EE2703 : Applied Programming Lab Assignment 10 : Spectra of Non-Periodic Signals

Vishnu Varma V
EE19B059

May 5, 2021

Introduction

In the previous assignment we looked at functions that were periodic and extracted their spectra. The approach was:

• Sample the signal so that $f_{Nyquist}$ is met, so that $\triangle f$ is small enough. Generate the frequency axis from $-f_{max}/2$ to $f_{max}/2$, taking care to drop the last term.
• Ensure that the signal starts at $t = 0+$ and ends at $t = 0-$
• Use $2k$ samples
• Obtain the DFT. Rotate the samples so that they go from $f = -f_{max}/2$ to $f = f_{max}/2 - \triangle f$
• Plot the magnitude and phase of the spectrum. Usually we would plot the magnitude in dB and the phase in degrees and the frequency axis would be logarithmic. This is to capture polynomial decay of the spectrum.

Now we are dealing with Non-Periodic Signals, the results with the same procedure followed earlier to non periodic signals is not as expected, So we explore new ways of plotting the spectrum of Non-Periodic Signals

Spectrum of $sin(\sqrt{2}t)$

Following the same procedure as for periodic signals for $sin(\sqrt{2}t)$ yields the following plot:

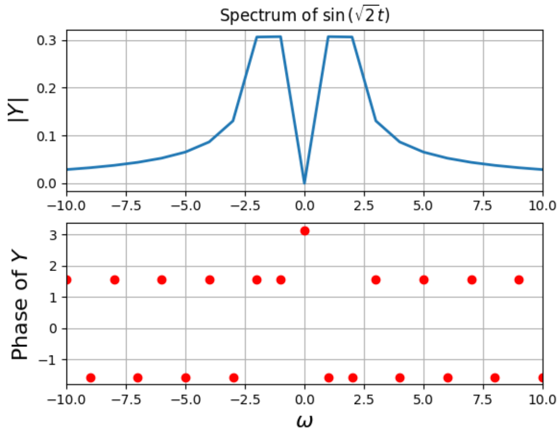

Figure 1: Spectrum of $sin(\sqrt{2}t)$

The plot is not as expected, We expected two spikes but we got two peaks with gradually decaying magnitude. Let us further explore what can be made better to get the expected plot.

Let us plot the function $sin(\sqrt{2}t)$ in seperate intervals dividing the interval -3π to 3π into three intervals of 2π each.

```
t1 = linspace(-pi,pi,65); t1 = t1[:-1]
t2 = linspace(-3*pi,-pi,65); t2 = t2[:-1]
t3 = linspace(pi,3*pi,65); t3 = t3[:-1]
y = sin(sqrt(2)*t1)
figure(3)
plot(t1,y,'b',lw=2)
plot(t2,y,'r',lw=2)
plot(t3,y,'r',lw=2)
ylabel(r"$y$",size=16)
xlabel(r"$t$",size=16)
title(r"$\sin\left(\sqrt{2}t\right)$")
grid(True)
show()
```

Plotting so we obtain the following plot:

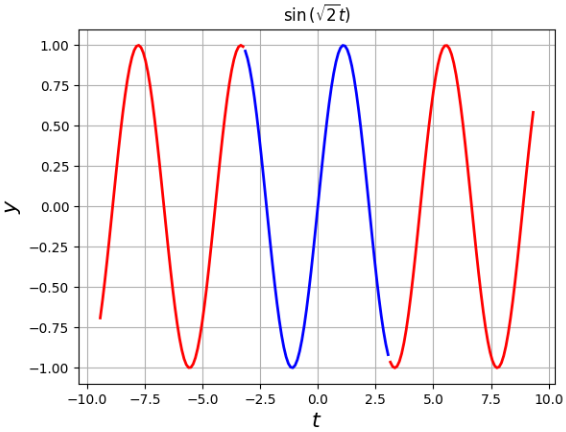

Figure 2: $sin(\sqrt{2}t)$

We can observe that $sin(\sqrt{2}t)$ is periodic but the function between $-\pi$ to π if periodically extended it does not give the actual function.

Let us plot the function with t wrapping every 2π using the following code:

```
t1 = linspace(-pi,pi,65); t1 = t1[:-1]
t2 = linspace(-3*pi,-pi,65); t2 = t2[:-1]
t3 = linspace(pi,3*pi,65); t3 = t3[:-1]
figure(2)
plot(t1,sin(sqrt(2)*t1),'b',lw=2)
plot(t2,sin(sqrt(2)*t2),'r',lw=2)
plot(t3,sin(sqrt(2)*t3),'r',lw=2)
ylabel(r"$y$",size=16)
xlabel(r"$t$",size=16)
title(r"$\sin\left(\sqrt{2}t\right)$")
grid(True)
show()
```

It yields the following plot :

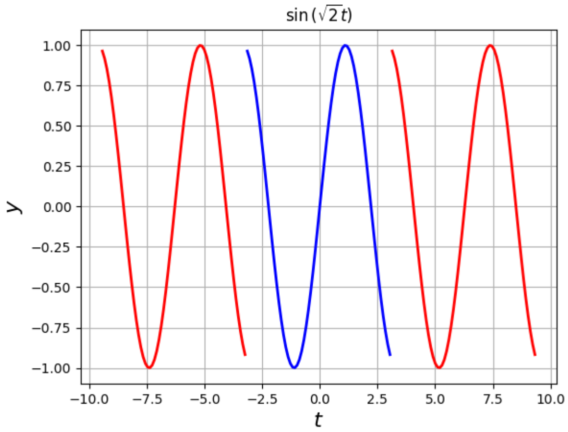

Figure 3: $sin(\sqrt{2}t)$ with t wrapping every 2π

The spectrum of the Box function decays very slowly, as $2/\omega$ Our function is an odd function with a big jump. So considering the periodic ramp :

$$f(t) = t, -\pi < t < \pi$$

Then the fourier series of this ramp is

$$f(t) = 2(\frac{sint}{1} - \frac{sin2t}{2} + \frac{sin3t}{3} - ...)$$

The coefficients decay very slowly. The DFT is just like the fourier series, except that both time and frequency are samples. So, if the time samples are like a ramp, the frequency samples will decay as $1/\omega$. Let us verify this for the ramp itself using the following code:

```
t=linspace(-pi,pi,65);t=t[:-1]
dt=t[1]-t[0];fmax=1/dt
y=t
y[0]=0
y=fftshift(y)
Y=fftshift(fft(y))/64.0
w=linspace(-pi*fmax,pi*fmax,65);w=w[:-1]
figure()
semilogx(abs(w),20*log10(abs(Y)),lw=2)
xlim([1,10])
ylim([-20,0])
xticks([1,2,5,10],["1","2","5","10"],size=16)
ylabel(r"$|Y|$ (dB)",size=16)
title(r"Spectrum of a digital ramp")
xlabel(r"$\omega$",size=16)
grid(True)
show()
```

The plot obtained is as follows:

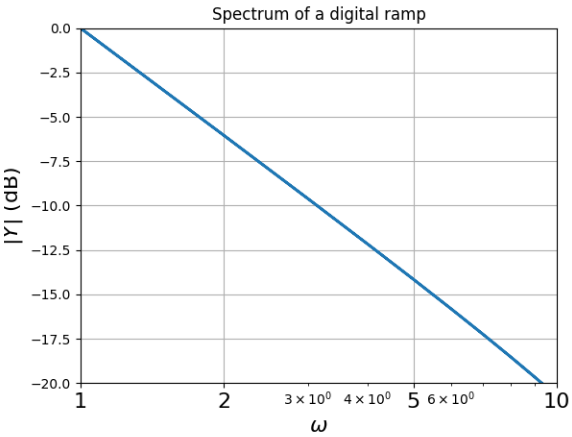

Figure 4: Spectrum of a digital ramp

This problem can be fixed by Windowing, We multiply the function with the Hamming Window defined as:

$$w[n] = \begin{cases} 0.54 + 0.46\ cos(\frac{2\pi n}{N-1}), & |n| < N \\ 0, & otherwise \end{cases}$$

Now the function after windowing is as follows:

