Fourier Transform

Ugenteraan Manogaran

E-mail : m.ugenteraan15@gmail.com

August 23, 2020

Contents

1.0 Problem Statement

Fourier series is great at approximating functions that are periodic. What if the function is not periodic? Let's start with the Complex Fourier series which is defined as

$$f(x) = \sum_{n=-\infty}^{\infty} c_n e^{in\omega x}$$

where

$$c_n = \frac{1}{T}\int_{-\frac{T}{2}}^{\frac{T}{2}} f(x)e^{-in\omega x}\ dx$$

and

$$\omega = \frac{2\pi}{T}$$

for a piece-wise function, $f(t)$ with a period of T.

The trick is to consider a non-periodic function to be periodic in the interval of $\frac{-T}{2} \leq x \leq \frac{T}{2}$ as $T \to \infty$. By incorporating this concept into Fourier series, we can use it to approximate non-periodic functions! The question is, what happens to Fourier series when $T \to \infty$?

2.0 When $T \to \infty$

2.1 Derivation of Fourier Transform

For a finite arbitrary period T, the Fourier series is defined as

$$f(x) = \sum_{n=-\infty}^{\infty} c_n e^{in\omega x} \tag{1}$$

where

$$c_n = \frac{1}{T}\int_{-\frac{T}{2}}^{\frac{T}{2}} f(x)e^{-in\omega x}\ dx \tag{2}$$

and

$$\omega = \frac{2\pi}{T} \tag{3}$$

As $T \to \infty$, $\omega \to 0$. However, since $-\infty < n < \infty$, $n\omega$ becomes continuous. Let

$$\xi = n\omega = \frac{2\pi n}{T}$$

Then the difference between two successive ξ would be

$$\xi_{n+1} - \xi_n = d\xi = \frac{2\pi(dn)}{T} = \frac{2\pi((n+1)-n)}{T} = \frac{2\pi(1)}{T}$$

In other words, $dn = 1$ and as T $\to \infty$, $d\xi \to 0$. Since $dn = 1$, **Eq.** 1 can be rewritten as

$$f(x) = \sum_{n=-\infty}^{\infty} c_n e^{i\xi x}\ dn \tag{4}$$

Notice that dn can also be expressed as

$$dn = \frac{T}{2\pi} d\xi \tag{5}$$

Substituting 5 into 4 yields

$$f(x) = \sum_{n=-\infty}^{\infty} c_n e^{i\xi x} (\frac{T}{2\pi} d\xi) = \sum_{n=-\infty}^{\infty} c_n (\frac{T}{2\pi}) e^{i\xi x}\ d\xi \tag{6}$$

Recall that the definite Riemann's Integral is defined as

$$\int_a^b f(x)\ dx = \lim_{\Delta x \to 0} \sum_{i=1}^{n} f(x_i^*)\ \Delta x_i$$

Since $d\xi \to 0$, 6 can be defined as

$$f(x) = \int_{-\infty}^{\infty} c_n (\frac{T}{2\pi}) e^{i\xi x}\ d\xi \tag{7}$$

Following from 2, $c_n(\frac{T}{2\pi})$ can be simplified as such

$$\begin{aligned} c_n(\frac{T}{2\pi}) &= (\frac{T}{2\pi}) \frac{1}{T} \int_{-\infty}^{\infty} f(x) e^{-i\xi x}\ dx \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} f(x) e^{-i\xi x}\ dx \\ &= \int_{-\infty}^{\infty} f(x) e^{-i2\pi\xi x}\ dx \end{aligned}$$

Notice that this simplified coefficient can also be rewritten as a function of ξ as ξ is the only independent variable in the expression during the integration process in 7. This was not possible previously because ξ was a discrete variable. Now that we found that ξ becomes a continuous variable as $T \to \infty$, the expression can be taken as a function of ξ. Therefore $C(\xi) \equiv c_n(\frac{T}{2\pi})$ and

$$f(x) = \int_{-\infty}^{\infty} C(\xi)e^{i\xi x}\, d\xi \tag{8}$$

where

$$C(\xi) = \int_{-\infty}^{\infty} f(x)e^{-i2\pi\xi x}\, dx \tag{9}$$

2.2 Fourier Transform and Inverse Fourier Transform

Previously we saw that as $T \to \infty$, the coefficient of Fourier series as stated in 2 became a continuous function of ξ in 9. Notice that ξ is the frequency of the wave functions. Therefore, $C(\xi)$ is a function that transforms the original function, $f(x)$ from time-domain to frequency-domain [1]. In other words, given a particular frequency, $C(\xi)$ will output the amplitude[2] at that frequency while $f(x)$ outputs the amplitude at a given time value x.

The function at 9 is known as Fourier Transform (**FT**) and the function at 8 is known as Inverse Fourier Transform (**IFT**). In simple words, Inverse Fourier Transform decomposes the given function into a combination of sine and cosine functions in time-domain while Fourier Transform gives us the amplitudes of those sine and cosine functions in the frequency-domain. Figure 1 below illustrates the process.

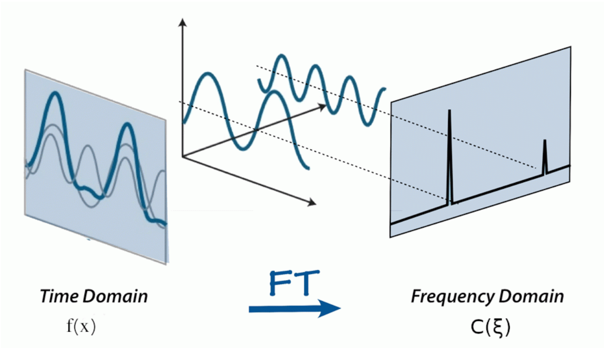

Figure 1 : Fourier Transform Illustration [1]

3.0 Convolution Theorem

One of the most significant contribution of Fourier Transform in the area of signal processing and image processing is the convolution theorem. Convolution is an operation on two functions (real-valued). In image processing, convolution is used as a general-purpose image filter effect for images. Mathematically the convolution operation between $f(t)$ and $g(t)$ is defined as [3]

[1] $C(\xi)$ takes a particular frequency ξ, integrates the function over the entire $-\infty < x < \infty$, and the result of $C(\xi)$ is used as the value of amplitude in 8

[2] Can also be referred to as energy or strength depending on the context.

[3] We used $f(x)$ instead of $f(t)$ because when performing convolution, the parameters would be in reference to that particular kernel ($g(x)$). The kernel will be placed on a particular t and x is the range that the kernel covers.

$$h(t) = \int_{-\infty}^{\infty} g(t-x)f(x)dx \tag{10}$$

Let's see what happens when two functions, $g(t)$ and $f(t)$ is transformed by Fourier Transform and multiplied. **NOTE :** We will be using different variables of integration in the two integrals because we're going to combine the product into and iterated integral.[4]

$$\begin{aligned}
Cg(\xi)Cf(\xi) &= \int_{-\infty}^{\infty} e^{-i2\pi\xi t}g(t)\ dt \int_{-\infty}^{\infty} e^{-i2\pi\xi x}f(x)\ dx \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-i2\pi\xi t}g(t)\ dt\ e^{-i2\pi\xi x}f(x)\ dx \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-i2\pi\xi t}g(t)\ dt\ e^{-i2\pi\xi x}f(x)\ dx \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-i2\pi\xi t}e^{-i2\pi\xi x}g(t)\ dt\ f(x)\ dx \\
&= \int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} e^{-i2\pi\xi(t+x)}g(t)\ dt\right]\ f(x)\ dx
\end{aligned}$$

Let $u = t\ +\ x$. Then, $t = u\ -\ x$ and $du = dt$. Substituting these yields

$$\begin{aligned}
Cg(\xi)Cf(\xi) &= \int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} e^{-i2\pi\xi u}g(u-x)\ du\right]\ f(x)\ dx \\
&= \int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} e^{-i2\pi\xi u}g(u-x)\ du\right]\ f(x)\ dx \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-i2\pi\xi u}g(u-x)f(x)\ dx\ du \\
&= \int_{-\infty}^{\infty} e^{-i2\pi\xi u}\left[\int_{-\infty}^{\infty} g(u-x)f(x)\ dx\right]\ du
\end{aligned}$$

Let $h(u) = \int_{-\infty}^{\infty} g(u-x)f(x)\ dx$. Substituting this yields the Fourier Transform of h

$$\begin{aligned}
Cg(\xi)Cf(\xi) &= \int_{-\infty}^{\infty} e^{-i2\pi\xi u}h(u)\ du \\
&= Ch(\xi)
\end{aligned}$$

[4]In calculus an iterated integral is the result of applying integrals to a function of more than one variable in a way that each of the integrals considers some of the variables as given constants.

By switching the variable for h from $h(u)$ to $h(t)$ (Note that this does not affect any mathematical reasoning.), we can see that

$$h(t) = \int_{-\infty}^{\infty} g(t-x)f(x)\ dx$$

and

$$Ch(\xi) = Cg(\xi)Cf(\xi) \tag{11}$$

In conclusion, a convolution process 10 in the time domain is just a multiplication process in the frequency domain 11. This technique can also be applied to convolutions of any dimensions. For example, in image processing, 2D and 3D convolutions are often used for image filtering purposes. By transforming the image and the convolution kernel to frequency domain, the convolution can be done with just a multiplication before converting it back to time domain.

4.0 Discrete Fourier Transform

The Fourier Transform that we derived in 9 works perfectly for continuous-time signals, many real-life instruments or tools such as digital computers can process a finite number of samples. Therefore, the original Fourier Transform at 9 has to be modified to cater this need. Thus, born the term Discrete Fourier Transform (**DFT**). DFT is defined as

$$C(k) \triangleq \sum_{n=0}^{N-1} x(n)e^{-i2\pi nk/N}, \qquad k = 0, 1, 2, \ldots, N-1 \tag{12}$$

The Inverse Discrete Fourier Transform (**IDFT**) is

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} C(k)e^{i2\pi nk/N}, \qquad n = 0, 1, 2, \ldots, N-1 \tag{13}$$

The definition of DFT follows directly from Fourier Transform. Each data point (or sample), $x(n)$ is treated as an input and the number of data points N is treated as one period. Thus the $\frac{2\pi}{N}$ in the exponents.

5.0 Fast Fourier Transform

In asymptotic limit, calculating DFT and IDFT scales as $\mathcal{O}(N^2)$ because of the two summation terms. Using DFT and IDFT for applications were not ideal due to this slow process. Fortunately, Cooley and Tukey 4 introduced the Fast Fourier Transform (**FFT**) in 1965. The FFT is a recursive approach to DFT that scales as $\mathcal{O}(N\ log\ N)$.

5.1 Symmetries in DFT

Let's evaluate $C(N+k)$ for $k \in \mathbb{R}$.

$$\begin{aligned} C(N+k) &= \sum_{n=0}^{N-1} x(n) \cdot e^{-i2\pi(N+k)n/N} \\ &= \sum_{n=0}^{N-1} x(n) \cdot e^{-i2\pi n} \cdot e^{-i2\pi kn/N} \end{aligned}$$

Euler's Identity states that $e^{-i\pi} = 1$. This equation also holds if for any real-number n, $e^{-in\pi} = 1$. Therefore,

$$C(N+k) = \sum_{n=0}^{N-1} x(n) \cdot e^{-i2\pi kn/N}$$

The final equation that we arrived at shows that $C(N+k) = C(k)$. Thus, itt can seen that the DFT has a symmetrical property.

