EXPERIMENT 4

Discrete Time Fourier Series and its Properties

Objective

In this experiment you will study how to decompose a periodic signal into a sum of sinusoidal signal components and obtain a pictorial representation for the frequency components that are contained in the signal.

Introduction

This discrete-time Fourier series representation provides notions of frequency content of discrete time signals, and it is very convenient for calculations involving linear, time-invariant systems.

For a signal $\boldsymbol{x(n)}$ with fundamental period N, the DTFS equations are given by:

$$x(n) = \sum_{k=0}^{N-1} a_k e^{jk(2\pi/N)n}$$

$$a_k = \frac{1}{N}\sum_{n=0}^{N-1} x(n)e^{-jk(2\pi/N)n}$$

<u>**Exercise 1:**</u>

Write a function and implement it on MATLAB, to compute the discrete time Fourier series as described below:

function y = DTFS(x, N)

where,
$\boldsymbol{x \;\; - \;\; input\ vector}$
$\boldsymbol{N \;\; - \;\; Fundamental\ period}$

Exercise 2:

Find DTFS of the following function:

$$\boldsymbol{x(n)} = cos(\frac{\pi n}{3})$$

Also find power density spectrum by applying Parseval's theorem.
Construct a function for Parseval's theorem as described below:

function [pow_xn, pow_Ck] = parsevals(xn,Ck)

where,

$$\boldsymbol{xn} \quad - \quad input\ vector\ in\ time\ domain$$
$$\boldsymbol{Ck} \quad - \quad Fourier\ coefficients$$

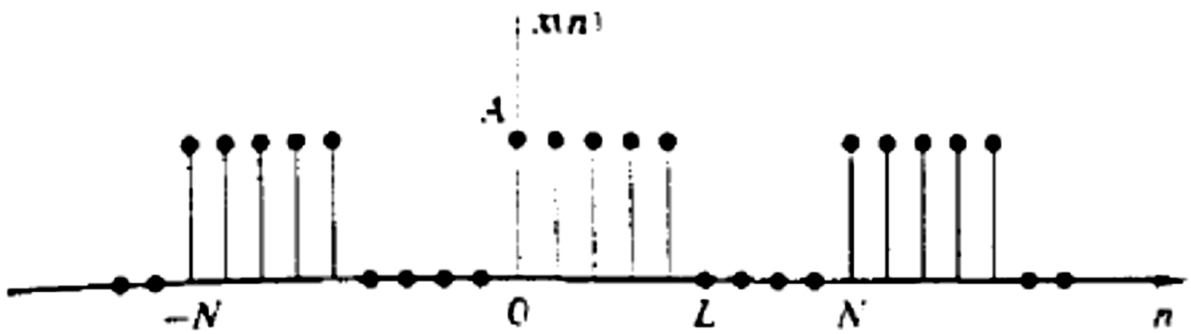
Attach the screen shot of the power density spectrum in the space provided below:

Exercise 3:
Find DTFS of the following function:

$$\boldsymbol{x(n)} = cos(\frac{\pi n}{3})$$

Also find Power Density Spectrum by applying Parseval's theorem.

Keep following values for the plots of Power Density Spectrum:
i) L=2, N=10, A=1
ii) L=2, N=40, A=1



Exercise 4:
Write a function of IDTFS ( ) as described below:

function y = IDTFS(Ck, N)

where,
$\boldsymbol{Ck} \quad - \quad Fourier\ coefficients$
$\boldsymbol{N} \quad - \quad Fundamental\ period\ of\ repetition$

Exercise 5:

Find the $\boldsymbol{x(n)}$ by applying IDTFS on the Cks obtained in **Exercise 3**.
Attach the screen shot of the $\boldsymbol{x(n)}$ in the space provided below:

PROPERTIES OF DISCRETE TIME FOURIER SERIES

Exercise 6:

Linearity property
Prove the following linearity property using:

$x(n) = \ [1\ 1\ 1\ 0]\ \ and\ \ y(n) = \ \ [2\ 1\ 2\ 4]$

A=2 and B=3

$$Ax(n) + By(n) \rightarrow AX(k) + BY(k)$$

HINT **(**For this question you have to once find DTFS of '$Ax(n) + By(n)$' and plot it and then find the DTFS of $x(n)$ and $y(n)$separately and multiply them with A and B respectively. Plot the results**)**

Attach the screen shot of the plots in the space provided below:

<u>**Exercise 7:**</u>

Time shifting property

Prove the following property for:

$$x(n)= [3\ 2\ 1\ 0]$$

$$x(n - n_0) \rightarrow X(k).\, e^{-2\pi jk(n_0/N)}$$

HINT: **(**For this question you have to once find DTFS of '$x(n - n_0)$' and plot it and then find the DTFS of $x(n)$ and multiply it with $e^{-2\pi jk(n_0/N)}$. Plot the results**)**

Attach the screen shot of the plots in the space provided below:

Post Lab:

Frequency Shifting property

Prove the following property and attach the screen shot in the space provided below:

$x(n)$= [3 2 1 0]

$x(n)e^{2\pi jk(n_0/N)} \rightarrow X(k-k_0)$

EXPERIMENT 5

Discrete Time Fourier Transform and its Properties

Objective

In this experiment you will study how to transform a signal into frequency domain and after decomposition how to get the signal back in time domain. Further on, you will also observe how the spectrum of DTFT differs from that of DTFS.

If $\boldsymbol{x(n)}$ is absolutely sum-able then its discrete time Fourier transform (DTFT) is given by

$$\boldsymbol{X(\omega) = \sum x(n).\, e^{-j\omega n}}$$

MATLAB Implementation

If $x(n)$ is of infinite duration then MATLAB cannot be used to directly compute $X(e^{j\omega})$ for $x(n)$.We can use MATLAB to evaluate its expression over the given duration. However, if $x(n)$ is of finite duration then the above equation can be used to compute $X(e^{j\omega})$. It can be implemented as a ***matrix-vector multiplication operation.*** Let us assume the sequence has M equally spaced samples and we want to evaluate $X(e^{j\omega})$ at $\omega = \frac{\pi}{M}k,\ k = 0,1,2,\ldots,M$. We can easily obtain a linear algebraic representation of the Fourier transform function $X(w) = \sum x(n)e^{-j\omega n}$.

Let $x(n)$be a sequence of length N and ω be a sequence of length M. Algebraically **Y** = **xW**, where **x** is a row vector representing the discrete-time signal $x(n)$, **Y** is a row vector representing the Fourier transform function $X(e^{j\omega})$and **W** is a *matrix* representing the complex exponential harmonic$e^{-j\omega n}$. The size of **x** is$1 \times N$(one row and N columns).

<u>Exercise 1:</u>

Write a function to estimate the discrete-time Fourier transform of a signal in the space below and implement it on MATLAB.

```
function y = DTFT(x, n, w)
```

Exercise 2:

Apply DTFT on the following signal $x(n)$. Subplot $x(n)$ and its energy density spectrum $S_{XX}(w)$:

$$x(n) = a^n u(n) \qquad\qquad -1 < a < 1$$

Assume $a = 0.5$ and $a = -0.5$ for the plots.
Also subplot the magnitude (use the function abs) and phase (use the function angle) of the function $X(\omega)$. Attach the screen shot of the plots in the space provided below:

Exercise 3:

Generate a sinc pulse (Assume $f = 1/8$ and $n = -40:40$)

I. Find its DTFT (Assume ω = -pi: pi, dω = 0.001/pi)
II. Then reconstruct the original signal using the following formula:

$$\boldsymbol{x[n] = \frac{1}{2\pi}\int_{-\pi}^{+\pi} X(e^{j\omega})e^{j\omega n}\, d\omega}$$

Attach the code in the space provided below:

PROPERTIES OF DTFT:

Exercise 4: Linearity

Let $x_1(n)$ and $x_2(n)$ be two random sequences of size 11. Let n= [0:10] and ω= [0:500]*pi/500, $\alpha = 3$, $\beta = 2$. First take individual DTFT of x_1 and x_2 and plot their sum $(\alpha X_1(\omega) + \beta X_2(\omega))$. Then add $\alpha x_1(n)$and $\beta x_2(n)$, take their DTFT and plot it. Your plots should be on the same figure.
Write a code on MATLAB and attach it in the space provided. Also use the DTFT function defined previously.

<u>Exercise 5</u>: Parseval'sTheorem

$$\boldsymbol{E} = \sum |\boldsymbol{x}(\boldsymbol{n})|^2 = \frac{\boldsymbol{1}}{\boldsymbol{2\pi}} \int_{\boldsymbol{0}}^{\boldsymbol{2\pi}} \left|\boldsymbol{X}(\boldsymbol{e}^{\boldsymbol{j\omega}})\right|^{\boldsymbol{2}} \boldsymbol{d\omega}\,, \qquad -\infty < n < +\infty$$

ω = 0:pi/100:2*pi;

x = [1 1 1]; n=[-1:1];

Write a script that separately uses the signal $\boldsymbol{x(n)}$ (the left-hand side of the above equation)**,** and its Fourier transform $\boldsymbol{X}(\boldsymbol{e}^{\boldsymbol{j\omega}})$ (the right-hand side of the above equation) to estimate energy E of a signal $x(n)$. Use the trapz function to estimate the integral.
Write down the code in the space provided below:

Exercise 6: Time shifting

Prove the following theorem given that the Fourier transform of $x(n)$ is $X(e^{jw})$
i.e. $x(n)$<->$X\left(e^{j\omega}\right)$.

$$\boldsymbol{x(n-d) \rightarrow \; e^{-j\omega d}X(e^{j\omega})}$$

Plot the signal $x(n)$ shown in Figure 6.1 and its shifted version, (shifted by -4). Attach the screen shot of their magnitude and phase spectra. In the space provided below:

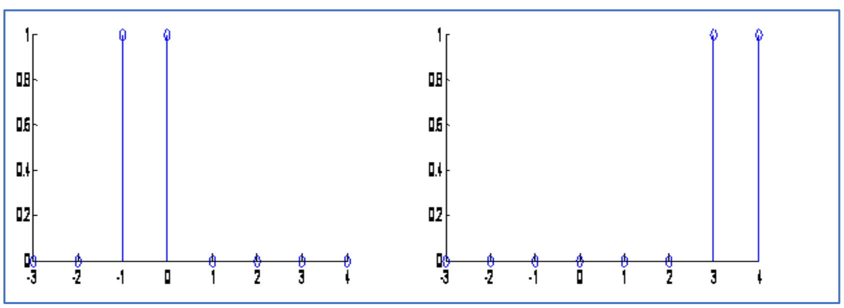


Figure 6.1: Unshifted and shifted signals

Post Lab:

Conjugation property of DTFT

x = [1/3, 1/3, 1/3] , n = [0 1 2] , ω = -pi : pi/100 : pi

Prove the conjugation property, by implementing it on MATLAB. Attach the code in the space provided below:

$$\boldsymbol{x^*(n) \rightarrow \; X^*(-\omega)}$$

EXPERIMENT 3

Quantization and SNR

Objectives

Understanding the process of converting the analogue signals into discrete time signals and then to discrete time discrete valued signals and its effect on signal to noise ratio.

Quantization

Quantization makes the range of a signal discrete, so that the quantized signal takes on only a discrete and finite set of values. Quantization is generally irreversible and results in loss of information. It therefore introduces distortion into the quantized signal that cannot be eliminated.

Exercise:

- Read and plot the audio signal (audio.wav), using audioread() command. For reading use F_S (sampling frequency) of 16000Hz.
- Find the unknowns for writing the code
    - Number of samples contained in the audio.wav file.
    - Number of bits used to encode these samples
    - Default sampling rate of audio files in MATLAB

- Write a MATLAB code to induce quantization noise in audio.wav by reducing the number of levels (i-e by reducing the number of bits)
- Using MATLAB, generate the plots of audio.wav file for different values of b. Where b is the number of bits.

Attach the code in the space provided below: