Book3 or revised parts of FGDP

Jon F. Claerbout

ABSTRACT

This document is a part of a collection of lectures, some drawn from revising my 1976 book, "Fundamental of Geophysical Data Processing," and some based on SEP research since then.

INTRODUCTION

Last year on my sabbatical leave I began the task of revising my 1976 book, "Fundamentals of Geophysical Data Processing" (FGDP) into a new set of lectures. These lectures are still two to four years from publication, but I thought you might like to see parts in their present form. All chapters were revised in some degree, but some revisions were so drastic that they don't resemble the original book. Such parts are included here. Although the material is nominally tutorial in nature, I find some of the examples are particularly informative, even to experienced researchers.

Chapter 3 about feedback filtering includes many new figures illustrating the Z-plane. I included a triangle smoothing program that I use in my research because it is fast and handles end effects nicely. I included my "favorite wavelet" for seismic modeling. Numerical roundoff was previously a "skeleton in the closet," but here, for the first time, is a simple explanation of how and why such problems occur.

Chapter 4 is about spectrum and phase. There are many new figures relating to Hilbert transform and instantaneous amplitude and frequency. This tutorial recognizes something that I didn't when I wrote FGDP, namely that instantaneous frequency can be stablized in a manner similar to the "expectation of an operator" method of quantum physics. This observation is new to me and may be new to most exploration researchers. Reformulating this old problem in this new way should significantly enhance stability. The explanation of the Kolmogoroff method of spectral factorization is significantly simplified and there are many examples, some unexpected.

Chapter 7 on Resolution (was chapter 4 in FGDP) is now much clearer and contains many interesting examples of Fourier transformations of partially random numbers.

Chapter 9, "The Conjugate Transformation" was missing from FGDP and it explains things I simply did not know in 1976. These are just the things you need to know if you are working on problems that simultaneously involve migration and least squares.

ELECTRONIC BOOK

The biggest improvement can't be seen on a paper report. Over the next decade, I visualize an "electronic book" where "touching" any figure allows the reader to alter the parameters of the program that created the figure. I'm carefully saving all the programs that generated the figures. Already a large number, perhaps half, of the figures are generated by interactive programs. These figures are really nothing more than particular initialization states of the interactive programs. It hasn't escaped my attention that the sum total of all the words and programs and data that make up my lectures has considerably fewer bytes than one of the laser disks that contains popular music and sells for ten dollars.

Chapter 3

CAUSALITY AND FEEDBACK

All physical systems share the property that they do not respond before they are excited. Thus the impulse response of any physical system is a one-sided time function (it vanishes before $t = 0$). In system theory such a filter function is called *realizable*. In wave propagation this property is associated with *causality* in that no wave may begin to arrive before it is transmitted. The lag-time point $t = 0$ plays a peculiar and an important role. For this reason, many subtle matters will be much more clearly understood with sampled time than with continuous time. When a filter responds at and after lag time $t = 0$, we say the filter is realizable or causal. The word *causal* is appropriate in physics where stress may cause (practically) instantaneous strain and vice versa, but one should return to the less pretentious words *realizable* or *one-sided* when using filter theory to describe economic or social systems where simultaneity is quite different from cause and effect.

Much of what we will be doing in this chapter is the inverse to convolution. We will be pulling apart convolution into its causal and anticausal parts. You could of course simply divide in the Fourier domain and remain ignorant of the convolutional mechanisms. But the inverse of convolution deserves careful attention because of the importance of petroleum found in thin beds—so thin that they are at and below the limit of resolution of the seismic method. Also, inverse filtering methodology is closely related to ordinary differential equations, which provide a concise parametrization for complicated objects like seismograms. Additionally, we will be looking at analytic expressions for some widely used filters. These are best parameterized and most efficiently implemented in the time domain. Parenthetically, the topic of my other book "Imaging the Earth's Interior" is essentially a downward extrapolation of wave fields. This process, being directed in the depth direction, is mathematically equivalent to being directed (causal) in the time domain. So a study of causality and feedback has broader application than just inverse convolution.

First a short review: The Z-transform of an arbitrary, time-discretized function x_t is defined by

$$X(Z) \quad = \quad \cdots + x_{-2} Z^{-2} + x_{-1} Z^{-1} + x_0 + x_1 Z + x_2 Z^2 + \cdots \qquad (3.1)$$

In chapter 1 we interpreted (3.1) as a Fourier transform where $Z = e^{i\omega}$. But it isn't necessary for Z to take on numerical values for the ideas of convolution and correlation to be useful. In chapter 2 we defined Z to be the unit delay operator. Then Z^2 delays two time units. Expressions like $X(Z)\,B(Z)$ and $X(Z)\,B(1/Z)$ are useful because they imply convolution and cross-correlation of the time-domain coefficients.

3.1 INVERSE FILTERS

To understand causal filters better, we now take up the task of undoing what a causal filter has done. Suppose that output y_t of a filter b_t is known, but the input x_t is unknown. See Figure 3.1.

FIG. 3.1. (FGDP Fig. 2-1) Sometimes the input to a filter is unknown.

This is the problem with any transducer/recorder system. For example, the output of a seismometer is a wiggly line on a piece of paper from which the seismologist may determine the displacement, velocity, or acceleration of the ground. To undo the filtering operation of the filter $B(Z)$, we will try to find another filter $A(Z)$ as shown in Figure 3.2.

FIG. 3.2. (FGDP Fig 2-2) The filter $A(Z)$ is inverse to the filter $B(Z)$.

To solve for the coefficients of the filter $A(Z)$, identify coefficients of powers of Z in $B(Z)A(Z) = 1$. Letting $B(Z)$ be a three-term filter, this is

$$(a_0 + a_1 Z + a_2 Z^2 + a_3 Z^3 + \cdots)(b_0 + b_1 Z + b_2 Z^2) \quad = \quad 1 \tag{3.2}$$

The coefficients of $Z^0, Z^1, Z^2, \ldots$ in (3.2) are

$$1 = a_0 b_0 \tag{3.3}$$

$$0 = a_1 b_0 + a_0 b_1 \tag{3.4}$$

$$0 = a_2 b_0 + a_1 b_1 + a_0 b_2 \tag{3.5}$$

$$0 = a_3 b_0 + a_2 b_1 + a_1 b_2$$

$$0 = a_4 b_0 + a_3 b_1 + a_2 b_2$$

$$= \cdots\cdots\cdots\cdots$$

$$0 = a_k b_0 + a_{k-1} b_1 + a_{k-2} b_2 \tag{3.6}$$

From (3.3) you can get a_0 from b_0. From (3.4) you can get a_1 from a_0 and the b_k. From (3.5) you can get a_2 from a_1, a_0, and the b_k. Likewise, in the general case a_k can be found from a_{k-1}, a_{k-2}, and the b_k. Specifically, from (3.6) the a_k can be determined recursively by

$$a_k \quad = \quad \frac{-\sum_{i=1}^{N_b} a_{k-i} b_i}{b_0} \tag{3.7}$$

where in this case $N_b = 2$.

3.2 LEAKY INTEGRATION

The convolution equation (2.8)

$$y_k \quad = \quad \sum_{i=0} x_{k-i} b_i$$

says that the present output is created entirely from present and past values of the *input*. Now we will include past values of the *output*. The simplest example is numerical integration such as

$$y_t \quad = \quad y_{t-1} + x_t \tag{3.8}$$

Notice that when $x_t = (0,0,0,1,0,0,\cdots)$ then $y_t = (0,0,0,1,1,1,1,\cdots)$ which shows that the integral of an impulse function is a step function.

A kind of deliberately imperfect integration used in numerical work is called "leaky" integration. The name comes from electrical circuits work where the voltage on a capacitor is the integral of the current. In real life, some of the current leaks away. An equation to model leaky integration is

$$y_t \quad = \quad \rho\, y_{t-1} + x_t \tag{3.9}$$

where ρ is a constant that is slightly less than plus one. Let us see what Z-transform equation is implied by (3.9). Move the y terms to the left.

$$y_t - \rho\, y_{t-1} \quad = \quad x_t \tag{3.10}$$

Given the Z-transform equation

$$(1 - \rho Z)\, Y(Z) \quad = \quad X(Z) \tag{3.11}$$

notice that (3.10) can be derived from (3.11) by finding the coefficient of Z^t. So we can say that the output $Y(Z)$ is derived from the input $X(Z)$ by the polynomial division

$$Y(Z) \quad = \quad \frac{X(Z)}{(1 - \rho Z)} \tag{3.12}$$

Ordinary integration has a Fourier response $1/(-i\omega)$ that blows up at $\omega = 0$. Leaky integration smooths off the infinity. Notice that if ρ were greater than unity, the output of (3.9) would grow with time instead of decaying. Considering both positive and negative values of ρ, stability is associated with $|\rho| < 1$. The value of $Z = Z_p$ for which the the denominator in (3.12) vanishes is called a pole, It is $Z_p = 1/\rho$. So stability is associated with the pole locations $|Z_p| > 1$.

Figure 3.3 shows the filter of equation (3.12) and its spectrum.

3.2.1 Plots

There will be a number of plots similar to Figure 3.3. A pole location is denoted by a "p" and a zero location by a "z". I decided to display the complex ω-plane instead of the Z-plane. Thus real frequencies run along the horizontal axis instead of around the unit circle. For poles and zeros the horizontal axis is $\Re\omega_0$ and the vertical axis increases nonlinearly with $-\Im\omega_0$. At the risk of confusion, I decided to display the plot of $|B(\omega)|$ versus ω on top

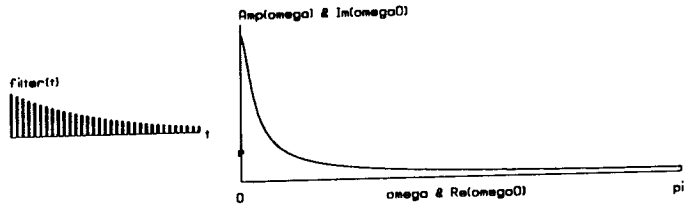

FIG. 3.3. (caus-leak) Left is the impulse response of leaky integration. Right is the amplitude $1/|1-\rho Z|$ in the Fourier domain.

of the complex ω-plane. This enables you to correlate the pole and zero locations to the spectrum. In Figure 3.3 moving the "p" closer to the horizontal axis would cause a slower time decay and a sharper frequency function. A few additional details about the imaginary axis are found on page 18.

EXERCISES:

1 A simple feedback operation is $y_t = (1-\epsilon)y_{t-1} + x_t$. Give a closed form expression for the output y_t if x_t is an impulse. Estimate the decay time τ of your solution (the time it takes for y_t to drop to $e^{-1}y_0$)? For small ϵ, say = 0.1, .001, or 0.0001, what is τ?

3.3 NARROW-BAND FILTERS

A simple way to represent a sinusoid by Z transforms is

$$B(Z) = \frac{1}{1-Ze^{-i\omega_0}} = 1 + Ze^{-i\omega_0} + Z^2e^{-i2\omega_0} + \cdots \tag{3.13}$$

The time function b_t is $(1, e^{-i\omega_0}, e^{-i2\omega_0}, \cdots)$, which is a complex sinusoidal function of time. It is not quite the sinusoid $e^{-i\omega_0 t}$, because b_t is "turned on" at $t = 0$ whereas $e^{-i\omega_0 t}$ is nonzero at negative time. The time function b_t is shown on the left in Figure 3.4. On the right is a graphical attempt to plot the impulse function of dividing by zero at $\omega = \omega_0$.

Next let us look at a damped case like leaky integration. Let $Z_p = e^{i\omega_0}/\rho$ and $|\rho| < 1$. Then $1/Z_p = \rho e^{-i\omega_0}$. Then define

$$B(Z) = \frac{1}{A(Z)} = \frac{1}{1-Z/Z_p} = 1 + \frac{Z}{Z_p} + \left(\frac{Z}{Z_p}\right)^2 + \cdots \tag{3.14}$$

$$B(Z) = 1 + Z\rho e^{-i\omega_0} + Z^2\rho^2 e^{-i2\omega_0} + \cdots \tag{3.15}$$

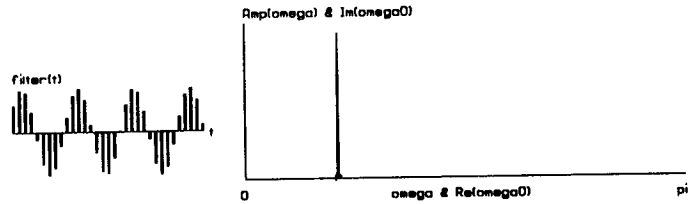

FIG. 3.4. (caus-sinus) A pole on the real axis gives an impulse function at that frequency and a sinusoidal function in time.

The time function b_t is zero before $t = 0$ and is $\rho^t e^{-i\omega_0 t}$ after $t = 0$. It is a damped sinusoidal function with amplitude decreasing with time as ρ^t. You can readily recognize this as an exponential decay

$$\rho^t = e^{t\log\rho} \approx e^{-t(1-\rho)} \tag{3.16}$$

where the approximation is best for values of ρ near unity.

