National University of Sciences and Technology (NUST)
School of Electrical Engineering and Computer Science

Department of Electrical Engineering and Computer Science

Faculty Member: Dr. Salman Ghafoor Dated: 24/10/2022

Semester: 5th Section: BEE 12C

EE-232: Signals and Systems

Lab 7: Fourier Series

Group Members

Name	Reg. No	PLO4 – CLO3 **Viva / Quiz / Lab Performance**	PLO5 - CLO3 **Analysis of data in Lab Report**	PLO8 - CLO4 **Modern Tool Usage**	PLO9 - CLO4 **Ethics and Safety**
		5 Marks	**5 Marks**	**5 Marks**	**5 Marks**
Danial Ahmad	331388				
Muhammad Umer	345834				
Syeda Fatima Zahra	334379				

1 Table of Contents

2 Introduction to Properties of Systems

2.1 Objectives

The goal of this laboratory is to be able to calculate the Fourier series of approximately continuous time and discrete time signals and plot the real part of the spectrum / Fourier series coefficients.

- MATLAB Demos on Fourier series
- Fourier Series Calculation of Discrete Time Signals
- Inverse Fourier Series Calculation given Fourier Series Coefficients
- Determine Frequency Response of an LTI Causal System

2.2 Equipment

Software

- *MATLAB*

2.3 Lab Instructions

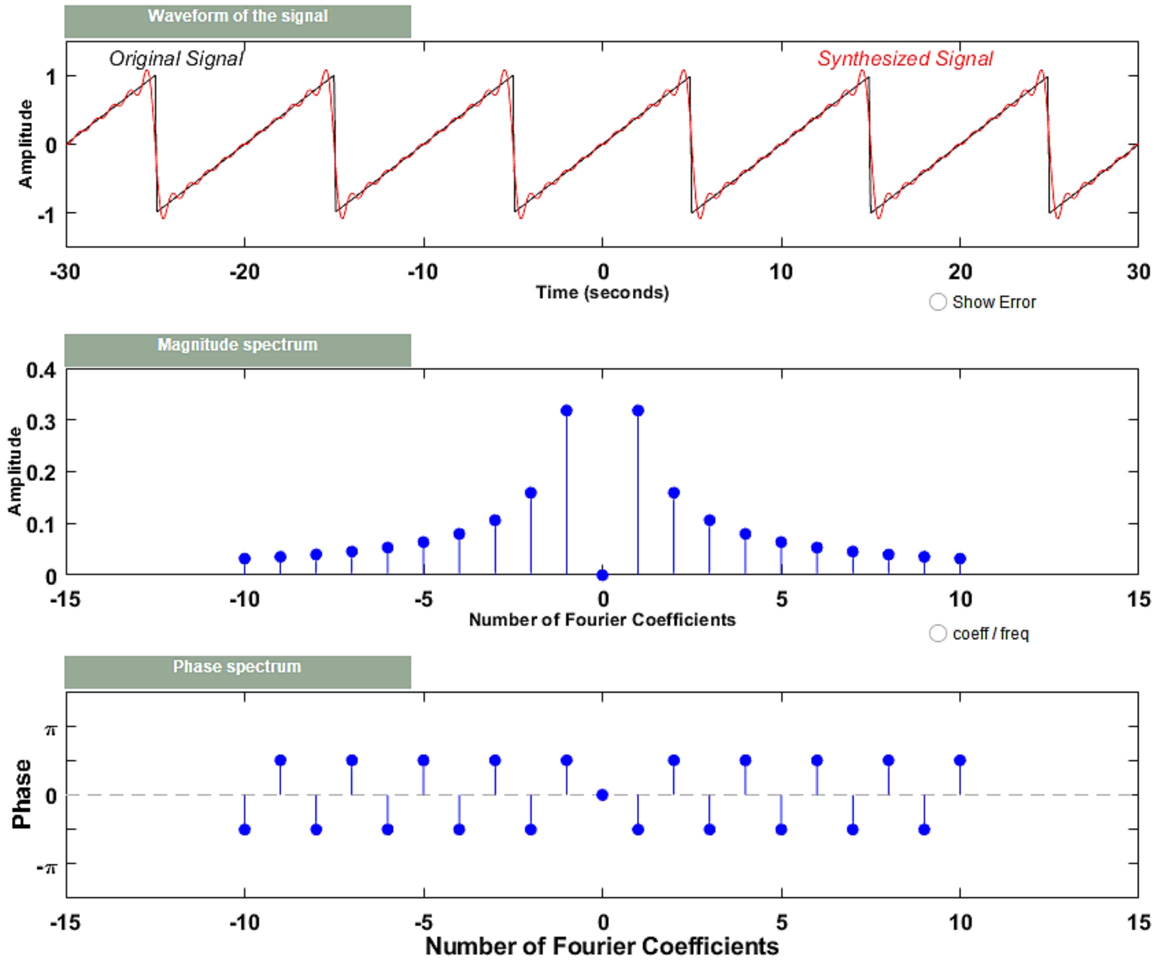
All questions should be answered precisely to get maximum credit. Lab report must ensure following items:

