EE2703: Applied Programming Lab
Assignment 9
Spectra of Non-Periodic Signals

Arjun Menon Vadakkeveedu
EE18B104
Electrical Engineering, IIT Madras

April 29, 2020

Contents

1 Introduction

In this program, the spectra of non-periodic signals and signals with irrational time periods are estimated using the FFT algorithm, with the help of time-domain windows wherever necessary.

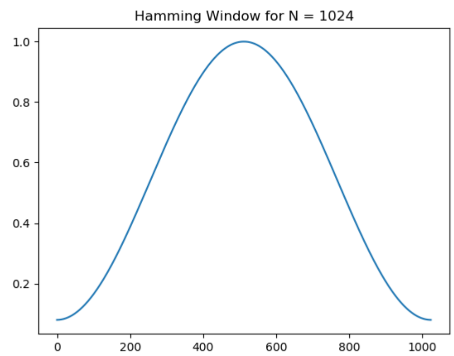
1.1 The Hamming Window

Windowing: Since the DFT is equivalent to the Discrete Time Fourier Series of the N-periodic extension of the input sequence, the DFTs of non-periodic sequences suffer from Gibbs Phenomenon (which occurs due to discontinuities at the ends of the sequence). To remove the high frequency components that arise due to Gibbs' Phenomenon, such non-periodic signals are multiplied by a window in time domain (convolution of FFTs) so that the discontinuities at the extreme points are suppressed.

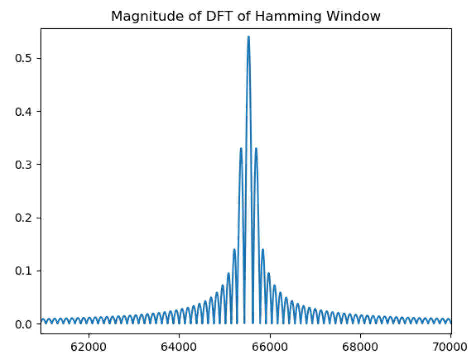
Since Hamming windows have high side-band suppression, they are used here. The window is a positive cycle of a cosine function, offset by a small positive value. It is defined as:

$$w_\alpha[n] = \alpha + (1-\alpha)cos(2\pi\frac{n}{N-1}) \quad 0 \le n \le N-1$$

In this program, $\alpha = 0.54$ is used. $\alpha = 0.5$ is the Hann window.



Hamming Window in time domain $\alpha = 0.54$



Fourier Transform of Hamming window, notice the side-band suppression

1.2 Frequency Estimation of Sinusoid using the FFT

Consider a signal $\cos(\omega_o t + \delta)$ such that both ω_o and δ are noisy. Suppose, $\omega_o \in [0.5, 1.5]$. Then the signal may be sampled for a sufficient duration of

time at a frequency greater than the Nyquist rate to obtain the DFT.
ω_o may be estimated by the peak frequency (> 0) in the spectrum while delta would correspond to the 4-quadrant phase of the FFT at ω_o.

1.3 Chirping

A signal of the form $\sin(\phi(t)+\delta)$ is said to be a chirped signal as its frequency varies with time. The instantaneous frequency of the signal is:

$$f(t) = \frac{1}{2\pi}\frac{d\phi}{dt}$$

In this program, a chirped signal cos(24t + $\frac{8t^2}{\pi}$) is generated and its frequencies analysed over the time duration $(-\pi, \pi)$. In this interval, the angular frequency of the signal varies continuously ($\omega(t) = 24 + \frac{16t}{\pi}$) from 8 rad/s to 40 rad/s. This variation is best visualised using a time frequency plot where the signal is divided into multiple time bins and the FFT of each slice is plotted versus time on a surface plot.

---

2 Approach

The DFT is computed using the FFT algorithm, implemented using numpy.fft module. To centre the fft vector about the 0-frequency bin, the vector is rearranged using the fftshift function.

2.1 Important blocks used in the program

1: Estimating ω_o and δ

```
def est_wd(f, title):
  y, Y, w, ii = gen_fft(f)
  est_idx = np.argmax(abs(Y[64:]))
  w_est = abs(w[est_idx+64])
  delta_est = np.angle(Y[est_idx+64])
  print("\nOmega estimate = {0} \nActual Omega = {1} \
  (Error = {2})".format(w_est, w0, round(abs(w_est - w0), 4)))
  print("Delta Estimate = {0} \nActual Delta =  {1} \
  (Error = {2})".format(delta_est, delta, round(abs(delta_est - delta), 4)))
  gen_plots(Y, w, ii, [-10, 10], title)
  return 0
```

