Brittany Duffy
EE 224: Section C
Due: 8/6/2012

Lab 01

Introduction

For lab 01, we were to get familiar with MATLAB functions and processes. My peers and I did this by doing various exercises found at the beginning of the lab hand-out. After doing these warm-up exercises, my main focus was on lab section three. As you will find below, exercise three consisted of working with sinusoids and complex numbers using MATLAB.

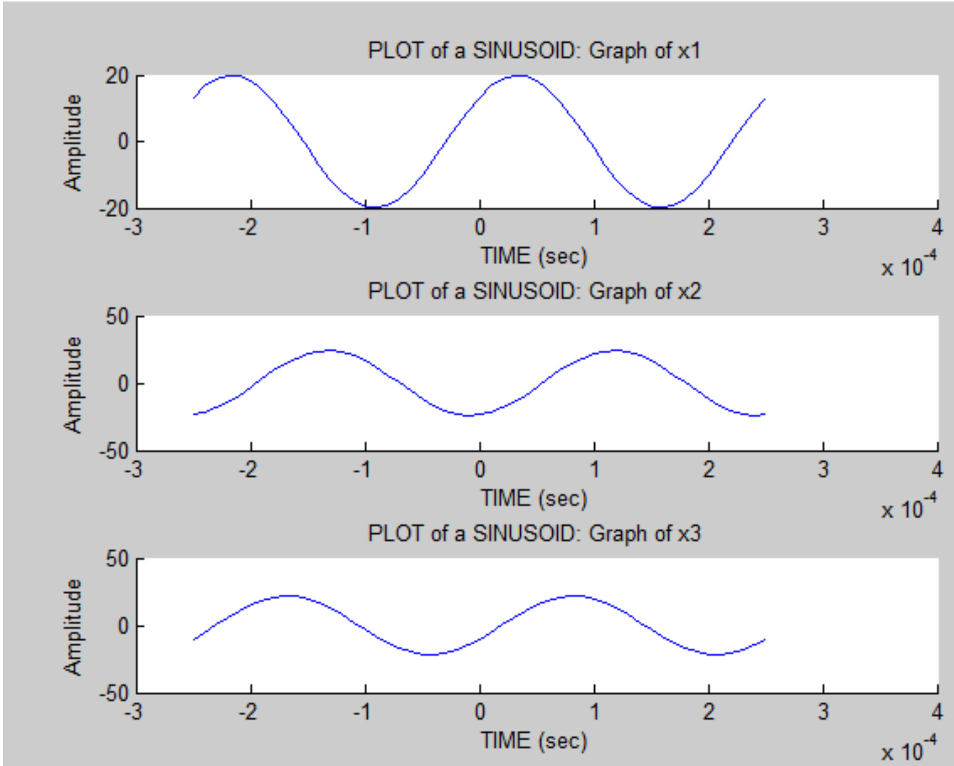
Lab Exercise 3: Manipulating Sinusoids with MATLAB

During exercise 3, we were to generate a time vector that covers the range of t that produces two cycles of 4000 Hz sinusoids. The variable "t" was created to have a large range in order to have at least 25 samples per period. This period was calculated as 1/frequency, or 1/4000. After this step, A1 was assigned to my age which doubles as the amplitude for the first graph. A2 is assigned to 1.2*A1 and becomes the amplitude of the second graph. In addition, 'M' is my month of birth and 'D' is my day of birth. 'M' and 'D' are related to the time shift as you will see later in this code. All of these variables aid in developing new sinusoidal equations: x1, x2, and x3. Please view this script:

```
T = 1/4000;                 %T=1/f
Step =1(4000*25);
tt = -T:Step:T;
A1 =20;                     %A1=age
A2 =1.2*A1;                 %A2=1.2*A1
M = 9;                      %M=month of bday
D = 27;                     %D=day of bday
tm1=(37.2/M)*T;
tm2=-(41.3/D)*T;
x1=A1*cos(2*pi*4000*(tt-tm1));
x2=A2*cos(2*pi*4000*(tt-tm2));
x3=x1+x2;

subplot(3,1,1)  %3 rows, 1 column of graph, 1st graph to make separate plots
hold on
plot(tt, x1)
title('PLOT of a SINUSOID: Graph of x1')
xlabel('TIME (sec)')
ylabel('Amplitude')

subplot(3,1,2)
hold on
plot(tt, x2)
title('PLOT of a SINUSOID: Graph of x2')
xlabel('TIME (sec)')
ylabel('Amplitude')

subplot(3,1,3)
hold on
plot(tt, x3)
title('PLOT of a SINUSOID: Graph of x3')
xlabel('TIME (sec)')
ylabel('Amplitude')
```

Brittany Duffy
EE 224: Section C
Due: 8/6/2012

After running the script, the figure below was displayed:



Lab Exercise 3.1: Theoretical Calculations

Once the sinusoid graphs were formed, we are able to prove they are correct. I first found the "time-location of a positive peak" by using the graphs formed. By using MATLAB, I was easily able to find this information:

```
A1                             %A1=20
A2                             %A2=24
A3                             %A3=44
tm1                            %tm1=.0010
tm2                            %tm2=-3.8241e-04
phase_shift1=2*pi*tm1/T        %phase_shift1= 25.9705
phase_shift2=2*pi*tm2/T        %phase_shift2= -9.6109
x1=A1*cos(2*pi*4000*tm1)       %x1= 13.3826
x2=A2*cos(2*pi*4000*(3.8241e-04))    %x2= -23.5850
x3=x1+x2                       %x3= -10.2024
A3=A1+A2                       %A3=44
```

The time shift of each function can be found by multiplying this amplitude by the step amount and then adding the min value.

For x1: 20*[1/(4000*25)]+ 1/4000 = 9/20000
For x2: 24*[1/(4000*25)] +1/4000= 49/100000
For x3: 44*[1/(4000*25)] +1/4000= 69/100000

Brittany Duffy
EE 224: Section C
Due: 8/6/2012

Phase shift can be found by f*Δt with frequency being 4000 Hz:

Phase shift1 = 1.8 rad
Phase shift2 = 1.96 rad
Phase shift3 = 2.76 rad

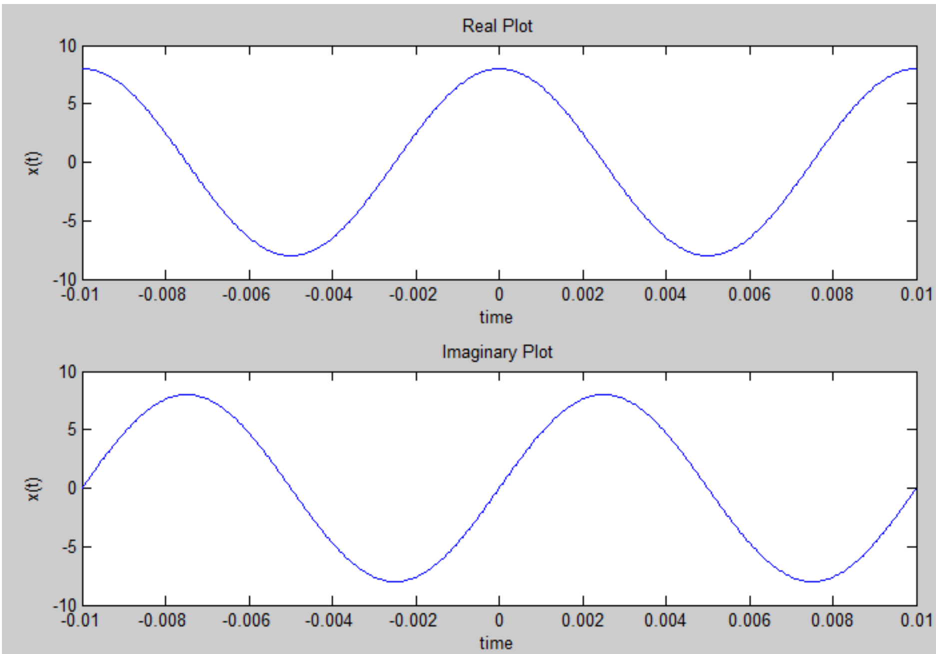
Next, phasor addition theorem lets us find the amplitude and phase of the x3 graph:

20cos(1.8)+j20sin(1.8)+24cos(1.96)+j24sin(1.96)= -13.65+j41.68

<u>Lab Exercise 3.2 Complex Exponential</u>

During this section of Exercise 3, we wrote a code to plot real and imaginary parts of the given complex exponential equation. The two cycles of both real and imaginary parts were then plotted using "subplot". I chose 1/25000 to get an efficient amount of steps.

```
A1=8;
T= .01;
t = -T:(1/25000):T;

x=A1*exp(j*(2*100*pi*t-0.0025));

subplot(2,1,1);
plot(t,real(x));
title('Real Plot');
xlabel('time');
ylabel('x(t)');

subplot(2,1,2);
plot(t,imag(x));
title('Imaginary Plot');
xlabel('time');
ylabel('x(t)');
```



Brittany Duffy
EE 224: Section C
Due: 8/6/2012

Lab Exercise 3.3: MATLAB functions

Finally, our last task was to develop a function by using the "function" command. This function is to take on 3 arguments where one is a matrix and the other 2 are scalars. The function is to return the number of elements in the matrix that are in the interval of the two scalars. You can find the function below:

```
function [y] = count(x,a,b)
n = sum(x<=b & x>=a);
y = sum(n);
disp(x);
disp(y);
end
```

This function displays the matrix and the answer of how many elements in the matrix are between the two scalars.