- Lab objectives
- MATLAB codes
- Results (Graphs/Tables) duly commented and discussed
- Conclusion

3 Lab Tasks

3.1 Pre Lab

1. Create a sawtooth waveform with T=10sec and increase the number of Fourier coefficients. Observe and state the significance of increasing number of coefficients.



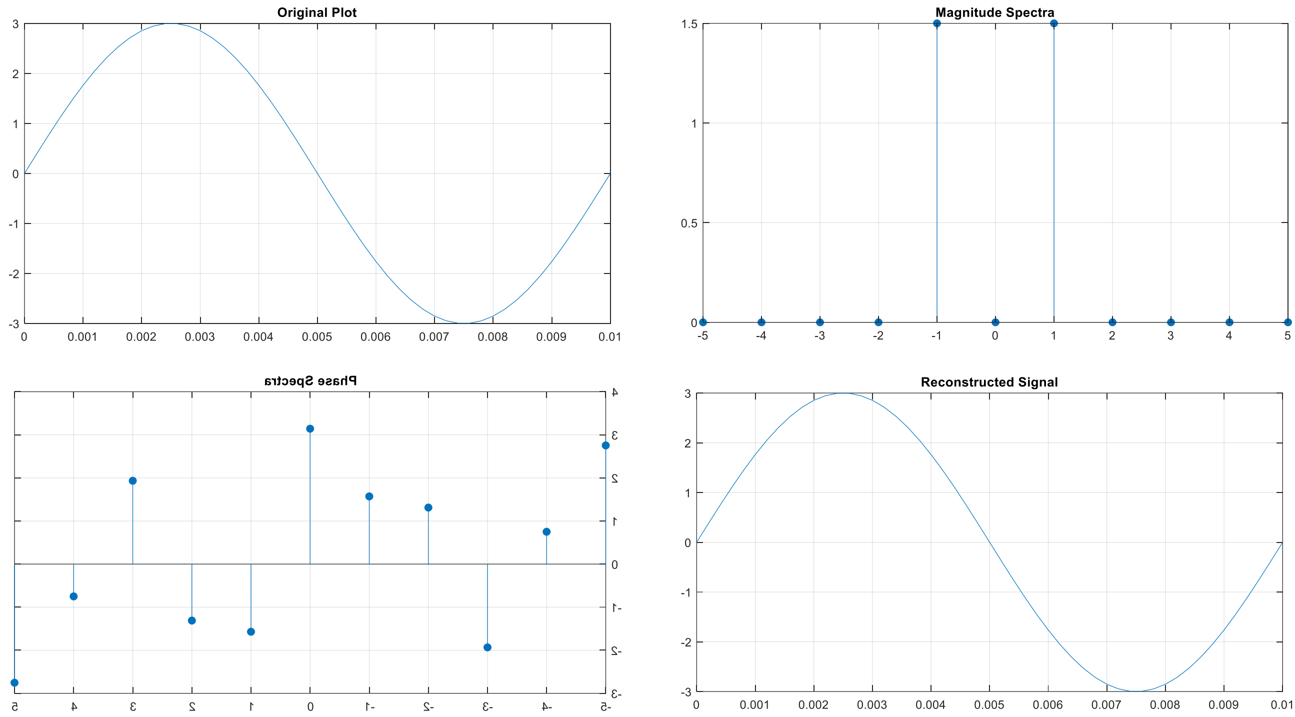
Comments: Upon increasing the number of coefficients of the FS, the sinusoidal waves fit the desires function better however the Gibbs phenomenon (an overshoot of Fourier series and other eigenfunction series occurring at simple discontinuities) starts to become apparent.

3.2 Lab Task 1 & 2 (Reconstruction done in a single code)

3.2.1 Fourier Series of a CT Sinusoid Wave

Write a function that will generate a single sinusoid $x(t) = A\sin(wt)$. $A = 3$ and period $T = 0.01s$. Choose an appropriate value of 'time_increment/sampling time' during the generation of signal. Determine the Fourier series coefficients and plot the magnitude and phase of the Fourier series coefficients.

```matlab
function [] = fourierSine(A, T, e)
    bias = 1 + e;
    n = -e:e;
    steps = T / 50;
    t = 0:steps:T;
    w = 2 * pi / T;
    x = A * sin(w * t);

    subplot(221)
    plot(t, x);
    grid on
    title('Original Plot')

    a = (0);

    for k = -e:e
        a(k + bias) = integral(@(t) (A * sin(w * t) .* exp(-1i * k * w * t)) / T, 0,
T);
    end

    mag = abs(a);
    phase = angle(a);

    subplot(222)
    stem(n, mag, 'filled');
    grid on
    title('Magnitude Spectra')
    subplot(223)
    stem(n, phase, 'filled');
    grid on
    title('Phase Spectra')

    y = (0);

    for k = -e:e
        y = y + a(k + bias) .* exp(1i * k * w * t);
    end

    subplot(224)
    plot(t, real(y));
    grid on
    title('Reconstructed Signal')
end
```

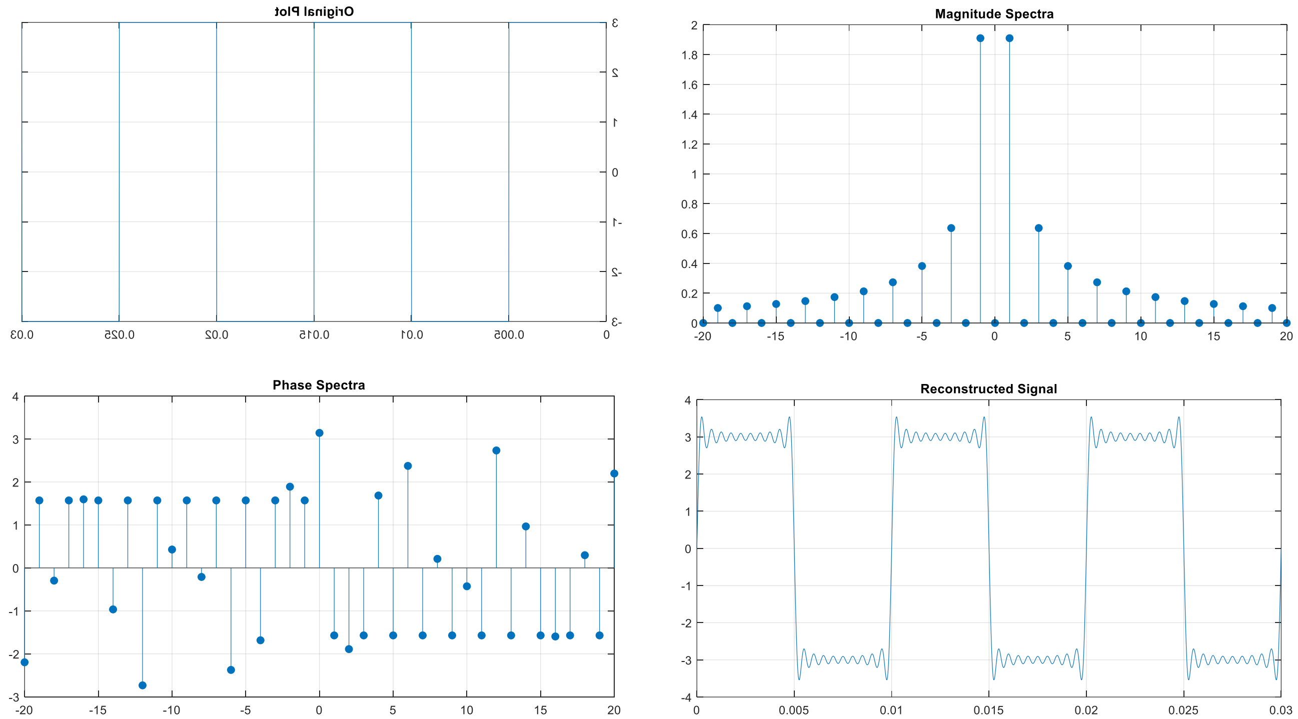


3.2.2 Fourier Series of a CT Rectangular Wave

Assume a rectangular wave as shown below. Using a similar approach outlined in the previous task, obtain the CTFS representation of the rectangular wave. Plot the magnitude and phase of the fourier series coefficients with appropriate axes, labels, and titles.

```
function [] = fourierSquare(A, T, e)
    bias = 1 + e;
    n = -e:e;
    steps = T / 1000;
    t = 0:steps:3 * T;
    w = 2 * pi / T;
    x = A * square(w * t);

    subplot(221)
    plot(t, x);
    grid on
    title('Original Plot')

    a = (0);

    for k = -e:e
        a(k + bias) = integral(@(t) (A * square(w * t) .* exp(-1i * k * w * t)) / T,
0, T);
    end

    mag = abs(a);
```

```
    phase = angle(a);

    subplot(222)
    stem(n, mag, 'filled');
    grid on
    title('Magnitude Spectra')
    subplot(223)
    stem(n, phase, 'filled');
    grid on
    title('Phase Spectra')

    y = (0);

    for k = -e:e
        y = y + a(k + bias) .* exp(1i * k * w * t);
    end

    subplot(224)
    plot(t, real(y));
    grid on
    title('Reconstructed Signal')
end
```



3.3 Lab Task 2

3.3.1 FFT and IFFT

- MATLAB contains efficient routines for computing CTFS and DTFS. If x is an N-point vector for the period $0 \leq n \leq N - 1$, then the DTFS of *x[n]*can be computed by a_k=(l/N)*fft(x), where the N-point vector a contains a_k for $0 \leq k \leq N - 1$. The function fft is simply an efficient implementation scaled by N. Thus, DTFS can by computed by typing a_k=(l/N)*fft(x). The function will return both real and imaginary parts of the DTFS coefficients.
- Given a vector containing the DTFS coefficients $\boldsymbol{a_k}$ for $0 \leq k \leq N - 1$, the function **ifft** can be used to construct a vector x containing *x[n]* for $0 \leq n \leq N - 1$ as x=N*ifft (a). The function **ifft** is an efficient realization of the DTFS synthesis equation, scaled by 1/N.

Choose an appropriate value of 'time_increment' during the generation of cosine function.

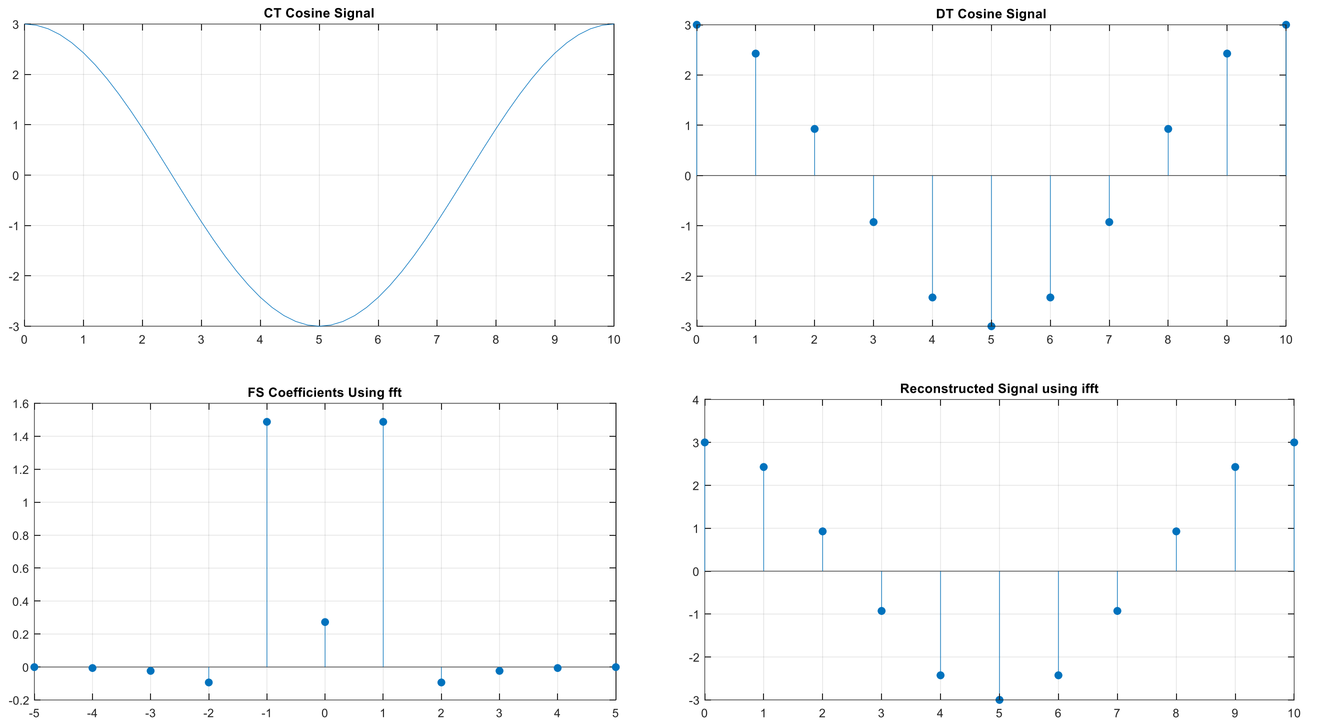
```
for n=0:time_increment:T
%Generate Cosine Wave
L=length(signal);
y=real(fft(signal,L))/L;
stem(y)
```

In case of the given signal t and T are being used in place of n and N because the increment between 0 and T will have small increments than 1.

Using the function 'ifft' and knowledge of FS coefficents of a Cosine waveform determine the signal x[n]. For the lab report plot both the fourier coefficients and time domain signal.

```
T = 10; A = 3;
steps = T / 50;
t = 0:steps:T;
x = A * cos(2 * pi * (1 / T) * t);

subplot(221);
plot(t, x);
grid on
title('CT Cosine Signal');

n = 0:T;
subplot(222);
y = A * cos(2 * pi * (1 / T) * n);
stem(n, y, 'filled');
grid on
title('DT Cosine Signal');

L = length(y);
fourierSeries = (1 / L) .* (fft(y, L));
```

```
reconstructedSignal = ifft(fourierSeries, L);

k_x = (-T / 2):(T / 2);
t_x = 0:T;

subplot(223);
stem(k_x, real(fftshift(fourierSeries)), 'filled');
grid on
title('FS Coefficients Using fft');

subplot(224);
stem(t_x, real(L .* reconstructedSignal), 'filled');
grid on
title('Reconstructed Signal using ifft');
```



4 Conclusion

After performing this lab, we conclude that fseriesdemo is an important MATLAB GUI for realizing Fourier series of different signals such as sinusoids, ramp etc., and how increasing the number of Fourier coefficients result in the reconstructed signal approaching the original time domain signal. Lastly, fft and ifft were used for DT Fourier Series Calculation which are an efficient implementation of analysis and synthesis equations respectively.