

PHYS 391 – Fourier Transform Primer

Eric Torrence with modifications by Dean Livelybrooks

1 Introduction

This document is intended to give you just enough information to follow the discussion of the experimental aspects of the discrete Fourier transform and related operations like the power spectrum. This will not be a mathematically rigorous treatment of the Fourier series or integral Fourier transform. A better description can be found in any decent textbook on mathematical methods for physics.

The key concepts for this class are the experimental aspects related to the Nyquist criterion (sampling rates, etc.), aliasing, the need for windowing your data, and the relationship between the amplitude or power spectrum and the raw Fourier transform. Without some mathematical foundation, however, these concepts will be meaningless.

Parts of this write-up were derived from “Mathematical Methods for Physicists” by Arfken, parts of the experimental description were motivated by a similar treatment in “Advanced LabVIEW Labs” by Essick, and parts came directly from Wikipedia.

2 Fourier Series

In 1807, Baron Jean Baptiste Joseph Fourier asserted that any arbitrary function defined over an interval $(-\pi, +\pi)$ could be represented by the series

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos nx + \sum_{n=1}^{\infty} b_n \sin nx \quad (1)$$

where a_n and b_n are the so-called Fourier coefficients for the function $f(x)$. This work came out of studies of heat distributions between two plates, but as a mathematician, natural philosopher, and all-around bright guy (he also is credited with discovering the greenhouse effect in the atmosphere) he realized the broad applicability of this sort of series expansion.

Many of Fourier’s contemporaries, including people as notable as Lagrange himself, thought Fourier was nuts, but with time and a few preconditions it has been shown that Fourier was essentially correct. One of the more important (mild) conditions on $f(x)$ shown by Dirichlet is that the function must only have a finite number of discontinuities, as well as a finite number of minima and maxima in the range $(-\pi, +\pi)$. Note that in particular there is no restriction that $f(x)$ has to be continuous (making the Fourier series much more useful than the Taylor series) nor does $f(x)$ in any way need to be periodic.

2.1 Fourier coefficients

One obvious property of the Fourier series is a direct separability into *even* terms where $f(-x) = f(x)$ defined by the cosine terms with coefficients a_n and *odd* terms where $f(-x) = -f(x)$ defined by the sine terms with coefficients b_n . It follows then that a purely even function will end up being described by non-zero a_n coefficients, while a purely odd function

will end up being described by non-zero b_n coefficients. The $a_0/2$ term, while even, is also special as it represents a constant term giving the average value of $f(x)$ across the interval. For all other terms, the average value across the interval is zero by construction.

The Fourier coefficients a_n, b_n can be determined by the Sturm-Liouville Theory to be

$$a_n = \frac{1}{\pi} \int_{-\pi}^{+\pi} f(x) \cos nx \, dx \quad (2)$$

$$b_n = \frac{1}{\pi} \int_{-\pi}^{+\pi} f(x) \sin nx \, dx. \quad (3)$$

Note that by convention the coefficient b_0 does not exist (or is always zero, depending on your point of view).

2.2 Examples

A trivial example of the Fourier series is the function $f(x) = A \cos 2x$. From inspection, we can see that the coefficient $a_2 = A$ while all other coefficients are zero. We can also see this by direct application of Equation 3 as

$$a_n = \frac{1}{\pi} \int_{-\pi}^{+\pi} A \cos 2x \cos nx \, dx.$$

The determination of coefficients a_n, b_n exploits a property of sinusoidal functions with integer frequencies called *orthogonality*. The integral above integrates to zero for all $n \neq 2$ and gives $a_2 = A$ for $n = 2$. Consider the explicit case of $n = 3$. Figure 1 shows the function $f(x) = \cos 2x \cos 3x$ over the range of the integral, and indeed this does evaluate to zero. Compare this to the case where $n = 2$ shown in Figure 2 and it is clear that the integral of this function will not be zero.

A better example of the application of the Fourier series is the step function described by

$$f(x) = \begin{cases} 0, & -\pi < x < 0, \\ h, & 0 < x < \pi. \end{cases}$$

The constant term can be seen directly to be $a_0 = h$ since the average value of $f(x)$ across the complete interval is $h/2$. Meanwhile it can be shown that the other even terms are all zero as

$$a_n = \frac{1}{\pi} \int_0^{+\pi} h \cos nx \, dx = 0.$$

Note the change in limits of integration since $f(x)$ is only non-zero for values of $0 < x < \pi$. The odd terms can then be found by integration to be

$$b_n = \frac{1}{\pi} \int_0^{+\pi} h \sin nx \, dx = \frac{h}{n\pi} (1 - \cos n\pi),$$

which can be simplified to give the result

$$b_n = \begin{cases} \frac{2h}{n\pi}, & \text{if } n \text{ is odd,} \\ 0, & \text{if } n \text{ is even.} \end{cases} \quad (4)$$

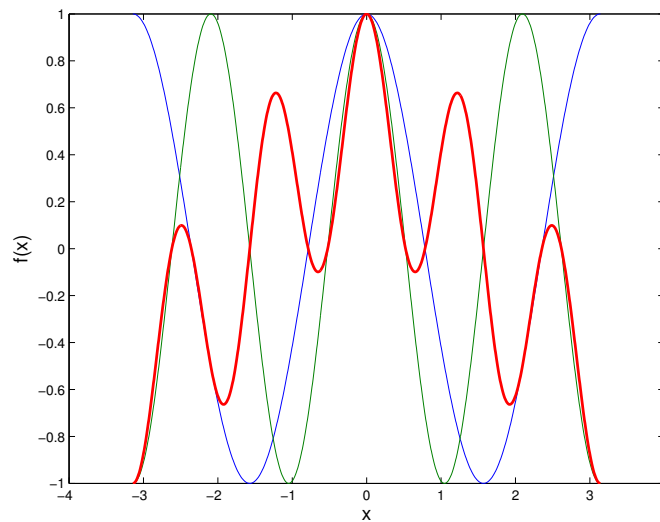


Figure 1: The function $f(x) = \cos 2x \cos 3x$. The integral of this function is zero over the range $-\pi \rightarrow \pi$.

