The Discrete Fourier Transform (DFT) and Its Properties

Recall the DFT and IDFT:

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N}, \qquad k = 0, 1, 2, \ldots, N-1$$

$$x(n) = \sum_{n=0}^{N-1} X(k) e^{j2\pi kn/N}, \qquad n = 0, 1, 2, \ldots, N-1$$

or,

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}, \qquad k = 0, 1, 2, \ldots, N-1$$

$$x(n) = \sum_{n=0}^{N-1} X(k) W_N^{-kn}, \qquad n = 0, 1, 2, \ldots, N-1$$

where $W_N = e^{-j2\pi/N}$ and $W_N^{kn} = e^{-j2\pi kn/N}$.

We use the notation: $x(n) \underset{N}{\overset{DFT}{\leftrightarrow}} X(k)$

The DFT is really the transform of a periodic extension of $x(n)$:

$$\begin{array}{lll} if & x(n+N) = x(n) & \forall n \\ then & X(k+N) = X(k) & \forall k \end{array}$$

The DFT obeys the principles of linearity and superposition:

$$if: \quad x_1(n) \underset{N}{\overset{DFT}{\leftrightarrow}} X_1(k) \quad and \quad x_2(n) \underset{N}{\overset{DFT}{\leftrightarrow}} X_2(k)$$

$$then: \quad a_1 x_1(n) + a_2 x_2(n) \underset{N}{\overset{DFT}{\leftrightarrow}} a_1 X_1(k) + a_2 X_2(k)$$

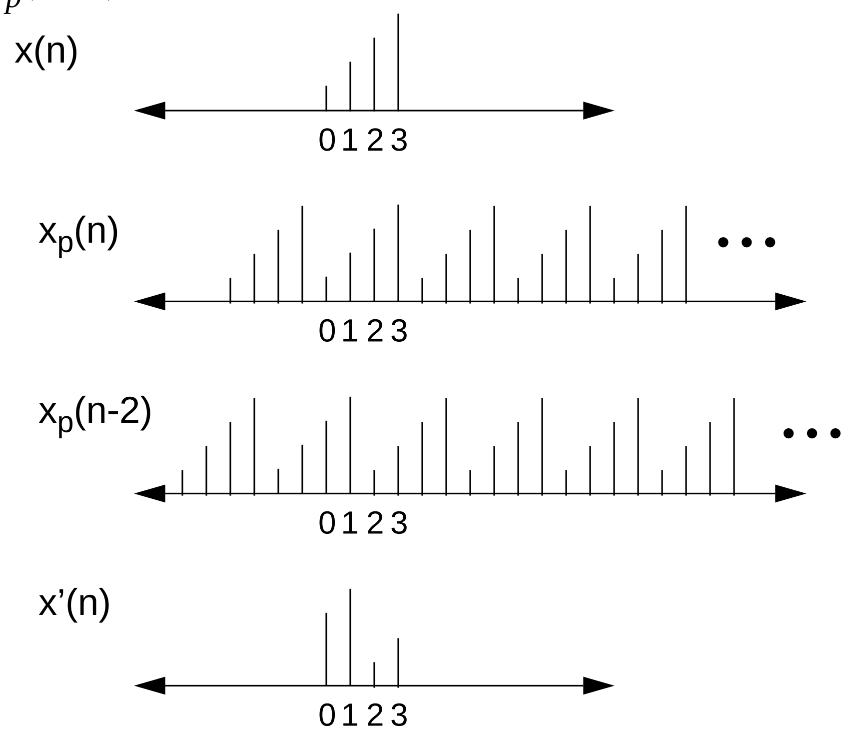
What are the implications of these properties?

Symmetry Properties

Recall our expression for the periodic extension of $x(n)$:

$$x_p(n) = \sum_{l=-\infty}^{\infty} x(n - lN)$$

What is $x_p(n-k)$?



We define this process as a circular shift: $\begin{aligned} x'(n) &= (x(n-k), modulo\ N) \\ &\equiv x((n-k))_N \end{aligned}$.

What should the impact of this operation be on the DFT?

Several interesting properties result:

$$(x((-n))_N = x(N-n)), \qquad 0 \le n \le N-1$$

and:

$$\text{conjugate even:} \qquad x_p(n) = x_p^*(N-n)$$

$$\text{conjugate odd:} \qquad x_p(n) = -x_p^*(N-n)$$

This symmetry gives rise to general properties of the DFT:

N-Point Sequence	N-Point DFT
$x(n)$	$X(k)$
$x^*(n)$	$X^*(N-k)$
$x^*(N-n)$	$X^*(k)$
$x_R(n)$	$X_{ce}(k) = \frac{1}{2}[X(k) + X^*(N-k)]$
$jx_I(n)$	$X_{co}(k) = \frac{1}{2}[X(k) - X^*(N-k)]$
$x_{ce}(n) = \frac{1}{2}[x(n) + x^*(N-n)]$	$X_R(k)$
$x_{ce}(n) = \frac{1}{2}[x(n) + x^*(N-n)]$	$jX_I(k)$
Real Signals	
Any real signal $x(n)$	$X(k) = X^*(N-k)$ $X_R(k) = X_R(N-k)$ $X_I(k) = -X_I(N-k)$ $\lvert X(k)\rvert = \lvert X(N-k)\rvert$ $\angle X(k) = -\angle X(N-k)$
$x_{ce}(n) = \frac{1}{2}[x(n) + x(N-n)]$	$X_R(k)$
$x_{ce}(n) = \frac{1}{2}[x(n) - x^*(N-n)]$	$jX_I(k)$

Circular Convolution (Finally - Something Different!)

