CS5275 Lecture 7: The Fourier Transform

Jonathan Scarlett

December 6, 2024

Acknowledgment. The first version of these notes was prepared by Lau Kang Ruey Gregory and Zhang Yang for a CS6235 assignment.

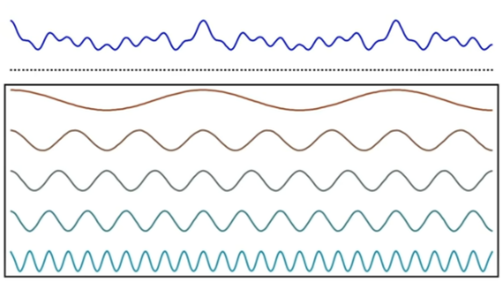
Useful references:

- 3Blue1Brown: `https://www.youtube.com/playlist?list=PL4VT47y1w7A1-T_VIcufa7mCM3XrSA5DD`
- Blog post primers – 'Fourier Analysis' under `https://www.jeremykun.com/primers/`
- Textbook "Principles of Fourier Analysis" by Kenneth B. Howell
- A Very Short Course on Time Series Analysis: `https://bookdown.org/rdpeng/timeseriesbook/`
- *(Not a topic we'll cover but particularly relevant to TCS)* Fourier analysis for analyzing Boolean functions: `https://www.youtube.com/watch?v=lXJP-UkTl-4`

1 Introduction

1.1 Overview

In simple terms, the Fourier transform is a tool that allows us to represent a real-valued function (i.e., a "signal", or an "image" if the coordinates are 2D) as a superposition (sum/integral) of sines and cosines (e.g., $\sin(\omega x)$ for various ω) or complex exponentials (e.g., $e^{i\omega x}$ for various ω) with suitably-chosen amplitudes and frequencies. An example from 3Blue1Brown (the signal at the top is a weighted combination of the signals below):



In some cases, the benefit of such representations is immediate, e.g.:

- If a signal that we are interested has all of its energy a certain frequency range but it gets corrupted by random noise (which typically spans all frequencies), we can immediately mitigate the noise by replacing "noise-only" frequency components by zero.

- In audio applications, we can seamlessly perform operations like "increase the bass level" as well as more complex tasks.
- In telecommunications, different transmitters can transmit their data within different frequency bands to avoid interfering with each other, and the Fourier transform facilitates this.

In other cases, the Fourier transform may be used in a more indirect or abstract manner, e.g.:

- Sinusoids and exponentials frequently arise as solutions to differential equations, and the Fourier transform can be very useful in finding such solutions more easily.
- When computing Ax for some $n \times n$ matrix A and length-n vector x (say), for certain classes of the matrix A (e.g., circulant), the Fourier transform can be used to perform the multiplication in time $O(n \log n)$ instead of $O(n^2)$.
- Fourier transforms are widely used in probability under the name *characteristic functions.*
- A Boolean version of the Fourier transform is widely used in the analysis of Boolean functions.

The practical impact of the Fourier transform is staggering – it is constantly used in our phones, computers, etc., and an algorithm called the 'Fast Fourier Transform (FFT)' may well be one of the most invoked algorithms ever.

1.2 Background on Bases

The Fourier transform can be viewed as a convenient choice of *basis* for (a class of) real-valued functions. Since the notion of basis may be more familiar for vectors rather than functions, it's useful to draw an analogy between the two.

First consider real vectors in Euclidean space, say $\mathbb{R}^d$, with the usual inner product $\langle u, v \rangle = \sum_{i=1}^n u_i v_i$, and note the following:

- The *standard basis* is given by $\{e_i\}_{i=1}^n$, where $e_i = (0, \dots, 0, 1, 0, \dots, 0)$ with the 1 being in the i-th position. The meaning of *basis* is that (i) any $u \in \mathbb{R}^d$ can be written as a linear combination of these, and (ii) none of the basis vectors can be written as a linear combinations of the others.
- Other bases are also possible – for example, in $\mathbb{R}^2$ we could write any vectors as a linear combination of $(0,1)$ and $(1,1)$, or a linear combination of $\big(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\big)$ and $\big(-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\big)$.
- Some bases are more "convenient" than others for certain purposes – for example, for a matrix $A \in \mathbb{R}^{n \times n}$, the eigenvalue decomposition can be viewed as *transforming to a basis in which the operation A becomes coordinate-wise scaling.*
- One particularly useful property a basis $\{v_i\}_{i=1}^n$ can have is *orthonormality*: $\langle v_i, v_j \rangle = 0$ for all i, j (orthogonal) and $\|v_i\| = 1$ for all i (normalized). In this case, we can write any vector $u \in \mathbb{R}^d$ as

  $$u = \sum_{i=1}^n \langle u, v_i \rangle v_i,$$

  with each term $\langle u, v_i \rangle v_i$ being interpreted as the projection of u onto v_i.

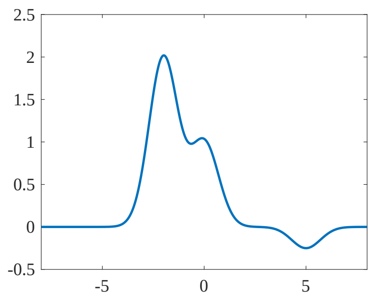
Now we move from vectors to real-valued functions, i.e., $f(x)$ with $x \in \mathbb{R}$. Here x replaces the role of the index $i \in \{1, \dots, n\}$ compared to the vector case. The above considerations naturally generalize as follows, with summations now replaced by integrals:

- We use the following notion of inner product:

$$\langle f, g \rangle = \int_{\mathbb{R}} f(x) g(x) dx,$$

  and the corresponding norm is given by $\|f\| = \sqrt{\langle f, f \rangle} = \sqrt{\int_{\mathbb{R}} f(x)^2 dx}$.

- Like with vectors, we can consider forming more complex functions via additions of simpler ones. The following is an example of a multi-modal function constructed from three Gaussian-like bumps (by combining more and more bumps, we could clearly produce increasingly complicated functions):



- The property of orthonormality is also useful here: The functions $\{g_i\}$ are orthonormal if all $\langle g_i, g_j \rangle = 0$ and $\|g_i\| = 1$, and for any function in the span of the $\{g_i\}$ we can decompose

$$f(x) = \sum_i \langle f, g_i \rangle g_i(x).$$

  (Or if f is not in the span, then the right-hand side represents the *projection of f onto the span.*)

- At first one might think that the space spanned by sines and cosines is limited, but in fact these are rich enough to model *general piecewise continuous functions*. The relevant individual sines/cosines (or complex exponentials) will serve as a convenient orthonormal basis for such functions. (Unlike the vector case, the number of functions in the basis will be infinite, which is why we used the ambiguous notation $\sum_i$ above.)

We conclude this discussion by mentioning that allowing complex values will also be useful: For vectors $u, v \in \mathbb{C}^d$ the inner product is then $\sum_{i=1}^n u_i \overline{v}_i$ (with the bar meaning complex conjugation, $\overline{a + bi} = a - bi$), and the functions f, g the inner product is $\langle f, g \rangle = \int_{\mathbb{R}} f(x) \overline{g(x)} dx$, and the norm is $\|f\| = \sqrt{\int_{\mathbb{R}} |f(x)|^2 dx}$.

2 Fourier Series

Before proceeding, a quick note on terminology:

- The *period* T of a sin wave is the width of a full oscillation. The function $f(x) = \sin\left(\frac{2\pi x}{T}\right)$ has period T.

- The *(ordinary) frequency* ξ is the reciprocal of the period, $\xi = \frac{1}{T}$, giving $f(x) = \sin(2\pi \xi x)$.

- We will work more with the *angular frequency*, defined as $\omega = 2\pi\xi$, so that $f(x) = \sin(\omega x)$. Doing so has various pros and cons (e.g., removing the need to carry around 2π, but it does lead to having some "nuisance terms" later on).

