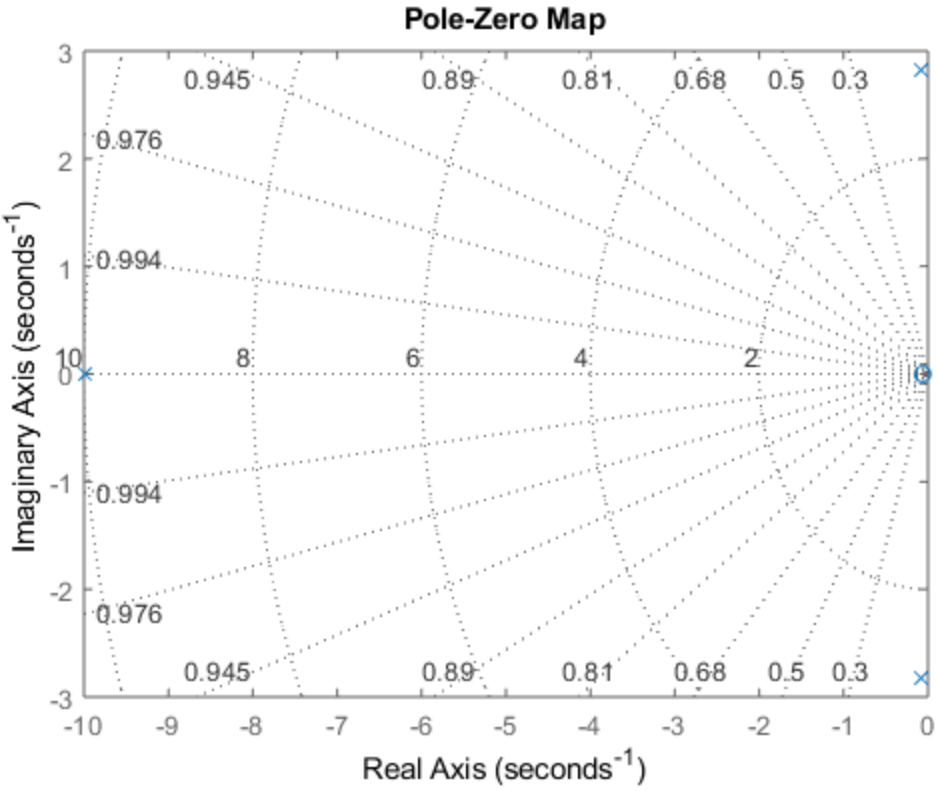
System analysis problem 3

OPEN LOOP part a

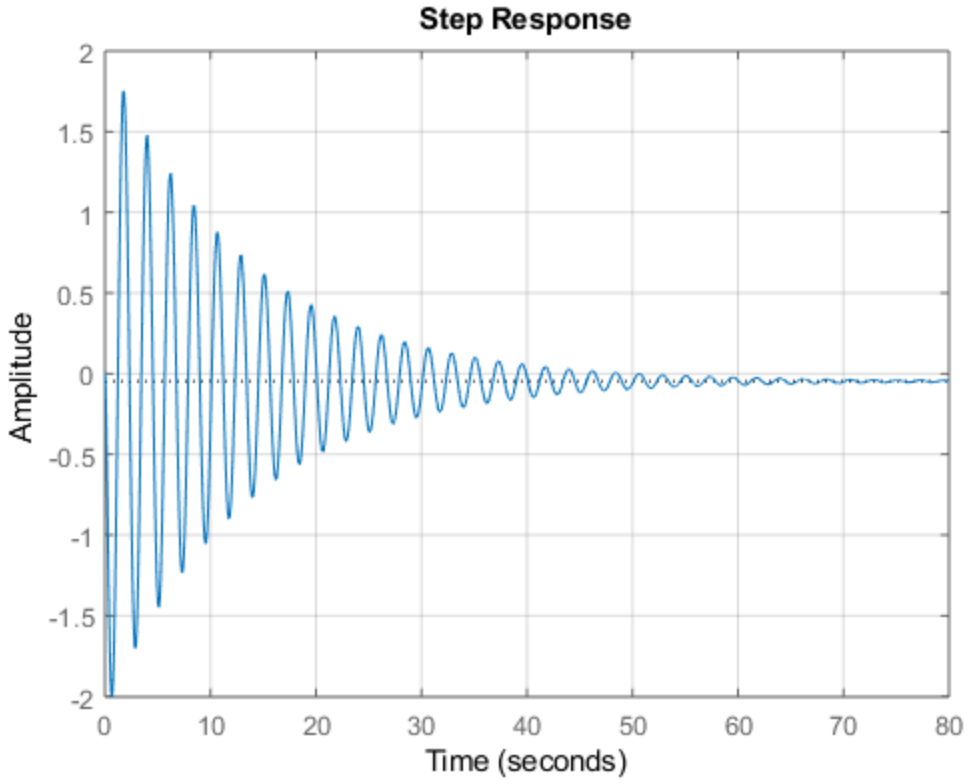
```
clear all; clc; close all; warning off;
A = [-0.09 1.0 -0.02; -8.0 -0.06 -6.0; 0 0 -10];
B = [0;0;10];
% Assuming rate of change of pitch angle as output

C = [0 1 0];
D = [0];

s=tf('s');

sys = ss(A,B,C,D)
[N, D] = ss2tf(A,B,C,D);
G = tf(N,D)
```

```
sys =

  A =
          x1     x2     x3
   x1  -0.09      1  -0.02
   x2     -8  -0.06     -6
   x3      0      0    -10

  B =
       u1
   x1   0
   x2   0
   x3  10

  C =
       x1  x2  x3
   y1   0   1   0

  D =
       u1
   y1   0

Continuous-time state-space model.


G =

             -60 s - 3.8
  ---------------------------------
  s^3 + 10.15 s^2 + 9.505 s + 80.05

Continuous-time transfer function.
```

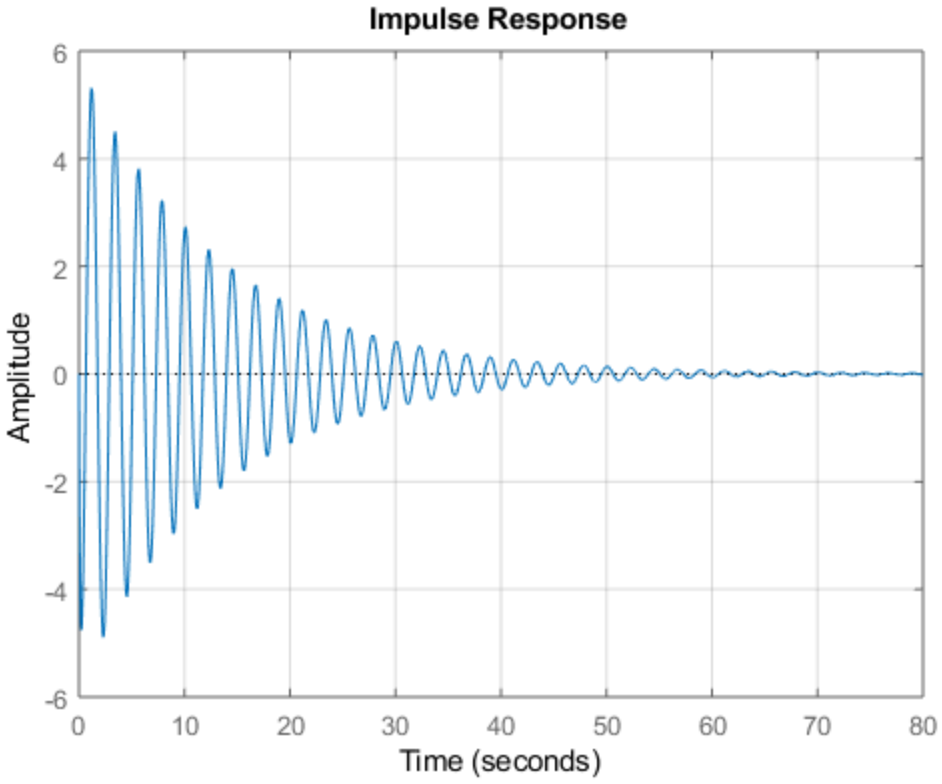
```
figure(1)
pzmap(G)
grid on;
```



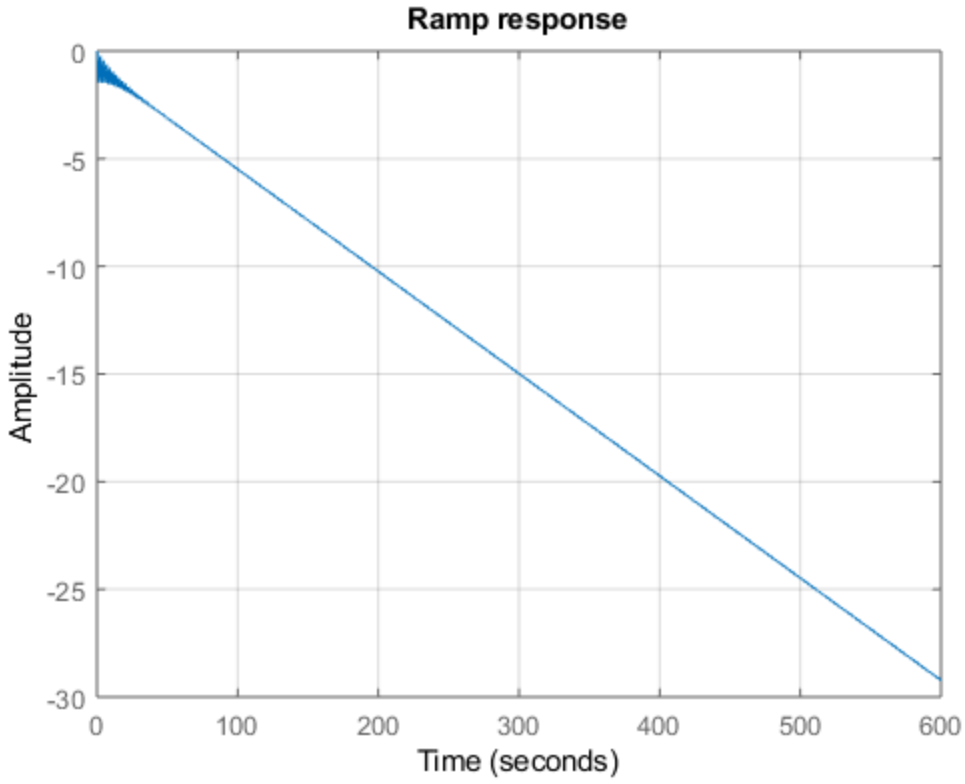

```
figure(2)
step(G)
grid on;
```



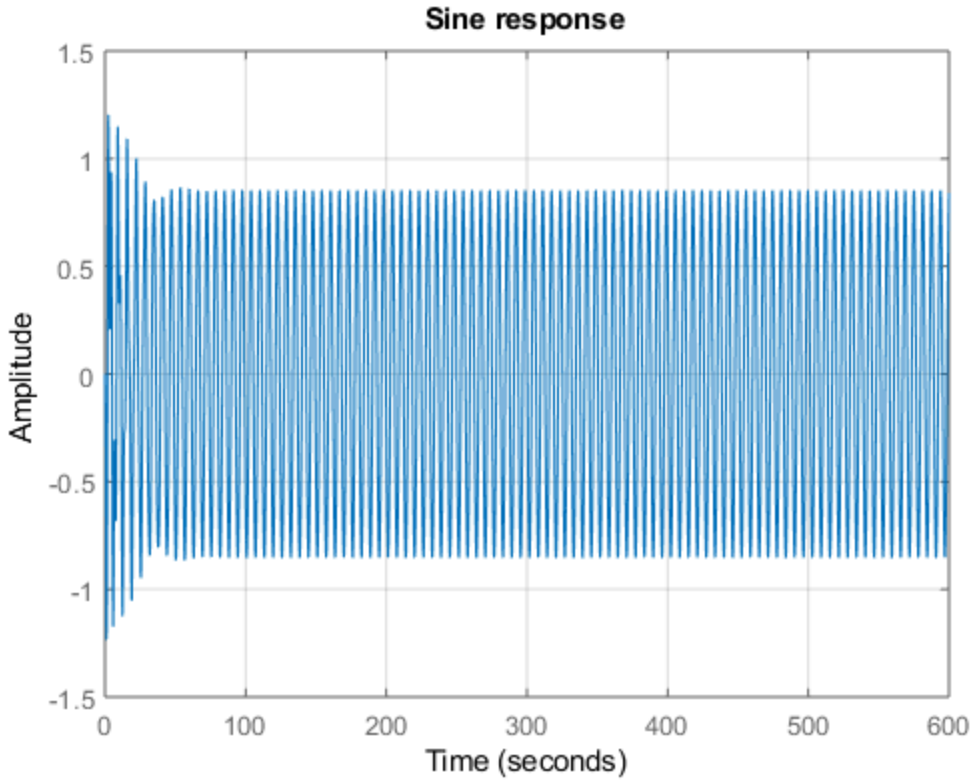

```
figure(3)
impulse(G)
grid on;
```




```
figure(4)
step(G*(1/s))
title('Ramp response')
grid on;
```



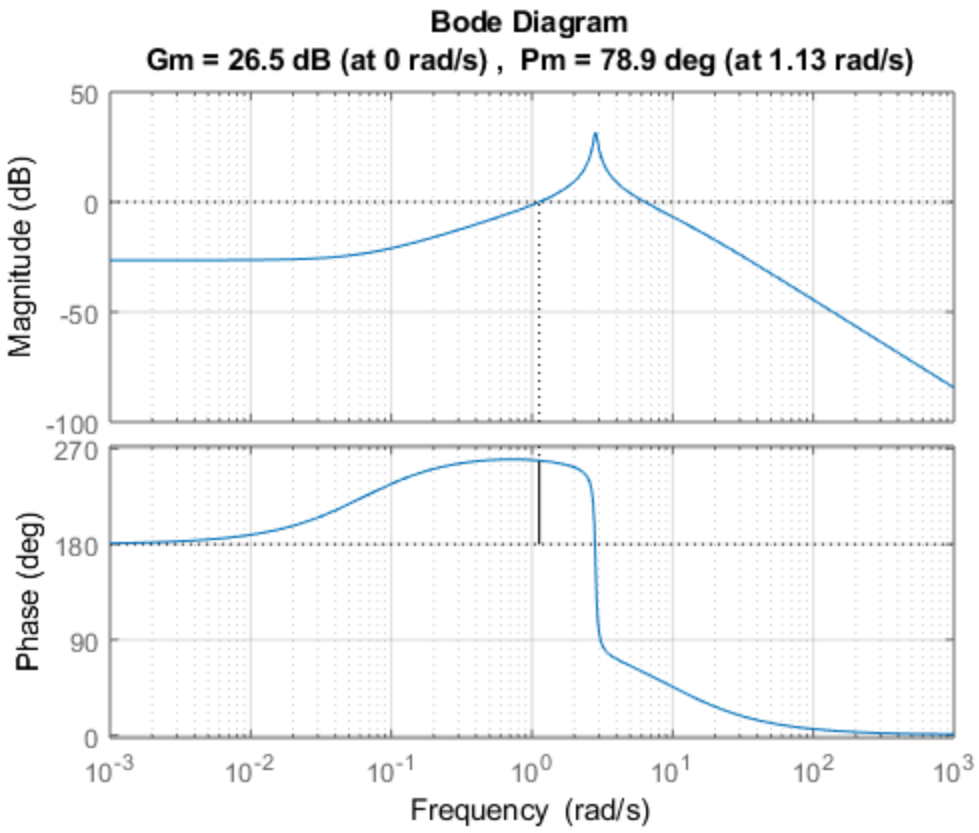

```
figure(5)
a=1;
impulse(G*(a/((s^2)+a^2)))
title('Sine response')
grid on;
```




```
info=stepinfo(G)
```

```
info =

  struct with fields:

        RiseTime: 0.0280
    SettlingTime: 51.8855
     SettlingMin: -2.0043
     SettlingMax: 1.7522
       Overshoot: 4.1225e+03
      Undershoot: 3.6913e+03
            Peak: 2.0043
        PeakTime: 0.6539
```

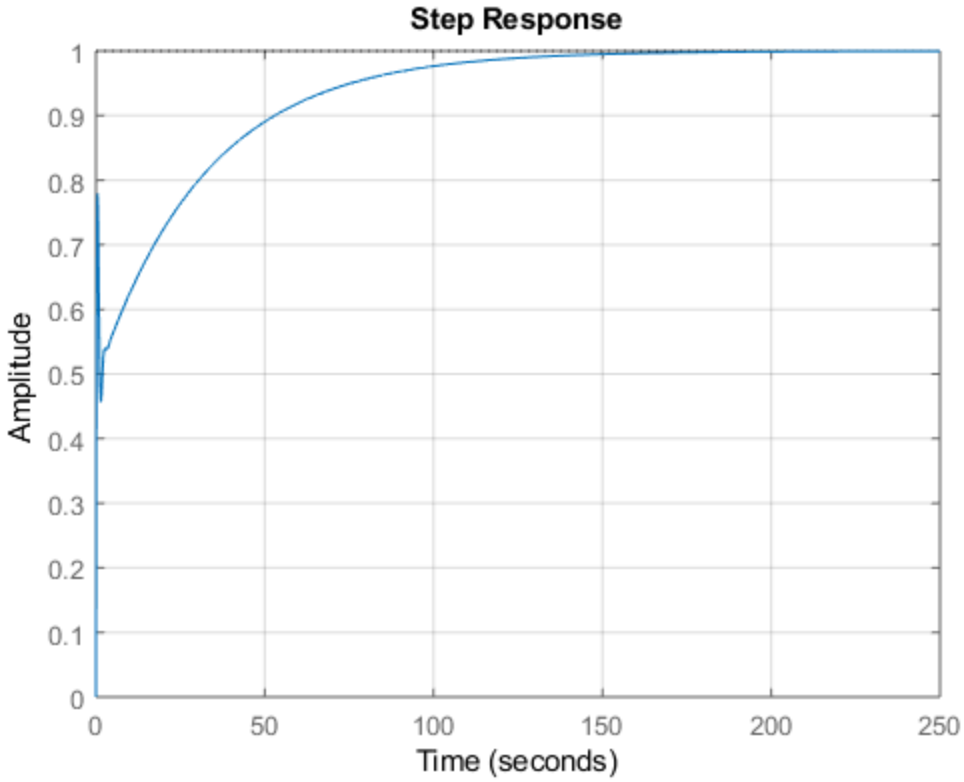
```
figure(6)
margin(G)
grid on;
```



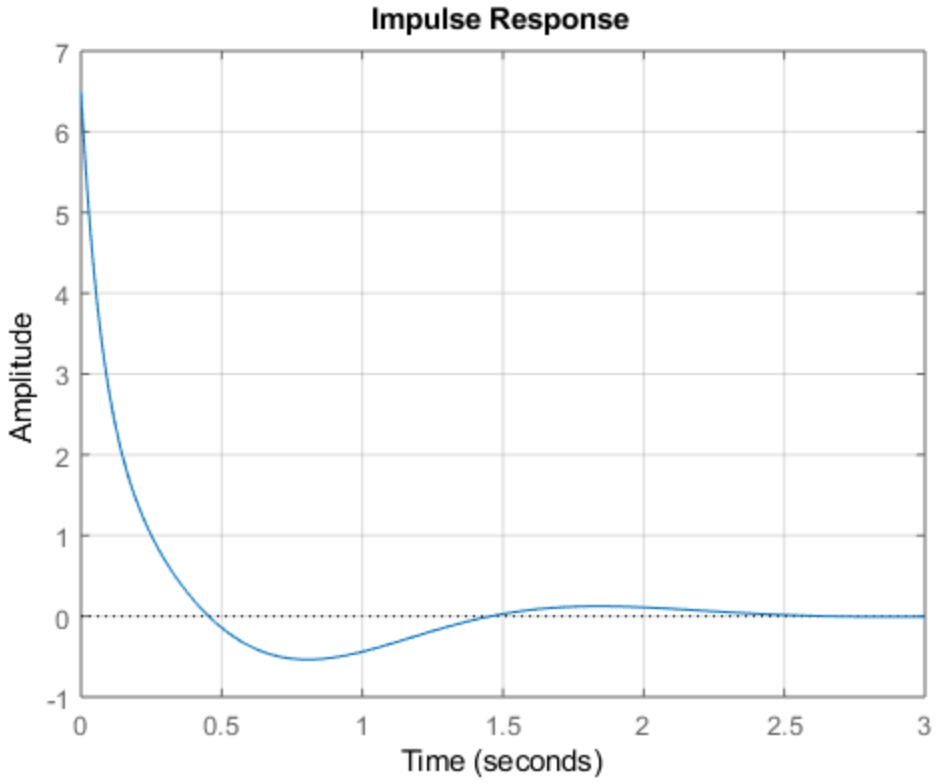

```
%Analysing the open loop simulation of the system we can say that this
%system is not stable system. If we see the step response, it takes
 80s of
%time for settling which is not required.
```

