International Institute of Information Technology
Bangalore

Principles of Communication Systems Lab

EC 303P

LAB 1

Mohd. Rizwan Shaikh
(IMT2019513)

August 17, 2021

LAB 1

Mohd. Rizwan Shaikh

August 17, 2021

Abstract

The aim of this lab report is to get familiar with MATLAB and basic signal properties. In this lab report we plot various continuous and discrete-time signals along with their convolution, auto-correlation and cross-correlation. We also plot the amplitude and phase spectrum of the signals.

1 Introduction

The aim of this lab report is to get familiar with MATLAB and basic signal properties and operations.

1.1 Question 1

In the first question, we consider the following signal for five complete cycles:

$$x(t) = sin(2\pi 50t) \tag{1}$$

where t is time.
For this signal, we plot and analyze the following:

(a) Analog continuous signal.

(b) Analog discrete-time signal.

(c) Quantized positive samples to 1 and negative samples to -1 and plot:

  (i) Digital discrete-time signal.

  (ii) Digital continuous signal.

(d) Amplitude spectrum.

(e) Phase spectrum.

1.2 Question 2

In this question, the following discrete time signals are given:

$$x_1[n] = [-1, 1, 2, 2, -2] \tag{2}$$

$$x_2[n] = [2, 1, -1, -2, -1, 1] \tag{3}$$

For these signals, we plot the following:

(a) $x_1[n]$ and x2[n].

(b) Auto-correlation of both $x_1[n]$ and $x_2[n]$.

(c) Cross-correlation between $x_1[n]$ and $x_2[n]$.

(d) Convolution between $x_1[n]$ and $x_2[n]$.

(e) Cross-correlation between $x_1[n]$ and $x_2[-n]$.

1.3 Question 3

In this question, we create a rectangular pulse with amplitude $A = 1$ and duration $\tau = 1$ second and plot the following:

(a) The pulse.

(b) Frequency spectrum of the pulse.

(c) Phase spectrum of the pulse.

2 Method

2.1 Question 1

To begin with, the signal vector (x) is implemented as follows:

```
t = 0:1/2000:0.1;            %Time vector
x = sin(2*pi*50*t);          %signal vector
```

(a) Using `plot(t,x)`, the analog continuous signal is plotted.

(b) Similarly, for plotting analog discrete-time signal `stem(t,x)` is used.

(c) A simple `for` loop is used to get quantized values of the signal in `xq` vector as shown below:

```
xq = x;                      %xq contains quantized values of the signal
for i = 1:length(t)
    if x(i) ≥ 0
        xq(i) = 1;
    else
        xq(i) = -1;
    end
end
```

(d) The Fourier transform of the sinusoidal signal is obtained through the following code:

```
N = 10000;                   %Number of samples
ts = 0.0002;                 %time
df = 1/(N*ts);               %frequency gap
t = [0:N-1]*ts;              %Time vector
x = sin(2*pi*50*t);
```

```
    Xf_shift = fftshift(fft(x))/length(x);  %perform zero-centered ...
        circular shift on fourier transform and normalize it
    Ff_shift = [-N/2:N/2-1]*df;    %Get frequency vector
```

This is followed by `plot(Ff_shift, abs(Xf_shift))` to plot the amplitude spectrum. Getting the amplitude and phase spectrum was challenging for the first time. My initial code for finding the Fourier Transform and getting the amplitude spectrum is as follows:

```
    fs = 1000;                  %Sampling frequency
    t = 0:1/fs:0.1;             %Time vector
    x = sin(2*pi*50*t);         %Signal vector
    y = fft(x);
    f = (0:length(y)-1)*fs/length(y);
    plot(f, abs(y));
```

On plotting, we realize that the amplitude spectrum is dependent on sampling frequency as shown below:

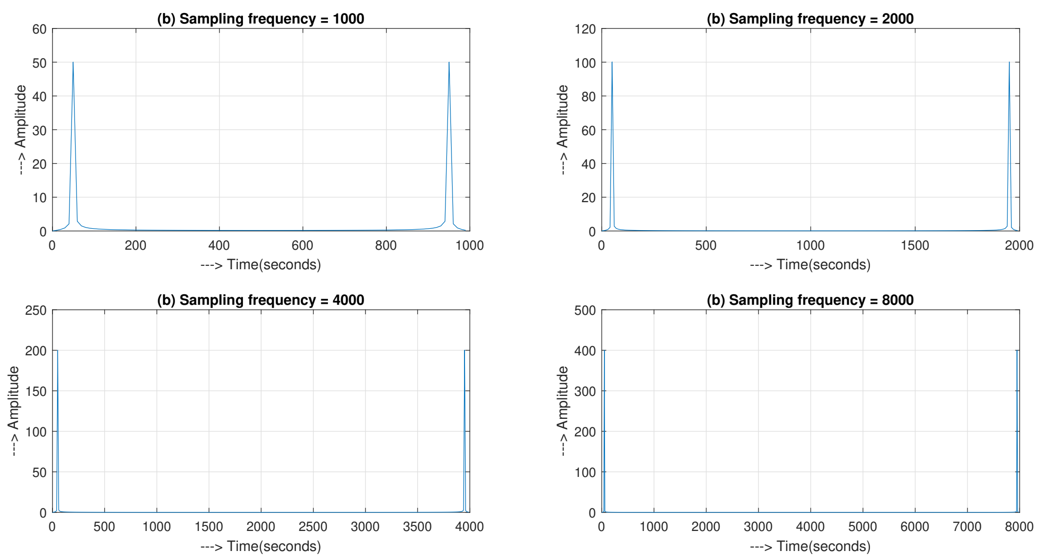

Figure 1: The amplitude spectrum as obtained by the first code.

This is contrary to the fact that the peaks in amplitude spectrum of sinusoidal wave are at $\pm f$ hertz (f is frequency of sine wave). Moreover, the maximum amplitude of the spectrum should be at 0.5. Thus, it is evident, that the above code is either wrong or incomplete. After going though the documentation of Fourier Transform in MATLAB, I used `fftshift` command to perform zero-centered circular shift on the transform. After that, the output is normalized by dividing it by `length(x)` to get the desired amplitude of 0.5 at ± 50 Hz.

(e) After performing the required operations(`fftshift`) on Fourier transform, phase spectrum is plotted using `plot(Ff_shift, angle(Xf_shift)*180/pi)`.

Using `subplot` command, all the sub-parts are plotted in the same plot.

2.2 Question 2

The signal vectors $x_1[n]$ and $x_2[n]$ are implemented along with their length vectors $n1$ and $n2$ as follows:

```
x1_n = [-1 1 2 2 -2];           %x1[n]
x2_n = [2 1 -1 -2 -1 1];        %x2[n]
n1 = 0:length(x1_n)-1;          %n1
n2 = 0:length(x2_n)-1;          %n2
```

(a) To plot $x_1[n]$ and $x_2[n]$, `stem(n1, x1_n)` and `stem(n2, x2_n)` are used respectively. Here `x1_n` and `x2_n` are signal vectors and `n1` and `n2` are the length vectors of `x1_n` and `x2_n` respectively.

(b) Using `[y1, x1] = xcorr(x1_n)` we find the auto-correlation of $x_1[n]$. This is followed by `stem(x1, y1)` for plotting. Using same steps, we obtain the auto-correlation of $x_2[n]$.

(c) To obtain cross-correlation of $x_1[n]$ and $x_2[n]$, we use `[y3, x3] = xcorr(x1_n, x2_n)`. Plotting is done by `stem(x3, y3)`.

(d) To find the convolution of $x_1[n]$ and $x_2[n]$, we use `y = conv(x1_n, x2_n)`. Next, we find `ny` vector by `ny = 0:(length(y)-1)`. Using `stem(ny, y)`, we plot the convolution of the signals.

(e) First of all, we flip `x2_n` vector to get $x_2[-n+6]$. Next, we find the cross-correlation of $x_1[n]$ and $x_2[-n+6]$ using `[y4, x4] = xcorr(x1_n, x2_n)`. Before plotting, we shift the output signal by 5 units to the right to get cross-correlation of $x_1[n]$ and $x_2[-n]$. To plot, we use `stem(x4, y4)`.

Using `subplot` command, all the sub-parts are plotted in the same plot.

2.3 Question 3

The signal vector of rectangular pulse is implemented as follows:

```
fs = 10000;                        %Sampling frequency
t = -1:1/fs:1;                     %Time vector
x = 1*(t ≥ -0.5 & t ≤ 0.5);        %Signal vector
```

(a) The rectangular pulse is plotted using `plot(t,x)` command.

(b) To plot the frequency spectrum, we first find the Fourier Transform using integration as follows:

```
syms t f
X(f) = int(1*exp(-1i*2*pi*f*t), t, -0.5, 0.5);  %Find Fourier transform
```

This is followed by plotting using `fplot(X, [-1 1]*4*pi)`.

(c) To plot phase spectrum, we use `fplot(angle(X), [-1 1]*4*pi)`.

3 Results and Analysis

3.1 Question 1

The MATLAB code for this question is as follows:

```
clear all;
close all;
clc;

t = 0:1/2000:0.1;            %Time vector
x = sin(2*pi*50*t);          %signal vector

%<------1(a)------>
subplot(3,2,1);
plot(t,x);                 %Plot analog continuous signal
xlabel('---> time(s)');    ylabel('---> x(t)');
title('(a) Analog Continuous Signal');
grid on;

%<------1(b)------>
subplot(3,2,2);
stem(t,x);                 %Plot analog discrete time signal
xlabel('---> time(s)');    ylabel('---> x(t)');
title('(b) Analog Discrete-Time Signal');
grid on;

%<------1(c1)------>
%<---Quantize the signal--->
xq = x;                    %xq contains quantized values of the signal
for i = 1:length(t)
    if x(i) ≥ 0
        xq(i) = 1;
    else
        xq(i) = -1;
    end
end

subplot(3,2,3);
stem(t,xq);                %Plot quantized digital discrete-time Signal
xlabel('---> time(s)');    ylabel('---> x(t)');
title('(c1) Quantized Digital Discrete-Time Signal');
grid on;

%<------1(c2)------>
subplot(3,2,4);
plot(t,xq);                %Plot quantized digital continuous signal
xlabel('---> time(s)');    ylabel('---> x(t)');
title('(c2) Quantized Digital Continuous Signal');
grid on;

%<------1(d)------>
ft = fft(x);
N = length(ft);            %Number of samples
ts = 0.002;                %time
df = 1/(N*ts);             %frequency gap
t = [0:N-1]*ts;            %Time vector
x = sin(2*pi*50*t);
```

```
    Xf_shift = fftshift(fft(x))/length(x);  %perform zero-centered circular ...
        shift on fourier transform and normalize it
    Ff_shift = [-N/2:N/2-1]*df;

    subplot(3,2,5);
    plot(Ff_shift, abs(Xf_shift));              %Plot amplitude spectrum
    xlim([-100 100]);               yticks(0:0.1:1);
    xlabel('---> Frequency(Hz)');   ylabel('---> Ampitude');
    title('(d) Amplitude Spectrum');
    grid on;

    %<------1(e)------>
    subplot(3,2,6);
    plot(Ff_shift, angle(Xf_shift)*180/pi);  %Plot phase spectrum
    xlim([-150 150]);               yticks(-360:90:360);
    xlabel('---> Frequency(Hz)');   ylabel('---> Phase(degrees)');
    title('(e) Phase Spectrum');
    grid on;
```

The plot obtained is as follows:

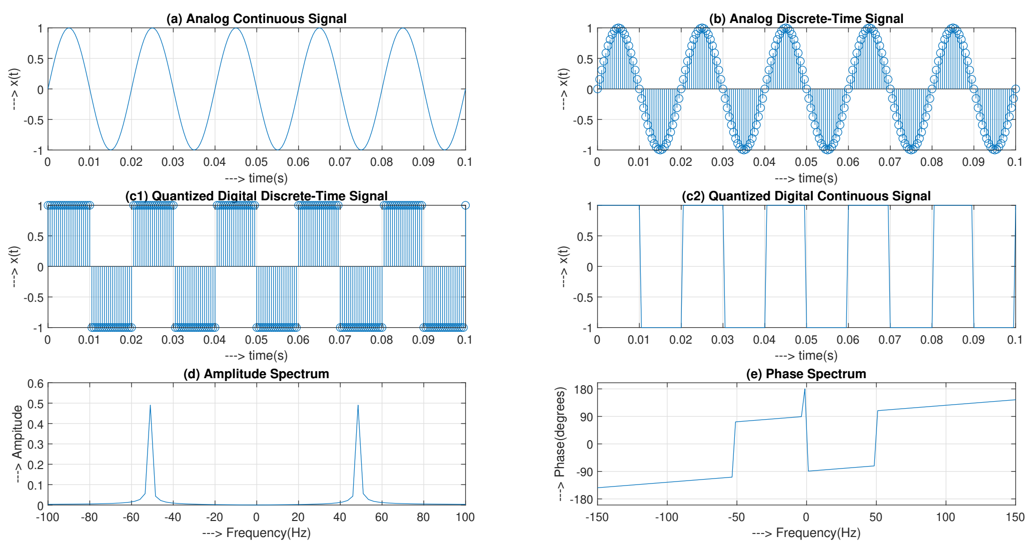

Figure 2: Plots obtained for Question 1

Here, we observe that the time period of sine wave is 0.02 seconds. After quantization, the positive values are quantized to 1 and negative values are quantized to -1. Thus the resulting wave looks like a square wave with time period of 0.02 seconds. The amplitude spectrum of the sine wave has two peaks of magnitude 0.5 at ± 50 Hz.

3.2 Question 2

The MATLAB code for this question is as follows:

```
    clear all;
    close all;
    clc;
```

```matlab

x1_n = [-1 1 2 2 -2];
x2_n = [2 1 -1 -2 -1 1];

n1 = 0:length(x1_n)-1;
n2 = 0:length(x2_n)-1;

%<------2(a)------>
subplot(2,4,1);
stem(n1, x1_n);
xlabel('---> n');   ylabel('---> x_1[n]');
xticks(0:1:4);      yticks(-3:1:3);
title('(a) Signal x_1[n]');
grid on;

subplot(2,4,2);
stem(n2, x2_n);
xlabel('---> n');   ylabel('---> x_2[n]');
xticks(0:1:6);      yticks(-3:1:3);
title('(a) Signal x_2[n]');
grid on;

%<------2(b)------>
[y1, x1] = xcorr(x1_n);
subplot(2,4,3);
stem(x1, y1);
xlabel('---> Lag');   ylabel('---> Autocorrelation');
xticks(-4:1:4);       yticks(-4:2:14);
title('(b) Autocorrelation of x_1[n] signal');
grid on;

[y2, x2] = xcorr(x2_n);
subplot(2,4,4);
stem(x2, y2);
xlabel('---> Lag');   ylabel('---> Autocorrelation');
xticks(-10:1:10);     yticks(-16:2:20);
title('(b) Autocorrelation of x_2[n] signal');
grid on;

%<------2(c)------>
[y3, x3] = xcorr(x1_n, x2_n);
subplot(2,4,5);
stem(x3, y3);
xlabel('---> Lag');   ylabel('---> cross-correlation');
xticks(-6:1:6);       yticks(-10:2:10);
title('(c) Cross-correlation between x_1[n] and x_2[n]');
grid on;

%<------2(d)------>
y = conv(x1_n, x2_n);
ny = 0:(length(y)-1);
subplot(2,4,6:7);
stem(ny, y);
xlabel('---> n');     ylabel('---> Convolution');
xticks(0:1:10);       yticks(-10:2:10);
title('(d) Convolution between x_1[n] and x_2[n]');
grid on;

%<------2(e)------>
```

```
x2_n = flip(x2_n);
[y4, x4] = xcorr(x1_n, x2_n);
x4 = 0:1:length(x4)-1;
subplot(2,4,8);
stem(x4, y4);
xlabel('---> Lag');    ylabel('---> cross-correlation');
xticks(0:1:10);       yticks(-10:2:10);
title('(e) cross-correlation between x_1[n] and x_2[-n]');
grid on;
```

The plot obtained is as follows:

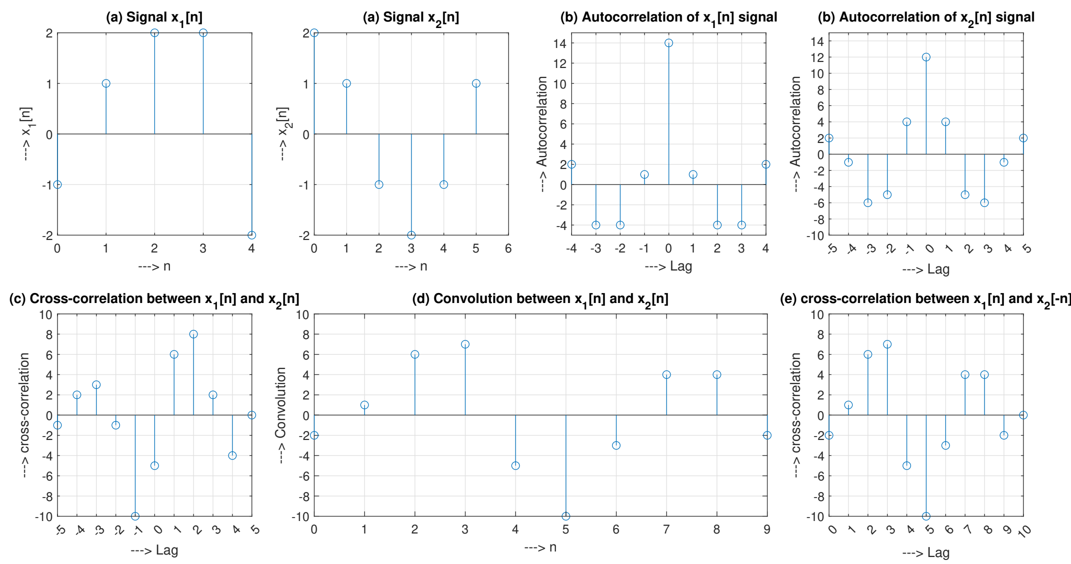

Figure 3: Plots obtained for Question 2

In the above figure, we plot signals $x_1[n]$ and $x_2[n]$ (a) along with auto-correlation (b), cross-correlation of $x_1[n]$ and $x_2[n]$ (c), convolution (d) and cross-correlation between $x_1[n]$ and $x_2[-n]$ (e).

3.3 Question 3

The MATLAB code for this question is as follows:

```
fs = 10000;                         %Sampling frequency
t = -1:1/fs:1;                      %Time vector
x = 1*(t ≥ -0.5 & t ≤ 0.5);    %Signal vector

%<------3(a)------>
subplot(2,2,1:2);
plot(t,x);                  %Plot rectangular pulse
xlabel('---> time(s)');    ylabel('---> x(t)');
title('(a) Rectangular Pulse');
grid on;

%<------3(b)------>
syms t f
```

```
X(f) = int(1*exp(-1i*2*pi*f*t), t, -0.5, 0.5);  %Find Fourier transform

subplot(2,2,3);
fplot(X, [-1 1]*4*pi);       %Plot frequency spectrum
xticks(-16:2:16);
xlabel('---> Frequency(Hz)'); ylabel('---> X(f)');
title('(b) Frequency Spectrum of Signal');
grid on;

% %<------3(c)------>
subplot(2,2,4);
fplot(angle(X), [-1 1]*4*pi); %Plot phase spectrum
xticks(-8:2:8);
xlim([-10 10]);
set(gca,'ytick',[0:pi/4:2*pi]);
set(gca,'yticklabels',{'0','\pi/4','\pi/2', '3\pi/4', '\pi'});
xlabel('---> Frequency(Hz)');   ylabel('---> Phase(rad)');
title('(c) Phase Spectrum of Signal');
grid on;
```

The plot obtained is as follows:

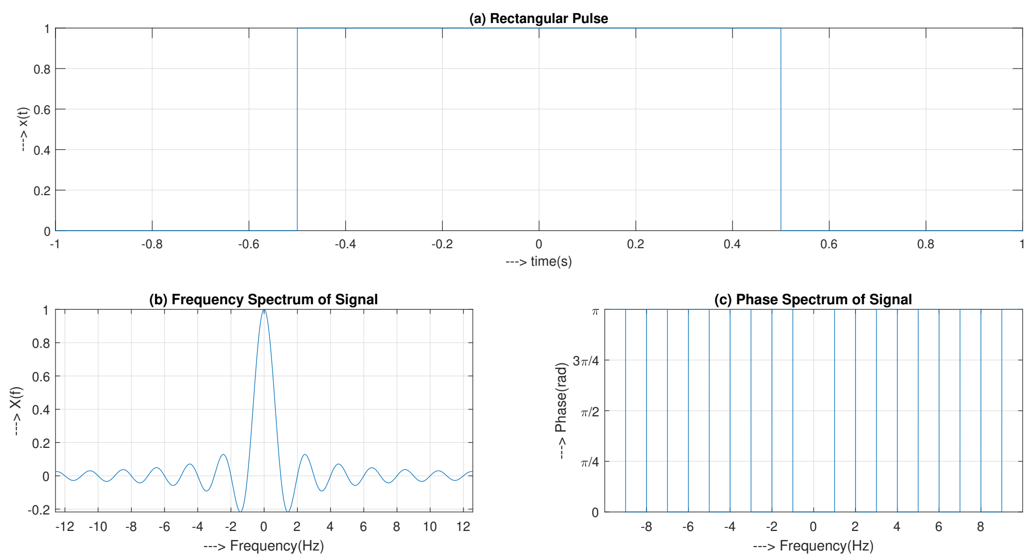

Figure 4: Plots obtained for Question 3

Here, we plot a rectangular pulse of amplitude 1 and duration 1 second (a). The obtained frequency plot (b) is in agreement with the equation of frequency spectrum of rectangular pulse which is $sinc(f)$.

4 Discussion and Conclusion

4.1 Question 1

We plotted the signals. All the plots obtained are in accordance with the expected values.

4.2 Question 2

We plotted the required graphs and the plots are in accordance with the expected values.

4.3 Question 3

We plotted the rectangular pulse. The frequency spectrum and phase spectrum are in accordance with the expected values.

References

[1] MATLAB: Fourier Transforms,
https://in.mathworks.com/help/matlab/math/fourier-transforms.html

[2] Fast Fourier Transform and MATLAB Implementation,
https://personal.utdallas.edu/ dlm/3350%20comm%20sys/FFTandMatLab-wanjun%20huang.pdf

[3] Evaluating Fourier Transforms with MATLAB,
http://www.csun.edu/ skatz/ece460/matlab$_{t}ut_{t}wo.pdf$