2: FFT Analysis of Chirped Signal:

```
N = 1024; num_cycles = 1; sample_freq = N/num_cycles
x = num_cycles*np.linspace(-1*pi, pi, N+1)[:-1]
y = chirp(x)
y_2d = np.reshape(y, (16, 64)).T
Y_dft = np.zeros(y_2d.shape, dtype = complex)
N_2d = N/16
w = np.linspace(-0.5, 0.5, int(N_2d+1))*sample_freq; w = w[:-1]
for i in range(16):
  y_2d[:, i] *= ham_win(N_2d)
  Y_dft[:, i] = fftshift(fft(fftshift(y_2d[:, i])))/N_2d
  # gen_surf() is a function to generate 3D surface plots
  fig1 = gen_surf(np.linspace(0, 15, 16), w, abs(Y_dft), \
  "Freq Axis", "Time Bins")
  fig2 = gen_surf(np.linspace(0, 15, 16), np.linspace(0, 63, 64), y_2d, \
  "Time Axis", "Time Bins")
  plt.show()
```

3: Input Format:

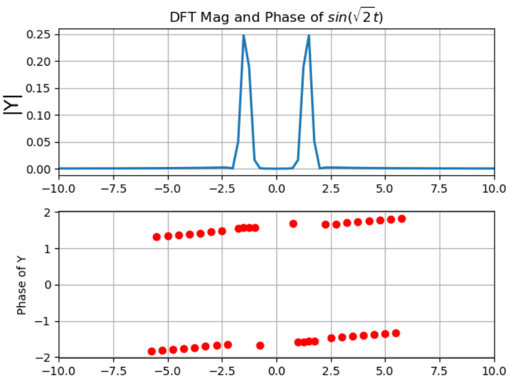
```
python3 EE18B104_Assign9.py --help
usage: EE18B104_Assign9.py [-h] [--fn_choice FN_CHOICE]

optional arguments:
  -h, --help            show this help message and exit
  --fn_choice FN_CHOICE
                        Choose time domain function:
                        1: Sinusoids
                        2: cos^3(0.86t)
                        3: Estimation of frequency and phase of a sinusoid
                        4: Chirped Signal- DFT
                        5: Chirped Signal- Time-Frequency Plot
                        6: Accuracy analysis of w0 and delta estimation
```

---

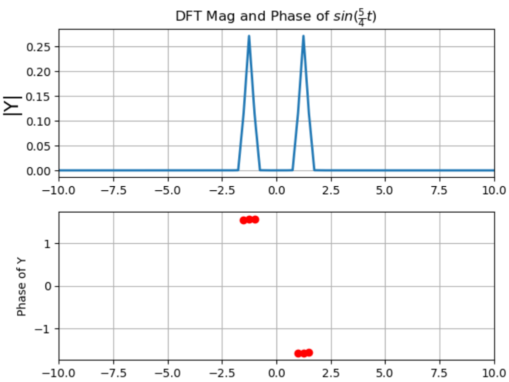
3 FFT Analysis

3.1 Sinusoids

In order to get a purely imaginary FFT, the signal is made anti-symmetric by setting the $\frac{N}{2}^{th}$ element 0.



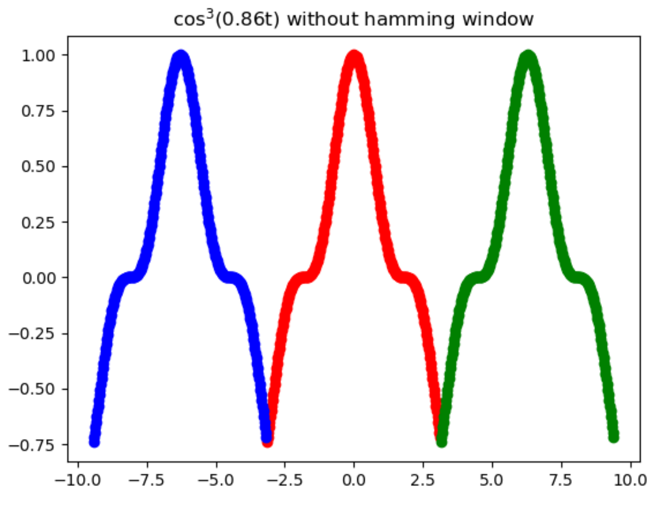
$\sin(\sqrt{2}\mathrm{t})$

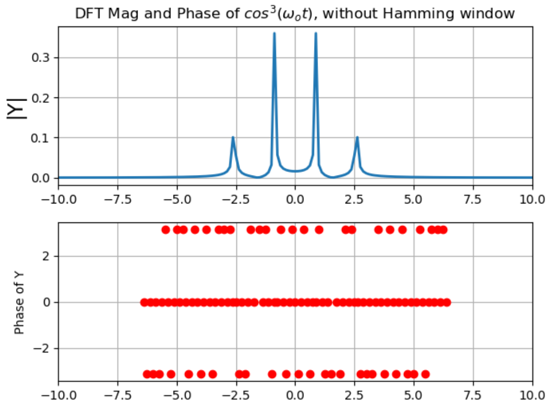


$\sin(1.25\mathrm{t})$

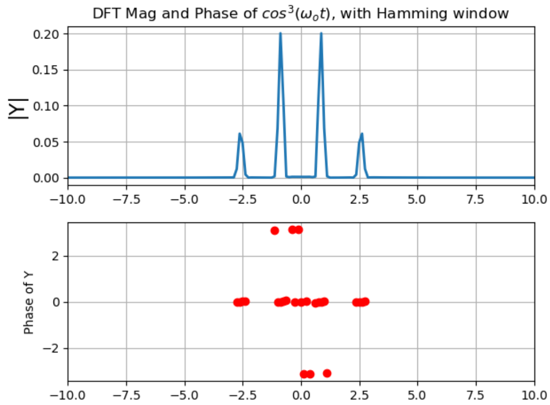
3.2 $\cos^3(0.86\mathrm{t})$

Unlike the sine signals, $\cos^3(0.86\mathrm{t})$ does not have discontinuities at the ends of the interval $(-\pi, \pi)$, Gibbs Phenomenon is not observed even when a Hamming window is not used.





Without Hamming Window



With Hamming Window

3.3 Frequency and Phase Estimation of a Sinusoid

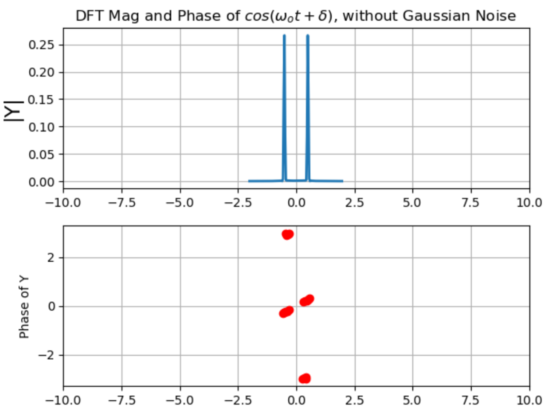
The cosine signal is generated as $\cos(\omega_o t + \delta)$ where

$$\omega_o = \textbf{random.uniform}(0.5, 1.5)$$
$$\delta = \textbf{random.uniform}(-\pi, \pi)$$

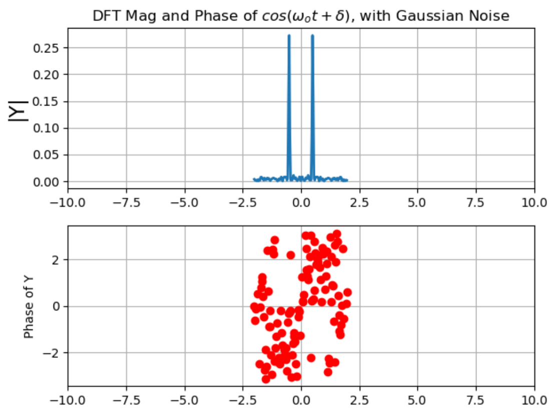
The accuracy of the estimation process was tested by generating $(\omega_o,\ \delta)$ 10000 times and then computing the mean squared error. The results were as follows:

```
Accuracy of w0 and delta estimation:
Max w error = 0.016250000000000098
Max delta error = 6.279223661325401, Number of 2pi occurrences = 19
Delta MSE = 0.07473201870272285
```

Note that $6.279 \approx 2\pi$, and $19\text{x}(6.279^2)/(10000) > 0.0747$, implying that the mse is primarily due to the 2π occurrences.

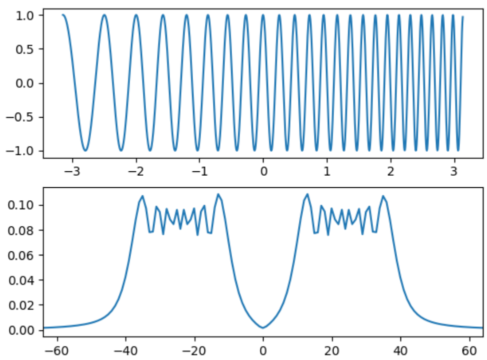


Without Gaussian Noise

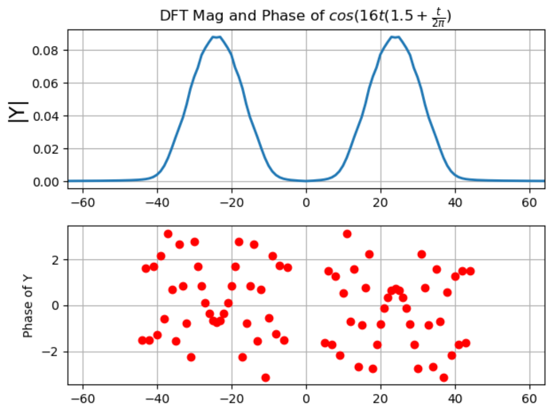


With Gaussian Noise

3.4 Chirped Signal



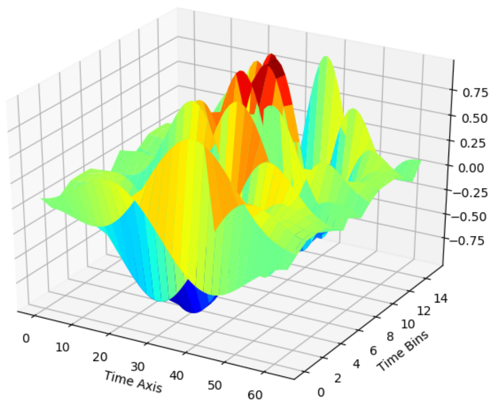
Signal w/o Hamming Window in Time and Frequency Domains



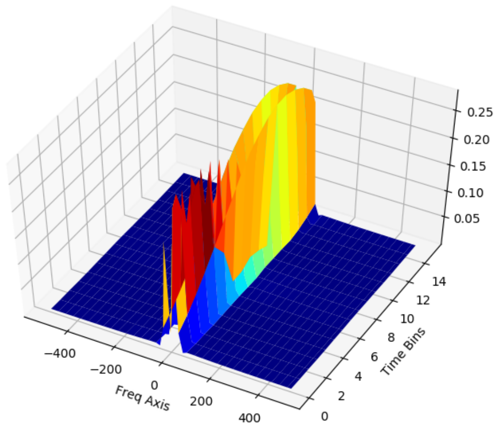
FFT of signal with Hamming window

3.4.1 Time-Frequency Plot

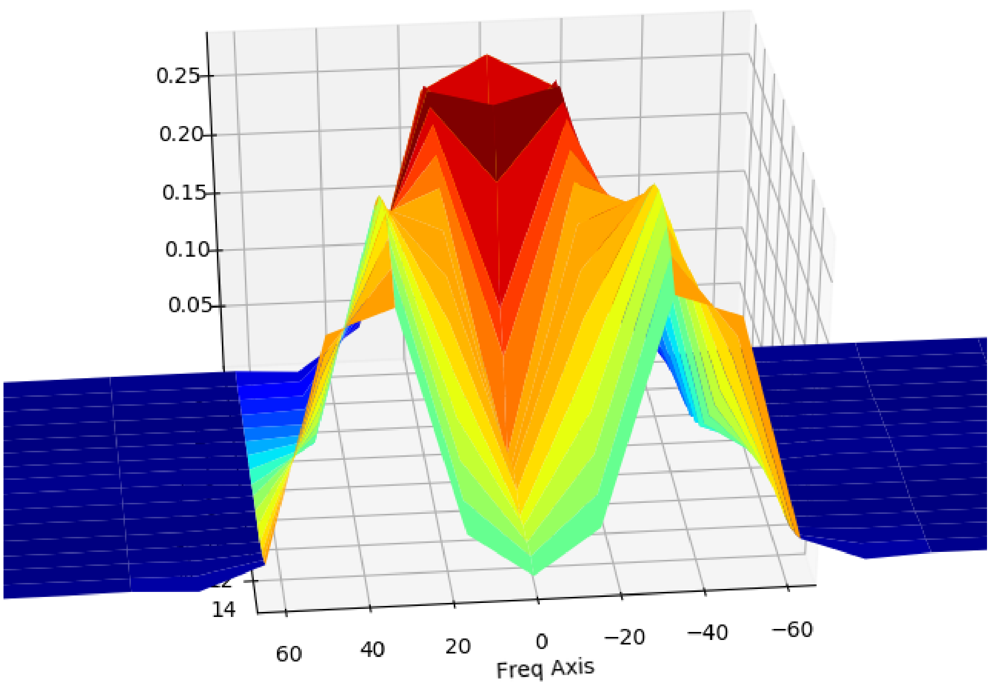
The 1024-long vector is divided into 16 sub-vectors each of length 64. The FFT of each of the sub-vectors are computed and then plotted on a surface plot. Before computing its FFT, the subvector is multiplied by a Hamming window.



Surface Plot of Time-Domain Signal



Surface Plot of FFT



Time-Frequency Plot zoomed into the relevant frequencies- notice that the frequency peaks move from around 8 at time bin 0 to near 40 at time bin 15

⋆⋆⋆