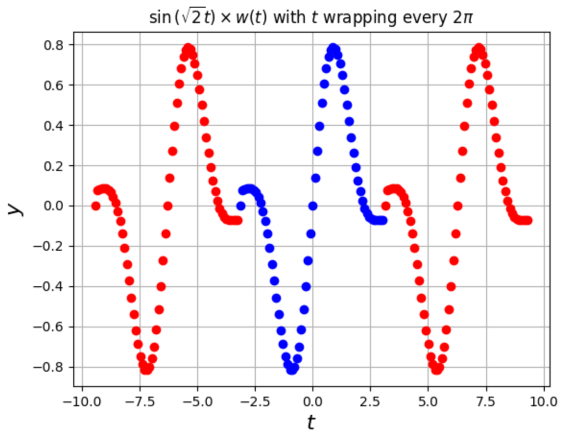

Figure 5: $sin(\sqrt{2}t)$ x $w(t)$ with t wrapping every 2π

Now using the Hamming window to find the DFT as follows:

```
t=linspace(-pi,pi,65);t=t[:-1]
dt=t[1]-t[0];fmax=1/dt
n=arange(64)
wnd=fftshift(0.54+0.46*cos(2*pi*n/63))
y=sin(sqrt(2)*t)*wnd
y[0]=0
y=fftshift(y)
```

```python
Y=fftshift(fft(y))/64.0
w=linspace(-pi*fmax,pi*fmax,65);w=w[:-1]
figure()
subplot(2,1,1)
plot(w,abs(Y),lw=2)
xlim([-8,8])
ylabel(r"$|Y|$",size=16)
title(r"Spectrum of $\sin\left(\sqrt{2}t\right)\times w(t)$")
grid(True)
subplot(2,1,2)
plot(w,angle(Y),'ro',lw=2)
xlim([-8,8])
ylabel(r"Phase of $Y$",size=16)
xlabel(r"$\omega$",size=16)
grid(True)
show()
```

The DFT obtained after windowing is as follows:

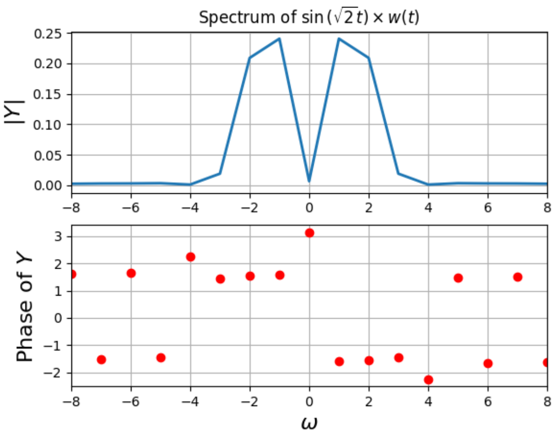

Figure 6: Spectrum of $sin(\sqrt{2}t)$ after windowing

If we use four times the number of points used earlier yields the following plot:

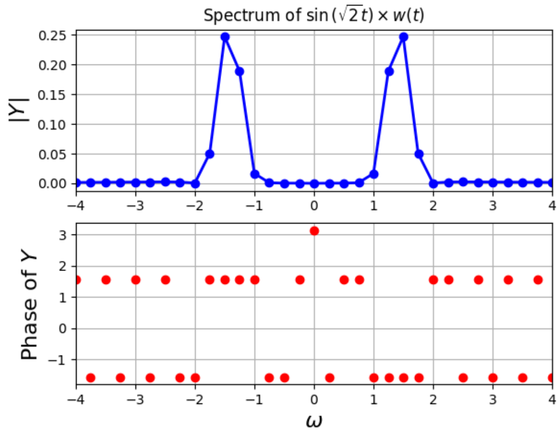

Figure 7: Spectrum of $sin(\sqrt{2}t)$ after windowing

DFT of $cos^3(w_o t)$

Now let us find the DFT of $cos^3(w_o t)$ where $w_o = 0.86$ with and without windowing respectively. The same code used earlier is used replacing the function. The plots are as follows:

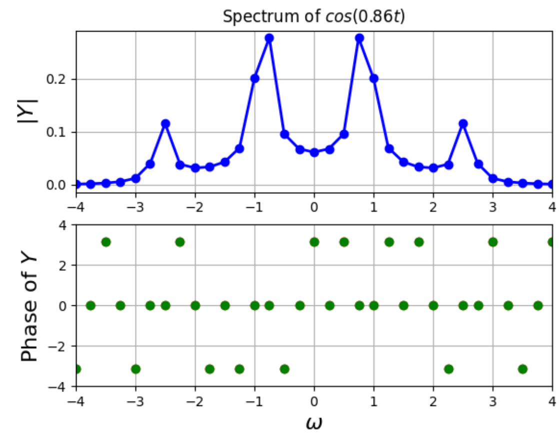

Figure 8: Spectrum of $cos^3(0.86t)$ without windowing

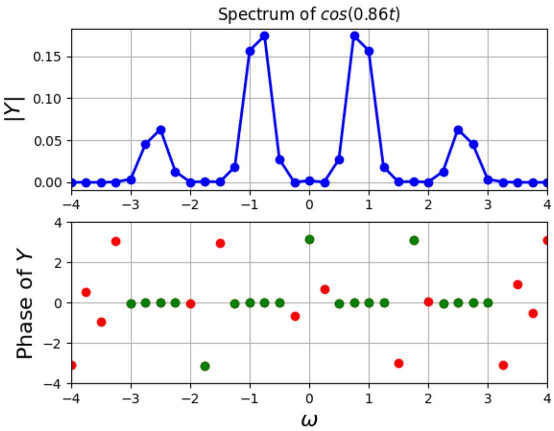

Figure 9: Spectrum of $cos^3(0.86t)$ with windowing

Estimation of Parameters

Given a 128 element vector known to contain $cos(w_o t + \delta)$ for arbitrary δ and $0.5 < \omega < 1.5$. t varies from -π to π. We need to estimate w_o and δ from the DFT of the function. Due to the low sampling rate it is difficult to estimate the frequency from the peak of the DFT, so we try to take the weighted average over a certain number of samples to obtain an estimate on w_o. To estimate the phase of the function we take the mean of a certain number of angles of the function whose magnitude is significant. The DFT obtained for $w_o = 0.8$ and $\delta = 1$ is as follows:

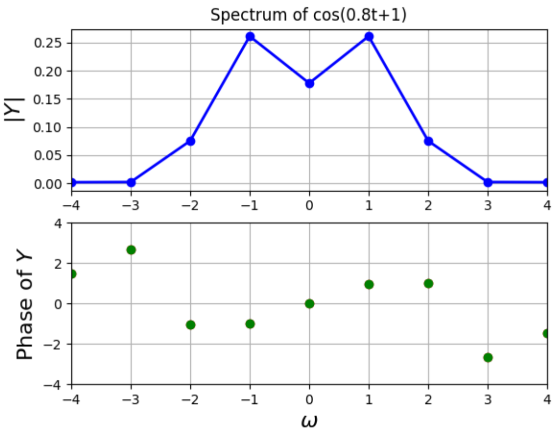

Figure 10: Spectrum of $cos(0.8t + 1)$ without noise

Now adding white gaussian noise to the above function, we try to estimate the parameters again. Noise is generated as follows: 0.1*randn(N), where N is the number of samples. The DFT after adding noise is as follows:

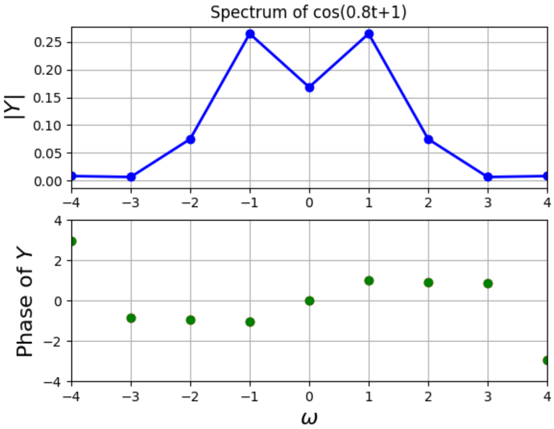

Figure 11: Spectrum of $cos(0.8t + 1)$ with noise

Now using the method as mentioned above yields the following estimation for the parameters and the errors corresponding to them is mentioned.

```
Estimated wo : 0.80974226481489 41
Estimated delta : 0.9837513005659742
Error in estimate of wo: 0.009742264814894042
Error in estimate of delta : 0.016248699434025 85
Estimated wo : 0.8150858889959489
Estimated delta : 1.0084691331286906
Error in estimate of wo: 0.015085888995948 87
Error in estimate of delta : 0.008469133128690576
```

The estimation are not very accurate as expected but precise enough with the given inputs and low sampling rate.

