CHAPTER 4

FOURIER SERIES AND INTEGRALS

4.1 FOURIER SERIES FOR PERIODIC FUNCTIONS

This section explains three Fourier series: **sines, cosines, and exponentials e^{ikx}**. Square waves (1 or 0 or -1) are great examples, with delta functions in the derivative. We look at a spike, a step function, and a ramp—and smoother functions too.

Start with $\sin x$. It has period 2π since $\sin(x + 2\pi) = \sin x$. It is an odd function since $\sin(-x) = -\sin x$, and it vanishes at $x = 0$ and $x = \pi$. Every function $\sin nx$ has those three properties, and Fourier looked at *infinite combinations of the sines*:

Fourier sine series $$S(x) = b_1 \sin x + b_2 \sin 2x + b_3 \sin 3x + \cdots = \sum_{n=1}^{\infty} b_n \sin nx \quad (1)$$

If the numbers $b_1, b_2, \ldots$ drop off quickly enough (we are foreshadowing the importance of the decay rate) then the sum $S(x)$ will inherit all three properties:

Periodic $S(x + 2\pi) = S(x)$ **Odd** $S(-x) = -S(x)$ $S(0) = S(\pi) = 0$

200 years ago, Fourier startled the mathematicians in France by suggesting that *any function* $S(x)$ with those properties could be expressed as an infinite series of sines. This idea started an enormous development of Fourier series. Our first step is to compute from $S(x)$ the number b_k that multiplies $\sin kx$.

Suppose $S(x) = \sum b_n \sin nx$. *Multiply both sides by* $\sin kx$. *Integrate from* 0 *to* π:

$$\int_0^\pi S(x) \sin kx \, dx = \int_0^\pi b_1 \sin x \sin kx \; dx + \cdots + \int_0^\pi \boldsymbol{b_k \sin kx \; \sin kx \, dx} + \cdots \quad (2)$$

On the right side, all integrals are zero except the highlighted one with $n = k$. This property of "**orthogonality**" will dominate the whole chapter. The sines make 90° angles in function space, when their inner products are integrals from 0 to π:

Orthogonality $$\int_0^\pi \sin nx \; \sin kx \, dx = 0 \quad \text{if} \quad n \neq k\,. \quad (3)$$

Zero comes quickly if we integrate $\int \cos mx\, dx = \left[\frac{\sin mx}{m}\right]_0^\pi = 0 - 0$. So we use this:

Product of sines
$$\sin nx \, \sin kx = \frac{1}{2}\cos(n-k)x - \frac{1}{2}\cos(n+k)x\,. \tag{4}$$

Integrating $\cos mx$ with $m = n - k$ and $m = n + k$ proves orthogonality of the sines.

The exception is when $n = k$. Then we are integrating $(\sin kx)^2 = \frac{1}{2} - \frac{1}{2}\cos 2kx$:

$$\int_0^\pi \sin kx \, \sin kx \, dx = \int_0^\pi \frac{1}{2}\, dx - \int_0^\pi \frac{1}{2}\cos 2kx \, dx = \frac{\pi}{2}\,. \tag{5}$$

The highlighted term in equation (2) is $\boldsymbol{b_k\pi/2}$. Multiply both sides of (2) by $2/\pi$:

Sine coefficients
$S(-x) = -S(x)$
$$b_k = \frac{2}{\pi}\int_0^\pi S(x)\sin kx\, dx = \frac{1}{\pi}\int_{-\pi}^\pi S(x)\sin kx\, dx. \tag{6}$$

Notice that $S(x)\sin kx$ is *even* (equal integrals from $-\pi$ to 0 and from 0 to π).

I will go immediately to the most important example of a Fourier sine series. $S(x)$ is an **odd square wave** with $SW(x) = 1$ for $0 < x < \pi$. It is drawn in Figure 4.1 as an odd function (with period 2π) that vanishes at $x = 0$ and $x = \pi$.

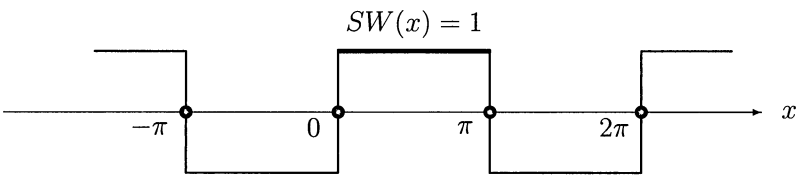


Figure 4.1: The odd square wave with $SW(x + 2\pi) = SW(x) = \{1 \text{ or } 0 \text{ or } -1\}$.

Example 1 Find the Fourier sine coefficients b_k of the square wave $SW(x)$.

Solution For $k = 1, 2, \ldots$ use the first formula (6) with $S(x) = 1$ between 0 and π:

$$b_k = \frac{2}{\pi}\int_0^\pi \sin kx\, dx = \frac{2}{\pi}\left[\frac{-\cos kx}{k}\right]_0^\pi = \frac{2}{\pi}\left\{\frac{\mathbf{2}}{\mathbf{1}}, \frac{0}{2}, \frac{\mathbf{2}}{\mathbf{3}}, \frac{0}{4}, \frac{\mathbf{2}}{\mathbf{5}}, \frac{0}{6}, \cdots\right\} \tag{7}$$

The even-numbered coefficients b_{2k} are all zero because $\cos 2k\pi = \cos 0 = 1$. The odd-numbered coefficients $b_k = 4/\pi k$ decrease at the rate $1/k$. We will see that same $1/k$ decay rate for all functions formed from *smooth pieces and jumps.*

Put those coefficients $4/\pi k$ and zero into the Fourier sine series for $SW(x)$:

Square wave
$$\boldsymbol{SW(x)} = \frac{4}{\pi}\left[\frac{\sin x}{\mathbf{1}} + \frac{\sin 3x}{\mathbf{3}} + \frac{\sin 5x}{\mathbf{5}} + \frac{\sin 7x}{\mathbf{7}} + \cdots\right] \tag{8}$$

Figure 4.2 graphs this sum after one term, then two terms, and then five terms. You can see the all-important **Gibbs phenomenon** appearing as these "partial sums"

include more terms. Away from the jumps, we safely approach $SW(x) = 1$ or -1. At $x = \pi/2$, the series gives a beautiful alternating formula for the number π:

$$1 = \frac{4}{\pi}\left[\frac{1}{1} - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots\right] \quad \text{so that} \quad \boldsymbol{\pi = 4\left[\frac{1}{1} - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots\right]}. \tag{9}$$

The Gibbs phenomenon is the overshoot that moves closer and closer to the jumps. Its height approaches 1.18... and it does not decrease with more terms of the series! Overshoot is the one greatest obstacle to calculation of all discontinuous functions (like shock waves in fluid flow). We try hard to avoid Gibbs but sometimes we can't.

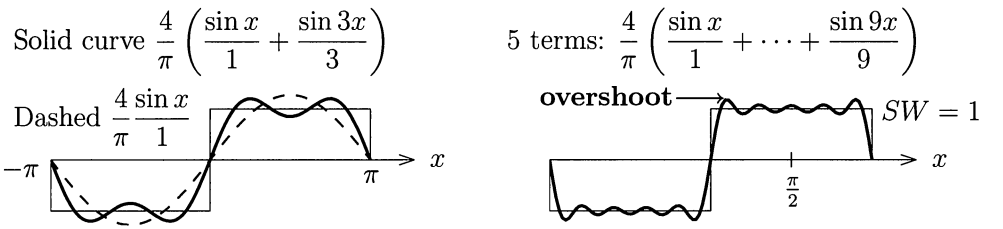


Figure 4.2: **Gibbs phenomenon**: Partial sums $\sum_1^N b_n \sin nx$ overshoot near jumps.

Fourier Coefficients are Best

Let me look again at the first term $b_1 \sin x = (4/\pi)\sin x$. This is the **closest possible approximation** to the square wave SW, by any multiple of $\sin x$ (closest in the least squares sense). To see this optimal property of the Fourier coefficients, minimize the error over all b_1:

The error is $\displaystyle\int_0^\pi (SW - b_1 \sin x)^2\, dx$ The b_1 derivative is $\displaystyle -2\int_0^\pi (SW - b_1 \sin x)\sin x\, dx$.

The integral of $\sin^2 x$ is $\pi/2$. So the derivative is zero when $b_1 = (2/\pi)\int_0^\pi S(x)\sin x\, dx$. This is exactly equation (6) for the Fourier coefficient.

Each $b_k \sin kx$ is as close as possible to $SW(x)$. We can find the coefficients b_k one at a time, *because the sines are orthogonal.* The square wave has $b_2 = 0$ because all other multiples of $\sin 2x$ increase the error. Term by term, we are "projecting the function onto each axis $\sin kx$."

Fourier Cosine Series

The cosine series applies to *even functions* with $C(-x) = C(x)$:

$$\textbf{Cosine series} \quad C(x) = a_0 + a_1 \cos x + a_2 \cos 2x + \cdots = a_0 + \sum_{n=1}^{\infty} a_n \cos nx. \tag{10}$$

Every cosine has period 2π. Figure 4.3 shows two even functions, the **repeating ramp** $RR(x)$ and the **up-down train** $UD(x)$ of delta functions. That sawtooth ramp RR is the integral of the square wave. The delta functions in UD give the derivative of the square wave. (For sines, the integral and derivative are cosines.) RR and UD will be valuable examples, one smoother than SW, one less smooth.

First we find formulas for the cosine coefficients a_0 and a_k. The constant term a_0 is the *average value* of the function $C(x)$:

a_0 = Average
$$a_0 = \frac{1}{\pi}\int_0^\pi C(x)\,dx = \frac{1}{2\pi}\int_{-\pi}^{\pi} C(x)\,dx. \tag{11}$$

I just integrated every term in the cosine series (10) from 0 to π. On the right side, the integral of a_0 is $a_0\pi$ (divide both sides by π). All other integrals are zero:

$$\int_0^\pi \cos nx\,dx = \left[\frac{\sin nx}{n}\right]_0^\pi = 0 - 0 = 0. \tag{12}$$

In words, the constant function 1 is orthogonal to $\cos nx$ over the interval $[0, \pi]$.

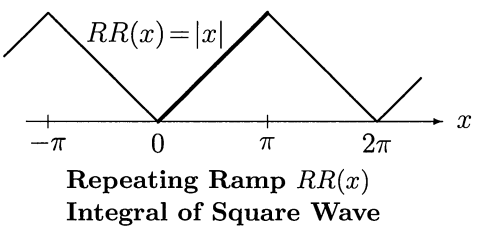
The other cosine coefficients a_k come from the *orthogonality of cosines*. As with sines, we multiply both sides of (10) by $\cos kx$ and integrate from 0 to π:

$$\int_0^\pi C(x)\cos kx\,dx = \int_0^\pi a_0\cos kx\,dx + \int_0^\pi a_1\cos x\cos kx\,dx + \cdots + \int_0^\pi \boldsymbol{a_k(\cos kx)^2\,dx} + \cdots$$

You know what is coming. On the right side, only the highlighted term can be nonzero. Problem 4.1.1 proves this by an identity for $\cos nx\cos kx$—now (4) has a plus sign. The bold nonzero term is $\boldsymbol{a_k\pi/2}$ and we multiply both sides by $2/\pi$:

Cosine coefficients
$C(-x) = C(x)$
$$a_k = \frac{2}{\pi}\int_0^\pi C(x)\cos kx\,dx = \frac{1}{\pi}\int_{-\pi}^{\pi} C(x)\cos kx\,dx\,. \tag{13}$$

Again the integral over a full period from $-\pi$ to π (also 0 to 2π) is just doubled.



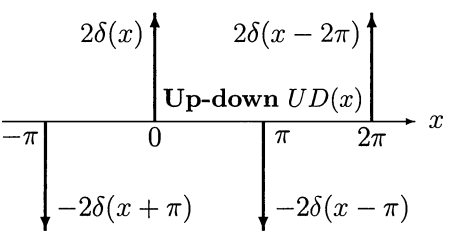


Figure 4.3: The repeating ramp RR and the up-down UD (periodic spikes) are even. The derivative of RR is the odd square wave SW. **The derivative of SW is UD.**

Example 2 Find the cosine coefficients of the ramp $RR(x)$ and the up-down $UD(x)$.

Solution The simplest way is to start with the sine series for the square wave:

$$SW(x) = \frac{4}{\pi}\left[\frac{\sin x}{1} + \frac{\sin 3x}{3} + \frac{\sin 5x}{5} + \frac{\sin 7x}{7} + \cdots\right].$$

Take the derivative of every term to produce cosines in the up-down delta function:

Up-down series $$UD(x) = \frac{4}{\pi}\left[\cos x + \cos 3x + \cos 5x + \cos 7x + \cdots\right]. \tag{14}$$

Those coefficients don't decay at all. The terms in the series don't approach zero, so officially the series cannot converge. Nevertheless it is somehow correct and important. Unofficially this sum of cosines has all 1's at $x = 0$ and all -1's at $x = \pi$. Then $+\infty$ and $-\infty$ are consistent with $2\delta(x)$ and $-2\delta(x - \pi)$. The true way to recognize $\delta(x)$ is by the test $\int \delta(x) f(x)\, dx = f(0)$ and Example 3 will do this.

For the repeating ramp, we integrate the square wave series for $SW(x)$ and add the average ramp height $a_0 = \pi/2$, halfway from 0 to π:

Ramp series $$\boldsymbol{RR(x)} = \frac{\pi}{2} - \frac{\pi}{4}\left[\frac{\cos x}{\mathbf{1^2}} + \frac{\cos 3x}{\mathbf{3^2}} + \frac{\cos 5x}{\mathbf{5^2}} + \frac{\cos 7x}{\mathbf{7^2}} + \cdots\right]. \tag{15}$$

The constant of integration is a_0. *Those coefficients a_k drop off like $1/k^2$.* They could be computed directly from formula (13) using $\int x \cos kx\, dx$, but this requires an integration by parts (or a table of integrals or an appeal to *Mathematica* or *Maple*). It was much easier to integrate every sine separately in $SW(x)$, which makes clear the crucial point: Each "degree of smoothness" in the function is reflected in a faster decay rate of its Fourier coefficients a_k and b_k.

No decay	**Delta** functions (with spikes)
$\mathbf{1/k}$ **decay**	**Step** functions (with jumps)
$\mathbf{1/k^2}$ **decay**	**Ramp** functions (with corners)
$\mathbf{1/k^4}$ **decay**	**Spline** functions (jumps in f''')
$\boldsymbol{r^k}$ **decay with** $\boldsymbol{r < 1}$	**Analytic** functions like $1/(2 - \cos x)$

Each integration divides the kth coefficient by k. So the decay rate has an extra $1/k$. The "Riemann-Lebesgue lemma" says that a_k and b_k approach zero for any continuous function (in fact whenever $\int |f(x)| dx$ is finite). Analytic functions achieve a new level of smoothness—they can be differentiated forever. Their Fourier series and Taylor series in Chapter 5 converge **exponentially fast**.

The poles of $1/(2 - \cos x)$ will be complex solutions of $\cos x = 2$. Its Fourier series converges quickly because r^k decays faster than any power $1/k^p$. Analytic functions are ideal for computations—the Gibbs phenomenon will never appear.

Now we go back to $\delta(x)$ for what could be the most important example of all.

Example 3 Find the (cosine) coefficients of the *delta function* $\delta(x)$, made 2π-periodic.

Solution The spike occurs at the start of the interval $[0, \pi]$ so safer to integrate from $-\pi$ to π. We find $a_0 = 1/2\pi$ and the other $a_k = 1/\pi$ (cosines because $\delta(x)$ is even):

$$\textbf{Average} \quad a_0 = \frac{1}{2\pi}\int_{-\pi}^{\pi} \delta(x)\,dx = \boldsymbol{\frac{1}{2\pi}} \qquad \textbf{Cosines} \quad a_k = \frac{1}{\pi}\int_{-\pi}^{\pi} \delta(x)\cos kx\,dx = \boldsymbol{\frac{1}{\pi}}$$

Then the series for the delta function has all cosines in equal amounts:

$$\textbf{Delta function} \qquad \delta(x) = \frac{1}{2\pi} + \frac{1}{\pi}\left[\cos x + \cos 2x + \cos 3x + \cdots\right]. \tag{16}$$

Again this series cannot truly converge (its terms don't approach zero). But we can graph the sum after $\cos 5x$ and after $\cos 10x$. Figure 4.4 shows how these "partial sums" are doing their best to approach $\delta(x)$. They oscillate faster and faster away from $x = 0$.

Actually there is a neat formula for the partial sum $\delta_N(x)$ that stops at $\cos Nx$. Start by writing each term $2\cos\theta$ as $e^{\imath\theta} + e^{-\imath\theta}$:

$$\delta_N = \frac{1}{2\pi}\left[1 + 2\cos x + \cdots + 2\cos Nx\right] = \frac{1}{2\pi}\left[1 + e^{\imath x} + e^{-\imath x} + \cdots + e^{\imath Nx} + e^{-\imath Nx}\right].$$

This is a geometric progression that starts from $e^{-\imath Nx}$ and ends at $e^{\imath Nx}$. We have powers of the same factor e^{ix}. The sum of a geometric series is known:

$$\textbf{Partial sum up to } \boldsymbol{\cos Nx} \qquad \delta_N(x) = \frac{1}{2\pi}\frac{e^{\imath(N+\frac{1}{2})x} - e^{-i(N+\frac{1}{2})x}}{e^{ix/2} - e^{-\imath x/2}} = \frac{1}{2\pi}\frac{\sin(N+\frac{1}{2})x}{\sin\frac{1}{2}x}. \tag{17}$$

This is the function graphed in Figure 4.4. We claim that for any N the area underneath $\delta_N(x)$ is 1. (Each cosine integrated from $-\pi$ to π gives zero. The integral of $1/2\pi$ is 1.) The central "lobe" in the graph ends when $\sin(N + \frac{1}{2})x$ comes down to zero, and that happens when $(N + \frac{1}{2})x = \pm\pi$. I think the area under that lobe (marked by bullets) approaches the same number $1.18\ldots$ that appears in the Gibbs phenomenon.

In what way does $\delta_N(x)$ approach $\delta(x)$? The terms $\cos nx$ in the series jump around at each point $x \neq 0$, not approaching zero. At $x = \pi$ we see $\frac{1}{2\pi}[1 - 2 + 2 - 2 + \cdots]$ and the sum is $1/2\pi$ or $-1/2\pi$. The bumps in the partial sums don't get smaller than $1/2\pi$. The right test for the delta function $\delta(x)$ is to multiply by a smooth $f(x) = \sum a_k \cos kx$ and integrate, because *we only know* $\delta(x)$ *from its integrals* $\int \delta(x)f(x)\,dx = f(0)$:

$$\textbf{Weak convergence of } \boldsymbol{\delta_N(x)} \textbf{ to } \boldsymbol{\delta(x)} \qquad \int_{-\pi}^{\pi} \boldsymbol{\delta_N(x) f(x)\,dx = a_0 + \cdots + a_N \to f(0)}. \tag{18}$$

In this integrated sense (*weak sense*) the sums $\delta_N(x)$ do approach the delta function! The convergence of $a_0 + \cdots + a_N$ is the statement that at $x = 0$ the Fourier series of a smooth $f(x) = \sum a_k \cos kx$ converges to the number $f(0)$.

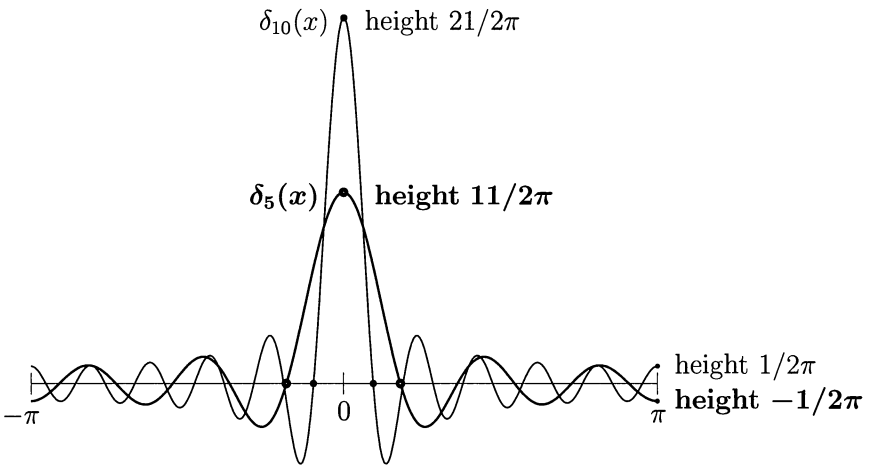


Figure 4.4: The sums $\delta_N(x) = (1 + 2\cos x + \cdots + 2\cos Nx)/2\pi$ try to approach $\delta(x)$.

Complete Series: Sines and Cosines

Over the half-period $[0, \pi]$, the sines are not orthogonal to all the cosines. In fact the integral of $\sin x$ times 1 is not zero. So for functions $F(x)$ that are not odd or even, we move to the complete series (sines plus cosines) on the full interval. Since our functions are periodic, that "full interval" can be $[-\pi, \pi]$ or $[0, 2\pi]$:

Complete Fourier series
$$F(x) = a_0 + \sum_{n=1}^{\infty} a_n \cos nx + \sum_{n=1}^{\infty} b_n \sin nx\,. \tag{19}$$

On every "2π interval" all sines and cosines are mutually orthogonal. We find the Fourier coefficients a_k and b_k in the usual way: **Multiply** (19) **by** 1 **and** $\cos kx$ **and** $\sin kx$, **and integrate both sides from** $-\pi$ **to** π:

$$\boldsymbol{a_0} = \frac{1}{2\pi}\int_{-\pi}^{\pi} F(x)\,dx \quad \boldsymbol{a_k} = \frac{1}{\pi}\int_{-\pi}^{\pi} F(x)\cos kx\,dx \quad \boldsymbol{b_k} = \frac{1}{\pi}\int_{-\pi}^{\pi} F(x)\sin kx\,dx. \tag{20}$$

Orthogonality kills off infinitely many integrals and leaves only the one we want.

Another approach is to split $F(x) = C(x) + S(x)$ into an even part and an odd part. Then we can use the earlier cosine and sine formulas. The two parts are

$$C(x) = F_{\mathbf{even}}(x) = \frac{F(x) + F(-x)}{2} \qquad S(x) = F_{\mathbf{odd}}(x) = \frac{F(x) - F(-x)}{2}. \tag{21}$$

The even part gives the a's and the odd part gives the b's. Test on a short square pulse from $x = 0$ to $x = h$—this one-sided function is not odd or even.

Example 4 Find the a's and b's if $F(x) = \textbf{square pulse} = \begin{cases} 1 & \text{for } 0 < x < h \\ 0 & \text{for } h < x < 2\pi \end{cases}$

Solution The integrals for a_0 and a_k and b_k stop at $x = h$ where $F(x)$ drops to zero. The coefficients decay like $1/k$ because of the jump at $x = 0$ and the drop at $x = h$:

Coefficients of square pulse $\quad a_0 = \dfrac{1}{2\pi}\displaystyle\int_0^h 1\,dx = \dfrac{\boldsymbol{h}}{\boldsymbol{2\pi}} = \textbf{average}$

$$a_k = \frac{1}{\pi}\int_0^h \cos kx\,dx = \frac{\boldsymbol{\sin kh}}{\boldsymbol{\pi k}} \qquad b_k = \frac{1}{\pi}\int_0^h \sin kx\,dx = \frac{\boldsymbol{1-\cos kh}}{\boldsymbol{\pi k}}. \tag{22}$$

If we divide $F(x)$ by h, its graph is a tall thin rectangle: height $\frac{1}{h}$, base h, and area $= 1$.

When h approaches zero, $F(x)/h$ is squeezed into a very thin interval. *The tall rectangle approaches (weakly) the delta function* $\delta(x)$. The average height is area$/2\pi = 1/2\pi$. Its other coefficients a_k/h and b_k/h approach $1/\pi$ and 0, already known for $\delta(x)$:

$$\frac{\boldsymbol{F(x)}}{\boldsymbol{h}} \to \boldsymbol{\delta(x)} \qquad \frac{a_k}{h} = \frac{1}{\pi}\frac{\sin kh}{kh} \to \frac{1}{\pi} \quad \text{and} \quad \frac{b_k}{h} = \frac{1-\cos kh}{\pi kh} \to 0 \text{ as } h \to 0. \tag{23}$$

When the function has a jump, its Fourier series picks the halfway point. This example would converge to $F(0) = \frac{1}{2}$ and $F(h) = \frac{1}{2}$, halfway up and halfway down.

The Fourier series converges to $F(x)$ at each point where the function is smooth. This is a highly developed theory, and Carleson won the 2006 Abel Prize by proving convergence for every x except a set of measure zero. If the function has finite energy $\int |F(x)|^2\,dx$, he showed that the Fourier series converges "almost everywhere."

Energy in Function = Energy in Coefficients

There is an extremely important equation (***the energy identity***) that comes from integrating $(F(x))^2$. When we square the Fourier series of $F(x)$, and integrate from $-\pi$ to π, all the "cross terms" drop out. The only nonzero integrals come from 1^2 and $\cos^2 kx$ and $\sin^2 kx$, multiplied by a_0^2 and a_k^2 and b_k^2:

$$\begin{aligned} \textbf{Energy in } \boldsymbol{F(x)} &= \textstyle\int_{-\pi}^{\pi}(\boldsymbol{a_0} + \sum \boldsymbol{a_k}\cos \boldsymbol{kx} + \sum \boldsymbol{b_k}\sin \boldsymbol{kx})^2 \boldsymbol{dx} \\ \textstyle\int_{-\pi}^{\pi}(\boldsymbol{F(x)})^2 \boldsymbol{dx} &= \boldsymbol{2\pi a_0^2 + \pi(a_1^2 + b_1^2 + a_2^2 + b_2^2 + \cdots)}. \end{aligned} \tag{24}$$

The energy in $F(x)$ equals the energy in the coefficients. The left side is like the length squared of a vector, except *the vector is a function.* The right side comes from an infinitely long vector of a's and b's. The lengths are equal, which says that the Fourier transform from function to vector is like an orthogonal matrix. Normalized by constants $\sqrt{2\pi}$ and $\sqrt{\pi}$, we have an *orthonormal basis in function space.*

What is this function space? It is like ordinary 3-dimensional space, except the "vectors" are functions. Their length $\|f\|$ comes from integrating instead of adding: $\|f\|^2 = \int |f(x)|^2 dx$. These functions fill **Hilbert space**. The rules of geometry hold:

Length $\|f\|^2 = (f, f)$ comes from the inner product $(f, g) = \int f(x)g(x)\,dx$

Orthogonal functions $(f, g) = 0$ produce a right triangle: $\|f + g\|^2 = \|f\|^2 + \|g\|^2$

I have tried to draw Hilbert space in Figure 4.5. It has infinitely many axes. *The energy identity* (24) *is exactly the Pythagoras Law in infinite-dimensional space.*

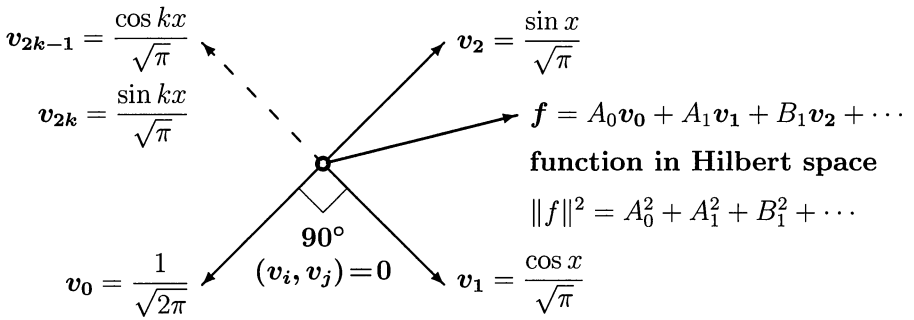


Figure 4.5: The Fourier series is a combination of orthonormal v's (sines and cosines).

Complex Exponentials $c_k e^{ikx}$

This is a small step and we have to take it. In place of separate formulas for a_0 and a_k and b_k, we will have *one formula* for all the complex coefficients c_k. And the function $F(x)$ might be complex (as in quantum mechanics). The Discrete Fourier Transform will be much simpler when we use N complex exponentials for a vector. We practice in advance with the complex infinite series for a 2π-periodic function:

Complex Fourier series
$$F(x) = c_0 + c_1e^{ix} + c_{-1}e^{-ix} + \cdots = \sum_{n=-\infty}^{\infty} c_n e^{inx} \quad (25)$$

If every $c_n = c_{-n}$, we can combine e^{inx} with e^{-inx} into $2\cos nx$. Then (25) is the cosine series for an even function. If every $c_n = -c_{-n}$, we use $e^{inx} - e^{-inx} = 2i\sin nx$. Then (25) is the sine series for an odd function and the c's are pure imaginary.

To find c_k, multiply (25) by e^{-ikx} (not e^{ikx}) **and integrate from $-\pi$ to π:**

$$\int_{-\pi}^{\pi} F(x)e^{-ikx}dx = \int_{-\pi}^{\pi} c_0e^{-ikx}dx + \int_{-\pi}^{\pi} c_1e^{ix}e^{-ikx}dx + \cdots + \int_{-\pi}^{\pi} \boldsymbol{c_k e^{ikx}e^{-ikx}dx} + \cdots$$

The complex exponentials are orthogonal. Every integral on the right side is zero, except for the highlighted term (when $n = k$ and $e^{ikx}e^{-ikx} = 1$). The integral of 1 is 2π. That surviving term gives the formula for c_k:

Fourier coefficients
$$\int_{-\pi}^{\pi} F(x)e^{-ikx}\,dx = 2\pi c_k \quad \text{for} \quad k = 0, \pm 1, \ldots \quad (26)$$

Notice that $c_0 = a_0$ is still the average of $F(x)$, because $e^0 = 1$. The orthogonality of e^{inx} and e^{ikx} is checked by integrating, as always. But the complex inner product (F, G) takes the *complex conjugate* $\overline{G}$ *of* G. Before integrating, change e^{ikx} to e^{-ikx}:

$$\textbf{Complex inner product} \quad (F,G) = \int_{-\pi}^{\pi} F(x)\overline{G(x)}\,dx \qquad \textbf{Orthogonality of } \boldsymbol{e^{inx}} \textbf{ and } \boldsymbol{e^{ikx}} \quad \int_{-\pi}^{\pi} e^{i(n-k)x}dx = \left[\frac{e^{i(n-k)x}}{i(n-k)}\right]_{-\pi}^{\pi} = 0. \tag{27}$$

Example 5 Add the complex series for $1/(2 - e^{ix})$ and $1/(2 - e^{-ix})$. These geometric series have exponentially fast decay from $1/2^k$. The functions are analytic.

$$\left(\frac{1}{2} + \frac{e^{ix}}{4} + \frac{e^{2ix}}{8} + \cdots\right) + \left(\frac{1}{2} + \frac{e^{-ix}}{4} + \frac{e^{-2ix}}{8} + \cdots\right) = 1 + \frac{\cos x}{2} + \frac{\cos 2x}{4} + \frac{\cos 3x}{8} + \cdots$$

When we add those functions, we get a real analytic function:

$$\frac{1}{2 - e^{ix}} + \frac{1}{2 - e^{-ix}} = \frac{(2 - e^{-ix}) + (2 - e^{ix})}{(2 - e^{ix})(2 - e^{-ix})} = \frac{4 - 2\cos x}{5 - 4\cos x} \tag{28}$$

This ratio is the infinitely smooth function whose cosine coefficients are $1/2^k$.

Example 6 Find c_k for the 2π-periodic shifted pulse $F(x) = \begin{cases} 1 & \text{for } s \le x \le s + h \\ 0 & \text{elsewhere in } [-\pi, \pi] \end{cases}$

Solution The integrals (26) from $-\pi$ to π become integrals from s to $s + h$:

$$c_k = \frac{1}{2\pi}\int_s^{s+h} 1 \cdot e^{-ikx}\,dx = \frac{1}{2\pi}\left[\frac{e^{-ikx}}{-ik}\right]_s^{s+h} = \boldsymbol{e^{-iks}}\left(\frac{1 - e^{-ikh}}{2\pi ik}\right). \tag{29}$$

Notice above all the simple effect of the shift by s. It "modulates" each c_k by e^{-iks}. The energy is unchanged, the integral of $|F|^2$ just shifts, and all $|e^{-iks}| = 1$:

$$\textbf{Shift} \quad F(x) \quad \text{to} \quad F(x - s) \longleftrightarrow \textbf{Multiply} \quad c_k \quad \text{by} \quad e^{-iks}. \tag{30}$$

Example 7 **Centered pulse** with shift $s = -h/2$. The square pulse becomes centered around $x = 0$. This even function equals 1 on the interval from $-h/2$ to $h/2$:

$$\textbf{Centered by } \boldsymbol{s = -\frac{h}{2}} \qquad c_k = e^{ikh/2}\frac{1 - e^{-ikh}}{2\pi ik} = \frac{\mathbf{1}}{\boldsymbol{2\pi}}\frac{\boldsymbol{\sin(kh/2)}}{\boldsymbol{k/2}}.$$

Divide by h for a tall pulse. The ratio of $\sin(kh/2)$ to $kh/2$ is the **sinc function**:

$$\textbf{Tall pulse} \qquad \frac{F_{\text{centered}}}{h} = \frac{1}{2\pi}\sum_{-\infty}^{\infty} \operatorname{sinc}\left(\frac{kh}{2}\right) e^{ikx} = \begin{cases} 1/h & \text{for } -h/2 \le x \le h/2 \\ 0 & \text{elsewhere in } [-\pi, \pi] \end{cases}$$

That division by h produces area $= 1$. **Every coefficient approaches $\frac{1}{2\pi}$ as $h \to 0$.** The Fourier series for the tall thin pulse again approaches the Fourier series for $\delta(x)$.

Hilbert space can contain vectors $c = (c_0, c_1, c_{-1}, c_2, c_{-2}, \cdots)$ instead of functions $F(x)$. The length of c is $2\pi \sum |c_k|^2 = \int |F|^2 dx$. The function space is often denoted by L^2 and the vector space is ℓ^2. The energy identity is trivial (but deep). Integrating the Fourier series for $F(x)$ times $\overline{F(x)}$, orthogonality kills every $c_n \overline{c_k}$ for $n \neq k$. This leaves the $c_k \overline{c_k} = |c_k|^2$:

$$\int_{-\pi}^{\pi} |F(x)|^2 dx = \int_{-\pi}^{\pi} (\sum c_n e^{inx})(\sum \overline{c_k} e^{-ikx}) dx = 2\pi(|c_0|^2 + |c_1|^2 + |c_{-1}|^2 + \cdots). \quad (31)$$

This is Plancherel's identity: The energy in x-space equals the energy in k-space.