In this section, we will consider functions produced by adding sines/cosines whose frequencies are *integer multiples* of some base frequency (e.g., $\sin(x)$, $\sin(2x)$, $\sin(3x)$, ...). This turns out to "only" be powerful enough to produce signals that have a regularly repeating pattern (periodicity), though that pattern itself can be highly complex. In Section 3 we will extend this idea to allow a continuum of frequencies (e.g., $\sin(\omega x)$ for arbitrary choices of ω), which will let us produce general functions that need not be periodic.

2.1 Using Trigonometric Functions (Sines and Cosines)

We first consider the space $\mathbb{S}$ of piecewise continuous functions $f : [-\pi, \pi] \to \mathbb{R}$. The specific interval $[-\pi, \pi]$ is considered for convenience, but this can be generalized to any finite interval $[a, b]$ We will also view these functions as being defined on the entire real line $\mathbb{R}$, where outside $[-\pi, \pi]$ the function is defined by having periodic behavior with period 2π. Since we are allowing complex values, we define the inner product for $\mathbb{S}$ as

$$\langle f, g \rangle = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x)g(x)dx.$$

Notice the restriction to $[-\pi, \pi]$), and the division by π which is related to our use of angular frequency.

The Fourier Series is based on using the following basis for the space $\mathbb{S}$:

$$\left\{ \frac{1}{\sqrt{2}}, \sin(x), \cos(x), \sin(2x), \cos(2x), ... \right\}. \tag{1}$$

We will omit proof that this is a basis, but make some comments on it in Appendix B. The orthonormality property can fairly easily be verified by direct integration.

Writing the basis elements in Eq. (1)as $\{e_n\}_{n=1}^{\infty}$, we can then decompose any function $f \in \mathbb{S}$ as:

$$f = \sum_{n=1}^{\infty} \langle f, e_n \rangle e_n, \tag{2}$$

or substituting the functions explicitly:

$$f = \frac{a_0}{2} + \sum_{n=1}^{\infty} \big(a_n \cos(nx) + b_n \sin(nx) \big), \tag{3}$$

where for $n = 1, 2, ...$

$$a_0 = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x)dx$$
$$a_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x)\cos(nx)dx$$
$$b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x)\sin(nx)dx.$$

Equation (3) is called the **Fourier series** expansion of f, and the terms a_n, b_n are called *Fourier coefficients.*

We can simplify this decomposition for certain classes of f. For example,

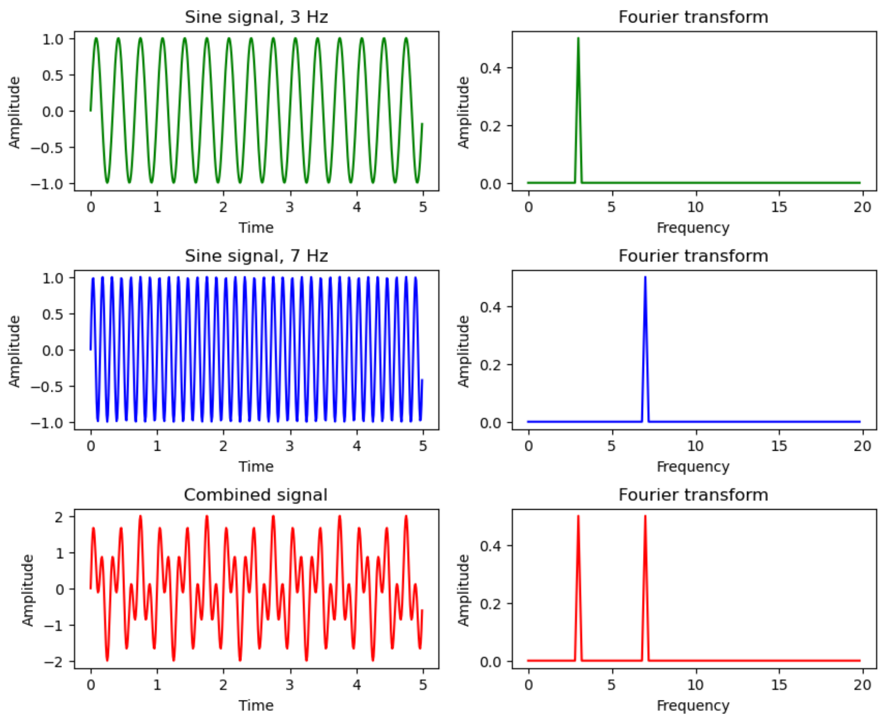


Figure 1: Illustration of two sine wave signals with different frequencies and their combined signal (first column), along with their Fourier transforms or decomposition into frequencies (second column).

- For an even function f where $f(-x) = f(x)$, one can show that the b_n terms are all zero.
- Similarly, for an odd function f where $f(-x) = -f(x)$, one can show that the a_n terms are all zero.

We see that the Fourier decomposition can be viewed as a decomposition of a function f into a set of frequencies, mapped by the trigonometric basis functions. Some examples are shown in Fig. 1, where we see that signal comprising two sine signals with different frequencies can be readily decomposed into distinct frequencies. In general, functions that appear less 'smooth' with more fluctuations at smaller scales will tend to have larger high frequency coefficients (i.e., a_n and b_n for large n).

2.2 Using Complex Exponentials

The previous section had only considered real-valued periodic functions, and used sines/cosines in the choice of basis. Here we generalize to consider complex-valued functions, and adopt a basis based on complex exponentials, which perhaps surprisingly comes out to be *neater* and more compact to write. Since we now allow complex values, the preceding choice of inner product generalizes to

$$\langle f, g \rangle = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) g(x) dx.$$

(Now we divide by 2π instead of π, which is related to complex numbers having 2 components.)

Similar to Eq. (1), we can introduce the following infinite set of functions:

$$\{1, e^{ix}, e^{-ix}, e^{i2x}, e^{-i2x}, ...\}. \tag{4}$$

We can again easily verify that these complex exponentials are orthogonal to one another by computing $\langle \psi_j, \psi_k \rangle = \int_{-\pi}^{\pi} e^{ijx} e^{-ikx} dx$ for each pair of functions (ψ_j, ψ_k) in the set. Similarly, we can verify that they have unit norm by writing $\|\psi\| = \sqrt{\langle \psi, \psi \rangle}$.

Then, the complex exponential Fourier series can be expressed as follows:

$$f(x) = \sum_{n=-\infty}^{\infty} c_n e^{inx}, \tag{5}$$

where for $n = 0, \pm 1, \pm 2, ...,$

$$c_n := \hat{f}_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) e^{-inx} dx. \tag{6}$$

Using Euler's formula $e^{i\theta} = \cos(\theta) + i \sin(\theta)$, we can relate the complex exponential Fourier series with that of the trigonometric Fourier series. Specifically, doing so gives the following:

- $c_n = \frac{a_n - ib_n}{2}$, $c_{-n} = \frac{a_n + ib_n}{2}$
- $a_n = c_n + c_{-n}$, $b_n = i(c_n - c_{-n})$
- Hence the symmetry properties: When f is an even function, we have $c_k = c_{-k}$ for $k = 1, 2, 3, ...,$ and similarly when f is an odd function, we have $c_k = -c_k$ for $k = 1, 2, 3...,$ zeroing out the odd sine and even cosine components accordingly.

2.3 Note on Finite-Term Approximation

In general, if we wanted to actually store the Fourier series of a function on a computer, we would have to truncate to a finite number of terms. Upon doing this, one might hope that a finite-term approximation of $f(x) = \sum_{n=-\infty}^{\infty} c_n e^{inx}$ is sufficient:

$$f(x) \overset{?}{\approx} f(x) = \sum_{n=-N}^{N} c_n e^{inx}$$

for suitably chosen N. It turns out that the approximation error is indeed very small for sufficiently "well-behaved" functions (e.g., ones with relatively smooth behavior and no abrupt changes).

