm motor stall torque
G = 143; %gearRatio
Im = 6.5085e-8; %kg*m^2 motor inertia
r = 50e-3; %m wheel radius
mw = 40e-3; %kg wheel mass
mb = 276e-3; %kg body mass
l = 50e-3; %m length from wheel axis to body centroid
Ib =  0.0016; %kg*m^2 body inertia
g= 9.81; % gravity
k = 2.1918e-06;%  motor torque constant
Iw = 2*((mw*r^2)/2 + (Im * G^2)); % wheel inertia

sampleRate1 = 100;
sampleRate2 = 10;
```

Linearization Coeficients

```
A = (Iw + (mw+mb)*r^2);
B = mb*r*l;
C = (Ib + mb*l^2);
D = mb*g*l;
E = 2*G*sbar;
F = 2*G^2*k;

J1 = B - (A*(B+C))/(A+B);
J2 = F + (F*(B+C))/(A+B);
J3 = (A*D)/(A+B);
J4 = (D*F)/(A+B);

R1 = (A+B);
R2 = (B+C);
```

```
R3 = D;

s = tf('s'); %% s-domain variable

Theta = (E/J1)*s; %%define theta

U = s^3 - s^2*(J2/J1) + s*(J3/J1) + (J4/J1);%%define U

Phi = R3/R1 - s^2*(R2/R1); %%define Phi

G1 = minreal(Theta/U); %%define G1
G2 = minreal(Phi/s^2); %%define G2
```

Inner Loop Body Angle Control

```
G1poles = pole(G1);
G1zeros = zero(G1);

%lead Controller z < p
z1lead = 5;
p1lead = 70;

%lag Controller z > p
z1lag = 55;
p1lag = 0;

K1 = -6; %%negative gain
D1lead =(s+z1lead)/(s+p1lead);
D1lag =(s+z1lag)/(s+p1lag);

%include pade approximation to account for DAC h/2 delay
d1 = (1/sampleRate1)/2;
padeApprox1 = (1-(d1*s/2) + ((d1*s)^2)/12)/(1+(d1*s/2) +
 ((d1*s)^2)/12);

D1 = K1 * D1lead * D1lag *padeApprox1;

L1 = minreal(G1*D1);
[GM_L1,PM_L1,WCg_L1,WCp_L1] = margin(L1);

[y1,t1] = step((L1)/(1+L1));

%calculate prescalar P from last 25 terms
P1 = 1/mean(y1(end-25 : end));
T1 = (G1*D1)/(1+G1*D1);
```

Outer Loop Wheel Position Control

```
G2poles = pole(G2);
G2zeros = zero(G2);

%lead Controller z < p
z2lead = .3;
```

```
p2lead = 7;

%lag Controller z < p
z2lag = .4;
p2lag = 0;

K2 = .5; %% gain
D2lead = (s+z2lead)/(s+p2lead);
D2lag =(s+z2lag)/(s+p2lag);

d2 = (1/sampleRate2)/2;
padeApprox2 = (1-(d2*s/2)+ ((d2*s)^2)/12)/(1+(d2*s/2)+ ((d2*s)^2)/12);

D2 = K2 * D2lead * D2lag *padeApprox2;

L2 = minreal(G2*D2);

[GM_L2,PM_L2,WCg_L2,WCp_L2] = margin(L2);

T2 = minreal((G2*D2)/(1+G2*D2));
```

Successive Loop Closure

```
L3 = minreal(D2*T1*G2);

Tsystem = minreal((D2*P1*T1*G2)/(1 + D2*P1*T1*G2));
[GM_Ts,PM_Ts,WCg_Ts,WCp_Ts] = margin(Tsystem);
```

Discrete Time Controller

```
h1 = 1 / sampleRate1;

% Convert to discrete time using tustin approximation with prewarping
%perwarp frequency designed around crossover frequency
opt1 = c2dOptions('Method','tustin','PrewarpFrequency',abs(WCp_L1));
Gz1 = c2d(G1,h1,'zoh');
Dz1 = c2d(D1,h1,opt1);
Tz1 = (Dz1*Gz1)/(1 + Dz1*Gz1);

%calculate discrete prescalar using last 5 terms
[yz1,tz1] = step(Tz1);
dz1Lim = find(tz1 >= .5);
Pz1 = 1/mean(yz1(dz1Lim(1)-5 : dz1Lim(1)));


h2 = 1/sampleRate2;

opt2 = c2dOptions('Method','tustin','PrewarpFrequency',abs(WCp_L2));
Gz2 = c2d(G2,h2,'zoh');
Dz2 = c2d(D2,h2,opt2);
Tz2 = (Dz2*Gz2)/(1 + Dz2*Gz2);

TzSystem = c2d(Tsystem,h2,'zoh');
```

Display Results

```
G1
G2
D1
P1
D2
Dz1
Pz1
Dz2
T1
T2
Tz1
Tz2
Tsystem


G1 =

              -570.7 s
  --------------------------------
  s^3 + 84.54 s^2 - 62.79 s - 1585

Continuous-time transfer function.


G2 =

  -0.7025 s^2 + 6.24e-16 s + 31.91
  --------------------------------
                s^2

Continuous-time transfer function.


D1 =

  -0.0072 s^4 + 8.208 s^3 - 2940 s^2 - 204984 s - 950400
  ------------------------------------------------------
       0.0012 s^4 + 1.524 s^3 + 676.8 s^2 + 40320 s

Continuous-time transfer function.


P1 =

    0.8819


D2 =

  0.06 s^4 - 7.158 s^3 + 283 s^2 + 200.7 s + 34.56
  ------------------------------------------------
```

```
     0.12 s^4 + 15.24 s^3 + 676.8 s^2 + 4032 s

Continuous-time transfer function.


Dz1 =

  -2.157 z^4 - 3.486 z^3 + 3.281 z^2 + 5.173 z - 3.125
  ----------------------------------------------------
  z^4 - 0.3149 z^3 - 0.8703 z^2 + 0.007581 z + 0.1777

Sample time: 0.01 seconds
Discrete-time transfer function.


Pz1 =

    0.8770


Dz2 =

  0.1418 z^4 + 0.171 z^3 - 0.3438 z^2 - 0.3249 z + 0.357
  ------------------------------------------------------
    z^4 - 0.318 z^3 - 0.8686 z^2 + 0.009047 z + 0.1776

Sample time: 0.1 seconds
Discrete-time transfer function.


T1 =


  0.004931 s^12 + 1.058 s^11 - 2291 s^10 - 5.058e05 s^9 + 1.1e09 s^8


          + 2.428e11 s^7 + 1.749e13 s^6 + 4.406e14 s^5 + 1.296e15 s^4


                                          - 9.429e15 s^3 - 3.466e16
 s^2


 --------------------------------------------------------------------------


  1.44e-06 s^14 + 0.003901 s^13 + 4.58 s^12 + 2854 s^11 + 9.715e05
 s^10
```

```
          + 1.674e08 s^9 + 1.6e10 s^8 + 8.923e11 s^7 + 2.8e13 s^6


               + 4.039e14 s^5 + 8.787e14 s^4 - 8.969e15 s^3 - 3.057e16
 s^2


Continuous-time transfer function.

T2 =


  -0.5415 s^6 + 64.6 s^5 - 2529 s^4 - 4746 s^3 + 1.157e05 s^2 +
 8.229e04 s


                                                               +
 1.417e04


 -------------------------------------------------------------------------


  s^6 + 260.4 s^5 + 6165 s^4 + 4.705e04 s^3 + 1.157e05 s^2 + 8.229e04
 s


                                                               +
 1.417e04


Continuous-time transfer function.

Tz1 =


  0.04734 z^13 - 0.06521 z^12 - 0.2218 z^11 + 0.3608 z^10 + 0.3603 z^9


          - 0.8446 z^8 + 0.03327 z^7 + 0.6586 z^6 - 0.2541 z^5 -
 0.1966 z^4


                         + 0.1268 z^3 + 0.01343 z^2 - 0.02225 z +
 0.003954
```

```
 ---------------------------------------------------------------------------

   z^14 - 5.451 z^13 + 11.01 z^12 - 7.671 z^11 - 5.304 z^10 + 11.47
 z^9

           - 3.378 z^8 - 5.024 z^7 + 3.891 z^6 + 0.3236 z^5 - 1.233
 z^4

                              + 0.3 z^3 + 0.1276 z^2 - 0.07223 z +
 0.009773


Sample time: 0.01 seconds
Discrete-time transfer function.


Tz2 =


  -0.09964 z^12 + 0.3327 z^11 + 0.1197 z^10 - 1.468 z^9 + 0.908 z^8


          + 2.332 z^7 - 3.271 z^6 + 0.3495 z^5 + 1.523 z^4 - 0.6714
 z^3

                                         - 0.2185 z^2 + 0.1976 z -
 0.03441


 -------------------------------------------------------------------------

  0.9004 z^12 - 4.303 z^11 + 7.027 z^10 - 2.169 z^9 - 6.543 z^8


          + 7.12 z^7 - 0.3594 z^6 - 3.031 z^5 + 1.31 z^4 + 0.3267 z^3


                                       - 0.3506 z^2 + 0.07467 z -
 0.002886
```

```
Sample time: 0.1 seconds
Discrete-time transfer function.


Tsystem =


  -1061 s^22 - 2.842e06 s^21 - 2.771e09 s^20 - 6.14e11 s^19 + 7.033e14
 s^18

          + 2.082e17 s^17 - 4.584e20 s^16 - 4.14e23 s^15 - 1.427e26
 s^14

          - 2.007e28 s^13 - 8.193e29 s^12 + 5.595e31 s^11 + 3.539e33
 s^10

          - 3.173e35 s^9 - 3.262e37 s^8 - 9.449e38 s^7 - 8.052e39 s^6

          + 3.384e39 s^5 + 3.624e41 s^4 + 1.79e42 s^3 + 3.193e42 s^2

                                                  + 1.636e42 s +
 2.488e41

 --------------------------------------------------------------------------

   s^24 + 5420 s^23 + 1.371e07 s^22 + 2.122e10 s^21 + 2.223e13 s^20

           + 1.651e16 s^19 + 8.878e18 s^18 + 3.473e21 s^17 + 9.863e23
 s^16

           + 2.026e26 s^15 + 3.031e28 s^14 + 3.341e30 s^13 + 2.742e32
 s^12

           + 1.674e34 s^11 + 7.474e35 s^10 + 2.354e37 s^9 + 4.976e38
 s^8
```

```
              + 6.745e39 s^7 + 5.924e40 s^6 + 3.387e41 s^5 + 1.232e42 s^4

                         + 2.699e42 s^3 + 3.193e42 s^2 + 1.636e42 s +
 2.488e41


Continuous-time transfer function.
```

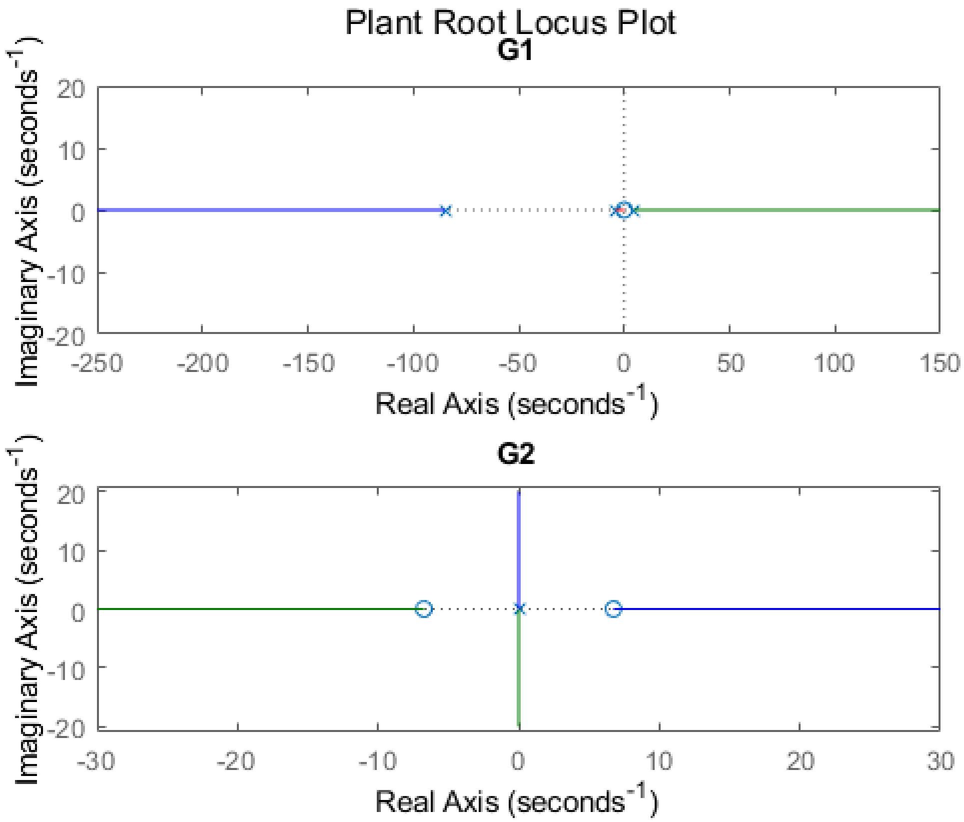
Plotting

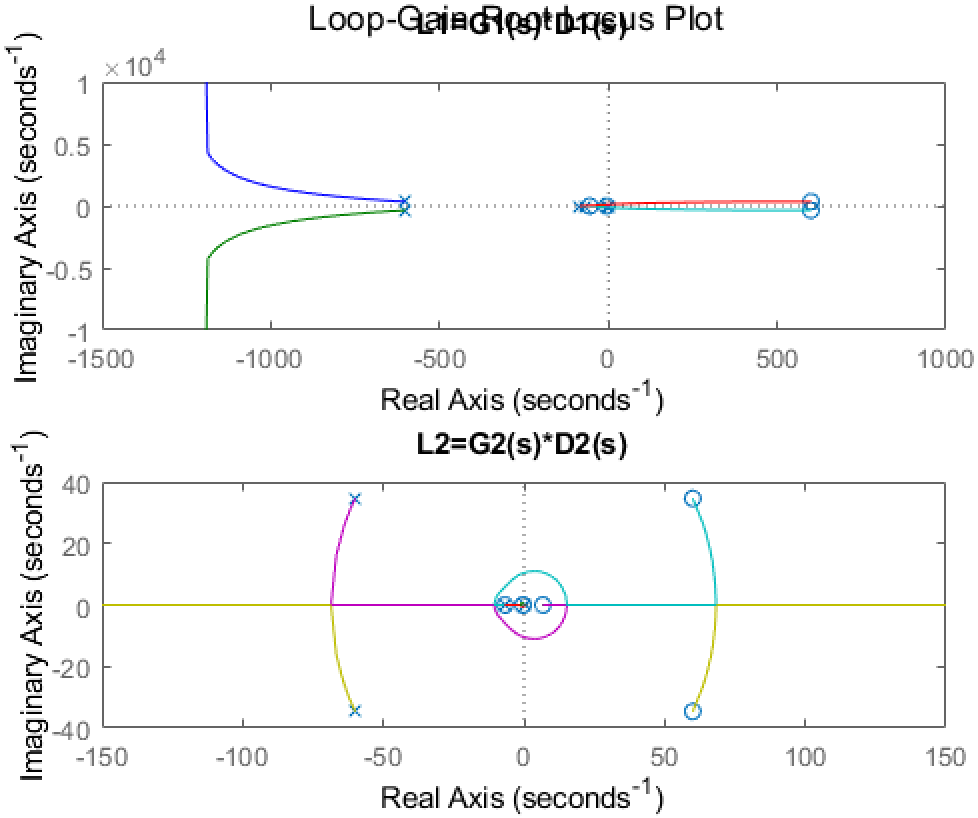
ROOT LOCUS PLOTS

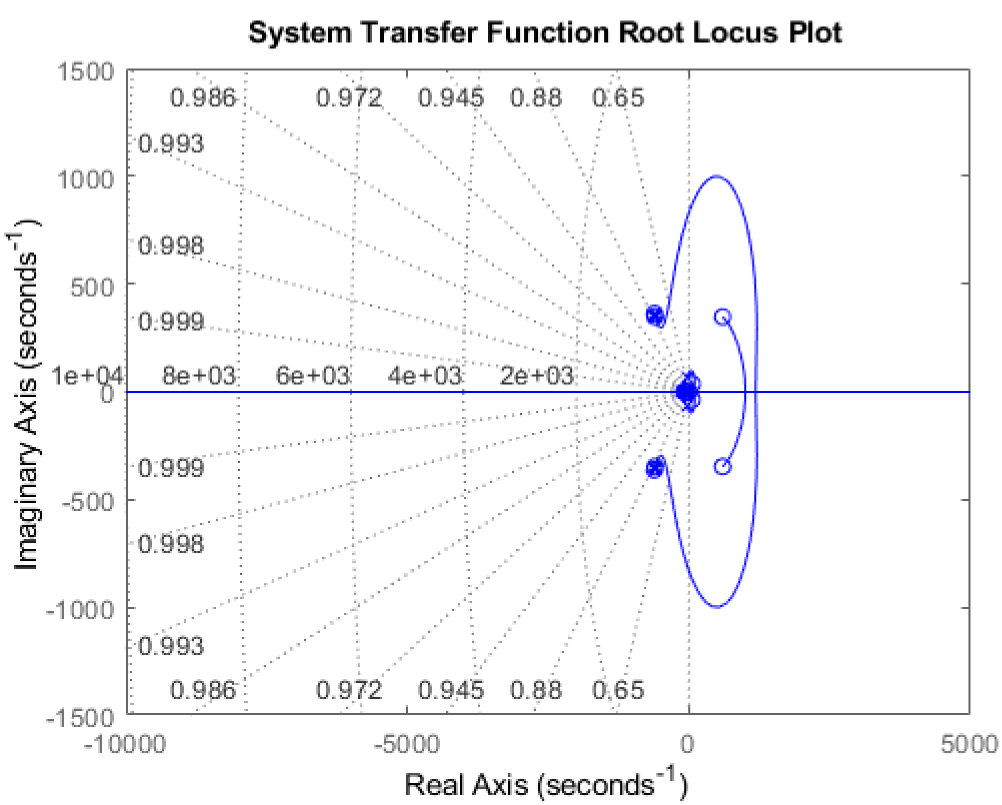
```
figure(1)
subplot(2,1,1)
rlocus(G1)
title('G1')

