Signals Systems and Communication Lab

Laboratory report submitted for the partial fulfillment
of the requirements for the degree of

Bachelor of Technology
in
Electronics and Communication Engineering

by

Mohit Akhouri - 19ucc023

Course Coordinator
Dr. Navneet Upadhyay

LNMIIT
The LNM Institute of Information Technology

Department of Electronics and Communication Engineering
The LNM Institute of Information Technology, Jaipur

February 2021

Contents

;

Chapter *4*

Experiment - 4

4.1 Aim of the experiment

1. To generate two periodic signals $x_1(t)$ and $x_2(t)$
2. To compute and plot the Fourier spectra for the aforementioned periodic signals
3. To illustrate the Gibb's phenomenon

4.2 Software Used

MATLAB

4.3 Theory

4.3.1 About Fourier Series :

Jean Baptiste Joseph Fourier,a French mathematician and a physicist, was born in Auxerre, France. He initialized Fourier series, Fourier transforms and their applications to problems of heat transfer and vibrations. The Fourier series, Fourier transforms and Fourier's Law are named in his honour.
To represent any periodic signal x(t), Fourier developed an expression called Fourier series. This is in terms of an infinite sum of sines and cosines or exponentials. Fourier series uses orthogonality condition. The computation and study of Fourier series is known as harmonic analysis and is extremely useful as a way to break up an arbitrary periodic function into a set of simple terms that can be plugged in, solved individually, and then recombined to obtain the solution to the original problem or an approximation to it to whatever accuracy is desired or practical.In particular, since the superposition principle holds for solutions of a linear homogeneous ordinary differential equation, if such an equation can be solved in the case of a single sinusoid, the solution for an arbitrary function is immediately available by expressing the original function as a Fourier series and then plugging in the solution for each sinusoidal component.

The Fourier Series of a periodic signal x(t) with period T is given by :

$$x(t) = \sum_{k=-\infty}^{+\infty} D_k e^{jk\omega_o t} \tag{4.1}$$

where $\omega_o = \frac{2\pi}{T}$ and D_k is the k^{th} fourier series coeffiecient.
The Fourier Series coefficient D_k is calculated by :

$$D_k = \frac{1}{T} \int_T x(t) e^{-jk\omega_o t} dt \tag{4.2}$$

In order to compute D_k discretely, approximating the aforementioned finite integral as :

$$D_k = \frac{1}{N} \sum_{n=0}^{N-1} x(nT_s) e^{-jk\Omega_o n} \tag{4.3}$$

where $\Omega_0 = \omega_o T_s$ where T_s is the sampling interval and $N = \frac{T}{T_s}$ is number of samples in one period T.

