Lab Manual 07

Z transform and Inverse z transform

Lab Objectives:

- To manipulate the z, transform and inverse z transform practically in the MATLAB

The Z-transform is an essential tool in digital signal processing (DSP) for analyzing and processing discrete-time signals. It is a mathematical transformation that maps a discrete-time signal in the time domain to a complex-valued function in the frequency domain.

The Z-transform of a discrete-time signal x(n) is defined as the infinite sum of the signal values weighted by powers of a complex variable z. X(z)=∑∞n=−∞x(n)z−n

Computing the Z-transform in MATLAB

MATLAB provides a convenient way to compute the Z-transform of a discrete-time signal using the ztrans function. The syntax of the function is as follows:

F = ztrans(f, n, z)

where f is the symbolic expression of the discrete-time signal, n is the symbolic variable representing the time index, and z is the symbolic variable representing the complex frequency variable. The output F is the symbolic expression of the Z-transform of the signal.

For example, let's consider discrete sine signal, then the following code:

syms n z

f = sin(n);

F = ztrans(f, n, z);

Here, we define a symbolic variable n to represent the time index and a symbolic variable z to represent the complex frequency variable. We then define the signal f as sin(n). Finally, we use the ztrans function to compute the Z-transform of the signal and store it in the variable F. syms is the toolbox of MATLAB when gives us the value in the variables form.

Plotting the Z-transform in MATLAB

Once we have computed the Z-transform of a signal, we can plot its magnitude and phase as a function of the normalized frequency using MATLAB's plotting functions. In this example, we will use the fplot function to plot the magnitude and phase of the Z-transform of the signal.

To plot the magnitude of the Z-transform, we can use the following code:

fplot(abs(F), [-pi, pi]);

title('Magnitude of F(z)');

xlabel('Frequency (radians)');

ylabel('|F(z)|');

Here, we use the fplot function to plot the absolute value (abs) of the Z-transform F over the frequency range from -π to π. We then set the title and axis labels of the plot.

Similarly, to plot the phase of the Z-transform, we can use the following code:

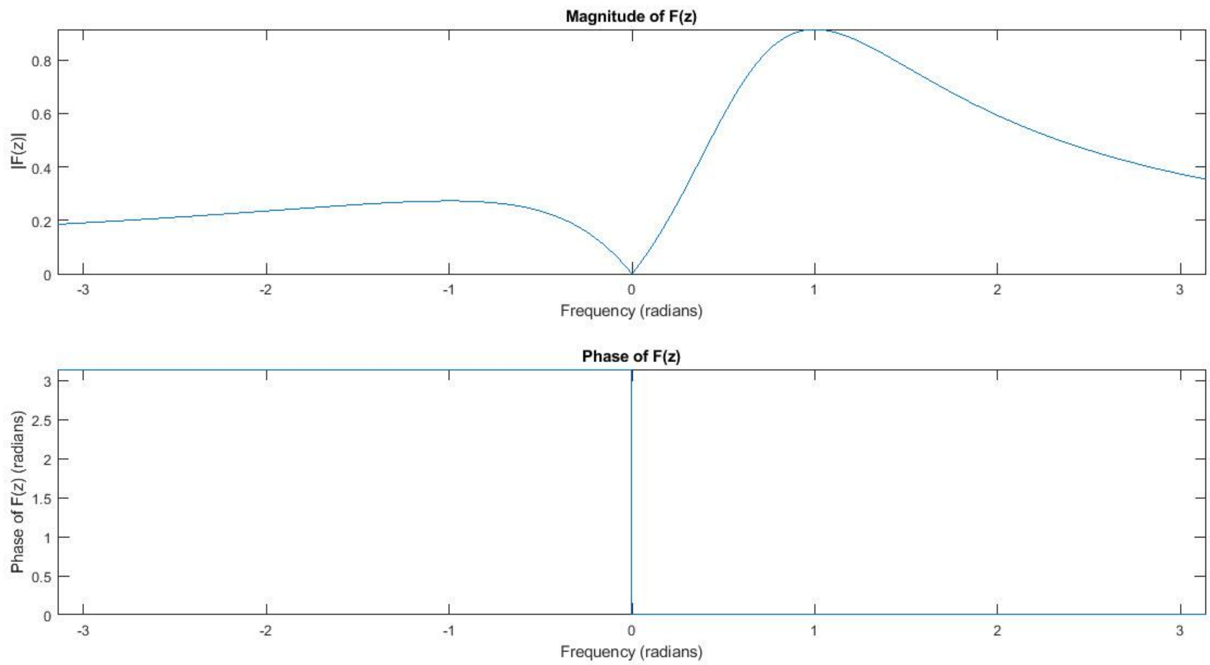
```
fplot(angle(F), [-pi, pi]);

title('Phase of F(z)');

xlabel('Frequency (radians)');

ylabel('Phase of F(z) (radians)');
```

Here, we use the fplot function to plot the phase (angle) of the Z-transform F over the same frequency range. We then set the title and axis labels of the plot.

Complete MATLAB Z-Transform code

```
clear

syms n z

f = sin(n);

F = ztrans(f, n, z);

% Plot the magnitude of F(z)

fplot(abs(F), [-pi, pi]);

title('Magnitude of F(z)');

xlabel('Frequency (radians)');

ylabel('|F(z)|');

% Plot the phase of F(z)

fplot(angle(F), [-pi, pi]);

title('Phase of F(z)');

xlabel('Frequency (radians)');

ylabel('Phase of F(z) (radians)');
```

MATLAB Z-Transform Plot



Solving equations with the z transform:

Here is a simple example of an equation that also shows some little details.

Code:

```
syms n;

f=(2*n+5)/3

disp('x[n]=')

disp(f)

F=ztrans(f)

disp('z[n]')

disp(F)
```

Output:

```
f =

(2*n)/3 + 5/3

x[n]=
(2*n)/3 + 5/3

z[n]
(5*z)/(3*(z - 1)) + (2*z)/(3*(z - 1)^2)
```

Similarly, for the inverse z transform, we need to use the MATLAB function of inverse z transform. To apply the inverse z transform in MATLAB, we use the following formula:

iztrans(x)

Lab tasks:

1. Write MATLAB code for the z transform of function 0.5^n?
2. Write MATLAB code for the z transform of function sin(n)?
3. Write MATLAB code for the inverse z transform of function $2*z/(z-2)^2$?
4. Find z transform of following:
    - a^n
    - $n*a^2$
    - n*((0.5)^n*cos9pi*n/3))
    - (1/2)^n*u[n]-(1/3)^n*u[n]
5. Find inverse z transform of (2*z)/(2*z-1)
6. Find inverse z transform of $(2*z)/(z-1)(z^2+1)$