On the other extreme, for discontinuous functions an important (and unfortunate) phenomenon known as the *Gibbs phenomenon* is observed, even when N is very large. The issue is that the basis functions (sines and cosines) are all continuous, so any finite combination of them will also be continuous. As a result, when we truncate the infinite series, we get a "ringing" pattern causing large approximation errors near the discontinuity; see Figure 2 for an example using the square wave. By comparison, further away from the discontinuities, the approximation error is very small.

2.4 Parseval's Theorem and Energy Interpretation

The following result is known as the *Parseval's theorem* (or sometimes *Rayleigh's identity*):

$$\|f\|^2 = \sum_{n=-\infty}^{\infty} |\hat{f}_n|^2 = \sum_{n=-\infty}^{\infty} |\langle f, e^{inx} \rangle|^2 \tag{7}$$

This roughly states that *the energy (i.e., sum or integral of the squared values) of a signal is the same as the energy of its Fourier transform.*

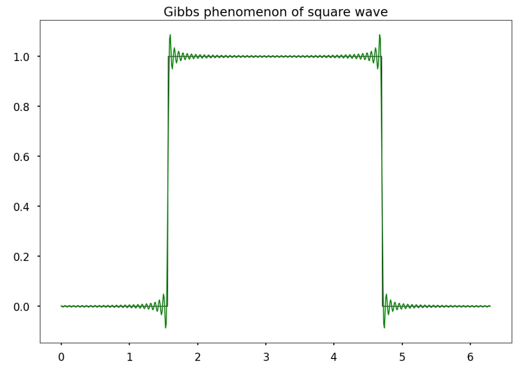


Figure 2: Illustration of Gibbs phenomenon for a square wave. Notice the "ringing" pattern near the discontinuities.

To show this property, letting e_n denote the n-th Fourier basis function, we have

$$\begin{aligned}
\|f\|^2 &= \langle f, f \rangle \qquad \text{(definition of norm)} \\
&= \left\langle \sum_{n=-\infty}^{\infty} \langle f, e_n \rangle \, e_n, \sum_{m=-\infty}^{\infty} \langle f, e_m \rangle \, e_m \right\rangle \quad \text{(Fourier expansion of } f\text{)} \\
&= \sum_{n,m=\infty}^{\infty} \langle f, e_n \rangle \overline{\langle f, e_m \rangle} \langle e_n, e_m \rangle \quad \text{(linearity of inner product)} \\
&= \sum_{n=\infty}^{\infty} \langle f, e_n \rangle \overline{\langle f, e_n \rangle} \quad \text{(orthonormality of basis)} \\
&= \sum_{n=-\infty}^{\infty} |\left\langle f, e^{inx} \right\rangle|^2. \quad \text{(since } a\overline{a} = |a|^2\text{)}
\end{aligned} \tag{8}$$

Note also that in view of this property, we can interpret $|\left\langle f, e^{inx} \right\rangle|^2$ as the amount of energy associated with the n^{th} frequency.

3 Fourier Transform

Thus far we have only looked at periodic functions (or alternatively, functions restricted to a finite interval, $[-\pi, \pi]$). In this section, we move to general functions from $\mathbb{R} \to \mathbb{C}$, using the periodic case as a building block. Specifically, we consider a general function $f : \mathbb{R} \to \mathbb{C}$ having the following two properties: (i) it is piecewise continuous on every finite interval, and (ii) it is absolutely integrable, i.e., $\int_{-\infty}^{\infty} |f(x)| dx < \infty$. (In fact, often we will not really need (ii) to hold, e.g., we will still talk about the Fourier transform of a sinusoid.)

The idea will be to apply the Fourier series to the function restricted to a finite range $[-L, L]$ (for some $L > 0$), and then consider what happens when $L \to \infty$. We start with the following observations:

- For any $L > 0$, by generalizing the case that $L = \pi$ (handled above), we can derive the Fourier series $f(x) = \sum_{n=-\infty}^{\infty} c_{n,L} e^{i\omega_n x}$, where $\omega_n = n\pi/L$ and $c_{n,L} = \frac{1}{2L} \int_{-L}^{L} f(x) e^{-i\omega_n x} dx$.
- Observe that each frequency is a multiple of π/L. We thus define $\Delta_{\omega,L} = \pi/L$, which gives $\omega_n = n\Delta_{\omega,L}$,

and the Fourier coefficient can be expressed as

$$c_{n,L} = \Delta_{\omega,L} \cdot \frac{1}{2\pi} \cdot \underbrace{\int_{-L}^{L} f(x) e^{-i\omega_n x} dx}_{\hat{f}_L(\omega_n)} . \tag{9}$$

Now consider the limiting case where $L \to \infty$. Then, the integration over $[-L, L]$ approaches integration over the whole real line,[1] and we also have $\Delta_{\omega,L} \to 0$. This means that the summation defining $f(x)$ approaches an integral:

$$f(x) = \lim_{L\to\infty} \sum_{n=-\infty}^{\infty} c_{n,L} e^{i\omega_n x} \tag{10}$$

$$= \lim_{L\to\infty} \frac{1}{2\pi} \sum_{n=-\infty}^{\infty} \hat{f}_L(\omega_n) e^{i\omega_n x} \Delta_{\omega,L} \tag{11}$$

$$= \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{f}(\omega) e^{i\omega x} d\omega, \tag{12}$$

where $\hat{f}(\omega) = \int_{-\infty}^{\infty} f(x) e^{-i\omega x} dx$.

The above analysis leads us to define to **Fourier Transform** and **Inverse Fourier Transform** as follows:

$$\hat{f}(\omega) = \mathcal{F}(f(x)) = \int_{-\infty}^{\infty} f(x) e^{-i\omega x} dx \tag{13}$$

$$f(x) = \mathcal{F}^{-1}(\hat{f}(\omega)) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{f}(\omega) e^{i\omega x} d\omega. \tag{14}$$

This pair of operators are the counterpart to the Fourier series for non-periodic functions. The transform $\mathcal{F}(\cdot)$ can still be viewed as a change of basis from a given "natural basis" to the "frequency basis". Notice that now *all* frequency values $\omega \in \mathbb{R}$ play a role, whereas for the Fourier series it was only integer multiples of a base frequency. Accordingly, we now get $f(x)$ by integrating over all ω, rather than summing over integers n.

As we will see soon, there are several properties in the frequency domain that can be exploited to be able to solve problems that would be challenging in the original domain.

Note on ordinary frequency vs. angular frequency. The quantity ω above is the *angular frequency*, and is related to the *ordinary frequency* ξ via $\omega = 2\pi\xi$. The distinction is just a minor matter of normalization, but it's useful to be aware of the three main variations of the Fourier transform:

- The one above is called non-unitary with angular frequency.
- The unitary version with angular frequency puts a factor $\frac{1}{\sqrt{2\pi}}$ in the expressions for $\hat{f}$ and f, rather than putting the entire 2π in the expression for f. This makes the equations "more symmetric" but can lead to a lot of "niusance" $\frac{1}{\sqrt{2\pi}}$ factors.
- The version with ordinary frequency puts 2π in the exponent instead, which turns out the remove the need for any pre-factor: $\hat{f}(\xi) = \int_{-\infty}^{\infty} f(x) e^{-i2\pi\xi x} dx$ and $f(x) = \int_{-\infty}^{\infty} \hat{f}(\xi) e^{i2\pi\xi x} d\xi$.

Very brief note on multi-input functions: The Fourier transform can also be defined for functions with inputs in $\mathbb{R}^d$, with $\mathbb{R}^2$ being particularly important in image processing. We will stick to single-input functions in this lecture, but the main properties generally carry over naturally.

[1] This is formally justified by the condition $\int_{-\infty}^{\infty} |f(x)| dx < \infty$, which we mentioned above.

4 Examples and Properties

Here we overview some examples and main properties of the Fourier transform – see the tables are `https://en.wikipedia.org/wiki/Fourier_transform#Tables_of_important_Fourier_transforms` for a useful summary and a more detailed list.

Before proceeding, we need to introduce a "strange but useful" notion called the *Dirac delta function* $\delta(x)$, which satisfies the following equality:

$$\delta(x) = \begin{cases} \infty & x = 0 \\ 0 & x \neq 0. \end{cases}$$

To make this more precise, we impose the following "definition" or "convention":

$$\int_{-\infty}^{\infty} \delta(x)\, dx = 1. \tag{15}$$

We can think of it in the following way: Imagine a rectangle-shaped function whose value is $\frac{1}{w}$ in the interval $[-w/2, w/2]$ for some $w > 0$, and 0 elsewhere. This is a perfectly standard function that we could work with in the usual way. The Dirac delta function captures how such a function behaves *in the limit as* $w \to 0$ – thus, it is an infinitely narrow and infinitely high "spike" but it still integrates to one.

For any function $f(x)$, it follows from (15) that

$$\int_{-\infty}^{\infty} \delta(x-a) f(x)\, dx = f(a). \tag{16}$$

The Dirac delta function is useful when studying Fourier transforms (and also for Linear Time Invariant systems, which we won't cover in detail), as two of the examples below demonstrate.

4.1 Examples

Some sketches of common functions and their Fourier transform pairs are given in Fig. 3. We give the details of one example (the rectangular function); the interested student may want to try some of the others (using Eq. (13) and Eq. (14)) and/or consult the Wikipedia page for a longer list of examples.

Consider the rectangular function

$$f(x) = \begin{cases} 1 & |x| \leq \frac{1}{2} \\ 0 & \text{otherwise.} \end{cases}$$

The Fourier transform is given by

$$\begin{aligned} \hat{f}(\omega) &= \int_{-1/2}^{1/2} e^{-i\omega x} dx \\ &= \left[\frac{1}{-i\omega} e^{-i\omega x} \right]_{-1/2}^{1/2} \\ &= \frac{e^{-i\omega/2} - e^{i\omega x/2}}{-i\omega} \\ &= \frac{\sin(\omega/2)}{\omega/2}, \end{aligned}$$

where the last line uses $\sin x = \frac{e^x - e^{-x}}{2i}$ (from Euler's identity). Then, introducing the notation $\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x}$

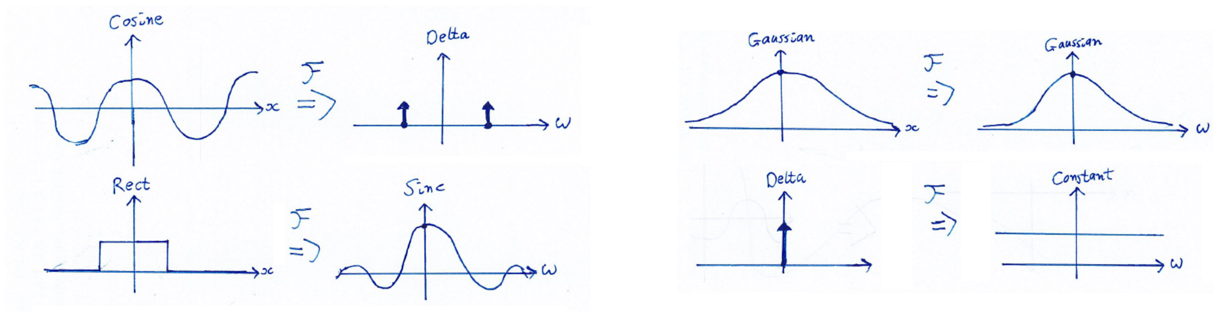


Figure 3: Examples of common functions and their Fourier transform pairs.

(the *sinc function*), we get

$$\hat{f}(\omega) = \text{sinc}\Big(\frac{\omega}{2\pi}\Big).$$

4.2 Linearity

It is straightforward to see that the Fourier transform is a linear operator, i.e., $\mathcal{F}[af(x) + bg(x)] = a\mathcal{F}[f(x)] + b\mathcal{F}[g(x)]$, given that the integral operators are linear. This applies similarly for the inverse Fourier transform.

4.3 Shifting Properties

The following properties are fairly straightforward to prove using the definition of the (inverse) Fourier transform and standard integration techniques like change of variable and/or $e^{a+b} = e^a e^b$: If $f(x)$ has Fourier transform $\hat{f}(\omega)$, then

- For any constant a, the function $f(ax)$ has Fourier transform $\frac{1}{|a|}\hat{f}(\omega/a)$. This means that when a function gets narrower in the original domain (i.e., $a > 1$) it gets broader in the frequency domain, and vice versa. This is intuitive, because making a function narrower (while otherwise maintaining the same shape) means that any function increases/decreases get sharper/steeper, and faster changes correspond to higher frequencies.
- For any constant c, the function $f(x - c)$ has Fourier transform $\hat{f}(x)e^{-ic\omega}$. Similarly, the function $f(x)e^{icx}$ has Fourier transform $\hat{f}(\omega - c)$. That is, a shift in one domain translates to multiplication by a complex exponential in the other.

4.4 Convolution Properties

We can also consider the convolution operation between two functions f and g :

$$(f * g)(x) = \int_{-\infty}^{\infty} f(x - y)g(y)dy. \tag{17}$$

Intuitively, we can view convolution as a kind of smoothing or averaging operation. It is a very important concept in both theory and practice; for example:

- We will briefly discuss how it is used in the context of understanding linear time-invariant (LTI) systems in Section 6.2.

- The Fourier transform applied to a probability density function is known as the *characteristic function*. In this context, the relevance of convolution is that if X, Y are independent with density functions f_X, f_Y, then the random variable $Z = X + Y$ has density function $f_Z = f_X * f_Y$.

A very useful property of the Fourier transform is that *convolution in the original domain corresponds to multiplication in the Fourier domain*:

$$\mathcal{F}(f * g) = \mathcal{F}(f)\mathcal{F}(g). \tag{18}$$

That is, the Fourier transform of $(f * g)(x)$ is $\hat{f}(\omega)\hat{g}(\omega)$.

We can see this by considering the inverse Fourier transform of $\hat{f}\hat{g}$:

$$\begin{aligned}
\mathcal{F}^{-1}[\hat{f}\hat{g}](x) &= \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{f}(\omega)\hat{g}(\omega)e^{i\omega x}d\omega \qquad \text{(by definition of } \mathcal{F}^{-1}\text{)} \\
&= \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{f}(\omega)\int_{-\infty}^{\infty} g(y)e^{-i\omega y}dy\ e^{i\omega x}d\omega \quad \text{(by definition of } \hat{g}\text{)} \\
&= \int_{-\infty}^{\infty} g(y)\underbrace{\frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{f}(\omega)e^{i\omega(x-y)}d\omega}_{f(x-y)}dy \quad \text{(by swapping order of integrals)} \qquad (19) \\
&= \int_{-\infty}^{\infty} g(y)f(x-y)dy \quad \text{(by formula for inverse FT)} \\
&= f * g. \qquad (20)
\end{aligned}$$

4.5 Fourier Transform vs. its Inverse

Comparing the formula for the Fourier transform and its inverse in (13) and (14), we see that the two are nearly identical – the only differences are (i) the $\frac{1}{2\pi}$ factor (this difference even disappears in the other version of the Fourier transform discussed after (14)), and (ii) the use of $-i$ vs. i in the exponent. In this sense, the Fourier transform and inverse Fourier transform are "essentially" doing the same thing.

For this reason, when we see a property in the original domain implying some other property in the frequency domain, the role of the domains can easily be reversed. For example:

- Convolution in the original domain translates to multiplication in frequency domain;
- Conversely, multiplication in the original domain translates to convolution in the frequency domain.

(Care may just be needed in getting certain signs and scaling factors correct.)

4.6 Derivative Properties

How we consider how the Fourier transform of a differentiable function $f(x)$ relates to the Fourier transform of its derivative $f'(x)$. This turns out to be very useful for solving differential equations; see Section 6.3 for an example.

Technically we need certain smoothness/continuity and absolutely integrable conditions, but we will not go into the details of those, and instead simply state that we can get the following using integration by parts:

$$\begin{aligned}
\mathcal{F}(f'(x)) &= \int_{-\infty}^{\infty} \underbrace{f'(x)}_{dv} \underbrace{e^{-i\omega x}}_{u} dx \\
&= \underbrace{\left[f(x)e^{-i\omega x}\right]_{-\infty}^{\infty}}_{0} - \int_{-\infty}^{\infty} f(x)(-i\omega e^{-i\omega x})dx \\
&= i\omega \underbrace{\int_{-\infty}^{\infty} f(x)(e^{-i\omega x})dx}_{\mathcal{F}(f(x)} \\
&= i\omega \mathcal{F}(f(x)). \qquad (21)
\end{aligned}$$

We can similarly take the n-th order derivative $f^{(n)}$, and obtain $\mathcal{F}(f^{(n)}) = (i\omega)^n \mathcal{F}(f(x))$.

Hence, the derivative of a function, when considered after a Fourier transform to the frequency domain, will just become multiplication or scaling by $i\omega$ in the frequency domain; in particular, the magnitude at frequency ω will get scaled by ω (analogous to how $\frac{d}{dx}\sin(\omega x) = \omega \cos(\omega x)$).

4.7 Parseval's Theorem

Parseval's theorem applies to the Fourier transform as well, and is stated as follows:

$$\int_{-\infty}^{\infty} |f(x)|^2 dx = \frac{1}{2\pi} \int_{-\infty}^{\infty} |\hat{f}(\omega)|^2 d\omega.$$

In other words, the Fourier transform preserves the (squared) L2-norm, up to a constant related to the normalization of the basis (in this case $\frac{1}{2\pi}$). Again, the intuition is that the total energy of the time signal equals the total energy of the frequency signal.

5 Discrete Time

Thus far, we have considered functions $f(x)$ with a real-valued input $x \in \mathbb{R}$. Naturally, when we store a signal on a computer, we need to use *finitely many bits*, and a natural approach is to only consider a finite number of x values. (Each $f(x)$ will also need to be "quantized" to finitely many bits, but usually that's less of an issue because 64-bit numbers tend to be "accurate enough".) In addition, some signals/functions are simply discrete in nature to begin with, and we would still like to have a notion of Fourier transform for them.

For concreteness, we will think of the function input as representing "time", even though in applications it could represent other dimensions such as spatial. In addition, we will switch notation (with apologies if it is confusing) to choices that are more common when working in discrete time:

- The input x will be renamed to t, representing "time";
- Instead of a function $f(\cdot)$ with Fourier transform $\hat{f}(\cdot)$, we will consider a function $x(\cdot)$ with Fourier transform $X(\cdot)$.
- We will shortly switch from continuous time ($t \in \mathbb{R}$) to discrete time ($n \in \mathbb{Z}$), and in discrete time we will use square brackets to highlight this distinction, e.g., $x[n]$.

- Instead of the angular frequency ω, we will consider the ordinary frequency ξ (they are related by $\omega = 2\pi\xi$). As we mentioned previously, using ordinary frequency, the continuous-time Fourier transform of a signal $x(t)$ is

$$X(\xi) = \int_{-\infty}^{\infty} x(t) e^{-i2\pi\xi t} \, dt. \tag{22}$$

5.1 Sampling of Continuous Signals

It is useful to first think about the following: If we can measure some real-valued signal $x(t)$ by querying various t values, how should we do this in a manner that lets us store it on a computer but still retain the relevant information? By either stopping the measurements at some point or working in blocks of a given length, we can assume that t is bounded, say $t \in [0, T]$.

The natural strategy is to query $x(\cdot)$ at regular intervals, say $\{0, T_s, 2T_s, 3T_s, \dots, T\}$ for some *sampling interval* T_s (which we'll assume to be such that $\frac{T}{T_s}$ is an integer):

$$x[n] = x(t)\big|_{t=nT_s}. \tag{23}$$

This gives us a discrete-time signal of length $\frac{T}{T_s} + 1$. A key question is then what choice of T_s to use. At first glance, it seems that any $T_s > 0$ leads to *some* loss of information, with the loss vanishing only in the limit as $T_s \to 0$. Remarkably, as long as $x(t)$ does not contain arbitrarily high frequencies, we can in fact have *zero loss* for a carefully chosen T_s. It is more natural to state the result in terms of $\xi_s = \frac{1}{T_s}$, which we call the *sampling frequency.*

Shannon-Nyquist Sampling Theorem: *Let $x(t)$ be a continuous-time signal, let $X(\xi)$ be its Fourier transform, and let B be the highest frequency for which $X(\xi) \neq 0$ (i.e., $X(\xi) = 0$ for all $\xi > B$. The letter B stands for "bandwidth".). Let $x[n]$ be the discrete-time signal formed via* (23)*. Then, $x(t)$ can be* perfectly reconstructed *from $x[n]$ provided that the sampling frequency ξ_s satisfies $\xi_s > 2B$.*

This is a very famous result in signal processing, but we will skip its proof. In short, moving to discrete time loses nothing (at least mathematically) if we sample above twice the highest frequency of the signal.

(Optional** Aliasing)** It is also useful to understand what happens if $\xi_s < 2B$ (which is unavoidable if $B = \infty$; not all signals are bandlimited). To answer this, it is useful to think of all $t \in \mathbb{R}$ rather than only $t \in [0, T]$ (e.g., by taking $x(t) = 0$ outside $[0, T]$). Then $x[n]$ from (23) is defined for all integers. If we use the formula for the Fourier transform on $x[n]$, but replace the integral by a sum since $x[n]$ is discrete-time, we get

$$X_s(\xi) = \sum_{n=-\infty}^{\infty} x[n] e^{-i2\pi\xi n T_s}. \tag{24}$$

Alternatively, this can be interpreted as the Fourier transform of $x_s(t)$, defined to have a delta Dirac function of height $x[n]$ whenever $t = nT_s$ for some integer n. Using the linearity of the Fourier transform and the fact that the Dirac function's Fourier transform is a complex exponential (details omitted), this interpretation can be shown to lead to the following:

$$X_s(\xi) = \sum_{n=-\infty}^{\infty} X(\xi - n/T_s) = \sum_{n=-\infty}^{\infty} X(\xi - n\xi_s),$$

meaning that the spectrum of $X_s(\cdot)$ is a sum of *all shifted copies* of $X(\cdot)$, where the shifts are by $\pm\xi_s, \pm 2\xi_s$, and so on. If $\xi_s < 2B$, these shifted copies start to overlap, a phenomenon known as *aliasing*:

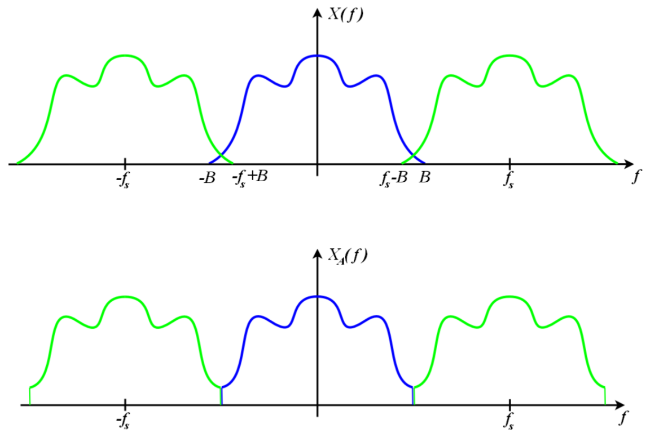


Figure 4: Illustration of aliasing in the frequency domain. (Frequencies here are denoted by f instead of ξ.)