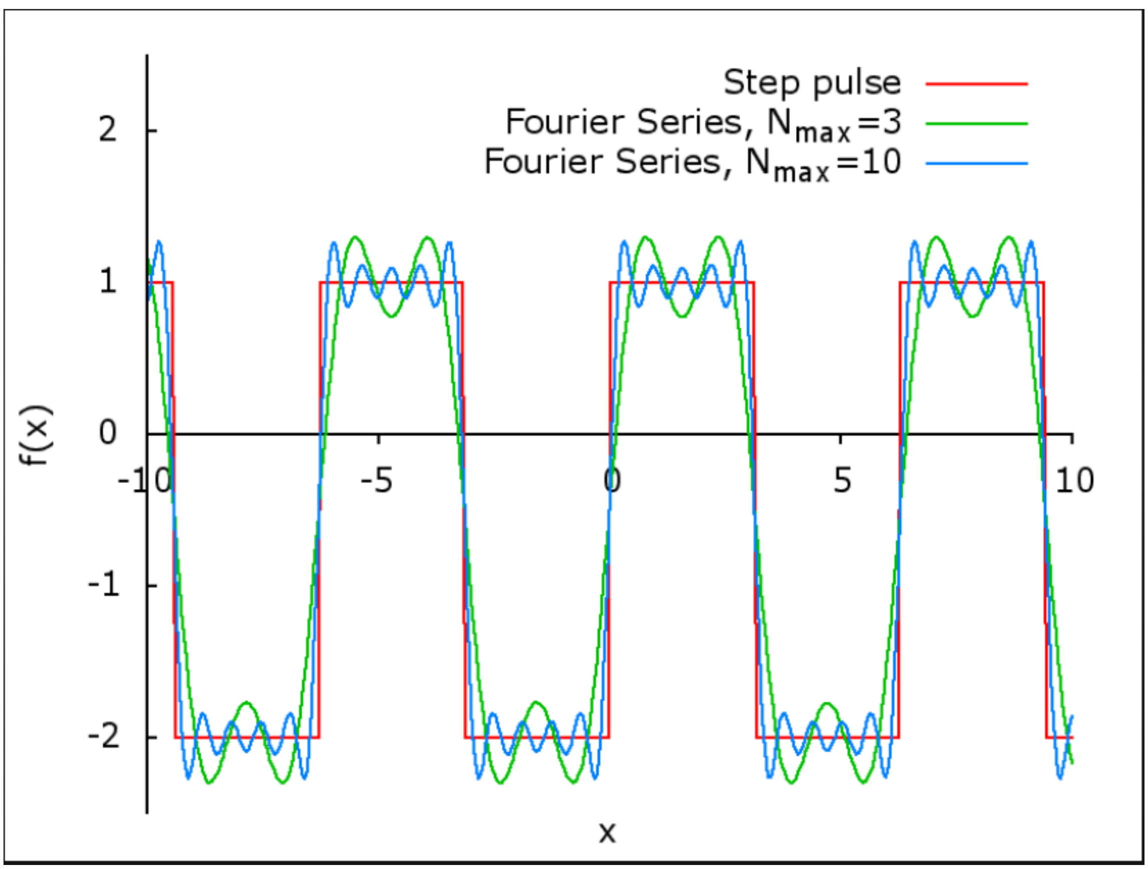


Figure 4.1 Fourier Series of a step pulse

4.3.2 About Gibb's Phenomenon :

In mathematics, the Gibbs phenomenon, discovered by **Henry Wilbraham** (1848) and rediscovered by **J. Willard Gibbs** (1899), is the peculiar manner in which the Fourier series of a piecewise continuously differentiable periodic function behaves at a jump discontinuity. The nth partial sum of the Fourier series has large oscillations near the jump, which might increase the maximum of the partial sum above that of the function itself. The overshoot does not die out as n increases, but approaches a finite limit. This sort of behavior was also observed by experimental physicists, but was believed to be due to imperfections in the measuring apparatus. This is one cause of ringing artifacts in signal processing. The Gibbs phenomenon involves both the fact that Fourier sums overshoot at a jump discontinuity, and that this overshoot does not die out as more terms are added to the sum.
The equation of the approximation of the original periodic signal x(t) is defined as :

$$x_M(t) = \sum_{k=-M}^{M} D_k e^{jk\omega_o t} \tag{4.4}$$

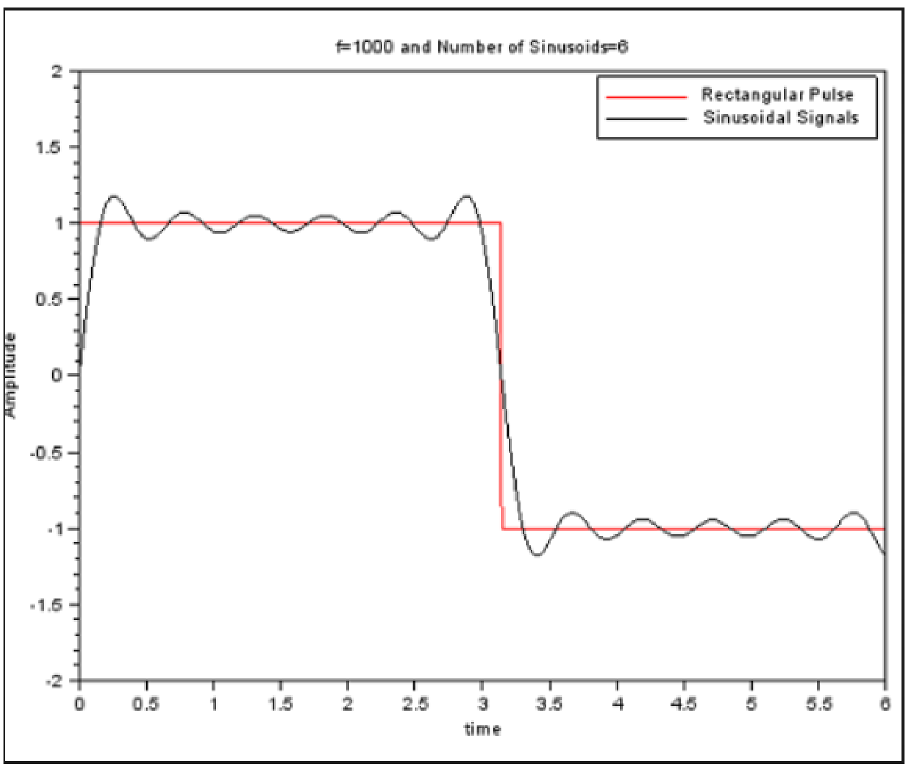


Figure 4.2 Illustration of Gibb's phenomenon

4.4 Code and Results

4.4.1 Exercise 1 : Plotting periodic signals $x_1(t)$ and $x_2(t)$ and plotting Fourier spectra :

```
function [Dk] = Fourier_Series_Coeff(x,N,num)
% This function will calculate the first 'num' coefficients of
% periodic signals, num = 1,2,3 ... 10...

% Name = Mohit Akhouri
% Roll no = 19UCC023
% SSC LAB Batch D1 - Monday ( 2-5 pm )

Dk = zeros(1,num); % initializing output variable Dk for storing
 Fourier series coefficients

% running loop to find first 'num' fourier series coefficients
for k=1:num
    sum=0;
    for n=1:N
        sum=sum+(x(n).*exp(-1j*(k-1)*(n-1)*2*pi/N));
    end
    sum=sum/N;
    Dk(k)=sum;
end

end
```

Published with MATLAB® R2020b

Figure 4.3 Code for the function Fourier_Series_Coeff to calculate the 10 fourier series coefficients D_k

```
function [xt] = Fourier_Spectra(Dk,T,t,num)
% This function will calculate and return the fourier spectra of
 signal
% with time period T for number of fourier series coefficients
% equal to 'num'

% Name = Mohit Akhouri
% Roll no = 19UCC023
% SSC LAB Batch D1 - Monday ( 2-5 pm )

xt = 0; % initializing output variable xt to store fourier spectra

% loop for finding fourier spectra of signal with 'num' Fourier
 coefficients
for k=1:num
    xt = xt + (Dk(k) .* exp(1j*(k-1)*t*2*pi/T));
end

end
```

Published with MATLAB® R2020b

Figure 4.4 Code for function Fourier_Spectra to plot fourier spectra of signals $x_1(t)$ and $x_2(t)$

```
% Name = Mohit Akhouri
% Roll no = 19UCC023
% SSC LAB Batch D1 - Monday ( 2-5 pm )

% plotting periodic signals x1(t) and x2(t)
N = 256; % defining total number of samples
T = 2; % defining bound on time
t = linspace(0,T,N); % defining the time(t) axis
num = 10; % number of fourier series coefficients to be calculated

% loop for defining periodic signals x1(t) and x2(t)
for i=1:length(t)
    if(t(i)<=1)
        x1(i)=exp(-t(i)/2);
        x2(i)=1;
    elseif(t(i)>1 & t(i)<=2)
        x1(i)=0;
        x2(i)=-1;
    end
end
% plotting signals x1(t) and x2(t)
figure;
subplot(2,1,1);
plot(t,x1,'Linewidth',1.5);
xlabel('time(t) ->');
ylabel('x_{1}(t) ->');
title('Plotting x_{1}(t) = e^{-t/2}');
grid on;
subplot(2,1,2);
plot(t,x2,'Linewidth',1.5);
xlabel('time(t) ->');
ylabel('x_{2}(t) ->');
title('Plotting x_{2}(t) = 1 ( 0 <= t <= T/2 ) , -1 ( T/2 < t <= T) ,
 T=2');
grid on;
sgtitle('19ucc023 - Mohit Akhouri');

Dk1=Fourier_Series_Coeff(x1,N,num); % calculating 10 fourier series
 coefficients for signal x1(t)
Dk2=Fourier_Series_Coeff(x2,N,num); % calculating 10 fourier series
 coefficients for signal x1(t)
mag_Dk1=abs(Dk1); % calculating magnitude of Dk1
ang_Dk1=angle(Dk1); % calculating phase angle of Dk1
mag_Dk2=abs(Dk2); % calculating magnitude of Dk2
ang_Dk2=angle(Dk2); % calculating phase angle of Dk2

% plotting magnitude and phase plot of Dk1
figure;
subplot(2,1,1);
stem(mag_Dk1,'Linewidth',1.5);
xlabel('Coefficient (k) ->');
ylabel('Magnitude of D_{k} ->');
```

1

Figure 4.5 Part 1 of the code for plotting periodic signals, calling functions to calculate D_k and fourier spectra of signals $x_1(t)$ and $x_2(t)$

```
title('Magnitude plot of D_{k} of signal x_{1}(t)');
grid on;
subplot(2,1,2);
stem(ang_Dk1,'Linewidth',1.5);
xlabel('Coefficient (k) ->');
ylabel('Phase angle of D_{k} ->');
title('Phase plot of D_{k} of signal x_{1}(t)');
grid on;
sgtitle('19ucc023 - Mohit Akhouri');

% plotting magnitude and phase plot of Dk2
figure;
subplot(2,1,1);
stem(mag_Dk2,'Linewidth',1.5);
xlabel('Coefficient (k) ->');
ylabel('Magnitude of D_{k} ->');
title('Magnitude plot of D_{k} of signal x_{2}(t)');
grid on;
subplot(2,1,2);
stem(ang_Dk2,'Linewidth',1.5);
xlabel('Coefficient (k) ->');
ylabel('Phase angle of D_{k} ->');
title('Phase plot of D_{k} of signal x_{2}(t)');
grid on;
sgtitle('19ucc023 - Mohit Akhouri');

figure;
xt1=Fourier_Spectra(Dk1,T,t,num); % Fourier spectra of signal x1(t)
xt2=Fourier_Spectra(Dk2,T,t,num); % Fourier spectra of signal x2(t)

% plotting fourier spectra of signals x1(t) and x2(t) for Dk, k=0-9
subplot(2,1,1);
plot(t,xt1,'Linewidth',1.5);
xlabel('time(t) ->');
ylabel('x_{1}(t) ->');
title('Fourier spectra of x_{1}(t)');
grid on;
subplot(2,1,2);
plot(t,xt2,'Linewidth',1.5);
xlabel('time(t) ->');
ylabel('x_{2}(t) ->');
title('Fourier spectra of x_{2}(t)');
grid on;
sgtitle('19ucc023 - Mohit Akhouri');

Warning: Imaginary parts of complex X and/or Y arguments ignored.
Warning: Imaginary parts of complex X and/or Y arguments ignored.
```