The wavelet b_t is complex. To have a real time function we need another pole at the negative frequency, say $\overline{Z_p}$. So the composite denominator is

$$A(Z) = \left(1-\frac{Z}{Z_p}\right)\left(1-\frac{Z}{\overline{Z_p}}\right) = 1 - Z\rho\cos\omega_0 + \rho^2Z^2 \tag{3.17}$$

Multiplying the two poles together as we did for roots on page 23 results in the plots in Figure 3.5. The time response is on the left and the amplitude $|B(Z)|$ versus ω graph is on the right.

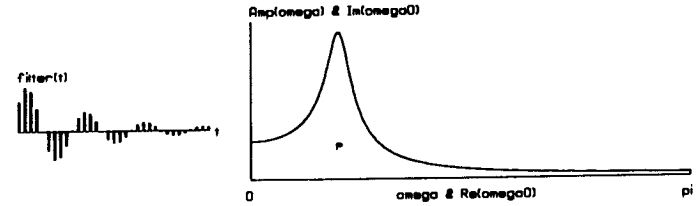

FIG. 3.5. (caus-dsinus) A damped sinusoidal function of time transforms to a pole near the real ω axis, i.e. just outside the unit circle in the Z-plane.

EXERCISES:

1 How far from the unit circle are the poles of $1/(1 - .1Z + .9Z^2)$? What is the decay time of the filter and its resonant frequency?

3.4 POLYNOMIAL DIVISION

Convolution with the coefficients b_t of (3.15), is a narrow-banded filtering operation. If the pole is chosen very close to the unit circle, the filter bandpass becomes very narrow and the coefficients of $B(Z)$ drop off very slowly. There is a method of narrow-band filtering that is much quicker than convolution with b_t. This method is polynomial division by $A(Z)$. We have for the output $Y(Z)$

$$Y(Z) \quad = \quad B(Z)\, X(Z) \quad = \quad \frac{X(Z)}{A(Z)} \tag{3.18}$$

Multiply both sides of (3.18) by $A(Z)$

$$X(Z) \quad = \quad Y(Z)\, A(Z) \tag{3.19}$$

For definiteness, let us suppose the x_t and y_t vanish before $t = 0$. Now identify coefficients of successive powers of Z to get

$$\begin{aligned} x_0 &= y_0 a_0 \\ x_1 &= y_1 a_0 + y_0 a_1 \\ x_2 &= y_2 a_0 + y_1 a_1 + y_0 a_2 \\ x_3 &= y_3 a_0 + y_2 a_1 + y_1 a_2 \\ x_4 &= y_4 a_0 + y_3 a_1 + y_2 a_2 \\ &= \cdots\cdots\cdots\cdots \end{aligned} \tag{3.20}$$

Let N_a be the number of coefficients in $A(Z)$. The k-th equation (where $k > N_a$) is

$$y_k a_0 + \sum_{i=1}^{N_a} y_{k-i} a_i \quad = \quad x_k \tag{3.21}$$

Solving for y_k we get

$$y_k \quad = \quad \frac{x_k - \sum_{i=1}^{N_a} y_{k-i} a_i}{a_0} \tag{3.22}$$

Equation (3.22) may be used to solve for y_k once $y_{k-1}, y_{k-2}, \cdots$ are known. Thus the solution is called *recursive*. The value of N_a is only 2 whereas N_b is technically infinite and would in practice need to be approximated by a fairly large value. So the feedback operation (3.22) is much quicker than convolving with the filter $B(Z) = 1/A(Z)$. A program for the task is:

```
#       polynomial division filter:     Y(Z) = X(Z) / A(Z)
#
subroutine polydiv( na, aa, nx, xx, ny, yy )
integer na,     # number of coefficients of denominator
        nx,     # length of the input function
        ny      # length of the output function
real    aa(na), # denominator recursive filter
        xx(nx), # input trace
        yy(ny)  # output trace, as long as input trace.

        integer ia, iy
```



```
        do iy = 1, ny
                if( iy <= nx)
                        yy(iy) = xx(iy)
                else
                        yy(iy) = 0.
        do iy = 1, na-1                                         # lead-in terms
                do ia= 2, iy
                        yy(iy) = yy(iy) - aa(ia) * yy(iy-ia+1)
        do iy = na, ny                                          # steady state
                do ia= 2, na
                        yy(iy) = yy(iy) - aa(ia) * yy(iy-ia+1)
        do iy = 1, ny
                yy(iy) = yy(iy) / aa(1)
        return; end
# modified since last tested.
```

3.5 SPECTRUM OF A POLE

Finally, let us find the analytical expression for the spectrum of a filter made from a single pole. Taking that pole to be $Z_p = e^{i\omega_0}/\rho$, we have

$$A(Z) \quad = \quad 1 - \frac{Z}{Z_p} \quad = \quad 1 - \frac{\rho}{e^{i\omega_0}}\, e^{i\omega} \quad = \quad 1 - \rho e^{i(\omega-\omega_0)} \tag{3.23}$$

The complex conjugate is

$$\overline{A}\left(\frac{1}{Z}\right) \quad = \quad 1 - \rho e^{-i(\omega-\omega_0)} \tag{3.24}$$

The spectrum of a pole filter is the inverse of

$$\begin{aligned} \overline{A}\left(\frac{1}{Z}\right) A(Z) &= \left(1 - \rho e^{-i(\omega-\omega_0)}\right)\left(1 - \rho e^{i(\omega-\omega_0)}\right) \\ &= 1 + \rho^2 - \rho\left(e^{-i(\omega-\omega_0)} + e^{i(\omega-\omega_0)}\right) \\ &= 1 + \rho^2 - 2\rho\cos(\omega - \omega_0) \\ &= 1 + \rho^2 - 2\rho + 2\rho[1 - \cos(\omega - \omega_0)] \\ &= (1-\rho)^2 + 4\rho\sin^2\frac{\omega - \omega_0}{2} \end{aligned} \tag{3.25}$$

With the definition of a small $\epsilon = 1 - \rho > 0$, inverting gives

$$\overline{B}\left(\frac{1}{Z}\right) B(Z) \quad \approx \quad \frac{1}{\epsilon^2 + 4\sin^2\frac{\omega-\omega_0}{2}} \tag{3.26}$$

Specialising to frequencies close to ω_0 where the denominator is small and the function is large gives

$$\overline{B}\left(\frac{1}{Z}\right) B(Z) \quad \approx \quad \frac{1}{\epsilon^2 + (\omega - \omega_0)^2} \tag{3.27}$$

This is called a narrowband filter because in the Fourier domain the function is large only in a narrow band of frequencies. Setting $\overline{B}B$ to half its peak value of $1/\epsilon^2$ we find a half bandwidth of $\Delta\omega/2 = |\omega - \omega_0| = \epsilon$. The damping time constant Δt of the damped sinusoid b_t is shown in the exercises to be $\Delta t = 1/\epsilon$.

Naturally we want a real time function, so we multiply the filter $1/(1 - Z/Z_p)$ times $1/(1 - Z/\bar{Z}_p)$. Thus the resulting time function is real (because conjugate poles are like conjugate roots on page 22). The spectrum of the conjugate factor $1/(1 - Z/\bar{Z}_p)$ is like (3.26) except that ω_0 replaced by $-\omega_0$. Multiplying the response (3.26) by itself with $-\omega_0$ yields the symmetric function of ω displayed on the right in Figure 3.6.

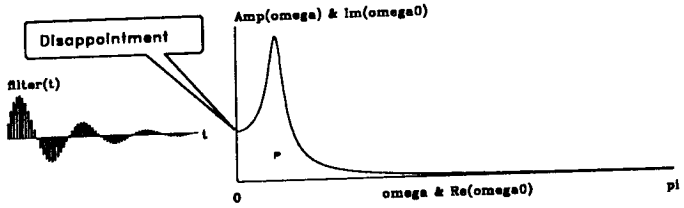

FIG. 3.6. (caus-disappoint) A pole near the real axis gives a damped sinusoid in time on the left. On the right is $1/|A(\omega)|$ for ω real. Poles are typically fairly close to $\omega = 0$ because seismic data is typically sampled fairly densely (compared to Nyquist).

You might be disappointed if you intend to apply the filter of Figure 3.6 as a narrow-band filter. Notice that the passband is rather unsymmetric and that it passes the zero frequency rather strongly. Equation (3.26) is symmetric about ω_0, but taking the product with its image about $-\omega$ has spoiled the symmetry. So in practice you might like to add a zero at zero frequency and the Nyquist frequency, i.e., $(1-Z)(1+Z)$, as shown in Figure 3.7.

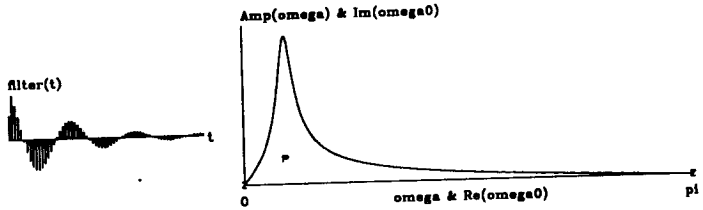

FIG. 3.7. (caus-symdsin) Poles at $\pm\omega_0$ and a root at $\omega = 0$ and another root at $\omega = \pi$.

EXERCISES:

1 Figure 3.6 shows a bump around ω_0 that is not very symmetric looking because it is the *product* of equation (3.26) with a frequency reversed copy. Consider the *sum* $[1/(1 - Z/Z_p)] + [1/(1 - Z/\bar{Z}_p)]$. It the time filter real? Where are its poles and zeros? How will its amplitude as a function of frequency compare with Figure 3.6? Will the bump be more symmetric looking?

3.6 RATIONAL FILTERS

A general model for filtering includes both convolution (numerator Z-transforms) and feedback filtering (denominator Z-transforms).

$$Y(Z) \quad = \quad \frac{B(Z)}{A(Z)} X(Z) \tag{3.28}$$

There are a variety of ways to implement equation (3.28) in a computer. You could do the polynomial division $X(Z)/A(Z)$ first and then multiply (convolve) with $B(Z)$ or you could do the multiplication first and the division later. Alternately, you can do them simultaneously if you identify coefficients of $A(Z)Y(Z) = B(Z)X(Z)$ and solve for recursive equations as we did for (3.22).

The rational model is more powerful than either a purely numerator model or a denominator model because like its numerator part, it can easily destroy any frequency totally, and like its denominator part, it can easily enhance any frequency without limit. Finite difference solutions of differential equations often appear as rational filters.

EXERCISES:

1 Consider equation (3.28). What time domain recurrance (analogous to equation (3.22)) is implied?

3.7 SMOOTHING WITH BOX AND TRIANGLE

Simple smoothing is a common application of filtering. A smoothing filter is one with all positive coefficients. On the time axis, smoothing is often done with a single-pole damped exponential function. But on space axes people generally prefer a symmetrical function. Of course you could filter twice with an exponential, once each way (and I plan to explain my `trisso` program somewhere else in these notes). An attractive alternative is smoothing with a rectangle or triangle function. When the function width is chosen to be long, then the computation time can be unnecessarily large, but recursion can shorten it immensely.

3.7.1 Smoothing with a rectangle

The inverse of any polynomial reverberates forever, although it might drop off fast enough for any practical need. On the other hand, a rational filter can suddenly drop to zero and stay there. Let us look at a popular rational filter, the "box car."

$$\frac{1 - Z^5}{1 - Z} \quad = \quad 1 + Z + Z^2 + Z^3 + Z^4 \tag{3.29}$$

The filter (3.29) gives a moving average under a *rectangular* window. This is a basic smoothing filter. A clever way to apply it is to move the rectangle by adding a new value at one end while dropping an old value from the other end. The clever way is formalized by the

polynomial division algorithm, which can be simplified since so many coefficients are either one or zero. It boils down to the following program which is so fast it is almost free!

```
subroutine boxconv( k, nx, xx, ny, yy)          # convolve with a box
# inputs:       nx,  x(i), i=1,nx       the data
#               k                       the box length
# output:       ny, y(i),i=1,ny         smoothed data (ny>nx)
integer nx, ny, k, i
real xx(nx), yy(ny), bb(4000)
ny = nx+k-1
#if( ny >= 4000) call ErrorExit('dimension error in boxconv')
do i=1,ny
        bb(i) = 0.
bb(1) = xx(1)
do i=2,nx
                bb(i) = bb(i-1) + xx(1)
do i=nx+1, ny
                bb(i) = bb(i-1)
do i=1,k
                yy(i) = bb(i)
do i=k+1, ny
                yy(i) = bb(i) - bb(i-k)
do i=1, ny
                yy(i) = yy(i) / k
return; end
```

Let us examine the pole and zero locations in equation (3.29). The denominator vanishes at $Z = 1$, so the filter has a pole at zero frequency. Smoothing something is like boosting frequencies near the zero frequency. The numerator vanishes at the five roots of unity, i.e. $Z = e^{i2\pi n/5}$. These five locations are uniformly spaced around the unit circle. Any sinusoid at exactly one of these frequencies is exactly destroyed by this filter because such a sinusoid has an integer number of wavelengths under the boxcar. An exception is the zero frequency where the root at $Z = 1$ is canceled by a pole at the same location. The cancellation is why the right-hand side terminates at the fourth power—there is no infinite series of higher powers.

3.7.2 Smoothing with a triangle. Edge effects.

If you wish to smooth under a *triangular* window, you simply square equation (3.29). This amounts to applying the rectangle smoothing program twice. Convolving a rectangle function with itself many times yields a result that mathematically tends towards a Gaussian function. Despite the sharp corner on the top of the triangle function, it has a shape that is remarkably similar to a Gaussian, as you can see by looking at Figure 2.8 or Figure 7.2. I used to smooth with damped exponentials, but I switched to trangles when I encountered several examples where the exponential tails were not small enough.

With filtering, *end effects* can be a nuisance. Filtering increases the length of the data, but people generally want to keep input and output the same length (for various practical reasons). This is particularly so when filtering a space axis. You could simply abandon the points off the ends. But I like to *fold* them back in. The result is like a wave reflected by a zero-slope end condition. An advantage of the folding is that a constant valued signal is unchanged by the smoothing. This is desirable since a smoothing filter is a low-pass filter which naturally should pass the lowest frequency $\omega = 0$ without distortion. My program is:

```
subroutine triconv( k, nx,  xx)
# Convolve triangle weighting with data.
# inputs:       xx(i),i=1,nx    the data
#               k               width (points) of rectangle to convolve twice
# outputs:      xx(i),i=1,nx
integer k,nx, i,np,nq
real xx(nx), pp(4000), qq(4000), rr(4000)
call boxconv( k, nx, xx, np, pp);
call boxconv( k, np, pp, nq, qq);
do i=1,nx
        rr(i) = qq(i+k-1)
do i=1,k-1                              # fold back near end
        rr(i) = rr(i) + qq(k-i)
do i=1,k-1                              # fold back far end
        rr(nx-i+1) = rr(nx-i+1) + qq(nx+(k-1)+i)
do i=1,nx
        xx(i) = rr(i)
return; end
```

The above program smooths impulses into triangles, except near the boundaries. What happens there is shown in Figure 3.8. Note that at the boundary, there is necessarily only

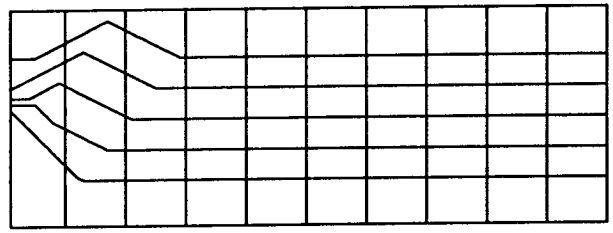

FIG. 3.8. (cause-triangle) Edge effects when smoothing an impulse with a triangle function. Inputs are spikes at various distances from the edge.

half a triangle, but it is twice as tall.

EXERCISES:

1 Identify the pole and zero locations of the box function, equation (3.29) in the Z-plane.

2 The Fourier transform of a rectangle function is $\sin(\alpha t)/\alpha t$, also known as a "sinc" function. In terms of α, how wide is the rectangle function?

3 Express $Z^{-2} + Z^{-1} + 1 + Z + Z^2$ in the ω domain. This is a discrete representation of a rectangle function. Identify the similarity and differences with the sinc function.

4 Explain the second from bottom signal in Figure 3.8.

5 The program `triconv` is intended to pass the zero frequency with no change. But it has a bug in the unusual case where k $\geq$ nx. Identify the bug. Recode the program to work for all values of k.

3.8 INSTABILITY

Consider the example $B(Z) = 1 - Z/2$. The inverse

$$A(Z) = \frac{1}{1-\frac{Z}{2}} = 1 + \frac{Z}{2} + \frac{Z^2}{4} + \frac{Z^3}{8} + \cdots \tag{3.30}$$

can be found by a variety of familiar techniques such as (1) polynomial division, i.e., equations like (3.3) to (3.6), (2) Taylor's power series formula, or (3) the binomial theorem.

In equation (3.30) we see there are an infinite number of filter coefficients but they drop off rapidly in size so that approximation in a computer presents no difficulty.

We are not so lucky with the filter $B(Z) = 1 - 2Z$. Here we have

$$A(Z) = \frac{1}{1-2Z} = 1 + 2Z + 4Z^2 + 8Z^3 + 16Z^4 + 32Z^5 + \cdots \tag{3.31}$$

The coefficients of this series increases without bound. The outputs of the filter $A(Z)$ depend infinitely strongly on inputs of the infinitely distant past. [Recall that the present output of $A(Z)$ is a_0 times the present input x_1 plus a_1 times the previous input x_{t-1}, etc., so a_n represents memory on n time units earlier.] This example shows that some filters $B(Z)$ will not have useful inverses $A(Z)$ determined by polynomial division. Two sample plots of divergence are in Figure 3.9 and in Figure 3.10.

