# Chapter 1

# Convolution and Spectra

When Fourier transforms are applicable, it means the "earth response" now is the same as the earth response later. Switching our point of view from time to space, the applicability of Fourier transformation means that the "**impulse response**" here is the same as the impulse response there. An *impulse* is a column vector full of zeros with somewhere a one, say $(0,0,1,0,0,\cdots)'$ (where the prime $()'$ means transpose the row into a column.) An *impulse response* is a column from the matrix

$$\mathbf{q} \quad = \quad \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \\ q_6 \\ q_7 \end{bmatrix} \quad = \quad \begin{bmatrix} b_0 & 0 & 0 & 0 & 0 & 0 \\ b_1 & b_0 & 0 & 0 & 0 & 0 \\ b_2 & b_1 & b_0 & 0 & 0 & 0 \\ 0 & b_2 & b_1 & b_0 & 0 & 0 \\ 0 & 0 & b_2 & b_1 & b_0 & 0 \\ 0 & 0 & 0 & b_2 & b_1 & b_0 \\ 0 & 0 & 0 & 0 & b_2 & b_1 \\ 0 & 0 & 0 & 0 & 0 & b_2 \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \end{bmatrix} \quad = \quad \mathbf{Bp} \tag{1.1}$$

The impulse response is the $\mathbf{q}$ that comes out when the input $\mathbf{p}$ is an impulse. In a typical application, the matrix would be about $1000 \times 1000$ and not the simple $8 \times 6$ example that I show you above. Notice that each column in the matrix contains the same waveform $(b_0, b_1, b_2)$. This waveform is called the "impulse response". The collection of impulse responses in equation (1.1) defines the the *convolution* operation.

Not only do the columns of the matrix contain the same impulse response, but each row likewise contains the same thing, and that thing is the backwards impulse response $(b_2, b_1, b_0)$. Suppose $(b_2, b_1, b_0)$ were numerically equal to $(1, -2, 1)/\Delta t^2$. Then equation (1.1) would be like the differential equation $\frac{d^2}{dt^2} p = q$. Equation (1.1) would be a finite-difference representation of a differential equation. Two important ideas are equivalent; either they are both true or they are both false:

1. The columns of the matrix all hold the same impulse response.

2. The differential equation has constant coefficients.

Let us take a quick peek ahead. The relationship of equation (1.1) with Fourier transforms is that the $k$-th row in (1.1) is the $k$-th power of $Z$ in a polynomial multiplication $Q(Z) = B(Z)P(Z)$. The relationship of any polynomial such as $Q(Z)$ to Fourier Transforms results from the relation $Z = e^{i\omega\Delta t}$, as we will see.

## 1.1 SAMPLED DATA AND Z-TRANSFORMS

Time and space are ordinarily thought of as continuous, but for the purposes of computer analysis we must discretize these axes. This is also called "**sampling**" or "**digitizing**." You might worry that discretization is a practical evil that muddies all later theoretical analysis. Actually, physical concepts have representations that are exact in the world of discrete mathematics.

Consider the idealized and simplified signal in Figure 1.1. To analyze such an observed

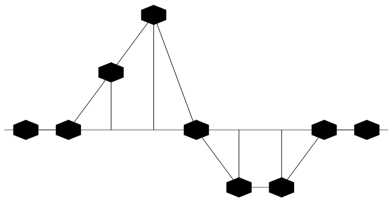

Figure 1.1: A continuous signal sampled at uniform time intervals. $\boxed{\text{cs-triv1}}$ [ER]

signal in a computer, it is necessary to approximate it in some way by a list of numbers. The usual way to do this is to evaluate or observe $b(t)$ at a uniform spacing of points in time, call this discretized signal $b_t$. For Figure 1.1, such a discrete approximation to the continuous function could be denoted by the vector

$$b_t \quad = \quad (\ldots\ 0, 0, 1, 2, 0, -1, -1, 0, 0, \ldots) \tag{1.2}$$

Naturally, if time points were closer together, the approximation would be more accurate. What we have done, then, is represent a signal by an abstract $n$-dimensional vector.

Another way to represent a signal is as a polynomial, where the *coefficients* of the polynomial represent the value of $b_t$ at successive times. For example,

$$B(Z) \quad = \quad 1 + 2Z + 0Z^2 - Z^3 - Z^4 \tag{1.3}$$

This polynomial is called a "$Z$**-transform**." What is the meaning of $Z$ here? $Z$ should not take on some numerical value; it is instead the **unit-delay operator**. For example, the coefficients of $ZB(Z) = Z + 2Z^2 - Z^4 - Z^5$ are plotted in Figure 1.2. Figure 1.2 shows the same waveform

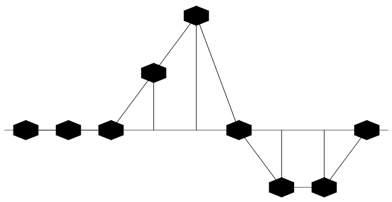

Figure 1.2: The coefficients of $ZB(Z)$ are the shifted version of the coefficients of $B(Z)$. $\boxed{\text{cs-triv2}}$ [ER]

as Figure 1.1, but now the waveform has been delayed. So the signal $b_t$ is delayed $n$ time units by multiplying $B(Z)$ by $Z^n$. The delay operator $Z$ is important in analyzing waves simply because waves take a certain amount of time to move from place to place.

Another value of the delay operator is that it may be used to build up more complicated signals from simpler ones. Suppose $b_t$ represents the acoustic pressure function or the seismogram observed after a distant explosion. Then $b_t$ is called the "**impulse response**." If another explosion occurred at $t = 10$ time units after the first, we would expect the pressure function $y(t)$ depicted in Figure 1.3. In terms of $Z$-transforms, this pressure function would be expressed as $Y(Z) = B(Z) + Z^{10}B(Z)$.

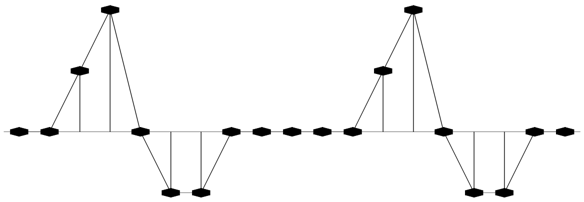

Figure 1.3: Response to two explosions. cs-triv3 [ER]

### 1.1.1 Linear superposition

If the first explosion were followed by an implosion of half-strength, we would have $B(Z) - \frac{1}{2}Z^{10}B(Z)$. If pulses overlapped one another in time (as would be the case if $B(Z)$ had degree greater than 10), the waveforms would simply add together in the region of overlap. The supposition that they would just add together without any interaction is called the "**linearity**" property. In seismology we find that—although the earth is a heterogeneous conglomeration of rocks of different shapes and types—when seismic waves travel through the earth, they do not interfere with one another. They satisfy linear **superposition**. The plague of **nonlinearity** arises from large amplitude disturbances. Nonlinearity is a dominating feature in hydrodynamics, where flow velocities are a noticeable fraction of the wave velocity. Nonlinearity is absent from reflection seismology except within a few meters from the source. Nonlinearity does not arise from geometrical complications in the propagation path. An example of two **plane wave**s superposing is shown in Figure 1.4.

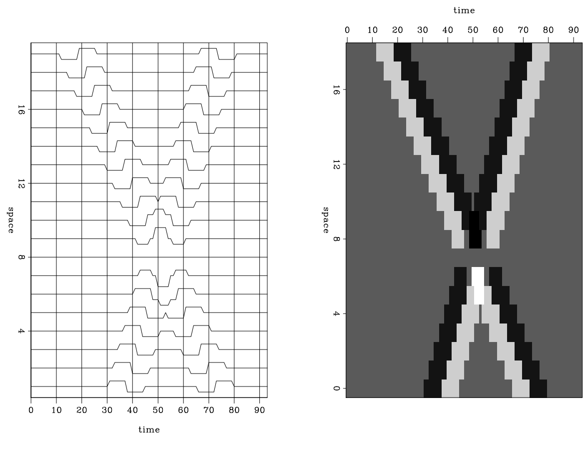


Figure 1.4: Crossing plane waves superposing viewed on the left as "wiggle traces" and on the right as "raster." cs-super [ER]

### 1.1.2 Convolution with Z-transform

Now suppose there was an explosion at $t=0$, a half-strength implosion at $t=1$, and another, quarter-strength explosion at $t=3$. This sequence of events determines a "source" time series, $x_t = (1, -\frac{1}{2}, 0, \frac{1}{4})$. The $Z$-transform of the source is $X(Z) = 1 - \frac{1}{2}Z + \frac{1}{4}Z^3$. The observed $y_t$ for this sequence of explosions and implosions through the seismometer has a $Z$-transform $Y(Z)$, given by

$$\begin{aligned} Y(Z) &= B(Z) - \frac{Z}{2}B(Z) + \frac{Z^3}{4}B(Z) \\ &= \left(1 - \frac{Z}{2} + \frac{Z^3}{4}\right) B(Z) \\ &= X(Z)\,B(Z) \qquad (1.4) \end{aligned}$$

The last equation shows **polynomial multiplication** as the underlying basis of time-invariant linear-system theory, namely that the output $Y(Z)$ can be expressed as the input $X(Z)$ times the impulse-response **filter** $B(Z)$. When signal values are insignificant except in a "small" region on the time axis, the signals are called "**wavelet**s."

### 1.1.3 Convolution equation and program

What do we actually do in a computer when we multiply two $Z$-transforms together? The filter $2+Z$ would be represented in a computer by the storage in memory of the coefficients $(2,1)$. Likewise, for $1-Z$, the numbers $(1,-1)$ would be stored. The polynomial multiplication program should take these inputs and produce the sequence $(2,-1,-1)$. Let us see how the computation proceeds in a general case, say

$$X(Z)\,B(Z) = Y(Z) \qquad (1.5)$$

$$(x_0 + x_1 Z + x_2 Z^2 + \cdots)(b_0 + b_1 Z + b_2 Z^2) = y_0 + y_1 Z + y_2 Z^2 + \cdots \qquad (1.6)$$

Identifying coefficients of successive powers of $Z$, we get

$$\begin{aligned} y_0 &= x_0 b_0 \\ y_1 &= x_1 b_0 + x_0 b_1 \\ y_2 &= x_2 b_0 + x_1 b_1 + x_0 b_2 \qquad (1.7) \\ y_3 &= x_3 b_0 + x_2 b_1 + x_1 b_2 \\ y_4 &= x_4 b_0 + x_3 b_1 + x_2 b_2 \\ &= \cdots\cdots\cdots\cdots\cdots\cdots \end{aligned}$$

In matrix form this looks like

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{bmatrix} = \begin{bmatrix} x_0 & 0 & 0 \\ x_1 & x_0 & 0 \\ x_2 & x_1 & x_0 \\ x_3 & x_2 & x_1 \\ x_4 & x_3 & x_2 \\ 0 & x_4 & x_3 \\ 0 & 0 & x_4 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \end{bmatrix} \tag{1.8}$$

The following equation, called the "convolution equation," carries the spirit of the group shown in (1.7):

$$y_k = \sum_{i=0}^{N_b} x_{k-i} b_i \tag{1.9}$$

To be correct in detail when we associate equation (1.9) with the group (1.7), we should also assert that either the input $x_k$ vanishes before $k = 0$ or $N_b$ must be adjusted so that the sum does not extend before $x_0$. These end conditions are expressed more conveniently by defining $j = k - i$ in equation (1.9) and eliminating $k$ getting

$$y_{j+i} = \sum_{i=0}^{N_b} x_j b_i \tag{1.10}$$

A convolution program based on equation (1.10) including end effects on both ends, is `convolve()`.

```
#       convolution:    Y(Z) = X(Z) * B(Z)
#
subroutine convolve( nb, bb, nx, xx, yy )
integer nb      # number of coefficients in filter
integer nx      # number of coefficients in input
                # number of coefficients in output will be nx+nb-1
real    bb(nb)  # filter coefficients
real    xx(nx)  # input trace
real    yy(1)   # output trace
integer ib, ix, iy, ny
ny = nx + nb -1
call null( yy, ny)
do ib= 1, nb
        do ix= 1, nx
                yy( ix+ib-1) = yy( ix+ib-1) + xx(ix) * bb(ib)
return; end
```

This program is written in a language called Ratfor, a "rational" dialect of Fortran. It is similar to the Matlab language. You are not responsible for anything in this program, but, if you are interested, more details in the last chapter of PVI[1], the book that I condensed this from.

[1] http://sepwww.stanford.edu/sep/prof/

### 1.1.4 Negative time

Notice that $X(Z)$ and $Y(Z)$ need not strictly be polynomials; they may contain both positive and negative powers of $Z$, such as

$$X(Z) \quad = \quad \cdots + \frac{x_{-2}}{Z^2} + \frac{x_{-1}}{Z} + x_0 + x_1 Z + \cdots \tag{1.11}$$

$$Y(Z) \quad = \quad \cdots + \frac{y_{-2}}{Z^2} + \frac{y_{-1}}{Z} + y_0 + y_1 Z + \cdots \tag{1.12}$$

The negative powers of $Z$ in $X(Z)$ and $Y(Z)$ show that the *data* is defined before $t = 0$. The effect of using negative powers of $Z$ in the *filter* is different. Inspection of (1.9) shows that the output $y_k$ that occurs at time $k$ is a linear combination of current and previous inputs; that is, $(x_i, i \leq k)$. If the filter $B(Z)$ had included a term like $b_{-1}/Z$, then the output $y_k$ at time $k$ would be a linear combination of current and previous inputs and $x_{k+1}$, an input that really has not arrived at time $k$. Such a filter is called a "**nonrealizable**" filter, because it could not operate in the real world where nothing can respond now to an excitation that has not yet occurred. However, nonrealizable filters are occasionally useful in computer simulations where all the data is prerecorded.

## 1.2 FOURIER SUMS

The world is filled with sines and cosines. The coordinates of a point on a spinning wheel are $(x, y) = (\cos(\omega t + \phi), \sin(\omega t + \phi))$, where $\omega$ is the angular frequency of revolution and $\phi$ is the phase angle. The purest tones and the purest colors are sinusoidal. The movement of a pendulum is nearly sinusoidal, the approximation going to perfection in the limit of small amplitude motions. The sum of all the tones in any signal is its "spectrum."

Small amplitude signals are widespread in nature, from the vibrations of atoms to the sound vibrations we create and observe in the earth. Sound typically compresses air by a volume fraction of $10^{-3}$ to $10^{-6}$. In water or solid, the compression is typically $10^{-6}$ to $10^{-9}$. A mathematical reason why sinusoids are so common in nature is that laws of nature are typically expressible as partial differential equations. Whenever the coefficients of the differentials (which are functions of material properties) are constant in time and space, the equations have exponential and sinusoidal solutions that correspond to waves propagating in all directions.

### 1.2.1 Superposition of sinusoids

Fourier analysis is built from the complex exponential

$$e^{-i\omega t} \quad = \quad \cos\omega t - i \sin\omega t \tag{1.13}$$

A Fourier component of a time signal is a complex number, a sum of real and imaginary parts, say

$$B \quad = \quad \mathrm{Re}\, B + i \mathrm{Im}\, B \tag{1.14}$$

which is attached to some frequency. Let $j$ be an integer and $\omega_j$ be a set of frequencies. A signal $b(t)$ can be manufactured by adding a collection of complex exponential signals, each complex exponential being scaled by a complex coefficient $B_j$, namely,

$$b(t) \quad = \quad \sum_j B_j \; e^{-i\omega_j t} \tag{1.15}$$

This manufactures a **complex-valued signal**. How do we arrange for $b(t)$ to be real? We can throw away the imaginary part, which is like adding $b(t)$ to its complex conjugate $\overline{b}(t)$, and then dividing by two:

$$\text{Re } b(t) \quad = \quad \frac{1}{2}\sum_j (B_j \; e^{-i\omega_j t} + \bar{B}_j \; e^{i\omega_j t}) \tag{1.16}$$

In other words, for each positive $\omega_j$ with amplitude $B_j$, we add a negative $-\omega_j$ with amplitude $\bar{B}_j$ (likewise, for every negative $\omega_j$ ...). The $B_j$ are called the "frequency function," or the "Fourier transform." Loosely, the $B_j$ are called the "**spectrum**," though in formal mathematics, the word "spectrum" is reserved for the product $\bar{B}_j B_j$. The words "**amplitude spectrum**" universally mean $\sqrt{\bar{B}_j B_j}$.

In practice, the collection of frequencies is almost always evenly spaced. Let $j$ be an integer $\omega = j \; \Delta\omega$ so that

$$b(t) \quad = \quad \sum_j B_j \, e^{-i(j \; \Delta\omega)t} \tag{1.17}$$

Representing a signal by a sum of sinusoids is technically known as "inverse Fourier transformation." An example of this is shown in Figure 1.5.

### 1.2.2 Sampled time and Nyquist frequency

In the world of computers, time is generally mapped into integers too, say $t = n\Delta t$. This is called "discretizing" or "sampling." The highest possible frequency expressible on a **mesh** is $(\cdots, 1, -1, +1, -1, +1, -1, \cdots)$, which is the same as $e^{i\pi n}$. Setting $e^{i\omega_{\max} t} = e^{i\pi n}$, we see that the maximum frequency is

$$\omega_{\max} \quad = \quad \frac{\pi}{\Delta t} \tag{1.18}$$

Time is commonly given in either seconds or sample units, which are the same when $\Delta t = 1$. In applications, frequency is usually expressed in cycles per second, which is the same as **Hertz**, abbreviated **Hz**. In computer work, frequency is usually specified in cycles per sample. In theoretical work, frequency is usually expressed in **radian**s where the relation between radians and cycles is $\omega = 2\pi f$. We use radians because, otherwise, equations are filled with $2\pi$'s. When time is given in sample units, the maximum frequency has a name: it is the "**Nyquist frequency**," which is $\pi$ radians or $1/2$ cycle per sample.

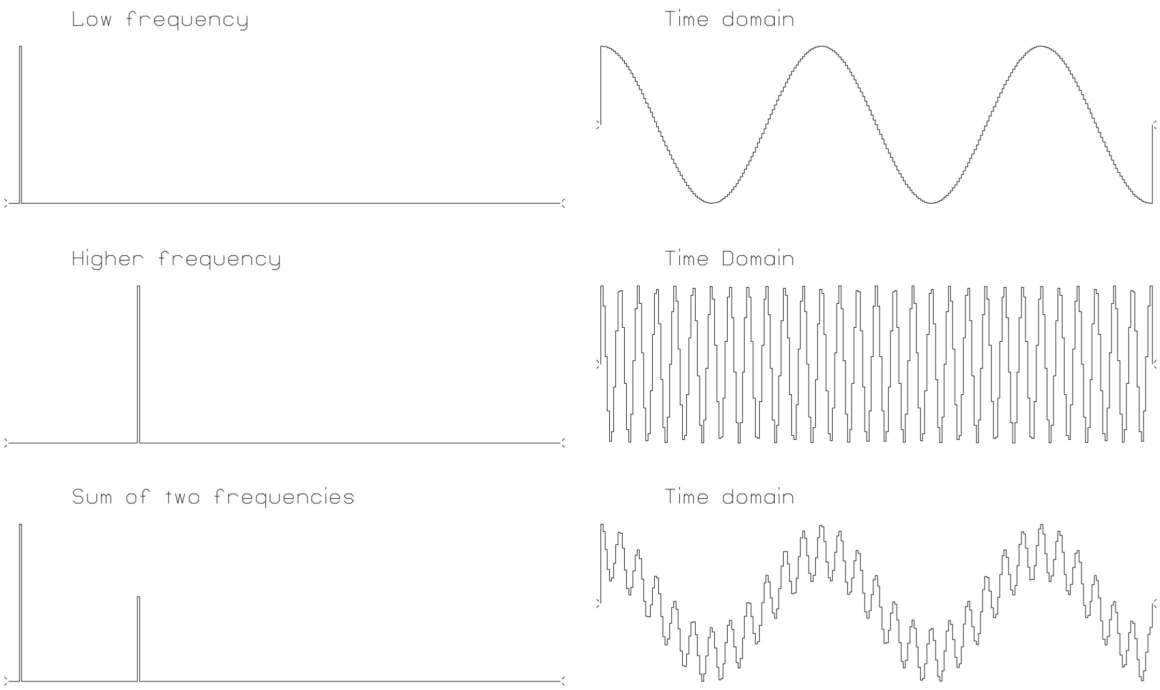


Figure 1.5: Superposition of two sinusoids. $\boxed{\text{cs-cosines}}$ [NR]

### 1.2.3 Fourier sum

In the previous section we superposed uniformly spaced frequencies. Now we will **superpose** delayed impulses. The frequency function of a delayed impulse at time delay $t_0$ is $e^{i\omega t_0}$. Adding some pulses yields the "**Fourier sum**":

$$B(\omega) \quad = \quad \sum_n b_n \, e^{i\omega t_n} \quad = \quad \sum_n b_n \, e^{i\omega n \Delta t} \tag{1.19}$$

The Fourier sum transforms the signal $b_t$ to the frequency function $B(\omega)$. Time will often be denoted by $t$, even though its units are sample units instead of physical units. Thus we often see $b_t$ in equations like (1.19) instead of $b_n$, resulting in an implied $\Delta t = 1$.

## 1.3 FOURIER AND Z-TRANSFORM

The frequency function of a pulse at time $t_n = n\Delta t$ is $e^{i\omega n \Delta t} = (e^{i\omega \Delta t})^n$. The factor $e^{i\omega \Delta t}$ occurs so often in applied work that it has a name:

$$Z \quad = \quad e^{i\omega \Delta t} \tag{1.20}$$

With this $Z$, the pulse at time $t_n$ is compactly represented as $Z^n$. The variable $Z$ makes **Fourier transform**s look like polynomials, the subject of a literature called "$Z$**-transform**s." The $Z$-transform is a variant form of the Fourier transform that is particularly useful for time-discretized (sampled) functions.

From the definition (1.20), we have $Z^2 = e^{i\omega 2\Delta t}$, $Z^3 = e^{i\omega 3\Delta t}$, etc. Using these equivalencies, equation (1.19) becomes

$$B(\omega) \quad = \quad B(\omega(Z)) \quad = \quad \sum_n b_n\, Z^n \tag{1.21}$$

### 1.3.1 Unit circle

In this chapter, $\omega$ is a real variable, so $Z = e^{i\omega\Delta t} = \cos\omega\Delta t + i\sin\omega\Delta t$ is a complex variable. It has unit magnitude because $\sin^2 + \cos^2 = 1$. As $\omega$ ranges on the real axis, $Z$ ranges on the unit circle $|Z| = 1$.

### 1.3.2 Differentiator

Calculus defines the differential of a time function like this

$$\frac{d}{dt}\, f(t) \quad = \quad \lim_{\Delta t \to 0} \frac{f(t) \, - \, f(t - \Delta t)}{\Delta t}$$

Computationally, we think of a differential as a finite difference, namely, a function is delayed a bit and then subtracted from its original self. Expressed as a $Z$-transform, the finite difference operator is $(1 - Z)$ with an implicit $\Delta t = 1$. In the language of filters, the time derivative is the filter $(+1, -1)$.

The filter $(1 - Z)$ is often simply called a "**differentiator**." It is displayed in Figure 1.6. Notice its amplitude spectrum increases with frequency. Theoretically, the amplitude spectrum

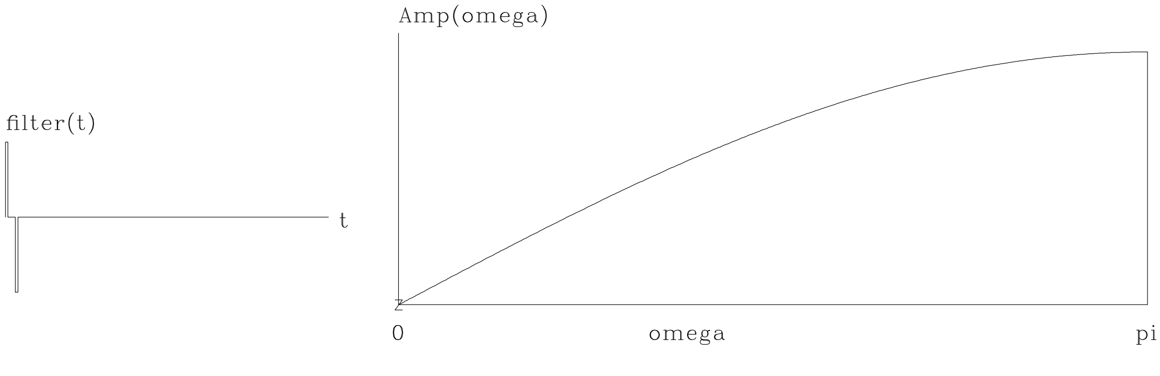


Figure 1.6: A discrete representation of the first-derivative operator. The filter $(1, -1)$ is plotted on the left, and on the right is an amplitude response, i.e., $|1 - Z|$ versus $\omega$. $\boxed{\text{cs-ddt}}$ [NR]

of a time derivative operator increases linearly with frequency. Here is why. Begin from Fourier representation of a time function (1.15).

$$b(t) \quad = \quad \sum_j B_j\, e^{-i\omega_j t} \tag{1.22}$$

$$\frac{d}{dt}\, b(t) \quad = \quad \sum_j -i\omega_j B_j\, e^{-i\omega_j t} \tag{1.23}$$

and notice that where the original function has Fourier coefficients $B_j$, the time derivative has Fourier coefficients $-i\omega B_j$.

In Figure 1.6 we notice the spectrum begins looking like a linear function of $\omega$, but at higher frequencies, it curves. This is because at high frequencies, a finite difference is different from a differential.

### 1.3.3 Gaussian examples

The filter $(1+Z)/2$ is a running average of two adjacent time points. Applying this filter $N$ times yields the filter $(1+Z)^N/2^N$. The coefficients of the filter $(1+Z)^N$ are generally known as **Pascal's triangle**. For large $N$ the coefficients tend to a mathematical limit known as a **Gaussian** function, $\exp(-\alpha(t-t_0)^2)$, where $\alpha$ and $t_0$ are constants that we will not determine here. We will not prove it here, but this Gaussian-shaped signal has a Fourier transform that also has a Gaussian shape, $\exp(-\beta\omega^2)$. The Gaussian shape is often called a "bell shape." Figure 1.7 shows an example for $N \approx 15$. Note that, except for the rounded ends, the bell shape seems a good fit to a triangle function. Curiously, the filter $(.75+.25Z)^N$ also tends

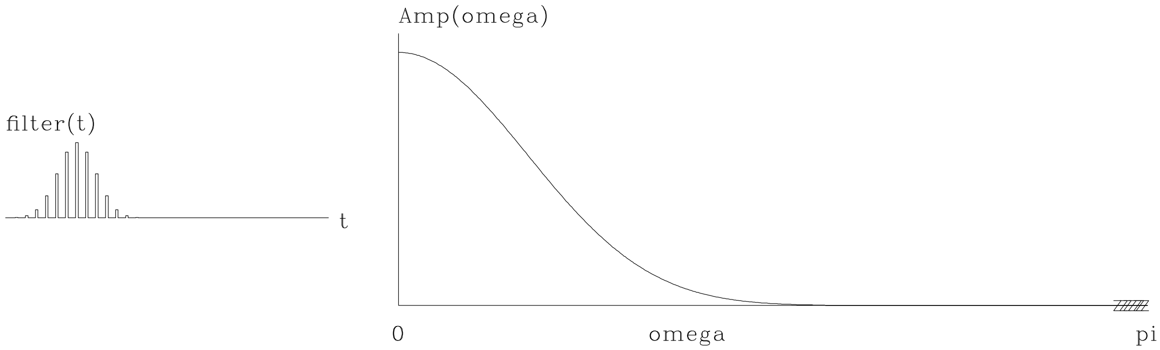


Figure 1.7: A Gaussian approximated by many powers of $(1+Z)$. cs-gauss [NR]

to the same Gaussian but with a different $t_0$. A mathematical theorem says that almost any polynomial raised to the $N$-th power yields a Gaussian.

In seismology we generally fail to observe the **zero frequency**. Thus the idealized seismic pulse cannot be a Gaussian. An analytic waveform of longstanding popularity in seismology is the second derivative of a Gaussian, also known as a "**Ricker wavelet**." Starting from the Gaussian and multiplying by $(1-Z)^2 = 1-2Z+Z^2$ produces this old, favorite wavelet, shown in Figure 1.8.

## 1.4 FT AS AN INVERTIBLE MATRIX

Happily, Fourier sums are exactly invertible: given the output, the input can be quickly found. Because signals can be transformed to the frequency domain, manipulated there, and then

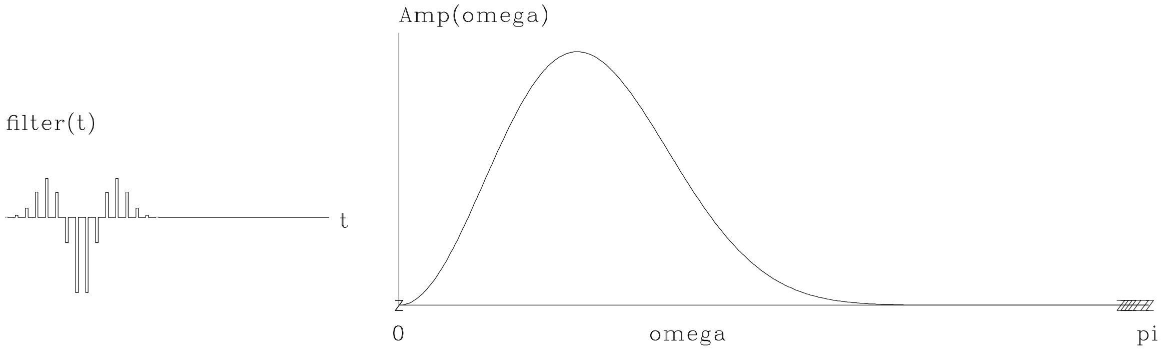


Figure 1.8: Ricker wavelet. $\boxed{\text{cs-ricker}}$ [NR]

returned to the time domain, convolution and correlation can be done faster. Time derivatives can also be computed with more accuracy in the frequency domain than in the time domain. Signals can be shifted a fraction of the time sample, and they can be shifted back again exactly. In this chapter we will see how many operations we associate with the time domain can often be done better in the frequency domain. We will also examine some two-dimensional Fourier transforms.

A **Fourier sum** may be written

$$B(\omega) \quad = \quad \sum_t b_t \, e^{i\omega t} \quad = \quad \sum_t b_t \, Z^t \tag{1.24}$$

where the complex value $Z$ is related to the real frequency $\omega$ by $Z = e^{i\omega}$. This Fourier sum is a way of building a continuous function of $\omega$ from discrete signal values $b_t$ in the time domain. Here we specify both time and frequency domains by a set of points. Begin with an example of a signal that is nonzero at four successive instants, $(b_0, b_1, b_2, b_3)$. The transform is

$$B(\omega) \quad = \quad b_0 + b_1 Z + b_2 Z^2 + b_3 Z^3 \tag{1.25}$$

The evaluation of this polynomial can be organized as a matrix times a vector, such as

$$\begin{bmatrix} B_0 \\ B_1 \\ B_2 \\ B_3 \end{bmatrix} \quad = \quad \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W & W^2 & W^3 \\ 1 & W^2 & W^4 & W^6 \\ 1 & W^3 & W^6 & W^9 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} \tag{1.26}$$

Observe that the top row of the matrix evaluates the polynomial at $Z = 1$, a point where also $\omega = 0$. The second row evaluates $B_1 = B(Z = W = e^{i\omega_0})$, where $\omega_0$ is some base frequency. The third row evaluates the Fourier transform for $2\omega_0$, and the bottom row for $3\omega_0$. The matrix could have more than four rows for more frequencies and more columns for more time points. I have made the matrix square in order to show you next how we can find the inverse matrix. The size of the matrix in (1.26) is $N = 4$. If we choose the base frequency $\omega_0$ and hence $W$

correctly, the inverse matrix will be

$$
\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} \quad = \quad 1/N \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1/W & 1/W^2 & 1/W^3 \\ 1 & 1/W^2 & 1/W^4 & 1/W^6 \\ 1 & 1/W^3 & 1/W^6 & 1/W^9 \end{bmatrix} \begin{bmatrix} B_0 \\ B_1 \\ B_2 \\ B_3 \end{bmatrix} \tag{1.27}
$$

Multiplying the matrix of (1.27) with that of (1.26), we first see that the diagonals are +1 as desired. To have the off diagonals vanish, we need various sums, such as $1+W+W^2+W^3$ and $1+W^2+W^4+W^6$, to vanish. Every element ($W^6$, for example, or $1/W^9$) is a unit vector in the complex plane. In order for the sums of the unit vectors to vanish, we must ensure that the vectors pull symmetrically away from the origin. A uniform distribution of directions meets this requirement. In other words, $W$ should be the $N$-th root of unity, i.e.,

$$
W \quad = \quad \sqrt[N]{1} \quad = \quad e^{2\pi i/N} \tag{1.28}
$$

The lowest frequency is zero, corresponding to the top row of (1.26). The next-to-the-lowest frequency we find by setting $W$ in (1.28) to $Z = e^{i\omega_0}$. So $\omega_0 = 2\pi/N$; and for (1.27) to be inverse to (1.26), the frequencies required are

$$
\omega_k \quad = \quad \frac{(0,1,2,\ldots,N-1)\,2\pi}{N} \tag{1.29}
$$

### 1.4.1 The Nyquist frequency

The highest frequency in equation (1.29), $\omega = 2\pi(N-1)/N$, is almost $2\pi$. This frequency is twice as high as the Nyquist frequency $\omega = \pi$. The **Nyquist frequency** is normally thought of as the "highest possible" frequency, because $e^{i\pi t}$, for integer $t$, plots as $(\cdots,1,-1,1,-1,1,-1,\cdots)$. The double Nyquist frequency function, $e^{i2\pi t}$, for integer $t$, plots as $(\cdots,1,1,1,1,1,\cdots)$. So this frequency above the highest frequency is really zero frequency! We need to recall that $B(\omega) = B(\omega - 2\pi)$. Thus, all the frequencies near the upper end of the range equation (1.29) are really small negative frequencies. Negative frequencies on the interval $(-\pi,0)$ were moved to interval $(\pi,2\pi)$ by the matrix form of Fourier summation.

A picture of the Fourier transform matrix of equation (1.26) is shown in Figure 1.9. Notice the Nyquist frequency is the center row and center column of each matrix.

### 1.4.2 Convolution in one domain...

$Z$-transforms taught us that a multiplication of polynomials is a convolution of their coefficients. Interpreting $Z = e^{i\omega}$ as having numerical values for real numerical values of $\omega$, we have the idea that

| |
|---|
| Convolution in the time domain is multiplication in the frequency domain. |

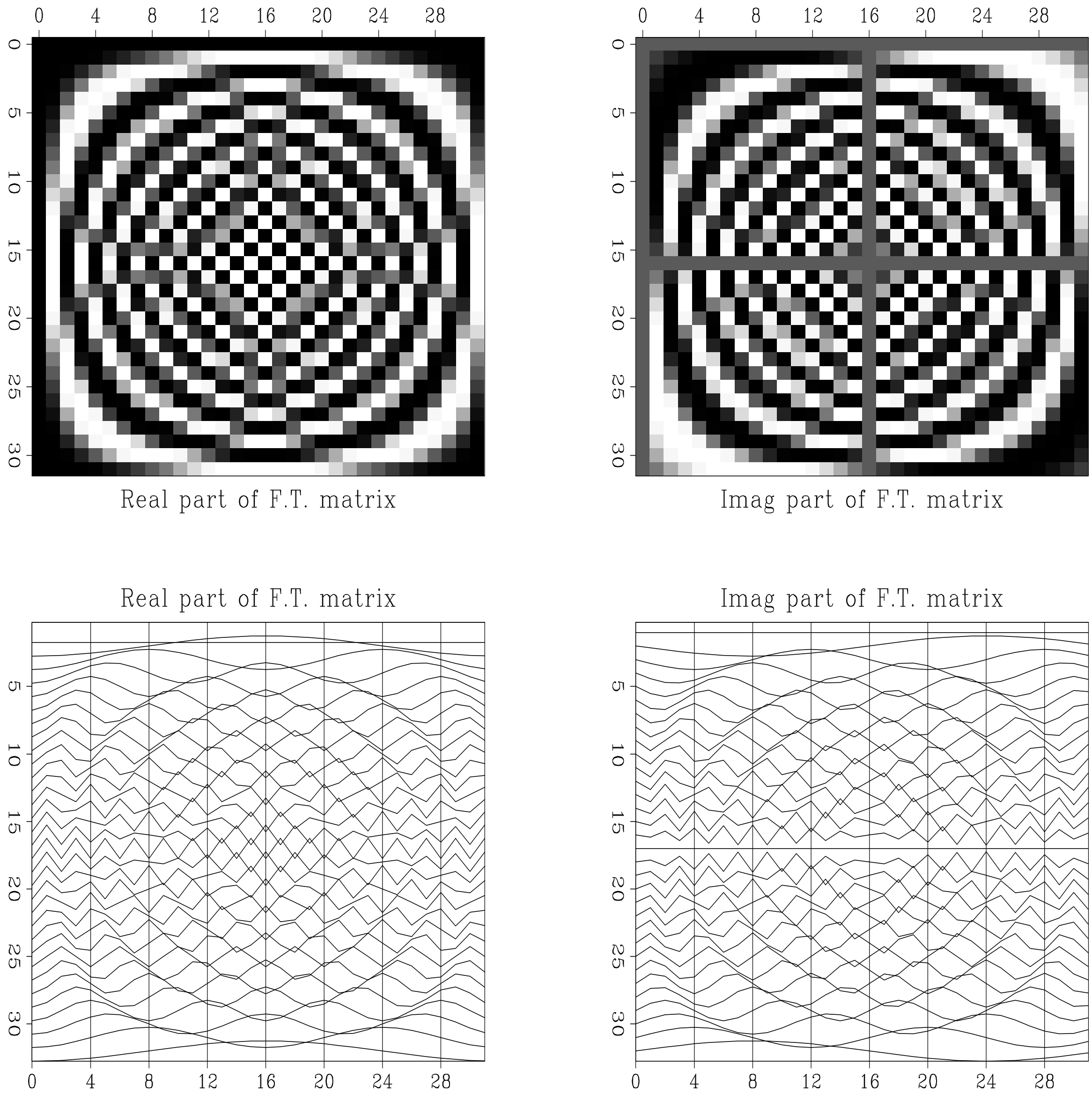


Figure 1.9: Two different graphical means of showing the real and imaginary parts of the Fourier transform matrix of size $32 \times 32$. cs-matrix [ER]

Expressing Fourier transformation as a matrix, we see that except for a choice of sign, Fourier transform is essentially the same process as as inverse fourier transform. This creates an underlying symmetry between the time domain and the frequency domain. We therefore deduce the symmetrical principle that

| |
|---|
| Convolution in the frequency domain is multiplication in the time domain. |

### 1.4.3 Inverse Z-transform

Fourier analysis is widely used in mathematics, physics, and engineering as a **Fourier integral** transformation pair:

$$B(\omega) \quad = \quad \int_{-\infty}^{+\infty} b(t) e^{i\omega t} \, dt \tag{1.30}$$

$$\bar{b}(t) \quad = \quad \int_{-\infty}^{+\infty} B(\omega) e^{-i\omega t} \, d\omega \tag{1.31}$$

These integrals correspond to the sums we are working with here except for some minor details. Books in electrical engineering redefine $e^{i\omega t}$ as $e^{-i\omega t}$. That is like switching $\omega$ to $-\omega$. Instead, we have chosen the **sign convention** of physics, which is better for wave-propagation studies (as explained in IEI). The infinite limits on the integrals result from expressing the **Nyquist frequency** in **radian**s/second as $\pi/\Delta t$. Thus, as $\Delta t$ tends to zero, the **Fourier sum** tends to the integral. It can be shown that if a scaling divisor of $2\pi$ is introduced into either (1.30) or (1.31), then $b(t)$ will equal $\bar{b}(t)$.

**EXERCISES:**

1 Let $B(Z) = 1 + Z + Z^2 + Z^3 + Z^4$. Graph the coefficients of $B(Z)$ as a function of the powers of $Z$. Graph the coefficients of $[B(Z)]^2$.

2 As $\omega$ moves from zero to positive frequencies, where is $Z$ and which way does it rotate around the unit circle, clockwise or counterclockwise?

3 Identify locations on the unit circle of the following frequencies: (1) the zero frequency, (2) the Nyquist frequency, (3) negative frequencies, and (4) a frequency sampled at 10 points per wavelength.

4 Sketch the amplitude spectrum of Figure 1.8 from 0 to $4\pi$.

## 1.5 SYMMETRIES

Next we examine odd/even **symmetries** to see how they are affected in Fourier transform. The even part $e_t$ of a signal $b_t$ is defined as

$$e_t \quad = \quad \frac{b_t + b_{-t}}{2} \tag{1.32}$$

The odd part is

$$o_t \quad = \quad \frac{b_t - b_{-t}}{2} \tag{1.33}$$

By adding (1.32) and (1.33), we see that a function is the sum of its even and odd parts:

$$b_t \quad = \quad e_t + o_t \tag{1.34}$$

Consider a simple, real, even signal such as $(b_{-1}, b_0, b_1) = (1, 0, 1)$. Its transform $Z + 1/Z = e^{i\omega} + e^{-i\omega} = 2\cos\omega$ is an even function of $\omega$, since $\cos\omega = \cos(-\omega)$.

Consider the real, odd signal $(b_{-1}, b_0, b_1) = (-1, 0, 1)$. Its transform $Z - 1/Z = 2i\sin\omega$ is imaginary and odd, since $\sin\omega = -\sin(-\omega)$.

Likewise, the transform of the imaginary even function $(i, 0, i)$ is the imaginary even function $i2\cos\omega$. Finally, the transform of the imaginary odd function $(-i, 0, i)$ is real and odd.

Let $r$ and $i$ refer to real and imaginary, $e$ and $o$ to even and odd, and lower-case and upper-case letters to time and frequency functions. A summary of the symmetries of Fourier transform is shown in Figure 1.10.

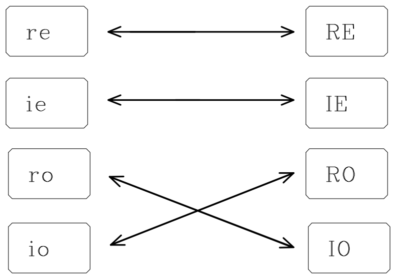


Figure 1.10: Odd functions swap real and imaginary. Even functions do not get mixed up with complex numbers. cs-reRE [NR]

More elaborate signals can be made by adding together the three-point functions we have considered. Since sums of even functions are even, and so on, the diagram in Figure 1.10 applies to all signals. An arbitrary signal is made from these four parts only, i.e., the function has the form $b_t = (\mathrm{re} + \mathrm{ro})_t + i(\mathrm{ie} + \mathrm{io})_t$. On transformation of $b_t$, each of the four individual parts transforms according to the table.

Most "industry standard" methods of Fourier transform set the zero frequency as the first element in the vector array holding the transformed signal, as implied by equation (1.26). This is a little inconvenient, as we saw a few pages back. The Nyquist frequency is then the first point past the middle of the even-length array, and the negative frequencies lie beyond. Figure 1.11 shows an example of an **even function** as it is customarily stored.

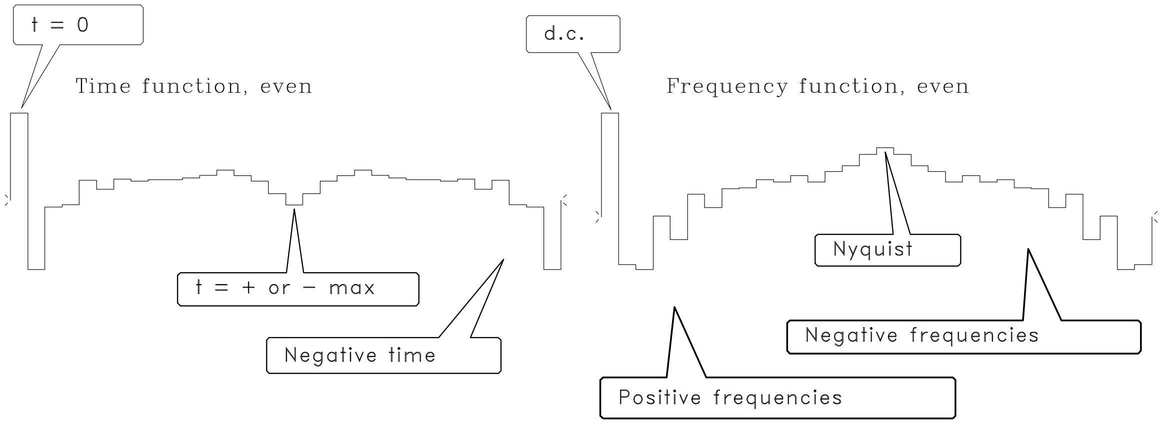


Figure 1.11: Even functions as customarily stored by "industry standard" FT programs. cs-even [NR]

**EXERCISES:**

1 You can learn about Fourier transformation by studying math equations in a book. You can also learn by experimenting with a program where you can "draw" a function in the time domain (or frequency domain) and instantly see the result in the other domain. The best way to learn about FT is to combine the above two methods. Spend at least an hour at the web site:

`http://sepwww.stanford.edu/oldsep/hale/FftLab.html`

which contains the interactive Java FT laboratory written by Dave Hale. This program applies equations (1.26) and (1.27).

### 1.5.1 Laying out a mesh

In theoretical work and in programs, the definition $Z = e^{i\omega\Delta t}$ is often simplified to $\Delta t = 1$, leaving us with $Z = e^{i\omega}$. How do we know whether $\omega$ is given in radians per second or radians per sample? We may not invoke a cosine or an exponential unless the argument has no physical dimensions. So where we see $\omega$ without $\Delta t$, we know it is in units of radians per sample.

In practical work, frequency is typically given in cycles or **Hertz**, $f$, rather than radians, $\omega$ (where $\omega = 2\pi f$). Here we will now switch to $f$. We will design a computer **mesh** on a physical object (such as a waveform or a function of space). We often take the mesh to begin at $t = 0$, and continue till the end $t_{\text{max}}$ of the object, so the time range $t_{\text{range}} = t_{\text{max}}$. Then we decide how many points we want to use. This will be the $N$ used in the discrete Fourier-transform program. Dividing the range by the number gives a mesh interval $\Delta t$.

Now let us see what this choice implies in the frequency domain. We customarily take the maximum frequency to be the Nyquist, either $f_{\text{max}} = .5/\Delta t$ Hz or $\omega_{\text{max}} = \pi/\Delta t$ radians/sec. The frequency range $f_{\text{range}}$ goes from $-.5/\Delta t$ to $.5/\Delta t$. In summary:

- $\Delta t \quad = \quad t_{\text{range}}/N$ is time **resolution**.
- $f_{\text{range}} \quad = \quad 1/\Delta t \quad = \quad N/t_{\text{range}}$ is frequency range.
- $\Delta f \quad = \quad f_{\text{range}}/N \quad = \quad 1/t_{\text{range}}$ is frequency **resolution**.

In principle, we can always increase $N$ to refine the calculation. For an experiment of a fixed size $t_{\text{range}}$, notice that increasing $N$ sharpens the time resolution (makes $\Delta t$ smaller) but does not sharpen the frequency resolution $\Delta f$, which remains fixed. Increasing $N$ increases the frequency *range,* but not the frequency *resolution.*

What if we want to increase the frequency resolution? Then we need to choose $t_{\text{range}}$ larger than required to cover our object of interest. Thus we either record data over a larger range, or we assert that such measurements would be zero. Three equations summarize the facts:

$$\Delta t \ f_{\text{range}} = 1 \tag{1.35}$$
$$\Delta f \ t_{\text{range}} = 1 \tag{1.36}$$
$$\Delta f \ \Delta t = \frac{1}{N} \tag{1.37}$$

Increasing *range* in the time domain increases *resolution* in the frequency domain and vice versa. Increasing **resolution** in one domain does not increase **resolution** in the other.

## 1.6 CORRELATION AND SPECTRA

The spectrum of a signal is a positive function of frequency that says how much of each tone is present. The Fourier transform of a spectrum yields an interesting function called an "**autocorrelation**," which measures the similarity of a signal to itself shifted.

### 1.6.1 Spectra in terms of Z-transforms

Let us look at spectra in terms of $Z$-transforms. Let a **spectrum** be denoted $S(\omega)$, where

$$S(\omega) \quad = \quad |B(\omega)|^2 \quad = \quad \overline{B(\omega)}B(\omega) \tag{1.38}$$

Expressing this in terms of a three-point $Z$-transform, we have

$$S(\omega) = (\bar{b}_0 + \bar{b}_1 e^{-i\omega} + \bar{b}_2 e^{-i2\omega})(b_0 + b_1 e^{i\omega} + b_2 e^{i2\omega}) \tag{1.39}$$
$$S(Z) = \left(\bar{b}_0 + \frac{\bar{b}_1}{Z} + \frac{\bar{b}_2}{Z^2}\right)(b_0 + b_1 Z + b_2 Z^2) \tag{1.40}$$
$$S(Z) = \overline{B}\left(\frac{1}{Z}\right) B(Z) \tag{1.41}$$

It is interesting to multiply out the polynomial $\bar{B}(1/Z)$ with $B(Z)$ in order to examine the coefficients of $S(Z)$:

$$
\begin{aligned}
S(Z) &= \frac{\bar{b}_2 b_0}{Z^2} + \frac{(\bar{b}_1 b_0 + \bar{b}_2 b_1)}{Z} + (\bar{b}_0 b_0 + \bar{b}_1 b_1 + \bar{b}_2 b_2) + (\bar{b}_0 b_1 + \bar{b}_1 b_2) Z + \bar{b}_0 b_2 Z^2 \\
S(Z) &= \frac{s_{-2}}{Z^2} + \frac{s_{-1}}{Z} + s_0 + s_1 Z + s_2 Z^2 \qquad (1.42)
\end{aligned}
$$

The coefficient $s_k$ of $Z^k$ is given by

$$s_k = \sum_i \bar{b}_i b_{i+k} \qquad (1.43)$$

Equation (1.43) is the **autocorrelation** formula. The autocorrelation value $s_k$ at lag 10 is $s_{10}$. It is a measure of the similarity of $b_i$ with itself shifted 10 units in time. In the most frequently occurring case, $b_i$ is real; then, by inspection of (1.43), we see that the autocorrelation coefficients are real, and $s_k = s_{-k}$.

Specializing to a real time series gives

$$
\begin{aligned}
S(Z) &= s_0 + s_1 \left( Z + \frac{1}{Z} \right) + s_2 \left( Z^2 + \frac{1}{Z^2} \right) && (1.44) \\
S(Z(\omega)) &= s_0 + s_1 (e^{i\omega} + e^{-i\omega}) + s_2 (e^{i2\omega} + e^{-i2\omega}) && (1.45) \\
S(\omega) &= s_0 + 2 s_1 \cos \omega + 2 s_2 \cos 2\omega && (1.46) \\
S(\omega) &= \sum_k s_k \cos k\omega && (1.47) \\
S(\omega) &= \text{cosine transform of } \; s_k && (1.48)
\end{aligned}
$$

This proves a classic theorem that for real-valued signals can be simply stated as follows:

> For any real signal, the cosine transform of the **autocorrelation** equals the magnitude squared of the Fourier transform.

### 1.6.2 Two ways to compute a spectrum

There are two computationally distinct methods by which we can compute a spectrum: (1) compute all the $s_k$ coefficients from (1.43) and then form the cosine sum (1.47) for each $\omega$; and alternately, (2) evaluate $B(Z)$ for some value of Z on the unit circle, and multiply the resulting number by its complex conjugate. Repeat for many values of $Z$ on the unit circle. When there are more than about twenty lags, method (2) is cheaper, because the fast Fourier transform discussed in chapter 2 can be used.

### 1.6.3 The comb function

Consider a constant function of time. In the frequency domain, it is an impulse at zero frequency. The **comb** function is defined to be zero at alternate time points. Multiply this constant function by the comb function. The resulting signal contains equal amounts of two frequencies; half is zero frequency, and half is Nyquist frequency. We see this in the second row in Figure 1.12, where the Nyquist energy is in the middle of the frequency axis. In the third row, 3 out of 4 points are zeroed by another comb. We now see something like a new Nyquist frequency at half the Nyquist frequency visible on the second row.

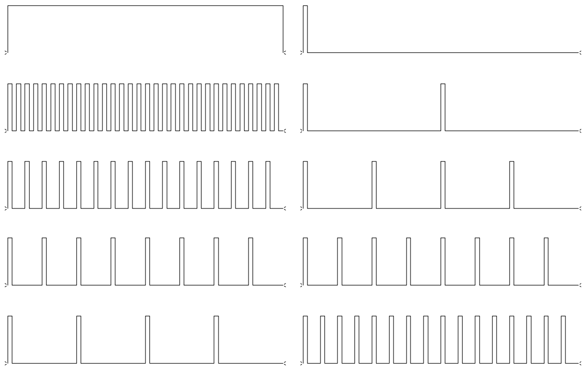

Figure 1.12: A zero-frequency function and its cosine transform. Successive rows show increasingly sparse sampling of the zero-frequency function. cs-comb [NR]

### 1.6.4 Undersampled field data

Figure 1.13 shows a recording of an **airgun** along with its spectrum. The original data is

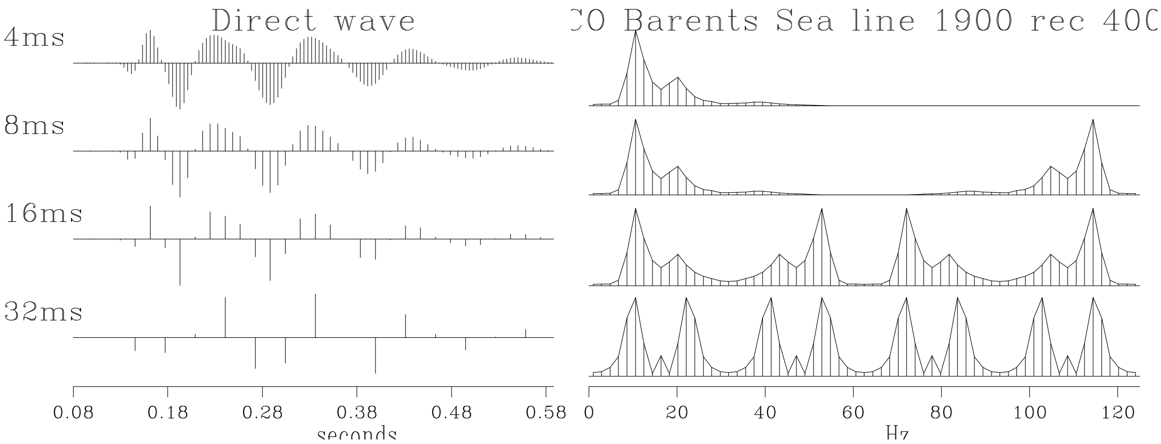


Figure 1.13: Raw data is shown on the top left, of about a half-second duration. Right shows amplitude spectra (magnitude of FT). In successive rows the data is sampled less densely. cs-undersample [ER]

sampled at an interval of 4 milliseconds, which is 250 times per second. Thus, the **Nyquist frequency** $1/(2\Delta t)$ is 125 Hz. Negative frequencies are not shown, since the amplitude spectrum at negative frequency is identical with that at positive frequency. Think of extending the top row of spectra in Figure 1.13 to range from minus 125 Hz to plus 125 Hz. Imagine the even function of frequency centered at zero frequency—we will soon see it. In the second row of the plot, I decimated the data to 8 ms. This drops the Nyquist frequency to 62.5 Hz. Energy that was at $-10$ Hz appears at $125-10$ Hz in the *second* row spectrum. The appearance of what were formerly small negative frequencies near the Nyquist frequency is called "**fold**ing" of the spectrum. In the next row the data is sampled at 16 ms intervals, and in the last row at 32 ms intervals. The 8 ms sampling seems OK, whereas the 32 ms sampling looks poor. Study how the spectrum changes from one row to the next.

The spectrum suffers no visible harm in the drop from 4 ms to 8 ms. The 8 ms data could be used to construct the original 4 ms data by transforming the 8 ms data to the frequency domain, replacing values at frequencies above 125/2 Hz by zero, and then inverse transforming to the time domain.

(Airguns usually have a higher frequency content than we see here. Some high-frequency energy was removed by the recording geometry, and I also removed some when preparing the data.)

### 1.6.5 Common signals

Figure 1.14 shows some common signals and their **autocorrelation**s. Figure 1.15 shows the cosine transforms of the autocorrelations. Cosine transform takes us from time to frequency and it also takes us from frequency to time. Thus, transform pairs in Figure 1.15 are sometimes more comprehensible if you interchange time and frequency. The various signals are given names in the figures, and a description of each follows:

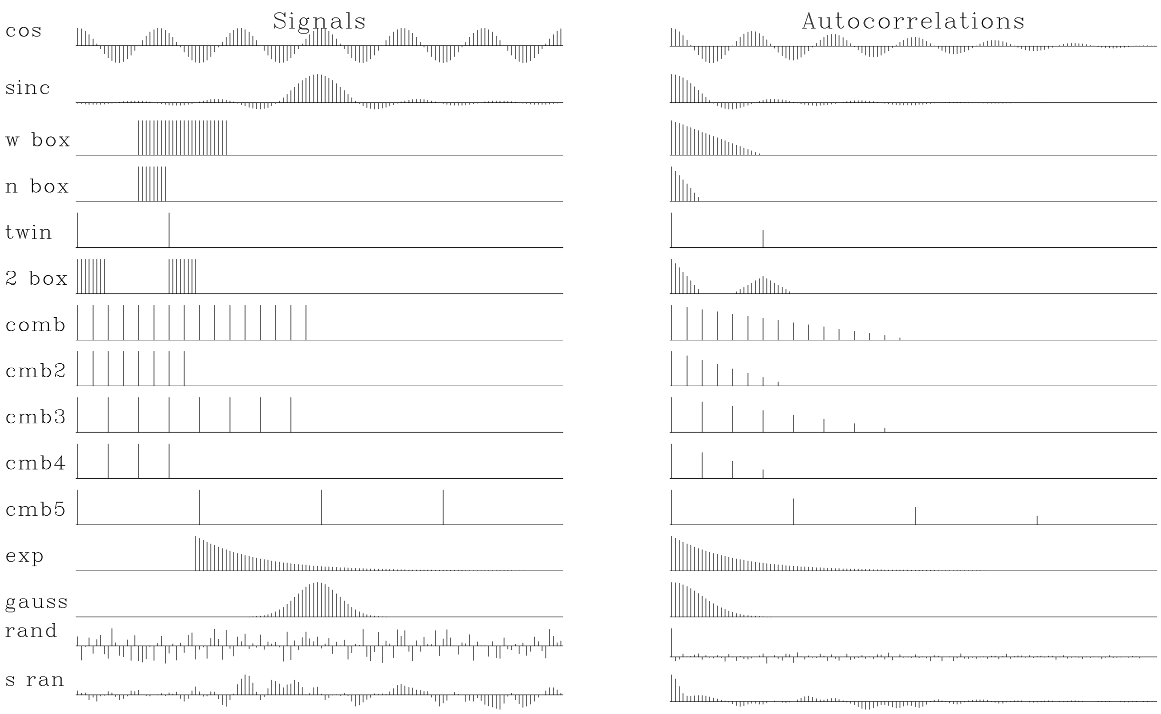


Figure 1.14: Common signals and one side of their autocorrelations. cs-autocor [ER]

**cos** The theoretical spectrum of a sinusoid is an impulse, but the sinusoid was truncated (mul-

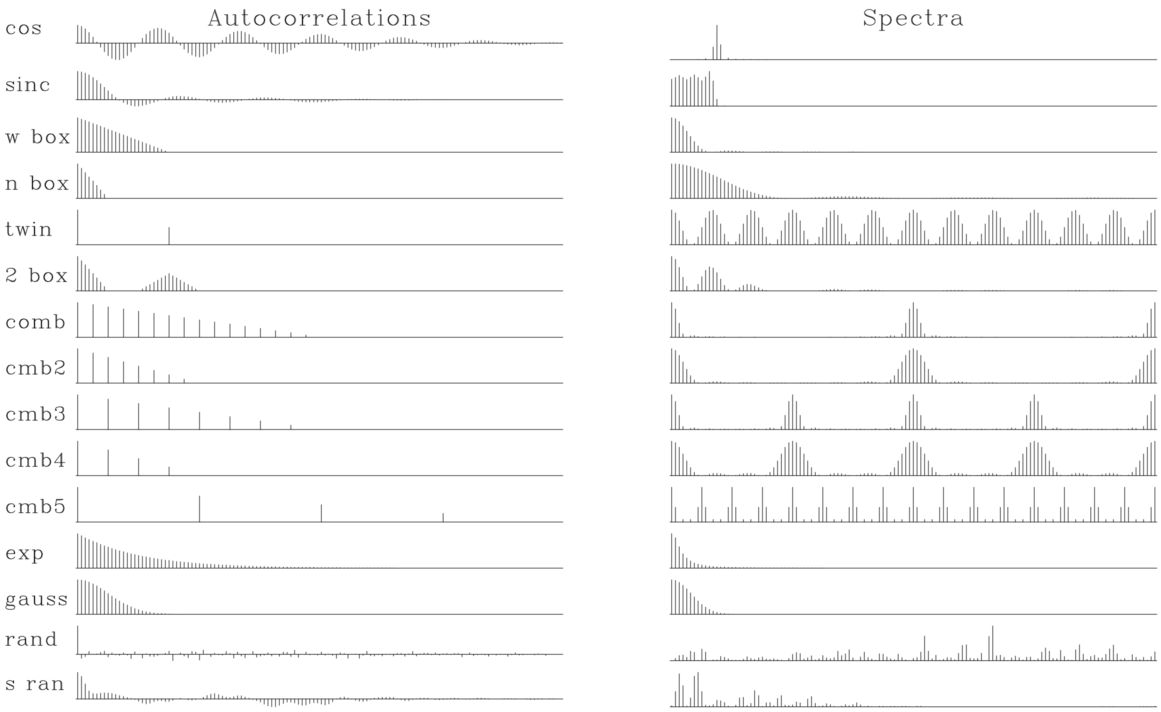


Figure 1.15: Autocorrelations and their cosine transforms, i.e., the (energy) spectra of the common signals. cs-spectra [ER]

tiplied by a rectangle function). The autocorrelation is a sinusoid under a triangle, and its spectrum is a broadened impulse (which can be shown to be a narrow sinc-squared function).

**sinc** The **sinc** function is $\sin(\omega_0 t)/(\omega_0 t)$. Its autocorrelation is another sinc function, and its spectrum is a rectangle function. Here the rectangle is corrupted slightly by "**Gibbs sidelobes**," which result from the time truncation of the original sinc.

**wide box** A wide **rectangle function** has a wide triangle function for an autocorrelation and a narrow sinc-squared spectrum.

**narrow box** A narrow rectangle has a wide sinc-squared spectrum.

**twin** Two pulses.

**2 boxes** Two separated narrow boxes have the spectrum of one of them, but this spectrum is modulated (multiplied) by a sinusoidal function of frequency, where the modulation frequency measures the time separation of the narrow boxes. (An oscillation seen in the frequency domain is sometimes called a "**quefrency**.")

**comb** Fine-toothed-**comb** functions are like rectangle functions with a lower Nyquist frequency. Coarse-toothed-comb functions have a spectrum which is a fine-toothed comb.

**exponential** The autocorrelation of a transient **exponential** function is a **double-sided exponential** function. The spectrum (energy) is a Cauchy function, $1/(\omega^2+\omega_0^2)$. The curious thing about the **Cauchy function** is that the amplitude spectrum diminishes inversely with frequency to the *first* power; hence, over an infinite frequency axis, the function has infinite integral. The sharp edge at the onset of the transient exponential has much high-frequency energy.

**Gauss** The autocorrelation of a **Gaussian** function is another Gaussian, and the spectrum is also a Gaussian.

**random** **Random** numbers have an autocorrelation that is an impulse surrounded by some short grass. The spectrum is positive random numbers.

**smoothed random** Smoothed random numbers are much the same as random numbers, but their spectral bandwidth is limited.

### 1.6.6 Spectral transfer function

Filters are often used to change the spectra of given data. With input $X(Z)$, filters $B(Z)$, and output $Y(Z)$, we have $Y(Z) = B(Z)X(Z)$ and the Fourier conjugate $\overline{Y}(1/Z) = \overline{B}(1/Z)\overline{X}(1/Z)$. Multiplying these two relations together, we get

$$\overline{Y}Y \quad = \quad (\overline{B}B)(\overline{X}X) \tag{1.49}$$

which says that the spectrum of the input times the spectrum of the filter equals the spectrum of the output. Filters are often characterized by the shape of their spectra; this shape is the same as the **spectral ratio** of the output over the input:

$$\overline{B}B \quad = \quad \frac{\overline{Y}Y}{\overline{X}X} \tag{1.50}$$

**EXERCISES:**

1 Suppose a wavelet is made up of complex numbers. Is the autocorrelation relation $s_k = s_{-k}$ true? Is $s_k$ real or complex? Is $S(\omega)$ real or complex?

# Index