subplot(2,1,2)
rlocus(G2)
title('G2')
sgtitle('     Plant Root Locus Plot')

figure(2)
sgtitle('     Loop-Gain Root Locus Plot')
subplot(2,1,1)
rlocus(L1)
title('L1=G1(s)*D1(s)')
subplot(2,1,2)
rlocus(L2)
title('L2=G2(s)*D2(s)')


figure(3)
rlocus(Tsystem,'b')
grid on
title('System Transfer Function Root Locus Plot')

% BODE PLOTS
figure(4)
subplot(2,1,1)
bode(G1)
grid on
title('L1=G1(s)*D1(s)')

subplot(2,1,2)
bode(G2)
grid on
```

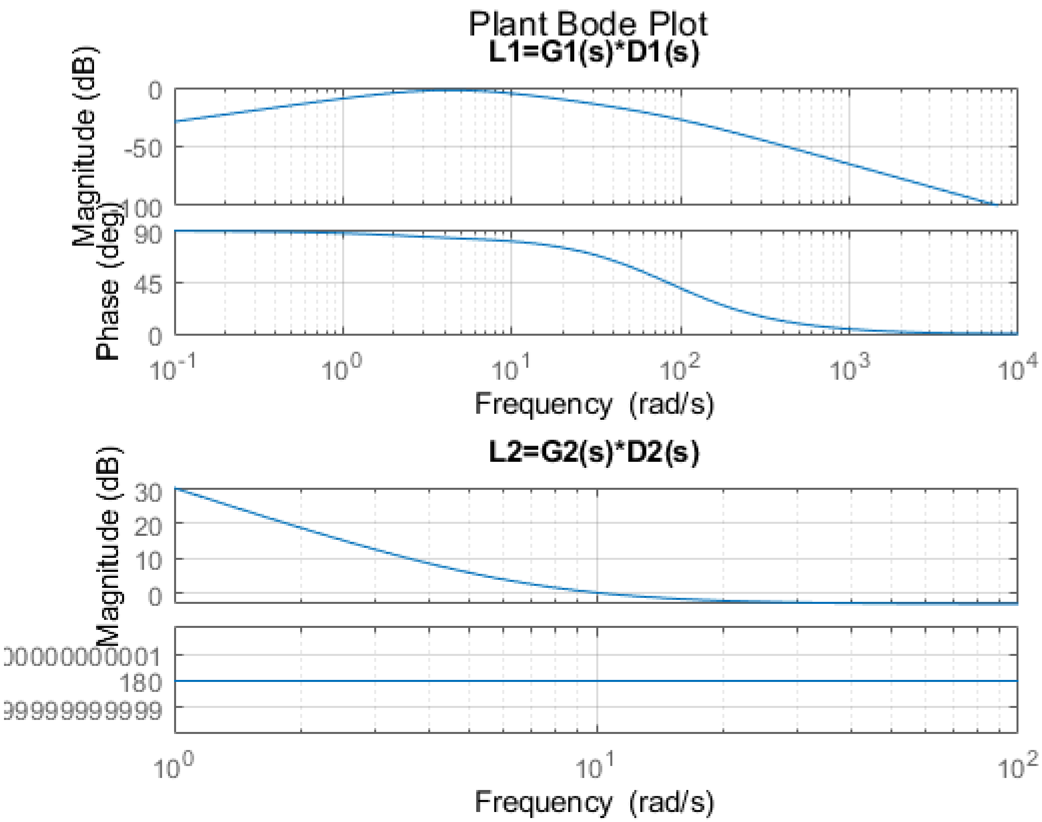
```
title('L2=G2(s)*D2(s)')

sgtitle('     Plant Bode Plot')

figure(5)
subplot(2,1,1)
bode(L1)
grid on
title('L1=G1(s)*D1(s)')

subplot(2,1,2)
bode(L3)
grid on
title('L3=G2(s)*D2(s)*T1')

sgtitle('     Controller Bode Plot')

figure(6)
margin(Tsystem)
hold on
bode(Tsystem,'b')
grid on
title('System Transfer Function Bode Plot')


% NYQUIST PLOTS
figure(7)
subplot(2,1,1)
nyquist(L1)
title('L1=G1(s)*D1(s)')
subplot(2,1,2)
nyquist(L3)
xlim([-1.5 .8])
title('L2=G2(s)*D2(s)')
sgtitle('     Loop-Gain Nyquist Plot')

% STEP RESPONSES
figure(8)
step(P1 * T1,'b',Pz1 * Tz1,'r',1)
hold on
grid on
legend('Continuous','Discrete')
title('Inner-Loop Transfer Function Step Response')

figure(9)
step(T2,'b',Tz2,'r',5)
hold on
grid on
legend('Continuous','Discrete')
title('Outer-Loop Transfer Function Step Response')

figure(10)
step(Tsystem,'b',TzSystem,'r')
hold on
```

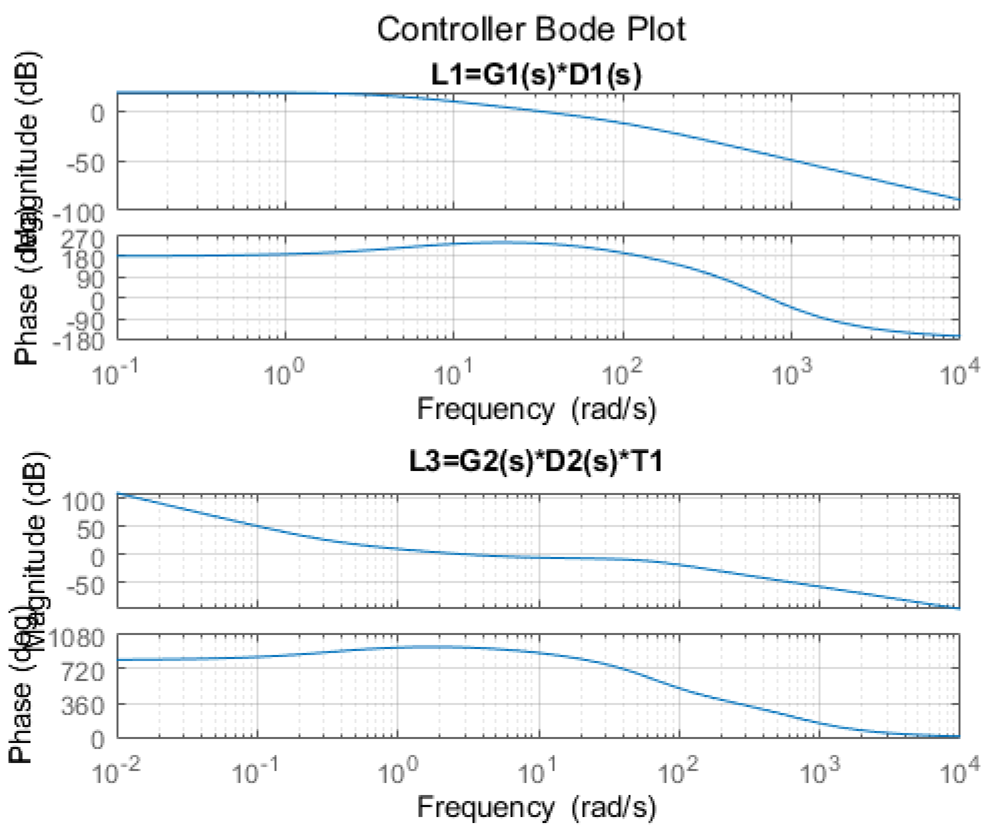
```
grid on
legend('Continuous','Discrete')
title('System Transfer Function Step Response')
```

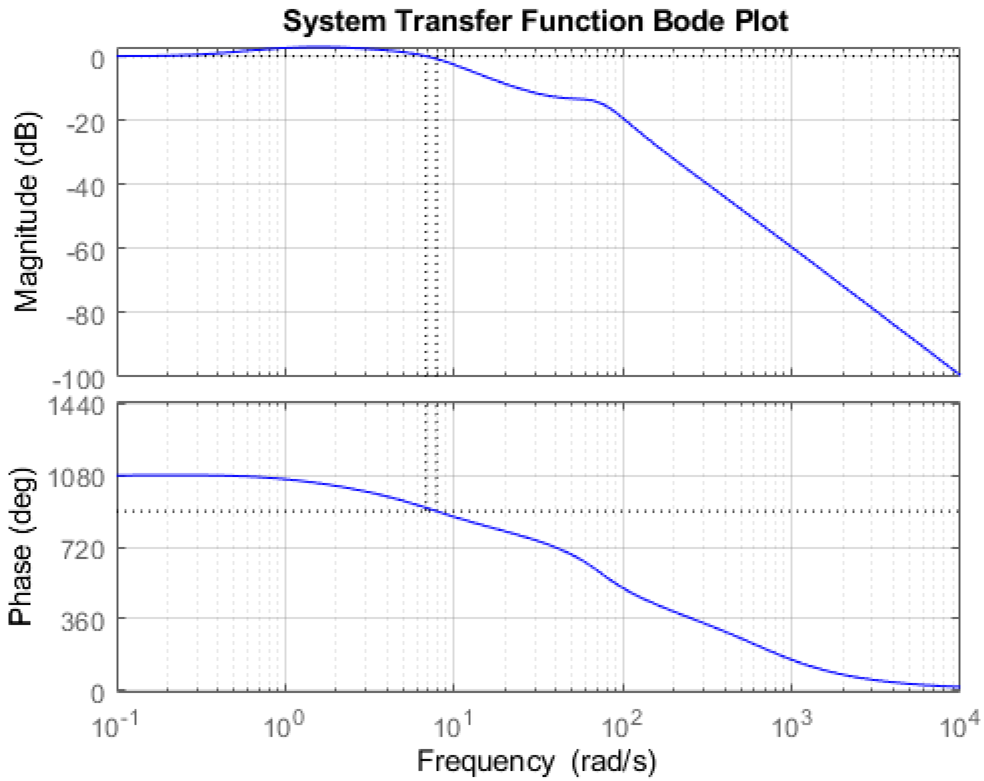


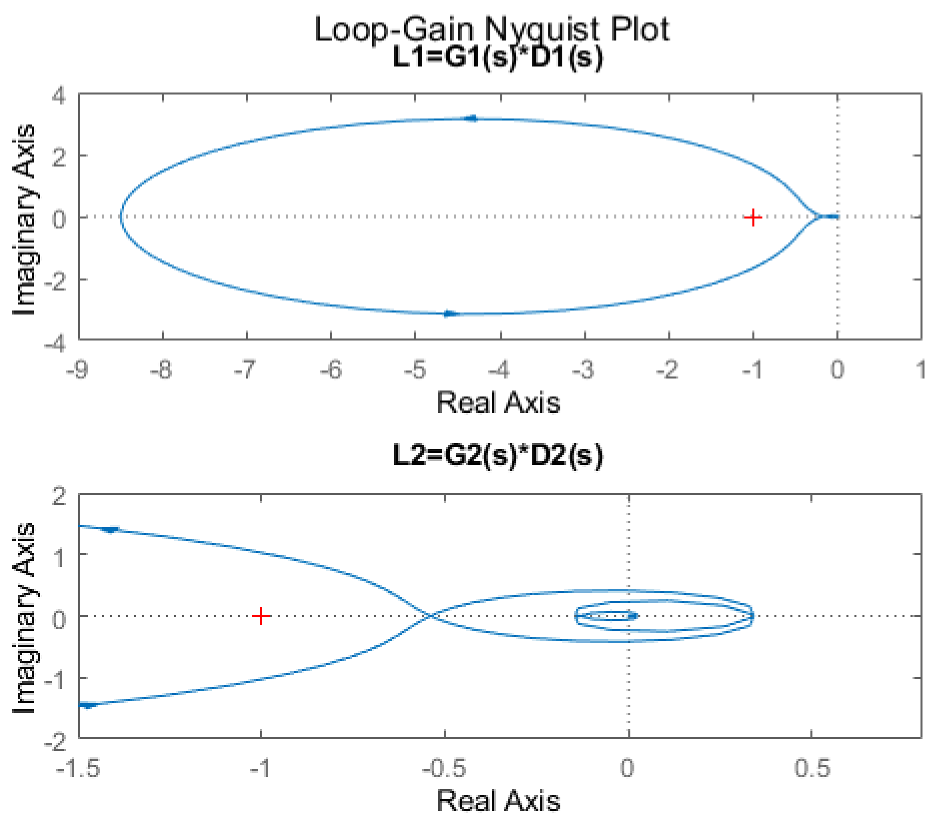

Loop-Gain Root Locus Plot
L1=G1(s)*D1(s)
L2=G2(s)*D2(s)
Imaginary Axis (seconds^{-1})
Real Axis (seconds^{-1})
×10^4

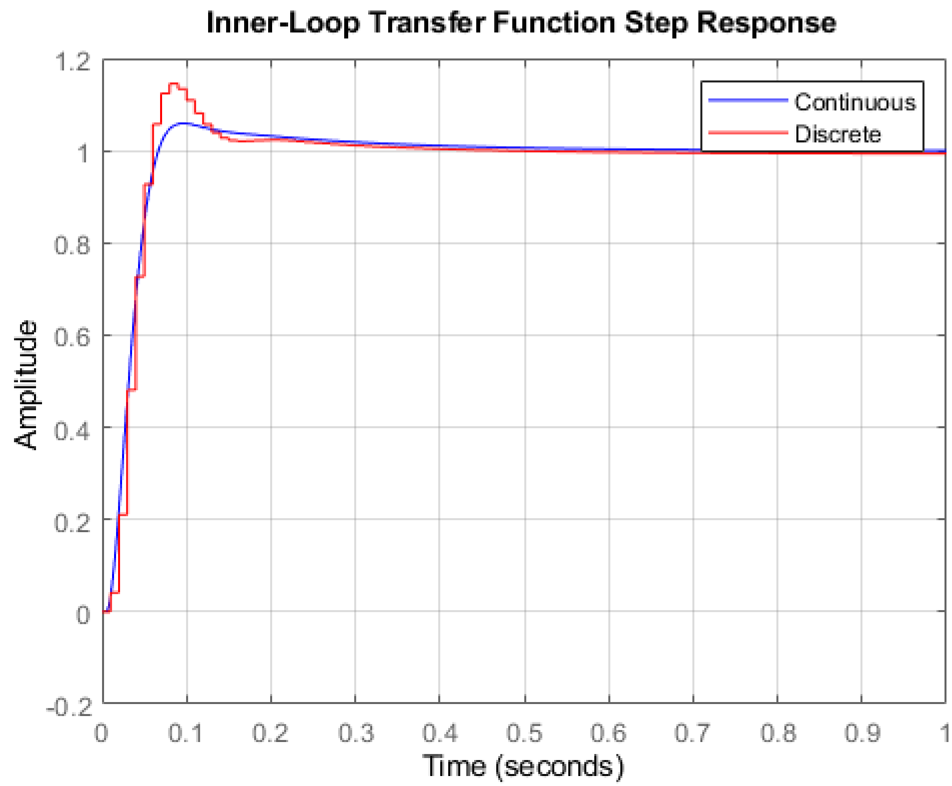

System Transfer Function Root Locus Plot
Imaginary Axis (seconds^{-1})
Real Axis (seconds^{-1})
0.986
0.972
0.945
0.88
0.65
0.993
0.998
0.999
1e+04
8e+03
6e+03
4e+03
2e+03

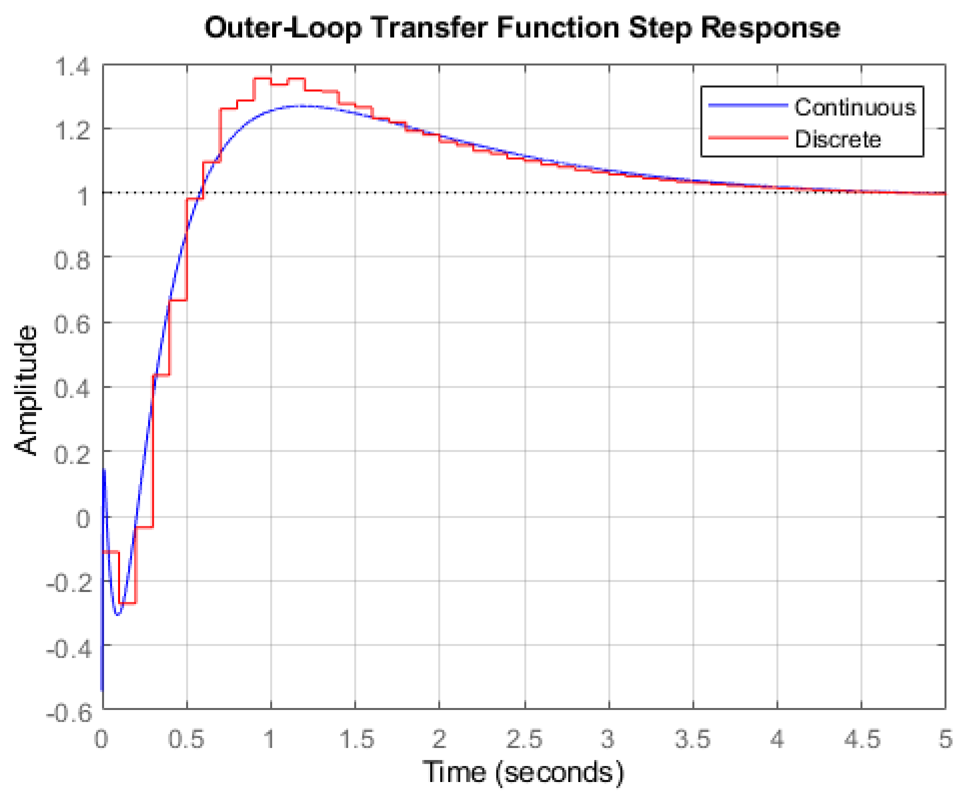

Plant Bode Plot
L1=G1(s)*D1(s)
Magnitude (dB)
0
-50
-100
Phase (deg)
90
45
0
10^-1
10^0
10^1
10^2
10^3
10^4
Frequency (rad/s)
L2=G2(s)*D2(s)
Magnitude (dB)
30
20
10
0
180
10^0
10^1
10^2
Frequency (rad/s)

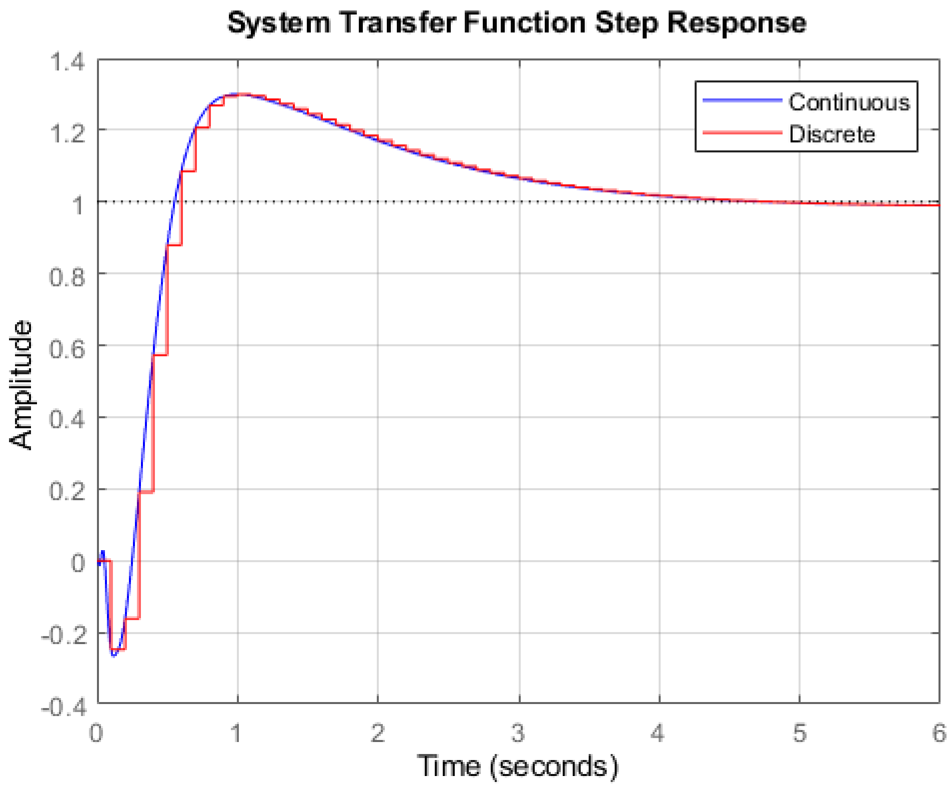

Controller Bode Plot
L1=G1(s)*D1(s)
Magnitude (dB)
0
-50
-100
Phase (deg)
270
180
90
0
-90
-180
10^-1
10^0
10^1
10^2
10^3
10^4
Frequency (rad/s)
L3=G2(s)*D2(s)*T1
Magnitude (dB)
100
50
0
-50
Phase (deg)
1080
720
360
0
10^-2
10^-1
10^0
10^1
10^2
10^3
10^4
Frequency (rad/s)


System Transfer Function Bode Plot
Magnitude (dB)
0
-20
-40
-60
-80
-100
Phase (deg)
1440
1080
720
360
0
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
Frequency (rad/s)


Loop-Gain Nyquist Plot
L1=G1(s)*D1(s)
Imaginary Axis
4
2
0
-2
-4
-9
-8
-7
-6
-5
-4
-3
-2
-1
0
1
Real Axis
L2=G2(s)*D2(s)
Imaginary Axis
2
1
0
-1
-2
-1.5
-1
-0.5
0
0.5
Real Axis


Inner-Loop Transfer Function Step Response
Continuous
Discrete
Amplitude
Time (seconds)
1.2
1
0.8
0.6
0.4
0.2
0
-0.2
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


Outer-Loop Transfer Function Step Response
Continuous
Discrete
Amplitude
Time (seconds)
1.4
1.2
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5



Published with MATLAB® R2020b