Suppose we have two finite duration sequences $x_1(n)$ and $x_2(n)$ of length N.

Suppose we multiply the two DFTs together: $X_3(k) = X_1(k)X_2(k)$.

What happens when I take the IDFT of $X_3(k)$?

$$x_3(m) = \frac{1}{N}\sum_{k=0}^{N-1} X_3(k)e^{j2\pi km/N}$$

$$= \frac{1}{N}\sum_{k=0}^{N-1} X_1(k)X_2(k)e^{j2\pi km/N}$$

$$= \frac{1}{N}\sum_{k=0}^{N-1}\left[\sum_{n=0}^{N-1} X_1(k)e^{-j2\pi kn/N}\right]\left[\sum_{k=0}^{N-1} X_2(k)e^{-j2\pi kn/N}\right]e^{j2\pi km/N}$$

$$\ldots$$

$$= \sum_{n=0}^{N-1} x_1(n)x_2((m-n))_N, \qquad m = 0, 1, \ldots, N-1$$

This is known as circular convolution, and has a simple graphical interpretation (see Figure 9.2):

$$x_1(n) = \{2, 1, 2, 1\}$$

$$x_2(n) = \{1, 2, 3, 4\}$$

$$x_3(0) = \begin{Bmatrix} 2, 1, 2, 1 \\ 1, 4, 3, 2 \end{Bmatrix} = 14$$

$$x_3(1) = \begin{Bmatrix} 2, 1, 2, 1 \\ 2, 1, 4, 3 \end{Bmatrix} = 16$$

$$x_3(2) = \begin{Bmatrix} 2, 1, 2, 1 \\ 3, 2, 1, 4 \end{Bmatrix} = 14$$

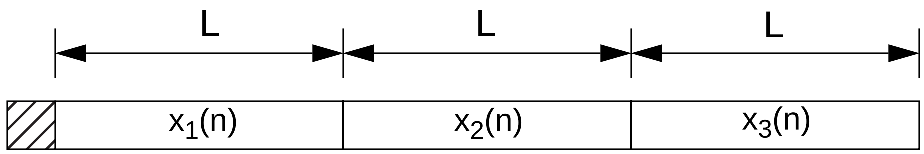
$$x_3(3) = \begin{Bmatrix} 2, 1, 2, 1 \\ 4, 3, 2, 1 \end{Bmatrix} = 16$$

Additional Properties of the DFT

Property	Time Domain	Frequency Domain
Notation	$x(n), y(n)$	$X(k), Y(k)$
Periodicity	$x(n) = x(n+N)$	$X(k) = X(k+N)$
Linearity	$a_1 x_1(n) + a_2 x_2(n)$	$a_1 X_1(k) + a_2 X_2(k)$
Time Reversal	$x(N-n)$	$X(N-k)$
Circular Time Shift	$x((n-l))_N$	$X(k)e^{-j2\pi kl/N}$
Circular Frequency Shift	$x(n)e^{j2\pi nl/N}$	$X((k-l))_N$
Complex Conjugate	$x^*(n)$	$X^*(N-k)$
Circular Convolution	$x_1(n) \oplus x_2(n)$	$X_1(k)X_2(k)$
Circular Correlation	$x(n) \oplus y^*(-n)$	$X(k)Y^*(k)$
Multiplication	$x_1(n)x_2(n)$	$\frac{1}{N}X_1(k) \oplus X_2(k)$
Parseval's Theorem	$\sum_{n=0}^{N-1} x(n)y^*(n)$	$\frac{1}{N}\sum_{n=0}^{N-1} X(k)Y^*(k)$

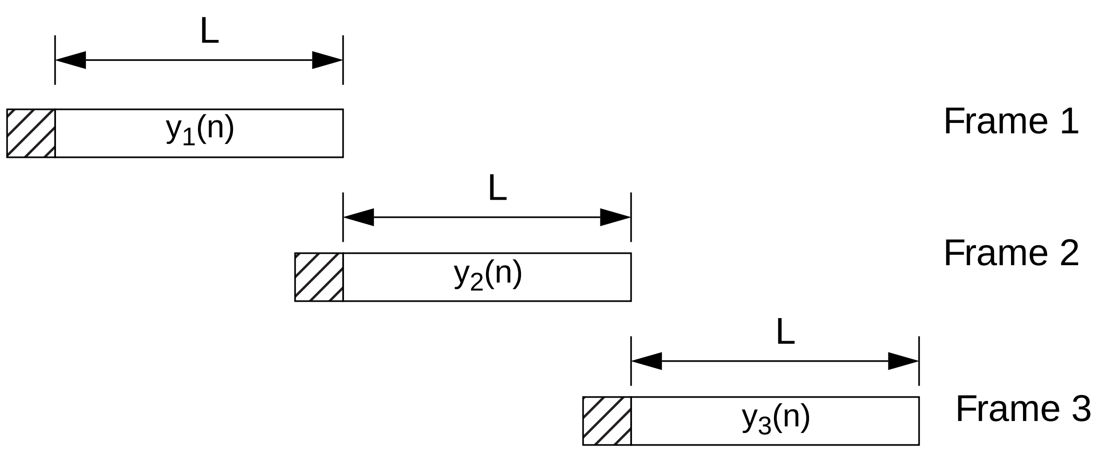
Using The DFT For Linear Filtering - Overlap-Save

Consider the problem of performing linear filtering in the frequency domain by processing signals a frame at a time:



In this example, we desire to filter the signal in the frequency domain using an M point filter where $L \gg M$. Out strategy is to compute a DFT of the signal, multiply this DFT by the pre-computed DFT of the filter, and compute an inverse DFT to obtain the output signal.

In the **Overlap and Save** method, we use $N = L + M - 1$ points in our DFT. First, we pad the filter impulse response to N points and take an N-point DFT. Next, we process the data as follows:

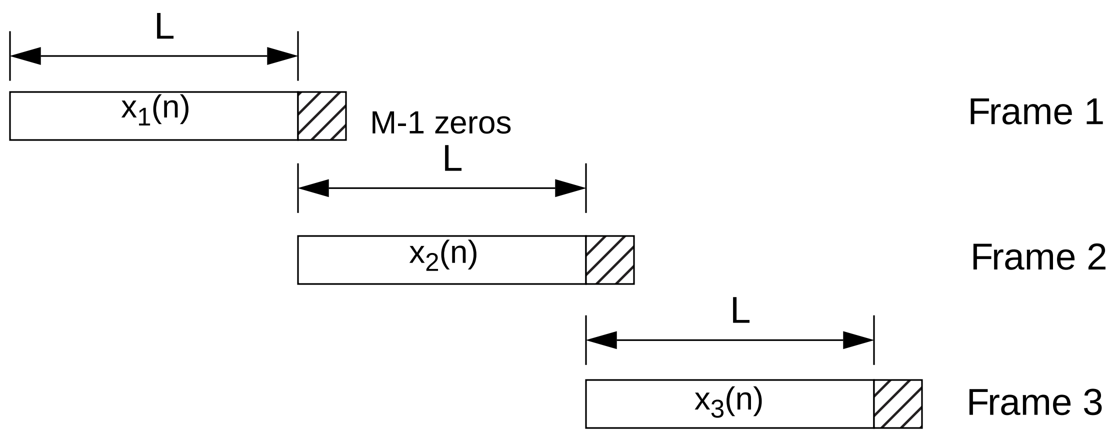


N points are processed each frame, using $M - 1$ points from the previous frame. The output signal is constructed by retaining the L last points.

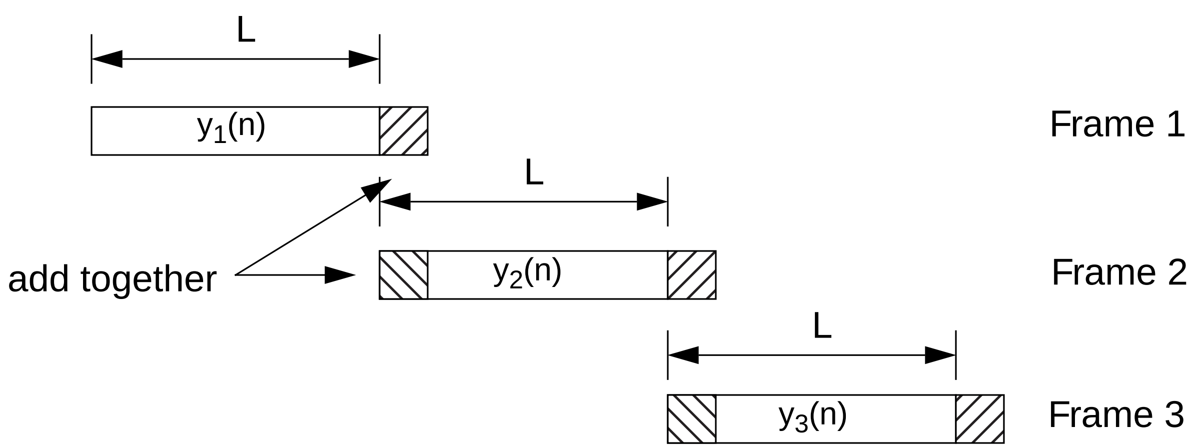
Note that L new data points are introduced each frame. What does this imply for the case $L \approx M$ when the signal is time-varying?

An Alternate Method — Overlap-Add

We can also zero-stuff the input signal:



The output signals for adjacent frames must be added:



The N-point IDFT produces data that extends into the next frame. The last $M-1$ points must be retained and added to the output for the next frame. This process adds delay to the system.

More exotic approaches can be used based on trapezoidal windows: