RYERSON UNIVERSITY

Faculty of Engineering, Architecture and Science

Department of Electrical and Computer Engineering

Course Number	**ELE532**
Course Title	**Signals and Systems I**
Semester/Year	Fall 2020

Instructor	**Javad Alirezaie**

ASSIGNMENT No.	**3**

Assignment Title	*Fourier Series Analysis Using MATLAB*

Submission Date	**November 15th, 2020**
Due Date	**November 15th, 2020**

Student Name	**Nathan Agnew**
Student ID	500819685
Signature*	Nathan Agnew
Student Name	**Ahmad Fahmy**
Student ID	500913092
Signature*	**AF**

**By signing above you attest that you have contributed to this written lab report and confirm that all work you have contributed to this lab report is your own work. Any suspicion of copying or plagiarism in this work will result in an investigation of Academic Misconduct and may result in a "0" on the work, an "F" in the course, or possibly more severe penalties, as well as a Disciplinary Notice on your academic record under the Student Code of Academic Conduct, which can be found online at: www.ryerson.ca/senate/current/pol60.pdf.*

Table of Contents

A. Fourier Series in MATLAB

Problem A.1 [1 Mark] Given the periodic signal $x_1(t)$:

$$x_1(t) = \cos\frac{3\pi}{10}t + \frac{1}{2}\cos\frac{\pi}{10}t, \tag{3}$$

derive an expression for the Fourier coefficients D_n.

$$x_1(t) = cos\frac{3\pi}{10}t + \frac{1}{2}cos\frac{\pi}{10}t$$

Signal is given to be periodic

$$\omega_{0\,1} = \frac{3\pi}{10}, \omega_{0\,2} = \frac{\pi}{10}$$

$$\frac{3\pi}{10} = \frac{3\pi}{10}/2\pi = \frac{3}{20} \qquad T_{02} = 1/\frac{3}{20} = \frac{20}{3}$$

$$\frac{\pi}{10} = \frac{\pi}{10}/2\pi = \frac{1}{20} \qquad T_{01} = 1/\frac{1}{20} = 20$$

$$T_0 = smallest\ multiple\ = 20 \quad \omega_0 = \frac{\pi}{10}$$

$$x_1(t) = \frac{e^{j(\frac{3\pi}{10}t)} - e^{-j(\frac{3\pi}{10}t)}}{2} + \frac{1}{2}\frac{e^{j(\frac{\pi}{10}t)} + e^{-j(\frac{\pi}{10}t)}}{2}$$

$$x_1(t) = \frac{e^{j(\frac{3\pi}{10}t)} - e^{-j(\frac{3\pi}{10}t)}}{2} + \frac{e^{j(\frac{\pi}{10}t)} + e^{-j(\frac{\pi}{10}t)}}{4}$$

$$x_1(t) = \frac{e^{j(3\omega_0 t)} - e^{-j(3\omega_0 t)}}{2} + \frac{e^{j(1\omega_0 t)} + e^{-j(1\omega_0 t)}}{4}$$

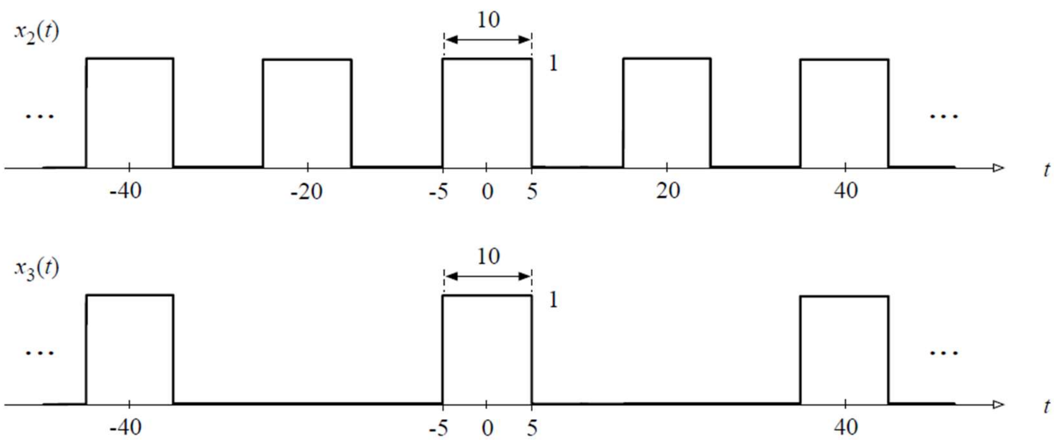


Figure 1: Periodic functions $x_2(t)$ and $x_3(t)$.

Problem A.2 [1 Mark] Repeat Problem A.1 for the periodic signals $x_2(t)$ and $x_3(t)$ shown in Figure 1.

X2:

$$\omega_0 = \frac{2\pi}{20} = \frac{\pi}{10}$$

$$D_n = 2\frac{T_1}{T_0} sinc(\omega_0 T_1 n)$$

$$D_{nx_2} = 2\frac{5}{20} sinc(\frac{\pi}{10} 5n)$$

X3:

$$\omega_0 = \frac{2\pi}{40} = \frac{\pi}{20}$$

$$D_n = 2\frac{T_1}{T_0} sinc(\omega_0 T_1 n)$$

$$D_{nx_2} = 2\frac{5}{40} sinc(\frac{\pi}{20} 5n)$$

$$D_{nx_2} = \frac{1}{4} sinc(\frac{\pi}{4} n)$$

Problem A.3 [3 Marks] Now that you have an expression for D_n, write a MATLAB function that generates D_n for a user specified range of values of n.

```
function [x_1_dn] = coefficents_x_1(n)
%Genrates n fourier series coeffiecents for x_1 function
%
syms t
x_1 = cos(3*pi/10*t)+.5*cos(pi/10*t);
t0_1 = 20;
w0_1 = 2*pi/t0_1;

n_low=-1*n;
n_high=n;
n_length=n_high-n_low;
i=1;
xv_1=zeros(n_length,1);
for n=n_low:n_high

    xv_1(i,1)= n;
    xv_1(i,2) = 1/t0_1*int(x_1*exp(-1i*w0_1*t*n),0,t0_1);

    i=i+1;
    Progress_percent =(i-2)/n_length*100;

    disp(['x_1: ',num2str(Progress_percent),'%']);
end
x_1_dn=xv_1;
end
```

```
function [x_2_dn] = coefficents_x_2(n)
%Genrates n fourier series coeffiecents for x_3 function
%
syms t
x_2 = heaviside(t+5)-heaviside(t-5);
t0_2 = 20;
w0_2 = 2*pi/t0_2;

n_low=-1*n;
n_high=n;
n_length=n_high-n_low;
i=1;
xv_2=zeros(n_length,1);
for n=n_low:n_high

    xv_2(i,1)= n;
    xv_2(i,2) = 1/t0_2*int(x_2*exp(-1i*w0_2*t*n),-1*t0_2,t0_2);

    i=i+1;
    Progress_percent=(i-2)/n_length*100;
    disp(['x_2: ',num2str(Progress_percent),'%']);
end
x_2_dn=xv_2;
end
function [x_3_dn] = coefficents_x_3(n)
%Genrates n fourier series coeffiecents for x_3 function
%
syms t
x_3 = heaviside(t+5)-heaviside(t-5);
t0_3 = 40;
w0_3 = 2*pi/t0_3;

n_low=-1*n;
n_high=n;
n_length=n_high-n_low;
i=1;
xv_3=zeros(n_length,1);
for n=n_low:n_high

    xv_3(i,1)= n;
    xv_3(i,2) = 1/t0_3*int(x_3*exp(-1i*w0_3*t*n),-1*t0_3,t0_3);

    i=i+1;
    Progress_percent=(i-2)/n_length*100;
    disp(['x_3: ',num2str(Progress_percent),'%']);
end
x_3_dn=xv_3;
end
```

Problem A.4 [3 Marks] Generate and plot the **magnitude** and **phase** spectra of $x_1(t)$, $x_2(t)$ and $x_3(t)$ (using the **stem** command) from their respective D_n sets for the following index ranges:

(a) $-5 \leq n \leq 5$;

(b) $-20 \leq n \leq 20$;

(c) $-50 \leq n \leq 50$;

(d) $-500 \leq n \leq 500$.

Note: You can use the MATLAB commands **abs** and **angle** to determine the magnitude and phase of a complex number.

```
%MAIN
%Generates and Graphs dn
xv_1=coefficents_x_1(5);
graph_dn(xv_1,1);
xv_1=coefficents_x_1(20);
graph_dn(xv_1,1);
xv_1=coefficents_x_1(50);
graph_dn(xv_1,1);
xv_1=coefficents_x_1(500);
graph_dn(xv_1,1);

xv_2=coefficents_x_2(5);
graph_dn(xv_2,2);
xv_2=coefficents_x_2(20);
graph_dn(xv_2,2);
xv_2=coefficents_x_2(50);
graph_dn(xv_2,2);
xv_2=coefficents_x_2(500);
graph_dn(xv_2,2);

xv_3=coefficents_x_3(5);
graph_dn(xv_3,3);
xv_3=coefficents_x_3(20);
graph_dn(xv_3,3);
xv_3=coefficents_x_3(50);
graph_dn(xv_3,3);
xv_3=coefficents_x_3(500);
graph_dn(xv_3,3);
```

```
function [] = graph_dn(dn,fx)
%Graphs fourier Series coefficients
%
figure('name',['X_',num2str(fx),'D_n Graph']);
x=dn(:,1);
y=dn(:,2);

subplot(2,1,1);
stem(x,abs(y));
title(['X_',num2str(fx),' |D_n|'])
xlabel('n')
ylabel('|D_n|')

subplot(2,1,2);
stem(x,angle(y));
title(['X_',num2str(fx),' ∠D_n'])
xlabel('n')
ylabel('∠D_n')
end
```

See Following pages for results

X_1

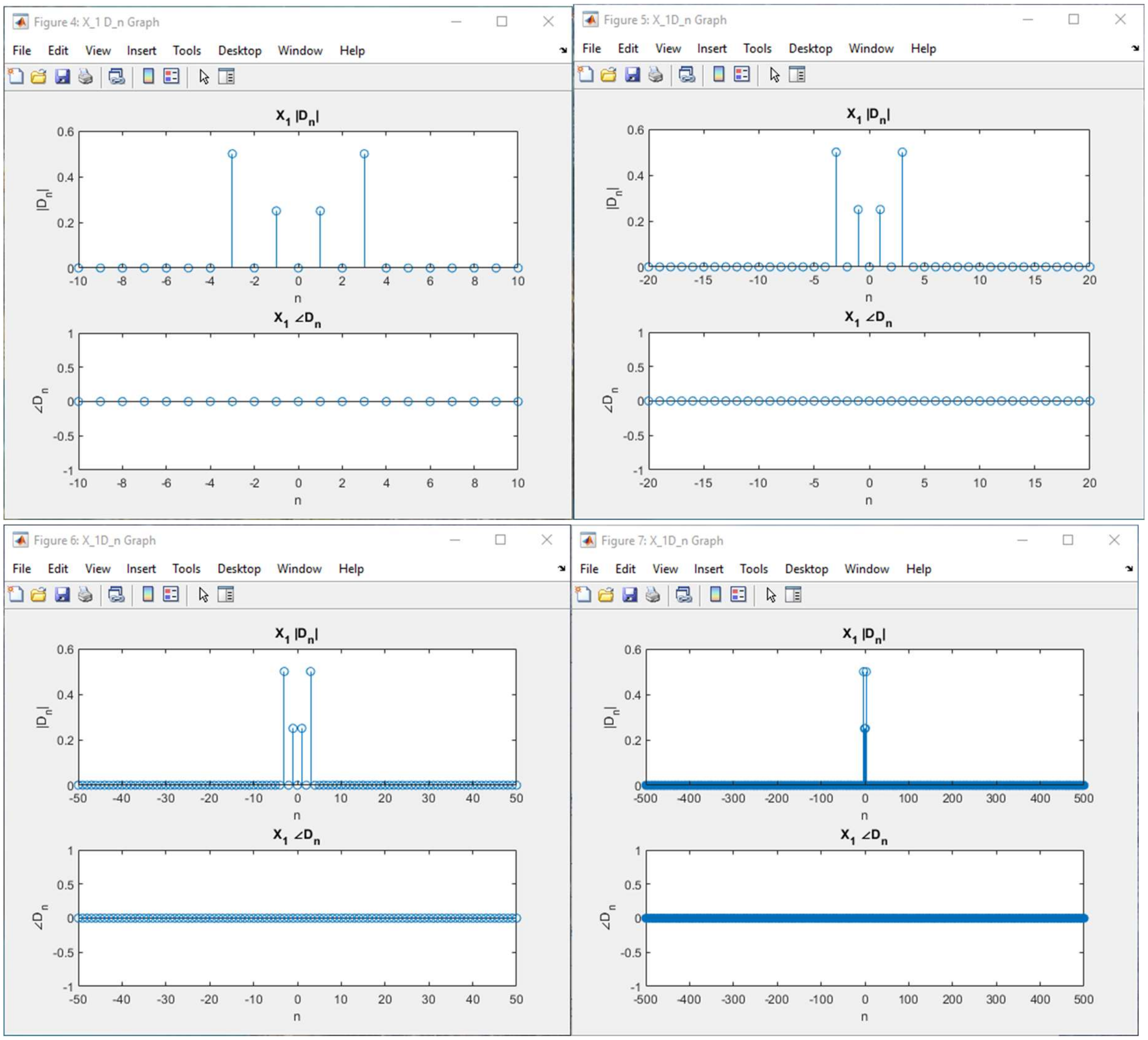


Figure 4: X_1 D_n Graph
X_1 $|D_n|$
$|D_n|$
n
X_1 $\angle D_n$
$\angle D_n$
n
Figure 5: X_1D_n Graph
X_1 $|D_n|$
$|D_n|$
n
X_1 $\angle D_n$
$\angle D_n$
n
Figure 6: X_1D_n Graph
X_1 $|D_n|$
$|D_n|$
n
X_1 $\angle D_n$
$\angle D_n$
n
Figure 7: X_1D_n Graph
X_1 $|D_n|$
$|D_n|$
n
X_1 $\angle D_n$
$\angle D_n$
n