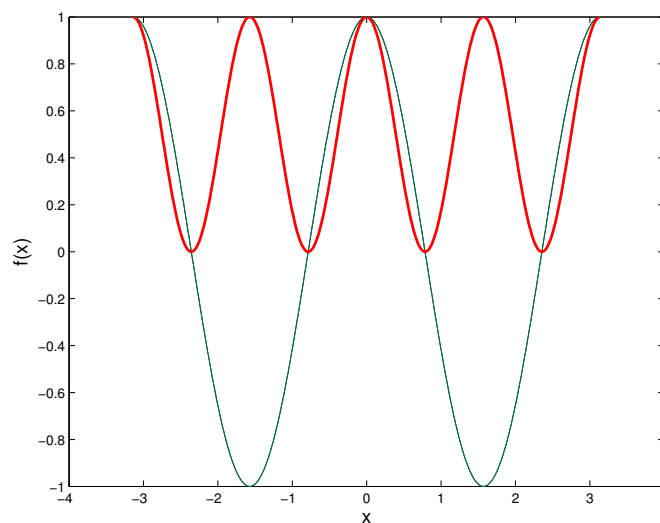


Figure 2: The function $f(x) = \cos^2 2x$. The integral of this function is π over the range $-\pi \rightarrow \pi$.

The complete Fourier series, then, for the step function is

$$f(x) = \frac{h}{2} + \frac{2h}{\pi} \left(\frac{\sin x}{1} + \frac{\sin 3x}{3} + \frac{\sin 5x}{5} + \dots \right). \quad (5)$$

The first four terms of this Fourier series are shown in Figure 3.

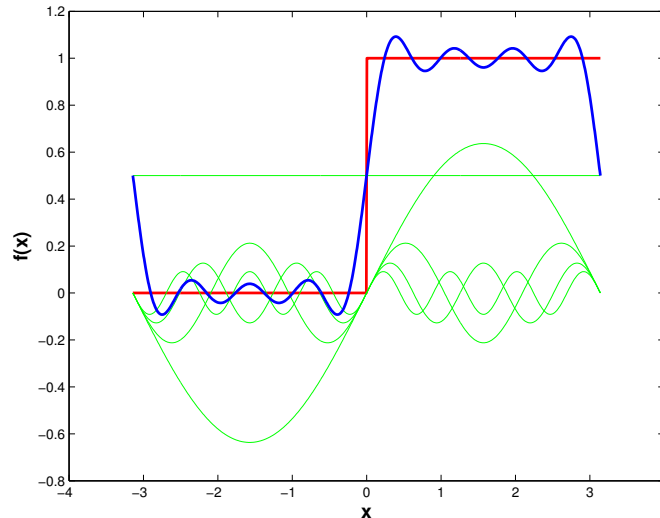


Figure 3: First four terms of the Fourier series describing a step function with $h = 1$.

2.3 Discussion

There are a couple of immediate observations to be made here. A Fourier series is a summation of fixed-frequency sine or cosine functions with argument nx . As such, the coefficients describe the amplitude of the component with natural frequency n . For the step function, the series converges very slowly, which implies that there are significant high-frequency components necessary to accurately describe the function. This is generally true with any discontinuous function, or other function with “sharp” features. A function which is very rounded and sinusoidal converges quickly in the Fourier series, while something like the step function or triangle wave needs many many high-frequency components.

2.4 Exponential Form

An alternative, more compact representation of the Fourier series can be made by using Euler’s formula

$$e^{inx} = \cos nx + i \sin nx. \quad (6)$$

Using this we can rewrite Equation 1 as

$$f(x) = \sum_{n=-\infty}^{\infty} c_n e^{inx}, \quad (7)$$

where we now have complex coefficients c_n with the index n having both positive and negative values. The coefficients can be seen to follow

$$c_n = \begin{cases} \frac{1}{2}(a_n - ib_n), & n > 0, \\ \frac{1}{2}(a_n + ib_n), & n < 0, \end{cases} \quad (8)$$

while $c_0 = \frac{1}{2}a_0$. Clearly the positive and negative coefficients are not independent, but rather are complex conjugates of each other as $c_n = c_{-n}^*$. Now the real part of c_n is related to the even part of the function $f(x)$, while the imaginary part of c_n is related to the odd part.

3 Extended Fourier Series

So far we have only discussed the application of Equation 1 to some arbitrary function on the range $(-\pi, +\pi)$. Now we want to look at extending this to other situations.