The issue is that high frequency signals look the same as low-frequency ones when we sample too slowly, an example is shown as follows (this image and the one above are from Wikipedia):

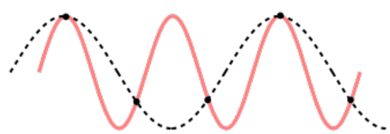


Figure 5: Illustration of aliasing in the time domain.

The effect of this is that any frequency $\xi \geq \frac{\xi_s}{2}$ gets "folded down" and looks the same as a lower frequency (e.g., a signal of frequency exactly ξ_s is indistinguishable from a signal of frequency 0, i.e., a constant signal). Having said this, if the frequency content of $X(\xi)$ is small enough for $\xi \geq \frac{\xi_s}{2}$, the impact of aliasing may be negligible.

5.2 Discrete Fourier Transform (DFT)

We are now ready to define the Discrete Fourier Transform (DFT). One way to view this is to start with $X_s(\xi)$ from (24) and simply evaluate it at a finite number (denoted by N) of frequencies:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-i \frac{2\pi}{N} kn}, \quad k = 0, 1, 2, ..., N-1. \tag{25}$$

This is another kind of sampling, but now it is *sampling in frequency domain* (with intervals of length $\frac{\xi_s}{N}$) rather than in time domain. With this definition, it can be shown that the following *inverse DFT* recovers the discrete-time signal:

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{i \frac{2\pi kn}{N}}, \quad n = 0, \dots, N-1. \tag{26}$$

Observe that both $x[n]$ and $X[k]$ are now described by just N numbers, making them amenable to storage and processing on a computer. The following should be noted:

- Any information about $x(t)$ outside $t \in [0, (N-1)T_s]$ is lost (though we could of course form a separate DFT corresponding to the later segments of the signal). Choosing N to make this consistent with

$t \in [0, T]$, we get $N = \frac{T}{T_s} + 1$. (The +1 term may be avoided by instead sampling at the *midpoints* of length-T_s regions, but we won't worry about this.)

- Aliasing may also incur a loss of information, as discussed above.

As we hinted earlier, the DFT is also useful for signals that are simply discrete-valued to begin with (rather than sampled version of continuous ones). In such scenarios, the length N is simply the length of the signal we are given (or perhaps the length of sub-blocks of the signal that we opt to work with).

The DFT has analogs of most of the main properties of the continuous-time Fourier transform (e.g., convolution property, Parseval's theorem, etc.). See `https://en.wikipedia.org/wiki/Discrete_Fourier_transform#Properties` for a summary.

5.3 Fast Fourier Transform (FFT)

The DFT consists of computing N coefficients, each defined as a sum over N terms, so naively the computation time is $O(N^2)$. (Or treating $x[n]$ as a vector, we can view the DFT as multiplication by an $N \times N$ "DFT matrix".) It turns out that the naive approach is actually doing a lot of repeated/redundant computation. The *Fast Fourier Transform* (FFT) is a (very) famous algorithm that avoids this, and brings the computation time down to $O(N \log N)$. There are several versions of the FFT, and the one that we present here is due to Cooley and Tukey.

The general procedure of Cooley and Tukey's algorithm is to re-express the discrete Fourier transform (DFT) recursively (divide-and-conquer), to reduce the computation time. We first rewrite the DFT to be

$$X[k] = \sum_{k=0}^{N-1} x[n] W_N^{kn}, \tag{27}$$

where

$$W_N^{kn} = e^{-i\frac{2\pi kn}{N}}. \tag{28}$$

It is useful to note that W_N^{kn} has a periodicity property: $W_N^{(k+N)n} = W_N^{kn}$.

To develop the FFT, we first split the single summation over N samples into 2 summations, each with $\frac{N}{2}$ samples, one for n even and the other for n odd. Substituting $m = \frac{n}{2}$ for n even and $m = \frac{n-1}{2}$ for n odd, we have

$$X[k] = \sum_{m=0}^{\frac{N}{2}-1} x[2m] W_N^{2mk} + \sum_{m=0}^{\frac{N}{2}-1} x[2m+1] W_N^{(2m+1)k}. \tag{29}$$

To simplify this, observe that

$$W_N^{2mk} = e^{-i\frac{2\pi(2mk)}{N}} = e^{-i\frac{2\pi mk}{\frac{N}{2}}} = W_{\frac{N}{2}}^{mk}. \tag{30}$$

Therefore,

$$X[k] = \sum_{m=0}^{\frac{N}{2}-1} x[2m] W_{\frac{N}{2}}^{mk} + W_N^k \sum_{m=0}^{\frac{N}{2}-1} x[2m+1] W_{\frac{N}{2}}^{mk}. \tag{31}$$

Defining $G[k]$ to be the length-$N/2$ DFT on even indices, and $H[K]$ the one on odd indices, it follows that

$$X[k] = G[k] + W_N^k H[k]. \tag{32}$$

Thus the N-point DFT $X[k]$ can be obtained from two $\frac{N}{2}$-point transforms! Although the frequency index k

ranges over N values, only $\frac{N}{2}$ values of $G[k]$ and $H[k]$ actually need to be computed, since $G[k]$ and $H[k]$ are periodic with period $\frac{N}{2}$. An illustration is given in Figure 6.

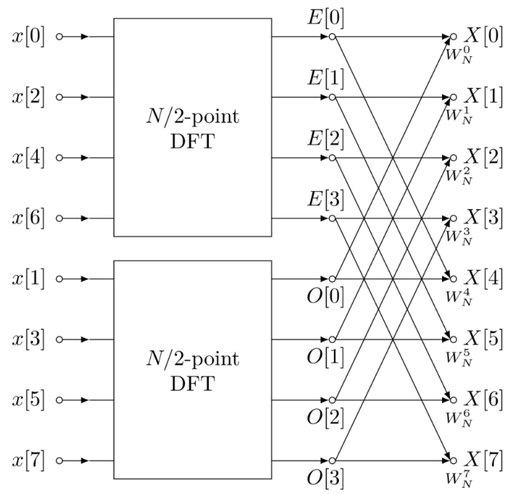


Figure 6: Calculating the N-point DFT from the $N/2$-point DFT. (Image from `https://lutpub.lut.fi/handle/10024/164024`)

Supposing that N is a power of 2, we can repeat the above process on the two $\frac{N}{2}$-point transforms, breaking them down to $\frac{N}{4}$-point transforms, and so on, until we come down to 2-point transform. At a recursion depth of $\log_2 N$ we are down to a length of 1, which is trivial. The $O(N \log N)$ scaling then simply comes from performing $O(N)$ computation at each depth.

6 (**Optional**) Applications

The Fourier transform is extensively used throughout signal processing, communications, machine learning, theoretical computer science, statistics, and more. We give just a few examples of applications here, and we don't go into much detail

6.1 Signal Denoising

A fundamental task in signal processing is *denoising* – removing as much noise as possible while keeping as much of the underlying signal as possible. The Fourier transform provides a very natural view for this task – if we know that the signal lies (mostly) in a certain frequency band, then we can filter out other frequencies and thus attenuate or remove any noise that occurs there. (Of course, any noise at the same frequencies as the signal will remain, but that tends to be a relatively small amount.)

Figure 7 gives an example of a signal corrupted by noise. The noise makes the signal highly corrupted, with very erratic oscillations. But since the noise fluctuates so much but the signal only exhibits mild oscillations, this suggests that the *signal lies at low frequencies* whereas the *noise contains high frequencies* that we should try to remove. The Fourier transform lets us see this much more precisely – see Figure 8. With this picture in mind, a natural denoising approach is to only keep frequencies whose value exceeds some thresold (e.g., the red line in the figure). Other methods are also possible, like keeping *all* low frequencies. There are still various design issues like what threshold or frequency range to use, or what prior knowledge (if any) we have about the signal, etc., but overall this is a very natural and effective approach to denoising.

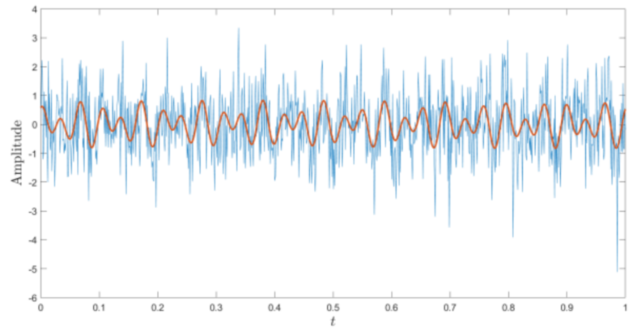


Figure 7: Signal (red) contaminated with Gaussian white noise (blue shows the noisy signal).

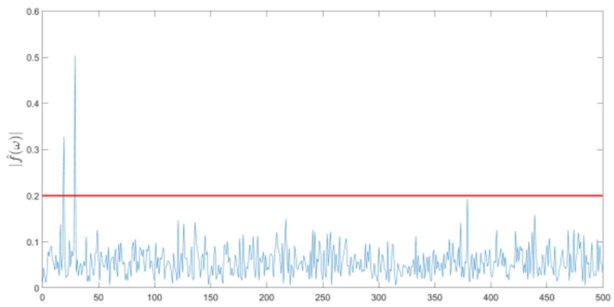


Figure 8: Frequency spectrum of the noisy signal after Fourier transform.

6.2 Linear Time Invariant Systems

In signal processing systems, we are interested in operations that input one signal $x(t)$ (or $x[n]$ in discrete time) and output another signal $y(t)$. (In fact, denoising as described above would be an example this!)

An important class of these is *linear time-invariant* (LTI) systems, which satisfy the linearity property (an input of $ax_1(t) + bx_2(t)$ gives output $ay_1(t) + by_2(t)$) and the time invariance property (an input of $x(t - \tau)$ gives output $y(t - \tau)$).

LTI systems are fully described by their *impulse response*, which is the output when $x(t) = \delta(t)$ (the Dirac delta function) – essentially, giving the system a "flick" and seeing how it responds. Once the impulse response is known, the output associated with an arbitrary input $x(t)$ is the convolution of x and the impulse response. (You may wish to try to prove this using the LTI properties and (15).)

By moving to the Fourier view, the convolution gets replaced by *multiplication*, which can be much easier to understand, interpret, and design based on.

6.3 Solving Differential Equations

The Fourier transform is a useful tool for solving differential equations efficiently, e.g., in the study of dynamical systems and control systems. We demonstrate one example below.

In this example, we consider a damped harmonic oscillator shown in Fig 9, which is a particle of mass m subject to a spring force and a damping force.

The motion of the particle can be derived using Newton's second law to obtain the ordinary differential equation (ODE):

$$\frac{d^2x}{dt^2} + 2\gamma y \frac{dx}{dt} + \omega_0^2 x(t) = 0. \tag{33}$$

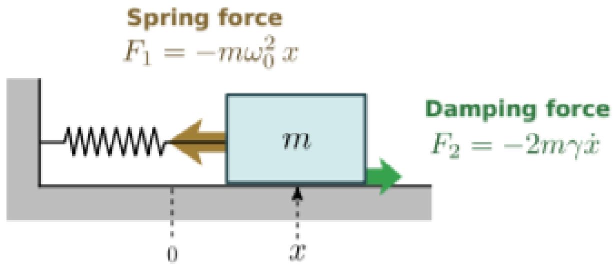


Figure 9: Illustration of a Damped Harmonic Oscillator

Moreover, if an additional *driving force* $f(t)$ is introduced that operates on the particle, we can further generalize this to

$$\frac{d^2x}{dt^2} + 2\gamma\frac{dx}{dt} + \omega_0^2 x(t) = \frac{f(t)}{m}. \tag{34}$$

To solve for $x(t)$, we first take the Fourier transform of both sides of the above equation. The result is

$$-\omega^2\hat{x}(\omega) - 2i\gamma\omega\hat{x}(\omega) + \omega_0^2\hat{x}(\omega) = \frac{\hat{f}(\omega)}{m}, \tag{35}$$

where $\hat{x}(\omega)$ and $\hat{f}(\omega)$ are the Fourier transforms of $x(t)$ and $f(t)$ respectively.

Thanks to moving to the Fourier domain, the inconvenient derivative operations are replaced by simpler and easier-to-analze operations. Specifically, we have obtained an algebraic equation that can easily be solved:

$$X(\omega) = \frac{F(\omega)/m}{-\omega^2 - 2i\gamma\omega + \omega_0^2}. \tag{36}$$

Knowing $X(\omega)$, we can use the inverse Fourier transform to obtain $x(t)$ (which is not an especially "simple" or "nice" expression, but it does give the solution we are after, and it can readily be computed numerically):

$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \frac{F(\omega)/m}{-\omega^2 - 2i\gamma\omega + \omega_0^2} e^{-i\omega t} d\omega, \quad \text{where } F(\omega) = \int_{-\infty}^{\infty} e^{i\omega t} f(t) dt. \tag{37}$$

This gives the system behavior of the particle under the effect of the external force. In control system design, we can further design the $f(t)$ at ease to realize the desired system behavior.

To summarize, the solution procedure for the driven harmonic oscillator equation consists of (i) using the Fourier transform on $f(t)$ to obtain $F(\omega)$, (ii) using the above equation to find $X(\omega)$ algebraically, and (iii) performing an inverse Fourier transform to obtain $x(t)$. We mainly leverage the derivative property of the Fourier transform to solve ODEs in real applications.

6.4 Fast Matrix Multiplication

Many problems (including some of those above) can be understood via the following picture:

Notably, taking the 3-step approach (to Fourier, solve, and then inverser Fourier) can be much easier than the 1-step approach. Here we give an example in the context of matrix multiplication, in which case it's the *discrete Fourier transform* that comes into play.

For an $n \times n$ matrix A and a $n \times 1$ vector x, the computation of Ax takes $O(n^2)$ time, and in general this can't be improved (since just reading A takes $O(n^2)$ time). But it can be improved for certain classes of

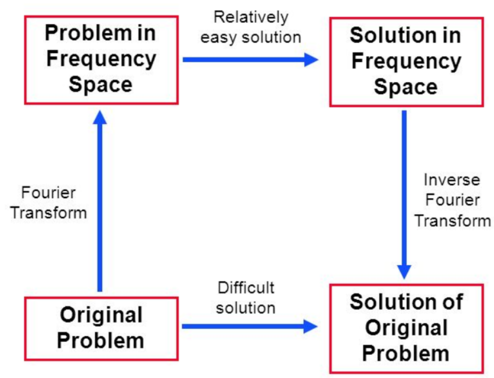


Figure 10: Problem-solving using Fourier transform

matrices. One such class is *circulant matrices*, which take the form

$$\begin{bmatrix} c_0 & c_1 & \dots & c_{n-2} & c_{n-1} \\ c_{n-1} & c_0 & c_1 & \dots & c_{n-2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ c_1 & c_2 & \dots & c_{n-1} & c_0 \end{bmatrix}$$

That is, the rows are cyclic shifts of each other, and we always have 'diagonals' taking a common value. If we let F be the $n \times n$ matrix that performs the DFT operation, and collect then c_i values into a vector c, then it can be shown that $A = F^{-1}\mathrm{diag}(Fc)F$.[2] Thus, we can compute Ax in $O(n \log n)$ time by 3 invocations of the FFT algorithm (two for F and one for F^{-1}), and using the fact that multiplication by a diagonal matrix only takes $O(n)$ time.

A circulant matrix is in fact performing a kind of convolution operation (with "wrap around" from index $n+1$ back to 1), so this example is essentially a different way of stating that convolution becomes multiplication after applying the (discrete) Fourier transform.

6.5 Other

Other applications of the Fourier transform include (but are not limited to) the following:

- In the field of communication systems, FT facilitates the modulation and demodulation of signals, thereby playing a crucial role in signal transmission and reception.
- It also finds significant usage in image and audio processing, aiding in the analysis, manipulation, and compression of image and sound signals.
- In machine learning, FT is used for various tasks such as feature encoding, fast calculation, modeling smoothness assumptions in optimization, and so on.
- As hinted at the start of the lecture, a more abstract use is the analsyis of Boolean functions (e.g., see `https://www.youtube.com/watch?v=lXJP-UkTl-4`), and there are many others.

[2]In more technical terms, this means that *the DFT matrix diagonalizes A*.

Appendix

A (**Optional**) More Detailed Background on Bases

Here we review in more detail some mathematical foundations and core concepts associated with vector spaces and the notion of basis. We assume that the reader is familiar with some linear algebra and concepts such as vector spaces, linear independence, and span.

- **Basis.** A basis is a set of vectors $\{v_i\}$ (for a vector space $\mathbb{V}$) if $\{v_i\}$ are linearly independent and $\mathbb{V} = \text{span}\{v_i\}$. For continuous vector spaces (e.g., $\mathbb{R}^d$), there are infinitely many possible bases.

- **Inner product.** The inner product can be thought of as a generalization of the dot product of Euclidean space. Given $\mathbb{V}$ that is a complex linear space and vectors $u, v \in \mathbb{V}$, the inner product $\langle u, v\rangle \in \mathbb{C}$ satisfies the following properties:
  - $\langle v, v\rangle \geq 0$
  - $\langle v, v\rangle = 0$ iff $v = 0$
  - $\forall u, v, w \in \mathbb{V}$ and $a, b \in \mathbb{C}$, $\langle au + bv, w\rangle = a\langle u, w\rangle + b\langle v, w\rangle$
  - $\langle u, v\rangle = \overline{\langle v, u\rangle}$

- **Euclidean space.** For $\mathbb{V} \in \mathbb{C}^n$, the inner product is $\langle u, v\rangle = \sum_{i=1}^n x_i \bar{y_i}$ (where $\bar{y}$ denotes the complex conjugate of y, i.e., $\overline{a + bi} = a - bi$.). For $\mathbb{V} \in \mathbb{R}^n$, this simplifies to $\langle u, v\rangle = \sum_{i=1}^n x_i y_i$.

- **Norm.** The norm associated with the inner product is $\|v\| = \sqrt{\langle v, v\rangle}$. For Euclidean space, this gives us the usual/familiar notion of length of a vector.

For studying the Fourier transform, rather than just finite vectors, we need to consider the space of continuous functions. We can similarly define an inner product for such a space. Let $V = S[a, b]$ be the space of continuous functions $f : [a, b] \to \mathbb{C}$. With the usual definition of sum of functions and multiplication by a scalar, V is a linear space and we can similarly define an inner product

$$\langle f, g\rangle = \int_a^b f(x)\overline{g(x)}dx. \tag{38}$$

Notice how intuitively the continuous functions over the interval are similar to a limiting case where we have an "infinite-length" vector and the discrete summation for the inner product replaced by an integral of the product of the two functions.

We can now introduce orthogonality and orthonormal systems.

- **Orthogonality.** u and v are orthogonal if $\langle u, v\rangle = 0$.

- **Orthogonal system.** An orthogonal system is a sequence of non-zero vectors $\{u_i\}$ such that u_i and u_j are orthogonal $\forall i \neq j$. Note that $\{u_i\}$ can consist of infinite many elements – we can call this an infinite orthogonal system.

- **Orthonormal system.** An orthonormal system is an orthogonal system that additionally satisfies $\|u_i\| = 1, \forall u_i$.

Given an orthonormal system $\{e_i\}$, if $v = \sum_{i=1}^{n} a_i e_i$ then $a_i = \langle v, e_i \rangle$. The proof is straightforward, following just from the linearity of the inner product and orthonormality of the system $\{e_i\}$. When v is in the span of $\{e_i\}$, then $v = \sum_{i=1}^{n} a_i e_i = \sum_{i=1}^{n} \langle v, e_i \rangle e_i$. This may not always be the case, in which case we can define a useful concept:

- **Orthogonal projection.** The orthogonal projection of v on the span of the orthonormal system $\{e_i\}$ can be defined as $\tilde{v} = \sum_{i=1}^{n} \langle v, e_i \rangle e_i$.

With these, we can then define a **closed orthonormal system**. Given an infinite orthonormal system $\{e_i\}$ and inner product space V, we can define it as closed in V if $\forall v \in V$,

$$\lim_{n\to\infty} \left\| v - \sum_{i=1}^{n} \langle v, e_i \rangle e_i \right\| = 0. \tag{39}$$

With a closed infinite orthonormal system, we can then represent any vector v in terms of the system basis $\{e_i\}$. Intuitively, this means that the set $\{e_i\}$ is rich enough to be able to capture all relevant features and express any vector v. In practice, we may only use a finite number of e_i to represent the function, in which case it becomes an approximation rather than being exact. The approximation generally improves as the number of terms taken increases, though in some scenarios the improvement may be slow.

B (**Optional**) Convergence Properties for the Fourier Series

We have not shown that the orthogonal systems Eq. (1) and Eq. (6) are closed, nor that the Fourier series can "reasonably represent" any piecewise continuous periodic function. Details regarding the pointwise, uniform, and norm convergence of the Fourier series are beyond our scope, but we briefly and informally mention some key results here:

- **Pointwise convergence.** Let f be a periodic, piecewise smooth function on $\mathbb{R}$, and let $FS[f]$ be the Fourier series (either using sines/cosines or complex exponentials) for f. Then:
  - $FS[f]$ convergences at every point where f is continuous, i.e., $f(x) = FS[f]|_x$ at every x where f is continuous.
  - If a is a point where the function is discontinuous, then as the number of Fourier terms in the partial sum $N \to \infty$, we get $FS_N[f] \to \frac{1}{2}\left(\lim_{x\to a^-} f(x) + \lim_{x\to a^+} f(x)\right)$.
- **Uniform convergence.** Let f be a continuous, piecewise smooth, periodic function with period p. Then:
  - Its complex exponential Fourier series $FS[f]$ converges uniformly to f. Furthermore, for any real value x and pairs of integers M and N where $M < 0 < N$, we have

$$\left| f(x) - \sum_{k=M}^{N} c_k e^{i2\pi\omega_k x} \right| \le \Big[\frac{1}{\sqrt{M}} + \frac{1}{\sqrt{N}}\Big] B,$$

    where $B = \frac{1}{2\pi}\left(p\int_{-\pi}^{\pi} |f'(x)|^2 dx\right)^{\frac{1}{2}}, \ \ \forall x$. Thus, the approximation error becomes low as M and N increases as long as none of the derivatives are too large.

– The proof is based on relating coefficients of the Fourier series of f to those of its derivative f', in the same way as what we did for the Fourier transform in Section 4.6.

– We can also get a better bound for smoother functions, i.e., when we know that a function is m-times differentiable, and $f^{(m)}$ is piecewise continuous. Then the bound becomes

$$\left| f(x) - \sum_{k=M}^{N} c_k e^{i2\pi\omega_k x} \right| \leq \left[\frac{1}{\sqrt{M^{2m-1}}} + \frac{1}{\sqrt{N^{2m-1}}} \right] \tilde{B},$$

where $\tilde{B}$ is a constant proportional to the magnitude of $f^{(m)}$.