5.2 From DFT to FFT

From the definition of DFT,

$$\begin{aligned} C(N) &= \sum_{n=0}^{N-1} x(n) \cdot e^{-i2\pi nk/N} \\ &= \sum_{m=0}^{(N/2)-1} x(2m) \cdot e^{-i2\pi(2m)k/N} + \sum_{m=0}^{(N/2)-1} x(2m+1) \cdot e^{-i2\pi(2m+1)k/N} \\ &= \sum_{m=0}^{(N/2)-1} x(2m) \cdot e^{-i2\pi mk/(N/2)} + \sum_{m=0}^{(N/2)-1} x(2m+1) \cdot e^{-i2\pi mk/(N/2)} \cdot e^{-i2\pi k/N} \\ &= \sum_{m=0}^{(N/2)-1} x(2m) \cdot e^{-i2\pi mk/(N/2)} + e^{-i2\pi k/N} \cdot \sum_{m=0}^{(N/2)-1} x(2m+1) \cdot e^{-i2\pi mk/(N/2)} \end{aligned}$$

It can be seen that the DFT can be divided into two parts where each part deals with half of the initial size problem. We can keep dividing it recursively until it is no longer possible. By doing so, we have halved our problem size at every division. This is a common divide and conquer approach in computer science. As stated earlier, this approach scales as $\mathcal{O}(N\ log\ N)$.

6.0 Conclusion and Limitations

To summarise, the **Inverse Fourier Transform**, $f(x)$ and **Fourier Transform**, $C(\xi)$ are

$$f(x) = \int_{-\infty}^{\infty} C(\xi)e^{i\xi x}\ d\xi$$

where

$$C(\xi) = \int_{-\infty}^{\infty} f(x)e^{-i2\pi\xi x}\ dx$$

Inverse Discrete Fourier Transform, $x(n)$ and **Discrete Fourier Transform**, $C(k)$ are

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} C(k)e^{i2\pi nk/N}, \qquad n = 0, 1, 2, \ldots, N-1$$

where

$$C(k) \triangleq \sum_{n=0}^{N-1} x(n)e^{-i2\pi nk/N}, \qquad k = 0, 1, 2, \ldots, N-1$$

The derivation of Fourier Transform from the Fourier Series is not rigorous. There are more rigorous derivations with lesser assumptions than we have made in this note. One such derivation can be found in 6. It is to be noted that both Fourier Series and Fourier Transform are more commonly referred as Fourier Analysis. Jean-Baptiste Joseph Fourier originally invented Fourier Analysis to solve differential equations involving heat transfer. Little did he know that the application of his invention is practically endless in many fields of science.

Fourier Transform converges for any functions provided that the function, $f(x)$

- is $\int_{-\infty}^{\infty} f(x)\ dx < \infty$ **i.e** the function is not infinite over a finite interval and,
- has finite discontinuities

In general, the limitations of Fourier Transform would not create any significant difficulties as most functions in the field of science are well-suited for Fourier Transform.

References

[1] *Fourier transform - AAVOS International. (2020). Retrieved 7 July 2020, from https://aavos.eu/glossary/fourier-transform/*

[2] *Erik Cheever, S. (2020). Linear Physical Systems - Erik Cheever. Retrieved 7 July 2020, from https://lpsa.swarthmore.edu/Fourier/Xforms/FXformIntro.html*

[3] *Osgood, B. (2020). The fourier Transform and its Applications. Retrieved 12 July 2020, from https://see.stanford.edu/materials/lsoftaee261/book-fall-07.pdf*

[4] *Cooley, J. W., Tukey, J. W. (1965). An algorithm for the machine calculation of complex Fourier series. Mathematics of computation, 19(90), 297-301.*

[5] *VanderPlas, J. (2020). Understanding the FFT Algorithm — Pythonic Perambulations. Retrieved 12 July 2020, from https://jakevdp.github.io/blog/2013/08/28/understanding-the-fft/*

[6] *Morin, D. (2020). Fourier Analysis. Retrieved 7 July 2020, from https://scholar.harvard.edu/files/david-morin/files/waves$_f$ourier.pdf*