Figure 4.6 Part 2 of the code for plotting periodic signals, calling functions to calculate D_k and fourier spectra of signals $x_1(t)$ and $x_2(t)$

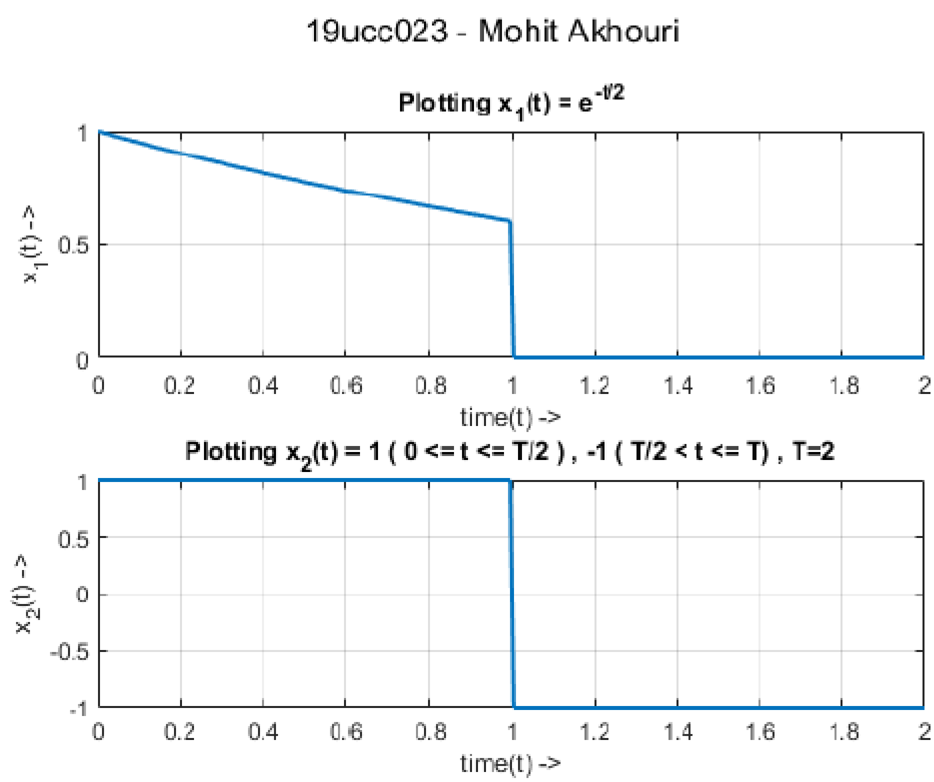


Figure 4.7 Plot of the periodic signals $x_1(t)$ and $x_2(t)$

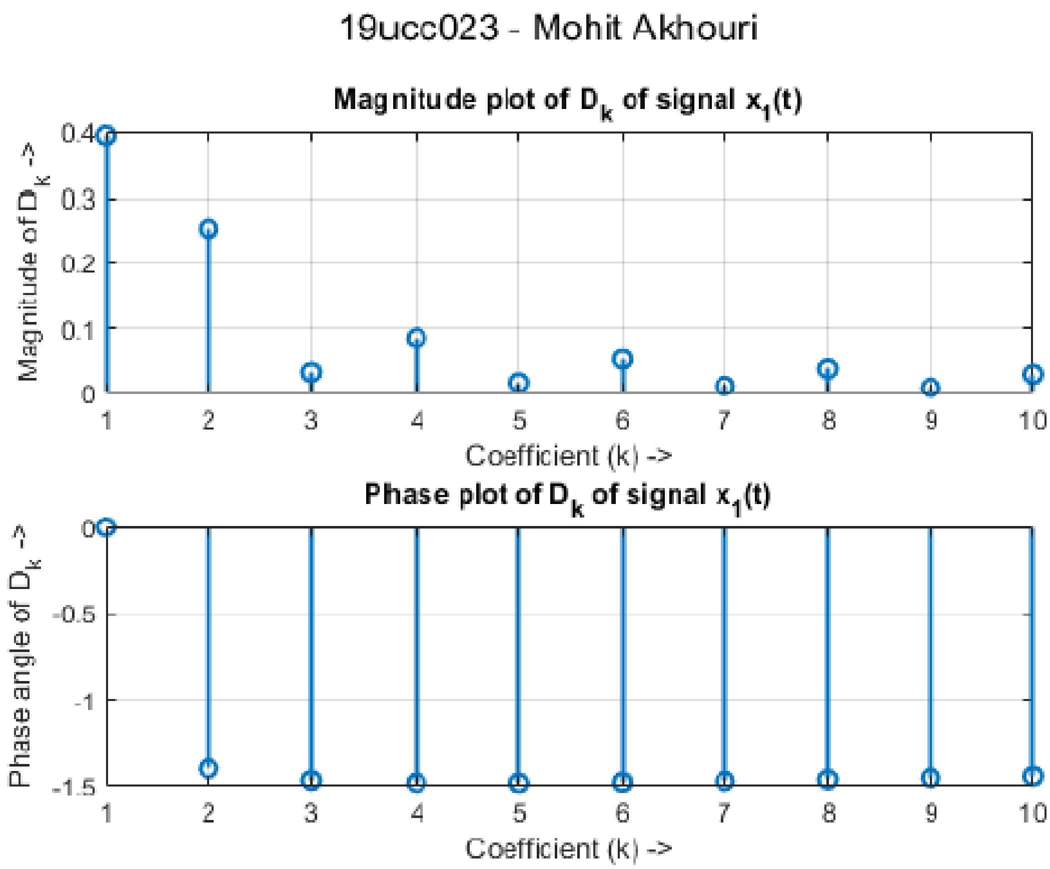


Figure 4.8 Plot of the magnitude and phase spectra of D_k for signal $x_1(t)$

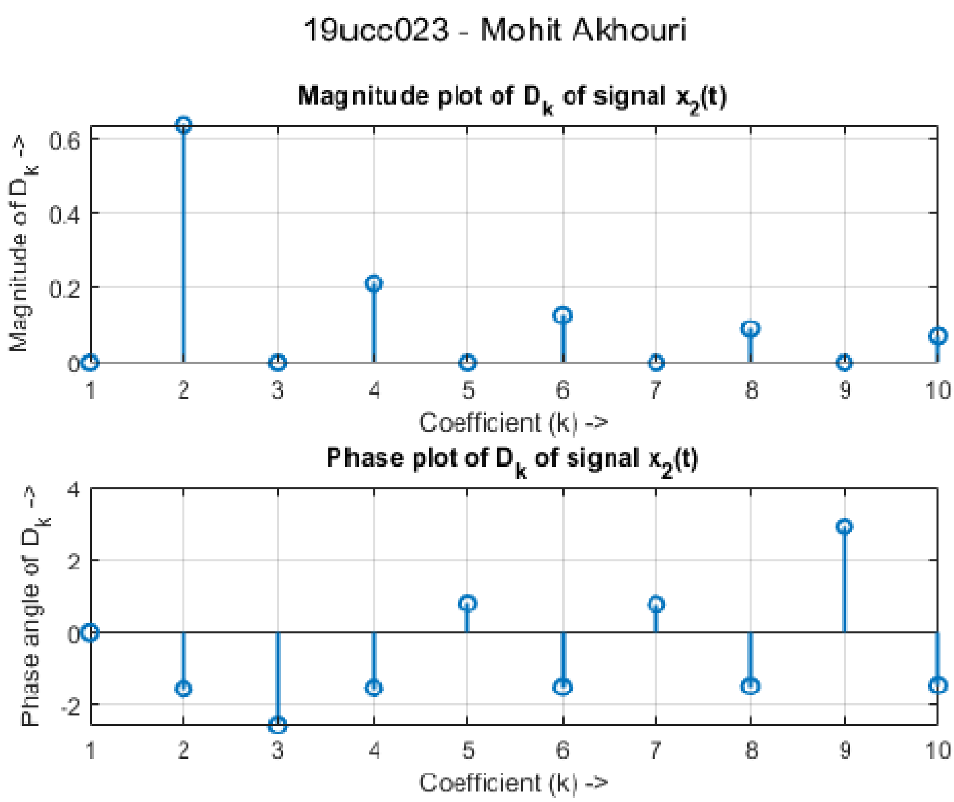


Figure 4.9 Plot of the magnitude and phase spectra of D_k for signal $x_2(t)$

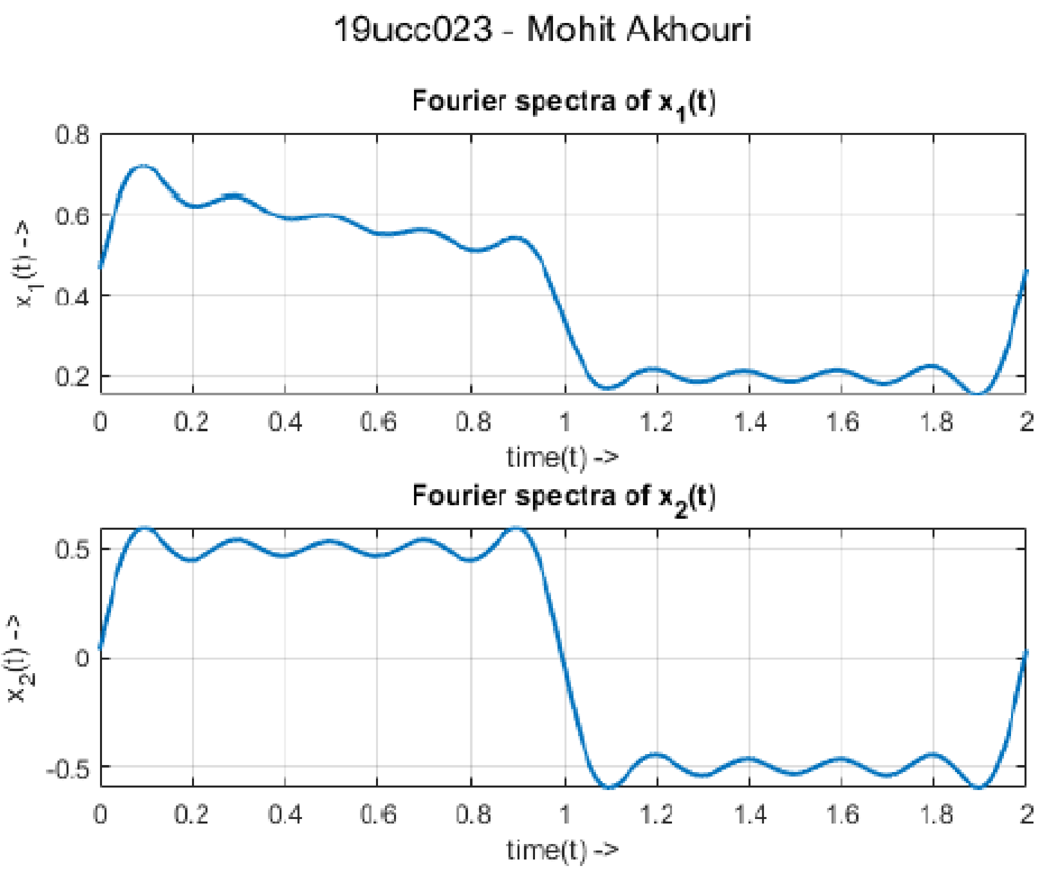


Figure 4.10 Plot of the Fourier spectra for signals $x_1(t)$ and $x_2(t)$ with 10 fourier series coefficients

4.4.2 Exercise 2: Illustration of Gibb's phenomenon for given signals for M=19 and M=99:

```
function [Dk] = Fourier_Series_Coeff_Gibbs(x,N,M)
% This function will calculate the Fourier Series Coefficient
% required for the Gibb's phenomenon that is Dk , where k = -M to M

% Name = Mohit Akhouri
% Roll no = 19UCC023
% SSC LAB Batch D1 - Monday ( 2-5 pm )

size = 2*M+1; % defining size of Dk variable
Dk = zeros(1,size); % initializing Dk variable to store coefficients

% running loop from k=-M to M to find fourier series coefficients
for k=-M:M
    sum=0;
    for n=1:N
        sum=sum+(x(n).*exp(-1j*k*(n-1)*2*pi/N));
    end
    sum=sum/N;
    Dk(k+M+1)=sum;
end

end
```

Published with MATLAB® R2020b

Figure 4.11 Code for the function to calculate Fourier Series Coefficient (D_k) used in function Gibbs_Phenomenon function

```
function [xm] = Gibbs_Phenomenon(x,t,T,N,M)
% This function will illustrate the gibb's phenomenon for signal x(t)
% for a value of M ( in this exp, M=19 and M=99 )

% This Gibb's phenomenon function returns the approximated signal
 xm(t)
% for original signal x(t) where M is a number which is given as 19
 and 99
% in this experiment

% Name = Mohit Akhouri
% Roll no = 19UCC023
% SSC LAB Batch D1 - Monday ( 2-5 pm )

Dk=Fourier_Series_Coeff_Gibbs(x,N,M); % calculating Dk used for
 calulating xm(t)
xm=0; % initializing output variable xm to store approximated signal
 xm(t)

% running loop to calculate approximated signal xm(t)
for k=-M:M
    xm=xm+(Dk(k+M+1).*exp(1j*k*t*2*pi/T));
end
end
```

Published with MATLAB® R2020b

Figure 4.12 Code for the function (Gibbs_Phenomenon) to illustrate the Gibb's phenomenon for M=19 and M=99

```
% Name = Mohit Akhouri
% Roll no = 19UCC023
% SSC LAB Batch D1 - Monday ( 2-5 pm )

% This script will illustrate the Gibb's phenonmenon on the signals
% x1(t) and x2(t)

N = 256; % defining constant N ( total samples )
T = 2; % defining bound on time axis
t = linspace(0,T,N); % defining time axis ( time from 0 to T )

% loop for defining periodic signals x1(t) and x2(t)
for i=1:length(t)
    if(t(i)<=1)
        x1(i)=exp(-t(i)/2);
        x2(i)=1;
    elseif(t(i)>1 & t(i)<=2)
        x1(i)=0;
        x2(i)=-1;
    end
end

xm1_19 = Gibbs_Phenomenon(x1,t,T,N,19); % Gibb's phenomenon for signal
 x1(t) when M=19
xm1_99 = Gibbs_Phenomenon(x1,t,T,N,99); % Gibb's phenomenon for signal
 x1(t) when M=99

% plotting approximated signal xm(t) for periodic signal x1(t)
figure;
subplot(2,1,1);
plot(t,xm1_19,'Linewidth',1.5');
xlabel('time(t) ->');
ylabel('x_{M}(t) ->');
title('approximation x_{M}(t) for periodic signal x_{1}(t) for M=19');
grid on;
subplot(2,1,2);
plot(t,xm1_99,'Linewidth',1.5);
xlabel('time(t) ->');
ylabel('x_{M}(t) ->');
title('approximation x_{M}(t) for periodic signal x_{1}(t) for M=99');
grid on;
sgtitle('19ucc023 - Mohit Akhouri');

xm2_19 = Gibbs_Phenomenon(x2,t,T,N,19); % Gibb's phenomenon for signal
 x2(t) when M=19
xm2_99 = Gibbs_Phenomenon(x2,t,T,N,99); % Gibb's phenomenon for signal
 x2(t) when M=99

% plotting approximated signal xm(t) for periodic signal x2(t)
figure;
subplot(2,1,1);
plot(t,xm2_19,'Linewidth',1.5');
```

1

Figure 4.13 Part 1 of the code for illustration of Gibb's phenomenon on periodic signals $x_1(t)$ and $x_2(t)$

```
xlabel('time(t) ->');
ylabel('x_{M}(t) ->');
title('approximation x_{M}(t) for periodic signal x_{2}(t) for M=19');
grid on;
subplot(2,1,2);
plot(t,xm2_99,'Linewidth',1.5);
xlabel('time(t) ->');
ylabel('x_{M}(t) ->');
title('approximation x_{M}(t) for periodic signal x_{2}(t) for M=99');
grid on;
sgtitle('19ucc023 - Mohit Akhouri');

Warning: Imaginary parts of complex X and/or Y arguments ignored.
Warning: Imaginary parts of complex X and/or Y arguments ignored.
Warning: Imaginary parts of complex X and/or Y arguments ignored.
Warning: Imaginary parts of complex X and/or Y arguments ignored.
```

Figure 4.14 Part 2 of the code for illustration of Gibb's phenomenon on periodic signals $x_1(t)$ and $x_2(t)$

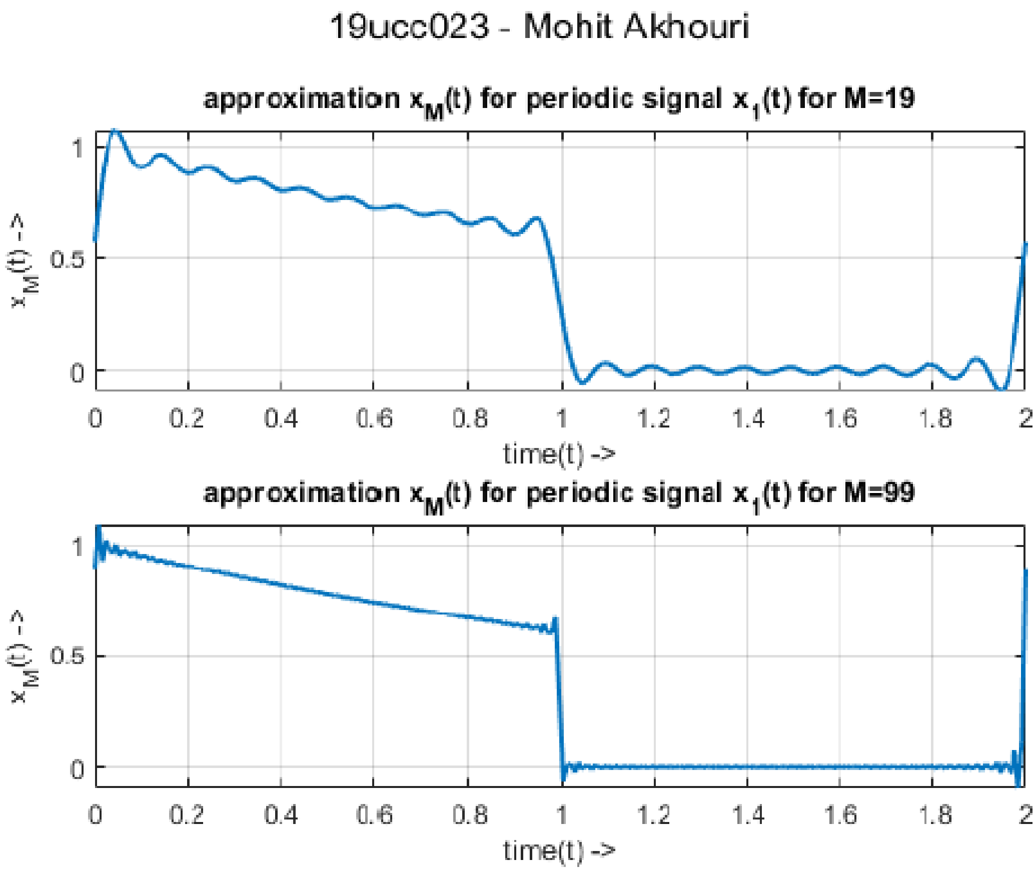
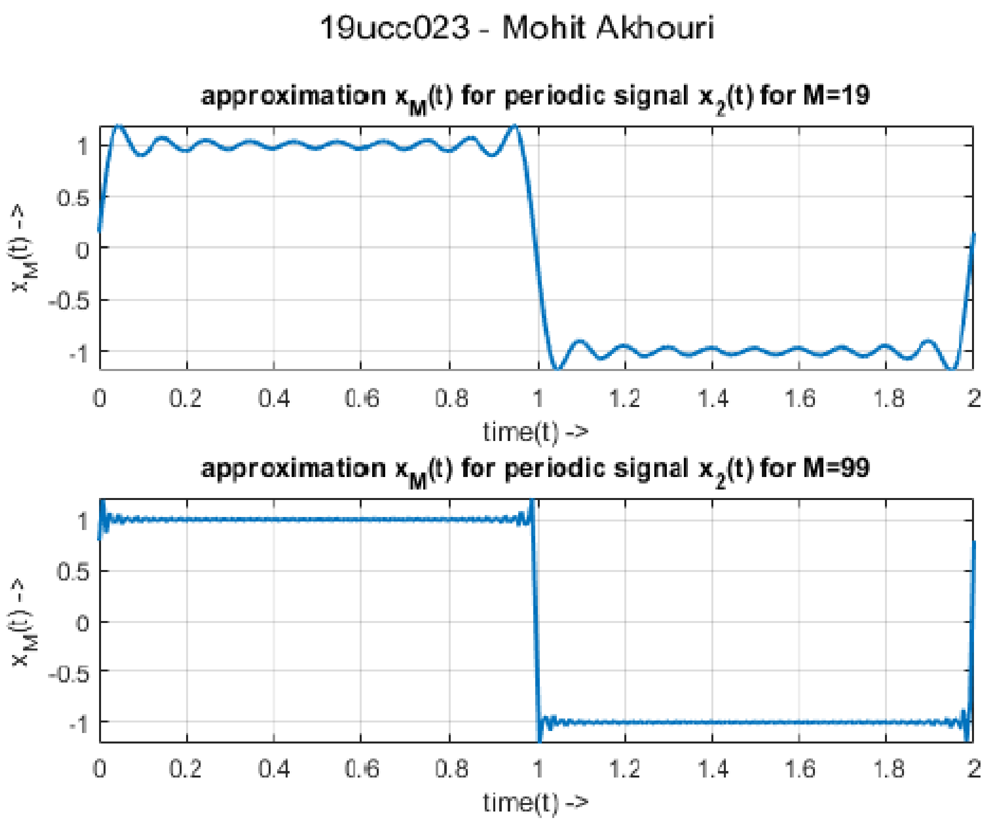


Figure 4.15 Graph of illustration of Gibb's phenomenon on signal $x_1(t)$ for M=19 and M=99



Published with MATLAB® R2020b

Figure 4.16 Graph of illustration of Gibb's phenomenon on signal $x_2(t)$ for M=19 and M=99

4.5 Conclusion

In this experiment, we learnt about the concept of Fourier Series and the representation of Fourier series of periodic signals. We also implemented the concept of plotting periodic signals , exponential and square wave signals. We learnt the concept of Fourier series coefficients D_k and implemented an algorithm to compute them. We also implemented algorithm to compute the Fourier Spectra of periodic signals.

We also learnt about the **Gibb's phenomenon** and how it can be used to obtain the **approximated version** of the original periodic signal. We implemented an algorithm to illustrate the Gibb's phenomenon for the two periodic signals (exponential and square wave) for two values of M , that is for M=19 and M=99.