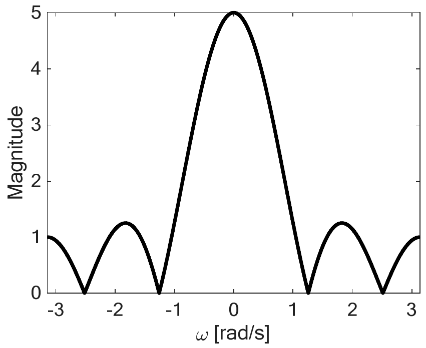
42. Fourier Transform for Discrete-Time Signals

Prof. Mohammed Hawa
Electrical Engineering
The University of Jordan



Discrete-time Fourier transform (DTFT)

Computers can only work with discrete-time signals. The Fourier transform for a discrete-time signal $x[n]$ (whether periodic or aperiodic), which shows the **frequency content** of that signal is,

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

Notice similarity to Fourier transform of continuous-time signal $x(t)$,

$$X(\omega) = \mathcal{F}\{x(t)\} = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt$$

The **inverse discrete-time Fourier transform (IDTFT)** is,

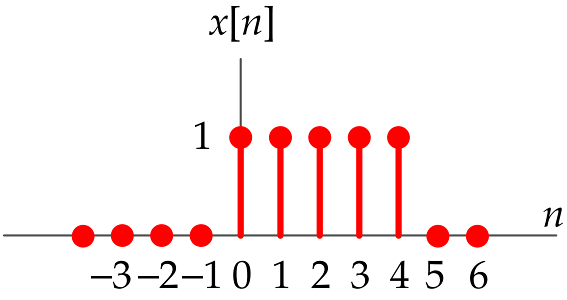
$$x[n] = \frac{1}{2\pi}\int_{2\pi} X(e^{j\omega})e^{jn\omega}\,d\omega$$

Notice the similarity to inverse Fourier transform of continuous-time signal $x(t)$,

$$x(t) = \mathcal{F}^{-1}\{X(\omega)\} = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(\omega)e^{j\omega t}\,d\omega$$

Notice that time-domain is discrete-time, while frequency-domain is continuous-frequency.

Q1. For the following discrete-time signal $x[n] = \left\{\text{rect}\left(\frac{n-2}{4}\right)\right\}$, determine the discrete-time Fourier transform (DTFT) $X(e^{j\omega})$. Sketch the **magnitude** $|X(e^{j\omega})|$ and **phase** $\angle X(e^{j\omega})$ spectrum density.



Q1. Solution.

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

$$X(e^{j\omega}) = \sum_{n=0}^{4} 1\, e^{-jn\omega} = \sum_{n=0}^{N-1} \left(e^{-j\omega}\right)^{n} = \frac{1-\left(e^{-j\omega}\right)^{N}}{1-e^{-j\omega}}$$

Since the geometric series for N terms is,

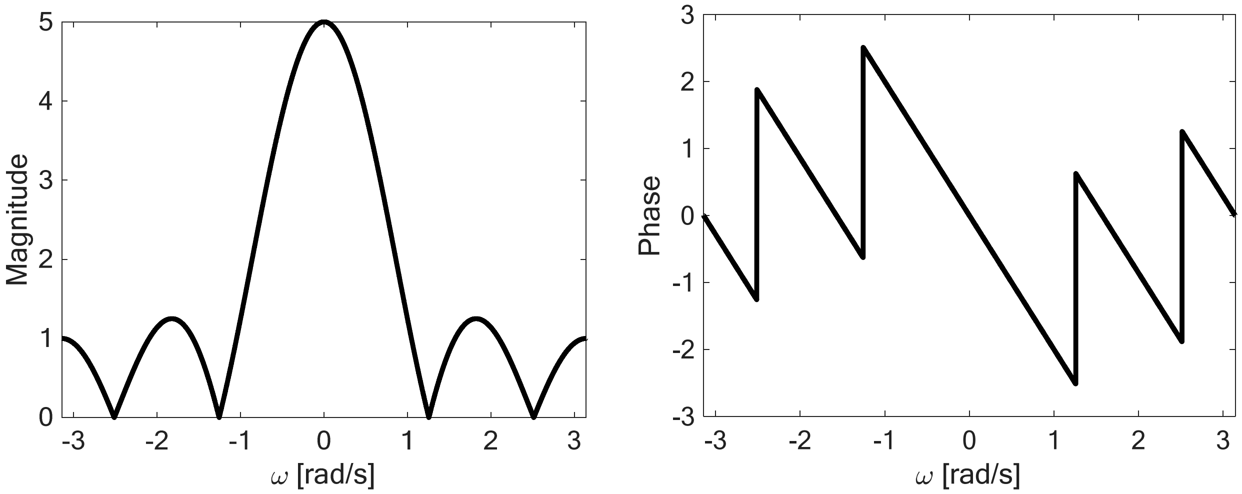
$$\sum_{n=0}^{N-1} a^{n} = \frac{1-a^{N}}{1-a}$$

$$X(e^{j\omega}) = \frac{1-\left(e^{-j\omega}\right)^{N}}{1-e^{-j\omega}} = \frac{1-e^{-j\omega N}}{1-e^{-j\omega}} = \frac{e^{-j\omega N/2}\left(e^{j\omega N/2}-e^{-j\omega N/2}\right)}{e^{-j\omega/2}\left(e^{j\omega/2}-e^{-j\omega/2}\right)}$$

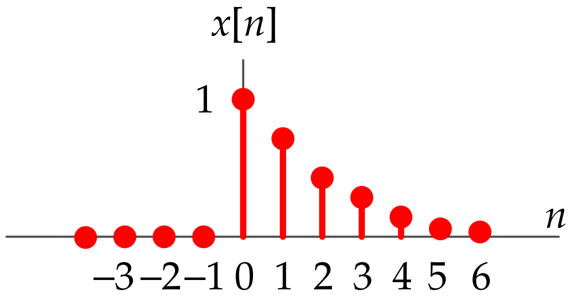
$$X(e^{j\omega}) = \frac{e^{-j\omega N/2}(2j\sin(\omega N/2))}{e^{-j\omega/2}(2j\sin(\omega/2))}$$

$$X(e^{j\omega}) = \frac{\sin(\omega N/2)}{\sin(\omega/2)}\, e^{-j\omega\left(\frac{N-1}{2}\right)} = \frac{\sin(2.5\omega)}{\sin(0.5\omega)}\, e^{-j2\omega}$$

Sketching **magnitude** $\left|X(e^{j\omega})\right|$ and **phase** $\angle X(e^{j\omega})$ from $-\pi$ to π, you will notice the familiar sinc() function in magnitude spectrum density.



Q2. For the following discrete-time signal $x[n] = \{0.7^n u[n]\}$, determine the discrete-time Fourier transform (DTFT) $X(e^{j\omega})$. Sketch the **magnitude** $|X(e^{j\omega})|$ and **phase** $\angle X(e^{j\omega})$ spectrum density.



Q2. Answer. $X(e^{j\omega}) = \frac{1}{1-0.7e^{-j\omega}}$

Discrete Fourier transform (DFT)

For DTFT, time-domain signal is discrete (say N samples), while frequency-domain is continuous,

$$X\left(e^{j\omega}\right) = \sum_{n=0}^{N-1} x[n]\, e^{-jn\omega}$$

In DFT, frequency-domain ω is also discrete (to be processed by computers) at sampling frequency,

$$\omega_0 = \frac{2\pi}{N}$$

And hence evaluated at frequencies $\omega_k = k\omega_0 = k\frac{2\pi}{N}, k = 0, 1, \dots, N-1$

So, for DTFT,

$$X\left(e^{j\omega}\right) = \sum_{n=0}^{N-1} x[n]\, e^{-jn\omega}$$

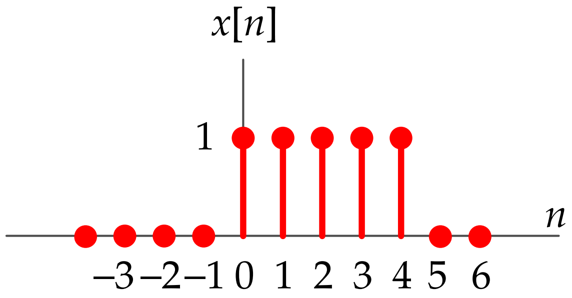
And for DFT,

$$X[k] = X\left(e^{jk\frac{2\pi}{N}}\right) = \sum_{n=0}^{N-1} x[n]\, e^{-jn\left(k\frac{2\pi}{N}\right)}$$

The inverse DFT (IDFT) is,

$$x[n] = \frac{1}{N}\sum_{k=0}^{N-1} X[k] e^{jn\left(k\frac{2\pi}{N}\right)}$$

Q3. For the following discrete-time signal $x[n] = \left\{\text{rect}\left(\frac{n-2}{4}\right)\right\}$, determine the discrete Fourier transform (DFT) $X[k]$.



Q3. Solution.

$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{-jn\left(k\frac{2\pi}{N}\right)} = \sum_{n=0}^{5-1} 1\, e^{-jn\left(k\frac{2\pi}{5}\right)} = \sum_{n=0}^{4} 1\, e^{-jnk\frac{\pi}{2.5}}$$

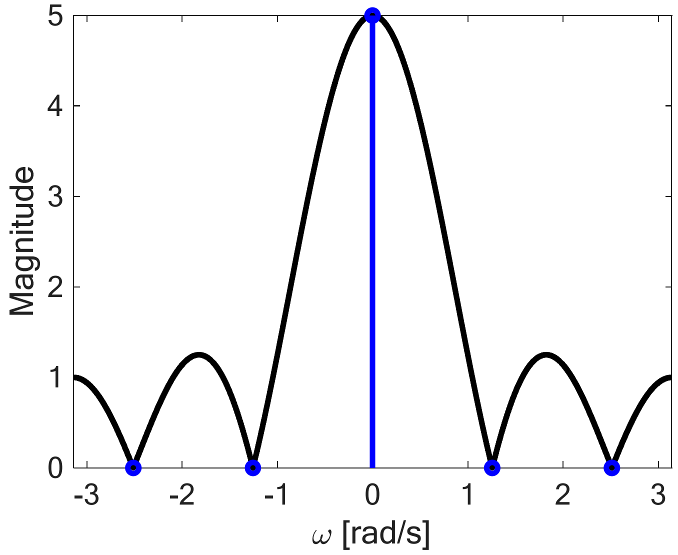
$$X[k] = 1 + e^{-jk\frac{\pi}{2.5}} + e^{-jk\frac{2\pi}{2.5}} + e^{-jk\frac{3\pi}{2.5}} + e^{-jk\frac{4\pi}{2.5}}$$

$$X[0] = 1 + 1 + 1 + 1 + 1 = 5$$

$$X[1] = 1 + e^{-j\frac{\pi}{2.5}} + e^{-j\frac{2\pi}{2.5}} + e^{-j\frac{3\pi}{2.5}} + e^{-j\frac{4\pi}{2.5}} = 0$$

$$X[2] = 1 + e^{-j\frac{2\pi}{2.5}} + e^{-j\frac{4\pi}{2.5}} + e^{-j\frac{6\pi}{2.5}} + e^{-j\frac{8\pi}{2.5}} = 0$$

$$\ldots$$



Fast Fourier Transform (FFT)

Computing DFT requires excessive number of computations, around $\mathcal{O}(N^2)$. Notice that we have N equations, each requires N complex number multiplications, plus extra additions.

$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{-jn\left(k\frac{2\pi}{N}\right)}, \quad = 0,1,2,\dots,N-1$$

$$X[k] = x[0]\, e^{-j0} + x[1]\, e^{-j\left(k\frac{2\pi}{N}\right)} + \cdots + x[N-1]\, e^{-j(N-1)\left(k\frac{2\pi}{N}\right)}$$

FFT is an efficient computer algorithm that reduces the number of computations required for DFT to around $\mathcal{O}(N\log_2 N)$.

FFT utilizes linearity of the Fourier transform (and DFT), to compute the Fourier transform of a signal $x[n]$ as a sum of the Fourier transforms of segments of $x[n]$ of shorter duration.

FFT is a very important algorithm for modern digital communications and digital signal processing.

Computers performing FFT (i.e., efficiently evaluating DFT) enable important modern applications, such as spectral analysis, de-noising of data, audio compression, image and video compression, digital filtering, fast convolution, approximate derivatives, solve PDEs, etc.

FFT is discussed in more advanced courses. Invented by Cooley and Tukey. More specialized algorithms were developed afterwards.

Q4. Compare the complexity of DFT versus FFT for compressing a 20 second audio signal, sampled at 44.1 kHz.

Q4. Solution.

An audio signal sampled at 44.1 kHz (i.e., discrete-time) has 44100 samples every second, or $N = 4.41 \times 10^4 \times 20 = 8.82 \times 10^5$ samples.

To compress the signal using native DFT, we need roughly $N^2 = (8.82 \times 10^5)^2 \approx 0.78 \times 10^{12}$ (thousand billion) complex number multiplications.

To compress the signal using FFT, we need roughly $N \log_2 N = 8.82 \times 10^5 \times \log_2(8.82 \times 10^5) = 8.82 \times 10^5 \times 19.7504 = 1.74 \times 10^7$ (twenty million) complex number multiplications.