2(a) Second Order MSD Equation

Table of Contents

Author: Pushkar Antony PS Number: 99003729 Date: 7th April 2021. Version: 1.0.

Plant Description

The Mass-damper Spring Second order system is taken as Plant. It is used in as suspension. Equation: Mx"(t)+ Bx'(t) + Kx(t)= Kf(t). f = force; B= coefficient of friction; M = mass ; v= velocity; k=spring

```
%constant.
% Values: K1= 0.9 B1= 0.4 M1=1000 Wn=0.03 ; K2= 1 B2= 0.5 M2= 500
 Wn=0.44;
%K3= 3 B3= 1.7 M3= 340 Wn=0.09;
```

Code:

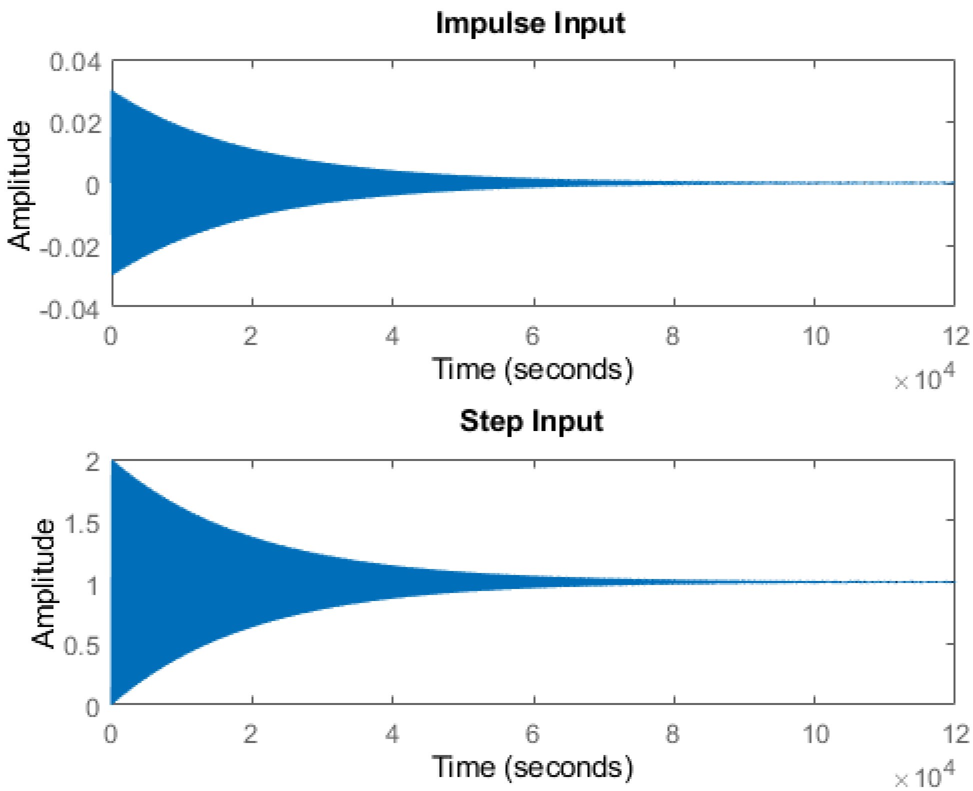
```
clc;
B1= ([0.1 0.5 1.7]);
M1=([1000 5 340]);
K1 = ([0.9 1 3]);
for i=1:3
    sys = tf([K1(i)/M1(i)],[1,B1(i)/M1(i),K1(i)/M1(i)])
    figure(i);
    subplot(2,1,1);
    impulse(sys);
    title('Impulse Input');
    subplot(2,1,2);
    step(sys);
    title('Step Input');
    [z,p,k]= tf2zp([K1(i)/M1(i)],[1,B1(i)/M1(i),K1(i)/M1(i)])
    figure(4);
    zplane(z,p);
    xlim([-5*1e5 3*1e5]);
    ylim([-5*1e5 3*1e5]);
    hold on;
    S = stepinfo(sys)
end
```

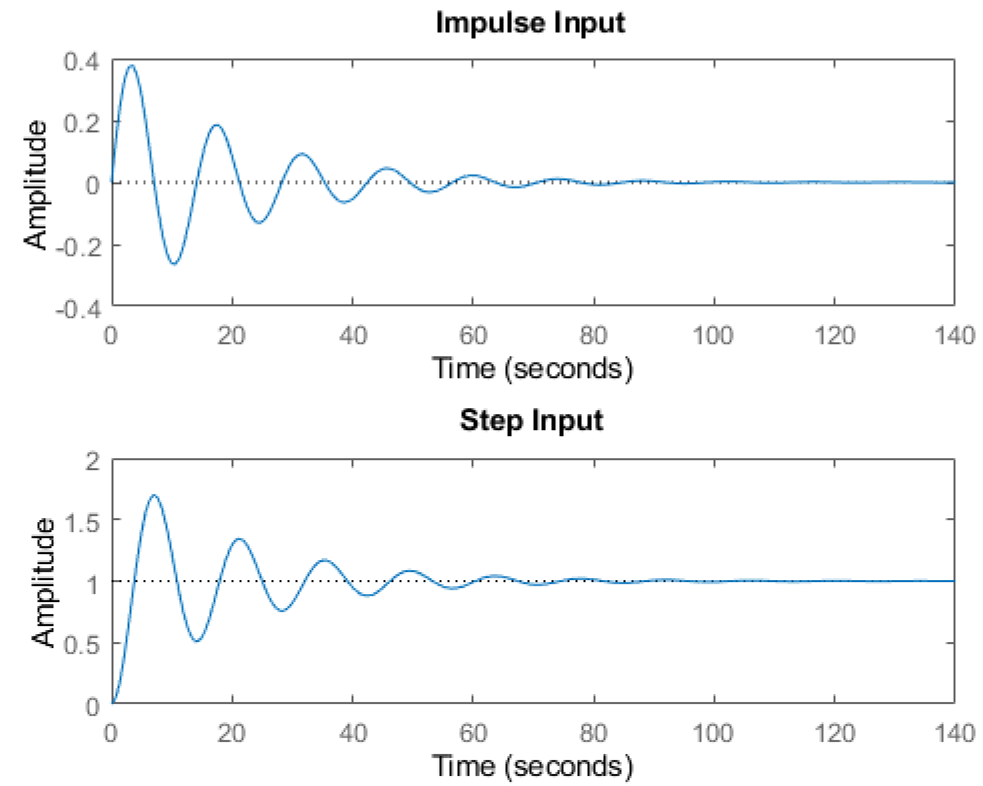
```
sys =

          0.0009
```

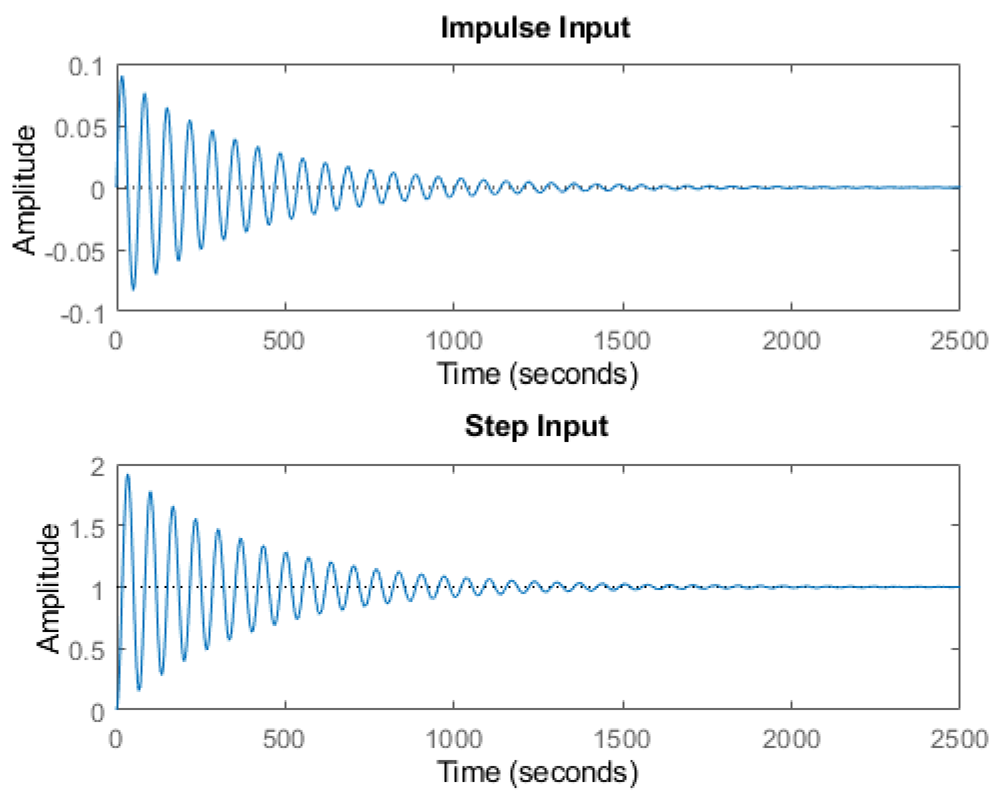
```
  -----------------------
  s^2 + 0.0001 s + 0.0009

Continuous-time transfer function.


z =

  0×1 empty double column vector


p =

  -0.0001 + 0.0300i
  -0.0001 - 0.0300i


k =

   9.0000e-04


S =

  struct with fields:

        RiseTime: 34.7791
    SettlingTime: 7.8226e+04
     SettlingMin: 0.0104
     SettlingMax: 1.9948
       Overshoot: 99.4778
      Undershoot: 0
            Peak: 1.9948
        PeakTime: 104.7198


sys =

         0.2
  -----------------
  s^2 + 0.1 s + 0.2

Continuous-time transfer function.


z =

  0×1 empty double column vector


p =

  -0.0500 + 0.4444i
  -0.0500 - 0.4444i
```

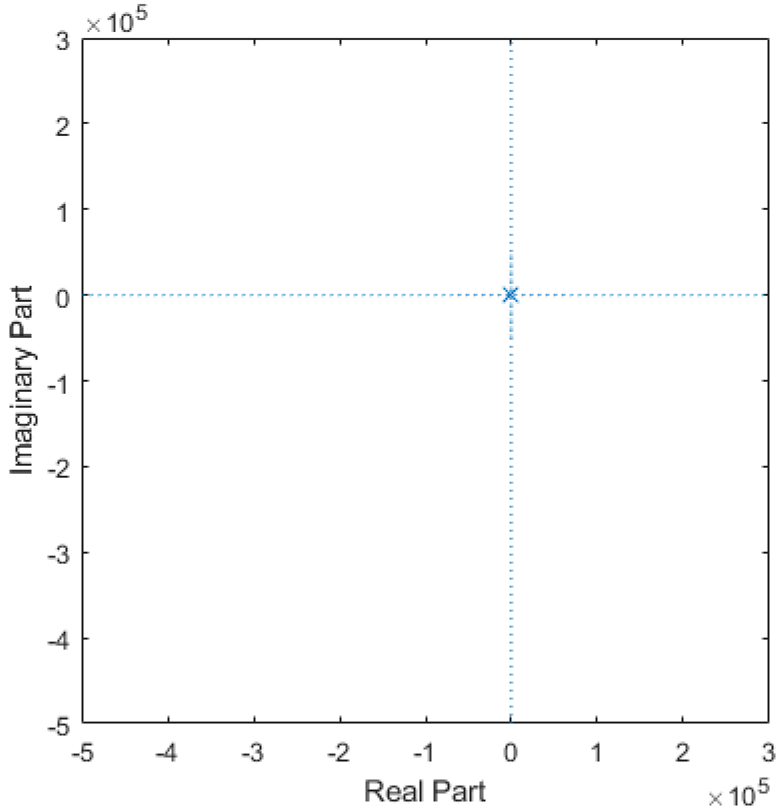
```
k =

    0.2000


S =

  struct with fields:

        RiseTime: 2.5448
    SettlingTime: 78.1524
     SettlingMin: 0.5072
     SettlingMax: 1.7021
       Overshoot: 70.2118
      Undershoot: 0
            Peak: 1.7021
        PeakTime: 7.0248


sys =

          0.008824
  ------------------------
  s^2 + 0.005 s + 0.008824

Continuous-time transfer function.


z =

  0×1 empty double column vector


p =

  -0.0025 + 0.0939i
  -0.0025 - 0.0939i


k =

    0.0088


S =

  struct with fields:

        RiseTime: 11.3230
    SettlingTime: 1.5426e+03
     SettlingMin: 0.1540
     SettlingMax: 1.9198
```

```
 Overshoot: 91.9760
Undershoot: 0
      Peak: 1.9198
  PeakTime: 33.4448
```




Impulse Input
0.4
0.2
0
-0.2
-0.4
Amplitude
0
20
40
60
80
100
120
140
Time (seconds)
Step Input
2
1.5
1
0.5
0
Amplitude
0
20
40
60
80
100
120
140
Time (seconds)


Impulse Input
0.1
0.05
0
-0.05
-0.1
Amplitude
0
500
1000
1500
2000
2500
Time (seconds)
Step Input
2
1.5
1
0.5
0
Amplitude
0
500
1000
1500
2000
2500
Time (seconds)



Math Analysis:

Independent: Time(t) Dependent: Velocity(v) and Force(f) Constant: Mass(M), Frictional Coefficient(B), Spring constant(K) Roots:((-B/M)+-sqrt(sq(B/M)-4K/M))/2

```
% IVT:
% 1. For step input: 0
% 2. For impulse input: 0

% FVT:
% 1. For step input: 1
% 2. For impulse input: K/M

% Time Response Results:
%         RiseTime: 34.7791
%     SettlingTime: 7.8226e+04
%      SettlingMin: 0.0104
%      SettlingMax: 1.9948
%        Overshoot: 99.4778
%       Undershoot: 0
%             Peak: 1.9948
%         PeakTime: 104.7198

%K2= 1 B2= 0.5 M2= 500
%         RiseTime: 2.5448
%     SettlingTime: 78.1524
```

```
%     SettlingMin: 0.5072
%     SettlingMax: 1.7021
%       Overshoot: 70.2118
%      Undershoot: 0
%            Peak: 1.7021
%        PeakTime: 7.0248

%K3= 3 B3= 1.7 M3= 340
%        RiseTime: 11.3230
%    SettlingTime: 1.5426e+03
%     SettlingMin: 0.1540
%     SettlingMax: 1.9198
%       Overshoot: 91.9760
%      Undershoot: 0
%            Peak: 1.9198
%        PeakTime: 33.4448
```

Comparison Analysis:(Speed, Accuracy and stability):

1) For sys 1 poles are on the LHS and they are complex conjugates which

```
%makes the system stable.
% 2) For sys 2 poles are on LHS and they are complex conjugates which
 makes
%the system stable.
% 3) For sys 3 poles are on LHS and they are complex conjugates which
 makes
%the system stable.
% 4) Sys 2 has the least rising time and settling time making the
 system
%fastest and most stable.
```

Published with MATLAB® R2020b