Table of Contents


```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% NAME: TOMOKI KOIKE
% CLASS: MA266-074
% PROFESSOR: DR. MARIANO
%
% DESCRIPTION: THIS PROGRAM PLOTS THE GRAPH OF A LINEAR SPRING-MASS
% SYSTEM TO FIGURE OUT THE DIFERENTIAL EQUATION FOR THE PHYSICAL
% MOTION OF THE SYSTEM.
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

QUESTION #1

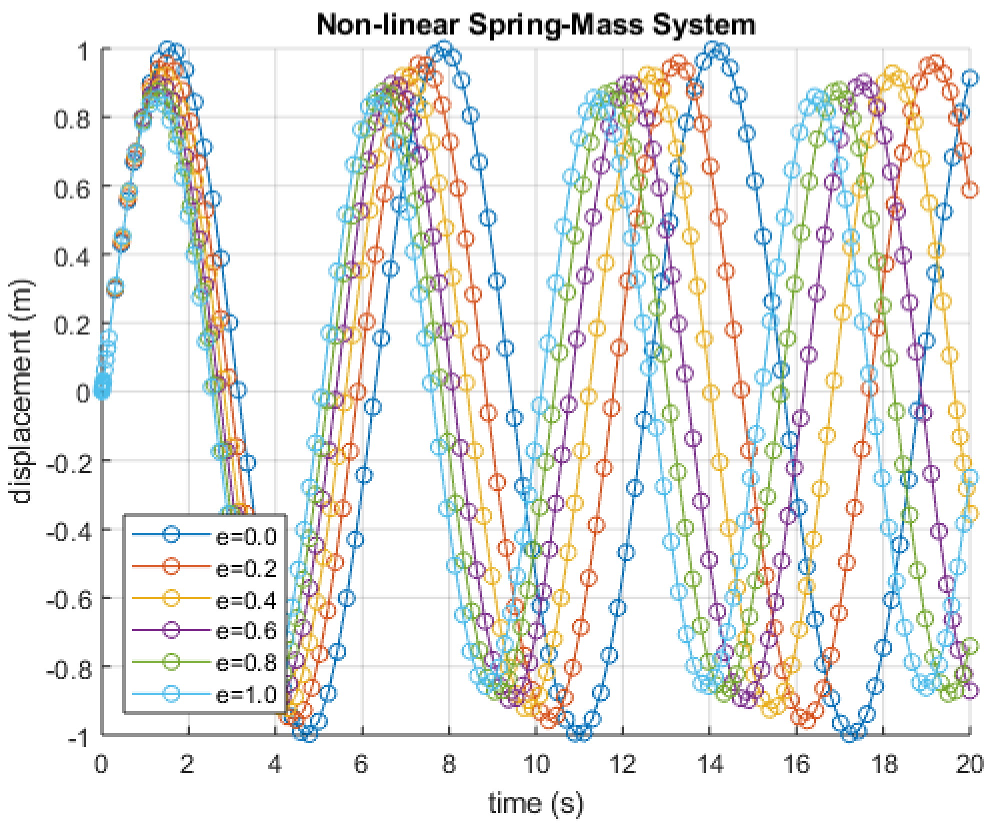
Let e = 0.0, 0.2, 0.4, 0.6, 0.8, 1.0 and plot the solutions of the above initial value problem for 0 ? t ? 20. Estimate the amplitude of the spring. Experiment with various e > 0. What appears to happen to the amplitude as e ? ?? Let μ + denote the first time the mass reaches equilibrium after t = 0. Estimate μ + when e = 0.0, 0.2, 0.4, 0.6, 0.8, 1.0. What appears to happen to μ + as e ? ? ?

CALCULATIONS

```
figure
for e = 0:0.2:1
    [t,u] = ode45(@(t,u) up(t,u,e), [0,20], [0;1]);
    % plotting
    hold on
    plot(t,u(:,1),'-o');
    title('Non-linear Spring-Mass System');
    xlabel('time (s)');
    ylabel('displacement (m)');
    grid on;
end
legend('e=0.0','e=0.2','e=0.4','e=0.6','e=0.8','e=1.0','location','southwest');
hold off

fprintf("As e goes to infinity the amplitude decrease and the
 frequency\n");
fprintf("increases making the oscillation more and more dense and at
\n");
```

```
fprintf("infinity it seems to become like a straight line.\n");

fprintf("from the graph the µ value is in the range of t=2~4, and the
 value\n");
fprintf("decreases as the e increases.\n");
```

As e goes to infinity the amplitude decrease and the frequency increases making the oscillation more and more dense and at infinity it seems to become like a straight line.
from the graph the µ value is in the range of t=2~4, and the value decreases as the e increases.



QUESTION #2

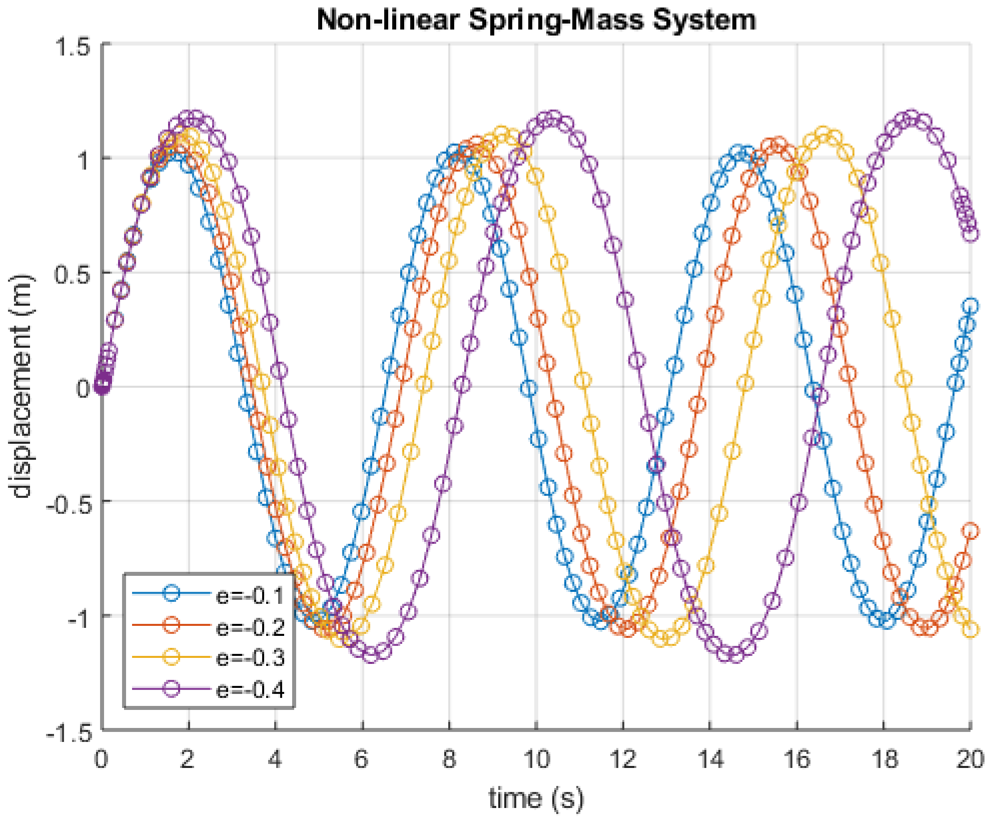
2. Let e = ?0.1, ?0.2, ?0.3, ?0.4 and plot the solutions of the above initial value problem for 0 ? t ? 20. Estimate the amplitude of the spring. Experiment with various e < 0. What appears to happen to the amplitude as e ? ??? Let µ ? denote the first time the mass reaches equilibrium after t = 0. Estimate µ ? when e = ?0.1, ?0.2, ?0.3, ?0.4. What appears to happen to µ ? as e ? ???

CALCULATIONS

```
figure
for e = -0.1:-0.1:-0.4
    [t,u] = ode45(@(t,u) up(t,u,e), [0,20], [0;1]);
    % plotting
```

```
    hold on
    plot(t,u(:,1),'-o');
    title('Non-linear Spring-Mass System');
    xlabel('time (s)');
    ylabel('displacement (m)');
    grid on;
end
legend('e=-0.1','e=-0.2','e=-0.3','e=-0.4','location','southwest');
hold off

fprintf("\nAs e goes to infinity the amplitude increase and the
 frequency\n");
fprintf("decreases making the oscillation more and more dense and at
\n");
fprintf("infinity it seems to become like a straight line.\n");

fprintf("From the graph the μ value is in the range of t=2~4, and the
 value\n");
fprintf("increases as the e decreases.\n");
```

As e goes to infinity the amplitude increase and the frequency decreases making the oscillation more and more dense and at infinity it seems to become like a straight line.
From the graph the μ value is in the range of t=2~4, and the value increases as the e decreases.



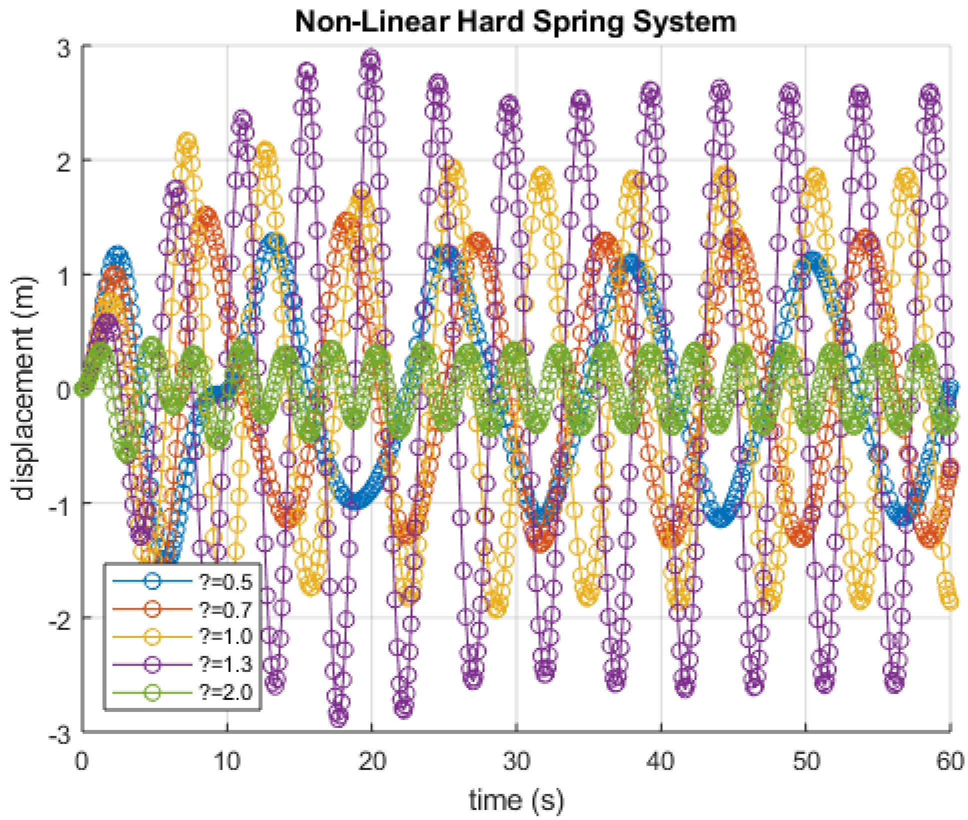
QUESTION #3

Given that a certain nonlinear hard spring satisfies the initial value problem <upp + 1/5 * up + (u + 1/5 * u^3) = cos?t <u(0) = 0, up(0) = 0; plot the solution u(t) over the interval 0 ? t ? 60 for ? = 0.5, 0.7, 1.0, 1.3, 2.0. Continue to experiment with various values of ?, where 0.5 ? ? ? 2.0, to find a value ? ? for which `u(t)` is largest over the interval 40 ? t ? 60

CALCULATIONS

```
figure
for omega = [0.5, 0.7, 1.0, 1.3, 2.0]
    [t,u] = ode45(@(t,u) up2(t,u,omega), [0 60],[0 0]);
    %plotting
    hold on
    plot(t,u(:,1),'-o');
    title('Non-Linear Hard Spring System');
    xlabel('time (s)');
    ylabel('displacement (m)');
    grid on;
end
lgd = legend('?=0.5','?=0.7','?=1.0','?=1.3','?
=2.0','location','southwest');
lgd.FontSize = 8;
hold off;

% figuring out the omega* that has the maximum |u(t)| over the
% range of 40<=t<=60 where 0.5<=omega<=2.0;
n = 1; %index

for omega = 0.5:0.001:2.0
    %solve equation
    [t,u] = ode45(@(t,u) up2(t,u,omega), [0 60],[0 0]);
    %find absolute maximum of u(t)
    max_u1_u2 = max(u);
    max_u1 = max_u1_u2(1,1);
    min_u1_u2 = min(u);
    min_u1 = min_u1_u2(1,1);
    abs_min_u1 = abs(min_u1);
    max_possible = [max_u1; abs_min_u1];
    loop_max(n) = max(max_possible);
    n = n + 1;
end

[final_max, nth] = max(loop_max);
omega_sharp = 0.5 + 0.001 * (nth - 1);

fprintf("\nThe omega at which the |u(t)| is at it's maximum is %f, and
 the maximum value of |u(t)| is %f"...
    , omega_sharp, final_max);
```

The omega at which the |u(t)| is at it's maximum is 1.405000, and the maximum value of |u(t)| is 3.071212



ACADEMIC INTEGRITY

```
PS07_academic_integrity_koike("Tomoki Koike")
```

I am submitting code that is my own original work. I have not used source code, either modified or unmodified, obtained from any unauthorized source. Neither have I provided access to my code to any peer or unauthorized source. Signed,
<Tomoki Koike>

Published with MATLAB® R2018a