Closed loop - PID tuned

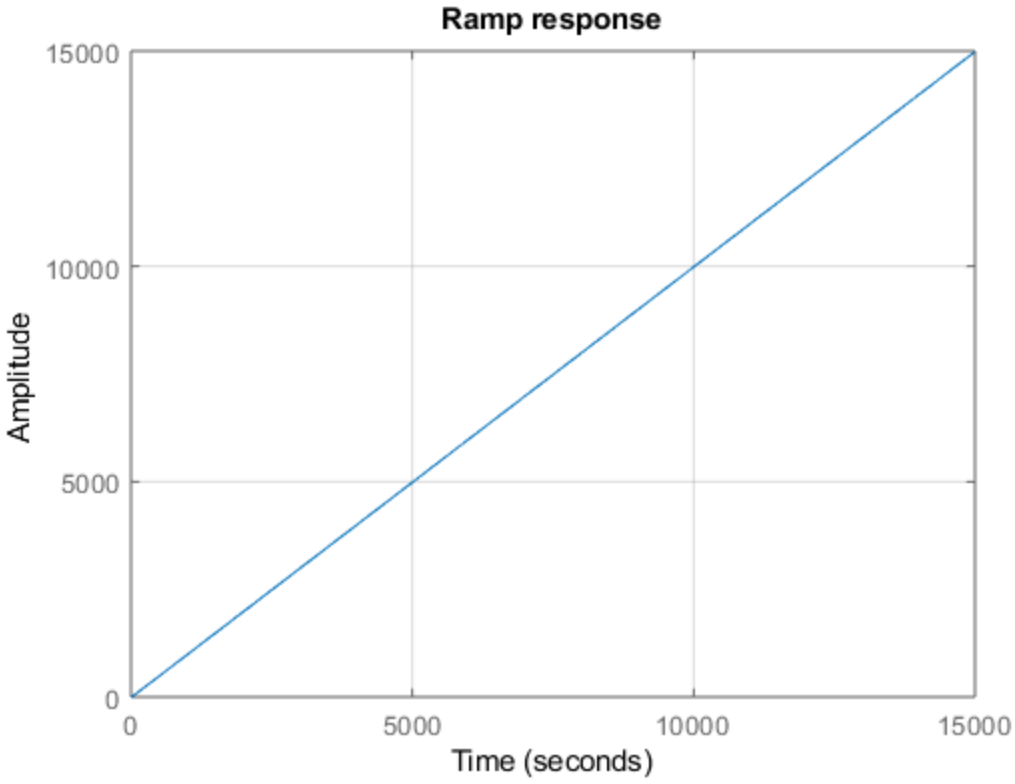
```
%In this part of the problem, I designed PID controller for this
 system by
%using PID tuner apps in Matlab and by loading it to workspace, I plot
%everything again so that we can check the better performance compare
 to
%open loop system
load PIDC.mat
figure(7)
step(feedback(G*PIDC,1))
grid on;
```



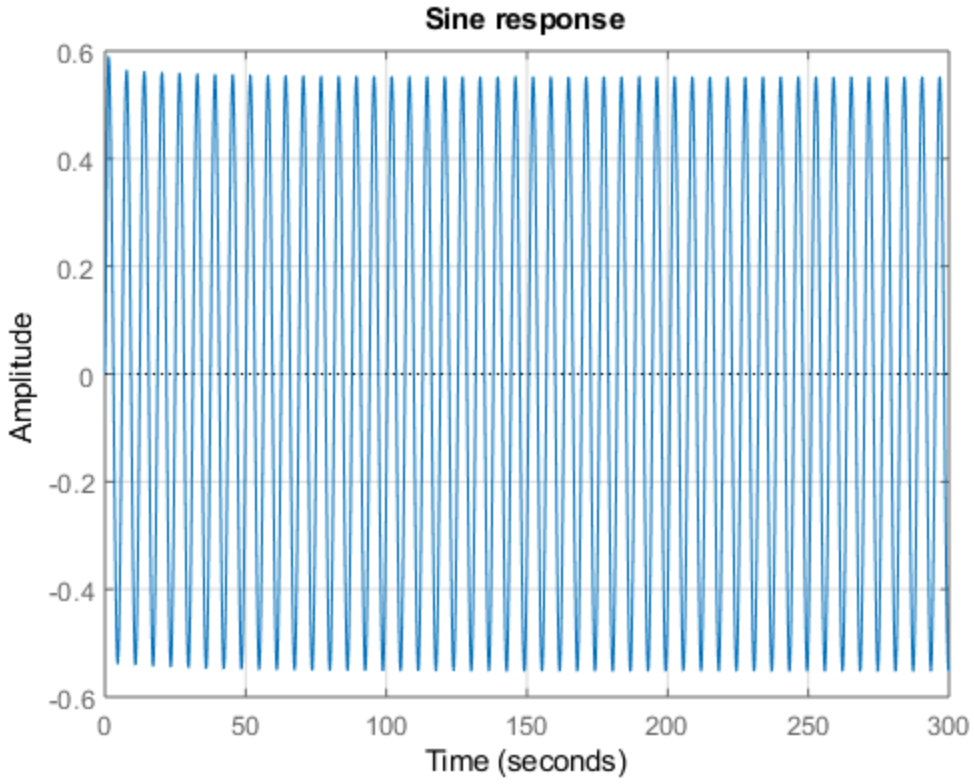

```
figure(8)
impulse(feedback(G*PIDC,1))
grid on;
```




```
figure(9)
step(feedback(G*PIDC,1)*(1/s))
title('Ramp response')
grid on;
```



