

A Report on

Problem solving using analytical methods and MATLAB

Submitted for the fulfillment of the CIE (Event-2) for the course

Control Systems - EC540

Submitted by:

Name	Mohamed Farhan Fazal
USN	01JST18EC055
Roll No.	29

Under the guidance of

Dr. Sudarshan Patilkulkarni

Associate Professor

DEPARTMENT OF ELECTRONICS AND COMMUNICATION

SJCE MYSURU- 570006

Problem Statement:

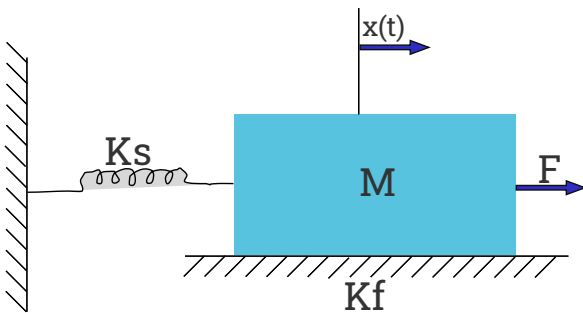
For the mass spring system derive expressions for ζ and ω_n . If $M=10 \text{ Kg}$, determine spring and friction constant for following cases:

1. $\zeta = 0.7, \omega_n = 10$
2. $\zeta = 0.3, \omega_n = 10$
3. $\zeta = 0.01, \omega_n = 10$

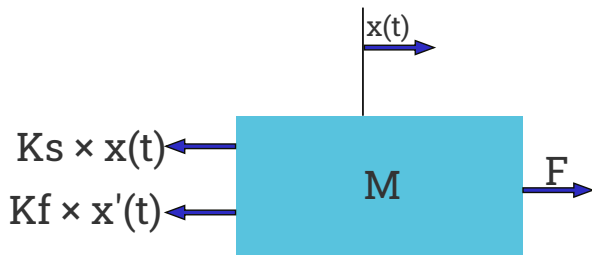
Show the plots for step responses and pole-zero locations in MATLAB.

Solution

- Mass Spring System



- Free Body Diagram



According to Newton's Laws of motion:

$$F(t) - \left\{ K_s x(t) + K_f \frac{dx(t)}{dt} \right\} = M \frac{\partial^2 x(t)}{\partial t^2}$$

$$f(t) \Rightarrow u(t) | x(t) \Rightarrow y(t)$$

$$u(t) = M \frac{\partial^2 y(t)}{\partial t^2} + K_s y(t) + K_f \frac{\partial y(t)}{\partial t}$$

In Laplace Domain

$$U(S) = [MS^2 + K_s + K_f S] \times Y(S)$$

$$\frac{Y(S)}{U(S)} = G(S) = \frac{1}{MS^2 + K_f S + K_s}$$

Comparing with the Standard Second Order Equation

$$G(S) = \frac{\omega_n^2}{S^2 + 2\zeta\omega_n S + \omega_n^2}$$

$$\omega_n = \sqrt{\frac{K_s}{M}} \text{ and } \zeta = \frac{K_f}{2\sqrt{K_s M}}$$

$$K_s = \omega_n^2 \times M \text{ and } K_f = 2\zeta \times \sqrt{K_s \times M}$$

Case 1:

- **M** = 10 Kg
- **ζ** = 0.7
- **ω_n** = 10
- $K_s = 1000N/m$
- $K_f = 140N/m^2$

Case 2:

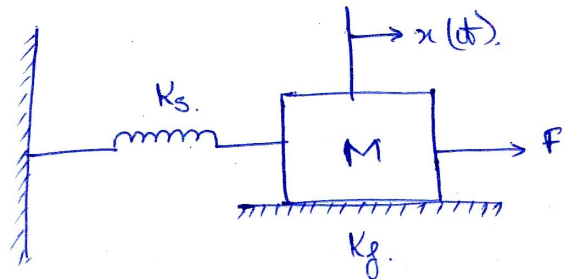
- **M** = 10 Kg
- **ζ** = 0.3
- **ω_n** = 10
- $K_s = 1000N/m$
- $K_f = 60N/m^2$

Case 3:

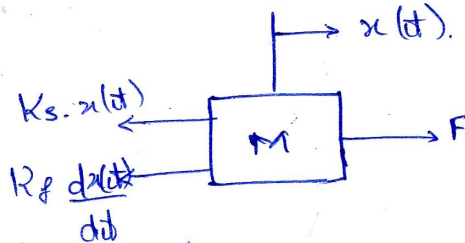
- **M** = 10 Kg
 - **ζ** = 0.01
 - **ω_n** = 10
 - $K_s = 1000N/m$
 - $K_f = 2N/m^2$
-

Derivation and Calculations

Mass spring system considering the effect of friction.



Free body Diagram.



$M \rightarrow$ Mass of the block.

$K_f \rightarrow$ Friction constant.

$K_s \rightarrow$ Spring constant.

$F \rightarrow$ applied Force.

$x(t) \rightarrow$ Displacement from mean position.

Now according to Newton's laws of motion.

Applied Force - Opposing force = Mass \times acceleration.

$$\text{ie } F(t) - \left\{ K_s x(t) + K_f \frac{dx(t)}{dt} \right\} = M \frac{d^2 x(t)}{dt^2}$$

$F(t) \rightarrow$ Input variable $x(t) \rightarrow$ Output variable.

$\therefore F(t) \rightarrow u(t)$ $x(t) \rightarrow y(t)$.

$$u(t) = M \frac{d^2}{dt^2} y(t) + K_s y(t) + K_f \frac{dy(t)}{dt}$$

In Laplace domain.

$$U(s) = \mathcal{L} \left[M \frac{d^2}{dt^2} y(t) + K_s y(t) + K_f \frac{dy(t)}{dt} \right]$$

$$U(s) = M[s^2 Y(s) - s y(0) + y'(0)] + K_s Y(s) + K_f [s Y(s) - y(0)]$$

(Assuming the block of mass M is at rest initially.

ie: Initial conditions are '0'.

$$\text{iii: } y(0) = y'(0) = \underline{0}.$$

$$U(s) = [Ms^2 + K_s + K_f s] Y(s).$$

$$\text{Now } G(s) = \frac{Y(s)}{U(s)}.$$

$$G(s) = \frac{1}{Ms^2 + K_f s + K_s} = \frac{1/M}{s^2 + \left(\frac{K_f}{M}\right)s + \frac{K_s}{M}}$$

Standard second order equation is given by.

$$G(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

$$\omega_n^2 = \frac{K_s}{M}$$

$$\omega_n = \sqrt{\frac{K_s}{M}}$$

$$\boxed{K_s = \omega_n^2 \times M}$$

$$2\zeta\omega_n = \frac{K_f}{M}$$

$$\zeta = \frac{K_f}{2M\sqrt{\frac{K_s}{M}}} = \frac{K_f}{2\sqrt{M \cdot K_s}}$$

$$\text{or } \boxed{K_f = 2\zeta\sqrt{K_s \cdot M}}$$

① Case I: $\zeta = 0.7$, $\omega_n = 10$

$$K_s = \omega_n^2 \times M = 10 \times 10 \times 10 = \underline{1000 \text{ N/m}}$$

$$K_f = 2\zeta \sqrt{K_s M} = 2 \times 0.7 \sqrt{1000 \times 10} = 1.4 \times 100 = \underline{140 \text{ N/m}^2}$$

Poles are located at: P_1, P_2

$$s^2 + 2\zeta \omega_n s + \omega_n^2$$

$$P_1, P_2 = \frac{-2\zeta \omega_n \pm \sqrt{4\zeta^2 \omega_n^2 - 4\omega_n^2}}{2}$$

$$\omega_n = 10$$

$$\therefore P_1, P_2 = \frac{-20\zeta \pm \sqrt{\zeta^2 - 1} \times 20}{2}$$

$$= -10\zeta \pm 10\sqrt{\zeta^2 - 1}$$

$$\zeta = \underline{0.7} \therefore P_1, P_2 = -10(0.7) \pm 10\sqrt{(0.7)^2 - 1}$$

$$= \underline{-7 \pm 7.141j}$$

② Case II: $\zeta = 0.3$, $\omega_n = 10$

$$\therefore K_s = \omega_n^2 \times M = 10 \times 10 \times 10 = 1000 \text{ N/m}$$

$$K_f = 2 \times 0.3 \times \sqrt{10^4} = 60 \text{ N/m}^2$$

$$\text{Poles } P_1, P_2 = -10(0.3) \pm 10\sqrt{0.3^2 - 1}$$

$$P_1, P_2 = \underline{-3 \pm 9.539j}$$

③ Case III: $\zeta = 0.01$, $\omega_n = 10$

$$K_s = 1000 \text{ N/m} \quad K_f = 200 \times 0.01 = 2 \text{ N/m}^2$$

$$P_1, P_2 = -10(0.01) \pm 10\sqrt{0.01^2 - 1}$$

$$P_1, P_2 = \underline{-0.1 \pm 9.999j}$$

MATLAB CODE

Main Code | [solution.m](#)

```
clear all;
close all;
clc;

% DECLARING CONSTANTS STATICALLY
% frictionConstant = [140 60 2];
% springConstant = 1000;

% MORE DYNAMIC CONSTANT CALCULATION
massOfBlock = 10;
zetaValues = [0.7, 0.3, 0.01];
naturalFrequency = 10;

springConstant = naturalFrequency^2 * massOfBlock;
frictionConstant = 2*zetaValues*sqrt(springConstant*massOfBlock);

for index = 1:length(frictionConstant)
    figure;
    G = tf([1], [massOfBlock frictionConstant(index) springConstant]);
    response = stepplot(G, "m");
    grid on;
    legend(strcat("ζ = ", num2str(zetaValues(index))), ", ωn = 10");
    disp(strcat("Time Domain Parameters of transfer function with ", "ζ = ",
num2str(zetaValues(index)), ", ωn = 10"))
    disp(stepinfo(G));

    response.showCharacteristic('PeakResponse');
    response.showCharacteristic('RiseTime');
    response.showCharacteristic('SettlingTime');
    response.showCharacteristic('SteadyState');

    setAxisLimits(axis);
end
```

Helper Snippets

1. [setAxisLimits.m](#)

```
% THIS SNIPPET IS TO ADD PADDING TO THE PLOT

function setAxisLimits(axisData, padding)
    % RELATIVE TO THE OVERALL PLOT
    % 0.1 IS 10% AND 0.5 IS 50%

    arguments
        axisData;
        padding = 0.05; % PADDING DEFALUTS TO 5%
    end

    axisLength = axisData(2) - axisData(1);
    axisHeight = axisData(4) - axisData(3);
    axis([axisData(1) - padding * axisLength axisData(2) + padding * axisLength
axisData(3) - padding * axisHeight axisData(4) + padding * axisHeight]);
end
```

2. [labelOnPlot.m](#)

```
function labelOnPlot(A)
    text(real(A(1)), imag(A(1)), strcat("\rightarrow", num2str(A(1))));
    text(real(A(2)), imag(A(2)), strcat("\rightarrow", num2str(A(2))));
end
```

RESULTS

Command Window Output

G =

$$\frac{1}{10 s^2 + 140 s + 1000}$$

Continuous-time transfer function.

Time Domain Parameters of transfer function with $\zeta = 0.7$, $\omega_n = 10$

RiseTime: 0.2127
SettlingTime: 0.5979
SettlingMin: 9.0010e-04
SettlingMax: 0.0010
Overshoot: 4.5986
Undershoot: 0
Peak: 0.0010
PeakTime: 0.4408

G =

$$\frac{1}{10 s^2 + 60 s + 1000}$$

Continuous-time transfer function.

Time Domain Parameters of transfer function with $\zeta = 0.3$, $\omega_n = 10$

RiseTime: 0.1324
SettlingTime: 1.1230
SettlingMin: 8.6139e-04
SettlingMax: 0.0014
Overshoot: 37.1410
Undershoot: 0
Peak: 0.0014
PeakTime: 0.3224

G =

$$\frac{1}{10 s^2 + 2 s + 1000}$$

Continuous-time transfer function.

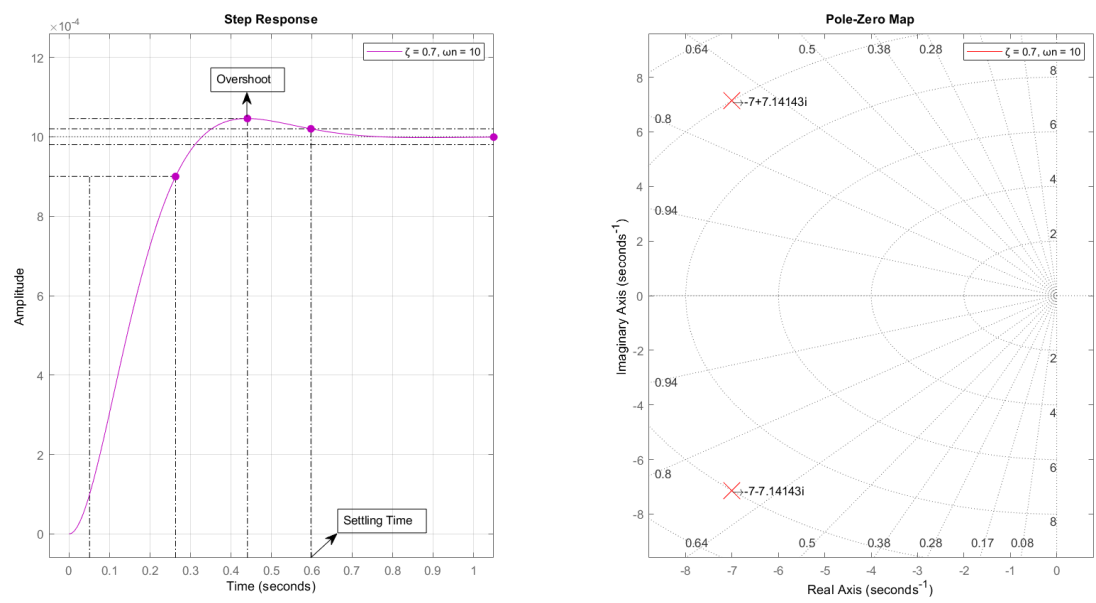
Time Domain Parameters of transfer function with $\zeta = 0.01$, $\omega_n = 10$

RiseTime: 0.1050

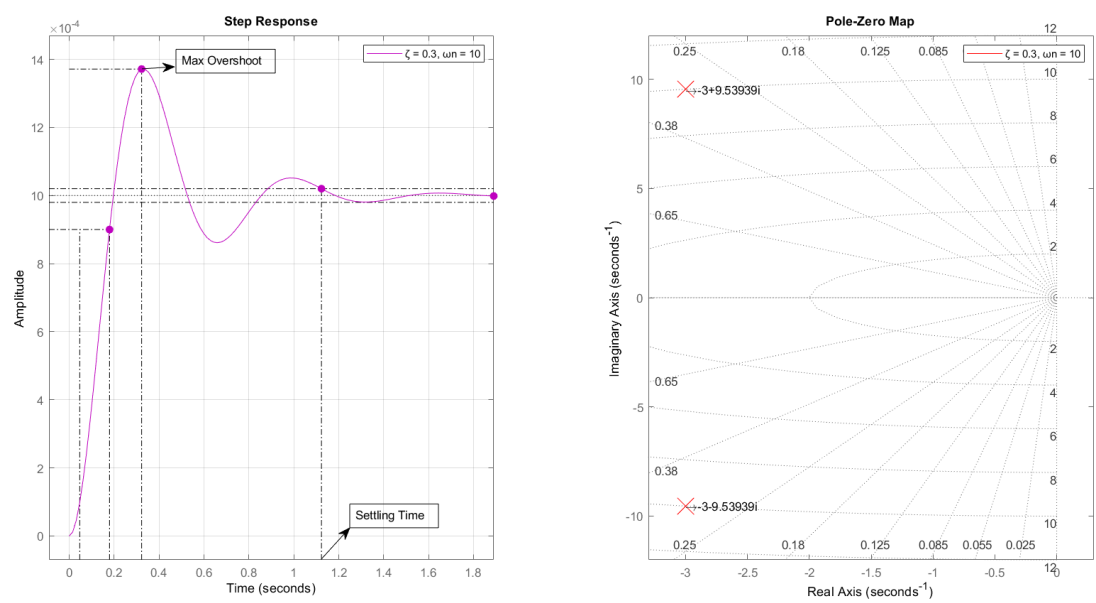
SettlingTime: 38.9674
SettlingMin: 6.0902e-05
SettlingMax: 0.0020
Overshoot: 96.9071
Undershoot: 0
Peak: 0.0020
PeakTime: 0.3142

Plots

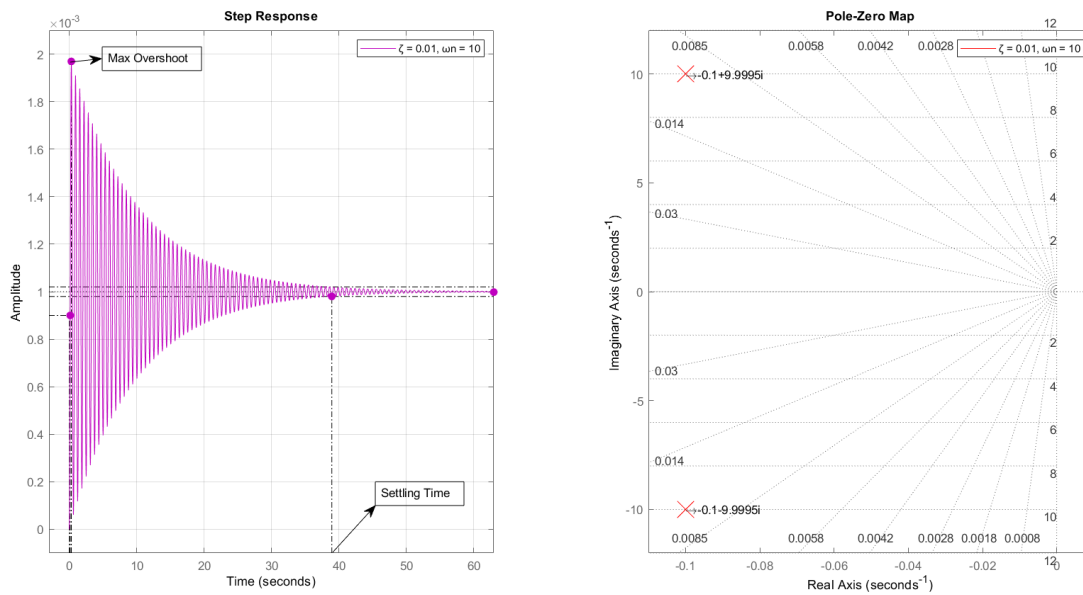
Case 1 Plot



Case 2 Plot



Case 3 Plot



Conclusion/Inference

From the above results and observations, we can conclude the following

- The system in all the 3 cases is **stable** as the poles of the system in all the 3 cases are to the **left** of the Imaginary Axis on the **S-plane**
- Friction co-efficient K_f is directly proportional to ζ , hence as ζ decreases, K_f also decreases.
- As the value of friction co-efficient decreases, that is when the effect of friction decreases, the number of oscillations increases, which means, there is an increase in the **settling time**, and also the **maximum overshoot** is relatively **higher** as seen in case 3 with $\zeta=0.01$.