Finally I want to emphasize the three big rules for operating on $F(x) = \sum c_k e^{ikx}$:

1. **The derivative $\dfrac{dF}{dx}$ has Fourier coefficients ikc_k** (energy moves to high k).

2. **The integral of $F(x)$ has Fourier coefficients $\dfrac{c_k}{ik}, k \neq 0$** (faster decay).

3. **The shift to $F(x-s)$ has Fourier coefficients $e^{-iks}c_k$** (no change in energy).

Application: Laplace's Equation in a Circle

Our first application is to Laplace's equation. The idea is to construct $u(x, y)$ as an infinite series, choosing its coefficients to match $u_0(x, y)$ along the boundary. Everything depends on the shape of the boundary, and we take a circle of radius 1.

Begin with the simple solutions $1, r\cos\theta, r\sin\theta, r^2\cos 2\theta, r^2 \sin 2\theta$, ... to Laplace's equation. Combinations of these special solutions give all solutions in the circle:

$$u(r, \theta) = a_0 + a_1 r\cos\theta + b_1 r\sin\theta + a_2 r^2 \cos 2\theta + b_2 r^2 \sin 2\theta + \cdots \quad (32)$$

It remains to choose the constants a_k and b_k to make $u = u_0$ on the boundary. For a circle $u_0(\theta)$ is periodic, since θ and $\theta + 2\pi$ give the same point:

$$\text{Set } r = 1 \qquad u_0(\theta) = a_0 + a_1 \cos\theta + b_1 \sin\theta + a_2 \cos 2\theta + b_2 \sin 2\theta + \cdots \quad (33)$$

This is exactly the Fourier series for u_0. **The constants a_k and b_k must be the Fourier coefficients of $u_0(\theta)$.** Thus the problem is completely solved, if an infinite series (32) is acceptable as the solution.

Example 8 **Point source** $u_0 = \delta(\theta)$ at $\theta = 0$ The whole boundary is held at $u_0 = 0$, except for the source at $x = 1$, $y = 0$. Find the temperature $u(r, \theta)$ inside.

Fourier series for δ $$u_0(\theta) = \frac{1}{2\pi} + \frac{1}{\pi}(\cos\theta + \cos 2\theta + \cos 3\theta + \cdots) = \frac{1}{2\pi} \sum_{-\infty}^{\infty} e^{in\theta}$$

Inside the circle, each $\cos n\theta$ is multiplied by r^n:

Infinite series for u $$u(r,\theta) = \frac{1}{2\pi} + \frac{1}{\pi}(r\cos\theta + r^2\cos 2\theta + r^3\cos 3\theta + \cdots) \quad (34)$$

Poisson managed to sum this infinite series! It involves a series of powers of $re^{i\theta}$. So we know the response at every (r,θ) to the point source at $r = 1$, $\theta = 0$:

Temperature inside circle $$u(r,\theta) = \frac{1}{2\pi}\frac{1-r^2}{1+r^2-2r\cos\theta} \quad (35)$$

At the center $r = 0$, this produces the average of $u_0 = \delta(\theta)$ which is $a_0 = 1/2\pi$. On the boundary $r = 1$, this produces $u = 0$ except at the point source where $\cos 0 = 1$:

On the ray $\theta = 0$ $$u(r,\theta) = \frac{1}{2\pi}\frac{1-r^2}{1+r^2-2r} = \frac{1}{2\pi}\frac{1+r}{1-r}. \quad (36)$$

As r approaches 1, the solution becomes infinite as the point source requires.

Example 9 Solve for any boundary values $u_0(\theta)$ by integrating over point sources.

When the point source swings around to angle φ, the solution (35) changes from θ to $\theta - \varphi$. Integrate this "Green's function" to solve in the circle:

Poisson's formula $$u(r,\theta) = \frac{1}{2\pi}\int_{-\pi}^{\pi} u_0(\varphi)\,\frac{1-r^2}{1+r^2-2r\cos(\theta-\varphi)}\,d\varphi \quad (37)$$

Ar $r = 0$ the fraction disappears and u is the average $\int u_0(\varphi)d\varphi/2\pi$. The steady state temperature at the center is the average temperature around the circle.

Poisson's formula illustrates a key idea. Think of any $u_0(\theta)$ as a circle of point sources. The source at angle $\varphi = \theta$ produces the solution inside the integral (37). Integrating around the circle adds up the responses to all sources and gives the response to $u_0(\theta)$.

Example 10 $u_0(\theta) = 1$ on the top half of the circle and $u_0 = -1$ on the bottom half.

Solution The boundary values are the square wave $SW(\theta)$. Its sine series is in (8):

Square wave for $u_0(\theta)$ $$SW(\theta) = \frac{4}{\pi}\left[\frac{\sin\theta}{1} + \frac{\sin 3\theta}{3} + \frac{\sin 5\theta}{5} + \cdots\right] \quad (38)$$

Inside the circle, multiplying by r, r^2, r^3,... gives fast decay of high frequencies:

Rapid decay inside $$u(r,\theta) = \frac{4}{\pi}\left[\frac{r\sin\theta}{1} + \frac{r^3\sin 3\theta}{3} + \frac{r^5\sin 5\theta}{5} + \cdots\right] \quad (39)$$

Laplace's equation has smooth solutions, even when $u_0(\theta)$ is not smooth.

▫ WORKED EXAMPLE ▫

A hot metal bar is moved into a freezer (zero temperature). The sides of the bar are coated so that heat only escapes at the ends. *What is the temperature* $u(x,t)$ along the bar at time t? It will approach $u = 0$ as all the heat leaves the bar.

Solution The heat equation is $u_t = u_{xx}$. At $t = 0$ the whole bar is at a constant temperature, say $u=1$. The ends of the bar are at zero temperature for all time $t>0$. This is an **initial-boundary value problem**:

Heat equation $u_t = u_{xx}$ with $u(x,0) = 1$ and $u(0,t) = u(\pi,t) = 0$. (40)

Those zero boundary conditions suggest a sine series. Its coefficients depend on t:

Series solution of the heat equation $$u(x,t) = \sum_1^\infty b_n(t)\,\sin nx. \tag{41}$$

The form of the solution shows **separation of variables**. In a comment below, we look for products $A(x)\,B(t)$ that solve the heat equation and the boundary conditions. What we reach is exactly $A(x) = \sin nx$ and the series solution (41).

Two steps remain. First, choose each $b_n(t)\sin nx$ to satisfy the heat equation:

Substitute into $u_t = u_{xx}$ $b_n'(t)\sin nx = -n^2 b_n(t)\,\sin nx \quad b_n(t) = e^{-n^2 t}\,b_n(0).$

Notice $b_n' = -n^2 b_n$. Now determine each $b_n(0)$ from the initial condition $u(x,0) = 1$ on $(0,\pi)$. Those numbers are the Fourier sine coefficients of $SW(x)$ in equation (38):

Box function/square wave $$\sum_1^\infty b_n(0)\,\sin nx = 1 \qquad b_n(0) = \frac{4}{\pi n} \text{ for odd } n$$

This completes the series solution of the initial-boundary value problem:

Bar temperature $$u(x,t) = \sum_{\text{odd } n} \frac{4}{\pi n}\,e^{-n^2 t}\,\sin nx. \tag{42}$$

For large n (high frequencies) the decay of $e^{-n^2 t}$ is very fast. The dominant term $(4/\pi)e^{-t}\sin x$ for large times will come from $n = 1$. This is typical of the heat equation and all diffusion, that the solution (the temperature profile) becomes very smooth as t increases.

Numerical difficulty I regret any bad news in such a beautiful solution. To compute $u(x,t)$, we would probably truncate the series in (42) to N terms. When that finite series is graphed on the website, serious bumps appear in $u_N(x,t)$. You ask if there is a physical reason but there isn't. The solution should have maximum temperature at the midpoint $x = \pi/2$, and decay smoothly to zero at the ends of the bar.

Those unphysical bumps are precisely the **Gibbs phenomenon**. The initial $u(x,0)$ is 1 on $(0,\pi)$ but its odd reflection is -1 on $(-\pi,0)$. That jump has produced the slow $4/\pi n$ decay of the coefficients, with Gibbs oscillations near $x=0$ and $x=\pi$. The sine series for $u(x,t)$ is not a success numerically. Would finite differences help?

Separation of variables We found $b_n(t)$ as the coefficient of an eigenfunction $\sin nx$. Another good approach is to put $u = A(x)\,B(t)$ directly into $u_t = u_{xx}$:

$$\textbf{Separation} \quad A(x)\,B'(t) = A''(x)\,B(t) \text{ requires } \frac{A''(x)}{A(x)} = \frac{B'(t)}{B(t)} = \textbf{constant}. \tag{43}$$

A''/A is constant in space, B'/B is constant in time, and they are equal:

$$\frac{A''}{A} = -\lambda \text{ gives } A = \sin\sqrt{\lambda}\,x \text{ and } \cos\sqrt{\lambda}\,x \qquad \frac{B'}{B} = -\lambda \text{ gives } B = e^{-\lambda t}$$

The products $AB = e^{-\lambda t}\sin\sqrt{\lambda}\,x$ and $e^{-\lambda t}\cos\sqrt{\lambda}\,x$ solve the heat equation for any number λ. But the boundary condition $u(0,t) = 0$ eliminates the cosines. Then $u(\pi,t) = 0$ requires $\lambda = n^2 = 1, 4, 9, \ldots$ to have $\sin\sqrt{\lambda}\,\pi = 0$. Separation of variables has recovered the functions in the series solution (42).

Finally $u(x,0) = 1$ determines the numbers $4/\pi n$ for odd n. We find zero for even n because $\sin nx$ has $n/2$ positive loops and $n/2$ negative loops. For odd n, the extra positive loop is a fraction $1/n$ of all loops, giving slow decay of the coefficients.

Heat bath (the opposite problem) The solution on the website is $1 - u(x,t)$, because it solves a different problem. **The bar is initially frozen at** $U(x,0) = 0$. It is placed into a heat bath at the fixed temperature $U = 1$ (or $U = T_0$). The new unknown is U and its boundary conditions are no longer zero.

The heat equation and its boundary conditions are solved first by $U_B(x,t)$. In this example $U_B \equiv 1$ is constant. Then the difference $V = U - U_B$ has zero boundary values, and its initial values are $V = -1$. Now the eigenfunction method (or separation of variables) solves for V. (The series in (42) is multiplied by -1 to account for $V(x,0) = -1$.) Adding back U_B solves the heat bath problem: $U = U_B + V = 1 - u(x,t)$.

Here $U_B \equiv 1$ is the *steady state* solution at $t = \infty$, and V is the *transient* solution. The transient starts at $V = -1$ and decays quickly to $V = 0$.

Heat bath at one end The website problem is different in another way too. The Dirichlet condition $u(\pi,t) = 1$ is replaced by the Neumann condition $u'(1,t) = 0$. Only the left end is in the heat bath. Heat flows down the metal bar and out at the far end, now located at $x = 1$. How does the solution change for fixed-free?

Again $U_B = 1$ is a steady state. The boundary conditions apply to $V = 1 - U_B$:

$$\textbf{Fixed-free eigenfunctions} \quad V(0) = 0 \text{ and } V'(1) = 0 \text{ lead to } A(x) = \sin\left(n + \frac{1}{2}\right)\pi x. \tag{44}$$

Those eigenfunctions give a new form for the sum of $B_n(t)\,A_n(x)$:

Fixed-free solution
$$V(x,t) = \sum_{\text{odd } n} B_n(0)\, e^{-(n+\frac{1}{2})^2\pi^2 t} \sin\left(n+\frac{1}{2}\right)\pi x. \tag{45}$$

All frequencies shift by $\frac{1}{2}$ and multiply by π, because $A'' = -\lambda A$ has a free end at $x = 1$. The crucial question is: **Does orthogonality still hold for** these new eigenfunctions $\sin\left(n+\frac{1}{2}\right)\pi x$ **on** $[0,1]$? The answer is *yes* because this fixed-free "Sturm–Liouville problem" $A'' = -\lambda A$ is still symmetric.

Summary The series solutions all succeed but the truncated series all fail. We can see the overall behavior of $u(x,t)$ and $V(x,t)$. But their exact values close to the jumps are not computed well until we improve on Gibbs.

We could have solved the fixed-free problem on $[0,1]$ with the fixed-fixed solution on $[0,2]$. That solution will be symmetric around $x = 1$ so its slope there is zero. Then rescaling x by 2π changes $\sin(n+\frac{1}{2})\pi x$ into $\sin(2n+1)x$. I hope you like the graphics created by Aslan Kasimov on the cse website.

Problem Set 4.1

1 Find the Fourier series on $-\pi \le x \le \pi$ for

(a) $f(x) = \sin^3 x$, an odd function
(b) $f(x) = |\sin x|$, an even function
(c) $f(x) = x$
(d) $f(x) = e^x$, using the complex form of the series.

What are the even and odd parts of $f(x) = e^x$ and $f(x) = e^{ix}$?

2 From Parseval's formula the square wave sine coefficients satisfy
$$\pi(b_1^2 + b_2^2 + \cdots) = \int_{-\pi}^{\pi} |f(x)|^2\, dx = \int_{-\pi}^{\pi} 1\, dx = 2\pi.$$
Derive the remarkable sum $\pi^2 = 8(1 + \frac{1}{9} + \frac{1}{25} + \cdots)$.

3 If a square pulse is centered at $x = 0$ to give
$$f(x) = 1 \quad \text{for} \quad |x| < \frac{\pi}{2}, f(x) = 0 \quad \text{for} \quad \frac{\pi}{2} < |x| < \pi,$$
draw its graph and find its Fourier coefficients a_k and b_k.

4 Suppose f has period T instead of $2x$, so that $f(x) = f(x+T)$. Its graph from $-T/2$ to $T/2$ is repeated on each successive interval and its real and complex Fourier series are
$$f(x) = a_0 + a_1 \cos\frac{2\pi x}{T} + b_1 \sin\frac{2\pi x}{T} + \cdots = \sum_{-\infty}^{\infty} c_k\, e^{ik2\pi x/T}$$
Multiplying by the right functions and integrating from $-T/2$ to $T/2$, find a_k, b_k, and c_k.

5 Plot the first three partial sums and the function itself:

$$x(\pi - x) = \frac{8}{\pi}\left(\frac{\sin x}{1} + \frac{\sin 3x}{27} + \frac{\sin 5x}{125} + \cdots\right), 0 < x < \pi.$$

Why is $1/k^3$ the decay rate for this function? What is the second derivative?

6 What constant function is closest in the least square sense to $f = \cos^2 x$? What multiple of $\cos x$ is closest to $f = \cos^3 x$?

7 Sketch the 2π-periodic half wave with $f(x) = \sin x$ for $0 < x < \pi$ and $f(x) = 0$ for $-\pi < x < 0$. Find its Fourier series.

8 (a) Find the lengths of the vectors $u = (1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \ldots)$ and $v = (1, \frac{1}{3}, \frac{1}{9}, \ldots)$ in Hilbert space and test the Schwarz inequality $|u^{\mathrm{T}}v|^2 \le (u^{\mathrm{T}}u)(v^{\mathrm{T}}v)$.

(b) For the functions $f = 1 + \frac{1}{2}e^{ix} + \frac{1}{4}e^{2ix} + \cdots$ and $g = 1 + \frac{1}{3}e^{ix} + \frac{1}{9}e^{2ix} + \cdots$ use part (a) to find the numerical value of each term in

$$\left|\int_{-\pi}^{\pi} \overline{f}(x)\, g(x)\, dx\right|^2 \le \int_{-\pi}^{\pi} |f(x)|^2\, dx \int_{-\pi}^{\pi} |g(x)|^2\, dx.$$

Substitute for f and g and use orthogonality (or Parseval).

9 Find the solution to Laplace's equation with $u_0 = \theta$ on the boundary. Why is this the imaginary part of $2(z - z^2/2 + z^3/3 \cdots) = 2\log(1 + z)$? Confirm that on the unit circle $z = e^{i\theta}$, the imaginary part of $2\log(1 + z)$ agrees with θ.

10 If the boundary condition for Laplace's equation is $u_0 = 1$ for $0 < \theta < \pi$ and $u_0 = 0$ for $-\pi < \theta < 0$, find the Fourier series solution $u(r, \theta)$ inside the unit circle. What is u at the origin?

11 With boundary values $u_0(\theta) = 1 + \frac{1}{2}e^{i\theta} + \frac{1}{4}e^{2i\theta} + \cdots$, what is the Fourier series solution to Laplace's equation in the circle? Sum the series.

12 (a) Verify that the fraction in Poisson's formula satisfies Laplace's equation.

(b) What is the response $u(r, \theta)$ to an impulse at the point $(0, 1)$, at the angle $\varphi = \pi/2$?

(c) If $u_0(\varphi) = 1$ in the quarter-circle $0 < \varphi < \pi/2$ and $u_0 = 0$ elsewhere, show that at points on the horizontal axis (and especially at the origin)

$$u(r, 0) = \frac{1}{2} + \frac{1}{2\pi}\tan^{-1}\left(\frac{1 - r^2}{-2r}\right) \quad \text{by using}$$

$$\int \frac{d\varphi}{b + c\cos\varphi} = \frac{1}{\sqrt{b^2 - c^2}}\tan^{-1}\left(\frac{\sqrt{b^2 - c^2}\,\sin\varphi}{c + b\cos\varphi}\right).$$

13 When the centered square pulse in Example 7 has width $h = \pi$, find

(a) its energy $\int |F(x)|^2\,dx$ by direct integration

(b) its Fourier coefficients c_k as specific numbers

(c) the sum in the energy identity (31) or (24)

If $h = 2\pi$, why is $c_0 = 1$ the only nonzero coefficient? What is $F(x)$?

14 In Example 5, $F(x) = 1 + (\cos x)/2 + \cdots + (\cos nx)/2^n + \cdots$ is infinitely smooth:

(a) If you take 10 derivatives, what is the Fourier series of $d^{10}F/dx^{10}$?

(b) Does that series still converge quickly? Compare n^{10} with 2^n for n^{1024}.

15 (*A touch of complex analysis*) The analytic function in Example 5 blows up when $4\cos x = 5$. This cannot happen for real x, but equation (28) shows blowup if $e^{ix} = 2$ or $\frac{1}{2}$. In that case we have *poles at* $x = \pm i \log 2$. Why are there also poles at all the complex numbers $x = \pm i \log 2 + 2\pi n$?

16 (*A second touch*) Change 2's to 3's so that equation (28) has $1/(3 - e^{ix}) + 1/(3 - e^{-ix})$. Complete that equation to find the function that gives fast decay at the rate $1/3^k$.

17 (*For complex professors only*) Change those 2's and 3's to 1's:

$$\frac{1}{1 - e^{ix}} + \frac{1}{1 - e^{-ix}} = \frac{(1 - e^{-ix}) + (1 - e^{ix})}{(1 - e^{ix})(1 - e^{-ix})} = \frac{2 - e^{ix} - e^{-ix}}{2 - e^{ix} - e^{-ix}} = 1\,.$$

A constant! What happened to the pole at $e^{ix} = 1$? Where is the dangerous series $(1 + e^{ix} + \cdots) + (1 + e^{-ix} + \cdots) = 2 + 2\cos x + \cdots$ involving $\delta(x)$?

18 Following the Worked Example, solve the heat equation $u_t = u_{xx}$ from a point source $u(x, 0) = \delta(x)$ with free boundary conditions $u'(\pi, t) = u'(-\pi, t) = 0$. Use the infinite cosine series for $\delta(x)$ with time decay factors $b_n(t)$.

4.2 CHEBYSHEV, LEGENDRE, AND BESSEL

The sines and cosines are orthogonal on $[-\pi, \pi]$, but not by accident. Those zeros in a table of definite integrals are not lucky chances. The real reason for this orthogonality is that $\sin kx$ and $\cos kx$ are the *eigenfunctions of a symmetric operator.* So are the exponentials e^{ikx}, when d^2/dx^2 has periodic boundary conditions.

Symmetric operators have orthogonal eigenfunctions. This section looks at the eigenfunctions of other symmetric operators. They give new and important families of orthogonal functions, named after their discoverers.

Two-dimensional Fourier Series

In 2D, the Laplacian is $L = \partial^2/\partial x^2 + \partial^2/\partial y^2$. The orthogonal functions are $e^{inx}e^{imy}$ (every e^{inx} times every e^{imy}). Those are eigenfunctions of L since $Le^{inx}e^{imy}$ produces $(-n^2 - m^2)e^{inx}e^{imy}$. We have **separation of variables** (x is separated from y):

Double Fourier series
$$F(x,y) = \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} c_{nm}\, e^{inx} e^{imy}. \tag{1}$$

These functions are periodic in x and also in y: $F(x+2\pi, y) = F(x, y+2\pi) = F(x,y)$. We check that $e^{inx}e^{imy}$ is orthogonal to $e^{ikx}e^{i\ell y}$ on a square $-\pi \le x \le \pi, -\pi \le y \le \pi$. The double integral separates into x and y integrals that we know are zero:

Orthogonality
$$\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} \left(e^{inx}e^{imy}\right)\left(e^{-ikx}e^{-i\ell y}\right)\, dx\, dy = 0 \quad \text{unless} \quad \begin{matrix} n = k \\ m = \ell \end{matrix} \tag{2}$$

The Fourier coefficient $c_{k\ell}$ comes from multiplying the series (1) by $e^{-\imath kx}e^{-\imath \ell y}$ and integrating over the square. One term survives and the formula looks familiar:

Double Fourier coefficients
$$c_{k\ell} = \left(\frac{1}{2\pi}\right)^2 \int_{-\pi}^{\pi}\int_{-\pi}^{\pi} F(x,y)\, e^{-ikx}e^{-i\ell y}\, dx\, dy. \tag{3}$$

The two-dimensional delta function $\delta(x)\delta(y)$ (made periodic) has all $c_{k\ell} = (1/2\pi)^2$.

Separating x from y simplifies the calculation of $c_{k\ell}$. The integral of $F(x,y)e^{-ikx}dx$ is a one-dimensional transform for each y. The result depends on y and k. Then multiply by $e^{-i\ell y}$ and integrate between $y = -\pi$ and $y = \pi$, to find $c_{k\ell}$.

I see this separation of variables in processing a square image. The x-transform goes along each row of pixels. Then the output is ordered by columns, and the y-transform goes down each column. *The two-dimensional transform is computed by one-dimensional software.* In practice the pixels are equally spaced and the computer is adding instead of integrating—the Discrete Fourier Transform in the next section is a sum at N equally spaced points. The DFT in 2D has N^2 points.

A Delta Puzzle

The two-dimensional $\delta(x)\delta(y)$ is concentrated at $(0,0)$. It is defined by integration:

Delta in 2D
$$\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} \delta(x)\delta(y)\, G(x,y)\, dx\, dy = G(0,0) \quad \text{for any smooth } G(x,y). \tag{4}$$

Choosing $G \equiv 1$ confirms "area = 1" under the spike. Choosing $G = e^{-ikx}e^{-i\ell y}$ confirms that all Fourier coefficients $c_{k\ell}$ are $1/4\pi^2$ for $\delta(x)\delta(y)$, since $G(0,0) = 1$:

Delta function
$$\delta(x)\delta(y) = \left(\sum \frac{e^{ikx}}{2\pi}\right)\left(\sum \frac{e^{i\ell y}}{2\pi}\right) = \frac{1}{4\pi^2}\sum\sum e^{ikx}e^{i\ell y}. \tag{5}$$

Now try a *vertical line of spikes* $F(x,y) = \delta(x)$. They go up the y-axis, where $x = 0$. Every horizontal x-integral crosses that line at $x = 0$, and picks out $G(0,y)$:

Line of spikes $\delta(x)$
$$\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} \delta(x)\, G(x,y)\, dx\, dy = \int_{-\pi}^{\pi} G(0,y)\, dy. \tag{6}$$

Choosing $G \equiv 1$ gives "area $= 2\pi$" under this line of spikes. The line has length 2π. Integrating $\delta(x)\, e^{-ikx}e^{-i\ell y}$ gives $c_{k\ell} = 0$ when $\ell \neq 0$, since $\int_{-\pi}^{\pi} e^{-i\ell y}\, dy = 0$. So the two-dimensional series for $F(x,y) = \delta(x)$ is really one-dimensional. So far no puzzle:

Spikes along $x = 0$
$$\delta(x) = \sum_{\ell}\sum_{k} c_{k\ell}e^{-ikx}e^{-i\ell y} = \sum_{k}\left(\frac{1}{2\pi}\right)e^{-ikx}. \tag{7}$$

The puzzle comes for a diagonal line of spikes $\delta(x+y)$. Let me ask, what is the area under the spikes along the line $x + y = 0$? This line runs from $x = -\pi, y = \pi$ diagonally down to the opposite corner $x = \pi, y = -\pi$. Its length is 2π times $\sqrt{2}$. For a "unit delta" I am now expecting area $= 2\pi\sqrt{2}$. But I don't see that $\sqrt{2}$ in the double integral. Each x-integral just meets a spike at $x = -y$:

$\sqrt{2}$ disappears
$$\textbf{Area} = \int_{-\pi}^{\pi}\left[\int_{-\pi}^{\pi} \delta(x+y)\, dx\right] dy = \int_{-\pi}^{\pi} 1\, dy = 2\pi. \tag{8}$$

But the area under a finite diagonal pulse (width h and height $1/h$) does include the $\sqrt{2}$ factor. It stays there as $h \to 0$. Diagonal lines must be thicker than I thought!

Maybe I should change variables in (8) to $X = x + y$. After many suggestions from students and faculty, this is what I believe (for now):

> *Everybody is right.* The function $\delta(x+y)$ has area 2π. Its Fourier series, including the parallel spikes $\delta(x+y-2\pi n)$ to be periodic, is $\frac{1}{2\pi}\sum e^{ik(x+y)}$. But this is not the unit spike I thought it was. The unit spike along the diagonal is a different function $\delta((x+y)/\sqrt{2})$. That one has area $2\pi\sqrt{2}$.

Dividing $x + y$ by $\sqrt{2}$ leads me to think about $\delta(2x)$. Its values are "zero or infinity" but actually **$\delta(2x)$ is half of $\delta(x)$**! For delta functions, the right way to

understand $\delta(2x)$ is by integration with a smooth function G. Set $2x = t$:

$$\boldsymbol{\delta(2x) = \tfrac{1}{2}\delta(x)} \qquad \int_{-\infty}^{\infty} \delta(2x)G(x)\,dx = \int_{-\infty}^{\infty} \delta(t)G(t/2)\,dt/2 = \frac{1}{2}G(0)\,. \tag{9}$$

The area under $\delta(2x)$ is $\frac{1}{2}$. Multiplied by any $G(x)$, this half-width spike produces $\frac{1}{2}G(0)$ in integration. Similarly $\delta((x+y)/\sqrt{2}) = \sqrt{2}\,\delta(x+y)$.

In three dimensions we could have a point spike $\delta(x)\delta(y)\delta(z)$, or a line of spikes $\delta(x)\delta(y)$ along the z-axis, or a horizontal plane $x = y = 0$ of one-dimensional spikes $\delta(z)$. Physically, those represent a point source f or a line source or a plane source. They can all appear in Poisson's equation $u_{xx} + u_{yy} + u_{zz} = f(x, y, z)$.

Chebyshev Polynomials

Start with the cosines 1, $\cos\theta$, $\cos 2\theta$, $\cos 3\theta, \ldots$ and ***change from $\cos\theta$ to x***. The first Chebyshev polynomials are $T_0 = 1$ and $T_1 = x$. The next polynomials T_2 and T_3 come from identities for $\cos 2\theta$ and $\cos 3\theta$:

$$\boldsymbol{x = \cos\theta} \qquad \begin{matrix} \cos 2\theta = 2\cos^2\theta - 1 \\ \cos 3\theta = 4\cos^3\theta - 3\cos\theta \end{matrix} \qquad \begin{matrix} T_2(x) = \boldsymbol{2x^2 - 1} \\ T_3(x) = \boldsymbol{4x^3 - 3x} \end{matrix}$$

These $T_k(x)$ are certain to be important, because the cosines are so important. There is a neat way to find T_{k+1} from the previous T_k and T_{k-1}, by using cosines:

Cosine identity $$\cos(k+1)\theta + \cos(k-1)\theta = 2\cos\theta\cos k\theta \tag{10}$$

Chebyshev recursion $$\boldsymbol{T_{k+1}(x) + T_{k-1}(x) = 2\,x\,T_k(x)} \tag{11}$$

Figure 4.6 shows the even polynomials $T_2(x)$ and $T_4(x) = 2xT_3(x) - T_2(x) = \cos 4\theta$ (four zeros).

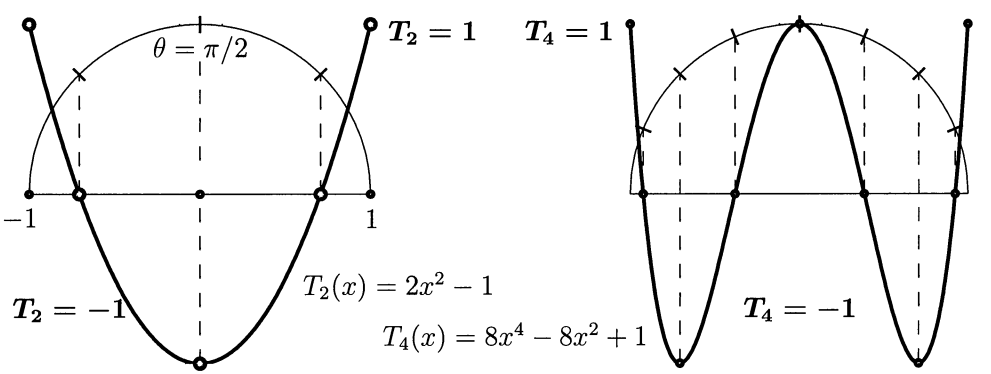


Figure 4.6: Chebyshev polynomials oscillate between 1 and -1 (because cosines do).

The quick formula $T_k(x) = \cos k\theta = \cos(k \cos^{-1} x)$ looks a little awkward because it involves the *arc cosine* (inverse to the cosine function):

Chebyshev polynomials

$$x = \cos\theta \quad \text{and} \quad \theta = \cos^{-1} x \qquad T_k(x) = \cos k\theta = \cos(k \cos^{-1} x)\,. \tag{12}$$

The cosine of $k\theta$ reaches top and bottom at the $k+1$ angles $\theta = 2\pi j/k$. Those angles give $\cos k\theta = \cos 2\pi j = \pm 1$. On the x-axis, those are the ***Chebyshev points*** where $T_k(x)$ has its maximum $+1$ or its minimum -1:

Chebyshev points

$$x_j = \cos\frac{2\pi j}{k} \quad \text{for} \quad j = 0, 1, \ldots, k \tag{13}$$

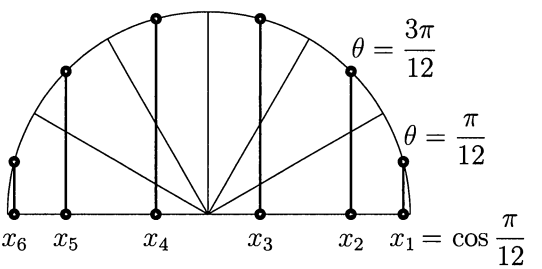
These points will play a central role in Section 5.4 on computational complex analysis. Evaluating a function $f(x)$ at the Chebyshev points is the same as evaluating $f(\cos\theta)$ at the equally spaced angles $\theta = 2\pi j/k$. These calculations use the Fast Fourier Transform to give an exponential convergence rate at top numerical speed.

The **zeros of the Chebyshev polynomials** lie between the Chebyshev points. We know that $\cos k\theta = 0$ when $k\theta$ is an odd multiple of $\pi/2$. Figure 4.7 shows those angles $\theta_1, \ldots, \theta_k$ equally spaced around a unit semicircle. To keep the correspondence $x = \cos\theta$, we just take the x-coordinates by dropping perpendicular lines. Those lines meet the x-axis at the zeros $x_1, \ldots, x_k$ of the Chebyshev polynomials:

Zeros of $T_k(x)$

$$\begin{aligned} &\cos k\theta = 0 \quad \text{if} \quad k\theta = (2j-1)\frac{\pi}{2} \\ &T_k(x_j) = 0 \quad \text{if} \quad x_j = \cos\theta_j = \cos\left[\frac{2j-1}{2}\,\frac{\pi}{k}\right]. \end{aligned} \tag{14}$$

Notice the spacing of Chebyshev points and zeros along the interval $-1 \le x \le 1$. More space at the center, more densely packed near the ends. This irregular spacing is infinitely better than equal spacing, if we want to fit a polynomial through the points.



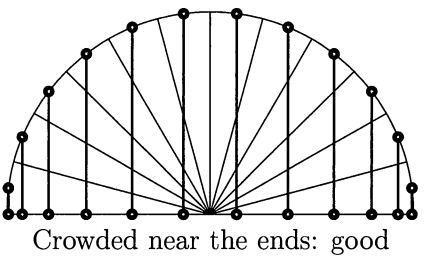


Figure 4.7: The six solutions $x = \cos\theta$ of $T_6(x) = \cos 6\theta = 0$. Twelve for $T_{12}(x)$.

Are the Chebyshev polynomials orthogonal over $-1 \le x \le 1$? The answer is "yes" but only if you include their **weight function** $1/\sqrt{1-x^2}$ in the integral.

Weighted Orthogonality

The Chebyshev weight $w(x) = 1/\sqrt{1-x^2}$ comes from $dx = -\sin\theta\, d\theta = -\sqrt{1-x^2}\, d\theta$:

Weighted orthogonality
$$\int_{-1}^{1} T_n(x)T_k(x)\frac{dx}{\sqrt{1-x^2}} = \int_0^{\pi} \cos n\theta \cos k\theta\, d\theta = 0\,. \tag{15}$$

The sines and cosines had weight function $w(x) = 1$. The Chebyshev polynomials have $w(x) = 1/\sqrt{1-x^2}$. There are Bessel functions on $[0,1]$ with $w(x) = x$ and Laguerre polynomials on $[0,\infty)$ with $w(x) = e^{-x}$ and Hermite polynomials on $(-\infty,\infty)$ with $w(x) = e^{-x^2/2}$. These weights are *never negative.* For all of those orthogonal functions, the coefficients in $F(x) = \sum c_k T_k(x)$ are found the same way:

Multiply $F(x) = c_0T_0(x) + c_1T_1(x) + \cdots$ *by* $T_k(x)$ *and also* $w(x)$. *Then integrate:*

$$\int F(x)T_k(x)w(x)\,dx = \int c_0T_0(x)T_k(x)w(x)\,dx + \cdots + \int \boldsymbol{c_k(T_k(x))^2 w(x)\,dx} + \cdots$$

On the right side only the boldface term survives, as in Fourier series. There T_k was $\cos kx$, the weight was $w(x) = 1$, and the integral from $-\pi$ to π was π times c_k. For any orthogonal functions, division leaves the formula for c_k:

Orthogonal $T_k(x)$ Weight $w(x)$ Coefficients c_k $\displaystyle = \frac{\int F(x)T_k(x)w(x)\,dx}{\int (T_k(x))^2 w(x)\,dx}$

The positive weight $w(x)$ is like the positive definite mass matrix in $Ku = \lambda Mu$. The eigenvectors u_k are orthogonal when weighted by M: for example $u_1^{\mathrm{T}}Mu_2 = 0$. In the continuous case, the $T_k(x)$ are **eigenfunctions** and $\int T_k(x)T_n(x)w(x)\,dx = 0$.

The eigenvalue problem could be Chebyshev's or Legendre's or Bessel's equation, with weight $w = 1/\sqrt{1-x^2}$ or $w = 1$ or $w = x$. The Chebyshev polynomials are seen as eigenfunctions, by changing from the cosines in $-d^2u/d\theta^2 = \lambda u$:

Chebyshev equation
$$-\frac{d}{dx}\left(\frac{1}{w}\frac{dT}{dx}\right) = \lambda\, w(x)T(x)\,. \tag{16}$$

That operator on the left looks to me like $A^{\mathrm{T}}CA$. This is a continuous $KT = \lambda MT$.

Legendre Polynomials

The direct way to the Legendre polynomials $P_n(x)$ is to start with $1, x, x^2, \ldots$ on the interval $[-1,1]$ with weight $w(x) = 1$. Those functions are *not orthogonal.* The integral of 1 times x is $\int_{-1}^{1} x\,dx = 0$, but the integral of 1 times x^2 is $\int_{-1}^{1} x^2\,dx = \frac{2}{3}$. The **Gram-Schmidt idea** will produce orthogonal functions out of $1, x$, and x^2:

Subtract from x^2 *its component* $\frac{1}{3}$ *in the direction of* 1. *Then* $\int_{-1}^{1}(x^2 - \frac{1}{3})\,1\,dx = 0$.

This Legendre polynomial $P_2(x) = x^2 - \frac{1}{3}$ is also orthogonal to the odd $P_1(x) = x$.

For $P_3(x)$, subtract from x^3 its component $\frac{3}{5}x$ in the direction of $P_1(x) = x$. Then $\int (x^3 - \frac{3}{5}x)\, x\, dx = 0$. Gram-Schmidt subtracts from each new x^n the right multiples of $P_0(x), \ldots, P_{n-1}(x)$. The convention is that every $P_n(x)$ equals 1 at $x = 1$, so we rescale P_2 and P_3 to their final form $\frac{1}{2}(3x^2 - 1)$ and $\frac{1}{2}(5x^3 - 3x)$.

I cannot miss telling you the beautiful formula and *three-term recurrence* for $P_n(x)$:

Rodrigues formula
$$P_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n} (x^2 - 1)^n \tag{17}$$

Three-term recurrence
$$P_n(x) = \frac{2n-1}{n}\, x\, P_{n-1}(x) - \frac{n-1}{n}\, P_{n-2}(x) \tag{18}$$

The key point about (18) is *automatic orthogonality* to all lower-degree polynomials. Gram-Schmidt stops early, which makes the calculations very efficient. The right side has $\int x\, P_{n-1} P_{n-3}\, dx = 0$, because x times P_{n-3} only has degree $n-2$. Therefore xP_{n-3} is orthogonal to P_{n-1}, and so is P_{n-2}. Then P_n from (18) is orthogonal to P_{n-3}.

The same three term recurrence appears in the discrete case, when Arnoldi orthogonalizes $b, Ab, \ldots, A^{n-1}b$ in Section 7.4. The orthogonal vectors lead to the "conjugate gradient method." For Legendre, $b \equiv 1$ and A is multiplication by x.

Bessel Functions

For a square, the right coordinate system is x, y. A double Fourier series with $e^{inx} e^{imy}$ is perfect. On a circle, polar coordinates r, θ are much better. If $u(r, \theta)$ depends only on the angle θ, sines and cosines are good. But if u depends on r *only* (the quite common case of radial symmetry) then new functions are needed.

The best example is a circular drum. If you strike it, it oscillates. Its motion contains a mixture of "pure" oscillations at single frequencies. The goal is to discover those natural frequencies (eigenvalues) of the drum and the shapes (eigenfunctions) of the drumhead. The problem is governed by Laplace's equation in polar coordinates:

Laplace equation in r, θ
$$\frac{\partial^2 u}{\partial r^2} + \frac{1}{r}\frac{\partial u}{\partial r} + \frac{1}{r^2}\frac{\partial^2 u}{\partial \theta^2} = -\lambda u\,. \tag{19}$$

The boundary condition is $u = 0$ at $r = 1$; the outside of the drum is fastened. The natural idea is to separate r from θ, and to look for eigenfunctions of the special form $u = A(\theta)B(r)$. This is **separation of variables**. It needs an exceptional geometry, and the circle is exceptional. Separation of variables will reach an ordinary differential equation for $A(\theta)$ and another one for $B(r)$.

Substitute $u = A(\theta)B(r)$
$$AB'' + \frac{1}{r}AB' + \frac{1}{r^2}A''B = -\lambda AB\,. \tag{20}$$

Now multiply by r^2 and divide by AB. *The key point is to separate r from θ*:

Separated variables
$$\frac{r^2 B'' + rB' + \lambda r^2 B}{B} = -\frac{A''(\theta)}{A(\theta)} = \textit{constant}\,. \tag{21}$$

The left side depends only on r. But $A''(\theta)/A(\theta)$ is independent of r. *Both sides must be constant.* If the constant is n^2, then on the right $A'' = -n^2 A$. This gives $A(\theta)$ as $\sin n\theta$ and $\cos n\theta$. Especially it requires n to be an integer. The solution must have the same value at $\theta = 0$ and $\theta = 2\pi$, since those are the same points on the circle.

The left side of (21) now gives an ordinary differential equation for $B = B_n(r)$:

Bessel's equation
$$r^2 B'' + rB' + \lambda r^2 B = n^2 B, \quad \text{with } B(1) = 0. \tag{22}$$

The eigenfunctions $u = A(\theta)B(r)$ of the Laplacian will be $\sin n\theta B_n(r)$ and $\cos n\theta B_n(r)$.

Solving Bessel's equation (22) is not easy. The direct approach looks for an infinite series $B(r) = \sum c_m r^m$. This technique can fill a whole chapter, which I frankly think is unreasonable (you could find it on the web). We construct only one power series, in the radially symmetric case $n = 0$ with no dependence on θ, to see what a Bessel function looks like. Substitute $\boldsymbol{B = \sum c_m r^m}$ into $r^2 B'' + rB' + \lambda r^2 B = 0$:

$$\sum c_m m(m-1) r^m + \sum c_m m r^m + \lambda \sum c_m r^{m+2} = 0. \tag{23}$$

The third sum multiplies r^m by λc_{m-2}. *Compare the coefficients of each* r^m:

c_m from c_{m-2}
$$c_m m(m-1) + c_m m + \lambda c_{m-2} = 0. \tag{24}$$

In other words $m^2 c_m = -\lambda c_{m-2}$. Suppose $c_0 = 1$. This recursion gives $c_2 = -\lambda/2^2$. Then c_4 is $-\lambda/4^2$ times c_2. Each step gives one more coefficient in the series for B:

Bessel function
$$B(r) = c_0 + c_2 r^2 + \cdots = 1 - \frac{\lambda r^2}{2^2} + \frac{\lambda^2 r^4}{2^2 4^2} - \frac{\lambda^3 r^6}{2^2 4^2 6^2} + \cdots \tag{25}$$

This is a Bessel function of order $n = 0$. Its standard notation is $B = \boldsymbol{J_0(\sqrt{\lambda}\, r)}$. The eigenvalues λ come from $J_0(\sqrt{\lambda}) = 0$ at the boundary $r = 1$. The best way to appreciate these functions is by comparison with the cosine, whose behavior we know:

$$\cos(\sqrt{\lambda}) = 1 - \frac{\lambda}{2!} + \frac{\lambda^2}{4!} - \frac{\lambda^3}{6!} + \cdots \quad \text{and} \quad \boldsymbol{J_0(\sqrt{\lambda})} = 1 - \frac{\lambda}{2^2} + \frac{\lambda^2}{2^2 4^2} - \frac{\lambda^3}{2^2 4^2 6^2} + \cdots \tag{26}$$

The zeros of the cosine, although you couldn't tell it from the series, have constant spacing π. The zeros of $J_0(\sqrt{\lambda})$ occur at $\sqrt{\lambda} \approx 2.4, 5.5, 8.65, 11.8, \ldots$ and their spacing converges rapidly to π (fortunately for our ears). The function $J_0(r)$ approaches a damped cosine $\sqrt{2/\pi r}\cos(r - \pi/4)$, with its amplitude slowly decreasing.

You must see the analogy between the Bessel function and the cosine. $B(r)$ comes from the oscillations of a circular drum; for $C(x) = \cos(k - \frac{1}{2})\pi x$ the drum is square. The circular drum is oscillating radially, as in Figure 4.8. The center of the circle and the left side of the square have zero slope (a free edge). $B(r)$ and $C(x)$ are eigenfunctions of Laplace's equation (with θ and y separated away):

$$-\frac{d}{dr}\left(r\frac{dB}{dr}\right) = \lambda r B \quad \text{and} \quad -\frac{d^2 C}{dx^2} = \lambda C. \tag{27}$$

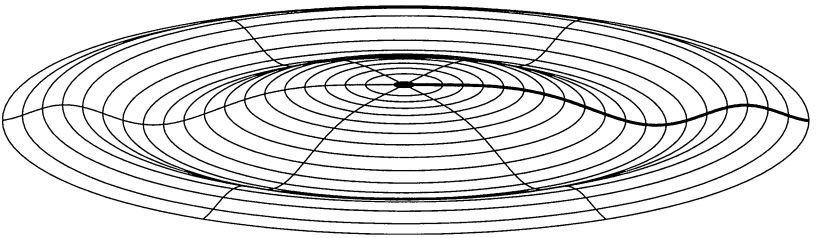


Figure 4.8: Bessel's $J_0(\sqrt{\lambda_3}\, r)$ shows the 3rd radial eigenfunction ($n = 0$) of a drum.

The first eigenfunction B drops from 1 to 0, like the cosine. $B(\sqrt{\lambda_2}\, r)$ crosses zero and comes up again. The kth eigenfunction, like the kth cosine, has k arches (Figure 4.8 shows $k = 3$). Each pure oscillation has its own frequency.

These are eigenfunctions of a symmetric problem. *Orthogonality must hold.* The Bessel functions $B_k(r) = J_0(\sqrt{\lambda_k} r)$ are orthogonal over a unit circle:

Orthogonality with $w = r$
$$\int_0^{2\pi} \int_0^1 B_k(r) B_l(r)\, r\, dr\, d\theta = 0 \quad \text{if} \quad k \neq l\,. \tag{28}$$

The cosines are orthogonal (with $w = 1$) over a unit square:

$$\int_0^1 \int_0^1 \cos(k - \tfrac{1}{2})\pi x \, \cos(l - \tfrac{1}{2})\pi x \, dx\, dy = 0 \quad \text{if} \quad k \neq l\,. \tag{29}$$

The θ integral and the y integral make no difference and can be ignored. The boundary conditions are identical, zero slope at the left endpoint 0 and zero value at the right endpoint 1. The difference for Bessel is the weighting factor $w = r$ in (28).

In closing we describe the other oscillations of a circular drum. If $A(\theta) = \cos n\theta$ then new Bessel functions will appear. Equation (22) has a solution which is finite at $r = 0$. That is the *Bessel function of order n*. (All other solutions blow up at $r = 0$; they involve Bessel functions of the second kind.) For every positive λ the solution is rescaled to $J_n(\sqrt{\lambda} r)$. The boundary condition $J_n(\sqrt{\lambda}) = 0$ at $r = 1$ picks out the eigenvalues. The products $A(\theta)B(r) = \cos n\theta J_n(\sqrt{\lambda_k} r)$ and $\sin n\theta J_n(\sqrt{\lambda_k} r)$ are the eigenfunctions of the drum in its pure oscillations.

The question "***Can you hear the shape of a drum?***" was unsolved for a long time. Do the Laplace eigenvalues determine the shape of a region? The answer turned out to be ***no***. Different shapes have the same λ's, and sound the same.

Along the "nodal lines" the drum does not move. Those are like the zeros of the sine function, where a violin string is still. For $A(\theta)B(r)$, there is a nodal line from the center whenever $A = 0$ and a nodal circle whenever $B = 0$. Figure 4.9 shows where the drumhead is still. The oscillations $A(\theta)B(r)e^{i\sqrt{\lambda} t}$ solve the wave equation.

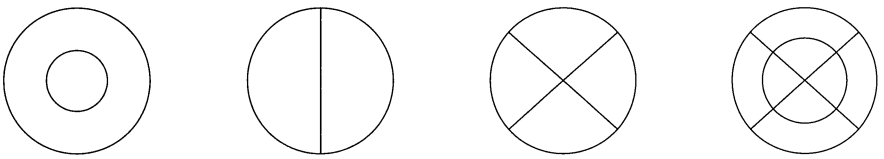


Figure 4.9: Nodal lines of a circular drum = zero lines of $A(\theta)B(r)$.

Problem Set 4.2

1 Find the double Fourier coefficients c_{mn} of these periodic functions $F(x, y)$:

(a) F = quarter square $= \begin{cases} 1 \text{ for } 0 \le x \le \pi, \ 0 \le y \le \pi \\ 0 \text{ if } -\pi < x < 0 \text{ or } -\pi < y < 0 \end{cases}$

(b) F = checkerboard $= \begin{cases} 1 \text{ if } xy \ge 0 \quad -\pi < x \le \pi \\ 0 \text{ if } xy < 0 \quad -\pi < y \le \pi \end{cases}$

2 Which functions $S(x, y)$ will have double sine series $\sum\sum b_{mn} \sin mx \sin ny$? State the orthogonality of those basis functions $\sin mx \sin ny$ (on what square?).

3 Find a formula for the coefficients b_{mn} in the double sine series.

4 Which functions $C(x, y)$ will have double cosine series $\sum\sum a_{mn} \cos mx \cos ny$? Show that this basis $\cos mx \cos ny$ is orthogonal on $[0, \pi]^2$.

5 Find a formula for the coefficients a_{mn} in the double cosine series. What is the coefficient a_{00} in the constant term?

6 Every $F(x) = C(x) + S(x) = \frac{1}{2}(F(x) + F(-x)) + \frac{1}{2}(F(x) - F(-x))$ is even plus odd. Split the function $F(x, y)$ similarly into $C(x, y) + S(x, y) +$ (two odd-even pieces). Those pieces use $\sin mx \cos ny$ and $\cos mx \sin ny$.

7 What is the double Fourier series of $\delta(x + y)$, the diagonal line of spikes?

8 Expand $F(x) = x^4$ as a combination of Chebyshev polynomials.

9 Estimate the distance from $x = 1$ to the next Chebyshev point $x = \cos(2\pi/k)$.

10 Chebyshev's $T_n(x)$ is a determinant from our second-difference matrix T:

$$T_n(x) = \det \begin{bmatrix} x & -1 & & \\ -1 & 2x & -1 & \\ & -1 & 2x & \cdot \\ & & \cdot & \cdot \end{bmatrix} \qquad \begin{aligned} T_1(x) &= x \\ T_2(x) &= 2x^2 - 1 \\ T_3(x) &= 4x^3 - 4x \end{aligned}$$

Including the next $-1, 2x, -1$ will give $T_4 = 2xT_3 - T_2$ from the rules for determinants. Explain why the determinant always has the same recursion $T_{n+1} = 2xT_n - T_{n-1}$ as the Chebyshev polynomial, so they are equal.

11 Chebyshev's $U_{n-1}(x)$ is a determinant from our second-difference matrix K:

$$U_{n-1}(x) = \frac{\sin n\theta}{\sin \theta} = \det \begin{bmatrix} 2x & -1 & \\ -1 & 2x & \cdot \\ & \cdot & \cdot \end{bmatrix} \qquad \begin{aligned} U_1(x) &= 2x \\ U_2(x) &= 4x^2 - 1 \end{aligned}$$

The recursion from the dots is still $U_{n+1} = 2xU_n - U_{n-1}$ but with a new start U_1, U_2. Show that $U_3(x) = 0$ at $x = \left\{\cos\frac{\pi}{4}, \cos\frac{2\pi}{4}, \cos\frac{3\pi}{4}\right\} = \left\{\frac{1}{\sqrt{2}}, 0, -\frac{1}{\sqrt{2}}\right\}$. How are the eigenvalues λ of K related to the zeros of $U_n(x)$?

12 Expand the determinant for T_n along row 1 and column 1 to get matrices (cofactors) of sizes $n-1$ and $n-2$, to show that $T_n(x) = xU_{n-1}(x) - U_{n-2}(x)$.

13 With $x = \cos\theta$ and $dx = -\sin\theta\, d\theta$, the derivative of $T_n(x) = \cos n\theta$ is $n\, U_{n-1}$:

$$T_n'(x) = -n \sin n\theta \frac{d\theta}{dx} = n \frac{\sin n\theta}{\sin \theta} = n\, U_{n-1}(x) = \text{second kind Chebyshev.}$$

Why are the extreme values of T_n at the zeros of U_{n-1}?

14 From its recursion, U_n starts with $2^n x^n$. Find the first term in T_n.

15 From the three-term recurrence (18), find the Legendre polynomial $P_4(x)$ when $P_2 = (3x^2-1)/2$ and $P_3 = (5x^3-3x)/2$. Which powers have $\int x^k P_4(x)\, dx = 0$?

16 Use *integration by parts* on the interval $-1 \le x \le 1$ to show that the third derivative of $(x^2-1)^3$ is orthogonal to the derivative of (x^2-1), coming from Rodrigues formula (17).

17 If $L_1 = x - a$ is orthogonal to $L_0 = 1$ with weight $w(x) = e^{-x}$ on $0 \le x < \infty$, what is a in that Laguerre polynomial $L_1(x)$?

18 If $H_2 = x^2 - b$ is orthogonal to 1 and x with weight e^{-x^2} on $-\infty < x < \infty$, what is b in that Hermite polynomial $H_2(x)$?

19 The polynomials $1, x, y, x^2 - y^2, 2xy, \ldots$ solve Laplace's equation in 2D. Find five combinations of $x^2, y^2, z^2, xy, xz, yz$ that satisfy $u_{xx} + u_{yy} + u_{zz} = 0$. With spherical polynomials of all degrees we can match $u = u_0$ on a sphere.

20 A Sturm-Liouville eigenvalue problem is $(pu')' + qu + \lambda wu = 0$. Multiply the equation for u_1 (with $\lambda = \lambda_1$) by u_2. Multiply the equation for u_2 by u_1 and subtract. With zero boundary conditions integrate $u_2(pu_1')'$ and $u_1(pu_2')'$ by parts to show *weighted orthogonality* $\int u_1 u_2 w\, dx = 0$ (if $\lambda_2 \neq \lambda_1$).

21 Fit the Bessel equation (22) into the framework of a Sturm-Liouville equation $(pu')' + qu + \lambda wu = 0$. What are p, q, and w? What are they for the Legendre equation $(1 - x^2)P'' - 2xP' + \lambda P = 0$?

22 The cosine series has $n!$ when the Bessel series has $2^2 4^2 \cdots n^2$. Write the latter as $2^n[(n/2)!]^2$ and use Stirling's formula $n! \approx \sqrt{2\pi n}\, n^n e^{-n}$ to show that the ratio of these coefficients approaches $\sqrt{\pi n/2}$. They have the same alternating signs.

23 Substitute $B = \sum c_m r^m$ into Bessel's equation and show from the analogue of (24) that λc_{m-2} must equal $(n^2 - m^2)c_m$. This recursion starts from $c_n = 1$ and successively finds $c_{n+2} = \lambda/(n^2 - (n+2)^2), c_{n+4}, \ldots$ as the coefficients in a *Bessel function of order* n:

$$B_n(r) = r^n \left[1 + \frac{\lambda r^2}{n^2 - (n+2)^2} + \frac{\lambda^2 r^2}{(n^2 - (n+2)^2)(n^2 - (n+4)^2)} + \cdots\right]$$

$$= \frac{n!}{2^n} \sum_{k=0}^{\infty} \frac{(-1)^k\, (\sqrt{\lambda}/2)^{2k+n}}{k!\,(k+n)!}.$$

24 Explain why the third Bessel function $J_0(\sqrt{\lambda_3}\, r)$ is zero at $r = \sqrt{\lambda_1/\lambda_3}, \sqrt{\lambda_2/\lambda_3}, 1$.

25 Show that the first Legendre polynomials $P_0 = 1, P_1 = \cos\varphi, P_2 = \cos^2\varphi - \frac{1}{3}$ are eigenfunctions of Laplace's equation $(wu_\varphi)_\varphi + w^{-1}u_{\theta\theta} = \lambda wu$ with $w = \sin\varphi$ on the surface of a sphere. Find the eigenvalues λ of these *spherical harmonics*. These $P_n(\cos\varphi)$ are the eigenfunctions that don't depend on longitude.

26 Where are the drum's nodal lines in Figure 4.9 if $n = 1, k = 2$ or $n = 2, k = 3$?

Table of Special Functions: Weighted Orthogonality, Recursion Formula, Differential Equation, and Series

Legendre Polynomial $P_n(x)$ with $w=1$ on $-1 \le x \le 1$ $\quad \int_{-1}^{1} P_m(x)P_n(x)\,dx = 0$

$$(n+1)P_{n+1} = (2n+1)xP_n - nP_{n-1} \text{ and } (1-x^2)P_n'' - 2xP_n' + n(n+1)P_n = 0$$

$$P_n(x) = \sum_{k=0}^{[n/2]} (-1)^k \binom{-\frac{1}{2}}{n-k} \binom{n-k}{k} (2x)^{n-k} = \frac{1}{2^n n!}\left(\frac{d}{dx}\right)^n (x^2-1)^n$$

Chebyshev Polynomial $T_n(x) = \cos n\theta$, with $x = \cos\theta$ and $w = 1/\sqrt{1-x^2}$

$$T_{n+1} = 2xT_n - T_{n-1} \text{ and } (1-x^2)T_n'' - xT_n' + n^2T_n = 0$$

$$\int_{-1}^{1} T_m(x)\,T_n(x)\,dx/\sqrt{1-x^2} = \int_{-\pi}^{\pi} \cos m\theta \cos n\theta\,d\theta = 0$$

$$T_n(x) = \frac{n}{2} \sum_{k=0}^{[n/2]} \frac{(-1)^k (n-k-1)!}{k!\,(n-2k)!} (2x)^{n-2k}$$

Bessel Function $J_p(x)$ with weight $w = x$ on $0 \le x \le 1$

$$xJ_{p+1} = 2pJ_p - xJ_{p-1} \text{ and } x^2J_p'' + xJ_p' + (x^2-p^2)J_p = 0$$

$$\int_0^1 x\,J_p(r_m x)\,J_p(r_n x)\,dx = 0 \quad \text{if} \quad J_p(r_m) = J_p(r_n) = 0$$

$$J_p(x) = \frac{\Gamma(p+1)}{2^p} \sum_{k=0}^{\infty} \frac{(-1)^k (x/2)^{2k+p}}{k!\,\Gamma(k+p+1)}$$

Laguerre Polynomial $L_n(x)$ with weight $w = e^{-x}$ on $0 \le x < \infty$

$$(n+1)L_{n+1} = (2n+1-x)L_n - nL_{n-1} \text{ and } xL_{n+1}'' + (1-x)L_n' + nL_n = 0$$

$$L_n(x) = \sum_{k=0}^{n} \frac{(-1)^k\,n!}{(k!)^2(n-k)!}\,x^k \qquad \int_0^\infty e^{-x} L_m(x) L_n(x)\,dx = 0$$

Hermite Polynomial $H_n(x)$ with weight $w = e^{-x^2}$ on $-\infty < x < \infty$

$$H_{n+1} = 2xH_n - 2nH_{n-1} \text{ and } H_n'' - 2xH_n' + 2nH_n = 0$$

$$H_n(x) = \sum_{k=1}^{[n/2]} \frac{(-1)^k\,n!}{k!\,(n-2k)!}\,(2x)^{n-2k} \qquad \int_{-\infty}^{\infty} e^{-x^2} H_m(x) H_n(x)\,dx = 0$$

Gamma Function $\Gamma(n+1) = n\Gamma(n)$ leading to $\Gamma(n+1) = n!$

$$\Gamma(n) = \int_0^\infty e^{-x} x^{n-1}\,dx \quad \text{gives} \quad \Gamma(1) = 0! = 1 \text{ and } \Gamma(\tfrac{1}{2}) = \sqrt{\pi}$$

$$\Gamma(n+1) = n! \approx \sqrt{2\pi n}\left(\frac{n}{e}\right)^n \qquad \text{(Stirling's factorial formula for large } n\text{)}$$

Binomial Numbers $\binom{n}{m} = \dfrac{n!}{m!\,(n-m)!} =$ "n choose m" $= \dfrac{\Gamma(n+1)}{\Gamma(m+1)\,\Gamma(n-m+1)}$

Binomial Theorem $(a+b)^n = \sum_{m=0} \binom{n}{m} a^{n-m} b^m$ (infinite series unless $n = 1, 2, \ldots$)

4.3 DISCRETE FOURIER TRANSFORM AND THE FFT

This section moves from functions $F(x)$ and infinite series to vectors $(f_0, \ldots, f_{N-1})$ and finite series. The vectors have N components and the series have N terms. The exponential e^{ikx} is still basic, but now x only takes N different values. Those values $x = 0, 2\pi/N, 4\pi/N, \ldots$ have equal spacing $2\pi/N$. This means that the N numbers e^{ix} are the powers of a totally important complex number $\boldsymbol{w = \exp(i2\pi/N)}$:

Powers of w $\quad e^{i0} = 1 = w^0 \quad e^{i2\pi/N} = w \quad e^{i4\pi/N} = w^2 \quad \ldots \quad w^{N-1}$

The **Discrete Fourier Transform (DFT)** deals entirely with those powers of w. Notice that the Nth power w^N cycles back to $e^{2\pi i N/N} = 1$.

The DFT and the inverse DFT are multiplications by the Fourier matrix F_N and its inverse matrix F_N^{-1}. The **Fast Fourier Transform (FFT)** is a brilliant way to multiply quickly. When a matrix has N^2 entries, an ordinary matrix-vector product uses N^2 multiplications. The Fast Fourier Transform uses only N times $\frac{1}{2}\log_2 N$ multiplications. It is the most valuable numerical algorithm in my lifetime, changing (1024)(1024) into (1024)(5). Whole industries have been speeded up by this one idea.

Roots of Unity and the Fourier Matrix

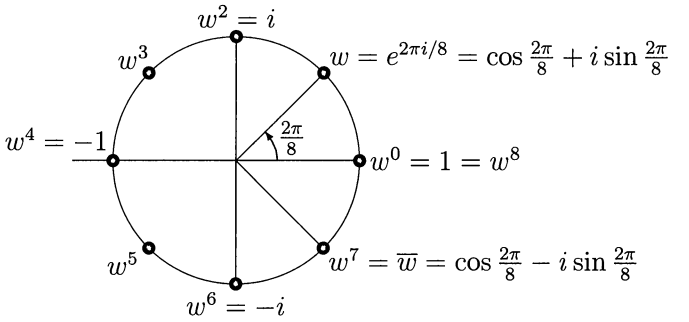
Quadratic equations have two roots (or a double root). Equations of degree n have n roots (counting repetitions). This is the Fundamental Theorem of Algebra, and to make it true we must allow complex roots. This section is about the very special equation $z^N = 1$. ***The solutions $z = 1, w, .., w^{N-1}$ are the "Nth roots of unity".*** They are N evenly spaced points around the unit circle in the complex plane.

Figure 4.10 shows the eight solutions to $z^8 = 1$. Their spacing is $\frac{1}{8}(360°) = 45°$. The first root is at 45° or $\theta = 2\pi/8$ radians. ***It is the complex number $w = e^{i2\pi/8}$.*** We call this number w_8 to emphasize that it is an 8th root. You could write it as $\cos\frac{2\pi}{8} + i\sin\frac{2\pi}{8}$, but don't do it. Powers of w are best in the form $e^{i\theta}$, because we work only with the angle.

The seven other 8th roots around the circle are $w^2, w^3, \ldots, w^8$, and that last one is $w^8 = 1$. The next to last root w^7 is the same as the complex conjugate $\overline{w} = e^{-i2\pi/8}$. (Multiply $\overline{w}$ by w to get $e^0 = 1$, so $\overline{w}$ is also w^{-1}.) The powers of $\overline{w}$ just go backward around the circle (clockwise). You will see them in the inverse Fourier matrix.

For the fourth roots of 1, the separation angle is $2\pi/4$ or 90°. The number $e^{2\pi i/4} = \cos\frac{\pi}{2} + i\sin\frac{\pi}{2}$ is nothing but i. The four roots are i, $i^2 = -1$, $i^3 = -i$, and $i^4 = 1$.

The idea behind the FFT is to go from an 8 by 8 Fourier matrix (powers of w_8) to a 4 by 4 matrix (powers of $w_4 = i$). By exploiting the connections of F_4 and F_8 and F_{16} and beyond, multiplication by F_{1024} is very quick. ***The key connection for the Fast Fourier Transform is the simple fact that $(w_8)^2 = w_4$.***



8 powers of w_8
4 powers of w_4
$w_4 = i = (w_8)^2$

Figure 4.10: The eight solutions to $z^8 = 1$ are $1, w, w^2, \ldots, w^7$ with $w = (1+i)/\sqrt{2}$.

Here is the **Fourier matrix** F_N for $N = 4$. Each entry is a power of $w_4 = i$:

Fourier matrix

$$F_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & w & w^2 & w^3 \\ 1 & w^2 & w^4 & w^6 \\ 1 & w^3 & w^6 & w^9 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & i^2 & i^3 \\ 1 & i^2 & i^4 & i^6 \\ 1 & i^3 & i^6 & i^9 \end{bmatrix}.$$

Those four columns are orthogonal! Columns 0 and 1 have zero inner product:

$$(\textbf{column 0})^{\mathrm{T}}(\textbf{column 1}) = w^0 + w^1 + w^2 + w^3 = \mathbf{0}. \text{ This is } 1 + i + i^2 + i^3 = 0. \quad (1)$$

We are adding four equally spaced points in Figure 4.10, and *each pair of opposite points cancels* (i^2 cancels 1 and i^3 cancels i). For the 8 by 8 Fourier matrix, we would be adding all eight points in the figure. The sum $1 + w + \cdots + w^7$ is again zero.

Now look at columns 1 and 3 (remember that the numbering starts at zero). Their ordinary inner product looks like 4, which we don't want:

$$1 \cdot 1 + i \cdot i^3 + i^2 \cdot i^6 + i^3 \cdot i^9 = \mathbf{1 + 1 + 1 + 1} \qquad \textbf{\textit{but this is wrong.}}$$

These are complex vectors, not real vectors. The correct inner product must take the **complex conjugate** of one vector (change i to $-i$). Now we see orthogonality:

$$(\overline{\textbf{col 1}})^{\mathrm{T}}(\textbf{col 3}) = 1 \cdot 1 + (-i) \cdot i^3 + (-i)^2 \cdot i^6 + (-i)^3 \cdot i^9 = \mathbf{1 - 1 + 1 - 1 = 0}. \quad (2)$$

The correct inner product of every column with itself is $1 + 1 + 1 + 1 = 4$:

$$\|\text{Column 1}\|^2 = (\overline{\text{col 1}})^{\mathrm{T}}(\text{col 1}) = 1 \cdot 1 + (-i) \cdot i + (-i)^2 \cdot i^2 + (-i)^3 \cdot i^3 = 4. \quad (3)$$

The columns of F_4 are not unit vectors. They all have length $\sqrt{4} = 2$, so $\frac{1}{2}F_4$ has **orthonormal columns**. Multiplying $\frac{1}{2}\overline{F}_4$ times $\frac{1}{2}F_4$ (row times column) produces I.

The inverse of F_4 is the matrix with $\overline{F}_4$ on the left (including both factors $\frac{1}{2}$):

$$\frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & (-i) & (-i)^2 & (-i)^3 \\ 1 & (-i)^2 & (-i)^4 & (-i)^6 \\ 1 & (-i)^3 & (-i)^6 & (-i)^9 \end{bmatrix} \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & i^2 & i^3 \\ 1 & i^2 & i^4 & i^6 \\ 1 & i^3 & i^6 & i^9 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = I. \quad (4)$$

Thus F_4^{-1} is $\frac{1}{4}\overline{F}_4^{\mathrm{T}}$, also written $\frac{1}{4}F_4^*$. Here is the general rule for F_N:

The columns of $\frac{1}{\sqrt{N}}F_N$ ***are orthonormal. Their inner products produce*** I.

$\left(\frac{1}{\sqrt{N}}\overline{F}_N^{\mathrm{T}}\right)\left(\frac{1}{\sqrt{N}}F_N\right) = I$ ***means that the inverse is*** $F_N^{-1} = \frac{1}{N}\overline{F}_N^{\mathrm{T}} = \frac{1}{N}F_N^*$ (5)

The Fourier matrix is symmetric, so transposing has no effect. The inverse matrix just divides by N and replaces i by $-i$, which changes every $w = \exp(i2\pi/N)$ into $\omega = \overline{w} = \exp(-i2\pi/N)$. For every N, the Fourier matrix contains the powers $(w_N)^{jk}$:

Fourier Matrix

$$F_N = \begin{bmatrix} 1 & 1 & 1 & \cdot & 1 \\ 1 & w & w^2 & \cdot & w^{N-1} \\ 1 & w^2 & w^4 & \cdot & w^{2(N-1)} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 1 & w^{N-1} & w^{2(N-1)} & \cdot & w^{(N-1)^2} \end{bmatrix} \begin{array}{l} \longleftarrow \textbf{ row 0} \\ \\ \textbf{with } \mathbf{F_{jk} = w^{jk}} \\ \\ \longleftarrow \textbf{ row N} - \mathbf{1} \end{array} \quad (6)$$

This matrix is the key to the Discrete Fourier Transform. Its special patterns give the fast DFT, which is the FFT. The row and column numbers j and k go from 0 to $N-1$. ***The entry in row*** j***, column*** k ***is*** $w^{jk} = \exp(ijk\,2\pi/N)$.

Fourier matrix in MATLAB $j = 0 : N-1$; $k = j'$; $F = w.\hat{}(k * j)$;

Important note. Many authors prefer to work with $\omega = e^{-2\pi i/N}$, which is the *complex conjugate* of our w. (They often use the Greek omega, and I will do that to keep the two options separate.) With this choice, their DFT matrix contains powers of ω not w. It is conj (F) = complex conjugate of our F.

This is a completely reasonable choice! MATLAB uses $\omega = e^{-2\pi i/N}$. The DFT matrix fft(eye(N)) contains powers of this number $\omega = \overline{w}$. In our notation that matrix is $\overline{F}$. **The Fourier matrix with** w**'s reconstructs** f **from** c. **The matrix** $\overline{F}$ **with** ω**'s transforms** f **to** c**, so we call it the DFT matrix**:

DFT matrix $= \overline{F}$ $\qquad \overline{F}_{jk} = \omega^{jk} = e^{-2\pi ijk/N} = \mathrm{fft}(\mathrm{eye}(N))$

The factor $1/N$ in F^{-1} is like the earlier $1/2\pi$. We include it in $c = F_N^{-1}f = \frac{1}{N}\overline{F}f$.

The Discrete Fourier Transform

The Fourier matrices F_N and F_N^{-1} produce the Discrete Fourier Transform and its inverse. Functions with infinite series turn into vectors $f_0, \ldots, f_{N-1}$ with finite sums:

$$F(x) = \sum_{-\infty}^{\infty} c_k e^{ikx} \quad \textbf{becomes} \quad f_j = \sum_{k=0}^{N-1} c_k w^{jk} \quad \textbf{which is } f = F_N c. \tag{7}$$

In the other direction, an integral with e^{-ikx} becomes a sum with $\overline{w}^{jk} = e^{-ikj2\pi/N}$:

$$c_k = \frac{1}{2\pi}\int_0^{2\pi} F(x)e^{-ikx}\,dx \quad \textbf{becomes} \quad c_k = \frac{1}{N}\sum_{j=0}^{N-1} f_j \overline{w}^{jk} \quad \textbf{which is } c = F_N^{-1} f \tag{8}$$

The power w^{jk} is the same as e^{ikx} at the jth point $x_j = j2\pi/N$. At those N points, we reconstruct $f_0, \ldots, f_{N-1}$ by combining the N columns of the Fourier matrix. Previously we reconstructed functions $F(x)$ at all points by an infinite sum.

The zeroth coefficient c_0 is always the average of f. Here $c_0 = (f_0 + \cdots + f_{N-1})/N$.

Example 1 The **Discrete Delta Function** is $\delta = (1, 0, 0, 0)$. For functions, the Fourier coefficients of the spike $\delta(x)$ are all equal. Here the coefficients $c = F_4^{-1}\delta$ are all $\frac{1}{N} = \frac{1}{4}$:

$$\begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} = \frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & (-i) & (-i)^2 & (-i)^3 \\ 1 & (-i)^2 & (-i)^4 & (-i)^6 \\ 1 & (-i)^3 & (-i)^6 & (-i)^9 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \frac{1}{4}\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}. \tag{9}$$

To reconstruct $f = (1, 0, 0, 0)$ from its transform $c = (\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$, multiply F times c. Notice again how all rows except the zeroth row of F add to zero:

$$Fc = \begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & i^2 & i^3 \\ 1 & i^2 & i^4 & i^6 \\ 1 & i^3 & i^6 & i^9 \end{bmatrix}\frac{1}{4}\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{10}$$

Example 2 The **constant vector** $f = (1, 1, 1, 1)$ produces $c =$ delta vector $= (1, 0, 0, 0)$ This reverses the previous example without $1/N$. We mention that f and c are both "even". What does it mean for vectors to be even or odd? **Think cyclically**.

Symmetry or antisymmetry is across the zero position. **The entry in the -1 position is by definition the entry in the $N-1$ position.** We are working "mod N" or "mod 4" so that $-1 \equiv 3$ and $-2 \equiv 2$. In this mod 4 arithmetic $2+2 \equiv 0$, because w^2 times w^2 is w^0. In the mod N world, f_{-k} is f_{N-k}:

Even vector f: $f_k = f_{N-k}$ as in (f_0, f_1, f_2, f_1) (11)

Odd vector f: $f_k = -f_{N-k}$ as in $(0, f_1, 0, -f_1)$ (12)

Example 3 The **discrete sine** $f = (0, 1, 0, -1)$ is an odd vector. Its Fourier coefficients c_k are pure imaginary, as in $\sin x = \frac{1}{2i}e^{ix} - \frac{1}{2i}e^{-ix}$. In fact we still get $\frac{1}{2i}$ and $-\frac{1}{2i}$:

$$c = F_4^{-1}f = \frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & (-i) & (-i)^2 & (-i)^3 \\ 1 & (-i)^2 & (-i)^4 & (-i)^6 \\ 1 & (-i)^3 & (-i)^6 & (-i)^9 \end{bmatrix}\begin{bmatrix} 0 \\ 1 \\ 0 \\ -1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1/2i \\ 0 \\ -1/2i \end{bmatrix}. \tag{13}$$

Odd inputs f produce pure imaginary Fourier coefficients $c_k = a_k + ib_k = ib_k$.

The Discrete Cosine Transform (**DCT**) is the key to **JPEG compression**. All .jpeg files were created from 8×8 DCT's, until wavelets arrived in the JPEG2000 standard. Cosines give a symmetric extension at the ends of vectors, like starting with a function from 0 to π and reflecting it to be even: $C(-x) = C(x)$. *No jump is created at the reflection points.* Compression of data and images and video is so important that we come back to it in Section 4.7.

One Step of the Fast Fourier Transform

To reconstruct f we want to multiply F_N times $\mathbf{c}$ as quickly as possible. The matrix has N^2 entries, so normally we would need N^2 separate multiplications. You might think it is impossible to do better. (Since F_N has no zero entries, no multiplications can be skipped.) By using the special pattern w^{jk} for its entries, F_N can be factored in a way that produces many zeros. This is the **FFT**.

The key idea is to connect F_N with the half-size Fourier matrix $F_{N/2}$. Assume that N is a power of 2 (say $N = 2^{10} = 1024$). We will connect F_{1024} to F_{512}—or rather to ***two copies*** of F_{512}. When $N = 4$, we connect F_4 to $[\,F_2\ 0\ ;\ 0\ F_2\,]$:

$$F_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & i^2 & i^3 \\ 1 & i^2 & i^4 & i^6 \\ 1 & i^3 & i^6 & i^9 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} F_2 & 0 \\ 0 & F_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & & \\ 1 & i^2 & & \\ & & 1 & 1 \\ & & 1 & i^2 \end{bmatrix}.$$

On the left is F_4, with no zeros. On the right is a matrix that is half zero. The work is cut in half. But wait, those matrices are not the same. The block matrix with F_2's is only one piece of the factorization of F_4. The other pieces have many zeros:

Key idea
$$F_4 = \begin{bmatrix} 1 & & 1 & \\ & 1 & & i \\ 1 & & -1 & \\ & 1 & & -i \end{bmatrix}\begin{bmatrix} 1 & 1 & & \\ 1 & i^2 & & \\ & & 1 & 1 \\ & & 1 & i^2 \end{bmatrix}\begin{bmatrix} 1 & & & \\ & & 1 & \\ & 1 & & \\ & & & 1 \end{bmatrix}. \tag{14}$$

The permutation matrix on the right puts c_0 and c_2 (evens) ahead of c_1 and c_3 (odds). The middle matrix performs separate half-size transforms on those evens and odds. The matrix at the left combines the two half-size outputs, in a way that produces the correct full-size output $\mathbf{f} = F_4\mathbf{c}$. You could multiply those three matrices to see F_4.

The same idea applies when $N = 1024$ and $M = \frac{1}{2}N = 512$. The number w is $e^{2\pi i/1024}$. It is at the angle $\theta = 2\pi/1024$ on the unit circle. The Fourier matrix F_{1024} is full of powers of w. The first stage of the FFT is the great factorization discovered by Cooley and Tukey (and foreshadowed in 1805 by Gauss):

$$\textbf{FFT (Step 1)} \quad F_{1024} = \begin{bmatrix} I_{512} & D_{512} \\ I_{512} & -D_{512} \end{bmatrix} \begin{bmatrix} F_{512} & \\ & F_{512} \end{bmatrix} \begin{bmatrix} \text{even-odd} \\ \text{permutation} \end{bmatrix} \tag{15}$$

I_{512} is the identity matrix. D_{512} is the diagonal matrix with entries $(1, w, \ldots, w^{511})$ using w_{1024}. The two copies of F_{512} are what we expected. Don't forget that they use the 512th root of unity, which is nothing but $(w_{1024})^2$. The even-odd permutation matrix separates the incoming vector $\mathbf{c}$ into $\mathbf{c}' = (c_0, c_2, \ldots, c_{1022})$ and $\mathbf{c}'' = (c_1, c_3, \ldots, c_{1023})$.

Here are the algebra formulas which express this neat FFT factorization of F_N:

(**FFT**) Set $M = \frac{1}{2}N$. The components of $\mathbf{f} = F_N\mathbf{c}$ are combinations of the half-size transforms $\mathbf{f}' = F_M\mathbf{c}'$ and $\mathbf{f}'' = F_M\mathbf{c}''$. Equation (15) shows $I\mathbf{f}' + D\mathbf{f}''$ and $I\mathbf{f}' - D\mathbf{f}''$:

$$\begin{array}{lrcl} \textbf{First half} & \mathbf{f}_j & = & \mathbf{f}'_j + (w_N)^j\mathbf{f}''_j, \quad j = 0, \ldots, M-1 \\ \textbf{Second half} & \mathbf{f}_{j+M} & = & \mathbf{f}'_j - (w_N)^j\mathbf{f}''_j, \quad j = 0, \ldots, M-1 \end{array} \tag{16}$$

Thus each FFT step has three parts: split $\mathbf{c}$ into $\mathbf{c}'$ and $\mathbf{c}''$, transform them separately by F_M into $\mathbf{f}'$ and $\mathbf{f}''$, and reconstruct $\mathbf{f}$ from equation (16). N must be even!

The algebra of (16) is a splitting into even numbers $2k$ and odd $2k+1$, with $w = w_N$:

$$\textbf{Even/Odd} \quad f_j = \sum_0^{N-1} w^{jk}c_k = \sum_0^{M-1} w^{2jk}c_{2k} + \sum_0^{M-1} w^{j(2k+1)}c_{2k+1} \text{ with } M = \tfrac{1}{2}N. \tag{17}$$

The even c's go into $c' = (c_0, c_2, \ldots)$ and the odd c's go into $c'' = (c_1, c_3, \ldots)$. Then come the transforms F_Mc' and F_Mc''. The key is $\boldsymbol{w_N^2 = w_M}$. This gives $w_N^{2jk} = w_M^{jk}$.

$$\textbf{Rewrite} \qquad f_j = \sum w_M^{jk}c'_k \;+\; (w_N)^j \sum w_M^{jk}c''_k = f'_j \;+\; (w_N)^j f''_j\,. \tag{18}$$

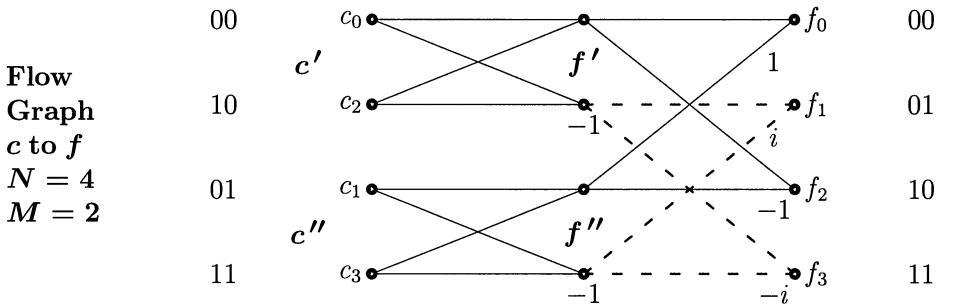
For $j \geq M$, the minus sign in (16) comes from factoring out $(w_N)^M = -1$.

MATLAB easily separates even c's from odd c's and multiplies by w_N^j. We use conj(F) or equivalently MATLAB's inverse transform ifft, because fft is based on $\omega = \overline{w} = e^{-2\pi i/N}$. Problem 2 shows that F and conj(F) are linked by permuting rows.

FFT Step from N to $N/2$ in MATLAB

```
f' = ifft(c(0:2:N-2)) * N/2;
f'' = ifft(c(1:2:N-1)) * N/2;
d = w.^(0:N/2-1)';
f = [f' + d.*f''; f' - d.*f''];
```

The flow graph shows c' and c'' going through the half-size F_2. Those steps are called "*butterflies*," from their shape. Then the outputs f' and f'' are combined (multiplying f'' by $1, i$ and also by $-1, -i$) to produce $f = F_4c$.



This reduction from F_N to two F_M's almost cuts the work in half—you see the zeros in the matrix factorization (15). That reduction is good but not great. The full idea of the FFT is much more powerful. It saves much more time than 50%.

The Full FFT by Recursion

If you have read this far, you have probably guessed what comes next. We reduced F_N to $F_{N/2}$. ***Keep going to*** $F_{N/4}$. The two copies of F_{512} lead to four copies of F_{256}. Then 256 leads to 128. *That is recursion.* It is a basic principle of many fast algorithms. Here is the second stage with $F = F_{256}$ and $D = \text{diag}\,(1, w_{512}, \ldots, (w_{512})^{255})$:

$$\begin{bmatrix} F_{512} & 0 \\ 0 & F_{512} \end{bmatrix} = \begin{bmatrix} I & D & & \\ I & -D & & \\ & & I & D \\ & & I & -D \end{bmatrix} \begin{bmatrix} F & & & \\ & F & & \\ & & F & \\ & & & F \end{bmatrix} \begin{bmatrix} \text{pick} & 0,4,8,\cdots \\ \text{pick} & 2,6,10,\cdots \\ \text{pick} & 1,5,9,\cdots \\ \text{pick} & 3,7,11,\cdots \end{bmatrix}.$$

We can count the individual multiplications, to see how much is saved. Before the FFT was invented, the count was $N^2 = (1024)^2$. This is about a million multiplications. I am not saying that they take a long time. The cost becomes large when we have many, many transforms to do—which is typical. Then the saving is also large:

The final count for size $N = 2^L$ ***is reduced from*** N^2 ***to*** $\frac{1}{2}NL$.

The number $N = 1024$ is 2^{10}, so $L = 10$. The original count of $(1024)^2$ is reduced to $(5)(1024)$. The saving is a factor of 200, because a million is reduced to five thousand. That is why the FFT has revolutionized signal processing.

Here is the reasoning behind $\frac{1}{2}NL$. There are L levels, going from $N = 2^L$ down to $N = 1$. Each level has $\frac{1}{2}N$ multiplications from diagonal D, to reassemble the half-size outputs. This yields the final count $\frac{1}{2}NL$, which is $\frac{1}{2}N \log_2 N$.

Exactly the same idea gives a fast inverse transform. The matrix F_N^{-1} contains powers of the conjugate $\overline{w}$. We just replace w by $\overline{w}$ in the diagonal matrix D, and in formula (16). At the end, divide by N.

One last note about this remarkable algorithm. There is an amazing rule for the order that the c's enter the butterflies, after all L of the odd-even permutations. Write the numbers 0 to $N-1$ in base 2. *Reverse the order of their bits (binary digits).* The complete flow graph shows the bit-reversed order at the start, then $L = \log_2 N$ recursion steps. The final output is F_N times $\mathbf{c}$.

The fastest FFT will be adapted to the processor and cache capacities of each specific computer. There will naturally be differences from a textbook description, but the idea of recursion is still crucial. For free software that automatically adjusts, we highly recommend the website fftw.org.

Problem Set 4.3

1 Multiply the three matrices in equation (14) and compare with F. In which six entries do you need to know that $i^2 = -1$? This is $(w_4)^2 = w_2$. If $M = N/2$, why is $(w_N)^M = -1$?

2 *Why is row i of $\overline{F}$ the same as row $N-i$ of F* (numbered from 0 to $N-1$)?

3 From Problem 2, find the 4 by 4 permutation matrix P so that $F = P\overline{F}$. Check that $P^2 = I$ so that $P = P^{-1}$. Then from $\overline{F}F = 4I$ show that $P = F^2/4$. It is amazing that $P^2 = F^4/16 = I$! Four transforms of c bring back $16\,c$.

Note For all N, F^2/N is a symmetric permutation matrix P. It has the rows of I in the order $1, N, N-1, \ldots, 2$. Then $P = \left[\begin{smallmatrix} 1 & 0 \\ 0 & J \end{smallmatrix}\right] = I([1, N\!:\!-1\!:\!2], :)$ for the *reverse identity* J. From $P^2 = I$ we find (surprisingly!) that $F^4 = N^2 I$. The key facts about P and F and their eigenvalues are on the cse website.

4 Invert the three factors in equation (14) to find a fast factorization of F^{-1}.

5 F is symmetric. Transpose equation (14) to find a new Fast Fourier Transform!

6 All entries in the factorization of F_6 involve powers of $w =$ sixth root of 1:

$$F_6 = \begin{bmatrix} I & D \\ I & -D \end{bmatrix} \begin{bmatrix} F_3 & \\ & F_3 \end{bmatrix} \begin{bmatrix} \text{even} \\ \text{odd} \end{bmatrix}.$$

Write down these factors with $1, w, w^2$ in D and $1, w^2 = \sqrt[3]{1}, w^4$ in F_3. Multiply!

7 By analogy with the discrete sine $(0, 1, 0, -1)$ what is the discrete cosine vector $(N = 4)$? What is its transform?

8 Put the vector $c = (1, 0, 1, 0)$ through the three steps of the FFT (those are the three multiplications in (14)) to find $y = Fc$. Do the same for $c = (0, 1, 0, 1)$.

9 Compute $y = F_8 c$ by the three FFT steps for $c = (1, 0, 1, 0, 1, 0, 1, 0)$. Repeat the computation for $c = (0, 1, 0, 1, 0, 1, 0, 1)$.

10 If $w = e^{2\pi i/64}$ then w^2 and $\sqrt{w}$ are among the ____ and ____ roots of 1.

11 (a) Draw all the sixth roots of 1 on the unit circle. Prove they add to zero.

(b) What are the three cube roots of 1? Do they also add to zero?

Problems 12–14 give an important speedup for the transform of real f's.

12 If the vector f is *real*, show that its transform c has the crucial property $\bar{c}_{N-k} = c_k$. This is the analog of $\bar{c}_{-k} = c_k$ for Fourier series $f(x) = \sum c_k e^{ikx}$. Start from

$$c_{N-k} = \frac{1}{N}\sum_{j=0}^{N-1} f_j w^{j(N-k)}. \quad \text{Use } w^N = 1, w^{-1} = \overline{w}, \overline{f}_j = f_j, \text{ to find } \bar{c}_{N-k}.$$

13 The DFT of *two real vectors* f and g comes from one complex DFT of $h = f + ig$. From the transform b of h, show that the transforms c and d of f and g are

$$c_k = \frac{1}{2}(b_k + \bar{b}_{N-k}) \quad \text{and} \quad d_k = \frac{i}{2}(\bar{b}_{N-k} - b_k).$$

14 To speed up the DFT of *one real vector* f, separate it into half-length vectors f_{even} and f_{odd}. From $h = f_{\text{even}} + if_{\text{odd}}$ find its M-point transform b (for $M = \frac{1}{2}N$). Then form the transforms c and d of f_{even} and f_{odd} as in Problem 13. Use equation (17) to construct the transform $\widehat{f}$ from c and d.

Note For real f, this reduces the number $\frac{1}{2}N\log_2 N$ of *complex* multiplications by $\frac{1}{2}$. Each complex $(a + ib)(c + id)$ requires only three real multiplications (not 4) with an extra addition.

15 The columns of the Fourier matrix F are the *eigenvectors* of the *cyclic* (not odd-even) permutation P. Multiply PF to find the eigenvalues λ_1 to λ_4 of P:

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & i^2 & i^3 \\ 1 & i^2 & i^4 & i^6 \\ 1 & i^3 & i^6 & i^9 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & i^2 & i^3 \\ 1 & i^2 & i^4 & i^6 \\ 1 & i^3 & i^6 & i^9 \end{bmatrix} \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \lambda_3 & \\ & & & \lambda_4 \end{bmatrix}.$$

This is $PF = F\Lambda$ or $P = F\Lambda F^{-1}$. The eigenvector matrix for P is F.

16 The equation $\det(P - \lambda I) = 0$ reduces to $\lambda^4 = 1$. Again the eigenvalues of P are ____. Which permutation matrix has eigenvalues = cube roots of 1?

17 Two eigenvectors of this "circulant matrix" C are $(1, 1, 1, 1)$ and $(1, i, i^2, i^3)$. Multiply these vectors by C to find the two eigenvalues λ_0 and λ_1:

Circulant matrix $\begin{bmatrix} c_0 & c_1 & c_2 & c_3 \\ c_3 & c_0 & c_1 & c_2 \\ c_2 & c_3 & c_0 & c_1 \\ c_1 & c_2 & c_3 & c_0 \end{bmatrix}$ has $C\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \lambda_0 \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$.

Notice that $C = c_0 I + c_1 P + c_2 P^2 + c_3 P^3$ with the cyclic permutation P in Problem 15. Therefore $C = F(c_0 I + c_1\Lambda + c_2\Lambda^2 + c_3\Lambda^3)F^{-1}$. That matrix in parentheses is diagonal. It contains the ____ of C.

18 Find the eigenvalues of the cyclic $-1, 2, -1$ matrix from $2I - \Lambda - \Lambda^3$ in Problem 17. The -1's in the corners make the second difference matrix periodic:

$$C = \begin{bmatrix} 2 & -1 & 0 & -1 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ -1 & 0 & -1 & 2 \end{bmatrix} \quad \text{has } (c_0, c_1, c_2, c_3) = (2, -1, 0, -1)\,.$$

19 An even vector $c = (a, b, d, b)$ produces a ____ circulant matrix. Its eigenvalues are real. An odd vector $c = (0, e, 0, -e)$ produces a ____ circulant matrix with imaginary eigenvalues.

20 To multiply $C = FEF^{-1}$ times a vector x, we can multiply $F(E(F^{-1}x))$. The direct Cx uses n^2 separate multiplications. The Fourier matrices F and F^{-1} and the eigenvalue matrix E use only $n \log_2 n + n$ multiplications. How many of those come from E, how many from F, and how many from F^{-1}?

21 How can you quickly compute these four components of Fc if you know $c_0 + c_2$, $c_0 - c_2$, $c_1 + c_3$, $c_1 - c_3$? You are finding the Fast Fourier Transform!

$$Fc = \begin{bmatrix} c_0 + c_1 + c_2 + c_3 \\ c_0 + ic_1 + i^2c_2 + i^3c_3 \\ c_0 + i^2c_1 + i^4c_2 + i^6c_3 \\ c_0 + i^3c_1 + i^6c_2 + i^9c_3 \end{bmatrix}.$$

22 The Fourier matrix $F2D$ in two dimensions (on a grid of N^2 points) is $\boldsymbol{F2D = \mathsf{kron}(F, F)}$ (1D transform of each row, then each column).

(a) Write out $F2D$ for $N = 2$ (size $N^2 = 4$, a "Hadamard matrix").

(b) The two-dimensional DFT matrix $\mathsf{conj}(F2D)$ comes from $\omega = e^{-2\pi i/N}$. Explain why $F2D$ times $\mathsf{conj}(F2D)$ equals $N^2(I2D)$.

(c) What is the two-dimensional δ vector for $N = 3$ and what are its 2D Fourier coefficients c_{jk}?

(d) Why does a multiplication $(F2D)\,u$ need only $O(N^2 \log N)$ operations?

4.4 CONVOLUTION AND SIGNAL PROCESSING

Convolution answers a question that we unavoidably ask. When $\sum c_k e^{ikx}$ multiplies $\sum d_k e^{ikx}$ (call those functions $f(x)$ and $g(x)$), **what are the Fourier coefficients of** $f(x)g(x)$? The answer is not $c_k d_k$. Those are *not* the coefficients of $h(x) = f(x)g(x)$. The correct coefficients of $(\sum c_k e^{ikx})(\sum d_k e^{ikx})$ come from *"convolving"* the vector of c's with the vector of d's. That convolution is written $\boldsymbol{c * d}$.

Before I explain convolution, let me ask a second question. **What function does have the Fourier coefficients $c_k d_k$?** This time we are multiplying in "transform space." In that case we should convolve $f(x)$ with $g(x)$ in "x-space." **Multiplication in one domain is convolution in the other domain**:

Convolution Rules	The multiplication $\boldsymbol{f(x)g(x)}$ has Fourier coefficients $\boldsymbol{c * d}$ Multiplying $2\pi \boldsymbol{c_k d_k}$ gives the coefficients of $\boldsymbol{f(x) * g(x)}$

Our job is to define those convolutions $c * d$ and $f * g$, and to see how useful they are.

The same convolution rules hold for the discrete N-point transforms. Those are easy to show by examples, because the sums are finite. The new twist is that the discrete convolutions have to be "cyclic", so we write $c \circledast d$. All the sums have N terms, because higher powers of w fold back into lower powers. They circle around. *N is the same as 0 because $w^N = w^0 = 1$.*

The vectors c and d and $c \circledast d$ all have N components. I will start with N-point examples because they are easier to see. We just multiply polynomials. *Cyclically.*

Example 1 What are the coefficients of $f(w) = 1 + 2w + 4w^2$ times $g(w) = 3 + 5w^2$ if $w^3 = 1$? This is **cyclic convolution** with $N = 3$.

$$\begin{aligned} &\textbf{Non-cyclic} \quad (1 + 2w + 4w^2)(3 + 0w + 5w^2) = 3 + 6w + 17w^2 + 10w^3 + 20w^4 \\ &\textbf{Cyclic} \quad\quad (1 + 2w + 4w^2)(3 + 0w + 5w^2) = \_\_\_\_ + \_\_\_\_ w + \_\_\_\_ w^2 . \end{aligned} \tag{1}$$

The w^2 term comes from 1 times $5w^2$, and $2w$ times $0w$, and $4w^2$ times 3: Total $\mathbf{17w^2}$.

The w term comes from $2w$ times 3, and $4w^2$ times $5w^2$ (because $w^4 = w$): Total $\mathbf{26w}$.

The constant term is $(1)(3) + (2)(5) + (4)(0)$ (because $(w)(w^2) = 1$): Total **13**.

$$\textbf{Cyclic Convolution} \qquad c \circledast d = (1, 2, 4) \circledast (3, 0, 5) = (13, 26, 17) . \tag{2}$$

Underneath this example is third-grade multiplication! For $w = 10$ the multiplication is $f = 421$ times $g = 503$. But teachers could get disturbed when 17 appears and you don't carry 1. That is just a difference of opinion. The serious reaction is when you say that w^3 counts the same as 1. The cyclic answer is 13 plus $26w$ plus $17w^2$.

Compare cyclic $c \circledast d$ with non-cyclic $c * d$. Read from right to left because $w = 10$:

		4	2	1			4	2	1	
		5	0	3			5	0	3	
		12	6	3			12	6	3	
	0	0	0				0	0	**0**	
20	**10**	5			**non-cyclic**		5	**20**	**10**	**cyclic**
					right to left					right to left
20	10	17	6	3	$\boldsymbol{c * d}$		17	26	13	$\boldsymbol{c \circledast d}$

That product $3+60+1700+10000+200000$ is the correct answer for 421 times 503. If it gets marked wrong for not carrying, don't forget that the life of teachers (and professors) is not easy. Just when we thought we understood multiplication...

Example 2 Multiply $f(x) = 1 + 2e^{ix} + 4e^{2ix}$ times $g(x) = 3 + 5e^{2ix}$. The answer is $3 + 6e^{ix} + 17e^{2ix} + 10e^{3ix} + 20e^{4ix}$. This shows **non-cyclic convolution**.

The e^{2ix} coefficient in $f(x)g(x)$ is the same $(4)(3)+(2)(0)+(1)(5) = 17$ as before. Now there is also $10e^{3ix} + 20e^{4ix}$ and the cyclic property $w^3 = 1$ is gone.

Non-cyclic Convolution $c * d = (1, 2, 4) * (3, 0, 5) = (3, 6, 17, 10, 20)$. (3)

This non-cyclic convolution is produced by MATLAB's **conv** command:

$$c = [\ 1\ \ 2\ \ 4\ ];\ \ d = [\ 3\ \ 0\ \ 5\ ];\ \ \mathbf{conv}(c, d) \ \text{ produces } \ c * d.$$

If c has length L and d has length N, then $\mathbf{conv}(c, d)$ has length $L + N - 1$.

Equation (5) will show that the nth component of $c * d$ is $\sum c_k d_{n-k}$.

These subscripts k and $n-k$ add to n. We are collecting **all pairs $\boldsymbol{c_k w^k}$ and $\boldsymbol{d_{n-k} w^{n-k}}$ whose product yields $\boldsymbol{w^n}$.** This eye-catching $k + (n-k) = n$ is the mark of a convolution.

Cyclic convolution has no MATLAB command. But $c \circledast d$ is easy to construct by folding back the non-cyclic part. Now c and d and $c \circledast d$ have the same length N:

```
q      = [ conv(c,d)  0 ];     % The extra zero gives length 2N
cconv  = q(1:N) + q(N+1:N+N);  % cconv = c ⊛ d has length N
```

The n^{th} component of $c \circledast d$ is still a sum of $c_k d_l$, but now $k + l = n \ (mod\ N) =$ remainder after dividing by N. That expression $\boldsymbol{mod\ N}$ is the cyclic part, coming from $w^N = 1$. So $\mathbf{1 + 2 = 0}(\boldsymbol{mod}\ \mathbf{3})$. Here are convolution and cyclic convolution:

Discrete convolution $(c * d)_n = \sum c_k d_{n-k} \quad (c \circledast d)_n = \sum c_k d_l$ for $k + l = n (mod\ N)$ (4)

Infinite Convolution

Convolution still applies when $f(x)$ and $g(x)$ and $f(x)g(x)$ have infinitely many terms. We are ready to see the rule for $c * d$, when $\sum c_k e^{ikx}$ multiplies $\sum d_l e^{ilx}$. *What is the coefficient of e^{inx} in the result?*

1. e^{inx} comes from multiplying e^{ikx} times e^{ilx} when $k + l = n$.
2. The product $(c_k e^{ikx})(d_l e^{ilx})$ equals $c_k d_l e^{inx}$ when $k + l = n$.
3. The e^{inx} term in $f(x)g(x)$ contains ***every product $c_k d_l$ in which $l = n - k$.***

Add these products $c_k d_l = c_k d_{n-k}$ to find the coefficient of e^{inx}. Convolution combines all products $c_k d_{n-k}$ whose indices add to n:

Infinite Convolution The nth component of $c * d$ is $\displaystyle\sum_{k=-\infty}^{\infty} c_k d_{n-k}$. (5)

Example 3 The "identity vector" δ in convolution has exactly one nonzero coefficient $\delta_0 = 1$. Then δ gives the Fourier coefficients of the unit function $f(x) = 1$. Multiplying $f(x) = 1$ times $g(x)$ gives $g(x)$, so convolving $\delta * d$ gives d:

$$\boldsymbol{\delta * d} = (\ldots, 0, 1, 0, \ldots) * (\ldots, d_{-1}, d_0, d_1, \ldots) = \boldsymbol{d}. \tag{6}$$

The only term in $\sum \delta_k d_{n-k}$ is $\delta_0 d_n$. That term is d_n. So $\delta * d$ recovers d.

Example 4 The **autocorrelation** of a vector c is the convolution of c with its "flip" or "conjugate transpose" or "time reversal" $d(n) = \overline{c(-n)}$. The real signal $c = (1, 2, 4)$ has $d_0 = 1$ and $d_{-1} = 2$ and $d_{-2} = 4$. The convolution $c * d$ is the autocorrelation of c:

Autocorrelation $(\ldots, 1, 2, 4, \ldots) * (\ldots, 4, 2, 1, \ldots) = (\ldots, 4, 10, 21, 10, 4, \ldots)$. (7)

The dots all represent zeros. The autocorrelation $4, 10, 21, 10, 4$ is symmetric around the zero position. To be honest, I did not actually use the convolution formula $\sum c_k d_{n-k}$. It is easier to multiply $f(x) = \sum c_k e^{ikx}$ times its conjugate $\overline{f(x)} = \sum \bar{c}_k e^{-ikx} = \sum d_k e^{ikx}$:

$$\begin{array}{ccc} \boldsymbol{f(x)} & \overline{\boldsymbol{f(x)}} & \boldsymbol{|f(x)|^2} \\ (1 + 2e^{ix} + 4e^{2ix})\ (4e^{-2ix} + 2e^{-ix} + 1\ ) & = & 4e^{-2ix} + 10e^{-ix} + 21 + 10e^{ix} + 4e^{2ix} \\ c_0 \quad c_1 \quad c_2 \quad d_{-2} \quad d_{-1} \quad d_0 & & \text{autocorrelation of } c \end{array} \tag{8}$$

This answer $f(x)\overline{f(x)}$ (often written $f(x)f^*(x)$) is always real, and never negative.

Note The autocorrelation $f(t) * \overline{f(-t)}$ is extremely important. Its transform is $|c_k|^2$ (discrete case) or $|\widehat{f}(k)|^2$ (continuous case). That transform is never negative. This is the **power spectral density** that appears in Section 4.5.

In MATLAB, the autocorrelation of c is conv$(c,$ fliplr$(c))$. The left-right flip produces the correct $d(n) = c(-n)$. When the vector c is complex, use conj(fliplr(c)).

Convolution of Functions

Reverse the process and multiply c_k times d_k. Now the numbers $2\pi c_k d_k$ are the coefficients of the convolution $f(x) * g(x)$. This is a 2π-periodic convolution because $f(x)$ and $g(x)$ are periodic. Instead of the sum of $c_k d_{n-k}$ in convolving coefficients, we have the integral of $f(t)g(x-t)$ in convolving functions.

Please notice: *The indices k and $n-k$ add to n. Similarly t and $x-t$ add to x:*

Convolution of Periodic Functions
$$(f * g)(x) = \int_0^{2\pi} f(t)\, g(x-t)\, dt\,. \tag{9}$$

Example 5 Convolve $f(x) = \sin x$ with itself. Check $2\pi c_k d_k$ in the convolution rule.

Solution The convolution $(\sin x) * (\sin x)$ is $\int_0^{2\pi} \sin t \sin(x-t) dt$. Separate $\sin(x-t)$ into $\sin x \cos t - \cos x \sin t$. The integral produces (to my surprise) $-\pi \cos x$:

$$(\sin x) * (\sin x) = \sin x \int_0^{2\pi} \sin t \cos t\, dt - \cos x \int_0^{2\pi} \sin^2 t\, dt = -\pi \cos x\,.$$

For $(\sin x) * (\sin x)$, the convolution rule has $c_k = d_k$. The coefficients of $\sin x = \frac{1}{2i}(e^{ix} - e^{-ix})$ are $\frac{1}{2i}$ and $-\frac{1}{2i}$. Square them to get $-\frac{1}{4}$ and multiply by 2π. Then $2\pi c_k d_k = -\frac{\pi}{2}$ gives the correct coefficients of $-\pi \cos x = -\frac{\pi}{2}(e^{ix} + e^{-ix})$.

Note that *autocorrelation* would convolve $f(x) = \sin x$ with $f(-x) = -\sin x$. The result is $+\pi \cos x$. Its coefficients $+\frac{\pi}{2}$ are now positive because they are $2\pi|c_k|^2$.

Example 6 If $I(x)$ is the integral of $f(x)$ and $D(x)$ is the derivative of $g(x)$, show that $\boldsymbol{I * D = f * g}$. *Give the reason in x-space and also in k-space.* This is my favorite.

Solution In k-space, $I * D = f * g$ is quick from the rules for integrals and derivatives. The integral $I(x)$ has coefficients $\boldsymbol{c_k/ik}$ and the derivative $D(x)$ has coefficients $\boldsymbol{ik\, d_k}$. Multiplying those coefficients, ik cancels $1/ik$. The same $c_k d_k$ appears for $I * D$ and $f * g$. Actually we should require $c_0 = 0$, to avoid dividing by $k = 0$.

In x-space, we use integration by parts (a great rule). The integral of $f(t)$ is $I(t)$. The derivative of $g(x-t)$ is *minus* $D(x-t)$. Since our functions are periodic, the integrated term $I(t)g(x-t)$ is the same at 0 and 2π. It vanishes to leave $f * g = I * D$.

After those examples, we confirm that $\boldsymbol{f * g}$ *has coefficients* $\boldsymbol{2\pi c_k d_k}$. First,

$$\int_0^{2\pi} (f * g)(x) e^{-ikx} dx = \int_0^{2\pi} \left[\int_0^{2\pi} f(t) g(x-t) e^{-ik[t+(x-t)]}\, dt \right] dx\,. \tag{10}$$

With the x-integral first, bring out $f(t)e^{-ikt}$. This separates (10) into two integrals:

$$\int_0^{2\pi} f(t) e^{-ikt} dt \int_0^{2\pi} g(x-t) e^{-ik(x-t)} dx \quad \text{which is} \quad (2\pi c_k)(2\pi d_k)\,. \tag{11}$$

In that last integral we substituted s for $x-t$. The new limits $s = 0-t$ and $s = 2\pi - t$ still cover a full period, and the integral is $2\pi d_k$. Dividing (10) and (11) by 2π gives the function / coefficient convolution rule: ***The coefficients of $f*g$ are $2\pi c_k d_k$.***

Cyclic Convolution Rules

This section began with the cyclic convolution $(1,2,4) \circledast (3,0,5) = (13,26,17)$. Those are the coefficients in $fg = (1+2w+4w^2)(3+5w^2) = (13+26w+17w^2)$ when $w^3 = 1$. A useful check is to set $w = 1$. Adding each set of coefficients gives $(7)(8) = (56)$.

The discrete convolution rule connects this cyclic convolution $c \circledast d$ with a multiplication of function values $f_j g_j$. Please keep the vectors f and g in j-space separate from c and d in k-space. The convolution rule *does not say* that $c_k \circledast d_k$ equals $f_k g_k$!

The correct rule for $c \circledast d$ transforms the vector with components $f_j g_j$ back into k-space. Writing $f = Fc$ and $g = Fd$ produces an identity that is true for all vectors. In MATLAB, the entry by entry product $(f_0 g_0, \ldots, f_{N-1} g_{N-1})$ is the N-vector $f .* g$. The dot removes summation and leaves N separate components $f_j g_j$. Of course $*$ in MATLAB does *not* mean convolution, and components are numbered 1 to N:

Cyclic Convolution: Multiply in j-space $c \circledast d$ is $F^{-1}((Fc) .* (Fd))$ (12)

In MATLAB, this is $N *$ fft(ifft(c) .* ifft(d)).

Suppose the convolution is $f \circledast g$. This is a **multiplication in k-space**. Possibly the reason for no cyclic convolution command in MATLAB is the simplicity of this one-line code for cconv(f, g). It copies (12) with ifft and fft reversed:

$$c = \text{fft}(f); \quad d = \text{fft}(g); \quad cd = c .* d; \quad f \circledast g = \text{ifft}(cd); \tag{13}$$

Combined into one command this cconv is ifft(fft(f) .* fft(g)). The factor N disappears when we do it this way, multiplying in k-space.

I have to say that the convolution rule is more bad news for the third-grade teacher. Long multiplication is being taught the slow way (as all third-graders have suspected). When we multiply N-digit numbers, we are doing N^2 separate multiplications for the convolution. *This is inefficient.* Three FFT's make convolution much faster.

One more thing. We may want $c * d$ and the FFT is producing the cyclic $c \circledast d$. To fix that, add $N-1$ zeros to c and d so that cyclic and non-cyclic convolution involve exactly the same multiplications. If c and d have length N, $c * d$ has length $2N-1$:

$$C = [\ c \quad \text{zeros}(1, N-1)\ ]; \quad D = [\ d \quad \text{zeros}(1, N-1)\ ]; \quad \text{then } c * d \text{ is } C \circledast D. \tag{14}$$

For $N = 2$, this $c * d$ is $(c_0, c_1, 0) \circledast (d_0, d_1, 0) = (c_0 d_0, c_0 d_1 + c_1 d_0, c_1 d_1) = C \circledast D$.

Convolution by Matrices

You have the essential ideas of $c * d$ and $c \circledast d$—their link to multiplication allows us to convolve quickly. I want to look again at those sums $\sum c_k d_{n-k}$, to see a matrix C multiplying a vector d. Convolution is linear so there has to be a matrix.

In the cyclic case, C is a **circulant matrix** C_N. The non-cyclic case has an infinite **constant-diagonal matrix** C_∞ (called a *Toeplitz matrix*). Here are those convolution matrices, cyclic C_N and non-cyclic C_∞, with the c's in every row and column. Notice how the diagonals wrap around in C_N:

Circulant matrix
$$C_N d = \begin{bmatrix} c_0 & c_{N-1} & \cdot & \cdot & c_1 \\ c_1 & c_0 & c_{N-1} & \cdot & c_2 \\ c_2 & c_1 & c_0 & \cdot & \cdot \\ \cdot & \cdot & c_1 & c_0 & \cdot \\ c_{N-1} & \cdot & c_2 & c_1 & c_0 \end{bmatrix} \begin{bmatrix} d_0 \\ d_1 \\ \cdot \\ \cdot \\ d_{N-1} \end{bmatrix} = c \circledast d \qquad (15)$$

Toeplitz matrix
$$C_\infty d = \begin{bmatrix} \cdot & \cdot & c_{-2} & \cdot & \cdot \\ \cdot & c_0 & c_{-1} & c_{-2} & \cdot \\ c_2 & c_1 & c_0 & c_{-1} & c_{-2} \\ \cdot & c_2 & c_1 & c_0 & \cdot \\ \cdot & \cdot & c_2 & \cdot & \cdot \end{bmatrix} \begin{bmatrix} \cdot \\ d_{-1} \\ d_0 \\ d_1 \\ \cdot \end{bmatrix} = c * d \qquad (16)$$

For the circulant C_N, the all-important polynomial is $C(w) = c_0 + c_1 w + \cdots + c_{N-1} w^{N-1}$. Multiplying by $D(w)$ shows $c \circledast d$ when $w^N = 1$. For the infinite matrix, $C_\infty d$ is multiplying infinite Fourier series: w becomes e^{ix} and all $|w| = 1$ are included. Section 4.6 will show that the matrices are invertible if and only if $C(w) \neq 0$ (the inverse has to divide by C). They are positive definite if and only if $C(w) > 0$. In this case $C(w) = |F(w)|^2$ and c is the autocorrelation of f as in (8). For the matrices, this is the Cholesky factorization $C = F^{\mathrm{T}} F$.

Processing a Signal by a Filter

Filtering (the key step in signal and image processing) is a convolution. One example is a running average A (*lowpass filter*). The second difference matrix $D = K/4$ is a *highpass filter*. For now we are pretending that the signal has no start and no end. With a long signal, like the audio on a CD, endpoint errors are not a serious problem. So we use the infinite matrix, not the finite wrap-around circulant matrix.

The "second average" filter A has centered coefficients $\frac{1}{4}, \frac{2}{4}, \frac{1}{4}$ that add to 1:

Output at time n = average of three inputs
$$y_n = \frac{1}{4} x_{n-1} + \frac{2}{4} x_n + \frac{1}{4} x_{n+1}.$$

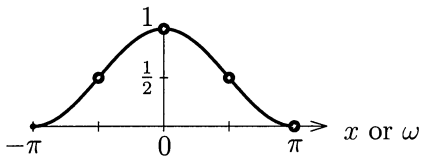
In matrix notation $y = a * x$ is $y = Ax$. The filter matrix A is "Toeplitz":

Averaging filter is a convolution $a = \frac{1}{4}(.,1,2,1,.)$

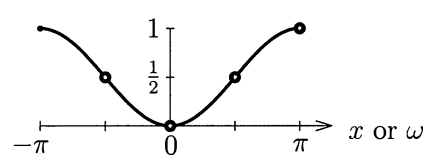
$$\begin{bmatrix} . \\ y_0 \\ y_1 \\ y_2 \\ . \end{bmatrix} = \frac{1}{4}\begin{bmatrix} . & . & & & \\ 1 & 2 & 1 & & \\ & 1 & 2 & 1 & \\ & & 1 & 2 & . \\ & & & . & . \end{bmatrix}\begin{bmatrix} x_{-1} \\ x_0 \\ x_1 \\ x_2 \\ . \end{bmatrix} = Ax = a * x. \quad (17)$$

When the input is $\boldsymbol{x}_{\mathbf{low}} = (.,1,1,1,1,.)$, the output is $y = x$. This zero frequency DC component passes unchanged through the filter, which is **lowpass**. The highest frequency input is the alternating vector $\boldsymbol{x}_{\mathbf{high}} = (.,1,-1,1,-1,.)$. In that case the output is $y = (0,0,0,0)$, and the highest frequency $\omega = \pi$ is stopped.

A lowpass filter like A will remove noise from the signal (since random noise tends to be high frequency). But filtering also blurs significant details in the input x. The big problem of signal processing is to choose the best filter.



lowpass $\boldsymbol{A(\omega)} = \frac{1}{2}(1 + \cos\omega)$



highpass $\boldsymbol{D(\omega)} = \frac{1}{2}(1 - \cos\omega)$

Figure 4.11: Frequency responses of the second average filter A and second difference filter D. The highpass response $D(\omega)$ is the lowpass response $A(\omega)$ shifted by π.

Frequency response $A(\omega) = \frac{1}{4}e^{-i\omega} + \frac{2}{4} + \frac{1}{4}e^{i\omega} = \frac{1}{2}(1 + \cos\omega)$ $\quad A(0) = 1$, $A(\pi) = 0$ (18)

Figure 4.11 is a graph of the frequency response $A(\omega)$, which is also written $A(e^{i\omega})$. The second graph shows the frequency response to a highpass filter $D = K/4$.

Highpass filter is $\boldsymbol{D = K/4}$

$$D = \frac{1}{4}\begin{bmatrix} . & . & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & -1 & \\ & & -1 & 2 & -1 \\ & & & . & . \end{bmatrix} \qquad \text{Output } y = Dx = d * x.$$

Now the lowest frequency $\omega = 0$ (the DC term) is stopped by the filter. The all-ones input $x(n) = 1$ produces a zero output $y(n) = 0$. The highest frequency $\omega = \pi$ is passed: the alternating input $x(n) = (-1)^n$ has $Dx = x$ with eigenvalue 1. The frequencies between 0 and π have eigenvalues $D(\omega)$ between 0 and 1 in Figure 4.11b:

Highpass response $D(\omega) = -\frac{1}{4}e^{-i\omega} + \frac{1}{2} - \frac{1}{4}e^{i\omega} = \frac{1}{2}(1 - \cos\omega).$ (19)

All pure frequencies $-\pi \le \omega \le \pi$ give eigenvectors with components $x(n) = e^{-i\omega n}$. The extreme low frequency $\omega = 0$ gave $x(n) = 1$, and the highest frequency $\omega = \pi$ gave $x(n) = (-1)^n$. The key to understanding filters is to see the response $y(n)$ or y_n or $y[n]$ to the pure input $x(n) = e^{-i\omega n}$. That response is just $A(\omega)e^{-i\omega n}$.

Better Filters

The truth is that the filters A and D are not very sharp. The purpose of a filter is to preserve a band of frequencies and destroy another band. Figure 4.11 only goes gradually between 1 and 0. The response $I(\omega)$ from an **ideal lowpass filter** is exactly 1 or 0. But we can't achieve that ideal with a finite number of filter coefficients. Figure 4.12 shows a nearly ideal FIR filter.

The vector a is called the **impulse response**, since $a * \delta = a$. Its entries are the Fourier coefficients of $A(\omega) = \sum a_n e^{-i\omega n}$. The filter is **FIR** when it has *finite* impulse response—only $d+1$ nonzero coefficients a_n. The ideal lowpass filter is **IIR** because the Fourier coefficients of the box function $A(\omega)$ come from the sinc function.

Which polynomial to choose? If we chop off the ideal filter, the result is not good! Truncating the Fourier series for the box function produces *large overshoot* in the Gibbs phenomenon. That truncation minimizes the energy in the error (mean square error), but the maximum error and the oscillations are unacceptable.

A popular choice is an **equiripple filter**. The oscillations in the frequency response $A(\omega)$ all have the same height (or depth) in Figure 4.12. If we try to reduce the error at one of those maximum points, other errors would become larger. **A polynomial of degree d cannot change sign at $d+2$ successive points. When the error has $d+2$ equal ripples, the maximum error is minimized.**

The command firpm (previously remez) will design this equiripple symmetric filter of length $30 + 1$. The passband-stopband interval is $.49 \le f \le .51$. The Signal Processing Toolbox normalizes by $f = \omega/\pi \le 1$ (help firpm specifies the inputs).

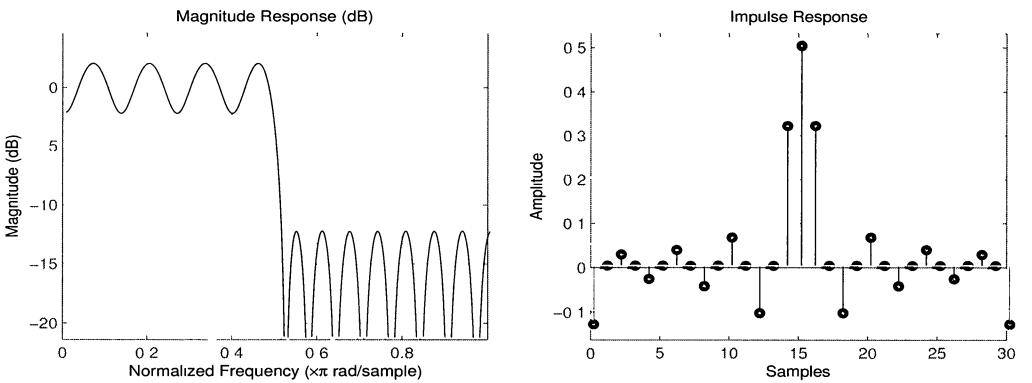


Figure 4.12: $A(\omega)$ and $a =$ firpm$(30, [0,\ .49,\ .51,\ 1],\ [1\ 1\ 0\ 0])$. Use fvtool$(a)$.

Finite Length Signals

The key point for this book is that *filters are convolutions*: needed constantly. We are seeing the second difference matrices in a new context, as highpass filters. A is an infinite matrix when the signals $x(n)$ and $y(n)$ extend over $-\infty < n < \infty$. If we want to work with finite length signals, one way is to assume *wraparound*. The signal becomes periodic. The infinite Toeplitz matrices giving $a * x$ and $d * x$ become N by N **circulant matrices** giving $a \circledast x$ and $d \circledast x$:

Periodic signals
Circulant matrices
Cyclic convolution

$$A = \frac{1}{4}\begin{bmatrix} 2 & 1 & 0 & 1 \\ 1 & 2 & 1 & 0 \\ 0 & 1 & 2 & 1 \\ 1 & 0 & 1 & 2 \end{bmatrix} \quad \text{and} \quad D = \frac{1}{4}\begin{bmatrix} 2 & -1 & 0 & -1 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ -1 & 0 & -1 & 2 \end{bmatrix} \tag{20}$$

The right side of Figure 4.11 shows the frequency response function $D(e^{i\omega})$ for this second difference (*highpass*) filter, at the four frequencies $\omega = 0, \pm\pi/2, \pi$:

$$D(e^{i\omega}) = -\frac{1}{4}e^{-i\omega} + \frac{2}{4} - \frac{1}{4}e^{i\omega} = \frac{1}{2}(1 - \cos\omega) \quad \text{has values} \quad \lambda = 0, \frac{1}{2}, \frac{1}{2}, 1. \tag{21}$$

The lowest frequency $\omega = 0$ corresponds to the DC input $x = (1, 1, 1, 1)$. This is killed by the filter ($\lambda = 0$ because $2 - 1 - 1 = 0$). The second differences of a constant are zero. The highest frequency $\omega = \pi$ corresponds to the AC input $x = (1, -1, 1, -1)$ which is passed by the filter, $Dx = x$. In between, the inputs $(1, i, -1, -i)$ and $(1, -i, -1, i)$ at $\omega = \pm\frac{\pi}{2}$ have outputs multiplied by $\lambda = \frac{1}{2}$. The discrete cosine $(1, 0, -1, 0)$ and discrete sine $(0, 1, 0, -1)$ are combinations of those eigenvectors of D.

The eigenvalues of D are the discrete transform of the filter coefficients:

$$\text{Eigenvalues } \mathsf{eig}(D) = 0, \tfrac{1}{2}, 1, \tfrac{1}{2} \overset{\textbf{transform}}{\longleftrightarrow} \text{coefficients } d_k = \tfrac{2}{4}, -\tfrac{1}{4}, 0, -\tfrac{1}{4}.$$

It is not surprising that signal processing theory is mostly in the *frequency domain*. The response function tells us everything. The filter could even be implemented by the convolution rule. But here we would certainly compute Cx and Dx directly from x, with a circuit composed of multipliers and adders and delays.

We end with a light-hearted puzzle involving these two particular filters.

Puzzle. The matrices A and D illustrate a strange possibility in linear algebra. They have the *same eigenvalues and eigenvectors*, but they are not the same matrix. This seems incredible, since both matrices factor into $S\Lambda S^{-1}$ (with S = eigenvector matrix and Λ = eigenvalue matrix). How can A and D be different matrices?

The trick is in the ordering. The eigenvector $(1, 1, 1, 1)$ goes with the eigenvalue $\lambda = 1$ of A and $\lambda = 0$ of D. The oscillating eigenvector $(1, -1, 1, -1)$ has the opposite eigenvalues. *The columns of F^{-1} (also of F) are the eigenvectors for all circulants*, explained in Section 4.6. But Λ can have the eigenvalues $1, \frac{1}{2}, \frac{1}{2}, 0$ in different orders.

Worked Example: Convolution of Probabilities

Suppose p_i is the probability that a random variable equals i ($p_i \geq 0$ and $\sum p_i = 1$). *For the sum* $i+j$ of two independent samples, what is the probability c_k that $i+j=k$? For two dice, what is the probability c_7 of rolling a 7, when each $p_i = \frac{1}{6}$?

Solution 1 The sample i followed by j will appear with probability $p_i p_j$. The result $i+j=k$ is the union of mutually exclusive events (sample i followed by $j=k-i$). That probability is $p_i p_{k-i}$. Combining all the ways to add to k yields a **convolution:**

$$\textbf{Probability of } \boldsymbol{i+j=k} \qquad c_k = \sum p_i\, p_{k-i} \quad \text{or} \quad c = p * p. \tag{22}$$

For each of the dice, the probability of $i = 1, 2, \ldots, 6$ is $p_i = 1/6$. For two dice, the probability of rolling $k = 12$ is $\frac{1}{36}$. For $k = 11$ it is $\frac{2}{36}$, from $5+6$ and $6+5$. Two dice produce $k = 2, 3, \ldots, 12$ with probabilities $\boldsymbol{p * p = c}$ (box $*$ box $=$ hat):

$$\frac{1}{6}(1,1,1,1,1,1) * \frac{1}{6}(1,1,1,1,1,1) = \frac{1}{36}(1,2,3,4,5,6,5,4,3,2,1). \tag{23}$$

Solution 2 The "**generating function**" is $P(z) = (z + z^2 + \cdots + z^6)/6$, the polynomial with coefficients p_i. For two dice the generating function is $\boldsymbol{P^2(z)}$.

This is $C(z) = \frac{1}{36}z^2 + \frac{2}{36}z^3 + \cdots + \frac{1}{36}z^{12}$ (coefficients c_k times powers z^k), by the convolution rule for (23). Multiply P times P when you convolve p with p.

Repeated Trials: Binomial and Poisson

Binomial probabilities come from n coin flips. The chance of heads is p on each flip. The chance b_i of i heads in n flips comes from convolving n copies of $(1-p, p)$:

Binomial $b_i = \binom{n}{i} p^i (1-p)^{n-i}$ from $(1-p, p) * \cdots * (1-p, p)$

Generating function $\boldsymbol{B(z) = (pz + 1 - p)^n}$

The factor $pz + 1 - p$ is the simple generator for one trial (probability p and $1-p$ of events 1 and 0). By taking the n^{th} power, b_i is the correct probability for the sum of n independent samples: convolution rule! Differentiating $B(z)$ at $z = 1$ gives the mean np (expected number of heads in n flips). Now try the Poisson distribution:

Poisson probabilities $p_i = e^{-\lambda}\, \lambda^i / i!$

Generating function $\boldsymbol{P(z)} = \sum p_i z^i = e^{-\lambda} \sum \lambda^i z^i / i! = \boldsymbol{e^{-\lambda} e^{\lambda z}}$

Squaring that generating function, $P^2 = e^{-2\lambda} e^{2\lambda z}$ is correct for the sum of two Poisson samples. So the sum is still Poisson with parameter 2λ. And differentiating $P(z)$ at $z = 1$ gives mean $= \lambda$ for each sample.

The central limit theorem looks at the sum of n samples as $n \to \infty$. It tells us that the (scaled) limit of many convolutions is Gaussian.

Problem Set 4.4

1 (from age 7) **When you multiply numbers you are convolving their digits.** We have to "carry" numbers in actual multiplication, while convolution leaves them in the same decimal place. What is t?

$$(12)(15) = (180) \quad \text{but} \quad (\ldots, 1, 2, \ldots) * (\ldots, 1, 5, \ldots) = (\ldots, 1, 7, t, \ldots).$$

2 Check the cyclic convolution rule $F(c \circledast d) = (Fc) .* (Fd)$ directly for $N = 2$:

$$F = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad Fc = \begin{bmatrix} c_0 + c_1 \\ c_0 - c_1 \end{bmatrix} \quad Fd = \begin{bmatrix} d_0 + d_1 \\ d_0 - d_1 \end{bmatrix} \quad c \circledast d = \begin{bmatrix} c_0 d_0 + c_1 d_1 \\ c_0 d_1 + c_1 d_0 \end{bmatrix}$$

3 Factor the 2 by 2 circulant $C = \begin{bmatrix} c_0 & c_1 \\ c_1 & c_0 \end{bmatrix}$ into $F^{-1}\mathsf{diag}(Fc)F$ from Problem 2.

4 The right side of (12) shows the fast way to convolve. Three fast transforms will compute Fc and Fd and transform back by F^{-1}. For $N = 128, 1024, 8192$ create random vectors c and d. Compare tic; cconv(c, d); toc; with this FFT way.

5 Write the steps to prove the Cyclic Convolution Rule (13) following this outline: $F(c \circledast d)$ has entries $\sum(\sum c_n d_{k-n})w^{jk}$. The inner sum on n produces $c \circledast d$ and the outer sum on k multiplies by F. Write w^{jk} as w^{jn} times $w^{j(k-n)}$. When you sum first on k and last on n, the double sum splits into $\sum c_n w^{jn} \sum d_k w^{jk}$.

6 What is the identity vector δ_N in cyclic convolution? It gives $\delta_N \circledast d = d$.

7 Which vectors s and s_N give ***one-step delays***, noncyclic and cyclic?

$s * (\ldots, d_0, d_1, \ldots) = (\ldots, d_{-1}, d_0, \ldots)$ and $s_N \circledast (d_0, \ldots, d_{N-1}) = (d_{N-1}, d_0, \ldots)$.

8 (a) Compute directly the convolution $f \circledast f$ (cyclic convolution with $N = 6$) when $f = (0, 0, 0, 1, 0, 0)$. Connect $(f_0, \ldots, f_5)$ with $f_0 + f_1 w + \cdots + f_5 w^5$.

(b) What is the Discrete Transform $c = (c_0, c_1, c_2, c_3, c_4, c_5)$ of this f?

(c) Compute $f \circledast f$ by using c in "transform space" and transforming back.

9 Multiplying infinite Toeplitz matrices $C_\infty D_\infty$ is convolution $c*d$ of the numbers on their diagonals. If C_∞ in (16) has $c_0 = 1$, $c_1 = 2$, $c_2 = 4$, then C_∞^{T} is upper triangular. Multiply $C_\infty C_\infty^{\mathrm{T}}$ to see the *autocorrelation* $(1, 2, 4) * (4, 2, 1) = (4, 10, 21, 10, 4)$ on its diagonals. Why is that matrix positive definite? [Multiplying circulant matrices is *cyclic* convolution of diagonal entries.]

10 The chance of grade $i = (70, 80, 90, 100)$ on one quiz is $p = (.3, .4, .2, .1)$. What are the probabilities c_k for the sum of two grades to be $k = (140, 150, \ldots, 200)$? You need to convolve $c = p * p$ or multiply 3421 by 3421 (without carrying).

11 What is the expected value (mean m) for the grade on that quiz? The generating function is $P(z) = .3z^{70} + .4z^{80} + .2z^{90} + .1z^{100}$. Show that $m = p'(1)$.

12 What is the mean M for the total grade on two quizzes, with those probabilities c_k? I expect $M = 2m$. The derivative of $(P(z))^2$ is $2P(z)P'(z) = (2)(1)(m)$ at $z = 1$.

13 With 9 coefficients, which firpm filter is closest to ideal in Figure 4.12?

4.5 FOURIER INTEGRALS

A Fourier series is perfect for a 2π-periodic function. The only frequencies in $\sum c_k e^{ikx}$ are whole numbers k. *When $f(x)$ is not periodic, all frequencies k are allowed.* That sum has to be replaced by an integral $\int \widehat{f}(k)e^{ikx}\,dk$ over $-\infty < k < \infty$.

The **Fourier transform** $\widehat{f}(k)$ measures the presence of e^{ikx} in the function $f(x)$. In changing from c_k to $\widehat{f}(k)$, you will see how the important things survive.

I can write the integral transforms by analogy with the formulas of Fourier series:

Transform $f(x)$ to $\widehat{f}(k)$
$$c_k = \frac{1}{2\pi}\int_{-\pi}^{\pi} f(x)e^{-ikx}\,dx \quad \text{becomes} \quad \widehat{f}(k) = \int_{-\infty}^{\infty} f(x)e^{-ikx}\,dx \qquad (1)$$

Reconstruction $\widehat{f}(k)$ to $f(x)$
$$f(x) = \sum_{k=-\infty}^{\infty} c_k e^{-ikx} \quad \text{becomes} \quad f(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \widehat{f}(k)e^{ikx}\,dk \qquad (2)$$

The analysis step (1) finds the density $\widehat{f}(k)$ of each pure oscillation e^{ikx}, inside $f(x)$. The synthesis step (2) combines all those oscillations $\widehat{f}(k)e^{ikx}$, to reconstruct $f(x)$.

At the zero frequency $k = 0$, notice $\widehat{f}(0) = \int_{-\infty}^{\infty} f(x)\,dx$. This is the area under the graph of $f(x)$. Thus $\widehat{f}(0)$ compares with the *average value* c_0 in a Fourier series.

We expect the graph of $|f(x)|$ to enclose a finite area. In applications $f(x)$ might drop off as quickly as e^{-x} or e^{-x^2}. It might have a "heavy tail" and decay like a power of $1/x$. **The smoothness of $f(x)$ controls the dropoff in the transform $\widehat{f}(k)$.** We approach this subject by examples—here are the first five.

Five Essential Transforms

Example 1 The transform of $f(x) =$ **delta function** $= \delta(x)$ is a constant (no decay):

$$\widehat{f}(k) = \widehat{\delta}(k) = \int_{-\infty}^{\infty} \delta(x)\, e^{-ikx}\,dx = 1 \quad \text{for all frequencies } k\,. \qquad (3)$$

The integral picks out the value 1 of e^{-ikx}, at the "spike point" $x = 0$.

Example 2 The transform of a **centered square pulse** is a **sinc function** of k:

Square pulse
$$f(x) = \left\{ \begin{array}{ll} 1 & -L \le x \le L \\ 0 & |x| > L \end{array} \right\} = \text{box function}$$

The integral from $-\infty$ to ∞ reduces to an easy integral from $-L$ to L. Notice $\widehat{f}(0) = 2L$:

$2L$ sinc kL
$$\widehat{f}(k) = \int_{-L}^{L} e^{-ikx}\,dx = \frac{e^{-ikL} - e^{ikL}}{-ik} = \frac{2\sin kL}{k} \qquad (4)$$

Example 3 The transform of a **one-sided decaying pulse** is $1/(a+ik)$:

Exponential decay
$$f(x) = \begin{cases} e^{-ax} & x \geq 0 \\ 0 & x < 0 \end{cases}$$

Now the integral is from 0 to ∞, and we integrate $e^{-(a+ik)x}$. The area is $\widehat{f}(0) = \frac{1}{a}$:

Pole at $k = -ia$
$$\widehat{f}(k) = \int_0^\infty e^{-ax}e^{-ikx}\,dx = \left[\frac{e^{-(a+ik)x}}{-(a+ik)}\right]_0^\infty = \frac{1}{a+ik}. \tag{5}$$

We are assuming $a > 0$ (for decay). It is somehow very pleasant to use $e^{-a\infty} = 0$. This transform $1/(a+ik)$ drops off slowly, like $1/k$, because $f(x)$ has a jump at $x = 0$.

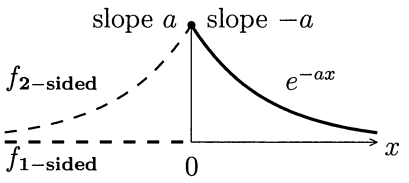
Example 4 An **even decaying pulse** has an even transform $\widehat{f}(k) = 2a/(a^2+k^2)$:

Two-sided pulse
$$f(x) = e^{-a|x|} = \begin{cases} e^{-ax} & \text{for } x \geq 0 \\ e^{ax} & \text{for } x \leq 0 \end{cases}$$

One-sided + one-sided
$$\widehat{f}(k) = \frac{1}{a+ik} + \frac{1}{a-ik} = \frac{2a}{a^2+k^2}. \tag{6}$$

We are adding two one-sided pulses, so add their transforms. The even pulse in Figure 4.13 has no jump at $x = 0$. But the slope drops from a to $-a$, so $\widehat{f}(k)$ decays like $2a/k^2$.

Real even functions $f(x) = f(-x)$ still lead to cosines. For the Fourier integral that means $\widehat{f}(k) = \widehat{f}(-k)$, since $\cos kx = (e^{ikx} + e^{-ikx})/2$. Real odd functions lead to sines, and $\widehat{f}(k)$ is imaginary and odd in Example 6.



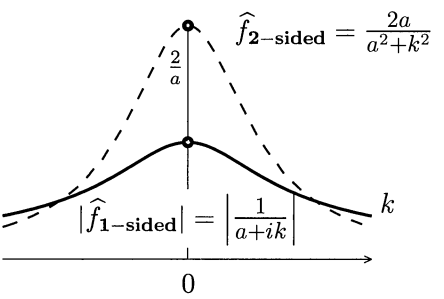


Figure 4.13: The one-sided pulse has a jump at $x = 0$, and slow $1/k$ decay in $\widehat{f}(k)$. The two-sided pulse has a corner at $x = 0$, and faster $1/k^2$ decay in $2a/(a^2+k^2)$.

Example 5 The transform of $f(x) =$ **constant function** $= 1$ is a delta $\widehat{f}(k) = 2\pi\delta(k)$.

This is a dangerous example, because $f(x) = 1$ encloses infinite area. I see it best as the limiting case $a \to 0$ in Example 4. Certainly $e^{-a|x|}$ approaches 1 as the decay rate a goes to zero. For all frequencies $k \neq 0$, the limit of $\widehat{f}(k) = 2a/(a^2+k^2)$ is $\widehat{f}(k) = 0$.

At the frequency $k = 0$ we need a delta function times 2π to recover $f(x) = 1$:

$$\text{Equation (2) reconstructs} \quad e^{-a|x|} = \frac{1}{2\pi}\int_{-\infty}^{\infty} \frac{2a}{a^2+k^2} e^{ikx} dk \tag{7}$$

$$\text{As } a \to 0 \text{ this becomes} \quad 1 = \frac{1}{2\pi}\int_{-\infty}^{\infty} 2\pi\delta(k) e^{ikx} dk \tag{8}$$

To understand equations (1) and (2), start with the Fourier series. Key idea: *Use a period T much larger than 2π.* The function $f_T(x)$ is chosen to agree with $f(x)$ from $-T/2$ to $T/2$, and then continue with period T. As $T \to \infty$ the Fourier series for f_T should approach (with the right scaling) the Fourier integral.

When the period is T instead of 2π the coefficient c_k of e^{iKx} comes from $f_T(x)$:

Period T with $K = k\dfrac{2\pi}{T}$

$$c_k = \frac{1}{T}\int_{-T/2}^{T/2} f_T(x) e^{-iKx}\, dx \tag{9}$$

The exponentials e^{iKx} have the right period T. They combine to reproduce $f_T(x)$:

Fourier series with period T

$$f_T(x) = \sum_{k=-\infty}^{\infty} c_k e^{iKx} = \sum_{k=-\infty}^{\infty} \frac{1}{T}\left[\int_{-T/2}^{T/2} f_T(x) e^{-iKx}\, dx\right] e^{iKx}. \tag{10}$$

As T gets larger, the function $f_T(x)$ agrees with $f(x)$ over a longer interval. **The sum from $k = -\infty$ to ∞ approaches an integral.** Each step in the sum changes k by 1, so K changes by $2\pi/T$; that is ΔK. We replace $1/T$ by $\Delta K/2\pi$. As $T \to \infty$, the sum in (10) becomes an integral with respect to K, and f_T approaches f:

Transform to $\widehat{f}(k)$ Then recover $f(x)$

$$f(x) = \int_{K=-\infty}^{\infty} \left[\int_{x=-\infty}^{\infty} f(x) e^{-iKx}\, dx\right] e^{iKx} \frac{dK}{2\pi}. \tag{11}$$

We are free to change the "dummy variable" from K back to k. The integral inside the brackets is (1), producing $\widehat{f}(k)$. The outer integral that reconstructs $f(x)$ is (2).

Derivatives, Integrals, and Shifts: The Key Rules

The transform of df/dx follows a simple rule. For Fourier series, ik multiplies c_k:

$$\text{The derivative of} \quad f(x) = \sum_{-\infty}^{\infty} c_k e^{ikx} \quad \text{leads to} \quad \frac{df}{dx} = \sum_{-\infty}^{\infty} ikc_k e^{ikx}.$$

For Fourier integrals the transform of df/dx is $ik\widehat{f}(k)$:

$$f(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \widehat{f}(k) e^{ikx}\, dk \quad \text{leads to} \quad \frac{df}{dx} = \frac{1}{2\pi}\int_{-\infty}^{\infty} ik\widehat{f}(k) e^{ikx}\, dk.$$

The underlying reason is that e^{ikx} is an eigenfunction of d/dx with eigenvalue ik. Fourier's formulas simply express $f(x)$ as a linear combination of these eigenfunctions:

$$\frac{d}{dx} e^{ikx} = ike^{ikx}. \tag{12}$$

The rule for indefinite integrals is the opposite. Since integration is the inverse of differentiation, we *divide* by ik instead of multiplying. **The transform of the integral is $\widehat{f}(k)/ik$.** There is one exception: $k = 0$ is ruled out. For the integral of $f(x)$ to approach zero as $|x| \to \infty$, we need $\widehat{f}(0) = \int_{-\infty}^{\infty} f(x)\,dx = 0$.

A third operation on $f(x)$ is a *shift of the graph*. For $f(x-d)$, the graph moves a distance d to the right. **The Fourier transform of $f(x-d)$ is $\widehat{f}(k)$ times e^{-ikd}:**

$$\textbf{Shift of } \boldsymbol{f(x)} \qquad \int_{-\infty}^{\infty} e^{-ikx} f(x-d)\,dx = \int_{-\infty}^{\infty} e^{-ik(y+d)} f(y)\,dy = e^{-ikd}\widehat{f}(k)\,. \tag{13}$$

This is especially clear for the delta function $\delta(x)$, which has $\widehat{\delta}(k) = 1$. Moving the impulse to $x = d$ multiplies the transform by e^{-ikd}. And multiplying $f(x)$ by an exponential e^{+ikd} will shift its transform! We summarize the four key rules:

Rule 1	Transform of	$\boldsymbol{df/dx}$	$\boldsymbol{ik\widehat{f}(k)}$	Increase high frequencies
Rule 2	Transform of	$\boldsymbol{\int_{-\infty}^{x} f(x)\,dx}$	$\boldsymbol{\widehat{f}(k)/ik}$	Decrease high frequencies
Rule 3	Transform of	$\boldsymbol{f(x-d)}$	$\boldsymbol{e^{-ikd}\widehat{f}(k)}$	Shift of f changes phase of $\widehat{f}$
Rule 4	Transform of	$\boldsymbol{e^{ixc}f(x)}$	$\boldsymbol{\widehat{f}(k-c)}$	Phase change of f shifts $\widehat{f}$

Example 6 The derivative of the two-sided pulse in Example 4 is an *odd pulse*:

$$\frac{d}{dx}\begin{cases} e^{-ax} & \text{for } x \ge 0 \\ e^{ax} & \text{for } x \le 0 \end{cases} = \begin{cases} -ae^{-ax} & \text{for } x > 0 \\ +ae^{ax} & \text{for } x < 0 \end{cases} = \textbf{odd pulse (times } -a).$$

The transform of this df/dx must be $ik\widehat{f}(k) = 2ika/(a^2+k^2)$. Check by Example 3!

$$\textbf{Transform of odd pulse} \qquad (-a)\left(\frac{1}{a+ik} - \frac{1}{a-ik}\right) = \frac{(-a)(-2ik)}{(a+ik)(a-ik)} = \frac{2ika}{a^2+k^2}\,.$$

The drop of $2a$ in df/dx at $x = 0$ produces this slower $2a/k$ decay in the transform.

Example 7 The box function (square pulse) in Example 2 has $\widehat{f}(k) = (e^{ikL} - e^{-ikL})/ik$. The derivative of the box function is $\delta(x+L) - \delta(x-L)$, with a spike at $x = -L$ where the box jumps to 1, minus a spike at $x = L$ where it jumps back to 0. *Check*: Those spikes transform by **Rule 3** to $e^{ikL} - e^{-ikL}$. This agrees with $ik\widehat{f}(k)$ from **Rule 1**.

The hat function needs *two derivatives* to produce delta functions from ramps:

$$\textbf{Hat Function} \qquad H(x) = \begin{cases} 1+x & \text{for } -1 \le x \le 0 \\ 1-x & \text{for } 0 \le x \le 1 \end{cases} \qquad H'(x) = \begin{cases} 1 \\ -1 \end{cases}$$

The slope $H'(x)$ has jumps of $+1, -2, +1$ and the second derivative has three spikes. By **Rule 3**, the transform of this H'' is $e^{ik} - 2 + e^{-ik} = 2\cos k - 2$. Then use **Rule 2**:

$$\textbf{Transform of hat function} = \frac{\text{Transform of } H''}{(ik)^2} = \frac{2 - 2\cos k}{k^2}\,. \tag{14}$$

Example 8 The **bell-shaped Gaussian** $f(x) = e^{-x^2/2}$ transforms to $\widehat{f}(k) = \sqrt{2\pi}e^{-k^2/2}$

This is a fascinating and important example. The function $f(x)$ is infinitely smooth, and $\widehat{f}(k)$ decreases rapidly. At the same time $f(x)$ decreases rapidly and $\widehat{f}(k)$ is infinitely smooth. To find $\widehat{f}(k)$, use the fact that df/dx = derivative of $e^{-x^2/2} = -xf(x)$:

$$\begin{aligned} ik\widehat{f}(k) &= \int_{-\infty}^{\infty} -xe^{-x^2/2}e^{-ikx}\,dx \quad \text{(transform of } \frac{df}{dx} \text{ by } \textbf{Rule 1)} \\ &= \frac{1}{i}\frac{d}{dk}\int_{-\infty}^{\infty} e^{-x^2/2}e^{-ikx}\,dx = \frac{1}{i}\frac{d}{dk}\widehat{f}(k)\,. \end{aligned}$$

Thus $\widehat{f}(k)$ solves the same equation $d\widehat{f}/dk = -k\widehat{f}(k)$ that $f(x)$ solved! This equation must have the same solution multiplied by some constant: $\widehat{f}(k) = Ce^{-k^2/2}$. The constant $C = \sqrt{2\pi}$ is determined at $k = 0$ by the known integral $\widehat{f}(0) = \int e^{-x^2/2}\,dx = \sqrt{2\pi}$.

This example leads to the most important probability distribution $p(x) = e^{-(x-m)^2/2\sigma^2}$, divided by $\sqrt{2\pi}\sigma$ so $\int p(x)\,dx = \widehat{p}(0)$ = total probability = 1. Shifting the center to the mean value m multiplies $\widehat{p}(k)$ by e^{-ikm} (this is **Rule 3**). Rescaling x to x/σ rescales k to σk (this is Problem 9). *The normal distribution has* $\widehat{p}(k) = e^{-ikm}e^{-\sigma^2k^2/2}$.

When all derivatives of $f(x)$ are smooth, all of their transforms $(ik)^n\widehat{f}(k)$ decay rapidly for large k. Reversing the roles, a rapidly decreasing $f(x)$ corresponds to a smooth $\widehat{f}(k)$. The one-sided pulse e^{-ax} is rapidly decaying but not smooth (at $x = 0$). Its transform $1/(a + ik)$ is smooth but not rapidly decreasing.

The bell-shaped Gaussian $e^{-x^2/2}$ and its transform $\sqrt{2\pi}\,e^{-k^2/2}$ illustrate how both $f(x)$ and $\widehat{f}(k)$ can decrease smoothly and rapidly. Heisenberg's Uncertainty Principle sets a limit; all these Gaussians reach it.

Green's Functions

With these rules for derivatives we can solve differential equations (when they have constant coefficients and no problems from boundaries). Here is an example:

$$\textbf{Equation in } \boldsymbol{x} \qquad -\frac{d^2u}{dx^2} + a^2u = h(x) \quad \text{for } -\infty < x < \infty\,. \tag{15}$$

There are three steps. Step 1 is to take the Fourier transform of each term:

$$\textbf{Equation in } \boldsymbol{k} \qquad -(ik)^2\,\widehat{u}(k) + a^2\,\widehat{u}(k) = \widehat{h}(k) \quad \text{for each } k\,. \tag{16}$$

Step 2 is to find the transform $\widehat{u}(k)$ of the solution (just divide):

$$\textbf{Solution in } \boldsymbol{k} \qquad \widehat{u}(k) = \frac{\widehat{h}(k)}{a^2 + k^2}\,. \tag{17}$$

Step 3 (the hard step) inverts this transform $\widehat{u}(k)$ to construct the solution $u(x)$.

The most important right side is a *delta function*: $h(x) = \delta(x)$. Its transform is $\widehat{\delta}(k) = 1$. Then $\widehat{u}(k) = 1/(a^2 + k^2)$ and we saw this transform in Example 4. The solution with $h(x) = \delta(x)$ is the **Green's function**. I will write $G(x)$ instead of $u(x)$:

Green's function $$G(x) = \frac{1}{2a} e^{-a|x|} = \text{even decaying pulse divided by } 2a. \quad (18)$$

In engineering, $G(x)$ is the ***impulse response*** (the response at x to an impulse at 0). In mathematics, $G(x)$ is the ***fundamental solution*** of the differential equation. The fraction $\widehat{G}(k) = 1/(a^2 + k^2)$ is the ***transfer function*** for each frequency k.

Check. Two derivatives of e^{-ax} (and also e^{ax}) give $-G'' + a^2 G = 0$. So equation (15) is correct away from $x = 0$. At that point the slope $G'(x)$ is $a/2a$ from the left and $-a/2a$ from the right. Therefore $-G''$ is the unit delta function $\delta(x)$, as required.

Convolution with Green's Function

Using this Green's function $G(x)$, we can solve the differential equation for any right side $h(x)$. From (17), we have a multiplication $\widehat{G}(k)\widehat{h}(k)$ in frequency space:

Multiply in frequency domain $$\widehat{u}(k) = \frac{\widehat{h}(k)}{a^2 + k^2} = \widehat{G}(k)\widehat{h}(k). \quad (19)$$

What function has this transform? The answer is not $G(x)h(x)$! The solution $u(x)$ to (15) is not the product but the ***convolution*** of $G(x)$ and $h(x)$. It combines all the responses at x to impulses $h(y)$ at every y, by integrating $G(x - y)h(y)$:

Convolution $G(x) * h(x)$ is the analogue of $\sum G_{j-k} h_k$. The sum becomes an integral:

Solution = Convolution $$u(x) = \int_{y=-\infty}^{\infty} G(x - y)h(y)\, dy = G(x) * h(x). \quad (20)$$

The Fourier transform of $u(x)$ is $\widehat{u}(k) = \widehat{G}(k)\widehat{h}(k)$. This is the ***convolution rule.***

Example 9 Solve equation (15) with $h(x) = \delta(x - d) =$ **point load at** d. The transform is $\widehat{h}(k) = e^{-ikd}$. Then $\widehat{u}(k) = e^{-ikd}/(a^2 + k^2)$. We can find $u(x)$ in three ways:

(1) If $\widehat{u}(k)$ is multiplied by e^{-ikd} then $u(x)$ is shifted by d: $u(x) = G(x - d)$.

(2) Convolution gives $u(x) = G(x) * h(x) = \int G(x - y)\delta(y - d)\, dy = G(x - d)$.

(3) When $h(x)$ is shifted by d, so is the solution! Constant coefficients are shift-invariant.

This is very different from Laplace's equation in a circle. There the Green's function has to change as the impulse moves toward the boundary. Here there is no boundary. The entire problem shifts by d. It is like Laplace's equation in free space, where the Green's function is $1/4\pi r$—and r is the distance from the impulse. **Our problem is shift-invariant.**

A direct proof of the convolution rule $\widehat{u} = \widehat{G}\,\widehat{h}$ starts with the formula for $\widehat{u}(k)$:

$$\widehat{u}(k) = \int_{-\infty}^{\infty} e^{-ikx} u(x)\,dx = \int_{x=-\infty}^{\infty} \int_{y=-\infty}^{\infty} e^{-ik(x-y)} e^{-iky} G(x-y) h(y)\,dy\,dx\,.$$

On the right, move e^{-iky} and $h(y)$ outside the x integral. Then change variables from $x - y$ to z. The two integrals are $\widehat{h}(k)$ and $\widehat{G}(k)$ as desired:

Convolution Rule: Integrals

$$\widehat{u}(k) = \int_{y=-\infty}^{\infty} e^{-iky} h(y)\,dy \int_{z=-\infty}^{\infty} e^{-ikz} G(z)\,dz = \widehat{h}(k)\widehat{G}(k)\,. \quad (21)$$

Example 10 The convolution **Box** $*$ **Box** gives a hat function! The convolution integral in x-space is not fun to do. But multiplication (squaring) in k-space is great. Take $L = \frac{1}{2}$ in Example 2 to get the hat $H(x)$ in Example 7:

$$\widehat{H}(k) = \left(\frac{e^{ik/2} - e^{-ik/2}}{ik}\right)^2 = \frac{e^{ik} - 2 + e^{-ik}}{-k^2} = \frac{2 - 2\cos k}{k^2}\,. \quad (22)$$

Example 11 The convolution of two bell-shaped Gaussian functions $e^{-x^2/2\sigma}$ and $e^{-x^2/2\tau}$ is still bell-shaped. I could have used σ^2 and τ^2, but this way **we just add σ and τ**:

Convolution of Gaussians

$$\frac{1}{\sqrt{2\pi\sigma}} e^{-x^2/2\sigma} * \frac{1}{\sqrt{2\pi\tau}} e^{-x^2/2\tau} = \frac{1}{\sqrt{2\pi(\sigma+\tau)}} e^{-x^2/2(\sigma+\tau)}\,. \quad (23)$$

The convolution integral is computable, but multiplication is a lot easier:

Multiply transforms

$$(e^{-\sigma k^2/2})(e^{-\tau k^2/2}) = e^{-(\sigma+\tau)k^2/2}\,. \quad (24)$$

Transforming back to x-space gives the Gaussian in (23), with $\sigma + \tau$ going into the denominator. The constants in (23) give "total probability = integral = 1."

Example 1 in Section 6.4 describes another proof by solving the heat equation $u_t = u_{xx}$. The solution at time 2σ starting from $u = \delta(x)$ is the first Gaussian. The second Gaussian takes us onward to $T = 2\sigma + 2\tau$. The third Gaussian gets to T in one step. All Gaussians give equality in Heisenberg's inequality.

Example 12 The graph of $e^{-x^2/2\sigma}$ gets *narrower* as σ goes to zero. Divided by $\sqrt{2\pi\sigma}$, it also gets *higher*. The area under the curve (the integral) stays at 1.

The limit as σ approaches zero is the delta function. You might say, what else could it be? With $-x^2/2\sigma$ in the exponent, the pointwise limit as $\sigma \to 0$ is certainly zero (except at $x = 0$). The integral stays at 1, because we divide by $\sqrt{2\pi\sigma}$. So the higher and narrower bells approach an infinite spike at $x = 0$. This is confirmed by the Fourier transforms: $e^{-\sigma k^2/2} \to 1$ as $\sigma \to 0$.

The Energy Equation

The energy in $f(x)$ equals the energy in its Fourier coefficients. In Fourier series, the length of $f(x)$ in the Hilbert function space L^2 equals the length of the vector c in the Hilbert vector space ℓ^2. The equation was Parseval's:

Energy in Fourier series
$$\int_{-\pi}^{\pi} |f(x)|^2\,dx = 2\pi \sum_{-\infty}^{\infty} |c_k|^2\,. \tag{25}$$

That was proved in Section 4.1 by multiplying $(\sum c_k e^{ikx})$ $(\sum \bar{c}_k e^{-ikx})$ and integrating. Now we state a similar energy equation for the Fourier *integral* pair $f(x)$ and $\widehat{f}(k)$.

Energy in Fourier integrals
$$\int_{-\infty}^{\infty} |f(x)|^2\,dx = \frac{1}{2\pi}\int_{-\infty}^{\infty} |\widehat{f}(k)|^2\,dk\,. \tag{26}$$

In the same way, inner products of $f(x)$ and $g(x)$ transform to inner products:

Inner products
$$\int_{-\infty}^{\infty} f(x)\overline{g(x)}\,dx = \frac{1}{2\pi}\int_{-\infty}^{\infty} \widehat{f}(k)\overline{\widehat{g}(k)}\,dk\,. \tag{27}$$

Example 13 The one-sided decaying pulse $f(x) = e^{-x}$ on $0 \le x < \infty$ has energy $\frac{1}{2}$:

$$\int_{-\infty}^{\infty} |f(x)|^2\,dx = \int_0^{\infty} e^{-2x}\,dx = \frac{1}{2}\,.$$

Its transform $\widehat{f}(k)$ has the same energy, after we multiply by 2π:

$$\int_{-\infty}^{\infty} \left|\frac{1}{1+ik}\right|^2 dk = \int_{-\infty}^{\infty} \frac{dk}{1+k^2} = \left[\tan^{-1} k\right]_{-\infty}^{\infty} = \pi\,.$$

Rescaling The factor 2π in the transform can be moved. If $\widehat{f}(k)$ is divided by $\sqrt{2\pi}$, call this new transform $F(k)$. After squaring, 2π disappears in the energy equation. This "symmetrized" transform is like an orthogonal matrix with $Q^{\mathrm{T}}Q = I$:

Energy in F = energy in f
$$F^{\mathrm{T}}F = f^{\mathrm{T}}Q^{\mathrm{T}}Qf = f^{\mathrm{T}}f\,.$$

More correctly, the Fourier transform preserves the length of every *complex* vector:

$$\overline{F}^{\mathrm{T}}F = \overline{f}^{\mathrm{T}}f \text{ corresponds to } \int |F(k)|^2\,dk = \int |f(x)|^2\,dx\,.$$

This complex symmetrized Fourier transform is unitary, as in $\overline{Q}^{\mathrm{T}}Q = I$:

$$F(k) = Qf = \frac{1}{\sqrt{2\pi}}\int e^{-ikx} f(x)\,dx \quad \text{and} \quad f(x) = \overline{Q}^{\mathrm{T}}F = \frac{1}{\sqrt{2\pi}}\int e^{ikx} F(k)\,dk\,. \tag{28}$$

Heisenberg's Uncertainty Principle

Heisenberg dealt with position and momentum in quantum mechanics. As one is measured more exactly, the other becomes less exact. There is a similar "product of uncertainty" for phase and amplitude of oscillations—and also time and energy.

Here the uncertainty principle involves $f(x)$ and $\widehat{f}(k)$. *If one is concentrated in a narrow band, the other fills a wide band.* An impulse $\delta(x)$ of zero width has a transform $\widehat{\delta}(k) = 1$ of infinite width. Probability suggests that the square root σ of the variance (normalized by the energy in f) is the right measure of width:

Widths σ_x and σ_k
$$\sigma_x^2 = \frac{\int x^2 (f(x))^2\,dx}{\int (f(x))^2\,dx} \qquad \sigma_k^2 = \frac{\int k^2 |\widehat{f}(k)|^2\,dk}{\int |\widehat{f}(k)|^2\,dk}.$$

All integrals go from $-\infty$ to ∞, and the uncertainty principle is quick to state.

Heisenberg's Uncertainty Principle *Every function has* $\sigma_x \sigma_k \geq \frac{1}{2}$.

The cosine of the angle between $xf(x)$ and $f'(x)$ is at most one, even in Hilbert space. The Schwarz inequality $|a^{\mathrm{T}}b|^2 \leq (a^{\mathrm{T}}a)(b^{\mathrm{T}}b)$ becomes

$$\left| \int x f(x) f'(x)\,dx \right|^2 \leq \left(\int (xf(x))^2\,dx \right) \left(\int (f'(x))^2\,dx \right). \tag{29}$$

Since $f(x)f'(x)$ is the derivative of $\frac{1}{2}(f(x))^2$, integrate the left side by parts:

$$\int x f(x) f'(x)\,dx = \left[x \frac{(f(x))^2}{2} \right]_{-\infty}^{\infty} - \int \frac{(f(x))^2}{2}\,dx. \tag{30}$$

The integrated term is zero at $\pm\infty$ whenever the bandwidths are finite.

Plancherel's energy equation allows us to switch $\int (f(x))^2\,dx$ and $\int (f'(x))^2\,dx$ to $\int |\widehat{f}(k)|^2\,dk$ and $\int |k\widehat{f}(k)|^2\,dk$. The factors 2π cancel when we combine (29) and (30):

$$\left(\int \frac{(f(x))^2}{2}\,dx \right) \left(\int \frac{|\widehat{f}(k)|^2}{2}\,dk \right) \leq \left(\int (xf(x))^2\,dx \right) \left(\int |k\widehat{f}(k)|^2\,dk \right). \tag{31}$$

Taking square roots, this is the uncertainty principle $\sigma_x \sigma_k \geq \frac{1}{2}$.

Second proof Quantum mechanics associates position with multiplication $xf(x)$. Momentum corresponds to differentiation df/dx (in other words with $ik\widehat{f}(k)$). These operations $Bf = xf$ and $Af = df/dx$ do not commute:

$$\frac{d}{dx}(xf(x)) - x\frac{d}{dx}f(x) = f(x) \quad \text{means that} \quad \boldsymbol{AB - BA = I}.$$

The uncertainty principle for $\|Af\|$ times $\|Bf\|$ is again the Schwarz inequality:

Heisenberg inequality
$$\|f\|^2 = |f^{\mathrm{T}}(AB - BA)f| \leq 2\,\|Af\|\,\|Bf\|. \tag{32}$$

Autocorrelation and Power Spectral Density

The autocorrelation of a vector is $f(n) * \overline{f(-n)}$. The autocorrelation of a function is $f(t) * \overline{f(-t)}$. We are using t instead of x, because the most important applications are to communications and electronics and power.

By the Convolution Rule, the transform of that convolution is $\widehat{f}(k)$ times its complex conjugate. That product is $|\widehat{f}(k)|^2$, the **power spectral density** of $f(t)$.

Autocorrelation $R(t)$
Power Spectral Density

$$R(t) = \int_{-\infty}^{\infty} f(s)\overline{f(t-s)}\,ds \qquad G(k) = \widehat{R}(k) = |\widehat{f}(k)|^2. \tag{33}$$

One advantage is $G \geq 0$. The key advantage is the energy identity (now for *power*):

$$\textbf{Power} = \int_{-\infty}^{\infty} |f(t)|^2\,dt = \frac{1}{2\pi}\int_{-\infty}^{\infty} |\widehat{f}(k)|^2\,dk = \frac{1}{2\pi}\int_{-\infty}^{\infty} G(k)\,dk. \tag{34}$$

$G(k)$ is the density of power at frequency k in the spectrum. Hence the name PSD.

Real signals come with noise. A sinusoidal waveform is never perfect, it shows rapid, random, usually small perturbations. The signal-to-noise ratio measures their importance. Since the noise is a random variable, we find its expected power from its probability distribution:

White noise has $G(k) = \text{constant}$ **1/f noise** has $G(k) = \text{constant}/k^{\alpha}$ (35)

The independent jumps of many electrons approach white noise (thermal noise). There is no correlation between jumps so the autocorrelation R is a delta function and G is a constant function. But $1/f$ noise is everywhere too: economic data, traffic flow, flicker noise in metals and semiconductors. *Notice*: $R = \text{constant}$ and $R = 1/k$ have infinite integrals. It is the **average power** that stays finite for time-invariant (stationary) noise distributions.

The fundamental nonstationary process is a **random walk**.

Example 14 A random walk $x(t)$ can jump by 1 or -1 at each time step Δt.

This random walk is like counting the difference "heads minus tails" in a sequence of coin flips. This is carefully studied in probability [51]. The limit as $\Delta t \to 0$ is a Wiener process or **Brownian motion**, which we meet in Section 6.5 as a model for stock prices.

The jump distribution can be binomial (± 1) or uniform or Gaussian or other. The *independence* of successive jumps is the key: *we can add spectral densities*. Each jump contributes a step function to $x(t)$, and its Fourier transform (from jump time to final time) is a sinc function. The sum of squares of those sinc functions gives $G(k) \approx 1/k^2$ for $k\,\Delta t >> 1$. So these random jumps are $1/f$ noise.

Periodic Components over Infinite Time

By separating Fourier series (periodic in time) from Fourier integrals (infinite time), the two transforms c_k and $\widehat{f}(k)$ are clear. But in reality, $f(t)$ could have *periodic components over infinite time.* The simplest example $f(t) = \cos \omega t = (e^{i\omega t} + e^{-i\omega t})/2$ has two inconvenient difficulties for Fourier analysis:

1 $f(t) = \cos \omega t$ does not approach zero **2** $\widehat{f}(k)$ has delta functions at $k = \pm\omega$

The power P and autocorrelation R and power spectral density G all have problems over infinite time. We need to work with *average power* over $0 \le t \le T$. On a finite interval with finite power, Parseval's identity connects $f(t)$ to its transform $\widehat{f}(T,k)$:

Average Power
$$\overline{P}(T) = \frac{1}{T}\int_0^T |f(t)|^2\,dt = \int_{-\infty}^{\infty} \frac{|\widehat{f}(T,k)|^2}{T}\,dk = \int_{-\infty}^{\infty} G(T,k)\,dk. \quad (36)$$

That identity indicates our plan: *Let* $T \to \infty$. A sharp eye might catch the difficulty: *The k-integrals also have* $k \to \infty$. Interchanging two infinite limits is not safe.

A similar problem was hidden (we didn't mention it) in the inversion formula from $\widehat{f}(k)$ to $f(x)$. The Fourier series over longer and longer intervals has coefficients c_k in (9), now called $\widehat{f}(T,k)$. In the energy identity (26) for Fourier integrals, $\int |\widehat{f}(k)|^2\,dk$ also has an infinite integral for $\widehat{f}(k)$ inside that infinite integral over k.

Exchanging limits is justified for the nicest functions $f(t)$ and $\widehat{f}(k)$, smooth and decaying. Then definitions are extended, as we know from $\int \delta(x)\,dx = 1$. Here the extension is to $\widehat{R} = G$, transform of autocorrelation equals power spectral density. Start with the identity (36) for average power, which is a useful measure in itself; $G(T,k)$ is a *periodogram.* Then work with the integral of G, always safer than G:

Wiener-Khintchine
$$F(k) = \lim_{T\to\infty} \int_{-\infty}^{k} G(T,\omega)\,d\omega \text{ is the transform of } R(t) = \int_{-\infty}^{\infty} e^{ikt}\,dF(k). \quad (37)$$

This "Stieltjes integral" allows steps in F, as in $\int \delta(x)\,dx = \int dF = 1$.

Summary of Fourier Integrals

1. Transform and inverse transform (1)-(2)

2. Transforms of $\delta(x)$, square pulse, decaying pulse, and Gaussian

3. Rules for derivatives, integrals, and shifts

4. Constant-coefficient equations solved by convolution (20)

5. Energy identity (26) for $f(x)$ and $\widehat{f}(k)$. Application to autocorrelation and $|\widehat{f}(k)|$

Problem Set 4.5

1 Find the transform $\widehat{g}(k)$ of the odd two-sided pulse $g(x)$:

$$g(x) = -e^{ax} \quad \text{for } x < 0\,, \quad g(x) = e^{-ax} \quad \text{for } x > 0\,.$$

The decay rate of $\widehat{g}(k)$ is ____. There is a ____ in $g(x)$.

2 Find the Fourier transforms (with $f(x) = 0$ outside the ranges given) of

(a) $f(x) = 1$ for $0 < x < L$

(b) $f(x) = 1$ for $x > 0$ and $f(x) = -1$ for $x < 0$ (set $a = 0$ in Problem 1)

(c) $f(x) = \int_0^1 e^{ikx} dk$ (no calculation needed to recognize $\widehat{f}(k)$)

(d) the double sine wave $f(x) = \sin x$ for $0 \le x \le 4\pi$

3 Find the inverse transforms of

(a) $\widehat{f}(k) = \delta(k)$

(b) $\widehat{f}(k) = e^{-|k|}$ (please separate $k < 0$ from $k > 0$).

4 Apply Plancherel's formula $2\pi \int |f(x)|^2\, dx = \int |\widehat{f}(k)|^2\, dk$ to

(1) the square pulse $f(x) = 1$ for $-1 < x < 1$, to find $\displaystyle\int_{-\infty}^{\infty} \frac{\sin^2 t}{t^2}\, dt$

(2) the even decaying pulse, to find $\displaystyle\int_{-\infty}^{\infty} \frac{dt}{(a^2 + t^2)^2}$.

Problems 5-9 involve $f(x) = e^{-x^2/2}$. Its transform is $\widehat{f}(k) = \sqrt{2\pi} e^{-k^2/2}$, by Example 8 and also by Cauchy's theorem on complex integration (x to $x + ik$):

$$\widehat{f}(k) = \int_{-\infty}^{\infty} e^{-x^2/2} e^{-ikx}\, dx = e^{-k^2/2} \int_{-\infty}^{\infty} e^{-(x+ik)^2/2}\, dx = \sqrt{2\pi}\, e^{-k^2/2}\,.$$

5 Verify Plancherel's energy equation for $\delta(x)$ and $e^{-x^2/2}$. Infinite energy allowed.

6 What are the half-widths σ_x and σ_k of the bell-shaped function $f(x) = e^{-x^2/2}$ and its transform? Show that equality holds in the uncertainty principle.

7 What is the transform of $xe^{-x^2/2}$ by the derivative rule? What about $x^2 e^{-x^2/2}$?

8 Suppose g is a stretched version of f, $g(x) = f(ax)$. Show that $\widehat{g}(k) = a^{-1}\widehat{f}(k/a)$. Illustrate with the even pulse $f(x) = e^{-|x|}$.

9 Use the previous exercise to find the transform of $g(x) = e^{-a^2x^2/2}$. Then show that $e^{-x^2/2} * e^{-x^2/2} = \sqrt{\pi}\, e^{-x^2/4}$, transforming the left side by the convolution rule (20) and the right side by the choice $a^2 = \frac{1}{2}$.

10 The decaying pulse $f(x) = e^{-ax}$ has derivative $df/dx = -ae^{-ax}$ (and 0 for $x < 0$). Why isn't the transform of df/dx just $-a\widehat{f}(k)$ instead of $ik\widehat{f}(k)$? What am I missing in thinking that df/dx is equal to $-af(x)$?

11 Find $\widehat{u}(k)$ for a point load at d by taking Fourier transforms: $u' + au = \delta(x-d)$. By inverse transform (or direct solution) find the Green's function $u(x) = G(x,d)$.

12 Take Fourier transforms of this unusual equation to find $\widehat{u}(k)$ and then $u(x)$:

$$(\text{integral of } u(x)) - (\text{derivative of } u(x)) = \delta(x)\,.$$

13 The convolution $f(x) * f(-x)$ of a decaying pulse (Ex. 3) and ascending pulse is an autocorrelation:

$$C(x) = \int_{-\infty}^{\infty} f(x-y)f(-y)\,dy \text{ with transform } \widehat{C}(k) = \frac{1}{a+ik}\frac{1}{a-ik} = \frac{1}{a^2+k^2}\,.$$

Find $C(x)$ from this transform, and also by computing the integral.

14 The hat function Box $*$ Box has transform $2(1-\cos k)/k^2$ in Examples 7 and 10. Use the convolution rule for $S(x) =$ Hat $*$ Hat to find $\widehat{S}(k)$. Show from $(ik)^4\widehat{S}(k)$ that the fourth derivative of $S(x)$ is a combination of spikes at $x = -2, -1, 0, 1, 2$. Since this fourth derivative is zero at all other points, $S(x) =$ Hat $*$ Hat $=$ Box $*$ Box $*$ Box $*$ Box is *piecewise cubic*, with 5 jumps in its third derivative. $S(x)$ is the famous ***cubic B-spline*** for $-2 \le x \le 2$.

15 Show that the Fourier transform of $g(x)h(x)$ is the convolution $\widehat{g}(k) * \widehat{h}(k)/2\pi$ by repeating the proof of the convolution rule—but with e^{+ikx} to produce the inverse transform.

16 The derivative $\delta'(x)$ of the delta function is the *doublet*. It is a "distribution" concentrated at $x = 0$. Integration by parts picks out not $f(0)$ but $-f'(0)$:

$$\int f(x)\delta'(x)\,dx = -\int f'(x)\delta(x)\,dx = -f'(0).$$

(a) Why should the Fourier transform of the doublet $\delta'(x)$ be ik?

(b) What does the inverse formula (2) give for $\int ke^{ikx}\,dk$?

(c) Exchanging k and x, what is the Fourier transform of $f(x) = x$?

17 Suppose g is the mirror image of f, $g(x) = f(-x)$. Show from (1) that $\widehat{g}(k) = \widehat{f}(-k)$. If $f(x)$ is real, show that $\widehat{f}(-k)$ is the conjugate of $\widehat{f}(k)$.

18 If $f(x)$ is an even function, the integrals for $x > 0$ and $x < 0$ combine into

$$\widehat{f}(k) = \int_{-\infty}^{\infty} f(x)e^{-ikx}\,dx = 2\int_{0}^{\infty} f(x)\cos kx\,dx$$
$$f(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \widehat{f}(k)e^{ikx}\,dk = \frac{1}{\pi}\int_{0}^{\infty} \widehat{f}(k)\cos kx\,dk$$

Find $\widehat{f}(k)$ in this way for the even decaying pulse $e^{-a|x|}$. What are the corresponding formulas for sine transforms when $f(x)$ is odd?

19 If $f(x)$ is a line of equally spaced delta functions explain why $\widehat{f}(k)$ is too:

$$\text{The transform of } f(x) = \sum_{n=-\infty}^{\infty} \delta(x - 2\pi n) \text{ is } \widehat{f}(k) = \sum_{n=-\infty}^{\infty} \delta(k - n)\,.$$

20 (a) Why is $F(x) = \sum_{n=-\infty}^{\infty} f(x + 2\pi n)$ a 2π-periodic function?

(b) Show that its Fourier coefficient $c_k = \frac{1}{2\pi}\int_{-\pi}^{\pi} F(x)e^{-ikx}\,dx$ equals $\widehat{f}(k)/2\pi$.

(c) From $F(x) = \sum c_k e^{ikx}$ at $x = 0$ find ***Poisson's summation formula***:

$$\sum_{n=-\infty}^{\infty} f(2\pi n) = \frac{1}{2\pi}\sum_{k=-\infty}^{\infty} \widehat{f}(k)\,.$$

21 $u(x) = 1$ is an eigenfunction for convolution with any $g(x)$. Find the eigenvalue.

22 Take Fourier transforms in $G''''(x) - 2G''(x) + G(x) = \delta(x)$ to find the transform $\widehat{G}(k)$ of the Green's function. How would it be possible to find $G(x)$?

23 What is $\delta * \delta$?

24 What is $\widehat{f}(k)$ if $f(x) = e^{5x}$ for $x \le 0$, $f(x) = e^{-3x}$ for $x \ge 0$? Find the function $f(x)$ whose Fourier transform is $\widehat{f}(k) = e^{-|k|}$.

25 Propose a two-dimensional Fourier transform, from $f(x, y)$ to $\widehat{f}(k_1, k_2)$. Given $\widehat{f}(k_1, k_2)$, what integral like (2) will invert the transform and recover $f(x, y)$?

26 Find the 2D Fourier transform $\widehat{f}(k_1, k_2)$ of $e^{-(x^2+y^2)/2}$.

Challenge: Find the 2D transform of $e^{-Q/2}$ by diagonalizing the matrix in $Q = ax^2 + 2bxy + cy^2$.

4.6 DECONVOLUTION AND INTEGRAL EQUATIONS

In explaining $f * g$ and $c \circledast d$, we were given the inputs. From functions f and g, or from vectors c and d, we found the convolution. **Deconvolution goes backward.** The unknown function $U(x)$ or the unknown vector u is *inside* the convolution (non-cyclic or cyclic). Let me reveal the key idea before the important examples.

We are now given the output $B(x) = G(x) * U(x)$ or $b = c \circledast u$. We know the kernel function $G(x)$ or kernel vector c. The problem is to solve for $U(x)$ or u.

The equation $G(x) * U(x) = B(x)$ looks complicated in x-space (convolution produces an integral equation). In the frequency domain, that convolution becomes a multiplication. And the inverse of multiplication is division:

$$\boldsymbol{G * U = B} \quad \text{becomes} \quad \boldsymbol{\widehat{G}\,\widehat{U} = \widehat{B}} \quad \text{which gives} \quad \boldsymbol{\widehat{U} = \widehat{B}/\widehat{G}}\,. \tag{1}$$

The last step is to transform $\widehat{U}$ back to x-space, to find the solution $U(x)$.

May I say that this is the same three-step solution that all transform methods use? With the Fourier transform, the basis functions are e^{ikx}:

1. Expand the given $B(x)$ as a combination of eigenfunctions e^{ikx} times $\widehat{B}(k)$.
2. Divide each $\widehat{B}(k)$ by the known eigenvalue $\widehat{G}(k)$.
3. Reconstruct $U(x)$ from its Fourier transform $\widehat{U} = \widehat{B}/\widehat{G}$.

The convolution $G * U$ has eigenfunctions e^{ikx} and eigenvalues $\widehat{G}(k)$. This section is solving the simplest and most beautiful linear equations of applied mathematics: shift-invariant, time-invariant, constant-coefficient (those are equivalent here).

Point-Spread Functions

Along with examples of the convolution rule, I have to tell you about applications. You see convolutions (literally) in a telescope. *A star looks blurred.* The true signal (the star) is practically a point source $\delta(x, y)$ at $(0, 0)$. The blur is the **point-spread function $\boldsymbol{G(x, y)}$**. That is the response at (x, y) to a delta function input at $(0, 0)$.

If the point source is moved to (t, s), then the blurred output $G(x - t, y - s)$ moves with it. This is *shift invariance*, extremely important. If the input is an integral that combines point sources of strength $U(t, s)$, then the output is an integral that combines blurred points $G(x - t, y - s)$ multiplied by U:

$$U(t,s) = \begin{matrix}\textbf{light density of}\\ \textbf{input at } \boldsymbol{(t,s)}\end{matrix} \quad \iint U(t,s)\, G(x-t, y-s)\, dt\, ds = \begin{matrix}\textbf{light density of}\\ \textbf{output at } \boldsymbol{(x,y)}\end{matrix} \tag{2}$$

The telescope has convolved the input U and its built-in point-spread function, to produce the output $G * U$. We need **deconvolution** to find the input U.

All sorts of imaging instruments present this same problem: Find the input from its convolution with G. Solving this problem is crucial for computed tomography (CT scanners won the Nobel Prize for Medicine in 1979). The company that makes the scanner measures its point-spread function G, once and for all. The same problem appears in magnetic resonance imaging (MRI) and in sensors carried on satellites.

Notice that a perfect convolution requires *shift-invariance and linearity*. Usually there are imperfections, especially near the edge of the field of vision. The telescope example involves two dimensions and Fourier integrals. Start in one dimension.

Example 1 Suppose a point source $\delta(x)$ spreads into a hat function $G(x) = 1 - |x|$ with area 1. Why is there a difficulty to recover an unknown distributed source $U(x)$ from the output $B = G * U$?

Solution Deconvolution in frequency space divides $\widehat{B}(k)$ by $\widehat{G}(k)$. This is only safe when $\widehat{G}(k)$ is never zero. A nonzero transform is the test for an invertible convolution.

The transform of the hat function $G(x)$ was computed in the previous section. The second derivative of the hat is $G'' = \delta(x+1) - 2\,\delta(x) + \delta(x-1)$, so we divide its transform $e^{ikx} - 2 + e^{-ikx}$ by $(ik)^2$:

Transform of the hat function
$$\widehat{G}(k) = \frac{2 - 2\cos k}{k^2}. \tag{3}$$

The difficulty is that $\widehat{G}(k) = 0$ when k is a nonzero multiple of 2π. (At $k = 0$ we have $\widehat{G}(0) = 1 =$ area under the hat.) If we divide by zero in $\widehat{U} = \widehat{B}/\widehat{G}$, we normally get an unacceptable transform $\widehat{U}$. This signals that our convolution equation $G * U = B$ is **ill-posed**. This often happens for integral equations:

Integral equation of the first kind
$$G * U = \int_{-\infty}^{\infty} G(x-t)\,U(t)\,dt = B(x) \tag{4}$$

If $U(k) = e^{ikx}$, the integral produces $\widehat{G}(k)e^{ikx}$. Thus $\widehat{G}(k)$ is an eigenvalue of convolution by G. Invertibility always requires nonzero eigenvalues.

I will mention a modification that makes the integral equation **well-posed**, when we start from $\widehat{G}(k) \geq 0$. Add any positive multiple of $U(x)$ to the left side:

Integral equation of the second kind
$$\alpha\,U(x) + \int_{-\infty}^{\infty} G(x-t)\,U(t)\,dt = B(x). \tag{5}$$

Now the transform is $\alpha + \widehat{G}(k)$, never zero. The solution U comes safely from the division $\widehat{B}/(\alpha + \widehat{G})$, followed by an inverse transform. This is like adding αI to a positive semidefinite circulant matrix C, to make it positive definite.

In a telescope, invertibility might come from a different point-spread function G (not a hat). Or the problem may truly be singular. It is impossible to recover complete information about the body when a scanner only looks in N directions. It integrates your density along rays in each direction, and some shapes are invisible (like Stealth aircraft). Spiral CT gives a more complete picture.

Note In **blind deconvolution**, G is not known. The equation $G * U = B$ may change to minimization of $\|G * U - B\|^2 + \alpha\|u\|_{\mathrm{TV}}$. Section 4.7 explains that total variation (TV) term and 8.2 returns to ill-posed equations. **Inverse problems** try to recover the differential equation from its solutions. Or they solve $Au = b$ when $A^{\mathrm{T}}A$ is singular. The adjustment by α is a stabilizing **penalty term** to produce $A^{\mathrm{T}}A + \alpha I$.

Integral equations are not necessarily in a shift-invariant convolution form:

Integral equation: 1st kind [2nd kind]
$$[\alpha\, U(x)] + \int G(x,t)\, U(t)\, dt = B(x). \tag{6}$$

In convolutions, $G(x,t)$ depends only on the difference $x-t$ (as for Toeplitz matrices G_{i-j} with constant diagonals). The sum of $x-t$ and t on the left side is x on the right side—the reliable indicator of a convolution. For kernels like $G = xt$, *no convolution.*

Deconvolution by Matrices

Example 2 (Discrete deconvolution) For C = **circulant matrix**, solve $Cu = b$.

This example immediately makes a key point. *Multiplying by the matrix C is the same as cyclic convolution with its zeroth column c.* For the second-difference circulant matrix C from Section 1.1, we can write the four equations as $Cu = b$ or $c \circledast u = b$:

Circulant Cu = convolution $c \circledast u$
$(2,-1,0,-1) \circledast (u_0,u_1,u_2,u_3)$
$$Cu = \begin{bmatrix} 2 & -1 & 0 & -1 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ -1 & 0 & -1 & 2 \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ u_3 \end{bmatrix} \tag{7}$$

This special matrix C is singular. The all-ones vector $(1,1,1,1)$ is in its nullspace, with zero eigenvalue. C^{-1} does not exist, because the eigenvalues of C are 0, 2, 4, 2.

It will be extremely valuable to see how deconvolution fails in this example. Dividing $\widehat{b}$ by $\widehat{c}$ (component by component) is impossible because one component of $\widehat{c}$ is zero. That vector $\widehat{c} = (0,2,4,2)$ contains the eigenvalues of C:

Discrete transform of c
$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & i^2 & i^3 \\ 1 & i^2 & i^4 & i^6 \\ 1 & i^3 & i^6 & i^9 \end{bmatrix} \begin{bmatrix} 2 \\ -1 \\ 0 \\ -1 \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 2 \\ 4 \\ 2 \end{bmatrix} = \widehat{c}. \tag{8}$$

The eigenvectors of C are the columns of the Fourier matrix! The first eigenvalue is zero, and its eigenvector is the column $(1,1,1,1)$. The four eigenvalues add to 8, which is the correct trace of C (sum of four 2's on the diagonal).

While those matrices are in front of us, let me verify $\boldsymbol{CF = F\Lambda}$:

Eigenvectors v, w, y, z are columns of F
$$\begin{bmatrix} 2 & -1 & 0 & -1 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ -1 & 0 & -1 & 2 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & i^2 & i^3 \\ 1 & i^2 & i^4 & i^6 \\ 1 & i^3 & i^6 & i^9 \\ \boldsymbol{v} & \boldsymbol{w} & \boldsymbol{y} & \boldsymbol{z} \end{bmatrix} = \begin{bmatrix} 0\boldsymbol{v} & 2\boldsymbol{w} & 4\boldsymbol{y} & 2\boldsymbol{z} \\ \boldsymbol{c} \circledast \boldsymbol{v} & \cdots & & \boldsymbol{c} \circledast \boldsymbol{z} \end{bmatrix}. \tag{9}$$

The Fourier matrix F is the eigenvector matrix for every circulant matrix.

Example 3 Adding I, the circulant matrix $C+I$ is invertible. Deconvolution succeeds for $c=(3,-1,0,-1)$. The eigenvalues are increased by 1 to $1,3,5,3$:

$$(\boldsymbol{C+I})\,\boldsymbol{u}=\boldsymbol{b} \qquad \begin{bmatrix} 3 & -1 & 0 & -1 \\ -1 & 3 & -1 & 0 \\ 0 & -1 & 3 & -1 \\ -1 & 0 & -1 & 3 \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} 4 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{10}$$

The four columns v,w,y,z are still eigenvectors of $C+I$. The right side $b=(4,0,0,0)$ is the sum $v+w+y+z$ of all four eigenvectors. This just says that the discrete transform is $\widehat{b}=(1,1,1,1)$. Deconvolution divides by the eigenvalues of C to construct the solution u:

$$u=\frac{1}{\mathbf{1}}v+\frac{1}{\mathbf{3}}w+\frac{1}{\mathbf{5}}y+\frac{1}{\mathbf{3}}z=\frac{1}{15}(18,12,8,12). \quad \text{This is} \quad \boldsymbol{u=F\Lambda^{-1}F^{-1}b}. \tag{11}$$

Every circulant matrix has the form $C=F\Lambda F^{-1}$. The eigenvalues in Λ come from $\widehat{c}$. The Fourier eigenvectors in F show the three steps of $u=C^{-1}b=F\Lambda^{-1}F^{-1}b$:

$$\boldsymbol{F^{-1}b \text{ finds } \widehat{b}} \qquad \boldsymbol{\Lambda^{-1} \text{ gives } \widehat{u}=\widehat{b}/\widehat{c}} \qquad \boldsymbol{F\widehat{u} \text{ reconstructs } u}$$

Thus deconvolution solves $Cu=c\circledast u=b$ with FFT speed:

$$\textbf{bhat = fft(b)}; \quad \textbf{chat = fft(c)}; \quad \textbf{uhat = bhat./chat}; \quad \textbf{u = ifft(uhat)}. \tag{12}$$

Deconvolution for Infinite Matrices

Circulant matrices have periodic boundary conditions to give cyclic convolution $c\circledast u$. Infinite Toeplitz matrices give non-cyclic convolution $C_\infty u=c*u$. Then the job of non-cyclic deconvolution is to solve $c*u=b$.

In the language of signal processing, we are inverting a filter. Its impulse response is $c*\delta=c$. The inverse of an infinite Toeplitz matrix (constant diagonals, time-invariant) will be another Toeplitz matrix. But there is a big difference: If C_∞ is a **banded matrix** from an **FIR filter** (finite impulse response c), then C_∞^{-1} is a **full matrix** from an **IIR filter** (infinite impulse response).

If $C(\omega)=\sum c_k e^{ik\omega}$ is a polynomial, $1/C(\omega)$ is not a polynomial. The only exception is a useless one-coefficient filter. So deconvolution (discrete case or continuous case) does not preserve a band structure.

Example 4 The second difference matrix K_∞ is only semidefinite. Change to $C_\infty=2K_\infty+I$, whose coefficients $-2,5,-2$ are the autocorrelation $(-1,2,0)*(0,2,-1)$:

$$C_\infty=\begin{bmatrix} \cdot & \cdot & & \\ -2 & 5 & -2 & \\ & -2 & 5 & -2 \\ & & \cdot & \cdot \end{bmatrix} = \begin{bmatrix} \cdot & & & \\ \cdot & 2 & & \\ & -1 & 2 & \\ & & -1 & \cdot \end{bmatrix} \begin{bmatrix} \cdot & & & \\ & 2 & -1 & \\ & & 2 & -1 \\ & & & \cdot \end{bmatrix} = L_\infty U_\infty. \tag{13}$$

A tridiagonal C_∞ has bidiagonal factors. Look at matrices or polynomials:

$$\boldsymbol{C(\omega) = L(\omega)U(\omega)} \quad -2e^{i\omega} + 5 - 2e^{-i\omega} = (2 - e^{i\omega})(2 - e^{-i\omega}) = |2 - e^{i\omega}|^2. \tag{14}$$

Positive definiteness of the matrix C_∞ is positivity of the polynomial $C_\infty(\omega)$. Then this 3-term frequency response has a *spectral factorization* (14) into $|A(\omega)|^2$. But the inverse matrix is full!

$$C_\infty^{-1} = U_\infty^{-1} L_\infty^{-1} = \begin{bmatrix} \cdot & \frac{1}{4} & \frac{1}{8} & \frac{1}{16} \\ & \frac{1}{2} & \frac{1}{4} & \frac{1}{8} \\ & & \frac{1}{2} & \frac{1}{4} \\ & & & \cdot \end{bmatrix} \begin{bmatrix} \cdot & & & \\ \frac{1}{4} & \frac{1}{2} & & \\ \frac{1}{8} & \frac{1}{4} & \frac{1}{2} & \\ \frac{1}{16} & \frac{1}{8} & \frac{1}{4} & \cdot \end{bmatrix} \tag{15}$$

Those triangular inverses come from $1/(2 - e^{i\omega}) = \frac{1}{2} + \frac{1}{4}e^{i\omega} + \frac{1}{8}e^{2i\omega} + \cdots$ and they are not polynomials. Their product $1/C(\omega)$ is not a polynomial. *But all these matrices are still convolutions.*

Convolution	$c * u = b$	**Divide by $C(\omega)$**	$D(\omega) = 1/C(\omega)$
Toeplitz matrix	$C_\infty u = b$	**Toeplitz inverse**	$D_\infty = (C_\infty)^{-1}$
Frequency space	$C(\omega)U(\omega) = B(\omega)$	**Deconvolution**	$U(\omega) = B(\omega)/C(\omega)$

(16)

Triangular Matrices and Causal Filters

The Toeplitz matrix L_∞ is **lower triangular** when the filter $\ell = (\ell_0, \ell_1, \ldots)$ is **causal**. The past affects the future, but the future has no effect on the past. There is a time arrow, and cause comes before effect.

An upper triangular matrix U_∞ is anticausal. A banded Toeplitz matrix is a product $L_\infty U_\infty$, found by factoring the polynomials. ***The inverse will exist if $C(\omega) \neq 0$ for all ω***. But here is a disturbing point. The lower triangular inverse from $1/L(\omega)$ might be an **unbounded matrix**:

$$\frac{1}{L(\omega)} = \frac{1}{1 - 3e^{-i\omega}} \quad L_\infty = \begin{bmatrix} \cdot & & & \\ -3 & 1 & & \\ & -3 & 1 & \\ & & -3 & 1 \end{bmatrix} \quad L_\infty^{-1} = \begin{bmatrix} \cdot & & & \\ 3 & 1 & & \\ 9 & 3 & 1 & \\ 27 & 9 & 3 & 1 \end{bmatrix} \tag{17}$$

For triangular matrices, causal or anticausal, there is a stronger condition when the inverse matrix is required to be bounded and still triangular. These are *one-sided* problems, changing from $-\infty < x < \infty$ to $0 \leq t < \infty$. *The Laplace transform replaces the Fourier transform.*

A bounded lower triangular inverse of L_∞ still depends on the zeros of $L(\omega)$. But now the test forbids $3 - e^{-i\omega} = 0$ or $z = 1/3$. $L(z)$ must have no zeros with $|z| \leq 1$ and $U(z)$ must have no zeros with $|z| \geq 1$. This stronger test comes in Section 5.3 on the Laplace transform.

Deconvolution in Two Dimensions

Our first example of deconvolution (for a telescope) was in 2D. Equation (2) was a double integral and $G * U$ was a 2D convolution. The computed examples went back to 1D, but not for any deep reason—only for simplicity. The two-dimensional problem needs double Fourier series or double Fourier integrals, with $e^{\imath\omega x}$ times $e^{i\theta y}$, but the principle stays the same:

$$G(x,y) * U(x,y) = B(x,y) \qquad \widehat{G}(\omega,\theta)\,\widehat{U}(\omega,\theta) = \widehat{B}(\omega,\theta) \qquad \widehat{U} = \widehat{B}/\widehat{G}. \quad (18)$$

The convolution rule is still all-important. But there can be a considerable difference in the algebra, from 1D to 2D. In one dimension, factorization is the key to explicit formulas. By computing the zeros of a polynomial $C(\omega)$, we get linear factors with easy inverses. That won't happen for $C(\omega,\theta) = \sum\sum c_{k\ell} e^{-\imath k\omega} e^{-\imath\ell\theta}$, except in the special case that we always hope for and very often construct:

Separation of variables	$C(\omega,\theta) = C_1(\omega)\,C_2(\theta)$	$1/C = (1/C_1)(1/C_2)$
Tensor products from 1D	$C = \mathsf{kron}(C_1, C_2)$	$C^{-1} = \mathsf{kron}(C_1^{-1}, C_2^{-1})$

(19)

This reduces the possibilities in 2D, but it makes the solution infinitely simpler.

Deblurring Images

A digital image is a matrix X of pixel values. A blurring matrix G multiplies X, and we observe the blurred image GX (plus noise, which is treated separately below). Knowing G, we want an efficient way to recover X from GX.

When the blurring is shift-invariant, G is a (two-dimensional) Toeplitz matrix. The image boundary has to be treated in a special way. Three methods are well described in the book *Deblurring Images* (Hansen, Nagy, and O'Leary, SIAM, 2006):

1. *Zero-padding* to embed X in a larger image (but those zeros may be very different from pixel values inside, and produce ringing)

2. *Periodic extension* to embed X in a larger matrix with repeated blocks of X

3. *Symmetric extension* uses $\mathsf{fliplr}(X)$ and $\mathsf{flipud}(X)$ to produce mirror images beyond the boundaries. Shift-invariance is lost but fast algorithms are available.

With shift-invariance, the blurring G is convolution with a *point-spread function* as in (18). For symmetric (even) extension, the Fast Cosine Transform replaces the FFT. It helps if the horizontal and vertical blurring separates as in (19). Blurring by a 2D Gaussian is also in MATLAB's Image Processing Toolbox.

Now include a noise matrix in the observed $Y = GX + N$. *Filtering is needed.* In the space domain, we can multiply Y by a lowpass averaging filter with molecule

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad \text{or} \quad \frac{1}{10}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad \text{or} \quad \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}.$$

With shift invariance, a two-dimensional DFT takes Y into the frequency domain.

For any $G = U\Sigma V^{\mathrm{T}}$, we can remove or damp the smallest singular values in Σ. This is "**safe inversion**", using $VD\Sigma^{-1}U^{\mathrm{T}}$ with filter factors in the diagonal matrix D: Factors $d_i = 1$ or 0 give a truncated SVD. Damping $1/\sigma_i$ by $d_i = \sigma_i^2/(\sigma_i^2 + \alpha)$ produces the Tychonov regularization of G^{-1} in Section 8.2, where X is minimizing $\|Y - GX\|^2 + \alpha\|X\|^2$. A larger penalty α removes more noise (and more signal too).

Problem Set 4.6

1 Solve this *cyclic convolution* equation for the vector d. (I would transform the convolution to multiplication.) Notice that $c = (5, 0, 0, 0) - (1, 1, 1, 1)$.

Deconvolution $c \circledast d = (4, -1, -1, -1) \circledast (d_0, d_1, d_2, d_3) = (1, 0, 0, 0)$.

2 There is no solution d if c changes to $C = (3, -1, -1, -1)$. Find the discrete transform of this C. Then find a nonzero D so that $C \circledast D = (0, 0, 0, 0)$.

3 These cyclic permutations are inverses. What are their eigenvalues?

$$C = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad D = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$$

4 If that cyclic delay C extends to a doubly infinite C_∞ (a non-cyclic delay), show that D_∞ (a non-cyclic advance) is still its inverse. For which complex numbers λ is $C_\infty - \lambda I$ not invertible? Use the test $e^{-i\omega} - \lambda \neq 0$ for all ω.

5 Now suppose C_+ is a *singly infinite* delay (lower triangular with 1's on the subdiagonal, *not invertible*). For which complex numbers λ is C_+ not invertible?

6 For singly infinite triangular Toeplitz matrices (starting at $n = 0$, not $-\infty < n < \infty$), show that U_+L_+ *stays Toeplitz but* L_+U_+ *does not.* The **Wiener-Hopf method** for $A_+u_+ = b_+$ factors $A(z) = U(z)L(z)$ and $A_+ = U_+L_+$.

7 What is the inverse of the 1D Gaussian convolution $G*U = \int e^{-s^2/2}U(x-3)\,ds$? What is a 2D Gaussian convolution $G(x, y) * U(x, y)$ and its inverse?

4.7 WAVELETS AND SIGNAL PROCESSING

A key idea of wavelets is to separate the incoming signal into *averages* (smooth parts) and *differences* (rough parts). Let me take the inputs two at a time, with no overlap, to get the simplest wavelet transform. This "2-point DFT" is named after Haar:

$$\textbf{Haar wavelet} \quad x = x_1, x_2, x_3, x_4 \longrightarrow \begin{matrix} \textbf{averages} & y = \dfrac{x_2 + x_1}{2} \text{ and } \dfrac{x_4 + x_3}{2} \\ \textbf{differences} & z = \dfrac{x_2 - x_1}{2} \text{ and } \dfrac{x_4 - x_3}{2} \end{matrix}$$

I will describe the inverse transform, the next iteration, and then the purpose.

First point You could quickly recover the four x's from the two y's and two z's. Addition would give x_2 and x_4. Subtraction would give x_1 and x_3. This **inverse transform** uses the same operations (plus and minus) as the forward transform.

Second point We could **iterate** by taking averages and differences of the y's:

$$\textbf{Next scale} \quad y = \frac{x_2 + x_1}{2} \text{ and } \frac{x_4 + x_3}{2} \longrightarrow \begin{matrix} \textbf{average} & yy = \dfrac{x_4 + x_3 + x_2 + x_1}{4} \\ \textbf{difference} & zy = \dfrac{x_4 + x_3 - x_2 - x_1}{4} \end{matrix}$$

From yy and zy we quickly recover the two y's. Then with the two z's we recover all four x's. The information is always there, but we have changed to a "wavelet basis." In matrix language, the transform is just multiplying x by an invertible matrix A. The inverse transform (to reconstruct x) multiplies by a synthesis matrix $S = A^{-1}$.

Third point A key application of wavelet transforms is in **compression**. Signals and images and videos come with more bits than we can hear or see. High Definition TV and medical imaging by MR produce enormous bit streams (an image is 8 bits per pixel, 24 for color, with millions of pixels). We can't drop small x's and leave blanks in the picture. But we can drop small z's without a significant loss. Compression comes between the transforms A and S:

$$\begin{matrix} \textbf{Input} \\ \textbf{signal } \boldsymbol{x} \end{matrix} \xrightarrow{A} \begin{matrix} \textbf{Wavelet} \\ \textbf{transform} \end{matrix} \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{z} \end{bmatrix} \longrightarrow \begin{matrix} \textit{Compressed} \\ \textit{transform} \end{matrix} \begin{bmatrix} \widehat{\boldsymbol{y}} \\ \widehat{\boldsymbol{z}} \end{bmatrix} \xrightarrow{S = A^{-1}} \begin{matrix} \textbf{Output} \\ \textbf{signal } \widehat{\boldsymbol{x}} \end{matrix}$$

Compression is nonlinear and lossy. The transforms are linear and lossless. Wavelet theory concentrates on finding transforms that keep this overall structure, but use more refined filters. The Haar filter coefficients are $\frac{1}{2}, \frac{1}{2}$ for "running averages" and $\frac{1}{2}, -\frac{1}{2}$ for "running differences." Better filters in A will have more coefficients (a favorite pair is 9/7), carefully chosen to keep the inverse transform simple and fast.

Signals and Images

This section has two purposes. One is to develop the wavelet transform. Haar's averages and differences are a first step—they opened the door to better discrete wavelet transforms. The DWT creates the wavelet coefficients from *filters*, not from formulas like $c_k = \sum f_j w^{-jk}$. The key is to build the transform from easily invertible pieces.

Our second purpose is to represent signals and images (often medical images) in a **sparse and piecewise smooth way**. Sparsity means few coefficients, to control cost and storage and transmission rate. Smoothness means close approximation to natural images. "**Piecewise**" is our recognition that **the edges in those images are highly important**. This is where Fourier falls down. Even in one dimension, the Gibbs phenomenon and the slow $1/k$ decay of coefficients lead to ringing and smearing at a jump in $f(t)$.

An $\boldsymbol{\ell^1}$ **penalty term** discourages a crowd of small coefficients. A **total variation penalty term** (ℓ^1 norm of the gradient) discourages oscillations. Get the jump over with and stay smooth on both sides. Section 8.6 will return to the algorithms and the duality theory behind sparse and smooth compression. This section motivates the energy minimizations that are transforming JPEG and discrete cosines into better codecs. Here are four steps in $f \approx \sum c_k \phi_k$:

1. Linear transform	Use the first n coefficients (Fourier, wavelet,...)
2. Nonlinear transform	Use the largest n coefficients (a form of basis pursuit)
3. Sparse transform	Minimize $\left\|f - \sum c_k \phi_k\right\|_2^2 + \alpha \sum \lvert c_k \rvert$ (the LASSO idea)
4. Smooth transform	Minimize $\left\|f - \sum c_k \phi_k\right\|^2 + \alpha \lvert \sum c_k \phi_k \rvert_{\mathrm{TV}}$ (total variation)

Fourier versus Wavelets

So much of mathematics involves the representation of functions—**the choice of basis**. A central example in pure and applied mathematics is the Fourier series. Its discrete version is computed by the Fast Fourier Transform, the most important algorithm of the 20th century. The Fourier basis is terrific—no basis will ever be so useful—but it is imperfect. Sines and cosines are global instead of local, and they give poor approximation at a jump (Gibbs phenomenon).

Four properties we want are: *local basis*, *easily refined*, *fast to compute*, *good approximation by a few terms*. Splines and finite elements achieve the first three, but removing terms will leave blank intervals. Wavelets permit compression of data—which is needed in so many applications where the volume of data is overwhelming.

To compare wavelets with sines and cosines, we need functions and not vectors. From discrete time, we move into the parallel world of continuous time. A lowpass filter like $\frac{1}{2}, \frac{1}{2}$ leads to the ***scaling function*** $\phi(t)$. A highpass filter like $\frac{1}{2}, -\frac{1}{2}$ leads to the ***wavelet*** $w(t)$. For a half-length vector like y, the parallel in the continuous case is to ***compress the t-axis***. We now meet $\phi(2t)$, which squeezes the graph of $\phi(t)$:

Averages lead to Haar scaling function ϕ $\quad \phi(t) = \phi(2t) + \phi(2t-1)$ (1)

Differences lead to the Haar wavelet w $\quad w(t) = \phi(2t) - \phi(2t-1)$ (2)

That two-scale "refinement equation" asks $\phi(t)$ to be the sum of its compression $\phi(2t)$ and the shifted compression $\phi(2t-1)$. The solution is the ***box function*** in Figure 4.14. Then the wavelet $w(t)$ is the difference of the two "half-boxes."

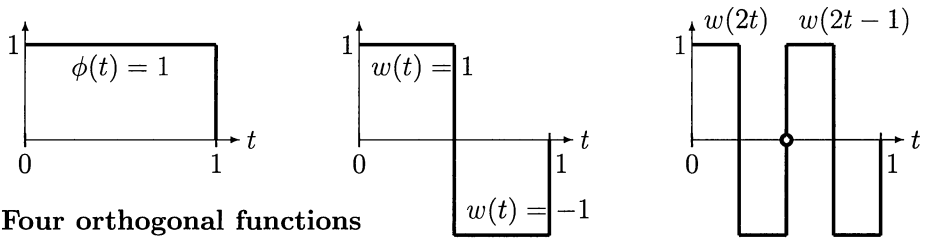


Figure 4.14: **Box function $\phi(t)$. Haar wavelet $w(t)$.** Rescaled $w(2t)$ and $w(2t-1)$.

Scaling functions give averages, and wavelets give details. When details are not significant, they can be compressed away to leave a smoothed signal. The image processing standard JPEG2000 chose filter pairs known as "9/7" and "5/3" from the count of coefficients. The coefficients decide the quality of the wavelet basis.

Wavelets are not perfect, and we are already seeing new ideas beyond these. To represent a face, or a signature, or the gravitational potential, we need basis functions that match particular inputs. When a video on the web stops because of congestion, you know that a more efficient representation is needed (and will be found).

Time-Frequency Multiscale Analysis

Overall, the purpose of wavelets is to represent signals in **time and frequency**. The Fourier description is entirely in the frequency domain. To know *when* something happened (such as a jump) the transform $\widehat{f}(k)$ has to be inverted to $f(t)$. The "short-time Fourier transform" operates on a sequence of windows of $f(t)$, to preserve part of the time information—but not optimally.

Wavelets capture high frequencies over short times (quick pulses). They see low frequencies over longer times. The building blocks are functions $w_{jk}(t) = w(2^j t - k)$, where j decides the scale ($w(2t)$ doubles all frequencies) and k decides the position ($w(t-k)$ shifts all time points). Those wavelet basis functions are LEGO blocks (Figure 4.15) in the Haar case of up-and-down square waves. New and smarter wavelets use better filters (short fast convolutions) on intervals that overlap. The construction takes patience but the purpose is clear: to combine higher accuracy with computational speed (of the inverse too).

At each scale, a lowpass filter captures averages and a highpass filter captures details. The details generally have low energy. Compression assigns them very few bits. The averages $y(n)$ contain most of the energy. By downsampling to $y(2n)$, we rescale the time. Then the lowpass and highpass filters are applied again, to capture averages and details at the coarser scale in Figure 4.15b. Those are **subband filters**.

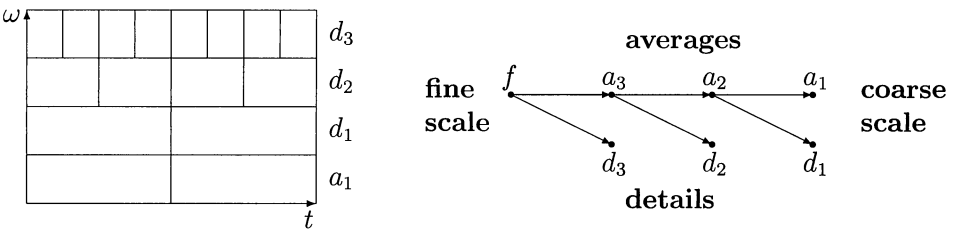


Figure 4.15: Time-frequency LEGO blocks: averages + details at time scales 2^j.

You could compare this time-frequency picture with musical notation. Time goes forward as you read the music. Frequency goes from low C to middle C to high C. Haar wavelets could be played with two fingers, left hand for averages and right hand for details. A chord gives you several frequencies, and $F(t,\omega)$ can contain *all* frequencies. But we have to point out what makes this subject difficult: $F(t,\omega)$ *is redundant.* If we know $f(t)$, we know everything. This is a deep topic [72], with uncertainty principles and Weyl-Heisenberg groups and fascinating transforms.

Altogether this multirate filter bank produces a **Discrete Wavelet Transform** (DWT). The inverse transform (IDWT) reassembles a_j from averages a_{j-1} and details d_{j-1}. This inverse process also uses a lowpass-highpass filter pair. Short filters are cheap, symmetric filters look best, orthogonal filters preserve energy, longer filters give sharp frequency cutoffs. We can't have all those properties at once!

Wavelet Basis and Refinement Equation

A wavelet basis is created from $w(t)$ by rescaling its graph (compressing by $2, 4, 8, \ldots$) and then shifting along the t axis to cover more intervals:

Rescaled by 2^j and shifted by k $$w_{jk}(t) = 2^{j/2} w(2^j t - k)\,. \tag{3}$$

The rescaled function $w(2^j t)$ is zero (unlike cosines) after an interval of length $2^{-j}N$. A fundamental property of all wavelets, like Haar's up-down $w(t)$, is *zero mean*:

Average value zero $$\int_{-\infty}^{\infty} w(t)\,dt = 0 \quad \text{and then} \quad \int_{-\infty}^{\infty} w_{jk}(t)\,dt = 0\,. \tag{4}$$

Thus the wavelets are orthogonal to the constant function 1. To approximate functions with nonzero integral, the scaling function $\phi(t)$ and its translates $\phi(t-k)$ are added to the basis. The continuous time expansion of $f(t)$ includes all ϕ and w:

Wavelet series
$$f(t) = \sum_{k=-\infty}^{\infty} a_k \phi(t-k) + \sum_{k=-\infty}^{\infty} \sum_{j=0}^{\infty} b_{jk} w_{jk}(t)\,. \tag{5}$$

This series (parallel to Fourier) indicates the key ideas of the wavelet basis:

1. All basis functions are now localized in time (compact support).
2. The scaling functions $\phi(t-k)$ produce an "averaged" or "smoothed" signal.
3. Wavelets $w_{jk}(t)$ fill in the multiscale details at all scales $j = 0, 1, 2, \ldots$.

Low frequencies are associated with $\phi(t)$ and high frequencies with $w(t)$. Since a typical signal is smooth or at least piecewise smooth, *most of the information is carried by the scaling functions.* The simplest form of wavelet compression is to delete the wavelet part and destroy the fine details (we still recognize the image).

To separate signal from noise, "thresholding" keeps coefficients a_k and b_{jk} that are larger than a specified value. A more subtle compression algorithm replaces each a_k and b_{jk} by a binary number (the smallest coefficients are replaced by zero). After this "quantization," the binary numbers are easy to save and transmit.

The key point is that $\phi(t)$ *is a combination of the rescaled functions* $\phi(2t-k)$:

Refinement equation
$$\phi(t) = 2\textstyle\sum_{k=0}^{N} h(k)\phi(2t-k)\,. \tag{6}$$

This is the most important equation in wavelet theory. The filter coefficients $h(k)$ are the only numbers that are needed and used to implement a wavelet basis, along with the corresponding numbers $g(k)$ in the *wavelet equation*:

Wavelet equation
$$w(t) = 2\textstyle\sum_{k=0}^{M} g(k)\phi(2t-k)\,. \tag{7}$$

The $h(k)$ are the coefficients in a lowpass filter, and the $g(k)$ are the coefficients in a highpass filter. This connects filter banks to wavelets. The choice of these numbers determines $\phi(t)$ and $w(t)$. This pattern is called ***multiresolution analysis***.

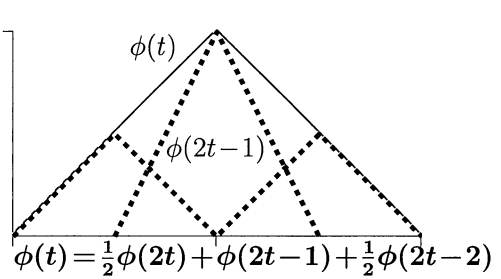
Analysis and synthesis can have different filter pairs, producing two ϕ-w pairs. One pair determines the coefficients a_k and b_{jk} in the series (5); this is the analysis step. The other pair yields $\phi(t)$ and $w(t)$ from (6) and (7); this is the synthesis step. We can interpret (6) and (7) as statements about three spaces of functions:

$$\begin{aligned} &\textbf{Coarse averages} && \boldsymbol{V_0} = \text{all combinations of } \phi(t-k) \\ &\textbf{Coarse details} && \boldsymbol{W_0} = \text{all combinations of } w(t-k) \\ &\textbf{Finer scale} && \boldsymbol{V_1} = \text{all combinations of } \phi(2t-k) \end{aligned} \tag{8}$$

By equations (6) and (7), $\boldsymbol{V_0}$ and $\boldsymbol{W_0}$ are contained in $\boldsymbol{V_1}$. **We want $\boldsymbol{V_0} + \boldsymbol{W_0} = \boldsymbol{V_1}$.**

The wavelet transform is a change of basis, to separate averages from details (y's and z's from x's). Fine signals in $\boldsymbol{V_1}$ split into pieces in $\boldsymbol{V_0}$ and $\boldsymbol{W_0}$. Then recursively, $\boldsymbol{V_1} + \boldsymbol{W_1} = \boldsymbol{V_2}$. Wavelets give scale + time; Fourier gives frequency.

Example After the box function, the simplest $\phi(t)$ comes from the filter $(1,2,1)/4$. $\phi(t)$ is the **hat function**. Figure 4.16 shows $\phi(t)$ as a combination of three half-hats. The wavelet $w(t)$ would be the combination of half-hats in (7), with mean zero from $g(k)$.



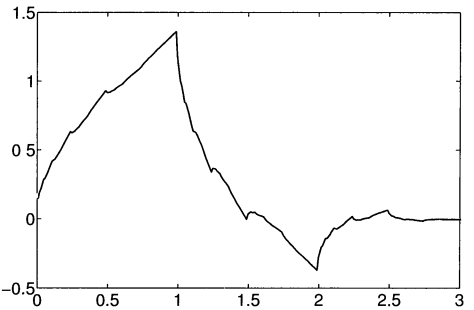


Figure 4.16: Scaling functions from lowpass filters $(1,2,1)/4$ and Daubechies (14).

The filter $(1,4,6,4,1)/16$ leads to a cubic B-spline (hats are linear B-splines). In fact splines are almost *the only simple solutions* to (6). The right side of Figure 4.16 is much more typical of $\phi(t)$. The **cascade algorithm** solves equation (6)—substitute $\phi^{(i)}(t)$ into the right side $\sum 2h(k)\phi^{(i)}(2t-k)$ to find $\phi^{(i+1)}(t)$. The code is on the cse site.

```
h = [1 + sqrt(3), 3 + sqrt(3), 3−sqrt(3), 1−sqrt(3)]/8; n = length(h)−1;
tsplit = 100; tt = 0:1/tsplit:n; ntt = length(tt); phi = double(tt < 1);
while 1                          % Iterate until convergence or divergence
   phinew = 0 * phi;
   for j = 1:ntt
     for k = 0:n
       index = 2 * j−k * tsplit + 1;
       if index >= 1 & index <= n * tsplit + 1
         phinew(j) = phinew(j) + 2 * h(k + 1) * phi(index);
       end
     end
   end
   plot(tt, phinew), pause(1e−1)
   if max(abs(phinew)) > 100, error('divergence'); end
   if max(abs(phinew−phi)) < 1e−3, break; end
   phi = phinew;
end
```

Filter Banks

The fundamental step is the choice of filter coefficients $h(k)$ and $g(k)$. They determine all the properties (good or bad) of the wavelets. We illustrate with the eight numbers in a very important 5/3 filter bank (notice the symmetry of each filter):

Lowpass coefficients $h(0), h(1), h(2), h(3), h(4) = \mathbf{-1, 2, 6, 2, -1}$ (divide by 8)

Highpass coefficients $g(0), g(1), g(2) = \mathbf{1, -2, 1}$ (divide by 4)

A filter is a **discrete convolution** acting on the inputs $x(n) = (\ldots, x(0), x(1), \ldots)$:

$$\textbf{Filter pair} \qquad y(n) = \sum_{k=0}^{4} h(k)x(n-k) \quad \text{and} \quad z(n) = \sum_{k=0}^{2} g(k)x(n-k). \tag{9}$$

The input $x = (\ldots, 1, 1, 1, \ldots)$ is unchanged by the lowpass filter, since $\sum h(k) = 1$. This constant signal is stopped by the highpass filter since $\sum g(k) = 0$.

The fastest oscillation $x = (\ldots, 1, -1, 1, -1, \ldots)$ sees the opposite effects. It is stopped by the lowpass filter ($\sum(-1)^k h(k) = 0$) and passed by the highpass filter ($\sum (-1)^k g(k) = 1$). Filtering a pure frequency input $x(n) = e^{in\omega}$ multiplies those inputs by $H(\omega)$ and $G(\omega)$, and those are the response functions to know:

$$\textbf{Frequency responses} \qquad H(\omega) = \sum h(k)e^{-ik\omega} \qquad G(\omega) = \sum g(k)e^{-ik\omega} \tag{10}$$

For the all-ones vector, $H = 1$ and $G = 0$ at $\omega = 0$. The oscillating vector $x(n) = (-1)^n = e^{in\pi}$ has opposite responses $H(\pi) = 0$ and $G(\pi) = 1$. *The multiplicity of this* "zero at π" *is a crucial property for the wavelet construction.* In the 5/3 example, $H(\omega)$ in Figure 4.17 has a double zero at $\omega = \pi$ because $(1 + e^{-i\omega})^2$ divides $H(\omega)$. Similarly $G(\omega) = (1 - e^{-i\omega})^2$ has a double zero at $\omega = 0$.

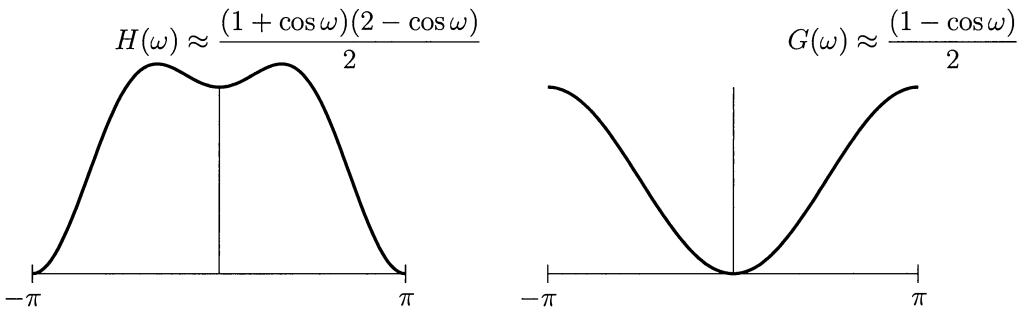


Figure 4.17: Frequency response functions, lowpass $H(\omega)$ and highpass $G(\omega)$.

The two filters combine into a ***filter bank*** (the wavelet transform!). The input is x, the filters give generalized averages y and differences z. To achieve an equal number of outputs and inputs, we ***downsample y and z***. By keeping only their even-numbered components $y(2n)$ and $z(2n)$, their length is cut in half. The Haar transform dropped $x_3 \pm x_2$. The block diagram shows filtering and downsampling:

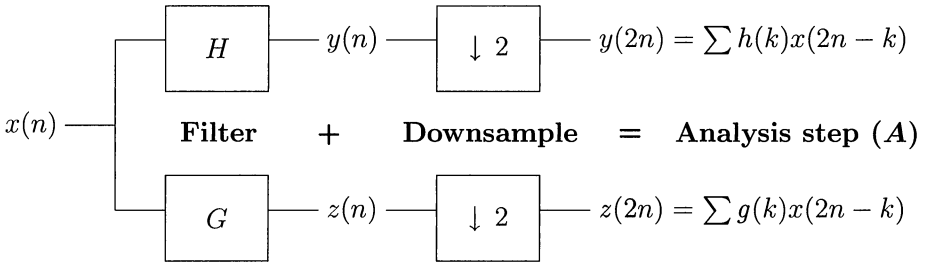


Figure 4.18: The discrete wavelet transform (**DWT**) separates averages and details.

In matrix language, the *wavelet transform* is a multiplication Ax with a *double shift* in the rows of A (from the downsampling step that removes odd-numbered rows):

DWT matrix $$\boldsymbol{A} = \begin{bmatrix} (\downarrow 2)\,\boldsymbol{H} \\ (\downarrow 2)\,\boldsymbol{G} \end{bmatrix} = \begin{bmatrix} -1 & 2 & 6 & 2 & -1 & & & \\ & & -1 & 2 & 6 & 2 & -1 & \\ & & & & \cdot & \cdot & \cdot & \cdot \\ 0 & 1 & -2 & 1 & 0 & & & \\ & & 0 & 1 & -2 & 1 & 0 & \\ & & & & \cdot & \cdot & \cdot & \cdot \end{bmatrix}.$$

An ordinary filter has rows shifted by one, not two. H and G are constant-diagonal Toeplitz matrices, before $\downarrow 2$. For long signals $x(n)$, the model has $-\infty < n < \infty$. Matrices are doubly infinite. For a finite-length input we could assume periodicity, and loop around. Extending $x(n)$ in a symmetric way at each end (Problem 2) is better than the wraparound (cyclic convolution) in S below.

With 1024 samples $x(n)$, the rows still have only five or three nonzeros. Ax is computed in 4 times 1024 multiplications and additions. The DWT is fast. Even with iteration **the transform is $O(N)$**, because signals get shorter and $\frac{1}{2} + \frac{1}{4} + \cdots = 1$.

Perfect Reconstruction

So far the two filters $h(k)$ and $g(k)$ have been separate—no connection. But their interrelation makes everything work. To display this connection we put a second pair of filters into the *columns* of a matrix S, again with double shifts. These "synthesis" filters f and e come from *alternating the signs in the first pair* g and h. Because the choice was good, ***S is the inverse of A***. I will use wraparound to make S finite:

Synthesis $$A^{-1} = S = \frac{1}{16}\begin{bmatrix} 0 & 0 & 2 & 2 & 0 & 2 \\ 1 & 0 & 1 & -6 & 1 & 1 \\ 2 & 0 & 0 & 2 & 2 & 0 \\ 1 & 1 & 0 & 1 & -6 & 1 \\ 0 & 2 & 0 & 0 & 2 & 2 \\ 0 & 1 & 1 & 1 & 1 & -6 \end{bmatrix}$$

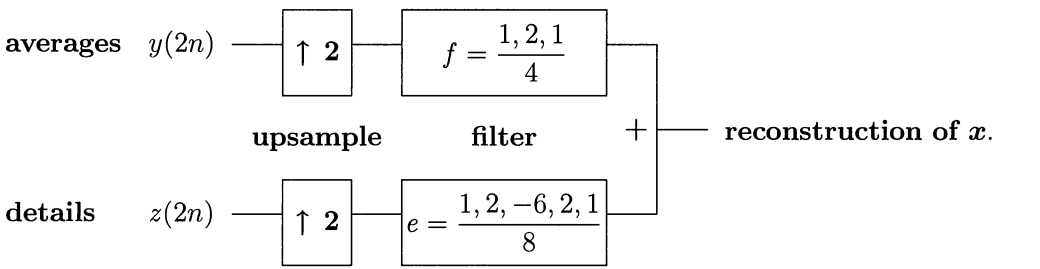
Lowpass $(1, 2, 1)$
Highpass $(1, 2, -6, 2, 1)$

S produces the **inverse wavelet transform**. A direct calculation verifies $AS = I$. The inverse transform is as fast as A. It is not usual for a sparse matrix A to have a sparse inverse S, but the wavelet construction makes this happen.

The columns of S are the wavelet basis vectors f and e (discrete ϕ's and w's). Multiplying by A produces the coefficients Ax in the discrete wavelet transform. Then SAx reconstructs x because $SA = I$:

Perfect reconstruction
$$x = S(Ax) = \sum (\text{basis vectors in } S)(\text{coefficients in } Ax). \quad (11)$$

It is useful to see the block form of the synthesis bank S, the inverse wavelet transform:



The *upsampling* step $(\uparrow 2)y$ gives the full-length vector $(\ldots, y(0), 0,\ y(2), 0, \ldots)$. The final output from SAx is a delay to $x(n - \ell)$ because the filters are "causal." This means that the coefficients are $h(0), \ldots, h(4)$ rather than $h(-2), \ldots, h(2)$. Then SA can have 1's on diagonal ℓ (an ℓ step delay) instead of diagonal 0.

What condition on the filters, two in analysis and two in synthesis, ensures that $S = A^{-1}$? The top half of A and the left half of S have lowpass filters h and f:

Lowpass
$$\frac{1}{16}(-1, 2, 6, 2, -1) * (1, 2, 1) = \frac{1}{16}(\mathbf{-1, 0, 9, 16, 9, 0, -1}) = \boldsymbol{p}. \quad (12)$$

This convolution is a multiplication of frequency responses $H(\omega)$ and $F(\omega)$:

$$\left(\sum h(k)e^{-ik\omega}\right)\left(\sum f(k)e^{-ik\omega}\right) = \frac{1}{16}\left(-1 + 9e^{-i2\omega} + 16e^{-i3\omega} + 9e^{-i4\omega} - e^{-i6\omega}\right). \quad (13)$$

Multiplying row zero of A times column zero of S produces the coefficient $\frac{1}{16}(16) = 1$. With the double shift in rows of A and columns of S, the key to perfection is this:

$$\boldsymbol{AS = I} \qquad \textbf{The product of lowpass responses } \boldsymbol{H(\omega)\,F(\omega)} \textbf{ must have only one odd power (like } \boldsymbol{e^{-i3\omega}}\textbf{).}$$

This condition also assures that the highpass product is correct. The last rows of A (with $1, -2, 1$) times the last columns of S (with $1, 2, -6, 2, 1$) look like (12) and (13), but with signs of even powers reversed. When we combine lowpass and highpass, they cancel. Only the odd term survives, to give one diagonal in AS.

The construction of good filter banks A and S now reduces to three quick steps:

1. Choose a symmetric filter p like (12), with $P(\omega) = \sum p(k)e^{-ik\omega}$.

2. Factor $P(\omega)$ into $H(\omega)\,F(\omega)$ to get lowpass filters $h(k)$ and $f(k)$.

3. Reverse order and alternate signs to get highpass coefficients $e(k)$ and $g(k)$.

Orthogonal Filters and Wavelets

A filter bank is **orthogonal** when $S = A^{\mathrm{T}}$. Then we have $A^{\mathrm{T}}A = I$ in discrete time. The continuous-time functions $\phi(t)$ and $w(t)$ use those filter coefficients and inherit orthogonality. All functions in the wavelet expansion (5) will be orthogonal. (We only know this from the construction—there are no simple formulas for $\phi(t)$ and $w(t)$!) Then wavelets compete with Fourier on this property too.

The key to $S = A^{\mathrm{T}}$ is a "spectral factorization" $P(\omega) = H(\omega)\overline{H(\omega)} = |H(\omega)|^2$. For the filter $p(k)$ in (12), this factorization of (13) leads to the orthogonal wavelets discovered by Ingrid Daubechies. Her $H(\omega)$ and $\overline{H(\omega)}$ have these neat coefficients:

$$\begin{array}{ll} \textbf{Daubechies 4/4} & h = (1+\sqrt{3}, 3+\sqrt{3}, 3-\sqrt{3}, 1-\sqrt{3})/8 \\ \textbf{orthogonal } \boldsymbol{S = A^{\mathrm{T}}} & g = (1-\sqrt{3}, -3+\sqrt{3}, 3+\sqrt{3}, -1-\sqrt{3})/8 \end{array} \tag{14}$$

Orthogonal filter banks have special importance (but not total importance). The rows of A are the columns of S, so the inverse is also the transpose: $S = A^{-1} = A^{\mathrm{T}}$. The product polynomial P is factored specially into $|H(e^{-i\omega})|^2$.

For image processing, symmetry is more important than orthogonality and we choose 5/3 or 9/7. Orthogonal filters lead to *one* pair $\phi(t)$ and $w(t)$, orthogonal to their own translates. Otherwise four filters h, g, f, e give two scaling functions and wavelets [145]. The analysis $\phi(t)$ and $w(t)$ are "**biorthogonal**" to the synthesis functions. Biorthogonality is what we always see in the rows of one matrix and the columns of its inverse:

Biorthogonality $AA^{-1} = I$ means (row i of A) $\cdot$ (column j of A^{-1}) $= \delta_{ij}$.

Those even-numbered zeros in p lead to orthogonality of the wavelet bases at all scales (analysis functions times synthesis functions). This is the magic of wavelets:

$$\int_{-\infty}^{\infty} \phi_A(t) w_S\,(2^j t - k)\,dt = \int_{-\infty}^{\infty} \phi_S(t) w_A(2^j t - k)\,dt = 0 \quad \text{for all } k \text{ and } j \tag{15}$$

$$\int_{-\infty}^{\infty} \phi_A(t)\phi_S(t-k)\,dt = \int_{-\infty}^{\infty} w_A(t) w_S(2^j t - k)\,dt = \delta_{0j}\,. \tag{16}$$

Sparse Compression

Sines and cosines capture smooth signals. The wavelet transform saves small-scale features. When wavelets are tied to an x-y grid, *ridgelets* and *curvelets* avoid staircasing along edges. In the end we have a **dictionary** of trial functions ϕ_i. They are not independent, and "basis" is not the right word. How can we quickly find an approximation (with few terms) to an input signal s, from a highly redundant and non-orthogonal dictionary of functions?

Here is a greedy approach and an optimization approach [160]:

1. **Orthogonal Matching Pursuit**. At step k, include the ϕ_k that has largest inner product with the current residual $r = s - (c_1\phi_1 + \cdots + c_{k-1}\phi_{k-1})$. Those c's at step $k-1$ were chosen to minimize $\|r\|$.

2. **Basis Pursuit Denoising.** Minimize $\frac{1}{2}\|s - \sum c_i\phi_i\|^2 + L\sum|c_i|$. That ℓ^1 penalty term forces fewer nonzero coeffecients c_i as L is increased. This approach has seen tremendous development. Perhaps the best way to see the sparsifying effect of $L\sum|c_i|$ is by a simple example.

An Example of Sparse Solutions

For two equations $Ax = b$ in three unknowns, the complete solution is $x_{\text{part}} + x_{\text{null}}$:

$$\begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \end{bmatrix} \quad \text{is solved by} \quad x = \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 5 \end{bmatrix} + \begin{bmatrix} c \\ c \\ c \end{bmatrix}. \tag{17}$$

That particular $x = (0, 1, 5)$ is one of the solutions with two nonzeros. MATLAB's $x = \mathsf{A\backslash b} = (-5, -4, 0)$ is another (with a larger ℓ^1 norm). The LASSO solution $x = (-1, 0, 4)$ has smallest ℓ^1 norm $\|x\| = 5$.

The smallest ℓ^2 norm comes from the pseudoinverse of A, but that solution $x^+ = \mathsf{pinv}(A) * b = (-2, -1, 3)$ is not sparse at all. (Remember that x^+ is orthogonal to $y = (1, 1, 1)$ in the nullspace of A.) Our goal is **more sparsity**, not less.

To reach *one* nonzero in x, we must give up exact solutions to $Ax = b$. For noisy measurements b, this is totally acceptable. A minimization with penalty term $L\|x\|_1$ is successful:

$$\textbf{Basis Pursuit Denoising} \qquad \text{Minimize } \tfrac{1}{2}\|Ax - b\|^2 + L(|u| + |v| + |w|) \tag{18}$$

Solution with $L = 2$	$x = (u, v, w) = (\mathbf{0}, \mathbf{0}, \mathbf{3})$	**very sparse**
Solution with $L = 8$	$x = (u, v, w) = (\mathbf{0}, \mathbf{0}, \mathbf{0})$	**fully sparse**

For every L the minimizer is $x = (-(1 - L/2)_+, 0, (4 - L/2)_+)$. The components of x fall to zero and stay there. When L reaches $\|A^{\mathrm{T}}b\|_1 = 8$, the fully sparse $x = 0$ is optimal and the penalty is too heavy.

From ℓ^0 to ℓ^1

The reader may ask why the ℓ^1 norm appears, when the true measure of sparsity is the **number of nonzeros** in x. This "cardinality" of x is its ℓ^0 norm $\|x\|_0$. Our real problem is $Ax = b$ (noiseless case) with minimum $\|x\|_0$, which means maximum sparsity. For fixed cost L and noise in b, we want to minimize $\frac{1}{2}\|Ax - b\|^2 + L\|x\|_0$. Why is $\|x\|_0$ replaced by $\|x\|_1$?

That **counting norm** is important in applications: the number of nonzeros in a fieter, or branches in a network of pipelines, or bars in a truss, or stocks in a portfolio. But minimizing a count is exponentially difficult (**NP-hard**).

There is a sudden change in $\|x\|_0$ and a gradual change in $\|x\|_1$. Integer (Boolean) problems are hard, fractional (convex) problems are easier.

A packing problem with large boxes is simpler when you subdivide the boxes. The mathematical equivalent is to subdivide each nonzero x_i into pieces of size $< \epsilon$ and count again:

$\|x\|_0$ counts large pieces $\quad \|x_\epsilon\|_0$ counts pieces $< \epsilon$ $\quad$ **$\epsilon\|x_\epsilon\|_0$ approaches $\|x\|_1$**

It is **convexity** that makes minimization simple in Section 8.6. The absolute value $|x|$ is convex (its slope never decreases). The one-zero cardinality of x is *not convex* (it drops from 1 to 0 at $x = 0$). In the same way, $\|x\|_1$ is the best convex replacement for the counting norm $\|x\|_0$.

The remarkable discovery in this century is that the ℓ^1 solution almost always has nonzeros in the right places! That statement is probabilistic, not deterministic. Suppose we know that $Ax = b$ has a sparse solution x_S with only S nonzero components (so $\|x\|_0 = S$). To find that x_S we solve an ℓ^1 problem:

Linear programming $\quad x^*$ minimizes $\|x\|_1$ subject to $Ax = b$. $\quad$ (19)

The analysis of Donoho, Candès, Romberg, Tao... shows that $\boldsymbol{x^* = x_S}$ **with high probability provided** $\boldsymbol{m > S \log n}$. No need to sample more, no use to sample less. When sensors are expensive, $m << n$ is attractive in many applications. But the sampling must be suitably incoherent, reflected in the m by n matrix A:

- m random Fourier coefficients out of n
- 1024 pixels along 22 rays out of $(1024)^2$ pixels in an MR scan

That 50 : 1 reduction still aims to locate the nonzeros in the image. In some way it is overcoming the *Nyquist condition* that applies to band-limited functions: at least two samples within the shortest wavelength. One goal is an analog-to-digital converter that can cope with very high bandwidth. When Nyquist requires 1 gigahertz as the sampling rate, random sampling may be the only way.

One feature to notice for NP-hard problems in general. *A fast algorithm can be close to correct with high probability.* It can't be exact every time, some level of danger has to be accepted.

The Total Variation Norm

Denoising is a fundamental problem in computational image processing. The goal is to preserve important features which the human visual system detects (the edges, the texture, the regularity). All successful models take advantage of the regularity of natural images and the irregularity of noise. Variational methods can **allow for discontinuities but disfavor oscillations**, by minimizing the right energy.

The L^1 norm of the gradient measures the variation in $u(x, y)$:

Total variation

$$\|u\|_{\mathrm{TV}} = \iint |\operatorname{grad} u|\, dx\, dy = \sup_{|w|\le 1} \iint u \operatorname{div} w\, dx\, dy. \tag{20}$$

To see this TV norm in action, suppose an image is black on the left side, white on the right side, and pixels are missing in between. How do we "inpaint" to minimize the TV norm?

The best u is monotonic, because oscillations will be punished by $\iint |\operatorname{grad} u|\, dx\, dy$. Here that u jumps from 0 to 1 across an edge. Its TV norm is the **length of the edge**. (I think of $\operatorname{grad} u$ as a line of delta functions along the edge. Integrating gives the length. The dual definition in (20) avoids delta functions and yields the same answer.) So minimization not only accepts the edge, it aims to make it short (therefore smooth). Three comments:

1. Minimizing $\iint |\operatorname{grad} u|^2\, dx\, dy$ gives a gradual ramp, not a jump. The minimizer now solves Laplace's equation. It is far from a delta function, which has infinite energy in this root mean square norm.

2. The ramp $u = x$ (from $x = 0$ to 1) is also a minimizer in this example. The integral of $\operatorname{grad} u = (1, 0)$ for that unit ramp equals the integral of $\operatorname{grad} u = (\delta(x), 0)$. *The TV norm is convex but not strictly convex.* It is possible (as in linear programming) to have multiple minimizers. When $\|u\|_{\mathrm{TV}}$ is combined with the L^2 norm, this won't happen.

3. An early success by Osher, who pioneered the TV norm in imaging, was to restore a very noisy image in a criminal case. Medical imaging is now the major application, detecting tumors instead of thieves.

Image Compression and Restoration

Imaging science is today's name for a classical problem: to represent images accurately and process them quickly. This is part of applied mathematics, but the reader may feel that "science" is becoming an overused word. The science of electromagnetism has fundamental laws (Maxwell's equations). The TV norm also connects to a partial differential equation (for minimal surfaces). But now human parameters enter too. Perhaps "engineering science" is a useful description, emphasizing that

this subject combines depth with practical importance. The fundamental problem, to understand the statistics of natural images, is still unsolved.

Compression depends on the choice of a good basis. Cosines were the long-time leader in **jpeg**. Wavelets became the choice of JPEG 2000, to reduce blocking artifacts. But the other enemy is ringing (false oscillation), which tends to increase as basis functions get longer. We look for a successful compromise between fitting the data $g(x,y)$ and preserving the (piecewise) smoothness of natural images. One way is to insert that compromise into the minimization:

Total variation restoration Minimize $\frac{1}{2}\iint |u-g|^2 dx\,dy + \alpha \iint |\operatorname{grad} u|\,dx\,dy$ (21)

Duality plays a crucial role in forming the optimality equations (Section 8.6). The duality of u and w, displacements and forces, voltages and currents, has been a powerful idea throughout this book. It still has more to reveal, but first we turn back to classical mathematics and $f(x+iy)$—complex analysis and its applications.

Problem Set 4.7

1 For $h = \frac{1}{2}, \frac{1}{2}$ Haar's equations (6)-(7) have unusually simple solutions:

$$\phi(t) = \phi(2t) + \phi(2t-1) = \text{box} + \text{box} = \text{scaling function}$$

$$w(t) = \phi(2t) - \phi(2t-1) = \text{box} - \text{box} = \text{wavelet}$$

Draw the sum and difference of these two half-boxes $\phi(2t)$ and $\phi(2t-1)$. Show that all wavelets $w(2^j t - k)$ are orthogonal to $\phi(t)$.

2 The Daubechies polynomial $p(z) = -1 + 9z^2 + 16z^3 + 9z^4 - z^6$ from (12) has $p(-1) = 0$. Show that $(z+1)^4$ divides $p(z)$, so there are *four roots* at $z = -1$. Find the other two roots z_5 and z_6, from $p(z)/(z+1)^4$.

3 What cubic has the roots $-1, -1$, and z_5? Connect its coefficients to h or g in (14) for Daubechies 4/4.

4 What quartic has the roots $-1, -1, z_5$ and z_6? Connect it to equation (12) and the symmetric 5/3 filters.

5 Create a 2/6 filter bank whose two lowpass coefficients are Haar's $h = \frac{1}{2}, \frac{1}{2}$. The six highpass coefficients come from dividing $p(z)/\frac{1}{2}(z-1)$ which has degree 5.

6 Show that the hat function $H(t)$ solves the refinement equation (6) with lowpass coefficients $h = (1,2,1)/4$.

7 Show that the cubic B-spline $S(t) = H(t) * H(t)$ from Section 3.2 solves the refinement equation (6) for $h = (1,2,1)*(1,2,1)/16$. (You could use the cascade code on the cse site to draw $S(t)$.)

8 $A_6 = \begin{bmatrix} -1 & 2 & 6 & 2 & -1 & 0 \\ -1 & 0 & -1 & 2 & 6 & 2 \\ 6 & 2 & -1 & 0 & -1 & 2 \\ 0 & 1 & -2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -2 & 1 \\ -2 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$.

is a circulant matrix?
is a block (2 by 2) circulant?
has what inverse by MATLAB?
extends to which matrix A_8?

9 The *ideal lowpass filter* h has $H(\omega) = \sum h(k)e^{-ik\omega} = 1$ for $|\omega| \le \pi/2$ (zero for $\pi/2 < |\omega| < \pi$). What are the coefficients $h(k)$? What highpass coefficients $g(k)$ give $G = 1 - H$?

10 Upsampling a signal $x(n)$ inserts zeros into $u = (\uparrow 2)x$, by the rule $u(2n+1) = 0$ and $u(2n) = x(n)$. Show that $U(\omega) = X(2\omega)$.

11 For a 2D image $x(m,n)$ instead of a 1D signal $x(n)$, Haar's 2D filter produces which averages $y(m,n)$? There are now *three differences*, z_H (horizontal) and z_V (vertical) and z_{HV}. What are those four outputs from a checkerboard input $x(m,n) = 1$ or 0 ($m+n$ even or odd)?

12 Apply the cascade code on the **cse** site to find $\phi(t)$ for $h = (-1, 2, 6, 2, -1)/8$.

13 (Solution unknown) In the example with $A = [-1\ 1\ 0; 0\ -1\ 1]$ and $b = [1; 4]$, use the ℓ^0 norm (number of nonzero components) directly in $\frac{1}{2}||Ax - b||^2 + L||x||_0$. Minimize for increasing L.

14 With one equation, minimize $\frac{1}{2}(u + 2v + 3w - 6)^2 + L(|u| + |v| + |w|)$. At what value of L does $(0, 0, 0)$ become the minimizer?

15 What is the TV norm of $u(x, y)$ if $u = 1$ in the unit disc $x^2 + y^2 \le 1$ (zero elsewhere)? What is $||u||_{\mathrm{TV}}$ for $u = (\sin 2\pi x)(\sin 2\pi y)$ in the unit square?