Taylor Rembos
Lab 2 Part 2
Due May 27, 2017
EEL3135

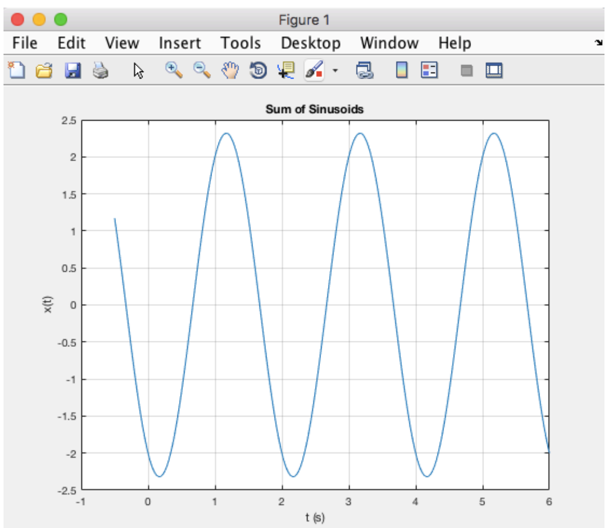
Lab Two Part Two

2.1 Sinusoid Addition

Given the signal $x(t) = Re\{2e^{j\pi t} + 2e^{j\pi(t-1.25)} + (1-j)e^{j\pi t}\}$

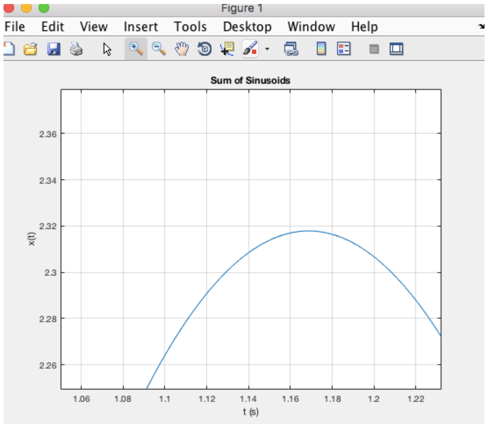
(a) Plot x(t) against t using your syn_sin() function

>> [xx0,tt0] = syn_sin([0.5,0.5,0.5], [2*exp(j*pi),2*exp(j*pi*(1-1.25)),(1-j)*exp(j*pi)],1000,6,(-1/2));

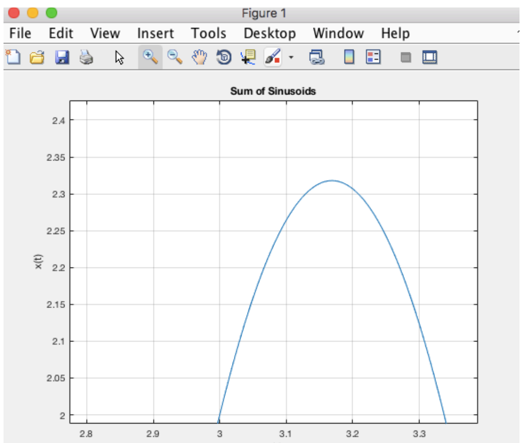


(b) From the plot, measure the frequency, phase, and amplitude by hand.

Period: 1.17 -> 3.17 = 2 s
Frequency: 1/2 = 0.5 Hz
Phase: 1/2 * (1.17) + phi ... = 0.585
Amplitude: 2.32
(see measurements below)



peak (amplitude) ~2.32
peak (time) ~1.17



peak(amplitude) ~2.32
peak(time) ~ 3.17

(c) Use the phasor addition theorem to determine magnitude and phase of x(t)

```
>> xx = 2*exp(j*pi) + 2*exp(j*pi*(1-1.25)) + (1-j)*exp(j*pi)
xx =
   -1.5858 - 0.4142i
>> real(xx)
ans =
  -1.5858
%Amplitude
>> abs(xx)
ans =
   1.6390
%Phase
>> angle(xx)
ans =
```

-2.8861

```
>> % check using zprint – printout complex # in rec and polar form
>> zprint(2*exp(j*pi) + 2*exp(j*pi*(1-1.25)) + (1-j)*exp(j*pi))
 Z =     X    +     jY     Magnitude    Phase    Ph/pi   Ph(deg)
     -1.586     -0.4142       1.639   -2.886   -0.919  -165.36
```

2.2 Fourier Synthesis

To illustrate how we'll use such an equation in Matlab, we'll use the below

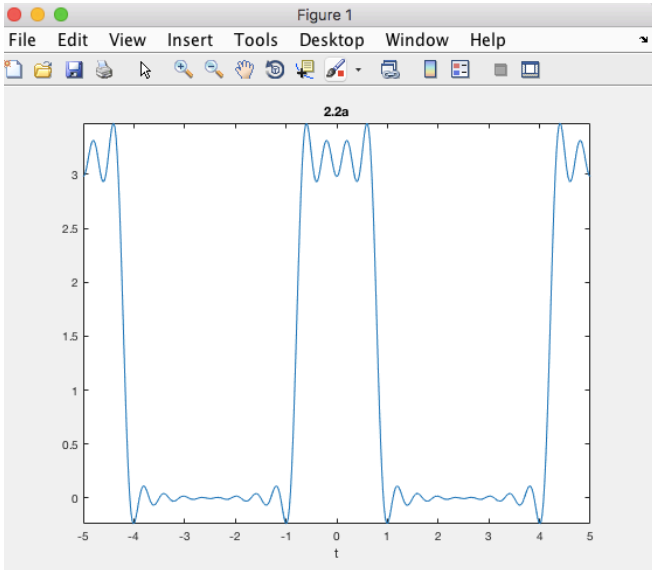
fouriersynth.m:

```
function xt = fouriersynth( ak, N, T0 )
 %FOURIERSYNTH synthesize Fourier Series formula SYMBOLICALLY
 %
 % usage: xt = fouriersynth( ak, N, T0 )
 %
 % ak = (symbolic) formula for the Fourier Series coefficients
 % N = (numeric) use the Fourier coeffs from -N to +N
 % T0 = (numeric) Period in secs
 % xt = (symbolic) signal synthesized
 %
 % example:
 % syms ck k xt
 % ck = sin(k)/k
 % xt = fouriersynth( ck, 4, pi );
 syms t k wwk
 kk = -N:N;
 try
 ak_num = subs( ak, k, kk+((kk==0)+sign(kk))*(1e-9))
 catch
 error('FOURIERSYNTH: ak must use k as its variable')
 end
wwk = exp(j*(2*pi*kk'/T0)*t);
xt = ak_num*wwk; %-- inner product
end
```

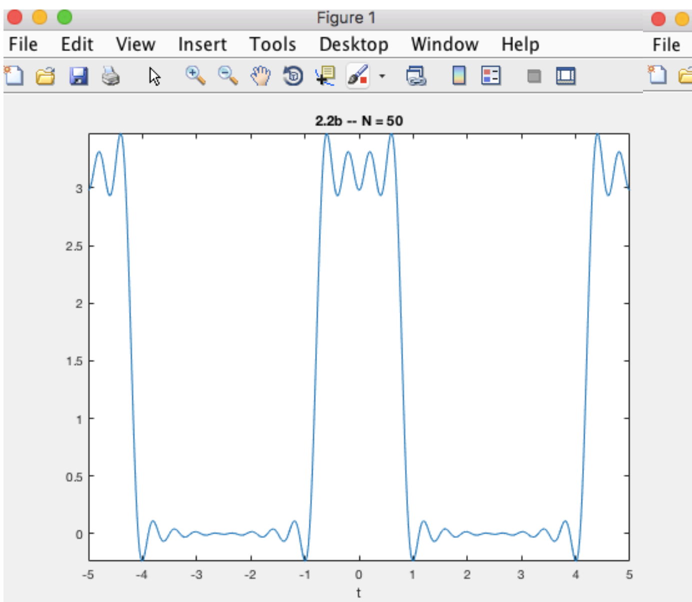
(a) run this code, show the plot

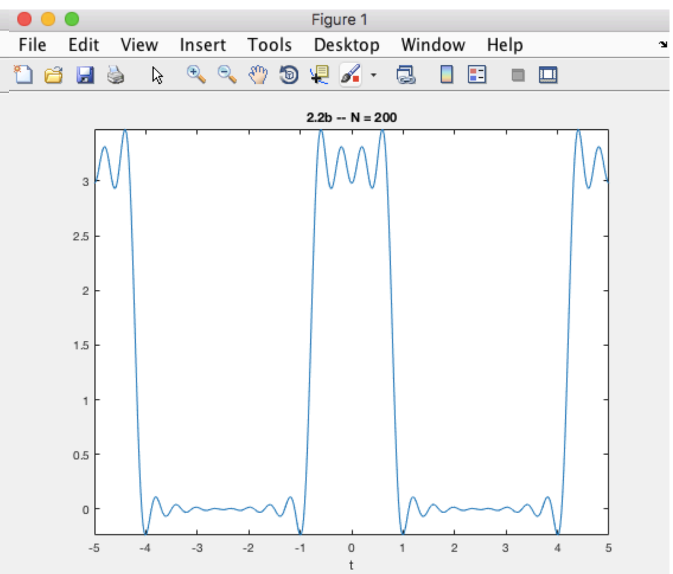
```
>> fouriersynth
>> syms wt t ak k
>> N=12; T0=5;
>> ak = sin(k)/k;
>> wt = fouriersynth(ak,N,T0);

ak_num =

[
(281474976710656*sin(3377699720809347/281474976710656))/3377699720809347,
(281474976710656*sin(3096224744098691/281474976710656))/3096224744098691,
(281474976710656*sin(2814749767388035/281474976710656))/2814749767388035,
(281474976710656*sin(2533274790677379/281474976710656))/2533274790677379,
(281474976710656*sin(2251799813966723/281474976710656))/2251799813966723,
(281474976710656*sin(1970324837256067/281474976710656))/19703248372
```

```
56067,
(281474976710656*sin(1688849860545411/281474976710656))/16888498605
45411,
(281474976710656*sin(1407374883834755/281474976710656))/14073748838
34755,
(281474976710656*sin(1125899907124099/281474976710656))/11258999071
24099,
(281474976710656*sin(844424930413443/281474976710656))/844424930413
443,
(281474976710656*sin(562949953702787/281474976710656))/562949953702
787,
(281474976710656*sin(281474976992131/281474976710656))/281474976992
131, 1000000000*sin(1/1000000000),
(281474976710656*sin(281474976992131/281474976710656))/281474976992
131,
(281474976710656*sin(562949953702787/281474976710656))/562949953702
787,
(281474976710656*sin(844424930413443/281474976710656))/844424930413
443,
(281474976710656*sin(1125899907124099/281474976710656))/11258999071
24099,
(281474976710656*sin(1407374883834755/281474976710656))/14073748838
34755,
(281474976710656*sin(1688849860545411/281474976710656))/16888498605
45411,
(281474976710656*sin(1970324837256067/281474976710656))/19703248372
56067,
(281474976710656*sin(2251799813966723/281474976710656))/22517998139
66723,
(281474976710656*sin(2533274790677379/281474976710656))/25332747906
77379,
(281474976710656*sin(2814749767388035/281474976710656))/28147497673
88035,
(281474976710656*sin(3096224744098691/281474976710656))/30962247440
98691,
(281474976710656*sin(3377699720809347/281474976710656))/33776997208
09347]


>> ezplot(wt, [-T0,T0]);
>> axis tight
>> title('2.2a');
```



(b) Now, run the code again, but this time increase the value of N. Describe how the plot is different. Can you describe how the graph would be if N was infinity?





The plot does not change as N increases. Therefore, the graph would remain the same if N was infinity.

2.3 Multipath Interference

Consider the scenario diagrammed in Fig. 1 where a vehicle traveling on the roadway receives signals from two sources: the ‘direct path’, directly from the transmitter, and the ‘reflected path’, reflected by another object such as a large building. The reflected path and direct path will have different lengths; therefore, the two signals will undergo different time delays.

The length of the direct path is found using the Pythagorean Theorem:

$$d = \sqrt{[(x_2 - x_1)^2 + (y_2 - y_1)^2]}$$

So the distance between the transmitter and the vehicle is :

$$d = \sqrt{[x_v{}^2 + d_t{}^2]}$$

(x_v vehicle, d_t transmitter)

Transmitter location and reflector location are constant.
Transmitter is located at (0,1500) meters and reflector at (100,900) meters.

(1) What is the length of the reflected path as a function of x_v?

```
>> % time delay of direct path
>> syms xv dt
>> d1 = (xv^2 + dt^2)^(1/2)

d1 =

(dt^2 + xv^2)^(1/2)

>> c = 3*10^8

c =

   300000000

>> t1 = d1/c

t1 =

(dt^2 + xv^2)^(1/2)/300000000
```

>> % time delay of reflected path

```
>> syms dyr dxr

>> d2 = ((dt-dyr)^2 + dxr^2)^(1/2) + ((dxr-xv)^2 + dyr^2)^(1/2)

d2 =

(dxr^2 + (dt - dyr)^2)^(1/2) + (dyr^2 + (dxr - xv)^2)^(1/2)

>> t2 = d2/c

t2 =

(dxr^2 + (dt - dyr)^2)^(1/2)/300000000 + (dyr^2 + (dxr -
xv)^2)^(1/2)/300000000
```

>> % difference in time delay

```
>> syms s t
>> rv = s*(t-t1)-s*(t-t2)

rv =

s*(t - (dt^2 + xv^2)^(1/2)/300000000) + s*((dxr^2 + (dt -
dyr)^2)^(1/2)/300000000 - t + (dyr^2 + (dxr - xv)^2)^(1/2)/300000000)
```

The amount of delay (in seconds) can be computed for both paths, because the time delay is the distance divided by the speed of the signal. In this case, the signal travels at the speed of light, which is $3x10^8$ m/s.

What is the time delay of the direct path as a function of x_v? (d1,t1)

What is the time delay of the reflected path as a function of x_v? (d2,t2)

Subtract the above two expressions from each other to express the difference in time delay between the two paths as a function of x_v.

```
>> % transmitter location
>> dt = 1500

dt =
```

```
   1500

>> % reflector location
>> dyr = 900

dyr =

  900

>> dxr = 100

dxr =

  100

>> % direct path, fixed location
>> d1

d1 =

(dt^2 + xv^2)^(1/2)

>> (dt^2 + 0)^(1/2)

ans =

   1500
```

>> % direct path time delay

```
>> t1

t1 =

(dt^2 + xv^2)^(1/2)/300000000

>> 1500/c

ans =

  5.0000e-06
```

```
>> % reflected path equation
>> d2

d2 =

(dxr^2 + (dt - dyr)^2)^(1/2) + (dyr^2 + (dxr - xv)^2)^(1/2)

>> (100^2 + (1500-900)^2)^(1/2) - (900^2 + (100^2))^(1/2)

ans =

 -297.2623
```

>> % reflected path time delay

```
>> t2

t2 =

(dxr^2 + (dt - dyr)^2)^(1/2)/300000000 + (dyr^2 + (dxr -
xv)^2)^(1/2)/300000000

>> d2/c

ans =

(dxr^2 + (dt - dyr)^2)^(1/2)/300000000 + (dyr^2 + (dxr -
xv)^2)^(1/2)/300000000

>> -297.2623/c

ans =

  -9.9087e-07
```

>> % difference in time delay

```
>> (t1/c) - (d2/c)

ans =

(dt^2 + xv^2)^(1/2)/90000000000000000 - (dxr^2 + (dt -
dyr)^2)^(1/2)/300000000 - (dyr^2 + (dxr - xv)^2)^(1/2)/300000000

>> 5.0000e-06 - ans

ans =
```

5.9909e-06

>> % phase difference between two paths
>> 150*2pi
ans =

300pi
>> % total received signal – signal cancellation
>> rv

rv =

s*(t - (dt^2 + xv^2)^(1/2)/300000000) + s*((dxr^2 + (dt - dyr)^2)^(1/2)/300000000 - t + (dyr^2 + (dxr - xv)^2)^(1/2)/300000000)

>> rv = exp(-j*pi*1500)-exp(j*pi*297)

rv =

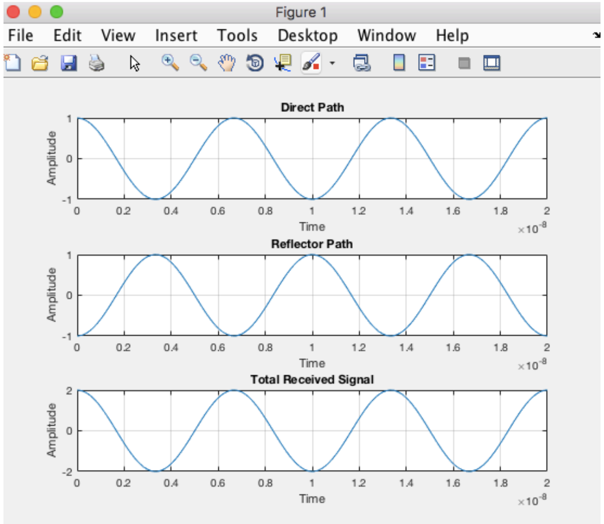
2.0000 + 0.0000i

Using MATLAB, use complex amplitudes to plot three periods of $r_v(t)$ when the vehicle position is $x_v = 0$ meters. Plot 3 periods and then measure the maximum amplitude. How can a single complex addition followed by a magnitude operation be used to find the amplitude of $r_v(t)$?

receiver.m

```
% variables
f = 150*10^6; % MHz
dur = 0.02*10^-6;
t = 0:(1/(f*(10^6))):dur;

% equations
dp = cos(2*pi*f*t-1500*pi); % direct path equation
rp = cos(2*pi*f*t+297*pi); % reflector path equation
rv = dp - rp; % total received signal - subtraction of direct and
reflector

% plots
subplot(311); % 3x1 matrix with 2nd axis for current plot
plot(t,dp)
xlabel('Time');
ylabel('Amplitude');
title('Direct Path');
axis tight
grid on

subplot(312); % 3x1 matrix with 3rd axis for current plot
plot(t,rp)
xlabel('Time');
ylabel('Amplitude');
title('Reflector Path');
axis tight
grid on

subplot(313); %  3x1 matrix with 1st axis for current plot
plot(t,rv)
xlabel('Time');
ylabel('Amplitude');
title('Total Received Signal');
axis tight
grid on
```

You add the sinusoids together and the resulting magnitude of those sinusoids is the amplitude of rv.



Plot signal strength (amplitude) versus vehicle position over the interval from 0 meters to 300 meters. Assume that the signal strength is the peak value of the received sinusoid, $r_v(t)$. Explain how you get the peak value from the complex sinusoid.

What are the largest and smallest values of received signal strength? Why do we get those values? Are there vehicle positions where we get complete signal cancellation, that is, where the received signal is zero? If so, determine those vehicle positions.

Is one of these vehicle positions the one you had calculated by hand above?

<u>amplitude.m</u>

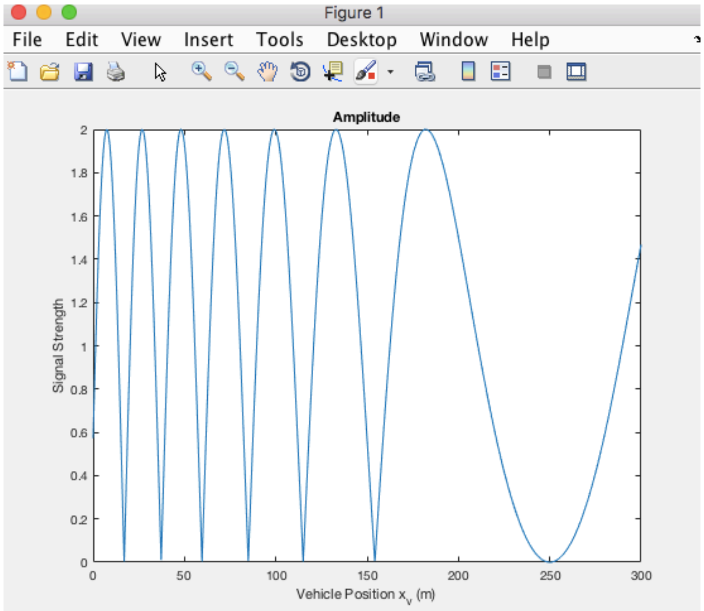
```
function [ A, t1, t2 ] = amplitude(xv,tx,rx,f0)
% xv = vector with x-coordinate of vehicle position
% tx = vector with coordinates of transmitter
% rx = vector with coordinates of reflector
% A = amplitude of xv(t)

c = 3*(10^8);
t1 = sqrt((xv-tx(1)).^2+tx(2)^2)/c; % t1 distance equation
t2 = (sqrt((rx(1)-tx(1))^2 + (tx(2)-rx(2))^2) + sqrt((rx(1)-xv).^2 +
rx(2)^2))/c; % t2 time delay equation
A = abs(exp(-j*2*pi*f0*t1) - exp(-j*2*pi*f0*t2)); % signal strength
```

>> xv = [0:0.1:300]; % interval from **0 meters to 300 meters**

>> [A, t1, t2] = amplitude(xv, [0,1500], [100,900], 150*(10^6)); % **Peak Value:** A = amplitude vector, t1 = 0,1500 – transmitter, t2 = 100,900 – reflector, f0 = frequency 150 MHz

>> plot(xv,A); % **plot** vehicle position vs amplitude

>> ylabel('**Signal Strength**');
>> xlabel('**Vehicle Position** x_v (m)');
>> title('Amplitude');



As seen in the plot above, max signal strength is 2 and min is 0.

When the sinusoids have constructive frequency, the largest values of strength occur. When the sinusoids have destructive frequency, the smallest values of strength occur.

There are vehicle positions where we get complete signal cancellation where the received signal is zero. This occurs when the phasor rv is 0, which occurs when f0*(t2-t1) is an integer value. At about 20m,6 , the first cancellation point occurs. If you take the minimum of the absolute value of f0*(t2-t1)=6 which is 17. This means the first cancellation point is at 17m. This is not one of the positions I had calculated by hand.