

11.1: Conversion of Continuous Fourier Transform to DFT

The DFT is commonly encountered when discretizing formulas involving Fourier integrals. Recall the definition of the Fourier transform: given a function $f(x)$, where $x \in (-\infty, \infty)$, the Fourier transform is a function $F(k)$, where $k \in (-\infty, \infty)$, and these two functions are related by a pair of integral formulas:

$$F(k) = \int_{-\infty}^{\infty} dx e^{-ikx} f(x) \quad (11.1.1)$$

$$f(x) = \int_{-\infty}^{\infty} \frac{dk}{2\pi} e^{ikx} F(k). \quad (11.1.2)$$

Typically, a computer simulation or experimental measurement will produce values of $f(x)$ at certain values of x that are discrete and evenly-spaced. Suppose these points are $\{x_0, x_1, \dots, x_{N-1}\}$, where the spacing is $\Delta x = x_{m+1} - x_m$; the corresponding data points are $\{f(x_0), \dots, f(x_{N-1})\}$. We are then interested in finding the Fourier spectrum, i.e. plotting either $|F(k)|$ or $|F(k)|^2$ versus k . To do this, we can approximate the Fourier integral by using the mid-point rule:

$$F(k) \approx \Delta x \sum_{m=0}^{N-1} e^{-ikx_m} f(x_m). \quad (11.1.3)$$

Note that this necessitates a truncation of the Fourier integral. The Fourier integral ran over $-\infty < x < \infty$, but our numerical integral runs over a finite range $x_0 \lesssim x \lesssim x_{N-1}$. This truncation will have important consequences later. Now, we have to decide the values of k at which to find $F(k)$. Let us choose a set of N equally-spaced points,

$$k_n \equiv \frac{2\pi n}{N\Delta x}. \quad (11.1.4)$$

At these points, the discretized Fourier integral takes the form

$$F(k_n) \approx \Delta x \sum_{m=0}^{N-1} \exp\left[-\frac{2\pi i n(x_0 + m\Delta x)}{N\Delta x}\right] f(x_m) \quad (11.1.5)$$

$$= \Delta x \exp\left[-\frac{2\pi i n x_0}{N\Delta x}\right] \sum_{m=0}^{N-1} e^{-i2\pi n m/N} f(x_m) \quad (11.1.6)$$

$$= \Delta x \exp\left[-\frac{2\pi i n x_0}{N\Delta x}\right] \text{DFT}\{f(x_m)\}_n. \quad (11.1.7)$$

Here $\text{DFT}\{f(x_m)\}_n$ denotes the n -th element of the Discrete Fourier Transform (DFT). The m index inside the curly brackets is a dummy index, indicating that the DFT involves an internal sum over this index (we're slightly abusing mathematical notation here). The phase factor, $\exp[-2\pi i n x_0 / N\Delta x]$, is determined by the choice of "origin" for the spatial coordinates; it does not affect $|F(k_n)|^2$ (which is what's used to plot the Fourier spectrum).

The DFT and IDFT can be computed very efficiently, in $O(N \log N)$ time, using an algorithm called the [Fast Fourier Transform \(FFT\)](#). We will not discuss the FFT algorithm in this article, but many good explanations can be found elsewhere online. In Python, you can perform an FFT (fast DFT) by calling `fft`, and an inverse FFT (fast IDFT) by calling `ifft`.

This page titled [11.1: Conversion of Continuous Fourier Transform to DFT](#) is shared under a [CC BY-SA 4.0](#) license and was authored, remixed, and/or curated by [Y. D. Chong](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.