III. Discrete-Time Transforms

In this chapter, we present discrete-time transforms that can represent discrete-time sequences in another desirable domain. In particular, we describe the discrete Fourier transform, its fast version called fast Fourier transform, the discrete cosine transform and the z-transform. Each of these transforms is an important tool in digital signal processing which is useful in the processing signals and the design of systems

A. The Discrete-Time Fourier Transform

In this section, we develop the discrete Fourier transform (DFT) which is only applicable to finite-length sequences and provides a transform-domain representation of such sequences. We also describe the inverse DFT (IDFT), summarise the main properties of the DFT and the IDFT and study some applications.

The DFT is defined by

$$X[k] = \sum_{m=0}^{N-1} x[m]\, e^{-j\frac{2\pi k m}{N}} , \quad 0 \leq k \leq N-1 ,$$

where $x[m]$ is a finite-length sequence defined for $0 \leq m \leq N-1$, and $X[k]$ is the DFT of $x[m]$ with length $x[m]$.

Using the notation $W_N = e^{-j2\pi/N}$, we can rewrite the DFT as described by

$$X[k] = \sum_{m=0}^{N-1} x[m] \cdot W_N^{km} , \quad 0 \leq k \leq N-1 .$$

The IDFT is given by

$$x[m] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] \cdot W_N^{-km} , \quad 0 \leq m \leq N-1$$

Ex: Compute the N-point DFT of the length-N sequence.

$$x[n] = \cos(2\pi r n/N), \quad 0 \le n \le N-1, \quad 0 \le r \le N-1$$

Using a trigonometric identity, we can rewrite $x[n]$

$$x[n] = \frac{1}{2}\left(e^{j\frac{2\pi r n}{N}} + e^{-j\frac{2\pi r n}{N}}\right) = \frac{1}{2}\left(W_N^{-rn} + W_N^{rn}\right)$$

Substituting the above into $X[k] = \sum_{n=0}^{N-1} x[n] \cdot W_N^{kn}$, we obtain

$$X[k] = \frac{1}{2}\left[\sum_{n=0}^{N-1} W_N^{-(r-k)n} + \sum_{n=0}^{N-1} W_N^{(r+k)n}\right].$$

A convenient approach to processing signals is to express them into matrix form. Specifically, the DFT samples $X[k]$ can be expressed as

$$\underbrace{\underline{X}}_{N\times 1} = \underbrace{\underline{F}_N}_{N\times N} \cdot \underbrace{\underline{x}}_{N\times 1},$$

where $\underline{X} = [X[0]\ X[1] \ldots X[N-1]]^T$ is an $N\times 1$ vector of DFT samples, $\underline{x} = [x[0]\ x[1] \ldots x[N-1]]^T$ is an $N\times 1$ vector of input samples and $\underline{F}_N$ is the $N\times N$ DFT matrix given by

$$\underline{F}_N = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N^1 & W_N^2 & \cdots & W_N^{N-1} \\ 1 & W_N^2 & W_N^4 & \cdots & W_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & \cdots & W_N^{(N-1)\times(N-1)} \end{bmatrix}$$

Likewise, the IDFT relation can be expressed by

$$\underline{x} = \underline{F}_N^{-1} \underline{X},$$

where F_N^{-1} is the $N\times N$ IDFT matrix given by

$$\underline{F}_N^{-1} = \frac{1}{N}\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N^{-1} & W_N^{-2} & \cdots & W_N^{-(N-1)} \\ 1 & W_N^{-2} & W_N^{-4} & \cdots & W_N^{-2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{-(N-1)} & W_N^{-2(N-1)} & \cdots & W_N^{-(N-1)\times(N-1)} \end{bmatrix}$$

and it follows that $F_N^{-1} = \frac{1}{N} \cdot F_N^H$

Computational Complexity :

In terms of computational complexity, the DFT and the IDFT require N^2 complex multiplications and $N(N-1)$ complex additions. Elegant methods have been developed to reduce the computational complexity to about $N(\log_2 N)$ operations. These techniques are called fast fourier transform (FFT) algorithms. As a result of the availability of these fast algorithms, the DFT and IDFT are often used in digital signal processing applications.

Properties of the DFT:

In this part of the section, we describe important properties of the DFT. which can be used in several situations.

i) Linearity

Consider two finite-duration sequences $x_1[n]$ and $x_2[n]$ which are linearly combined. If their combination is

$$x_3[n] = a\, x_1[n] + b\, x_2[n],$$

then the DFT of $x_3[n]$ is

$$X_3[k] = a\, X_1[k] + b\, X_2[k]$$

In summary, the linearity property states that

$$a\, x_1[n] + b\, x_2[n] \overset{DFT}{\longleftrightarrow} a\, X_1[k] + b\, X_2[k]$$

ii) Circular Shift of a Sequence

The DFT of the circularly time-shifted sequence $x[n] = g[(n-n_0)_N]$ with n_0 being an integer is given by

$$X[k] = W_N^{kn_0}\, G[k]$$

where $(\cdot)_N$ denotes a modulo operation

This corresponds to

$$g[(n-n_0)_N] \xleftrightarrow{DFT} W_N^{kn_0} G[k]$$

The inverse DFT of the circularly-shifted DFT $X[k] = G[(k-k_0)_N]$ with k_0 being an integer and $(\ )_N$ denoting modulo N, is given by

$$x[n] = W_k^{-k_0 n} g[n].$$

This corresponds to

$$W_N^{-k_0 n} g[n] \xleftrightarrow{DFT} G[(k-k_0)_N]$$

iii) Duality

If the N-point DFT of the length-N sequence $g[n]$ is $G[k]$, then the N-point DFT of the length-N sequence $G[n]$ is given by

$$N g[(-k)_N]$$

which corresponds to

$$G[n] \xleftrightarrow{DFT} N g[(-k)_N]$$

iv) Symmetry

If $x[n]$ is real-valued then $X[k]$ is conjugate symmetric:

$$X[k] = X^*[-k] = X^*[(N-k)_N]$$

If $x[n]$ is imaginary then $X[k]$ is conjugate anti-symmetric:

$$X[k] = -X^*[-k] = -X^*[(N-k)_N]$$

v) Circular Convolution

Let $h[n]$ and $x[n]$ be finite-length sequences of length N with N-point DFTs $H[k]$ and $X[k]$. The sequence that has a DFT equal to the product $Y[k] = H[k].X[k]$ is

$$y[m] = \left[\sum_{k=0}^{N-1} h[(k)_N] \cdot x[(m-k)_N] \right], \quad 0 \le m \le N-1$$

where $h[(k)_N]$ and $x[(m-k)_N]$ are the periodic extensions of $x[n]$ and $h[n]$, respectively.

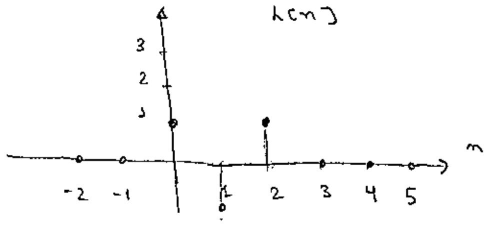
The N-point circular convolution of $h[m]$ with $x[m]$ is given by

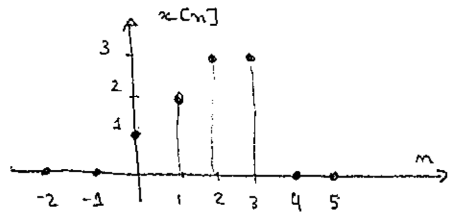
$$y[m] = h[m] \, \textcircled{N} \, x[m]$$

The circular convolution of two finite-length sequences $h[n]$ and $x[n]$ is equivalent to one period of the periodic convolution of the periodic sequences $h[(m)_N]$ and $x[(m)_N]$ as described by

$$y[m] = h[m] \, \textcircled{N} \, x[m] = \Big[\underbrace{h[(m)_N] * x[(m)_N]}_{\sum_{k=-\infty}^{\infty} y[m+kN]} \Big] \bmod N = y[(m+kN)_N]$$

Ex: Let us perform the four-point circular convolution of the two sequences $h[n]$ and $x[n]$ below





N = 4

the four-point circular is

$$y[m] = \left[\sum_{k=0}^{3} h[(m-k)_N] \, x[(m)_N] \right]_N$$

To evaluate $y[0]$, we multiply this sequence by $x[k]$ and sum the product from $k=0$ to $k=3$. The result is $y[0] = 1$. Next, to find the value of $y[1]$, we evaluate the sum

$$y[1] = \sum_{k=0}^{3} h[(1-k)_N] \, x[(k)_N] = 4$$

Repeating for $m=2$ and $m=3$, we have

$$y[2] = \sum_{k=0}^{3} h[(2-k)_N] \, x[(k)_N] = 2$$

$$y[3] = \sum_{k=0}^{3} h[(3-k)_N] \, x[(k)_N] = 2$$

Therefore, we have $y[m] = h[n] \, \textcircled{N=4} \, x[n] = \delta[n] + 4\delta[n-1] + 2\delta[n-2] + 2\delta[n-3]$