X_2

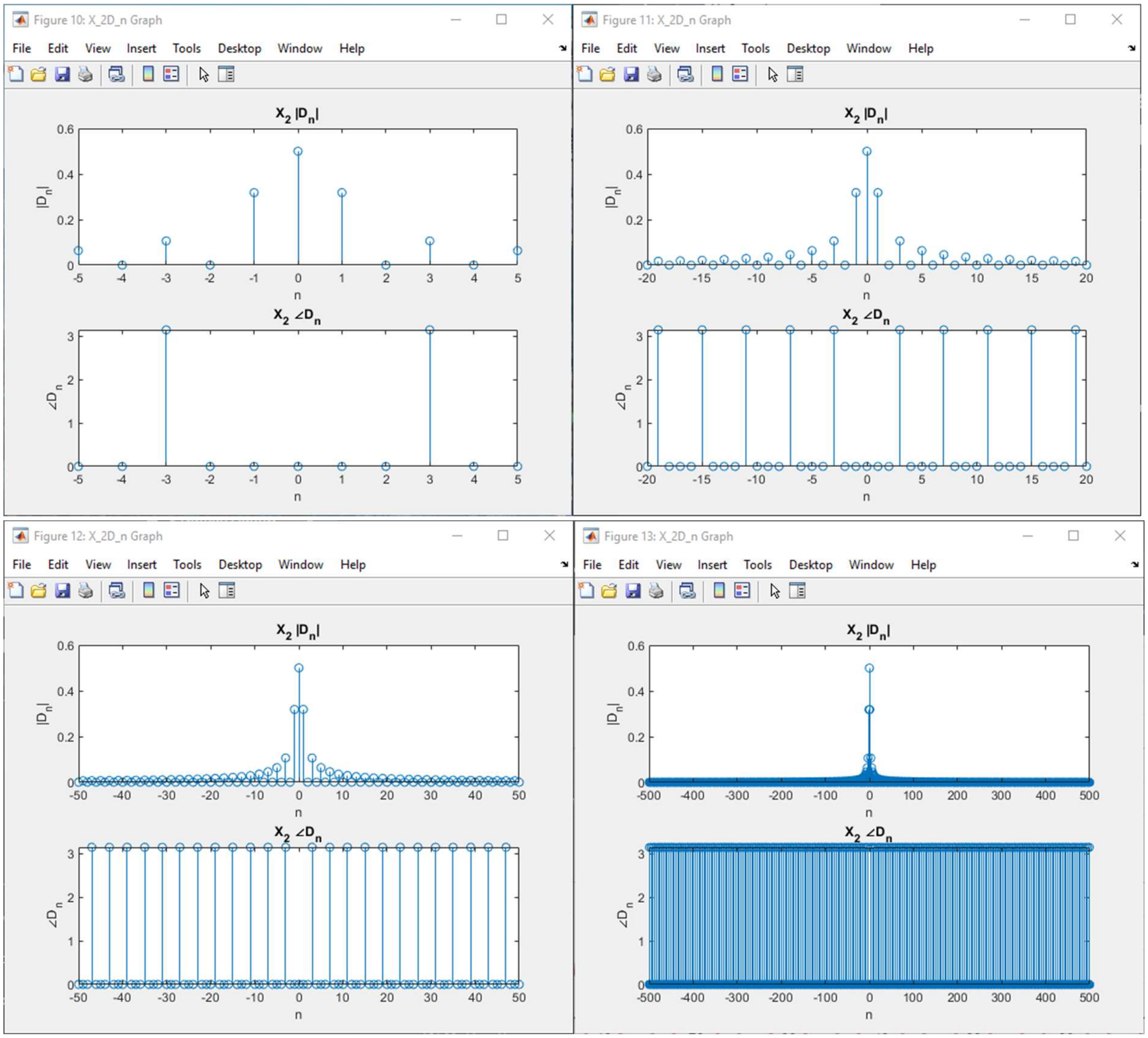


Figure 10: X_2D_n Graph
File Edit View Insert Tools Desktop Window Help
X_2 $|D_n|$
$|D_n|$
n
X_2 $\angle D_n$
$\angle D_n$
n
Figure 11: X_2D_n Graph
File Edit View Insert Tools Desktop Window Help
X_2 $|D_n|$
$|D_n|$
n
X_2 $\angle D_n$
$\angle D_n$
n
Figure 12: X_2D_n Graph
File Edit View Insert Tools Desktop Window Help
X_2 $|D_n|$
$|D_n|$
n
X_2 $\angle D_n$
$\angle D_n$
n
Figure 13: X_2D_n Graph
File Edit View Insert Tools Desktop Window Help
X_2 $|D_n|$
$|D_n|$
n
X_2 $\angle D_n$
$\angle D_n$
n

X_3

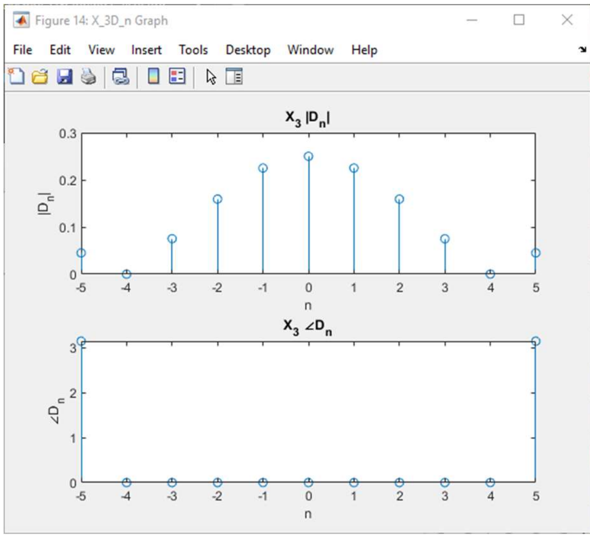


Figure 14: X_3D_n Graph
File Edit View Insert Tools Desktop Window Help
X3 |Dn|
|Dn|
0 0.1 0.2 0.3
-5 -4 -3 -2 -1 0 1 2 3 4 5
n
X3 ∠Dn
∠Dn
0 1 2 3
-5 -4 -3 -2 -1 0 1 2 3 4 5
n

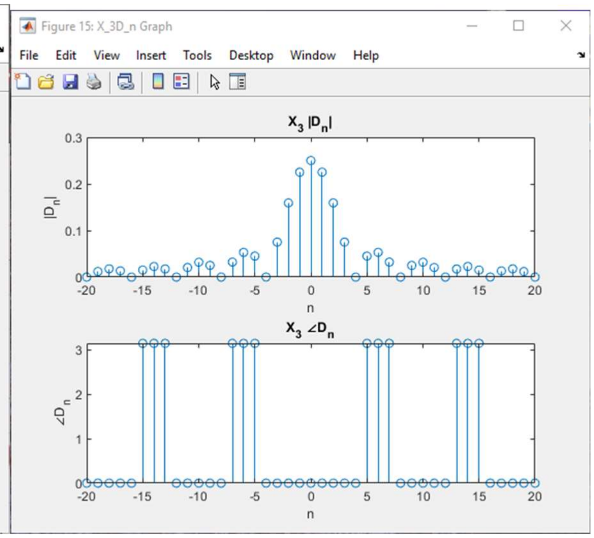


Figure 15: X_3D_n Graph
File Edit View Insert Tools Desktop Window Help
X3 |Dn|
|Dn|
0 0.1 0.2 0.3
-20 -15 -10 -5 0 5 10 15 20
n
X3 ∠Dn
∠Dn
0 1 2 3
-20 -15 -10 -5 0 5 10 15 20
n

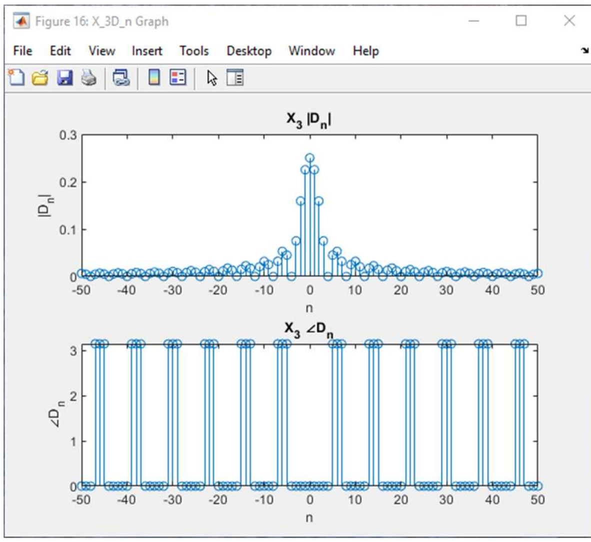


Figure 16: X_3D_n Graph
File Edit View Insert Tools Desktop Window Help
X3 |Dn|
|Dn|
0 0.1 0.2 0.3
-50 -40 -30 -20 -10 0 10 20 30 40 50
n
X3 ∠Dn
∠Dn
0 1 2 3
-50 -40 -30 -20 -10 0 10 20 30 40 50
n

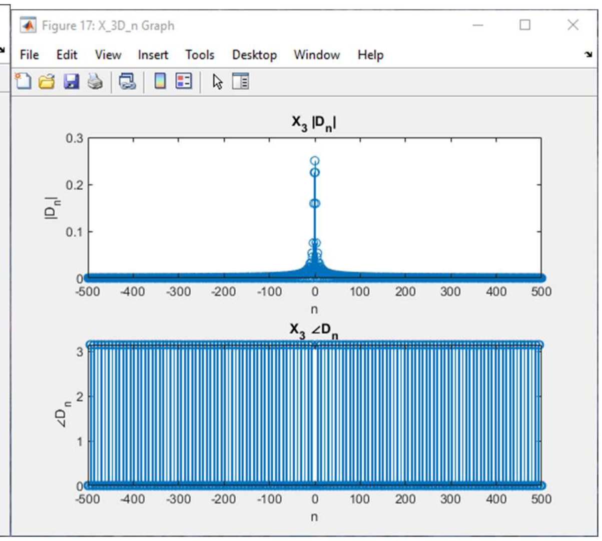


Figure 17: X_3D_n Graph
File Edit View Insert Tools Desktop Window Help
X3 |Dn|
|Dn|
0 0.1 0.2 0.3
-500 -400 -300 -200 -100 0 100 200 300 400 500
n
X3 ∠Dn
∠Dn
0 1 2 3
-500 -400 -300 -200 -100 0 100 200 300 400 500
n

Problem A.5 [3 Marks] Write a MATLAB function that takes a MATLAB generated D_n set and reconstructs the original time-domain signal from which the Fourier coefficients had been derived. For example, given the set of truncated Fourier coefficients $\{D_n,\ n = 0, \pm 1, \ldots, \pm 20\}$, your code should reconstruct the time-domain signal from this set using Equation (1). **Note:** Use the time variable t defined with the MATLAB command `t=[-300:1:300]`.

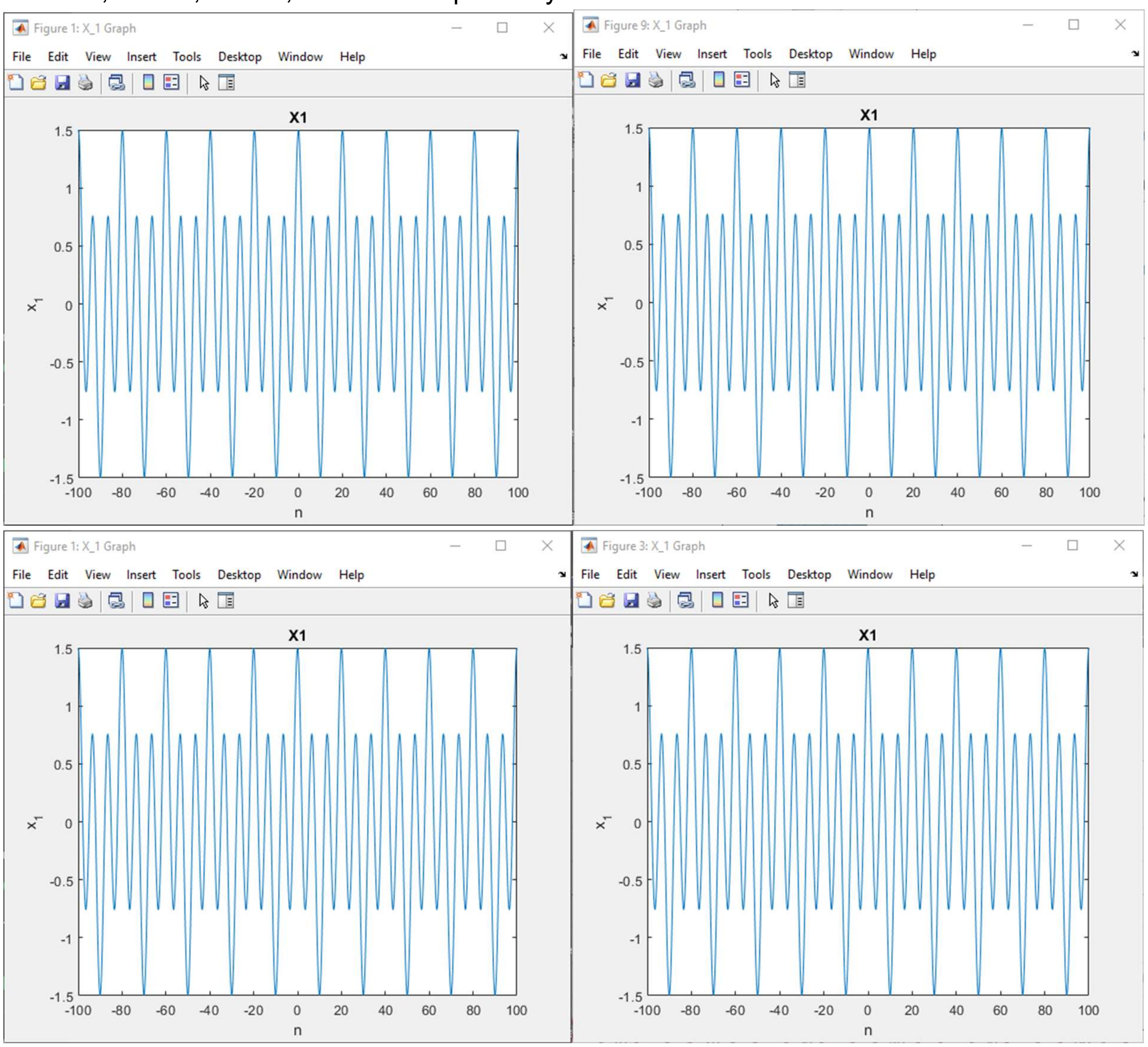
```
function [] = OriginalSignal(dn_input,w0,fx)
% ** OriginalSignal **
%Takes fourier series coeficents in matrix allong with w0 and constructs
%original signal.
t_start=-300;
t_end=300;
interval=.25;
t_length= (t_end-t_start)/interval;
n_start=dn_input(1,1);
n_length=size(dn_input,1);
n_end=n_start+n_length-1;
x_t=zeros(t_length,2);
u=1;
    for t = t_start:interval:t_end
        i=1;
        x=0;
        for n = n_start:n_end
            dn_temp=dn_input(i,2);
            n_temp = dn_input(i,1);
            x = x + dn_temp*exp(-1i*w0*n_temp*t);
            i=i+1;
        end
        %disp(x);
        x_t(u,1)=t;
        x_t(u,2)=x;
        u=u+1;

        Progress_percent =(u-2)/t_length*100;
        disp(['x_t: ',num2str(Progress_percent),'%']);
    end
%disp(x_t);
disp('done');
xdata=x_t(:,1);
ydata=x_t(:,2);
figure('name',['X_',num2str(fx),' Graph']);
plot(xdata,ydata);
title(['X',num2str(fx)])
xlabel('n')
ylabel(['x_',num2str(fx)])
end
```

Problem A.6 [3 Marks] Reconstruct the time-domain signals $x_1(t)$, $x_2(t)$ and $x_3(t)$ with the Fourier coefficient sets you generated in Problem A.4. Plot each reconstructed signal.

```
%Reconstruct x_1
OriginalSignal(xv_1_5,pi/10,1);
OriginalSignal(xv_1_20,pi/10,1);
OriginalSignal(xv_1_50,pi/10,1);
OriginalSignal(xv_1_500,pi/10,1);
```

X_1

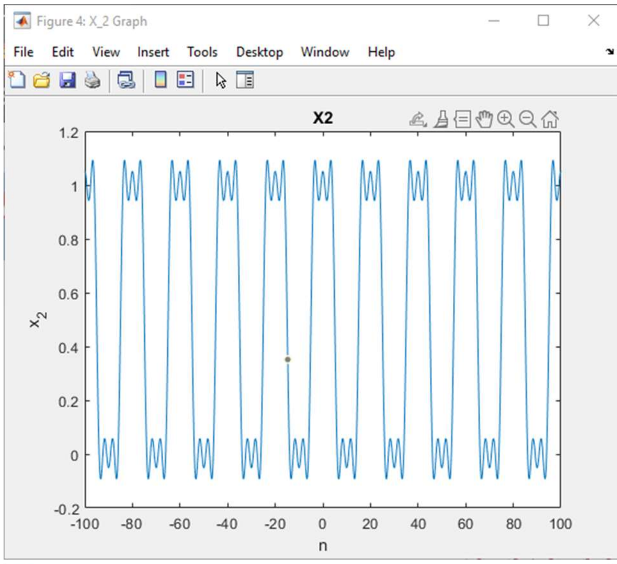
N = -5:5, -20:20, -50:50, 500:-500 respectively

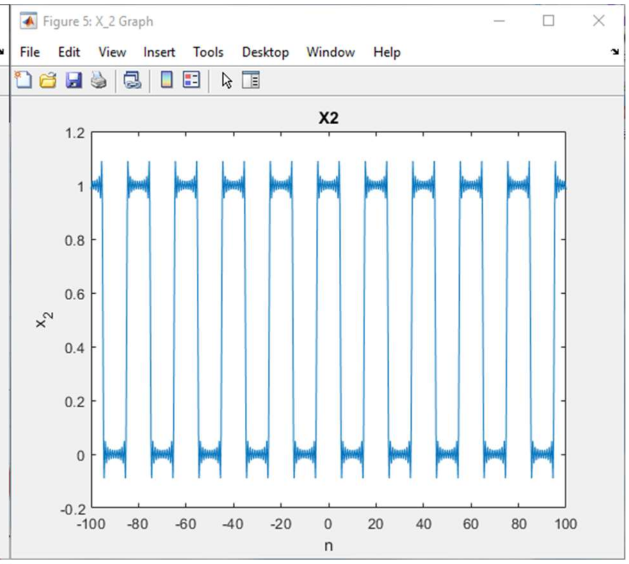


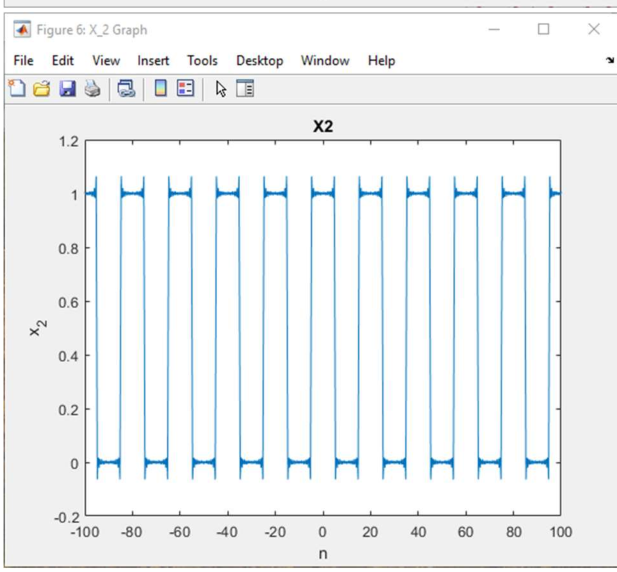
```
%Reconstruct x_2
OriginalSignal(xv_2_5,pi/10,2);
OriginalSignal(xv_2_20,pi/10,2);
OriginalSignal(xv_2_50,pi/10,2);
OriginalSignal(xv_2_500,pi/10,2);
```

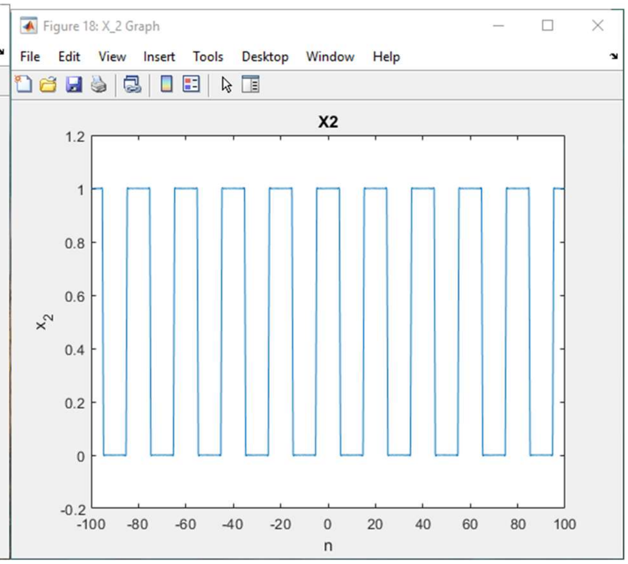
X_2

N = -5:5, -20:20, -50:50, 500:-500 respectively



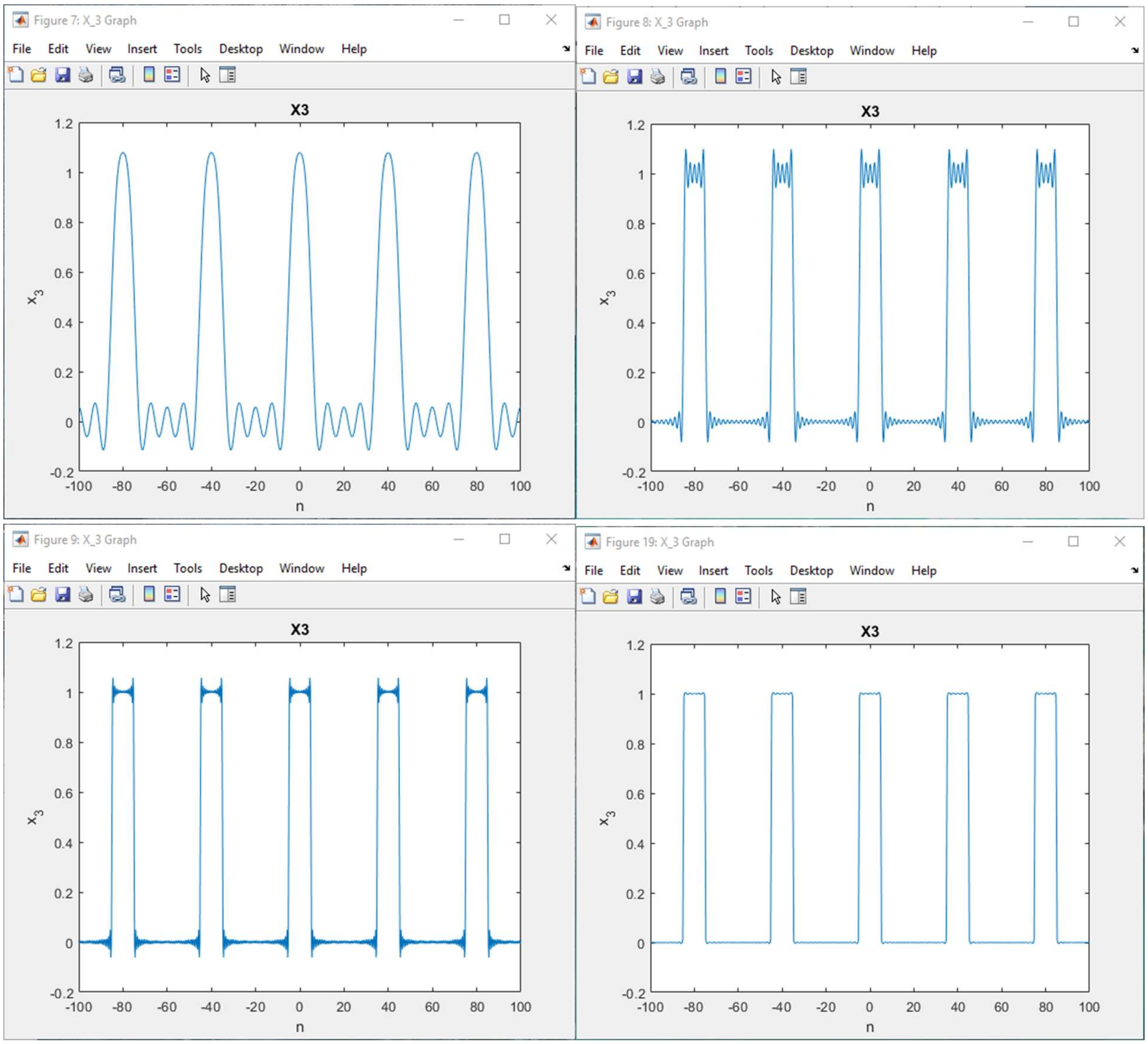






```
%Reconstruct x_3
OriginalSignal(xv_3_5,pi/20,3);
OriginalSignal(xv_3_20,pi/20,3);
OriginalSignal(xv_3_50,pi/20,3);
OriginalSignal(xv_3_500,pi/20,3);
```

X_3
N = -5:5, -20:20, -50:50, 500:-500 respectively



B. Discussion

Problem B.1 [1 Mark] Determine the fundamental frequencies of $x_1(t)$, $x_2(t)$ and $x_3(t)$.

The fundamental frequencies for $x_1(t)$, $x_2(t)$, and $x_3(t)$ are π/10, π/10, and π/20 respectively.

1. We can find the fundamental frequency for $x_1(t)$ by simply finding the common fundamental period of both portions of the function and then dividing 2π by that number to get π/10
2. For the functions $x_2(t)$, and $x_3(t)$ we looked at the graphs and saw graphically their fundamental frequencies were π/10 and π/20.

Problem B.2 [1 Mark] What is the main difference between the Fourier coefficients of $x_1(t)$ and $x_2(t)$?

The function $x_2(t)$ is a periodic function and so it's Fourier coefficients go on infinitely whereas for $x_1(t)$ we were only given two snippets of the function and so it only has a limited number of Fourier coefficients.

Problem B.3 [1 Mark] Signals $x_2(t)$ and $x_3(t)$ have the same rectangular pulse shape but different periods. How are these characteristics reflected in their respective Fourier coefficients?

The difference in period is reflected in the general equation that supplies the fourier coefficients for both $x_2(t)$, and $x_3(t)$; $D_n = 2\frac{T_1}{T_0} sinc(\omega_0 T_1 n)$

The general equation contains the term T_1/To and so it's obvious the amplitude of the sinc function is what will be reliant on T_0. Additionally, the function with the smaller period will have more fourier coefficients, in this case that's $x_2(t)$.

Problem B.4 [2 Marks] The Fourier coefficient D_0 represents the DC value of the signal. Let $x_4(t)$ be the periodic waveform shown in Figure 2. Derive D_0 of $x_4(t)$ from D_0 of $x_2(t)$.

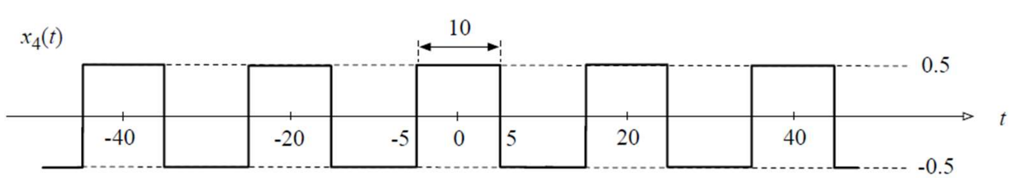
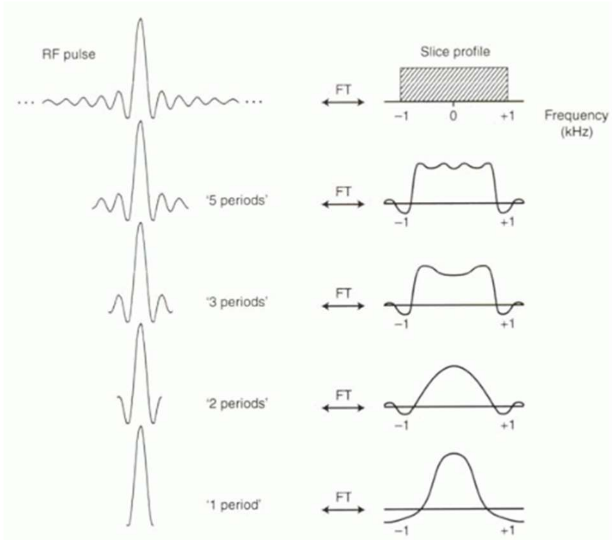


Figure 2: Periodic function $x_4(t)$.

It is known that the D_0 of $x_2(t)$ and $x_4(t)$ will be the same on x = 0 because they overlap on the same position for the DC portion of the signal, therefore we can determine D_0 of $x_4(t)$ by using $x_2(t)$, and then considering the 0.5 vertical shift for $x_4(t)$. Using our equation $D_{nx_2} = 2\frac{5}{20} sinc(\frac{\pi}{10} 5n)$and substituting n = 0 we get: 0.5.

Problem B.5 [2 Marks] Using the results of Problem A.6, explain how the reconstructed signal changes as you increase the number of Fourier coefficients used in the reconstruction. Discuss for both $x_1(t)$ and $x_2(t)$.

The reconstructed signal changes to become more similar to the original signal when you increase the number of fourier coefficients used in the reconstruction, this is true for both $x_1(t)$, $x_2(t)$, and all other functions. An example of this can be seen in the following image:



Problem B.6 [2 Marks] How many Fourier coefficients do you need to **perfectly** reconstruct the periodic waveforms discussed in this lab experiment?

It would take an infinite number of fourier coefficients to **perfectly** reconstruct the periodic waveforms discussed in this lab experiment because the point of the fourier series is to try to reconstruct the signal with as sub-functions as deemed necessary, the more sub-functions are added the closer and closer you get to a perfect reconstruction.

Problem B.7 [2 Marks] Let $x(t)$ be an arbitrary periodic signal. Instead of storing $x(t)$ on a computer, we consider storing the corresponding Fourier coefficients. When we need to access $x(t)$, we read the Fourier coefficients stored on the computer hard drive and reconstruct the signal. Is this a viable scenario? Explain your answer.

No this is not a viable scenario because it would take more information than just the fourier coefficients to reconstruct the signal; two functions can have the exact same fourier coefficients but look wildly different as whole functions - such as $\sin(\pi/3t)$ and $\cos(2t+ \pi/2)$.