Introduction

- The Discrete Fourier Transform (DFT) is a work horse of Digital Signal Processing.
- Its primary uses include:
  - Measuring the spectrum of a signal from samples
  - Fast algorithms for convolution or correlation
- The DFT is computed from a block of N samples $x[0], \ldots, x[N-1]$.
- It computes the DTFT at N evenly spaced, discrete frequencies:

$$X[k] = X(e^{j2\pi \cdot k/N \cdot n}) \quad \text{for } k = 0, \ldots, N-1$$

- Fast algorithms (Fast Fourier Transform (FFT)) exist to compute the DFT.

Definitions

- **(Forward) Discrete Fourier transform:** for a block of N samples $x[n]$, the DFT $X[k]$ is given by

$$X[k] = \sum_{n=0}^{N-1} x[n] \cdot \exp(-j2\pi \cdot k/N \cdot n) \quad \text{for } k = 0, \ldots, N-1$$

- **Inverse Discrete Fourier transform:** a block of N samples $x[n]$, is obtained from the DFT $X[k]$ by

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] \cdot \exp(j2\pi \cdot k/N \cdot n) \quad \text{for } n = 0, \ldots, N-1$$

Observations

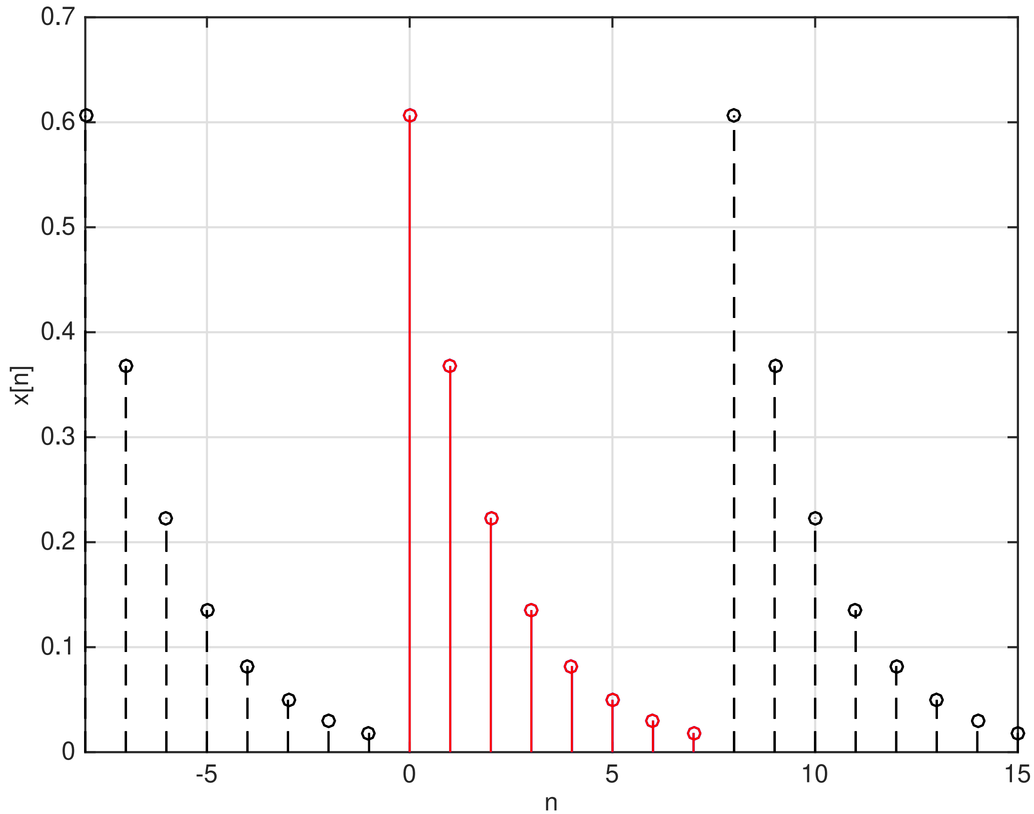
- The DFT is *discrete* in **both** time and frequency.
  - In contrast, the DTFT is discrete in time but continuous in frequency.
- The signal $x[n]$ is implicitly assumed to repeat periodically with period N.

$$x[n+N] = \frac{1}{N}\sum_{k=0}^{N-1} X[k]\cdot \exp(j2\pi \cdot k/N \cdot (n+N))$$

$$= \frac{1}{N}\sum_{k=0}^{N-1} X[k]\cdot \exp(j2\pi \cdot k/N \cdot n)\cdot \exp(j2\pi \cdot k) = x[n]$$

  - This observation has ramifications for the delay and convolution properties of the DFT.

Implicit Periodicity



The signal with DFT $X[k]$ is implicitly periodic; the period equals the block length N.

Example

- The DFT[1] of the length $N = 4$ signal $\{1, 1, 0, 0\}$:

$$\begin{aligned}
X[0] &= 1e^{-j0} + 1e^{-j0} + 0e^{-j0} + 0e^{-j0} \\
&= 1 + 1 + 0 + 0 = 2 \\
X[1] &= 1e^{-j0} + 1e^{-j2\pi/4} + 0e^{-j4\pi/4} + 0e^{-j6\pi/4} \\
&= 1 + (-j) + 0 + 0 = \sqrt{2}e^{-j\pi/4} \\
X[2] &= 1e^{-j0} + 1e^{-j4\pi/4} + 0e^{-j8\pi/4} + 0e^{-j12\pi/4} \\
&= 1 + (-1) + 0 + 0 = 0 \\
X[3] &= 1e^{-j0} + 1e^{-j6\pi/4} + 0e^{-j12\pi/4} + 0e^{-j18\pi/4} \\
&= 1 + (j) + 0 + 0 = \sqrt{2}e^{j\pi/4}
\end{aligned}$$

Thus, $X[k] = \{2, \sqrt{2}e^{-j\pi/4}, 0, \sqrt{2}e^{j\pi/4}\}$

[1] Exponentials are $e^{-j2kn\pi/N}$

Fast Transform (FFT)

- The main practical benefit of the DFT stems from the fact that a computationally efficient algorithm exists.
- A naive (brute-force) implementation of the DFT requires N^2 complex multioplications and additions.
  - N outputs must be computed
  - Each requires N multiplications and additions
- The Fast Fourier Transform algorithm (FFT) reduces the number of complex multiplications and additions to $N \cdot \log_2(N)$.
  - It recursively splits the DFT of length N into 2 DFTs of length $N/2$ (divide-and-conquer)
  - Until length-2 DFTs can be computed trivially.
- A naive DFT of length $N = 1024$ requires approximately 10^6 multiplications and additions; the FFT requires only approximately 10^4.

DFT of a Shifted Impulse

- The finite, length N duration of the signal block and the associated, implicit assumption that $x[n]$ is periodic with period N has some unexpected consequences.
- We showed that the DTFT of a shifted impulse is

$$\delta[n-n_d] \overset{\text{DTFT}}{\longleftrightarrow} e^{-j2\pi f_d n_d}$$

- **DFT with shift $n_d < N$:** assume $N = 8$ and $n_d = 3$

$$X[k] = e^{-j2\pi k/N n_d} = e^{-j3\pi/4k}$$

- **DFT with shift $n_d \geq N$:** assume $N = 8$ and $n_d = 11$

$$X[k] = e^{-j2\pi k/N n_d} = e^{-j11\pi/4k} = e^{-j3\pi/4k} \cdot e{-j2\pi} = e^{-j3\pi/4k}$$

  - Delays induce phase shifts proportional to n_d mod N:

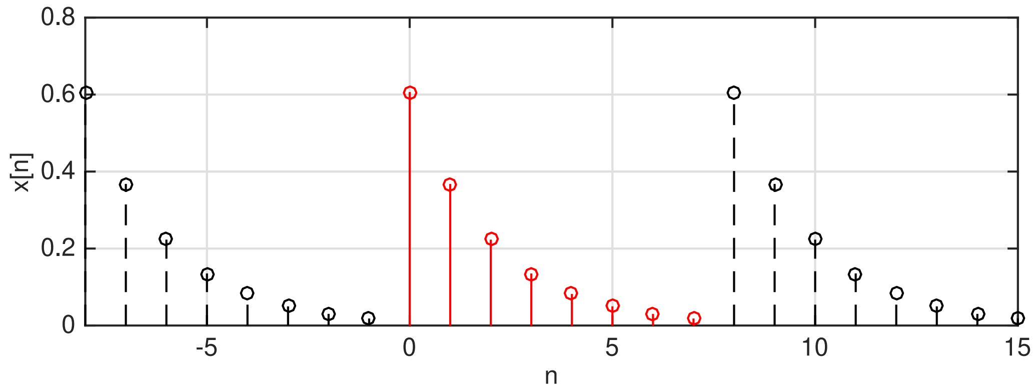
$$X[k] = e^{-j2\pi k/N n_d} = e^{-j2\pi k/N(n_d \text{ mod } N)}$$

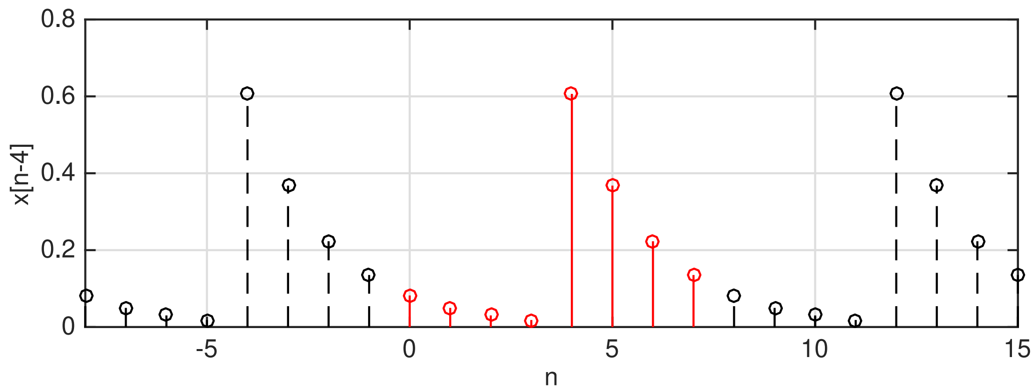
Delay Property

- The same phenomenon affects the delay property.
  - When the implicitly periodic signal is delayed, the block of N samples is filled with periodic samples.
  - For example, when the signal $x[n] = \{1, 2, 3, 4\}$ is shifted by $n_d = 2$ positions it becomes $x[(n - n_d) \bmod N] = \{3, 4, 1, 2\}$.
  - This is refered to as circular shifting.
- For the DFT, the delay property is therefore

$$x[(n - n_d) \bmod N] \overset{\text{DFT}}{\longleftrightarrow} X[k] \cdot e^{-j2\pi k/N n_d}$$

Implicit Periodicity





Shifting the implicitly periodic signal induces a circular shift over the block of N samples.

Convolution Property

- Similarly, the convolution property for the DFT is different from that for the DTFT or z-Transform.
- A modified form of convolution, called circular convolution has a product-form transform.
  - Let $x[n]$ and $h[n]$ be length-N signals with DFT $X[k]$ and $H[k]$, respectively.
  - Then, the (circular) convolution property is

$$\sum_{m=0}^{N-1} h[m]x[(n-m) \bmod N] \stackrel{\text{DFT}}{\longleftrightarrow} X[k] \cdot H[k]$$

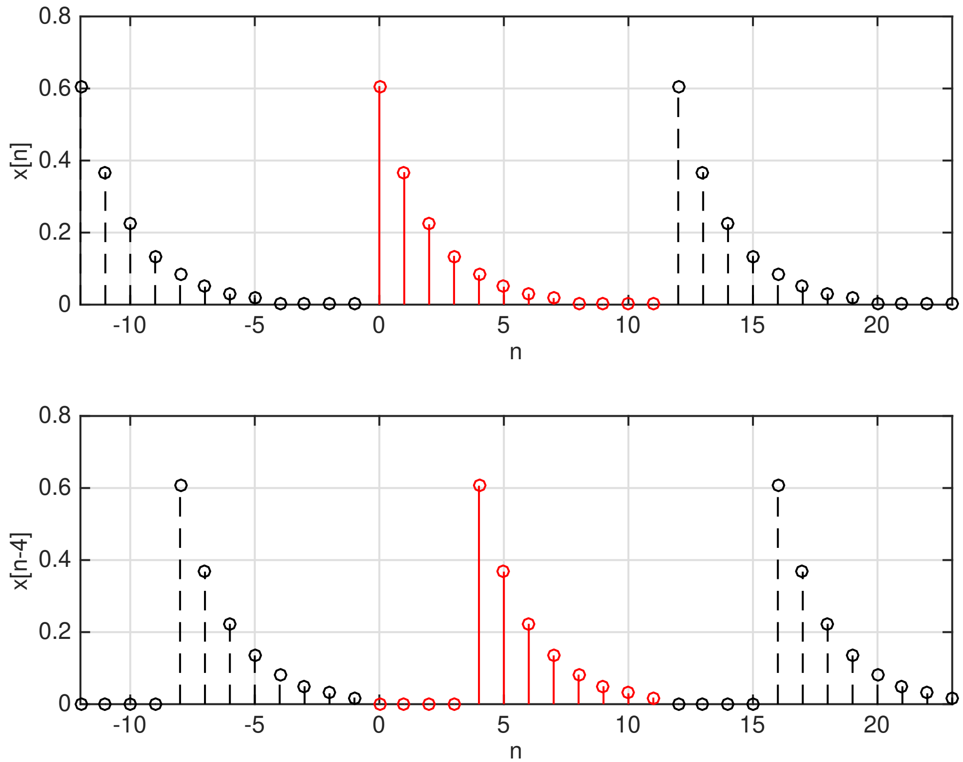
  - Note that circular convolution is very different from normal convolution.
- **Question:** How can the (circular) convolution property be used for fast convolution?

Zero-Padding

- Turning circular convolution into regular convolution is straightforward:
  - The signals $x[n]$ and $h[n]$ to be convolved must be extended by appending zeros such that
    - They have the same length N, and
    - if $x[n]$ has length N_x and $h[n]$ has length N_h, then $N \geq N_x + N_h - 1$.
  - This is called zero-padding.
- **Example:** Let $x[n] = \{1, 2, 3, 4\}$ and $h[n] = \{3, 2, 1\}$, then the zero-padded signals are

$$\tilde{x}[n] = \{1, 2, 3, 4, 0, 0\} \quad \tilde{x}[n] = \{3, 2, 1, 0, 0, 0\}$$

Implicit Periodicity



With zero-padding, the shifting of the implicitly periodic signal introduces only zero samples in the block of N samples.

Convolution with FFTs

- Fast convolution based on FFTs of zero-padded signals can be implemented as follows:

```
% signals
x = [1,2,3];
h = [1,1];

% zero-padding to length 4
xp = [x, 0];
hp = [h, 0, 0];

% transforms
Xp = fft(xp);
Hp = fft(hp);

% multiply and inverse transform
y = ifft(Xp.*Hp)
```