By comparison $y'[n] = h[n] * x[n] = \delta[n] + \delta[n-1] + 2\delta[n-2] + 2\delta[n-3] + 3\delta[n-5]$

Ex: Let us perform the N-point circular convolution of $x_1[n]$ and $x_2[n]$ where

$$x_1[n] = x_2[n] = \begin{cases} 1 & , \quad 0 \le n \le N-1 \\ 0 & , \quad \text{otherwise} \end{cases}$$

Because the N-point DFTs $x_1[n]$ and $x_2[n]$ are

$$X_1[k] = X_2[k] = \sum_{n=0}^{N-1} W_N^{nk} = \begin{cases} N & , \quad k=0 \\ 0 & , \quad \text{otherwise} \end{cases}$$

then

$$X[k] = X_1[k] \cdot X_2[k] = \begin{cases} N^2 & , \quad k=0 \\ 0 & , \quad \text{otherwise} \end{cases}$$

Therefore, the N-point circular of $x_1[n]$ with $x_2[n]$ is the inverse DFT of $X[k]$, which is

$$x[n] = \begin{cases} N & , \quad 0 \le n \le N-1 \\ 0 & , \quad \text{otherwise} \end{cases}$$

B. Fast Fourier Transform Algorithms

In this section, we look at fast algorithms for computing the DFT, which are collectively known as fast Fourier transform (FFT) algorithms. The basic strategy is to decompose the N-point DFT computation into computational tasks of smaller-size DFTs and to take advantage of the periodicity and symmetry of the complex number W_N^{kn}.

We begin with the radix-2 decimation-in-time FFT proposed by Cooley and Tukey in 1965. The N-point DFT of the sequence $x[n]$ of length N is

$$X[k] = \sum_{n=0}^{N-1} x[n] \, W_N^{nk}$$

Evaluating $X[k]$ requires on the order of N complex multiplications and N complex additions for each value of k, resulting in a total of N^2 complex multiplications and additions to compute an N-point DFT.

The basic strategy is one of "divide and conquer", which involves decomposing an N-point DFT into successively smaller DFTs. For example, for a sequence $x[n]$ of even length (i.e., N is divisible by 2). If $x[n]$ is decimated into two sequences of length $N/2$, then the computation of an $N/2$-point DFT of each sequence requires $(N/2)^2$ multiplications and additions. The two DFTs require $2(N/2)^2 = \frac{1}{2}N^2$ multiplications and additions. In what follows, we will show how to compute the N-point DFT of $x[n]$ from these two $N/2$-point DFTs in fewer than $N^2/2$ operations.

i) Decimation-in-Time FFT

The decimation-in-time FFT algorithm is based on splitting (decimating) $x[n]$ into smaller sequences and finding $X[k]$ from the DFTs of these decimated sequences. The key point is to exploit sequences whose length is a power of 2.

Let $x[n]$ be a sequence of length $N = 2^v$ and suppose that $x[n]$ is split (decimated) into two subsequences, each of length $N/2$ as illustrated in the figure below.

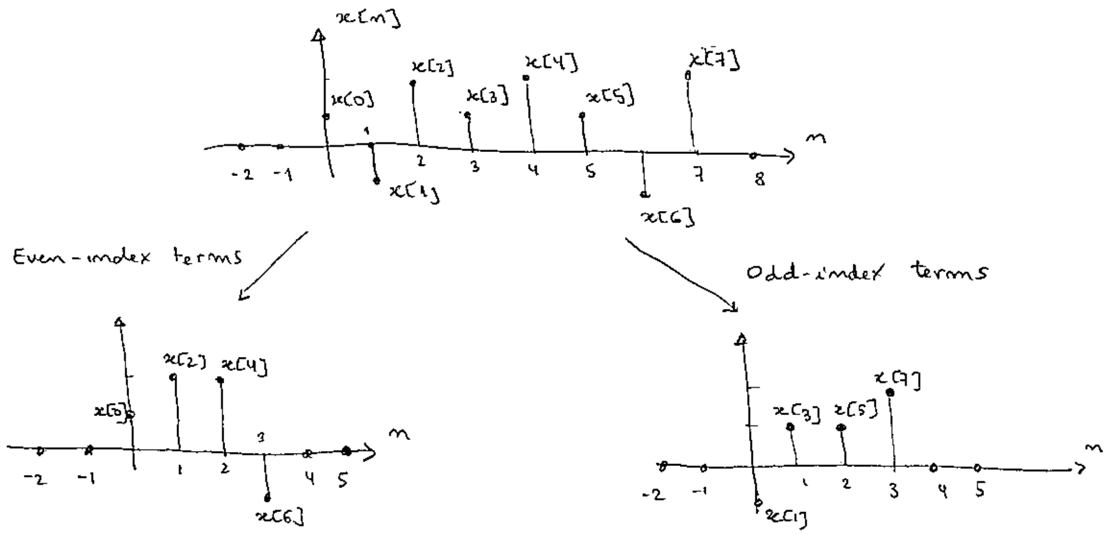


Fig. 5 Decimating a sequence of length N=8 by a factor of 2.

Following the decimation shown in Fig. 5, the first sequence $g[n]$ is formed from the even-index terms as described by

$$g[n] = x[2n], \quad n = 0, 1, \ldots, \frac{N}{2} - 1$$

and the second, $h[n]$, is formed from the odd-index terms as given by

$$h[n] = x[2n+1], \quad n = 0, 1, \ldots, \frac{N}{2} - 1$$

We can then express the N-point DFT of $x[n]$ as

$$X[k] = \sum_{n=0}^{N-1} x[n]\, W_N^{nk} = \sum_{n \text{ even}} x[n]\, W_N^{nk} + \sum_{n \text{ odd}} x[n]\, W_N^{nk}$$

$$= \sum_{l=0}^{\frac{N}{2}-1} g[l]\, W_N^{2lk} + \sum_{l=0}^{\frac{N}{2}-1} h[l] \cdot W_N^{(2l+1)k}$$

Because $W_N^{2lk} = W_{N/2}^{lk}$, $X[k]$ can be rewritten as

$$X[k] = \sum_{l=0}^{\frac{N}{2}-1} g[l]\, W_{N/2}^{lk} + W_N^{k} \sum_{l=0}^{\frac{N}{2}-1} h[l]\, W_{N/2}^{lk}$$

$$= G[k] + W_N^{k}\, H[k], \quad k = 0, 1, \ldots, N-1,$$

where $G[k]$ is the $N/2$-point DFT of $g[n]$ and $W_N^k H[k]$ is the $N/2$-point DFT of $h[n]$.

Although the $N/2$-point DFT of $g[n]$ and $h[n]$ are sequences of length $N/2$ the periodicity of the complex exponentials allows us to write

$$G[k] = G[k + N/2] \quad \text{and} \quad H[k] = H[k + N/2]$$

Therefore, $X[k]$ may be computed from the $N/2$-point DFTs $G[k]$ and $H[k]$. It should be noted that because

$$W_N^{k+N/2} = W_N^{k}\, W_N^{N/2} = -W_N^{k}$$

then we have

$$W_N^{k+N/2}\, H[k + N/2] = -W_N^{k}\, H[k],$$

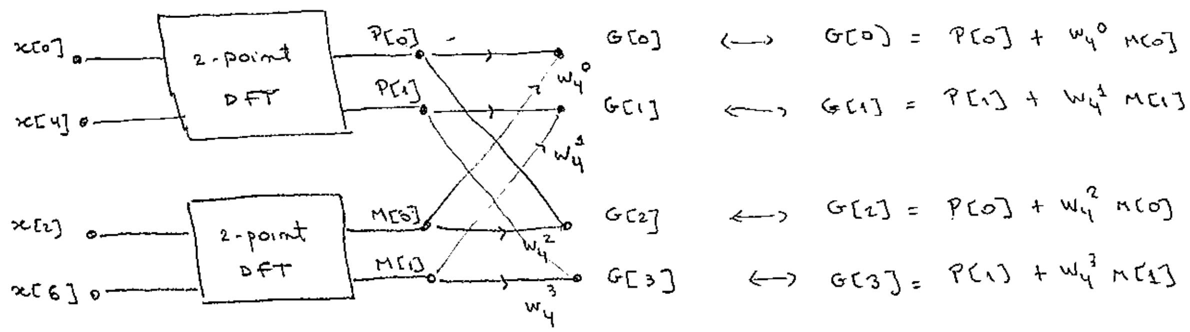


Fig. 1 Decimation of the four-point DFT into two two-point DFTs in the decimation-in-time FFT.

If N is a power of 2 then the decimation may be continued until there are only two-point DFTs of the form shown in the figure below.

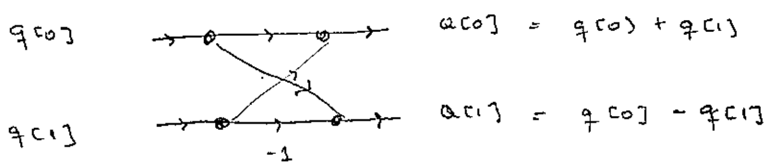


Fig. 2 A two-point DFT.

The basic computational unit of the FFT is called a butterfly. This structure may be simplified by factoring out a term W_N^r from the lower branch and the factor that remains is $W_N^{N/2} = -1$ as illustrated below.

Fig. 3 (a) The butterfly which is the basic computational element of the FFT algorithm

(b) A simplified butterfly with only one complex multiplication which shows that it is only necessary to form the products $W_N^k H[k]$ for $k = 0, 1, \dots, N/2 - 1$. The complex exponentials multiplying $H[k]$ in $X[k] = G[k] + W_N^k H[k]$ are called twiddle factors. A block diagram showing the computations that are required for the first stage of an eight-point decimation-in-time FFT is shown in the figure below.

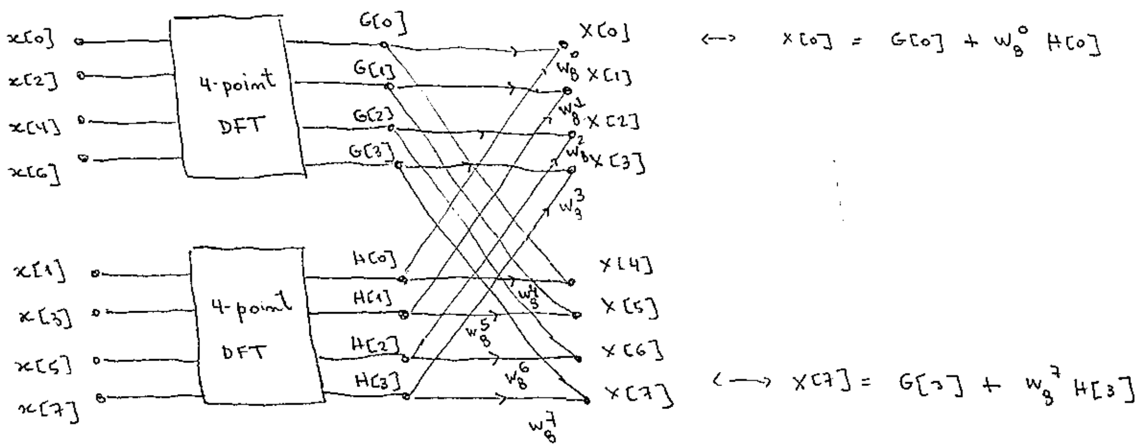


Fig. 4 An eight-point decimation-in-time FFT algorithm after the first decimation.

If $N/2$ is even then $g[m]$ and $h[m]$ may again be decimated. For example, $G[k]$ may be evaluated as follows:

$$G[k] = \sum_{m=0}^{N/2-1} g[m] \cdot W_{N/2}^{mk} = \sum_{m \text{ even}}^{N/2-1} g[m]\, W_{N/2}^{mk} + \sum_{m \text{ odd}}^{N/2-1} g[m] \cdot W_{N/2}^{mk}$$

$$= \sum_{m=0}^{N/4-1} g[2m]\, W_{N/4}^{mk} + W_{N/2}^{k} \sum_{m=0}^{N/4-1} g[2m+1]\, W_{N/4}^{mk},$$

where the first term above is the $N/4$-point DFT of the even samples of $g[m]$ and the second is the $N/4$-point DFT of the odd samples. A block diagram illustrating this decomposition is shown in the next figure.

Computational complexity :

Computing an N-point DFT using a radix-2 decimation-in-time FFT is much more efficient than calculating the DFT directly. For example, if $N = 2^v$ then there are $\log_2 N = v$ stages of computation. Because each stage requires $N/2$ complex multiplications by the twiddle factors W_N^r and N complex additions, there are a total of $\frac{1}{2} N \log_2 N$ complex multiplications and $N \log_2 N$ complex additions.

From the structure of the decimation-in-time FFT algorithm once a butterfly operation has been performed on a pair of complex numbers there is no need to save the input pair. Therefore, the output pair may be stored in the same registers as the input.

EX: Assume that a complex multiplication takes $1\mu s$ and that the amount of time to compute a DFT is determined by the time it takes to perform all of the multiplications.

(a) How much time does it take to compute a 1024-point DFT directly?

Since an N-point DFT directly requires N^2 complex multiplications and a complex multiplication takes $1\,\mu s$, the direct evaluation of a 1024-point DFT requires

$$t_{DFT} = (1024)^2 \cdot 10^{-6}\, s \approx 1.05\, s$$

(b) How much time is required if an FFT is used?

With a radix-2 FFT, we need $(N/2)\log_2 N$ multiplications. For $N = 1024$, an FFT requires $512 \cdot \log_2 2^{10} = 5120$ multiplications. Therefore, an 1024-point DFT using an FFT requires

$$t_{FFT} = 5.120 \times 10^{-6}\, s \;.= \; 5.12\ ms.$$

ii) Decimation-in-Frequency FFT

Another class of FFT algorithms may be derived by decimating the output sequence $X[k]$ into smaller and smaller subsequences. These algorithms are called decimation-in-frequency FFTs and may be derived as follows. Let N be a power of 2, $N = 2^{\nu}$, and consider separately evaluating the even-index and odd-index samples of $X[k]$. The even samples are

$$X[2k] = \sum_{m=0}^{N-1} x[m] \cdot W_N^{2mk}$$

Separating the above sum into the first $N/2$ points and the last $N/2$ points, and using the fact that $W_N^{2mk} = W_{N/2}^{mk}$, this becomes

$$X[2k] = \sum_{m=0}^{\frac{N}{2}-1} x[m]\, W_{N/2}^{mk} + \sum_{m=N/2}^{N-1} x[m]\, W_{N/2}^{mk}$$

With a change in the indexing on the second sum we have

$$X[2k] = \sum_{m=0}^{\frac{N}{2}-1} x[m]\, W_{N/2}^{mk} + \sum_{m=0}^{\frac{N}{2}-1} x\left[m+\tfrac{N}{2}\right] W_{N/2}^{(m+\frac{N}{2})k}$$

Then, since $W_{N/2}^{(m+\frac{N}{2})k} = W_{N/2}^{mk}$ we have

$$X[2k] = \sum_{m=0}^{\frac{N}{2}-1} \left[x[m] + x[m+N/2] \right] W_{N/2}^{mk} ,$$

which is the $N/2$-point DFT of the sequence that is formed by adding the first $N/2$ points of $x[m]$ to the last $N/2$.

Proceeding in the same way for the odd samples of $X[k]$ leads to

$$X[2k+1] = \sum_{m=0}^{\frac{N}{2}-1} W_N^{m} \left[x[m] - x[m+N/2] \right] W_{N/2}^{mk}$$

The above expressions represent the $\frac{N}{2}$-point DFT of the following sequences:

$$x_e = \left(x[m] + x[N/2+m] \right)$$
$$x_o = \left(x[m] - x[m+N/2] \right) , \quad 0 \le m \le \tfrac{N}{2}-1$$

respectively.

A flowgraph illustrating the first stage of decimation is shown in the figure below. As occurs with the decimation-in-time FFT, the decimation may be continued until only two-point DFTs remain.

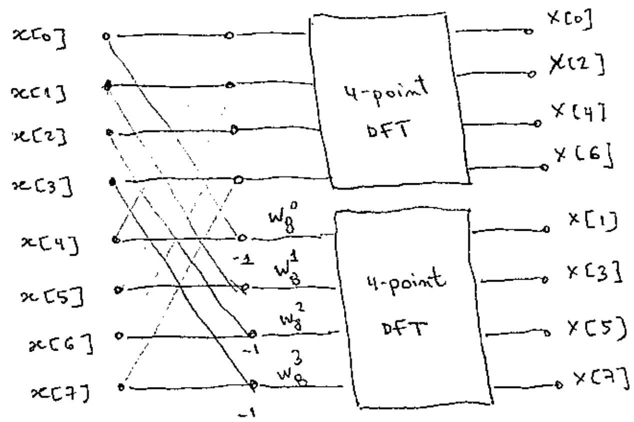


Fig. 6 An eight-point decimation-in-frequency FFT algorithm after the first stage of decimation.

As can be seen from this figure, the input samples $x[n]$ are in a sequential order, while the output DFT samples appear in a decimated form, with the even-indexed samples appearing as the outputs of one DFT and the odd-indexed samples appearing as the outputs of the other DFT. We can continue this decomposition until the smallest DFTs are two-point DFTs. The computational complexity of the decimation-in-frequency FFT is the same as the decimation-in-time FFT, i.e., $\frac{N}{2} \log_2 N$ complex multiplications and $N \log_2 N$ complex additions.

iii) Inverse DFT computation

An FFT algorithm for computing the DFT samples can also be used to calculate efficiently the inverse DFT. Consider an N-point sequence $x[n]$ with an N-point DFT $X[k]$. The sequence is related to the samples $X[k]$ through

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] W_N^{-nk}$$

If we multiply both sides of the above equation by N and take the complex conjugate, we arrive at

$$N x^*[n] = \frac{1}{N} \sum_{k=0}^{N-1} X^*[k] \cdot W_N^{nk}$$

The right-hand side of the above expression can be recognised as the N-point DFT of the sequence $X^*[k]$ and can be computed by any FFT algorithm. The desired IDFT $x[n]$ is then obtained as

$$x[n] = \frac{1}{N} \left\{ \sum_{k=0}^{N-1} X^*[k] \, W_N^{nk} \right\}^*$$

In summary, given an N-point DFT $X[k]$, we first form its complex conjugate sequence $X^*[k]$, then compute the N-point DFT of $X^*[k]$, form the complex conjugate of the DFT computed and divide each sample by N.

To conclude this part, it is useful to know that for sequences whose length N is not a power of two integer, it is possible to modify the procedure by zero padding or by a procedure known as index mapping, which maps the sample indices n and k into two-dimensional indices.

Ex: Show how an N-point FFT may be used to evaluate the N-point DFT of two real-valued sequences.

We can first form the N-point complex sequence

$$x[n] = x_1[n] + j\, x_2[n]$$

After computing the N-point DFT de $x[n]$, we extract $X_1[k]$ and $X_2[k]$ from $X[k]$ by exploiting the symmetry of the DFT. Specifically,

$$X_1[k] = \frac{1}{2}\left[X[k] + X^*[(N-k)_N]\right]$$ — conj symmetric part of $X[k]$

$$X_2[k] = \frac{1}{2}\left[X[k] - X^*[(N-k)_N]\right]$$ — conj antisymmetric part of $X[k]$

C. Linear Convolution using the DFT

Linear convolution is a key operation in most signal processing applications. Since an N-point DFT can be implemented very efficiently using the FFT which requires $N(\log_2 N)$ arithmetic operations, it is of interest to investigate methods for the implementation of the linear convolution using the DFT.

Let us first consider two finite-length sequences $h[n]$ and $x[n]$ of lengths N and M, respectively. Denote $L = M+N-1$ and define two length-L sequences as described by

$$x_p[n] = \begin{cases} g[n] & , \quad 0 \leq n \leq M-1 \\ 0 & , \quad M \leq n \leq L-1 \end{cases}$$

$$h_p[n] = \begin{cases} h[n] & , \quad 0 \leq n \leq N-1 \\ 0 & , \quad N \leq n \leq L-1 , \end{cases}$$

obtained by appending $g[n]$ and $h[n]$ with zeros.

The linear convolution is described by

$$y[n] = h[n] * x[n]$$

In order to emplement the above linear convolution, we perform the following steps:

i) we compute the L-point DFTs of $x_p[n]$ and $h_p[n]$, resulting in $X_p[k]$ and $H_p[k]$, respectively.

ii) we compute the L-point IDFT of the product $X_p[k] \cdot H_p[k]$ which results in $y[n]$.

Despite its computational savings, there are some difficulties with the DFT approach. If $x[n]$ is very long we must commit a significant amount of time computing very long DFTs, resulting in very long delays.

The solution to these problems is to use block convolution, which involves segmenting the signal to be filtered $x[n]$ into sections. Each section is then filtered with the impulse response (or filter) $h[n]$, and the filtered sections are pieced together to form the sequence $y[n]$. There are two block convolution techniques: overlap-add and overlap-save.

i) Overlap-Add

Let $x[n]$ be a sequence that is to be convolved with an impulse response $h[n]$ of length L:

$$y[n] = h[n] * x[n] = \sum_{k=0}^{L-1} h[k]\, x[n-k]$$

Assume that $x[n] = 0$ for $n > 0$ and that the length of $x[n]$ is much greater than M. The signal $x[n]$ is then partitioned into nonoverlapping subsequences of length M as illustrated in the figure below.

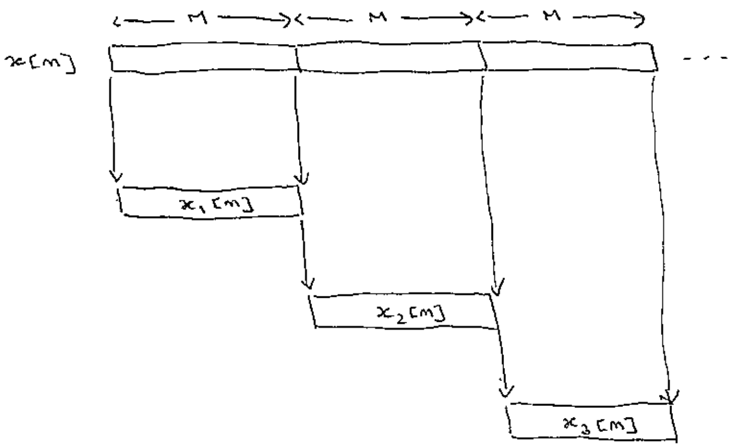


Fig. 8 Partitioning a sequence into subsequences of length M.

The signal $x[n]$ may be written as a sum of shifted finite-length sequences of length M:

$$x[n] = \sum_{i=0}^{\infty} x_i[n - Mi],$$

where

$$x_i[n] = \begin{cases} x[n + Mi] & , \quad n = 0, 1, \dots, M-1 \\ 0 & , \quad \text{otherwise} \end{cases}$$

Therefore, the linear convolution of $x[n]$ with $h[n]$ is given by

$$y[n] = h[n] * x[n]$$
$$= \sum_{i=0}^{\infty} h[n] * x_i[n-Mi]$$
$$= \sum_{i=0}^{\infty} y_i[n-Mi],$$

where $y_i[n]$ is the linear convolution of $x_i[n]$ with $h[n]$ given by

$$y_i[n] = h[n] * x_i[n]$$

Since each sequence $y_i[n]$ is of length $N = L+M-1$, it may be found by multiplying the N-point DFTs of $h[n]$ and $x_i[n]$. The reason for the name overlap-add is that, for each i, the sequences $y_i[n]$ and $y_{i+1}[n]$ overlap at $(N-M)$ points, and in performing the sum $\sum_{i=0}^{\infty} y_i[n-Mi]$ these overlapping points are added, as illustrated in the figure below.

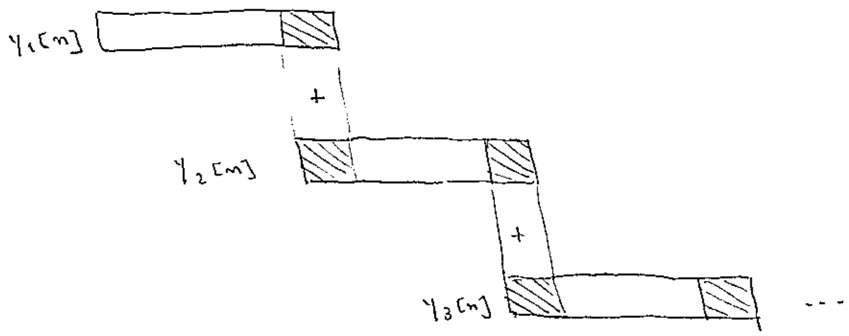


Fig. 9. Performing the sum of the overlapping segments.

ii) Overlap-save

This method takes advantage of the fact that the aliasing that occurs in circular convolution only affects a portion of the sequence. For example, if $x[n]$ and $h[n]$ are finite-length sequences of lengths N and L, respectively, the linear convolution $y[n]$ is a finite-length sequence of lengths $N+L-1$. Therefore, assuming that $N > L$, if we perform an N-point circular convolution of $h[n]$ and $x[n]$ we obtain

$$y_c[n] = h[n] \, Ⓝ \, x[n] = \sum_{k=0}^{L-1} h[k]\, x[(n-k)_N]$$

Because $y[n+N]$ is the only term that is aliased into the interval $0 \le n \le N-1$ and $y[n+N]$ only overlaps the first $L-1$ values of $y[n]$, the remaining values in the circular convolution will not be aliased. In other words, the first $L-1$ values of the circular convolution are not equal to the linear convolution, whereas the last $M = N - L + 1$ are the same as illustrated in the Figure below.

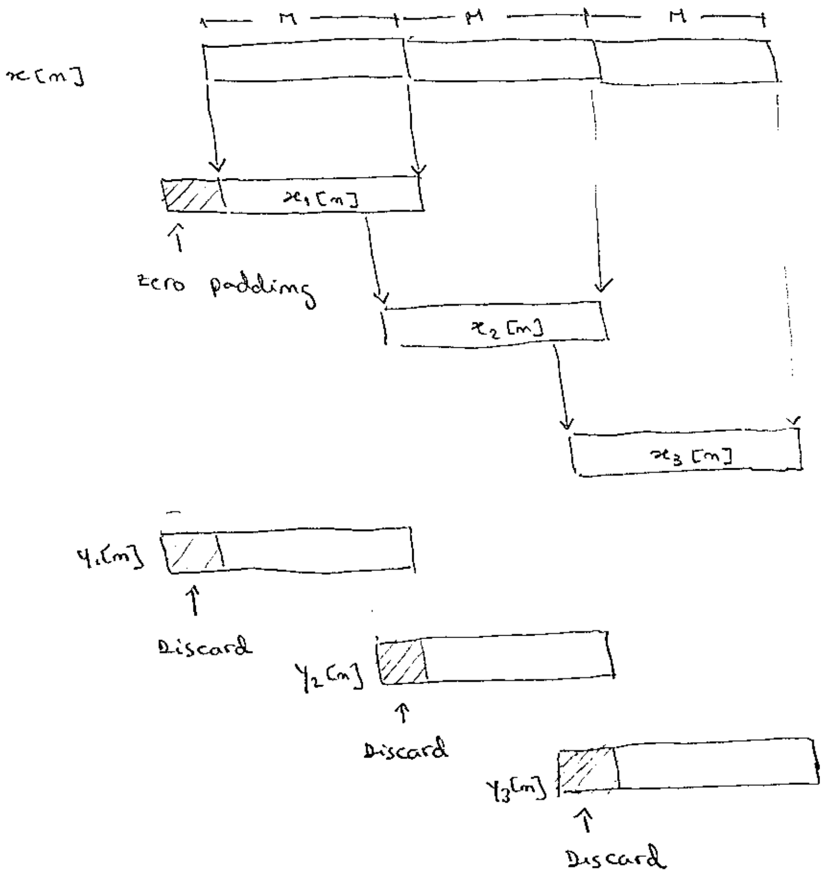


Fig. 10 Illustration of the overlap-save method of block convolution.

Thus, with the appropriate partitioning of the input sequence $x[n]$, linear convolutional may be performed by piecing together circular convolutions. The procedure is as follows.

1 - Let $x_i[n]$ be the sequence

$$x_i[n] = \begin{cases} 0 & , \quad 0 \le n < L-1 \\ x[n-L+1] & , \quad L-1 \le n \le N-1 \end{cases}$$

2 - Perform the N-point circular convolution of $x_1[n]$ with $h[n]$ by forming the product $H[k].X_1[k]$ and then finding the inverse DFT, $y_1[n]$. The first L-1 values of the circular convolution are aliased, and the last N-L+1 values correspond to the linear convolution of $x[n]$ with $h[n]$. Due to zero padding at the start of $x_1[n]$, these last N-L+1 values are the first N-L+1 values of $y[n]$:

$$y[n] = y_1[n+L-1] \quad , \quad 0 \le n \le N-L$$

3 - Let $x_2[n]$ be the N-point sequence that is extracted from $x[n]$ with the first L-1 values overlapping with those of $x_1[n]$.

4 - Perform an N-point circular convolution of $x_2[n]$ with $h[n]$ by forming the product $H[k].X_2[k]$ and taking the inverse DFT. The first L-1 values of $y_2[n]$ are discarded and the final N-L+1 values are saved and concatenated with the saved value of $y_1[n]$:

$$y[n+N-L+1] = y_2[n+L-1] \quad , \quad 0 \le n \le N-L$$

5 - steps 3 and 4 are repeated until all the values of the linear convolution have been computed.

The reason for the name overlap-save is that $x[n]$ is partitioned into overlapping sequences of length N and, after performing the N-point circular convolution, only the last N-L+1 values are saved.

D. The Discrete Cosine Transform

In some applications of signal compression, it is of interest to have orthogonal transforms that represent a real time-domain sequence $x[n]$ by a real transform-domain sequence $X[k]$. In this section, we present such a transform which is referred to as discrete cosine transform (DCT).

Let us consider a general class of finite-length transform representations of the form

$$A[k] = \sum_{m=0}^{N-1} x[m]\, \phi_k^*[m],$$

$$x[m] = \frac{1}{N} \sum_{k=0}^{N-1} A[k]\, \phi_k[m],$$

where the sequences $\phi_k[m]$, referred to as the basis sequences, are orthogonal to one another, i.e.,

$$\frac{1}{N} \sum_{m=0}^{N-1} \phi_k[m]\, \phi_m^*[m] = \begin{cases} 1, & m = k \\ 0, & m \neq k. \end{cases}$$

The DCT is a transform with basis sequences $\phi_k[m]$ that are cosines. Since cosines are both periodic and have even symmetry, the extension of $x[m]$ outside the range $0 \le m \le (N-1)$ will be both periodic and symmetric. The idea of the DCT is to form a periodic, symmetric sequence from a finite-length sequence in such a way that the original finite-length sequence can be uniquely recovered. Since there are many ways to do this, there are many definitions of the DCT. In this exposition, we focus on the type-2 periodic extension for the so-called DCT-2 form of the DCT. shown below

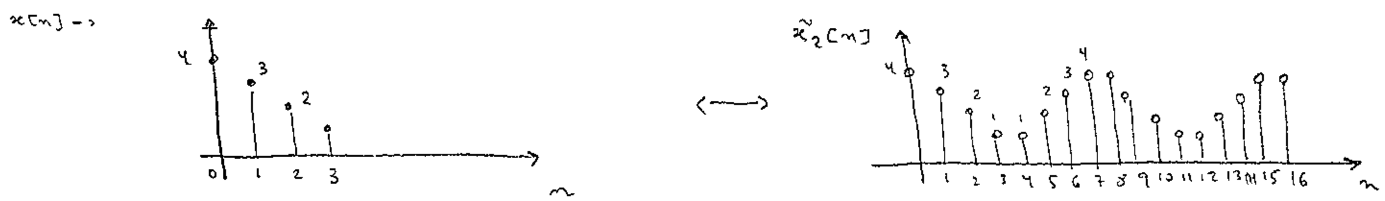


Fig. 11 Extension of a four-point sequence $x[n]$

Let us consider the signal $x[m]$ and its extension to have period $2N$, and the periodic sequence given by

$$\tilde{x}[m] = x\left[(m)_{2N}\right] + x\left[(-m-1)_{2N}\right],$$

which is called Type-2 periodic symmetry because the periodic sequence $\tilde{x}[m]$ has even periodic symmetry about the "half sample" points $-1/2$, $N-1/2$, $2N-1/2$, etc.

The DCT is defined by the transform described by

$$X^c[k] = \alpha[k] \sum_{m=0}^{N-1} x[m] \cos\left(\frac{\pi k (2m+1)}{2N}\right), \qquad 0 \leq k \leq N-1,$$

where the weighting factor $\alpha[k]$ is given by

$$\alpha[k] = \begin{cases} \frac{1}{\sqrt{N}}, & \text{for } k=0 \\ \sqrt{\frac{2}{N}}, & \text{for } k \neq 0 \end{cases}$$

The inverse DCT is defined by

$$x[m] = \sum_{k=0}^{N-1} \beta[k]\, X^c[k] \cos\left(\frac{\pi k (2m+1)}{2N}\right), \qquad 0 \leq m \leq N-1,$$

where the weighting factor $\beta[k]$ is given by

$$\beta[k] = \begin{cases} \frac{1}{\sqrt{N}}, & \text{for } k=0 \\ \sqrt{\frac{2}{N}}, & \text{for } k \neq 0 \end{cases}$$

A convenient approach to processing signals with the DCT is to express them into matrix form. In particular, let the DCT samples be expressed by

$$\underset{N \times 1}{X} = \underset{N \times N}{\underline{C}_N} \cdot \underset{N \times 1}{\underline{x}},$$

where $\underline{X} = [X[0]\ X[1] \ldots X[N-1]]^T$ is an $N \times 1$ vector of DCT samples

$\underline{x} = [x[0]\ x[1] \ldots x[N-1]]^T$ is an $N \times 1$ vector of the input samples

and

$\underline{C}_N$ is the $N \times N$ DCT matrix given by

$$\underline{C}_N = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ \sqrt{2}\cos\left(\frac{\pi}{2N}\right) & \sqrt{2}\cos\left(\frac{3\pi}{2N}\right) & & \ddots & \\ \sqrt{2}\cos\left(\frac{\pi}{2N}\right) & & & & \\ \vdots & & & & \\ \sqrt{2}\cos\left(\frac{\pi}{2N}\right) & & & & \sqrt{2}\cos\left(\frac{(N-1)(2N+1)\pi}{2N}\right) \end{bmatrix}$$

Likewise, the inverse DCT can be expressed by

$$\underline{x} = \underline{C}_N^{-1} \underline{X},$$

where $\underline{C}_N^{-1}$ is the $N \times N$ inverse DCT matrix given by

$$\underline{C}_N^{-1} = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & \sqrt{2}\cos\left(\frac{\pi}{2N}\right)^* & \sqrt{2}\cos\left(\frac{\pi}{2N}\right)^* & \cdots & \sqrt{2}\cos\left(\frac{\pi}{2N}\right)^* \\ 1 & \sqrt{2}\cos\left(\frac{3\pi}{2N}\right)^* & & \ddots & \\ 1 & \vdots & & & \\ \vdots & \vdots & & & \\ 1 & & & & \sqrt{2}\cos\left(\frac{(N-1)(2N+1)\pi}{2N}\right) \end{bmatrix}$$

and it follows that $\underline{C}_N^{-1} = \underline{C}_N^{H}$.

The properties of the DCT are similar to those of the DFT and can be summarised for the pairs $g[n] \overset{DCT}{\longleftrightarrow} G[k]$ and $h[n] \overset{DCT}{\longleftrightarrow} H[k]$ as:

i) Linearity : Given a sequence $x[n] = \alpha g[n] + \beta h[n]$, we have

$$\alpha g[n] + \beta h[n] \overset{DCT}{\longleftrightarrow} \alpha G_{DCT}[k] + \beta H_{DCT}[k]$$

ii) Symmetry : The DCT of the conjugate sequence $g^*[n]$ is $G^*_{DCT}[k]$, i.e.,

$$g^*[n] \overset{DCT}{\longleftrightarrow} G^*_{DCT}[k]$$

iii) Energy Preservation : This property is similar to Parseval's relation for the DFT and given by

$$\sum_{n=0}^{N-1} |g[n]|^2 = \frac{1}{2N} \sum_{k=0}^{N-1} \alpha[k]\, |G_{DCT}[k]|^2$$

E. The z-Transform

The z-transform is a useful tool in the analysis of discrete-time signals and systems that generalizes the DTFT. It can be used to solve difference equations, evaluate the response of LTI systems and design linear filters.

Let us recall the DTFT of a sequence $x[n]$ defined by

$$X(e^{jw}) = \sum_{n=-\infty}^{\infty} x[n]\, e^{-jnw},$$

where it is required that the signal $x[n]$ be absolutely summable for this series to converge. Unfortunately, many of the signals that we would like to consider are not absolutely summable and do not have a DTFT. Examples of such signals include $x[n] = u[n]$, $x[n] = (0.5)^n u[-n]$ and $x[n] = \sin n w_0$. The z-transform allows one to deal with such signals.

The z-transform of a discrete-time signal $x[n]$ is defined by

$$X[z] = \sum_{n=-\infty}^{\infty} x[n]\, z^{-n},$$

where $z = re^{jw}$ is a complex variable. The values of z for which the sum converges define a region in the z-plane referred to as the region of convergence (ROC). In terms of notation, if $x[n]$ has a z-transform $X[z]$, we write

$$x[n] \overset{z}{\longleftrightarrow} X[z]$$

The z-transform may be viewed as the DTFT of an exponentially weighted sequence. Specifically, with $z = re^{jw}$ we have

$$X[z] = \sum_{n=-\infty}^{\infty} x[n]\, z^{-n} = \sum_{n=-\infty}^{\infty} (r^{-n} x[n])\, e^{-jnw},$$

which indicates that $X[z]$ is the DTFT of $r^{-n} x[n]$. The ROC is then determined by the range of values of r for which

$$\sum_{n=-\infty}^{\infty} |x[n]\, r^{-n}| < \infty$$

Because the z-transform is a function of a complex variable, it is convenient to describe it using the complex z-plane:

$$z = \operatorname{Re}(z) + j\operatorname{Im}(z) = r e^{j\omega},$$

which can be illustrated with the axes of the z-plane as follows.

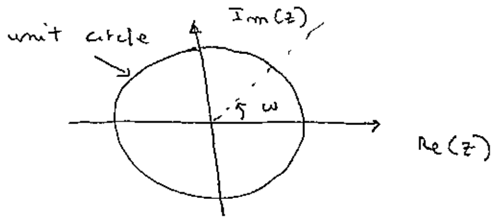


Fig 11. The unit circuit in the complex z-plane.

The contour corresponding to $|z| = 1$ is a circle of unit radius referred to as the unit circle. The z-transform evaluated on the unit circle corresponds to the DTFT

$$X(e^{j\omega}) = X(z)\Big|_{z = e^{j\omega}}$$

More specifically, evaluating $X(z)$ at points around the unit circle, beginning at $z = 1$ $(\omega = 0)$, through $z = j$ $(\omega = \pi/2)$ to $z = -1$ $(\omega = \pi)$, we obtain the values of $X(e^{j\omega})$ for $0 \leq \omega \leq \pi$. Note that in order for the DTFT of a signal to exist, the unit circle must be within the region of convergence of $X(z)$.

Many of the signals of interest in digital signal processing have z-transforms that are rational functions of z:

$$X(z) = \frac{B(z)}{A(z)} = \frac{\sum_{k=0}^{q} b[k]\, z^{-k}}{\sum_{k=0}^{p} a[k]\, z^{-k}}$$

Factoring the numerator and denominator polynomials, a rational z-transform may be expressed as follows:

$$X(z) = c\, \frac{\prod_{k=1}^{q} (1 - \beta_k z^{-1})}{\prod_{k=1}^{p} (1 - \alpha_k z^{-1})}$$

The roots of the numerator polynomial, β_k, are known as the zeros of $X[z]$, and the roots of the denominator polynomial, α_k, are referred to as the poles. The poles and zeros uniquely define the functional form of a rational z-transform within a constant. Therefore, they provide a concise representation for $X(z)$ that is often represented pictorially in terms of a pole-zero plot in the z-plane. With a pole-zero plot, the location of each pole is indicated by an "x" and the location of each zero is indicated by an "o", with the region of convergence indicated by shading the appropriate region of the z-plane. The region of convergence is, in general, in the form of a ring-shaped object

$$\alpha < |z| < \beta$$

If $\alpha = 0$, the ROC may also include the point $z=0$, and if $\beta = \infty$, the ROC may also include infinity. For a rational $X[z]$, the region of convergence will contain no poles. Properties of ROC include:

1 - A finite-length sequence has a z-transform with a ROC that includes the entire z-plane except, possibly, $z=0$ and $z=\infty$. The point $z=\infty$ will be included if $x[n]=0$ for $n<0$, and the point $z=0$ will be included if $x[n]=0$ for $n>0$.

2 - A right-sided sequence has a z-transform with a region of convergence that is the exterior of a circle:

$$\text{ROC}: |z| > \alpha$$

3 - A left-sided sequence has a z-transform with a ROC that is the interior of a circle:

$$\text{ROC}: |z| < \beta$$

Ex : Find the z-transform of $x[n] = \alpha^n u[n]$ and show the ROC.

Using the z-transform, we have

$$X[z] = \sum_{n=-\infty}^{\infty} x[n]\, z^{-n} = \sum_{n=0}^{\infty} \alpha^n z^{-n}$$

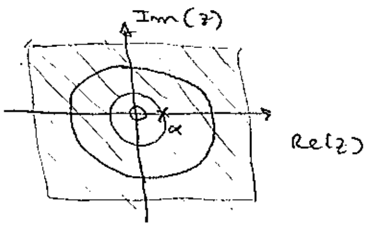
$$= \sum_{n=0}^{\infty} (\alpha z^{-1})^n = \frac{1}{1-\alpha z^{-1}},$$

with the sum converging if $|\alpha z^{-1}| < 1$.

Therefore, the ROC is the exterior of a circle defined by the set $|z| > |\alpha|$. Expressing $X[z]$ in terms of positive powers of z, we obtain

$$X[z] = \frac{z}{z-\alpha}$$

and see that $X[z]$ has a zero at $z=0$ and a pole at $z=\alpha$. A pole-zero diagram with the ROC is shown below.



Note : this refers to the z-tranform of a right-sided sequence, i.e. $x[n] = \alpha^n u[n]$

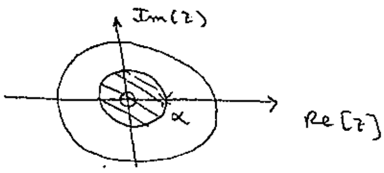
Note that if $|\alpha| < 1$, the unit circle in include in the ROC and the DTFT of $x[n]$ exists.

Ex : Find the z-transform of the sequence $x[n] = -\alpha^n u[-n-1]$. We then have

$$X[z] = \sum_{n=-\infty}^{\infty} x[n] . z^{-n} = -\sum_{n=-\infty}^{-1} \alpha^n z^{-n} = -\sum_{n=0}^{\infty} (\alpha^{-1} z)^{n+1}$$

$$= -\alpha^{-1} z \sum_{n=0}^{\infty} (\alpha^{-1} z)^n = \frac{\alpha^{-1} z}{1-\alpha^{-1} z} = \frac{1}{1-\alpha z^{-1}} = \frac{z}{z-\alpha}$$

with the sum converging if $|\alpha^{-1} z| < 1$ or $|z| < |\alpha|$. A pole-zero diagram with the ROC is shown below



Note that if $|\alpha| \leq 1$, the unit circle is not included within the ROC and the DTFT of $x[n]$ does not exist.

Ex: Find the z-transform of $x[n] = \left(\frac{1}{2}\right)^n u[n] - 2^n u[-n-1]$ and another signal that has the same z-transform but a different ROC.

Since we have a sum of two sequences, we can find the z-transform of each sequence separately and add them together. We can first compute the z-transform of $x_1[n] = \left(\frac{1}{2}\right)^n u[n]$

$$X_1[z] = \frac{1}{1 - \frac{1}{2} z^{-1}} \qquad |z| > \frac{1}{2}$$

and of $x_2[n] = -2^n u[-n-1]$ as described by

$$X_2[z] = \frac{1}{1 - 2 z^{-1}} \qquad |z| < 2$$

Therefore, the z-transform of $x[n] = x_1[n] + x_2[n]$ is

$$X[z] = X_1[z] + X_2[z] = \frac{1}{1 - \frac{1}{2} z^{-1}} + \frac{1}{1 - 2z^{-1}}$$

with a ROC $\frac{1}{2} < |z| < 2$, which is the set of all points in the ROC of both $X_1[z]$ and $X_2[z]$.

To find another sequence with the same z-transform each term corresponds to the z-transform of either a right-sided or a left-sided sequence depending on the ROC. Choosing right-sided sequences for both terms, we obtain

$$x'[n] = \left(\frac{1}{2}\right)^n u[n] + 2^n u[n],$$

where the ROC is $|z| > 2$.

Common z-transform pairs include the following:

Sequence	z-transform	ROC
$\delta[n]$	1	all z
$\alpha^n u[n]$	$\frac{1}{1-\alpha z^{-1}}$	$\|z\| > \|\alpha\|$
$-\alpha^n u[-n-1]$	$\frac{1}{1-\alpha z^{-1}}$	$\|z\| < \|\alpha\|$
$n\alpha^n u[n]$	$\frac{\alpha z^{-1}}{(1-\alpha z^{-1})^2}$	$\|z\| > \|\alpha\|$
$-n\alpha^n u[-n-1]$	$\frac{\alpha z^{-1}}{(1-\alpha z^{-1})^2}$	$\|z\| < \|\alpha\|$
$\cos(n\omega_0)u[n]$	$\frac{1-(\cos\omega_0)z^{-1}}{1-2(\cos\omega_0)z^{-1}+z^{-2}}$	$\|z\| > 1$
$\sin(n\omega_0)u[n]$	$\frac{(\sin\omega_0)z^{-1}}{1-2(\cos\omega_0)z^{-1}+z^{-2}}$	$\|z\| > 1$

Properties of the z-transform :

i) Linearity

The z-transform is a linear operator. If $x[n]$ has a z-transform $X[z]$ with a ROC R_x and $y[n]$ has a z-transform $Y[z]$ with a ROC R_y then

$$w[n] = a\,x[n] + b\,y[n] \overset{z}{\longleftrightarrow} W[z] = a\,X[z] + b\,Y[z]$$

The ROC of $w[n]$ will include the intersection of R_x and R_y, i.e

R_w contains $R_x \cap R_y$

However, the ROC of $W[z]$ may be larger. For example, if $x[n] = u[n]$ and $y[n] = u[n-1]$, the ROC of $X[z]$ and $Y[z]$ is $|z| > 1$. However, the z-transform of $w[n] = x[n] - y[n] = \delta[n]$ is the entire z plane.

ii) Shifting Property

The operation of shifting a sequence corresponds to multiplying the z-transform by a power of z. If a $x[n]$ has a z-transform $X[z]$,

$$x[n-n_0] \overset{z}{\longleftrightarrow} z^{-n_0}\,X[z]$$

Since shifting a sequence does not affect its absolute summability, shifting does not change the ROC. Therefore, the z-transform of $x[n]$ and $x[n-n_0]$ have the same ROC with the possible exception of adding or deleting the points $z=0$ and $z=\infty$.

iii) Time Reversal

If $x[n]$ has a z-transform $X[z]$ with a ROC R_x that is the ring $\alpha < |z| < \beta$, the z-transform of the time-reversed sequence $x[-n]$ is

$$x[-n] \overset{z}{\longleftrightarrow} X[z^{-1}]$$

and has a ROC $1/\beta < |z| < 1/\alpha$ which is denoted by $1/R_x$.

iv) Multiplication by an exponential

If a sequence $x[n]$ is multiplied by a complex exponential α^n:

$$\alpha^n x[n] \xleftrightarrow{z} X[\alpha^{-1} z]$$

This corresponds to a scaling of the z-plane. If the ROC of $X[z]$ is $r_- < |z| < r_+$, which will be denoted by R_x, the ROC of $X[\alpha^{-1} z]$ is $|\alpha| r_- < |z| < |\alpha| r_+$ which is denoted by $|\alpha| R_x$. As a special case, note that if $x[n]$ is multiplied by a complex exponential e^{jnw_0}, we have

$$e^{jnw_0} x[n] \xleftrightarrow{z} X[e^{-jnw_0} z],$$

which corresponds to a rotation of the z-plane.

v) Convolution Theorem

The convolution in the time domain is mapped into multiplication in the frequency domain, i.e.,

$$y[n] = x[n] * h[n] \xleftrightarrow{z} Y[z] = X[z] . H[z]$$

The ROC of $Y[z]$ includes the intersection of R_x and R_y,

$$R_w \text{ contains } R_x \cap R_y$$

However, the ROC of $Y[z]$ may be larger, if there is a pole-zero cancellation in the product $X[z] . H[z]$.

Ex: consider the sequences $x[n] = \alpha^n u[n]$ and $h[n] = \delta[n] - \alpha\delta[n-1]$

The z-transform of $x[n]$ is $X[z] = \dfrac{1}{1 - \alpha z^{-1}}$, $|z| > |\alpha|$

The z-transform of $h[n]$ is $H[z] = 1 - \alpha z^{-1}$, $0 < |z|$

However, the z-transform of the convolution of $x[n]$ with $h[n]$ is

$$Y[z] = H[z] . X[z] = \frac{1}{1 - \alpha z^{-1}} . (1 - \alpha z^{-1}) = 1,$$

which due to a pole-zero cancellation is the entire z-plane.

vi) Conjugation

If $X[z]$ is the z-transform of $x[n]$, the z-transform of $x^*[n]$ is

$$x^*[n] \xleftrightarrow{z} X^*[z^*]$$

As a corollary, if $x[n]$ is real-valued, $x[n] = x^*[n]$, then

$$X[z] = X^*[z^*].$$

vii) Derivative

If $X[z]$ is the z-transform of $x[n]$, the z-transform of $n\,x[n]$ is

$$n\,x[n] \xleftrightarrow{z} -z\frac{dX[z]}{dz}$$

Ex: Find the z-transform of $x[n] = n\,\alpha^n u[-n]$

$$\alpha^n u[n] \xleftrightarrow{z} \frac{1}{1-\alpha z^{-1}}, \quad |z| > \alpha$$

$$\left(\frac{1}{\alpha}\right)^n u[n] \xleftrightarrow{z} \frac{1}{1-\alpha^{-1}z^{-1}}, \quad |z| > \frac{1}{\alpha}$$

Using the time-reversal property, we obtain

$$\alpha^n u[-n] \xleftrightarrow{z} \frac{1}{1-\alpha^{-1}z}, \quad |z| < \alpha$$

Using the derivative property, we have

$$-z\frac{d}{dz}\frac{1}{1-\alpha^{-1}z} = -\frac{\alpha^{-1}z}{(1-\alpha^{-1}z)^2}, \quad |z| < \alpha$$

viii) Initial Value Theorem

If $x[n]$ is equal to zero for $n<0$, the initial value $x[0]$, can be found from $X[z]$ as follows

$$x[0] = \lim_{z\to\infty} X[z]$$

This property is a consequence of the fact that if $x[n]=0$ for $n<0$,

$$X[z] = x[0] + x[1]z^{-1} + x[2]z^{-2} + \dots$$

Therefore, if we let $z\to\infty$, each term in $X[z]$ goes to zero except the first.

The inverse z-transform :

In order to recover the sequence $x[n]$ from $X[z]$, we must compute the inverse z-transform. Possible approaches to computing the inverse z-transform are described in what follows.

i) Partial Fraction Expansion

For z-transforms that are rational functions of z :

$$X[z] = \frac{\sum_{k=0}^{q} b[k] \cdot z^{-k}}{\sum_{k=0}^{p} a[k] \cdot z^{-k}} = c \frac{\prod_{k=1}^{q} (1-\beta_k z^{-1})}{\prod_{k=1}^{p} (1-\alpha_k \cdot z^{-1})}$$

a simple and straight forward approaches to find the inverse z-transform is to perform a partial fraction expansion of $X[z]$. Assuming that $p > q$ and that the roots in the denominator are simple, $\alpha_i \neq \alpha_k$ for $i \neq k$, $X[z]$ may be expanded as follows:

$$X[z] = \sum_{k=1}^{p} \frac{A_k}{1-\alpha_k z^{-1}}$$

for some constants A_k for $k = 1, 2, \ldots, p$.

The coefficients A_k may be found by multiplying both sides of the above equation by $(1-\alpha_k z^{-1})$ and setting $z = \alpha_k$. The result is

$$A_k = \left[(1-\alpha_k z^{-1}) \, X[z] \right]_{z=\alpha_k}$$

If $p \leq q$ the partial fraction expansion must include a polynomial in z^{-1} of order $(p-q)$. The coefficients of this polynomial may be found by long division.

For multiple-order poles, the expansion must be modified. For example, if $X[z]$ has a second-order pole at $z = a_k$, the expansion includes

$$\frac{B_1}{1-\alpha_k z^{-1}} + \frac{B_2}{(1-\alpha_k z^{-1})^2} ,$$

where $B_1 = \alpha_k \left[\frac{d}{dz} (1-\alpha_k z^{-1})^2 X[z] \right]_{z=a_k}$ and $B_2 = \left[(1-\alpha_k z^{-1})^2 X[z] \right]_{z=a_k}$

Ex: Consider the sequence $x[n]$ with the z-transform

$$X[z] = \frac{4 - \frac{7}{4} z^{-1} + \frac{1}{4} z^{-2}}{1 - \frac{3}{4} z^{-1} + \frac{1}{8} z^{-2}} = \frac{4 - \frac{7}{4} z^{-1} + \frac{1}{4} z^{-2}}{\left(1 - \frac{1}{2} z^{-1}\right)\left(1 - \frac{1}{4} z^{-1}\right)}$$

with a ROC $|z| > \frac{1}{2}$. Since $p = q = 2$ the two poles are simple and the partial fraction expansion has the form

$$X[z] = c + \frac{A_1}{1 - \frac{1}{2} z^{-1}} + \frac{A_2}{1 - \frac{1}{4} z^{-1}}$$

$$= 2 + \frac{2 - \frac{1}{4} z^{-1}}{\left(1 - \frac{1}{2} z^{-1}\right)\left(1 - \frac{1}{4} z^{-1}\right)}$$

The coefficients A_1 and A_2 are

$$A_1 = \left[\left(1 - \tfrac{1}{2} z^{-1}\right) X[z]\right]_{z^{-1} = 2} = \left.\frac{4 - \frac{7}{4} z^{-1} + \frac{1}{4} z^{-2}}{1 - \frac{1}{4} z^{-1}}\right|_{z^{-1} = 2} = 3$$

$$A_2 = \left[\left(1 - \tfrac{1}{4} z^{-1}\right) X[z]\right]_{z^{-1} = 4} = \left.\frac{4 - \frac{7}{4} z^{-1} + \frac{1}{4} z^{-2}}{1 - \frac{1}{2} z^{-1}}\right|_{z^{-1} = 4} = -1$$

The complete partial fraction expansion becomes

$$X[z] = 2 + \frac{3}{1 - \frac{1}{2} z^{-1}} - \frac{1}{1 - \frac{1}{4} z^{-1}}$$

The ROC is $|z| > \frac{1}{2}$, $x[n]$ is the right-sided sequence

$$x[n] = 2\,\delta[n] + 3\left(\frac{1}{2}\right)^n u[n] - \left(\frac{1}{4}\right)^n u[n]$$

ii) Power series

The z-transform is a power series expansion described by

$$X[z] = \sum_{n=-\infty}^{\infty} x[n]\, z^{-n}$$

$$= \cdots + x[-2] z^{2} + x[-1] z + x[0] + x[1] z^{-1} + x[2] z^{-2} + \cdots,$$

where the sequence values $x[n]$ are the coefficients of z^{-n} in the expansion. Therefore, if we can find the power series expansion of $X[z]$, the sequence values $x[n]$ may be found by simply choosing the coefficients of z^{-n}.

Ex: Consider the z-transform

$$X[z] = \log(1 + a z^{-1}), \quad |z| > |a|$$

The power series expansion of this function is

$$\log(1 + a z^{-1}) = \sum_{m=1}^{\infty} \frac{1}{m} (-1)^{m+1} a^m z^{-m}$$

The sequence $x[n]$ having this z-transform is

$$x[n] = \begin{cases} \frac{1}{n} (-1)^{n+1} a^n, & n > 0 \\ 0, & n \leq 0 \end{cases}$$

iii) Contour Integration

This procedure relies on Cauchy's integral theorem, which states that if c is a closed contour that encircles the origin in a counterclockwise direction:

$$\frac{1}{2\pi j} \oint_c z^{-k} \, dz = \begin{cases} 1, & k = 1 \\ 0, & k \neq 1 \end{cases}$$

With $X[z] = \sum_{m=-\infty}^{\infty} x[m] \cdot z^{-m}$

Cauchy's integral theorem may be used to show that the coefficients $x[n]$ may be found from $X(z)$ as follows:

$$x[n] = \frac{1}{2\pi j} \oint_c X[z] \, z^{n-1} \, dz,$$

where c is a closed contour with the ROC of $X[z]$ that encircles the origin in a counterclockwise direction. Contour integrals of this form may often be evaluated with the help of Cauchy's residue theorem,

$$x[n] = \frac{1}{2\pi j} \oint_c X[z] \cdot z^{n-1} \, dz = \sum [\text{residues of } X[z] \, z^{n-1} \text{ at the poles in } c$$

If $X[z]$ is a rational function of z with a first-order pole at $z = \alpha_k$

$$\text{Res}\left[X[z] \cdot z^{n-1} \text{ at } z = \alpha_k \right] = \left[(1 - \alpha_k z^{-1}) X[z] \cdot z^{n-1} \right]_{z = \alpha_k}$$

Contour integration is useful if only a few values of $x[n]$ are needed.

The one-sided z-transform is defined by

$$X_1[z] = \sum_{n=0}^{\infty} x[n] \cdot z^{-n}$$

The main purpose of the one-sided z-transform is to solve difference equations that have initial conditions. Most of the properties of the one-sided z-transform are the same as those for the two-sided z-transform. An exception is the shift property. Specifically, if $x[n]$ has a one-sided z-transform $X_1[z]$, the one-sided z-transform of $x[n-1]$ is

$$x[n-1] \xleftrightarrow{z} z^{-1} X_1[z] + x[-1]$$

It is this property that makes the one-sided z-transform useful for solving difference equations with initial conditions.

Ex: Consider the difference equation $y[n] = 0.25\, y[n-2] + x[n]$. Find the solution for this equation with $x[n] = \delta[n-1]$, $y[-1] = y[-2] = 1$.

Consider the one-sided z-transform of $y[n]$, i.e., $Y[z]$ and the one-sided z-transform of $y[n-2]$:

$$y[n-2] \xleftrightarrow{z} \sum_{n=0}^{\infty} y[n-2]\, z^{-n} = y[-2] + y[-1] \cdot z^{-1} + \sum_{n=0}^{\infty} y[n] \cdot z^{-n-2}$$

$$= y[-2] + y[-1] \cdot z^{-1} + z^{-2} \cdot Y_1[z]$$

Taking the z-transform of both sides of $y[n] = 0.25\, y[n-2] + x[n]$, we have

$$Y_1[z] = 0.25 \left(y[-2] + y[-1]\, z^{-1} + z^{-2}\, Y_1[z] \right) + X_1[z]$$

where $X_1[z] = z^{-1}$. Substituting for $y[-1]$ and $y[-2]$ and solving for $Y_1[z]$, we have

$$Y_1[z] = \frac{1}{4} \; \frac{1 + 5z^{-1}}{1 - \frac{1}{4} z^{-2}}$$

To find $y[n]$, note that $Y_1[z]$ may be expanded as follows

$$Y_1[z] = \frac{11/8}{1 - \frac{1}{2} z^{-1}} - \frac{9/8}{1 + \frac{1}{2} z^{-1}}$$

Therefore, we have

$$y[n] = \left[\frac{11}{8} \left(\frac{1}{2}\right)^n - \frac{9}{8} \left(-\frac{1}{2}\right)^n \right] u[n]$$