DIGITAL SIGNAL PROCESSING

ACTIVITY #5

DISCRETE TIME FOURIER TRANSFORM (DTFT)

Definition of Discrete-Time Fourier Transform,

$$X(\omega) = \sum_{-\infty}^{\infty} x[n]e^{-j\omega n}$$

MATLAB functions for DTFT,

1. real() – Complex real part.
2. imag() – Complex imaginary part.
3. abs() – Magnitude of the value.
4. angle() – Phase of the value.

Example:

1. Determine the discrete-time Fourier Transform of $x[n] = (0.5)^n u[n]$.
2. Determine the discrete-time Fourier Transform of the following finite-duration sequence: $x[n] = \{1, \underbrace{\mathbf{2}}_{\uparrow}, 3, 4, 5\}$.

DTFT Properties:

1. **Periodicity**:

$$X(e^{jw}) = X(e^{j[\omega+2\pi]})$$

Ex. Let $x(n) = (0.9e^{\frac{j\pi}{3}})^n$, $0 \le n \le 10$. Determine $X(e^{-j\omega})$ and investigate its periodicity.

```
n = 0:10;
x = (0.9*exp(1j*pi/3)).^n;
k = -200:200;
w = (pi/100)*k;
X = x * (exp(-1j*pi/100)) .^ (n'*k);
magX = abs(X);
angX =angle(X);
subplot(2,1,1); plot(w/pi,magX);grid
xlabel('frequency in units of pi');
ylabel('|X|')
title('Magnitude Part')
subplot(2,1,2); plot(w/pi,angX/pi);
grid
xlabel('frequency in units of pi');
ylabel('radians/pi')
title('Angle Part')
```

2. **Conjugate Symmetry**: $X\left(e^{-jw}\right) = X^*(e^{j\omega})$

Conditions: $Re[X(e^{-j\omega})] = Re[X(e^{j\omega})]$ (Even symmetry)

$Im[X(e^{-j\omega})] = -Im[X(e^{j\omega})]$ (Odd symmetry)

$|X(e^{-j\omega})| = |X(e^{j\omega})|$ (Even symmetry)

$\angle X(e^{-j\omega}) = -\angle X(e^{j\omega})$ (Odd symmetry)

Ex. Let $x(n) = 2^n, -10 \leq n \leq 10$. Investigate the conjugate-symmetry property of its discrete-time Fourier transform.

```
n=-5:5;
x=(-0.9).^n;
k=-200:200;
w=(pi/100)*k;
X=x*(exp(-1j*pi/100)) .^ (n'*k);
magX=abs(X);
angX=angle(X);
subplot(2,1,1);
plot(w/pi, magX);
grid;
axis([-2,2,0,15]);
xlabel('frequency in units of pi');
ylabel('|X|')
title('Magnitude Part')
subplot(2,1,2);
plot(w/pi, angX/pi);
ylabel('radians/pi');
title('Angle Part');
```

3. **Linearity:**

The discrete-time Fourier transform is a linear transformation; that is

$$ax(n) + by(n) = aX(e^{\omega}) + bY(e^{\omega})$$

```
x1=rand(1, 11);
x2=rand(1, 11);
n=0:10;
alpha=2;
beta=3;
k=0:500;
w=(pi/500)*k;
%DTFT of x1 and x2
X1=x1*(exp(-1j*pi/500)).^(n'*k);
X2=x2*(exp(-1j*pi/500)).^(n'*k);
%Linear Combination
x=alpha*x1 + beta *x2;
X= x * (exp(-1j*pi/500)).^(n'*k);
magX=abs(X);
angX=angle(X);
subplot(2, 2, 1);
plot(w/pi, magX);
xlabel('frequency in units of pi');
ylabel('|X|')
title('Magnitude Part')
subplot(2, 2, 2);
plot(w/pi, angX);
xlabel('frequency in units of pi');
ylabel('radians/pi');
title('Angle Part');
```

```
%verification
X_check=alpha*X1 + beta*X2;
magX_check=abs(X_check);
angX_check=angle(X_check);
subplot(2, 2, 3);
plot(w/pi, magX_check);
xlabel('frequency in units of pi');
ylabel('|X|')
title('Magnitude Part')
subplot(2, 2, 4);
plot(w/pi, angX_check);
xlabel('frequency in units of pi');
ylabel('radians/pi');
title('Angle Part');


%get the max error
error = max(abs(X-X_check));
disp(error)
```

4. **Time Shifting**: A shift in the time domain corresponds to the phase shifting.

$$x(n-n_0) = e^{-j\omega n_0} X(e^{-j\omega})$$

Ex. Let $x(n)$ be a random sequence uniformly distributed between $[0, 1]$ over $0 \leq \mathrm{n} \leq 10$ and let $y(n) = x(n - 2)$.

```
x = rand(1,11); n = 0:10;
k = 0:500; w = (pi/500)*k;
X = x * (exp(-1j*pi/500)).^(n'*k); % DTFT of x
magX=abs(X);
angX=angle(X);
subplot(3, 2, 1);
plot(w/pi, magX);
xlabel('frequency in units of pi');
ylabel('|X|')
title('Magnitude Part')
subplot(3, 2, 2);
plot(w/pi, angX);
xlabel('frequency in units of pi');
ylabel('radians/pi')
title('Angle Part')
% signal shifted by two samples
y = x; m = n+2;
Y = y * (exp(-1j*pi/500)).^(m'*k); % DTFT of y
magY=abs(Y);
angY=angle(Y);
subplot(3, 2, 3);
plot(w/pi, magY);
xlabel('frequency in units of pi');
ylabel('|Y|')
title('Magnitude Part')
```

```
subplot(3, 2, 4);
plot(w/pi, angY);
xlabel('frequency in units of pi');
ylabel('radians/pi')
title('Angle Part')

% verification
Y_check = (exp(-1j*2).^w).*X; % multiplication by exp(-j2w)
magY_check=abs(Y_check);
angY_check=angle(Y_check);
subplot(3, 2, 5);
plot(w/pi, magY_check);
xlabel('frequency in units of pi');
ylabel('|Y_check|')
title('Magnitude Part')
subplot(3, 2, 6);
plot(w/pi, angY_check);
xlabel('frequency in units of pi');
ylabel('radians/pi')
title('Angle Part')

error = max(abs(Y-Y_check)); % Difference
disp(error)
```

5. **Frequency Shifting**:

Multiplication by a complex exponential corresponds to a shift in the frequency domain.

$$e^{-j\omega_0 n}x(n) = X(e^{j(\omega-\omega_0)})$$

Ex. To verify the frequency shift property, we will use the graphical approach.

Let $x(n) = \cos\left(\frac{\pi n}{2}\right)$, $0 \le n \le 100$ and $y(n) = e^{\frac{j\pi n}{4}}x(n)$

```
n = 0:100; x = cos(pi*n/2);
k = -100:100; w = (pi/100)*k; % frequency between -pi and +pi
X = x * (exp(-1j*pi/100)).^(n'*k); % DTFT of x
%
y = exp(1j*pi*n/4).*x; % signal multiplied by exp(j*pi*n/4)
Y = y * (exp(-1j*pi/100)).^(n'*k); % DTFT of y
% Graphical verification
subplot(2,2,1); plot(w/pi,abs(X)); grid; axis([-1,1,0,60])
xlabel('frequency in pi units'); ylabel('|X|')
title('Magnitude of X')
subplot(2,2,2); plot(w/pi,angle(X)/pi); grid; axis([-1,1,-1,1])
xlabel('frequency in pi units'); ylabel('radiands/pi')
title('Angle of X')
subplot(2,2,3); plot(w/pi,abs(Y)); grid; axis([-1,1,0,60])
xlabel('frequency in pi units'); ylabel('|Y|')
title('Magnitude of Y')
subplot(2,2,4); plot(w/pi,angle(Y)/pi); grid; axis([-1,1,-1,1])
xlabel('frequency in pi units'); ylabel('radians/pi')
title('Angle of Y')
```

6. **Folding**: Folding in the time domain corresponds to the folding in the frequency domain.

$$x(-n) = X(e^{-j\omega})$$

Ex. To verify the folding property, let x(n) be a random sequence over $-5 \leq n \leq 10$ uniformly distributed between $[0, 1]$.

```
n = -5:10; x = rand(1,length(n));
k = -100:100; w = (pi/100)*k; % frequency between -pi and +pi
X = x * (exp(-1j*pi/100)).^(n'*k); % DTFT of x
% folding property
y = fliplr(x); m = -fliplr(n); % signal folding
Y = y * (exp(-1j*pi/100)).^(m'*k); % DTFT of y
% verification
Y_check = fliplr(X); % X(-w)
error = max(abs(Y-Y_check)); % Difference
disp(error)
subplot(3,2,1); plot(w/pi,abs(X)); grid;
xlabel('frequency in pi units'); ylabel('|X|')
title('Magnitude of X')
subplot(3,2,2); plot(w/pi,angle(X)/pi); grid;
xlabel('frequency in pi units'); ylabel('radiands/pi')
title('Angle of X')
subplot(3,2,3); plot(w/pi,abs(Y)); grid;
xlabel('frequency in pi units'); ylabel('|Y|')
title('Magnitude of Y')
subplot(3,2,4); plot(w/pi,angle(Y)/pi); grid;
xlabel('frequency in pi units'); ylabel('radians/pi')
title('Angle of Y')
subplot(3,2,5); plot(w/pi,abs(Y_check)); grid;
xlabel('frequency in pi units'); ylabel('|Y|')
title('Magnitude of Y check')
subplot(3,2,6); plot(w/pi,angle(Y_check)/pi); grid;
xlabel('frequency in pi units'); ylabel('radians/pi')
title('Angle of Y_ check')
```

7. **Symmetries in real sequences**: We have already studied the conjugate symmetry of real sequences. These real sequences can be decomposed into their even and odd parts.

$$x(n) = x_e(n) + x_o(n)$$

Then,

$$x_e(n) = Re\left|X(e^{j\omega})\right|$$

$$x_o(n) = jIm\left|X(e^{j\omega})\right|$$

Ex. We verify the symmetry property of real signals. Let

$$x(n) = sin(\pi n/2), -5 \le n \le 10$$

```
n = -5:10; x = sin(pi*n/2);
k = -100:100; w = (pi/100)*k;        % frequency between -pi and +pi
X = x * (exp(-1j*pi/100)).^(n'*k);    % DTFT of x
% signal decomposition
[xe,xo,m] = evenodd(x,n); % even and odd parts
XE = xe * (exp(-1j*pi/100)).^(m'*k); % DTFT of xe
XO = xo * (exp(-1j*pi/100)).^(m'*k); % DTFT of xo
% verification
XR = real(X); % real part of X
error1 = max(abs(XE-XR)); % Difference
XI = imag(X); % imag part of X
error2 = max(abs(XO-1j*XI)); % Difference
% graphical verification
subplot(2,2,1); plot(w/pi,XR); grid; axis([-1,1,-2,2])
xlabel('frequency in pi units'); ylabel('Re(X)');
title('Real part of X')
subplot(2,2,2); plot(w/pi,XI); grid; axis([-1,1,-10,10])
xlabel('frequency in pi units'); ylabel('Im(X)');
title('Imaginary part of X')
subplot(2,2,3); plot(w/pi,real(XE)); grid; axis([-1,1,-2,2])
xlabel('frequency in pi units'); ylabel('XE');
title('Transform of even part')
subplot(2,2,4); plot(w/pi,imag(XO)); grid; axis([-1,1,-10,10])
xlabel('frequency in pi units'); ylabel('XO');
title('Transform of odd part')
```

```
function [xe, xo, m] = evenodd(x,n)
% Real signal decomposition into even and odd parts
% -------------------------------------------------
% [xe, xo, m] = evenodd(x,n)
if any(imag(x) ~= 0)
    error('x is not a real sequence')
end

m = -fliplr(n);
m1 = min([m,n]); m2 = max([m,n]); m = m1:m2;
nm = n(1)-m(1); n1 = 1:length(n);
x1 = zeros(1,length(m)); x1(n1+nm) = x; x = x1;
xe = 0.5*(x + fliplr(x)); xo = 0.5*(x - fliplr(x));
```

MATLAB Exercise: Plot the real, imaginary, magnitude, and phase of the DTFT of the following signal:

1. $x(n) = \{\underset{\uparrow}{4}, 3, 2, 1, 1, 2, 3, 4\}$. Comment on the angle plot.
2. $x(n) = [cos(0.5\pi n) + j\ sin(0.5\pi n)][u(n) - u(n - 51)]$. Comment on the magnitude plot.
3. Solve analytically for $x(n) = 2\ (0.5)^3\ u(n + 2)$.