Lab 2 - Simulink Exercise

Table of Contents

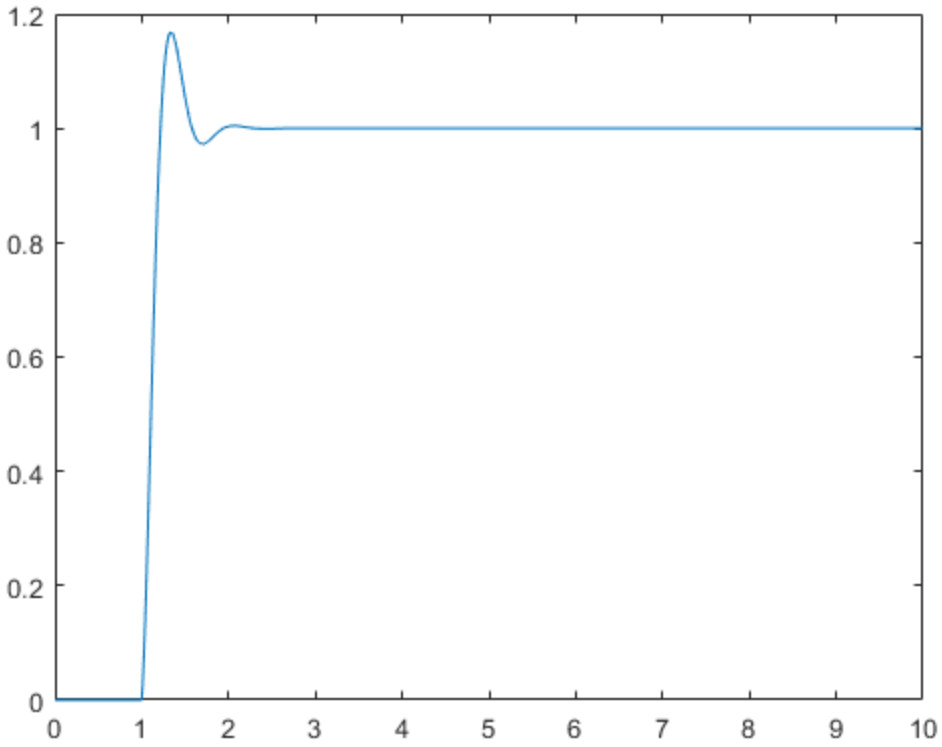
Justin Garcia Kim Kuchinka Due 9/21/2016

Summary

In this lab we began learning how to model differential equations by using simulink to represent the equation physically as a control loop. After we determined how to make the loop on simulink we ran it and were able to see the results on the scope in simulink as well as push the data back to the workspace and call up a plot of the information. We also got more pratice with using for loops and displaying information on plots and in the command window. We learned that using a clock and sending variables to the workspace is best done in an array so that both variables have the same size and plotting is easier. We also learned that when we are calling a variable that it is necessary to define it again, even if the definition is not new. The end result of this lab is that we learned a fast way to simulate a mass-spring-damper system with the ability to change the inputs and constants of the system and display our findings.

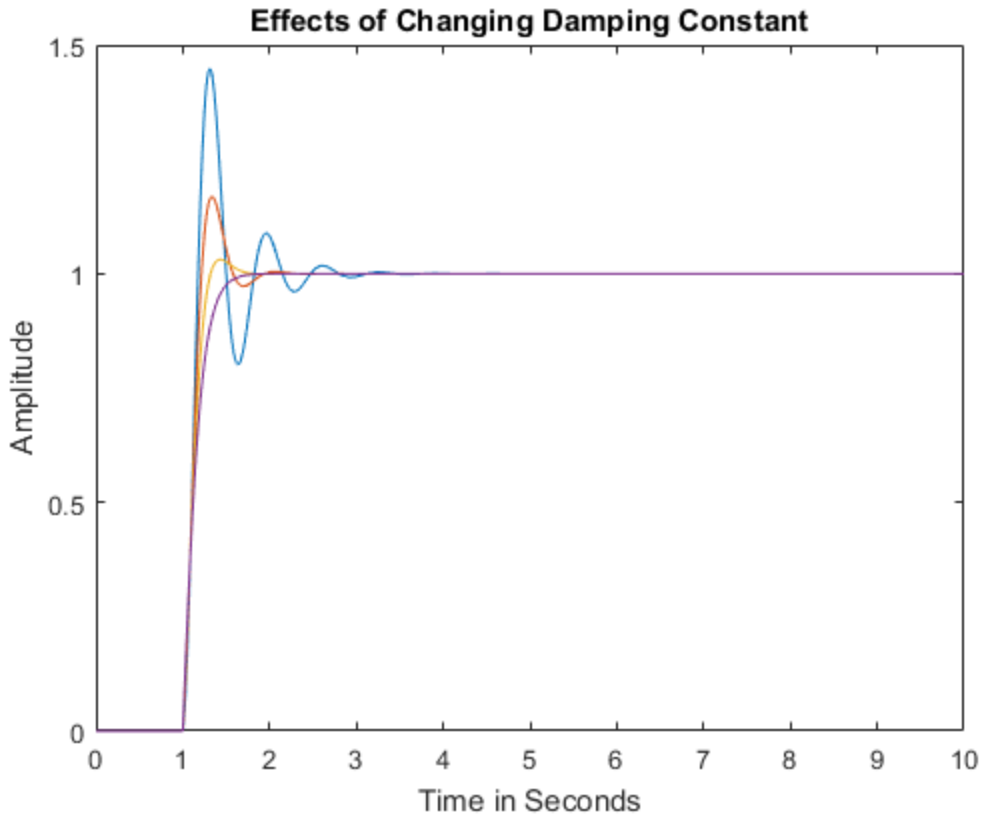
Exercise 1

```
K = 1000;
C = 100;
m = 10;

sim('Simulink_Lab2')
plot(T,X);
```



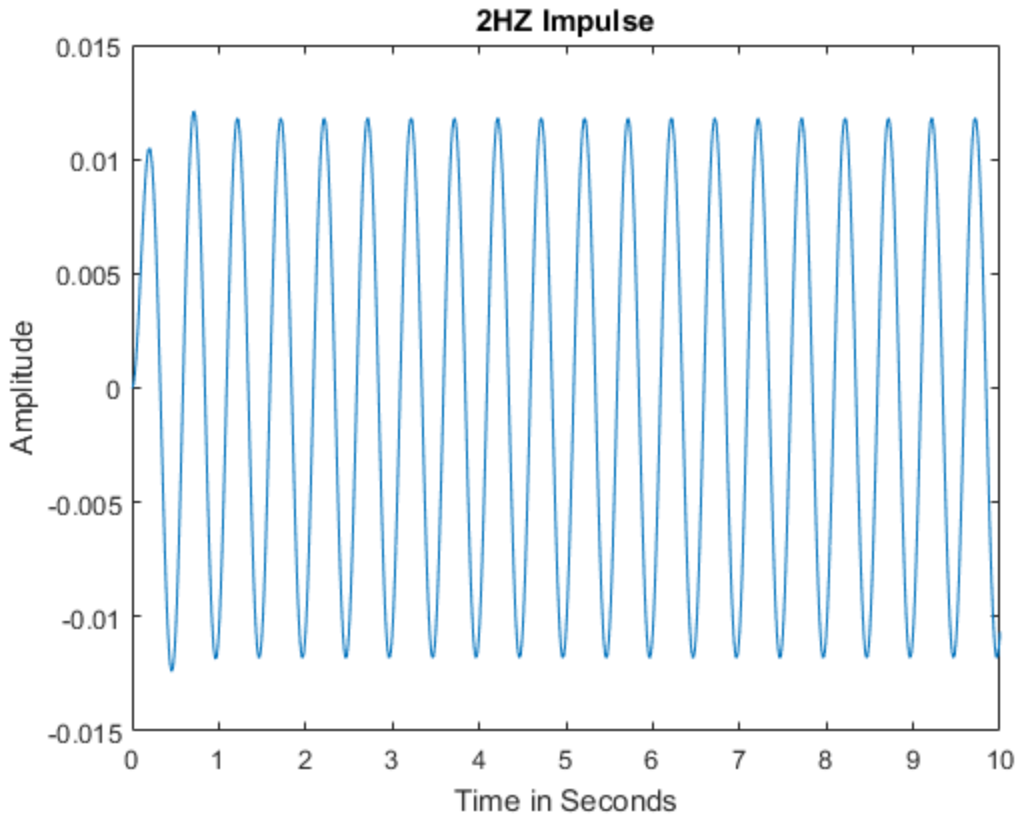
Exercise 2 - Changing the damping constant.

```
for C = 50:50:200
    sim('Simulink_Lab2')

    plot(T,X);
    hold on
end
    title('Effects of Changing Damping Constant')
    xlabel('Time in Seconds');
    ylabel('Amplitude')
    hold off
```



Exercise 3 -

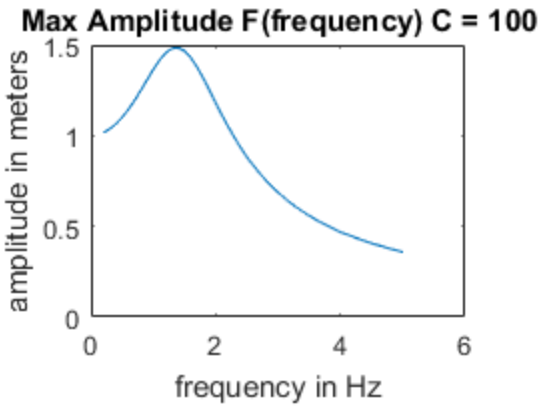
```
C = 100;
freq = 2;
time_t = 0:0.001:10;
signal_y = 0.01*sin(2*pi*freq*time_t);
y_input = [time_t',signal_y'];
sim('Simulink_Lab2_2')
plot(T,X);
title('2HZ Impulse')
xlabel('Time in Seconds');
ylabel('Amplitude')
```



Exercise 4

```
 C = 100;
 amp_y = .01;
 freq = 0.2:0.1:5;
%  Amp_ratio = zeros(1, length(freq));
 amp_x = 0;
 for i = 1:length(freq)
     signal_y = 0.01*sin(2*pi*freq(i)*time_t);
     y_input = [time_t',signal_y'];
     sim('Simulink_Lab2_2')
     amp_x = max(X(200:1001));

     Amp_ratio(i) = amp_x/amp_y;
%      clear amp_x
     subplot(2,2,1)
     plot(freq,Amp_ratio);
     title('Max Amplitude F(frequency) C = 100')
     xlabel('frequency in Hz')
     ylabel('amplitude in meters')

 end

resonant_frequency = max(Amp_ratio);
display(resonant_frequency)
```

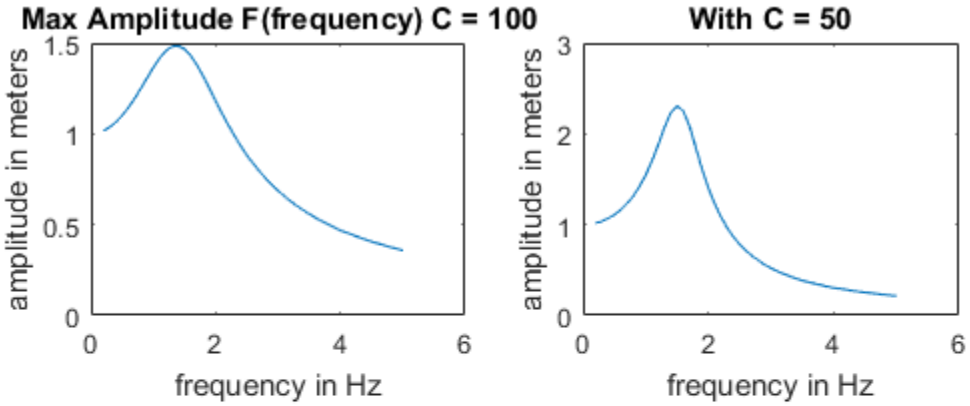
```
resonant_frequency =

    1.4850
```




```
C =50;
amp_x2 = 0;
Amp_ratio2 = zeros(1, length(freq));
 for i = 1:length(freq)
     signal_y = 0.01*sin(2*pi*freq(i)*time_t);
     y_input = [time_t',signal_y'];
     sim('Simulink_Lab2_2')
     amp_x2 = max(X(200:1001));

     Amp_ratio2(i) = amp_x2/amp_y;
%      clear amp_x
     subplot(2,2,2)
     plot(freq,Amp_ratio2);
     title('With C = 50')
     xlabel('frequency in Hz')
     ylabel('amplitude in meters')

 end

resonant_frequency2 = max(Amp_ratio2);
```

```
display(resonant_frequency2)
```

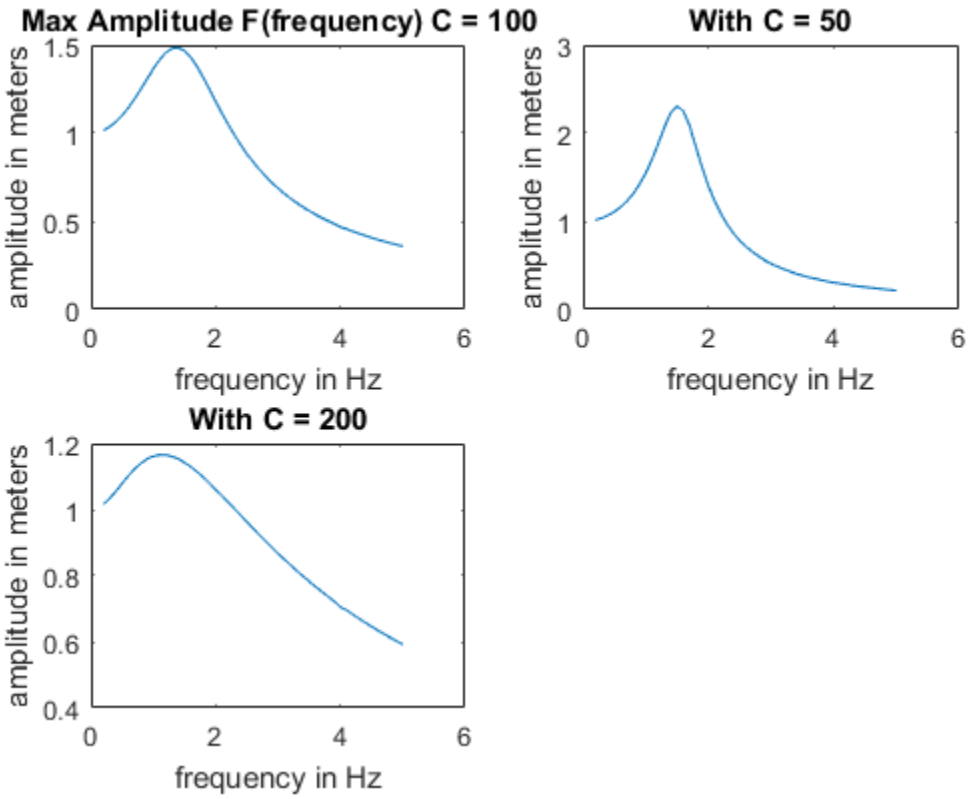
```
resonant_frequency2 =

    2.3068
```




```
C = 200;
amp_x3 = 0;
Amp_ratio3 = zeros(1, length(freq));
 for i = 1:length(freq)
     signal_y = 0.01*sin(2*pi*freq(i)*time_t);
     y_input = [time_t',signal_y'];
     sim('Simulink_Lab2_2')
     amp_x3 = max(X(200:1001));

     Amp_ratio3(i) = amp_x3/amp_y;
%      clear amp_x
     subplot(2,2,3)
     plot(freq,Amp_ratio3)
     title('With C = 200')
     xlabel('frequency in Hz')
     ylabel('amplitude in meters')

 end
```

```
resonant_frequency3 = max(Amp_ratio3);
display(resonant_frequency3)
```

```
resonant_frequency3 =

    1.1656
```




```
open_system('Simulink_Lab2')
```

```
open_system('Simulink_Lab2_2')
```

Published with MATLAB® R2016a