Chapter 9 The Discrete Fourier transform

9.1 Definition

When computing spectra on a computer it is not possible to carry out the integrals involved in the continuous time Fourier transform. Instead a related transform called the *discrete Fourier transform* is used. We shall examine the properties of this transform and its relationship to the continuous time Fourier transform.

The discrete Fourier transform (also known as the finite Fourier transform) relates two finite sequences of length N. Given a sequence with components $x[k]$ for $k = 0, 1, ..., N-1$, the discrete Fourier transform of this sequence is a sequence $X[r]$ for $r = 0, 1, ..., N-1$ defined by

$$X[r] = \frac{1}{N} \sum_{k=0}^{N-1} x[k] \exp\left(-\frac{\mathrm{j}2\pi rk}{N}\right) \tag{9.1}$$

The corresponding inverse transform is

$$x[k] = \sum_{r=0}^{N-1} X[r] \exp\left(\frac{\mathrm{j}2\pi rk}{N}\right) \tag{9.2}$$

Notice that we shall adopt the definition in which there is a factor of $1/N$ in front of the forward transform. Other conventions place this factor in front of the inverse transform (this is used by MatLab) or a factor of $1/\sqrt{N}$ in front of both transforms.

Exercise: Show that these relationships are indeed inverses of each other. It is useful first to establish the relationship

$$\sum_{r=0}^{N-1} \exp\left(\frac{\mathrm{j}2\pi rk}{N}\right) = \begin{cases} N & \text{if } k \text{ is a multiple of } N \\ 0 & \text{otherwise} \end{cases} \tag{9.3}$$

9.2 Discrete Fourier transform of a sampled complex exponential

A common application of the discrete Fourier transform is to find sinusoidal components within a signal. The continuous Fourier transform of $\exp(\mathrm{j}2\pi\nu_0 t)$ is simply a delta function $\delta(\nu - \nu_0)$ at the frequency ν_0. We now calculate the discrete Fourier transform of a sampled complex exponential

$$x[k] = A\exp\left(\mathrm{j}2\pi\nu_0 k\tau\right), \quad \text{for } k = 0, 1, \ldots, N-1 \tag{9.4}$$

The sampling interval is τ and we denote the duration of the entire sampled signal by $T = N\tau$.

Substituting (9.4) into the definition of the discrete Fourier transform (9.1) yields

$$X[r] = \frac{1}{N} \sum_{k=0}^{N-1} A\exp\left(\mathrm{j}\frac{2\pi k}{N}\right)[\nu_0 T - r] \tag{9.5}$$

If $\nu_0 T$ is an integer, i.e., if there are an integer number of cycles in the frame of duration T, we see that

$$X[r] = \begin{cases} A & \text{if } r - \nu_0 T \text{ is a multiple of } N \\ 0 & \text{otherwise} \end{cases} \tag{9.6}$$

In this case, there is only a single non-zero term in the discrete Fourier transform at an index which depends on the frequency ν_0. The value of this non-zero term is A, the complex amplitude of the component.

If $\nu_0 T$ is not an integer however, we find that

$$X[r] = \frac{A}{N} \exp\left(-j\pi(N-1)(r-\nu_0 T)/N\right) \frac{\sin[\pi(r-\nu_0 T)]}{\sin[\pi(r-\nu_0 T)/N]} \tag{9.7}$$

This is nonzero for all values of r. Thus even though there is only a single frequency component in the signal, it can affect all the terms in the discrete Fourier transform.

A plot of the last factor in the equation (9.7) is shown in Figure 9.1. Since r only takes on integer values, the terms in the output sequence are samples from this function taken at unit spacing. Unless the centre of the function at $\nu_0 T$ coincides with an integer, all samples will be non-zero. The sidelobes of the underlying "circular sinc" function cause the single frequency component to be smeared out into adjacent samples. This phenomenon is known as "spectral leakage".

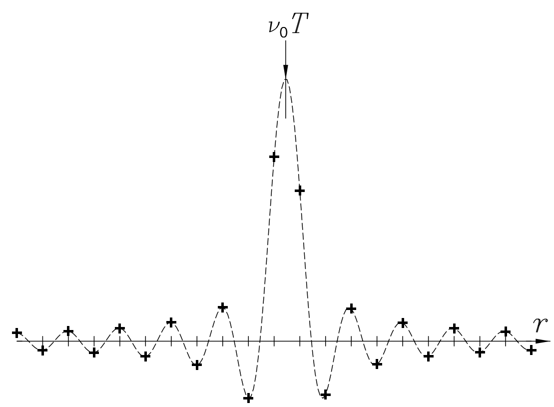

Figure 9.1 Values of discrete Fourier transform of a complex exponential when $\nu_0 T$ is not an integer. Crosses show values of $\sin\left[\pi\left(r-\nu_0 T\right)\right]/\sin\left[\pi\left(r-\nu_0 T\right)/N\right]$ for integer values of r.

Note that if N is large and we are considering those components for which $r - \nu_0 T \ll N$, we can approximate $\sin[\pi(r-\nu_0 T)/N]$ by $\pi(r-\nu_0 T)/N$ and $(N-1)/N \approx 1$ so that (9.7) becomes

$$X[r] \approx A \exp\left(-j\pi(r-\nu_0 T)\right) \operatorname{sinc}(r-\nu_0 T) \tag{9.8}$$

9.3 Sinusoidal and Cosinusoidal components

Suppose that we have

$$x[k] = A\cos\left(\frac{2\pi mk}{N}\right) \quad \text{for } k = 0, 1, \ldots, N-1 \tag{9.9}$$

where m is an integer lying between 1 and $N/2 - 1$ so that there are m cycles within the sequence of length N. Writing this in terms of complex exponentials, we have

$$x[k] = \frac{1}{2}A\exp\left(\frac{\mathrm{j}2\pi mk}{N}\right) + \frac{1}{2}A\exp\left(\frac{-\mathrm{j}2\pi mk}{N}\right) \tag{9.10}$$

Taking the discrete Fourier transform of this leads to two nonzero terms in $X[r]$. The first occurs at $r = m$ and is of amplitude $A/2$ corresponding to the first complex exponential whereas the second is at $r = N - m$ since

$$\begin{aligned} X[N-m] &= \frac{1}{N}\sum_{k=0}^{N-1}\left[\frac{1}{2}A\exp\left(\frac{\mathrm{j}2\pi mk}{N}\right)+\frac{1}{2}A\exp\left(\frac{-\mathrm{j}2\pi mk}{N}\right)\right]\exp\left(\frac{-\mathrm{j}2\pi(N-m)k}{N}\right) \\ &= \frac{A}{2N}\sum_{k=0}^{N-1}\left[\exp\left(\frac{\mathrm{j}2\pi mk}{N}\right)+\exp\left(\frac{-\mathrm{j}2\pi mk}{N}\right)\right]\exp\left(\frac{\mathrm{j}2\pi mk}{N}\right) \\ &= \frac{A}{2N}\sum_{k=0}^{N-1}\left[\exp\left(\frac{\mathrm{j}4\pi mk}{N}\right)+1\right] = \frac{A}{2} \end{aligned} \tag{9.11}$$

Similarly if

$$x[k] = B\sin\left(\frac{2\pi mk}{N}\right) \quad \text{for } k = 0, 1, \ldots, N-1 \tag{9.12}$$

it is easy to show that there are again two nonzero elements in $X[r]$, namely

$$X[r] = \begin{cases} -\mathrm{j}B/2 & \text{if } r = m \\ \mathrm{j}B/2 & \text{if } r = N-m \\ 0 & \text{otherwise} \end{cases} \tag{9.13}$$

We can interpret $X[1]$ through $X[N/2]$ as representing positive frequencies while $X[N/2{+}1]$ through $X[N-1]$ represent negative frequencies.

So if

$$\begin{aligned} f(t) = A_0 + A_1\cos(\omega_0 t) + A_2\cos(2\omega_0 t) + \ldots A_{N/2}\cos(N\omega_0 t/2) + \\ B_1\sin(\omega_0 t) + B_2\sin(2\omega_0 t) + \ldots B_{N/2-1}\cos((N/2-1)\omega_0 t) \end{aligned} \tag{9.14}$$

where $\omega_0 = 2\pi/T$ and $x[k] = f[kT/N]$ for $k = 0, 1, \ldots, N-1$, the discrete Fourier transform $X[r]$ satisfies

$$\begin{aligned} X_0 &= A_0 \\ X_1 &= \frac{1}{2}(A_1 - jB_1) \quad X_{N-1} = \frac{1}{2}(A_1 + jB_1) \\ &\vdots \\ X_{N/2} &= A_{N/2} \end{aligned}$$

9.4 Relationship to the continuous time Fourier transform

There are two distinct but related ways of expressing the discrete Fourier transform to the continuous time Fourier transform. In the first approach, starting with the sequence with terms $x[k]$ for $k = 0, 1, \ldots, N-1$ we define the function

$$x_c(t) = \sum_{k=0}^{N-1} x[k]\delta(t - k\tau) \tag{9.15}$$

where τ is the sampling interval. The continuous time Fourier transform of $x_c(t)$ is given by

$$\begin{aligned} X_c(\nu) &= \int_{-\infty}^{\infty} \sum_{k=0}^{N-1} x[k]\delta(t-k\tau)\exp(-\mathrm{j}2\pi\nu t)\,\mathrm{d}t \\ &= \sum_{k=0}^{N-1} x[k]\exp(-\mathrm{j}2\pi\nu k\tau) \end{aligned} \tag{9.16}$$

$X_c(\nu)$ is periodic with period $1/\tau$. Comparing this with the definition of the discrete Fourier transform we see that

$$X[r] = \frac{1}{N} X_c\left(\frac{r}{N\tau}\right) \tag{9.17}$$

So in this approach the discrete Fourier transform is interpreted as a sampled version of the continuous Fourier transform of $x_c(t)$.

In the alternative view, we consider the continuous time signal which is the infinite periodic extension of $x_c(t)$, i.e.,

$$x_p(t) = \sum_{k=-\infty}^{\infty} x[k \bmod N]\delta(t-k\tau) \tag{9.18}$$

where $k \bmod N$ denotes the non-negative remainder when k is divided by N. Since $x_p(t)$ is the convolution of $x_c(t)$ with $\sum_r \delta(t - rN\tau)$, the Fourier transform of $x_p(t)$ is

$$\begin{aligned} X_p(\nu) &= X_c(\nu) \times \frac{1}{N\tau} \sum_{r=-\infty}^{\infty} \delta\left(\nu - \frac{r}{N\tau}\right) \\ &= \frac{1}{N\tau} \sum_{r=-\infty}^{\infty} X_c\left(\frac{r}{N\tau}\right)\delta\left(\nu - \frac{r}{N\tau}\right) \\ &= \frac{1}{\tau} \sum_{r=-\infty}^{\infty} X[r \bmod N]\delta\left(\nu - \frac{r}{N\tau}\right) \end{aligned} \tag{9.19}$$

In this alternative approach, the terms in the discrete Fourier transform are seen to give the areas of the delta functions in $X_p(\nu)$. This picture has the advantage of symmetry between the time and frequency domains since the functions $x_p(t)$ and $X_p(\nu)$ are each sampled and periodic. In particular

In the time domain, the sampling interval is τ and the function repeats every $N\tau$ In the frequency domain, the sampling interval is $1/(N\tau)$ and the function repeats every $1/\tau$

This reciprocal relationship between the sampling in the time and frequency spaces of the discrete Fourier transform is very useful and should be memorized.

If $x[k]$ originally comes from sampling a continuous signal $x(t)$, i.e., $x[k] = x(k\tau)$, it is easy to relate $x_c(t)$ to $x(t)$ since

$$\begin{aligned} x_c(t) &= \sum_{k=0}^{N-1} x(k\tau)\delta(t-k\tau) \\ &= x(t) \times \Pi\left(\frac{t-(N-1)\tau/2}{N\tau}\right) \times \sum_{k=-\infty}^{\infty} \delta(t-k\tau) \end{aligned}$$

The Fourier transform of this thus involves two convolutions

$$X_c(\nu) = X(\nu) * T\text{sinc}(\nu T)\exp(-j\pi(N-1)\nu T/N) * \frac{1}{\tau}\sum_{k=-\infty}^{\infty}\delta\left(\nu - \frac{k}{\tau}\right) \tag{9.20}$$

where $T = N\tau$. The first convolution is responsible for the spectral leakage whereas the second is responsible for aliasing.

We see that spectral leakage arises from taking the Fourier transform of a rectangular window $\Pi\left(\frac{t-(N-1)\tau/2}{N\tau}\right)$ which multiplies the original function $x(t)$. In the frequency domain, its effect is to convolve the true spectrum with a sinc function which has relatively large sidelobes. It is possible to reduce the spectral leakage considerably by introducing a non-rectangular window function which has a Fourier transform with small sidelobes. For example, the *Hanning window* is defined on $[0, N\tau]$ by

$$h(t) = \begin{cases} 1 - \cos\left(\dfrac{2\pi t}{N\tau}\right) & \text{if } 0 < t < N\tau \\ 0 & \text{otherwise} \end{cases} \tag{9.21}$$

The Fourier transform of the Hanning window and the rectangular window are shown in Figure 9.2. For the purposes of drawing the graph, each window is shifted so that it is an real even function which leads to a real even Fourier transform. The sidelobes of the Hanning window are much smaller than those of the rectangular window although its central peak is somewhat wider. In order to use a Hanning window, it is only necessary to define the sequence $x[k]$ whose discrete Fourier transform we compute by

$$x[k] = h(k\tau)x(k\tau) = (1 - \cos(2\pi k/N))x(k\tau) \tag{9.22}$$

Most digital spectrum analyzers have the option of selecting some form of window to reduce spectral leakage.

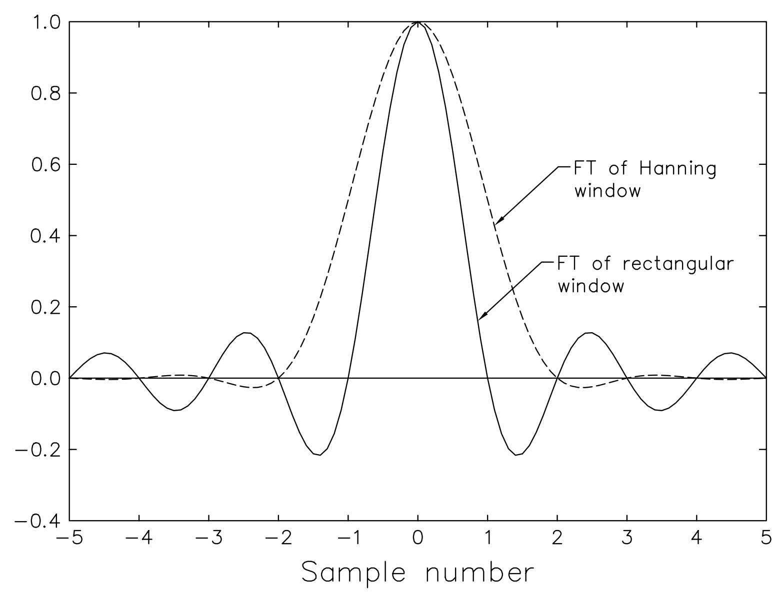

Figure 9.2 Fourier transform of a rectangular window and a Hanning window showing the reduction of spectral leakage with the Hanning window

9.5 The fast Fourier transform (FFT) algorithm

The discrete Fourier transform of a sequence of N points requires $O(N^2)$ arithmetic operations to compute if a straightforward implementation of its definition is carried out. For large N, this can become prohibitive. In 1965, Cooley and Tukey rediscovered (it was previously used by Gauss and Runge) a very efficient way of calculating the discrete Fourier transform which involves $O(N \log N)$ operations. This is known as the *fast Fourier transform* (FFT) algorithm.

Many variants of the FFT algorithm exist. We shall discuss the simplest form known as the *decimation in time* algorithm. The central insight which leads to this algorithm is the realization that a discrete Fourier transform of a sequence of N points can be written in terms of two discrete Fourier transforms of length $N/2$. Thus if N is a power of two, it is possible to recursively apply this decomposition until we are left with discrete Fourier transforms of single points.

Consider an unnormalized discrete Fourier transform of N points which we can write as

$$X[r] = \sum_{k=0}^{N-1} x[k] W_N^{-rk} \tag{9.23}$$

where $W_N = \exp(j2\pi/N)$. The normalization factor $1/N$ can always be applied at the end of the algorithm

We split the sum into terms with even indices and with odd indices yielding

$$X[r] = \sum_{k=0}^{N/2-1} x[2k] W_N^{-2rk} + \sum_{k=0}^{N/2-1} x[2k+1] W_N^{-r(2k+1)} \tag{9.24}$$

Using the fact that $W_N^2 = W_{N/2}$, we may write this as

$$\begin{aligned} X[r] &= \sum_{k=0}^{N/2-1} x[2k] W_{N/2}^{-rk} + W_N^{-r} \sum_{k=0}^{N/2-1} x[2k+1] W_{N/2}^{-rk} \\ &= A[r] + W_N^{-r} B[r] \end{aligned} \tag{9.25}$$

where $A[r]$ is the $N/2$ point Fourier transform of the even terms of x and B_r is the $N/2$ point Fourier transform of the odd terms of x. The above result is valid for $r = 0, 1, ..., N/2 - 1$. In order to obtain the rest of X we note that

$$\begin{aligned} X[r+N/2] &= \sum_{k=0}^{N/2-1} x[2k] W_{N/2}^{-(r+N/2)k} + W_N^{-(r+N/2)} \sum_{k=0}^{N/2-1} x[2k+1] W_{N/2}^{-(r+N/2)k} \\ &= \sum_{k=0}^{N/2-1} x[2k] W_{N/2}^{-rk} - W_N^{-r} \sum_{k=0}^{N/2-1} x[2k+1] W_{N/2}^{-rk} \\ &= A[r] - W_N^{r} B[r] \end{aligned} \tag{9.26}$$

Equations (9.25) and (9.26) express the N point FFT of $x[k]$ in terms of two FFTs of length $N/2$.

Consider the calculation for the situation in which $N = 8$. The diagram in Figure 9.3 is called a "butterfly diagram" and it shows the flow of data through the algorithm. Starting at the right, notice how the terms $X[0]$ through $X[3]$ are calculated from $A[r]$ and $B[r]$ via the relationship (9.25). Similarly, $X[4]$ through $X[7]$ are calculated from $A[r]$ and $B[r]$ via the relationship (9.26).

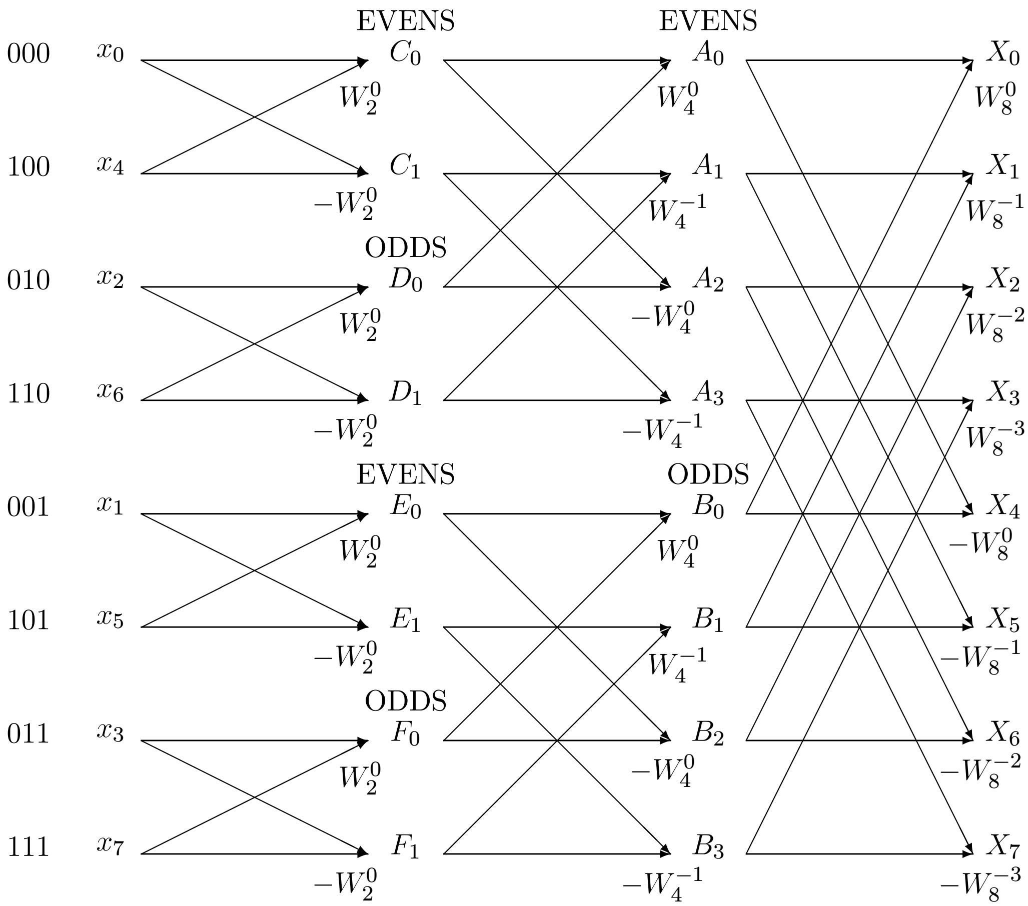

Figure 9.3 Butterfly diagram for $N = 8$ fast Fourier transform

$A[r]$ is the sequence which is the fast Fourier transform of the even terms of $x[k]$ namely $a[k] = \{x[0], x[2], x[4], x[6]\}$. Similarly, $B[r]$ is the sequence which is the fast Fourier transform of the odd terms of $x[k]$ namely $b[k] = \{x[1], x[3], x[5], x[7]\}$.

We can similarly express $A[r]$ and $B[r]$ in terms of fast Fourier transforms of length 2. Moving to the left in the diagram, we have denoted by $C[r]$ and $D[r]$ the two point Fourier transforms that contribute to $A[r]$. We see that $C[r]$ is the Fourier transform of the even points of $a[k]$ which are $a[0] = x[0]$ and $a[2] = x[4]$. These are the "even of evens" of the sequence $x[k]$. Similarly $D[r]$ is the Fourier transform of the odd points of $a[k]$ or the "odds of evens" of the sequence $x[k]$. These are $a[1] = x[2]$ and $a[3] = x[6]$. These two point transforms are finally expressed as sums and differences of the single point transforms which are the original data.

All calculations can be done in-place if the original data is shuffled into the correct order initially. If we look at the binary form of the indices of the data points, it is clear that the data must be arranged in bit-reversed order. The complex exponentials required in the calculation may be precomputed and accessed by table lookup in order to speed up the algorithm.

In the straightforward implementation of the discrete Fourier transform, there are N^2 complex multiplications and N^2 complex additions. In the fast Fourier transform algorithm, there are $\log_2 N$ stages, each of which involves $\frac{1}{2}N$ complex multiplications and N complex additions or subtractions. The operation count (excluding computation of the complex exponentials and the bit-reversed indexing) is thus $\frac{1}{2}N \log_2 N$ complex multiplications and $N \log_2 N$ complex additions. If additions and multiplications take about the same time, the speed up which can be achieved relative to the straightforward implementation is $4N/(3 \log_2 N)$. This is about 137 when $N = 1024$. Larger transforms show even greater factors of improvement.