3.1 Periodic functions

Nothing in the definition of Equation 1 requires the function $f(x)$ to be periodic. In fact, in our solution for the step function it was never specified what happened to the function beyond the range $(-\pi, +\pi)$. It should be clear, however, that since Equation 1 is formed from sine and cosine functions which are periodic with period 2π , the resulting Fourier series found on the defined range is also periodic outside that range with period 2π . As such, to represent a continuously periodic function, one simply needs to find the Fourier coefficients for a single period, and the solution will automatically also represent the solution over the entire range of x . This technique can clearly *not* be used to represent a function with a period *greater* than 2π , however.

3.2 Change of Interval

To apply equation 1 to a periodic function does require the function $f(x)$ to be periodic over an interval with length 2π . Clearly a simple change of variables can be applied to create instead a transform applicable over a range $2L$. By replacing n with the scaled quantity $\frac{n\pi}{L}$, any range can be accommodated. The Fourier series, then, is written as

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos \frac{n\pi x}{L} + \sum_{n=1}^{\infty} b_n \sin \frac{n\pi x}{L} \quad (9)$$

with coefficients determined by

$$a_n = \frac{1}{L} \int_{-L}^{+L} f(x) \cos \frac{n\pi x}{L} dx \quad (10)$$

$$b_n = \frac{1}{L} \int_{-L}^{+L} f(x) \sin \frac{n\pi x}{L} dx. \quad (11)$$

In this formulation it is clear that the quantity $\frac{n\pi}{L}$ is a measure of angular frequency, which could be a spatial frequency when applied to some function of position $f(x)$ or a temporal frequency when applied to some function of time $f(t)$.

3.3 Change of Basis

While we won't get into this here, it is worth noting that the Fourier series defined with sine and cosine functions is just one example of a Fourier-style expansion. Any complete and orthogonal set of functions can be used to define a similar Fourier-style series. While sines and cosines are convenient for describing plane-wave solutions and physical problems like waves on strings, for many other geometries other functions, such as Bessel functions or Legendre polynomials are more appropriate. There are equivalent Fourier-Bessel and Fourier-Legendre series which are well known, and in general any function can be expanded as a series of any other set of complete, orthogonal functions. This last point is in fact a key tenant of quantum mechanics.

4 Fourier Integral

We have shown how the Fourier series can describe a function bounded on a range $(-L, L)$, or periodic with period $2L$, but what about describing a continuous and non-periodic function such as our favorite Gaussian function $f(x) = e^{-x^2}$? One obvious approach is to use Equation 9 in the limit $\lim_{L \rightarrow \infty} (\frac{n\pi}{L})$. Identifying this as a frequency (spatial or temporal), this limit is equivalent to making the frequency resolution of the Fourier series infinitesimal, or in other words turning the discrete frequency sum of Equation 9 into a continuous frequency integral.

In this limit, the Fourier series becomes the Fourier integral written as

$$f(x) = \int_0^{\infty} [a(\omega) \cos \omega x + b(\omega) \sin \omega x] d\omega, \quad (12)$$

where the Fourier coefficients $a(\omega)$ and $b(\omega)$ are now continuous real functions of the frequency ω . These coefficients are determined by

$$a(\omega) = \frac{1}{\pi} \int_{-\infty}^{+\infty} f(x) \cos \omega x dx \quad (13)$$

$$b(\omega) = \frac{1}{\pi} \int_{-\infty}^{+\infty} f(x) \sin \omega x dx. \quad (14)$$

As before, we could also write this in exponential form leading to a Fourier integral of form

$$f(x) = \int_{-\infty}^{+\infty} c(\omega) e^{i\omega x} d\omega, \quad (15)$$

where again $c(\omega)$ is now a complex function of the frequency parameter ω , related to the real functions $a(\omega)$ and $b(\omega)$ as before.

5 Fourier Transform

The Fourier transform is one of a class of integral transforms which converts a function from one form to another. The Fourier transform is really just a restatement of the results

of the continuous Fourier integral described by Equation 15. More specifically, the Fourier transform is the operation for finding the complex Fourier coefficients in the exponential form. The Fourier transform of the function $f(x)$ is then defined as

$$\hat{f}(\omega) \equiv \int_{-\infty}^{+\infty} f(x)e^{-i\omega x} dx, \quad (16)$$

such that the inverse Fourier transform, which is equivalent to the Fourier integral from last section, is given by

$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \hat{f}(\omega)e^{i\omega x} d\omega. \quad (17)$$

The factor of $\frac{1}{2\pi}$ here is a choice of convention, and some definitions place an equal factor of $\frac{1}{\sqrt{2\pi}}$ in front of both the Fourier and inverse Fourier transforms.

Again the transformed function $\hat{f}(\omega)$ is functionally the same thing as the Fourier coefficients $c(\omega)$ from Equation 15. This function is complex, and again the real part of this function represents the amplitude (as a function of frequency) of the even frequency components while the imaginary part represents the amplitude of the odd frequency components.

In physics, one often applies the Fourier transform to a function of time $f(t)$ to produce the transformed function $\hat{f}(\omega)$ as a function of natural frequency. As no information is gained or lost in this transformation (as can be seen from the ability to recover the original function through the inverse Fourier transform), the functions $f(t)$ and $\hat{f}(\omega)$ are two equivalent ways of looking at the same information. These are typically called the *time domain* and *frequency domain* respectively.

5.1 Fourier properties

While we aren't really going to use much of this, there are a few key mathematical properties of the Fourier transform which are useful to understand.

5.1.1 Linearity

The Fourier transform is linear in the sense that for any complex numbers α and β , the function

$$h(x) = \alpha f(x) + \beta g(x),$$

leads to a Fourier transform given by

$$\hat{h}(\omega) = \alpha \hat{f}(\omega) + \beta \hat{g}(\omega).$$

5.1.2 Translation

For any real number x_0 and function $h(x) = f(x - x_0)$, the Fourier transform is given by

$$\hat{h}(\omega) = e^{-i\omega x_0} \hat{f}(\omega),$$

where $\hat{f}(\omega)$ means the Fourier transform of the function $f(x)$. This shows that the difference between the Fourier transform of a sine or cosine function is simply a constant complex factor (phase shift).

5.1.3 Modulation

For any real number ω_0 and function $h(x) = e^{i\omega_0} f(x)$, the Fourier transform is given by

$$\hat{h}(\omega) = \hat{f}(\omega - \omega_0).$$

This is directly related to the process of amplitude modulation, where a signal function $f(x)$ is multiplied (modulated) by a fixed carrier frequency ω_0 . The result in the frequency spectrum is a variation in amplitude around the base carrier frequency. This is the dual relation to the translation property shown above.

5.1.4 Scaling

For any real non-zero number a the Fourier transform of the function $h(x) = f(ax)$ is given by

$$\hat{h}(\omega) = \frac{1}{|a|} \hat{f}(\omega/a).$$

5.1.5 Differentiation

For the derivative of a function $h(x) = \frac{d^n}{dx^n} f(x)$, the Fourier transform is given by

$$\hat{h}(\omega) = (i\omega)^n \hat{f}(\omega).$$

This result is one of the key useful properties of the Fourier transform, as it allows derivatives in differential equations to be converted via the Fourier transform into a product of factors $i\omega$ and the Fourier transform of the original function.

5.1.6 Convolution

The convolution of two functions $f(x)$ and $g(x)$ is defined as

$$f(x) \otimes g(x) \equiv \int_{-\infty}^{+\infty} f(x-y)g(y)dy,$$

which is often used to describe how detector resolution effects the observed (or recorded) function compared to the true or input function which is typically what we really want to measure.

The Fourier transform of a convoluted function $h(x) = f(x) \otimes g(x)$ is given by

$$\hat{h}(\omega) = \hat{f}(\omega)\hat{g}(\omega),$$

or the simple product of the transformed functions.

Similarly, the dual relationship exists for the product of two functions $h(x) = f(x)g(x)$ such that

$$\hat{h}(\omega) = \frac{1}{2\pi} \hat{f}(\omega) \otimes \hat{g}(\omega).$$

| $f(x)$ | $\hat{f}(\omega)$ |
|--------------------|---|
| $e^{i\omega_0 x}$ | $2\pi\delta(\omega - \omega_0)$ |
| $\cos(\omega_0 x)$ | $\pi[\delta(\omega - \omega_0) + \delta(\omega + \omega_0)]$ |
| $\sin(\omega_0 x)$ | $i\pi[\delta(\omega + \omega_0) - \delta(\omega - \omega_0)]$ |
| $\delta(x)$ | 1 |
| $e^{-\alpha x^2}$ | $\sqrt{\pi/\alpha} e^{-\frac{\omega^2}{4\alpha}}$ |

Table 1: Some common Fourier transforms

5.2 Specific Transforms

It is generally straightforward to evaluate Fourier transforms, but there are a few special functions which are important enough to highlight here.

The first three entries in Table 1 are all variants of the usual interpretation of a Fourier transform operation as picking out the frequency components of an input waveform. For a sinusoidal function with frequency ω_0 the transform $\hat{f}(\omega)$ is non-zero (Delta function) only at the frequencies $\omega = \omega_0$.

The dual relation is the result that the transform of a Delta function (the most extreme version of a “sharp feature”) is a constant, or in other words an infinite number of constant terms across the entire frequency spectrum.

The last result shows another interesting feature of the Gaussian function. Under the proper conditions (parameter $\alpha = 1/2$), the Gaussian function is its own Fourier transform.

One final specific function needs some particular consideration, which is a rectangular function

$$f(ax) = \begin{cases} 1, & |x| < \frac{1}{2a}, \\ 0, & |x| > \frac{1}{2a}, \end{cases} \quad (18)$$

which is a rectangular function centered at the origin with width $1/a$. The Fourier transform can be written as

$$\hat{f}(\omega) = \int_{-1/(2a)}^{+1/(2a)} e^{-i\omega x} dx, \quad (19)$$

which with a little bit of manipulation leads to the result

$$\hat{f}(\omega) = \frac{1}{|a|} \frac{\sin u}{u}, \quad (20)$$

where $u = \omega/(2a)$. This is another example of a function with “sharp features” having a large number of high frequency components in the Fourier transform.

6 Experimental Applications

A common use of the Fourier transform experimentally is to take some waveform, which could be a audible sound signal or a voltage signal with respect to time $V(t)$, and apply the Fourier transform to the raw time-dependent function $f(t)$ to extract information about amplitude as a function of frequency $\hat{f}(\omega)$. For any problem which involves oscillations, this

is typically a much better way to visualize the data than to look at the time series directly. Since we typically measure things as a function of time (although there is no reason we couldn't make equivalent measurements of something as a function of position) I am now going to switch notation and start writing things naturally as a function of time.

Invariably, this experimental Fourier transform involves a first step of digitizing some continuous analog waveform, also known as sampling. It doesn't matter whether this is done by a DVM, an ADC, an oscilloscope, or even a student with a stopwatch, the process of digitizing a continuous function ends up approximating $f(t)$ with the discrete time series $f(t_i)$ where discrete measurements are made at equally-spaced time intervals t_i . The actual value of $f(t)$ also tends to be digitized into discrete values by the fixed-resolution nature of an ADC, but that isn't important for this discussion.

6.1 Nyquist Frequency

One of the first consequences of making a time series of discrete measurements $f(t_i)$ comes from the Nyquist-Shannon sampling theorem. For simplicity it is always assumed that the sampling is done at a constant rate $\nu_s = 1/\Delta t$ known as the sampling frequency. For example, if I take 1000 evenly-spaced samples over a 1 second period, I have a sampling frequency $\nu_s = 1$ kHz. Here I am reverting to the more experimentally-friendly unit of frequency ν which is measured in cycles per second (or Hertz) rather than the theoretically-friendly natural frequency $\omega = 2\pi\nu$.

The sampling theorem says that for a given sampling frequency ν_s , the fastest real signal you can theoretically hope to reconstruct accurately is given by the Nyquist (or critical) frequency

$$\nu_c = \nu_s/2. \tag{21}$$

In other words, if you want to measure a signal with a frequency of 1 kHz, you will need to sample this signal at a rate of *at least* 2 kHz. This is the reason why audio CDs are encoded at a bit rate of 44 kHz. In order to accurately reproduce sounds through the range of human hearing (up to 20 kHz) you need to have a Nyquist frequency which is slightly above 20 kHz, which leads to a sampling frequency at twice the Nyquist or critical frequency.

6.2 Aliasing

An interesting (and potentially dangerous) experimental condition results from undersampling a signal, that is to say trying to measure a frequency ν which violates the Nyquist condition such that $\nu > \nu_s/2$. The best example of this comes from considering a sampling frequency which is just slightly below the actual signal frequency for a sinusoidal signal. If the frequencies were exactly the same, the sampling frequency would always pick a point of constant phase with respect to the signal, and the result would be a constant value every time. If the sampling frequency is slightly less than the signal by an amount $\Delta\nu$, the sample will pick a point on the signal waveform which has been shifted by a small phase $\Delta\phi = 2\pi\Delta\nu\Delta t = 2\pi(\nu_s - \nu)/\nu_s$. The resulting sampled waveform, then, will be a slowly varying function as each successive sample picks a point shifted by $\Delta\phi$ which will have a frequency not given by ν but rather will have an apparent frequency $\nu_s - \nu$. In the extreme limit of $\nu = \nu_s$, the sampled signal will not appear to have any oscillation at all.

This effect is known as *aliasing* and with any discrete sample of data it is impossible to tell the difference between an accurate measurement of a frequency and an aliased measurement of a higher frequency back below the Nyquist frequency. As you will see in your lab, when your signal is below the Nyquist frequency, you accurately measure that frequency. As the signal exceeds the Nyquist frequency, the measured frequency seems to “turn around” and you no longer measure ν but rather a frequency given by $\nu_c - \nu$ which continues to decrease as ν gets larger. As ν approaches ν_s , the apparent frequency goes to zero as described above, then the process repeats itself.

For this reason, any careful experimental measurement of waveforms needs to make sure high frequency components above the Nyquist frequency are filtered out (attenuated) such that they don’t get aliased down below the Nyquist frequency and give erroneous measurements. This necessity for some “headroom” to do this low-pass filtering is exactly why CDs are sampled at 44 kHz and not 40 kHz.

7 Discrete Fourier Transform

Since experimentally we have a discrete time series $f(t_i)$ rather than a continuous function $f(t)$, to perform a Fourier transform we need to replace the integral with a discrete summation known as the Discrete Fourier Transform (DFT):

$$X_k = \sum_{j=0}^{N-1} f(t_j) e^{-i2\pi\nu_k t_j} \Delta t, \quad (22)$$

where now the integer number k identifies the complex discrete Fourier coefficient X_k , while the index j is the summation index over the waveform, each spaced a distance Δt apart.

7.1 Discrete Frequency Index

It is important to understand the similarities and differences between the discrete coefficients labeled by k in Equation 22 and the discrete coefficients labeled by n in Equation 3. Here, we have a discrete time series which, due to the nature of the Nyquist criterion, can only give information on frequencies up a limit of to $\nu_s/2$. In addition, and similar to Equation 3, the finite nature of the interval gives rise to the result that the longest period the Fourier expansion can reproduce is given by the length of the interval. Here, our interval is given by $N\Delta t = N/\nu_s$. Another way of expressing this is that the *smallest* frequency we can resolve is

$$\Delta\nu = \nu_s/N. \quad (23)$$

Clearly we can observe a component with a frequency of $\nu_{k=0} = 0$, but the next frequency component then has a value given by $\nu_1 = \Delta\nu k$ and in general the frequency corresponding to X_k is given by

$$\nu_k = \nu_s k/N. \quad (24)$$

Remembering that in the exponential form we need both positive and negative indexes k , this naturally leads to a range of k running from $-N/2, \dots, -1, 0, +1, \dots, N/2$ and the largest possible frequency being $\nu_s/2$ just as the Nyquist criterion would tell us.

7.2 Frequency Resolution

The Discrete Fourier transform has an important but non-intuitive result buried in the discussion of the frequency index. Because we have sampled a discrete number of points spaced a distance $\Delta t = 1/\nu_s$ apart, there is a minimum frequency difference that we can resolve given by

$$\Delta\nu = \nu_s/N.$$

This can also be thought of as the *frequency resolution* as it is explicitly the smallest change in frequency that we can resolve in our sampled waveform. Rather counter-intuitively, this frequency resolution does not necessarily increase with a higher sampling frequency (remember, increasing frequency resolution corresponds with a *decrease* in $\Delta\nu$), in fact it gets worse, as ν_s goes up if N remains fixed. To maintain a fixed interval of time ($T = N\Delta t = N/\nu_s$), increasing the sampling frequency requires, also, linearly increasing the total number of points (N), and the frequency resolution $\Delta\nu$ will remain unchanged. To increase the frequency resolution (reducing $\Delta\nu$) requires increasing N (total sample time) while keeping ν_s constant, decreasing the sampling frequency ν_s while keeping N constant (which also increases total sample time), or both.

So in other words, as long as your sampling frequency does not violate the Nyquist criterion for the signal frequency you want to measure, the only way to improve the frequency resolution is to sample for a longer time. Increasing the sampling frequency at best does nothing, and if the total number of points N you can store is limited (as is often the case in a buffered ADC) it will actually make the frequency resolution worse. Increasing the sampling frequency is only necessary to make higher frequencies accessible, it does not in any way improve the frequency resolution.

Clearly if you are trying to measure a transient signal (which implies lots of high frequency components) you need a fast sampling frequency. The frequency resolution you can obtain, however, will still be limited by the length of the pulse itself, regardless of your sampling frequency.

7.3 Inverse Discrete Fourier Transform

While not so commonly used, to complete the story the inverse of the DFT operation, which is the equivalent to the original Fourier series, is given by

$$x_j = \sum_{k=0}^{N-1} \frac{X_k}{N} e^{i2\pi\nu_k t_j}, \quad (25)$$

which we can re-write in the more convenient form

$$x_j = \sum_{k=0}^{N-1} A_k e^{i2\pi k j/N}, \quad (26)$$

where we have explicitly expressed the complex amplitude component of frequency ν_k as

$$A_k = X_k/N. \quad (27)$$

It is important to note that in this convention, the coefficients returned by the DFT must explicitly be divided by the total number of points N to obtain a properly normalized amplitude.

7.4 Fast Fourier Transform

The most common term heard in regards to experimental uses of the Fourier transform is the “Fast” Fourier Transform or FFT. The Discrete Fourier transform as written in Equation 22 is very slow to compute numerically, scaling as N^2 . The FFT (of which there are several variants) is a specific algorithm for computing a DFT which uses certain symmetries in the problem (like the fact that $X_k = X_{-k}^*$) to speed up the execution time to at most $N \ln N$. This can be very significant for large data sets. One common feature enforced by most FFT algorithms is that they assume the data sample size is some power of 2 such that $N = 2^n$. If a data sample is specified with N less than an even power of two, the series will be padded with zeroes. This isn’t a big deal, but for maximal computational efficiency it is always good to specify N as some binary power of two.

Some implementations of the FFT also assume that the input data series $f(t_i)$ is real-valued. While there is no requirement for this to be true mathematically, since the FFT is usually used on real experimental data, throwing out the imaginary part also reduces the total amount of data to process by a factor of 2.

8 Using the FFT

Not surprisingly there is an `fft` function built into MATLAB, and also provided in the `numpy.fft` and `scipy.fftpack` libraries for python. All of these functions work in the same way. The input and output of this function are both arrays of numbers which correspond to the operation described in Equation 22. For most people, this is annoyingly far from the desired output of a histogram of amplitude vs. frequency. Here we want to make the connection between the mathematical operation implemented in `fft` and the qualitative understanding of an FFT as a tool for finding frequency components.

First, Equation 22 returns complex coefficients. Sometimes this is useful, but often we are just interested in the total amplitude. One could either separate these values into their real and imaginary components (which would show the even and odd functional components separately) or probably more useful is to simply find the magnitude of each component $|A_k| = \sqrt{A_k A_k^*}$. The absolute value function works as well. Alternately if you were really interested in the relative phase of each frequency, you could find the phase of each component A_k instead. In converting to amplitude, one must remember to include the factor of N as shown in Equation 27, since the MATLAB and python `fft` functions returns X_k directly.

Second, the ordering of the coefficients may not be what you expect. The X_k parameters are complex coefficients which in principle run from $k = -N/2, \dots, -1, 0, 1, \dots, N/2 - 1$ (a total of N output coefficients). Computers don’t tend to like negative indexes, and there are various schemes used to return these values in various implementations. In both the MATLAB and python implementations, the first elements of the returned array corresponds to X_0, X_1, \dots , up to $k = N/2 - 1$ for a total of $N/2$ terms (i.e. the positive terms of X_k

are returned first). When you go beyond the Nyquist limit at $k = N/2$, you next get the $k = -N/2$ term, and continue in increasing k until reaching $k = -1$ as the last array element, for a total of N terms. This sequence is true when N is even. When N is odd, the positive terms increment until $k = (n - 1)/2$ then switch to the first negative term at $k = -(n - 1)/2$. If you plot this data in the order it is returned in the array, you will see the frequency spectrum in the first half of the array reflected in the upper half of the array past the Nyquist limit. The benefit of this arrangement is that you can take the magnitude of the first $N/2$ elements and see the amplitude spectrum, but to use the complete array is difficult.

Third, to make a plot of amplitude vs. frequency, one also needs to know what frequency each amplitude component corresponds to. This is not in the output of the `fft` algorithm, in fact there is nothing in the arguments to the `fft` algorithm which says anything about the Δt or sampling frequency ν_s of the data. One must use Equation 24 to create a vector of frequencies which can then be used as the horizontal axis for a vector of Fourier amplitudes constructed as described above. This is interesting, as it demonstrates that the `fft` function universally converts some equally spaced set of input data into an equally spaced set of output coefficients, with the input Δt and output $\Delta\omega$ being irrelevant to how the `fft` function works.

Fortunately, there are some helper functions to make this easier. The most important is `fftfreq(N, ν_s)` which will return the frequency values corresponding to the coefficients in the `fft` output *in the correct order*. With the output arrays from `fft` and `fftfreq` it is straightforward to get a correct scatter plot of coefficients vs. frequency. A second useful function is `fftshift`. This will change the ordering of both the `fft` output array and the `fftfreq` output array to put the coefficients in increasing k order with the zero frequency term in the middle. For making plots, this isn't really needed, but there are times this can be useful. Doing this re-ordering by hand can be rather error prone.

If you want to plot amplitudes vs. frequency, there are more pitfalls. One has to be very careful to get the correct amplitudes from the repeated $\pm k$ coefficients (which are complex conjugates of each other). The amplitude of a cosine term, for example, will be split evenly between the $k > 0$ and $k < 0$ coefficient. You need to count both of these (or alternately multiply the magnitude of one half of the coefficients by 2) to get the amplitude to properly work out. Annoyingly, this does *not* need to be done for the $k = 0$ term (which only shows up once in the output coefficients). You have to be very careful with your array manipulation if you want a proper amplitude vs. frequency plot ranging from $0 < \nu < \nu_c$ since the constant term only comes out once, as does the coefficient exactly at the Nyquist frequency. Multiplying the two (complex conjugate) halves of the values over the range $k = 1, N/2 - 1$ will give you directly the magnitude squared, which is also equal to the Power spectrum, or take the square root to get the amplitude spectrum. A less error-prone method is to simply truncate the outputs before the Nyquist frequency and take the magnitude of the complex numbers directly, doubling all but the zero-term.

9 Windowing and Leakage

In addition to the problem of aliasing, there is a second experimental aspect to using the FFT which we will explore in our lab which needs to be understood a bit theoretically. This

is the concept of Leakage and the necessity for Windowing your data.

9.1 Leakage

Consider what a FFT actually sees. For a discrete set of time (divided up into N specific points) the function $f(t_i)$ is defined, but outside that range absolutely nothing is known about the function. A finite sampling of a continuous function, then, looks to the FFT operation like a continuous function multiplied by a square rectangle such that outside the sampling window the function is zero.

If we take the FFT of a pure cosine, then, which we would expect to give us a nice sharp peak at the specific frequency of the sine wave, what we actually end up with is the FFT of the function $h(t) = f(t)g(t)$ where $g(t) = \cos 2\pi\nu_0 t$ for example, and $f(t)$ is the rectangular function we saw above

$$f(ax) = \begin{cases} 1, & |x| < \frac{1}{2a}, \\ 0, & |x| > \frac{1}{2a}, \end{cases} \quad (28)$$

We can use one of the Fourier transform rules to simplify $\hat{h}(\nu)$ to the convolution of $\hat{f}(\nu) \otimes \hat{g}(\nu)$. For our pure cosine, we have the result that $\hat{g}(\nu) = \delta(\nu - \nu_0) + \delta(\nu + \nu_0)$ where I have dropped the normalization constant for simplicity, and we must remember that ν can be generally positive or negative. If we only care about the real magnitude, we can restrict ourselves to positive frequencies, and we find just $|\hat{g}(\nu)| = \delta(\nu - \nu_0)$.

The Fourier transform of $g(t)$ meanwhile, is given by the formula found in Equation 20

$$\hat{f}(at) = \frac{1}{|a|} \frac{\sin u}{u},$$

with the identification of $1/a = N/\nu_s$ as the total sample length in time, and $u = 2\pi\nu/(2a) = \pi\nu N/\nu_s = \pi\nu/\Delta\nu$. The convolution of a delta function $\hat{g}(\nu) = \delta(\nu - \nu_0)$ with the function $\hat{f}(\nu)$ just results in shifting the original function by a distance given by ν_0 so that $\hat{h}(\nu) = \hat{f}(\nu - \nu_0)$.

Lets think about this a moment. The $\frac{\sin u}{u}$ function will naturally peak at a frequency ν_0 , and it will have zeroes spaced evenly away from this peak every time u changes by one unit of π , or alternately every time ν changes by $\Delta\nu$. If we had infinite resolution on ν , we would indeed see this long series of frequencies represented by the $\frac{\sin u}{u}$ function (or at least the magnitude thereof). Since we really only *sample* the amplitude as specific frequencies given by ν_k , if we are lucky and ν_0 happens to line up with one of the sampled frequencies ν_k we will get one prominent spike, and all of the other frequencies will line up perfectly with the zeroes of the $\frac{\sin u}{u}$ function (since they are all exactly $\Delta\nu$ away. If we are unlucky (more likely) and ν_0 does not line up with one of the sampled frequencies ν_k , what we see is going to look a lot more like the $\frac{\sin u}{u}$ function.

Another way of thinking about this is that the fact that we always sample a finite window introduces higher order harmonics into our sample whether we like it or not. Just like the Fourier Series, the discrete Fourier transform will be periodic beyond the boundaries of our sampled interval. If there is a discontinuity at that boundary, the coefficients will try to reproduce that discontinuity, leading to higher frequency components. In other words, our sample *always* has some finite window applied to it, and if we do nothing, the edges of our

window are very sharp and lead to lots of higher-order frequency terms. This general feature is called frequency leakage.

9.2 Windowing

As you will see in the lab, left uncontrolled, frequency leakage leads to some very nasty effects. In particular, the total amplitude of a signal is going to remain constant, but if there is frequency leakage, this amplitude will be split amongst all of the frequency components and the apparent height of the central “spike” will vary up and down as the signal frequency moves around. The width of the spike also varies similarly, making it hard to use the FFT to measure either amplitude or width of a signal.

The solution is to apply a different window “by hand” which is much more smooth and doesn’t contain the sharp edges which give rise to the high frequency components in the first place. In practice you are trading off uncontrolled leakage with leakage with well defined and stable characteristics. There are several common windows used, with names like Hamming, Hanning, Blackman, and Gaussian, with each defining a different mathematical function. Without going into a lot of detail here, each has characteristics which makes it better for measuring amplitudes, measuring frequencies, separating frequencies, or measuring spectral widths. The Hamming window is probably the best general purpose window for finding specific frequencies, as it gives a nice sharp spike without too much leakage. The Hanning window (actually named for an Austrian named Hann, but colloquially called Hanning just to confuse graduate students) tends to have less leakage, but in exchange for a broader central spike. For a low amplitude frequency close to a high amplitude frequency, this can be advantageous for resolving the smaller one, which otherwise might get lost in the tail of a Hanning window. The Blackman window takes this to the extreme, having a minimal amount of leakage, but in exchange for a very broad central spike. If there isn’t a necessity of resolving two frequencies close together, however, this can be very useful.

Windowing is important, as you are windowing your data no matter what you do (with a rectangular window, which is probably the worst choice of all). In python, there are standard windows available in the `numpy` library, such as Hanning, Hamming, and Blackman. The function simply takes the number of data points as an argument and returns a properly normalized array of coefficients that can be simply multiplied by the input data array before the `fft` function is called.

The `numpy` documentation has a nice discussion of these windows, including plots of the frequency response, which is very helpful to visualize what the impact of each window will be in frequency space. The Hann window and the frequency response (FFT amplitude) of this function is shown in Figure 4.

10 Final Word

Much of the theory behind Fourier transforms and FFTs is (in my humble opinion) straightforward but not at all intuitive. The actual process of dividing a time series waveform into frequency components is, however, one of the most powerful tools a physicist has at their disposal when dealing with periodic (or even quasi-periodic) data. Obtaining a better intu-

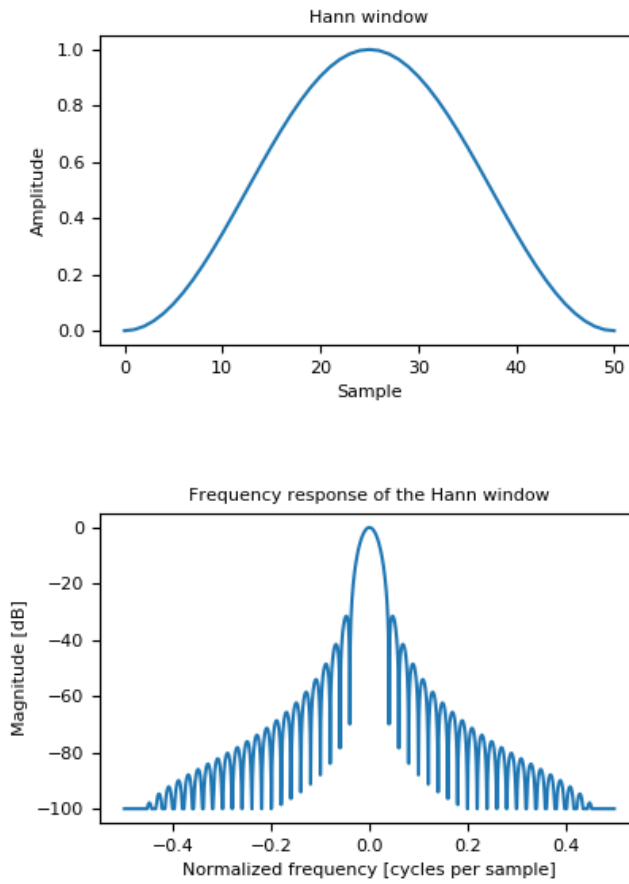


Figure 4: The function form (above) and the frequency response (below) of the Hann window.

itive understanding of the concepts described in this handout is the last goal of this course, and the specific goal of your final lab assignment. The best use of your lab time is to just play with the FFT application and try to make a connection between the theory presented here and the way it actually behaves when looking at a real signal.