Chirped Signal

Chirped signal is defined as follows:

$$y(t) = cos(16t(1.5 + \frac{t}{2\pi}))$$

we find the DFT of this function for t going from $-\pi$ to π in 1024 steps. Frequency of a chirped signal changes from 16 to 32 continuously. Plotting the DFT of the chirped signal with and without windowing yields the following plots:

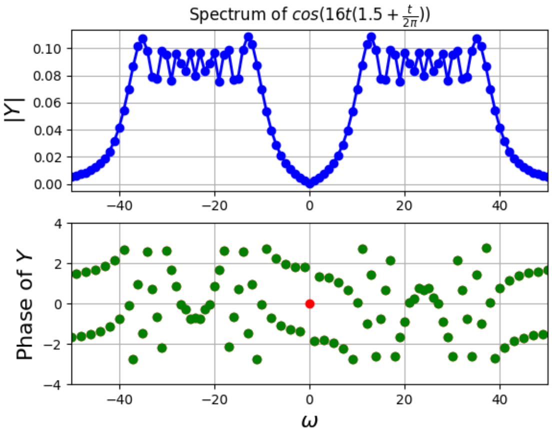

Figure 12: Spectrum of $cos(16t(1.5 + \frac{t}{2\pi}))$ without windowing

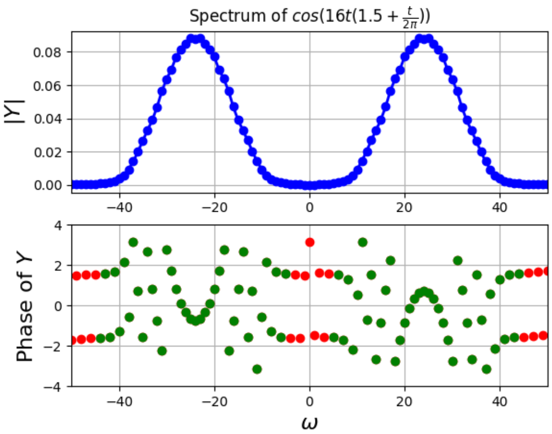

Figure 13: Spectrum of $cos(16t(1.5 + \frac{t}{2\pi})))$ with windowing

Now we break the 1024 length vector into pieces that are 64 samples wide. Find the DFT of each and store as a column in a 2D array. Plotting this array as a surface plot will show how the frequency of the signal varies with time and how the spectrum varies. The Surface plot of Magnitude and Phase of the spectrum is as follows:

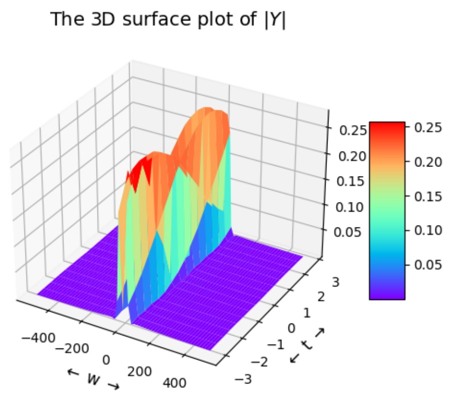

Figure 14: 3D Surface plot of $|Y|$

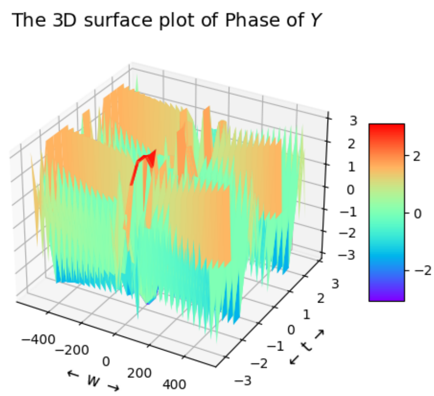

Figure 15: 3D Surface plot of Plase of Y

Conclusion

We analysed the DFT of non periodic signals, if we tried the same method as used for periodic signals the spectrum was a not as expected as the function were not 2π periodic extending periodically gave wrong results. So we used windowing technique to get better results, we used Hamming window for our purpose. We plotted the DFT of some non periodic signals, estimated parameters such as frequency and phase of the signal from the DFT. We plotted the DFT of the chirped signal and as we know its frequency varies continuously with time so plotted a surface plot to examine the same.