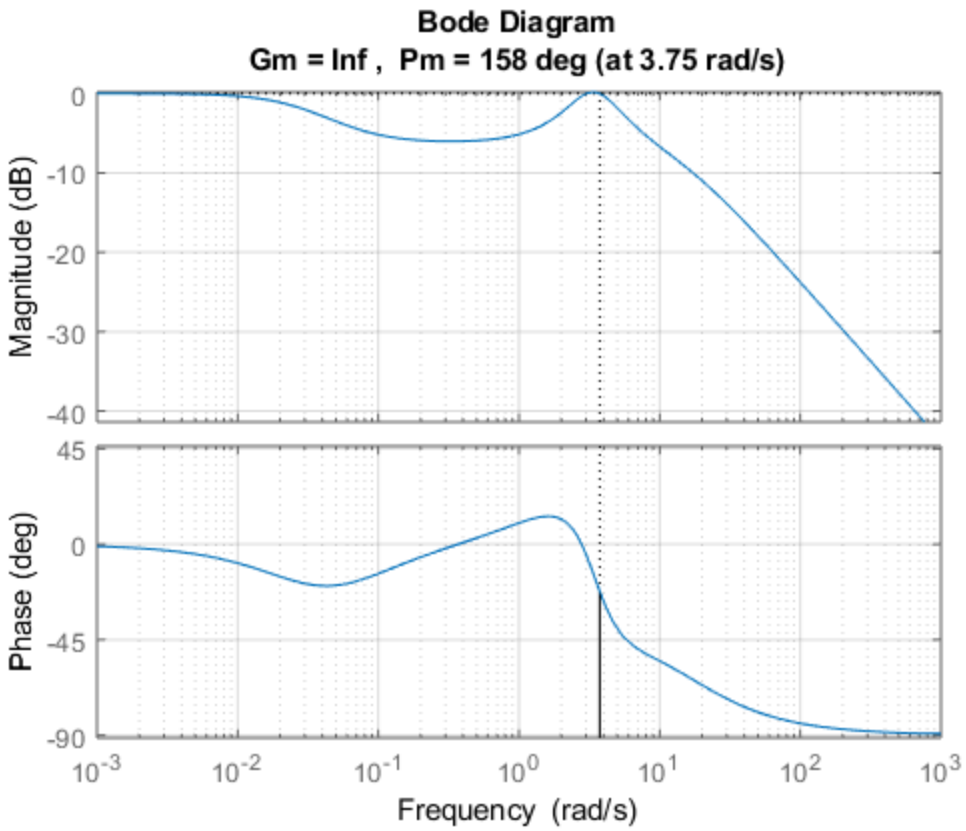

```
figure(10)
a=1;
impulse(feedback(G*PIDC,1)*(a/((s^2)+a^2)))
title('Sine response')
grid on;
```




```
info=stepinfo(feedback(G*PIDC,1))
```

```
info =

  struct with fields:

        RiseTime: 52.9183
    SettlingTime: 105.2191
     SettlingMin: 0.9000
     SettlingMax: 0.9998
       Overshoot: 0
      Undershoot: 0
            Peak: 0.9998
        PeakTime: 259.7290
```

```
figure(11)
margin(feedback(G*PIDC,1))
grid on;
```




```
PIDC
```

```
PIDC =

             1
  Kp + Ki * --- + Kd * s
             s

  with Kp = -0.745, Ki = -1.28, Kd = -0.109

Continuous-time PID controller in parallel form.
```

Published with MATLAB® R2018a