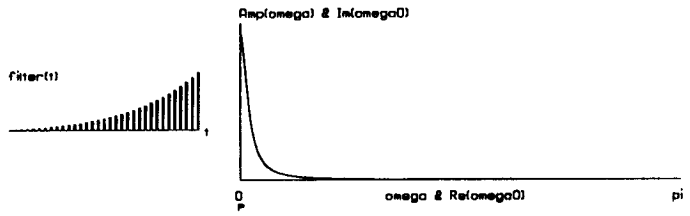

FIG. 3.9. (caus-exp) Here we see the growing time function of a pole inside the unit circle at zero frequency. The time function is a growing exponential.

3.8.1 Anticausality

Can anything at all be done if there is a root inside the circle? An answer is suggested by the example

$$\frac{1}{1-2Z} = -\frac{1}{2Z}\frac{1}{1-\frac{1}{2Z}} = -\frac{1}{2Z}\left[1 + \frac{1}{2Z} + \frac{1}{(2Z)^2} + \cdots\right] \tag{3.32}$$

Equation (3.32) is a series expansion in $1/Z$, in other words, a series about infinity. It converges from $|Z| = \infty$ all the way in to a circle of radius $|Z| = 1/2$. This means that the inverse converges on the unit circle where it must, if the coefficients are to be bounded. In

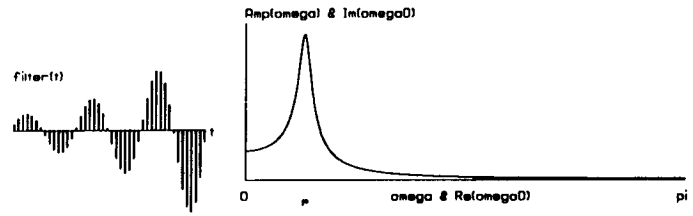

FIG. 3.10. (caus-gsin) Here we see the growing time function of a pole inside the unit circle at a positive frequency. If it doesn't appear to be reaching towards infinity it is because the plotting program rescales any plot to lie within bounds. The essential feature is that the function is largest at the last point plotted. It is increasing indefinitely.

terms of filters it means that the inverse filter must be one of those filters that responds to future inputs and hence is not physically realisable but may be used in computer simulation. The spectra plotted in Figure 3.9 and in Figure 3.10 apply to the anticausal expansion. Obviously the causal expansion, which is unbounded, has an infinite spectrum.

Stability for wavelets with complex coefficients is this: If the solution value Z_0 of $B(Z_0) = 0$ lies inside the unit circle in the complex plane, then $1/B(Z)$ will have coefficients which blow up; and if the root lies outside the unit circle, then the inverse $1/B(Z)$ will be bounded.

On page 21 we saw that a polynomial $B(Z)$ of degree N may be factored into N subsystems and that the ordering of subsystems is unimportant. Suppose we have factored $B(Z)$ and some of its roots lie outside the unit circle and some lie inside. We first invert the outside roots with equation (3.30) and then invert the inside roots with equation (3.32). If there are any roots exactly on the unit circle then you have a special case in which you can try either inverse, but neither might give a satisfactory result in practice. Implied zero division is nature's way of telling you that what you are trying to do can't be done that way (if at all).

FIG. 3.11. (FGDP Fig 2-3) Factoring the polynomial $B(Z)$ breaks the filter into many two-term filters. Each one should have a bounded inverse.

3.9 THE UNIT DISK

To prove that one equals zero take an infinite series such as 1, −1, +1, −1, +1,... and group the terms in two different ways, and add them in this way:

$$\begin{aligned} (1-1) + (1-1) + (1-1) + \cdots &= 1 + (-1+1) + (-1+1) + \cdots \\ 0 + 0 + 0 + \cdots &= 1 + 0 + 0 + \cdots \\ 0 &= 1 \end{aligned}$$

Of course this does not prove that one equals zero: it proves that care must be taken with infinite series. Next, take another infinite series in which the terms may be regrouped into any order without fear of paradoxical results. Let a pie be divided into halves. Let one of the halves be divided in two, giving two quarters. Then let one of the two quarters be divided into two eighths. Continue likewise. The infinite series is 1/2, 1/4, 1/8, 1/16,.... No matter how the pieces are rearranged, they should all fit back into the pie plate and exactly fill it.

The danger of infinite series is not that they have an infinite number of terms but that they may sum to infinity. Safety is assured if the sum of the absolute values of the terms is finite. Such a series is called *absolutely convergent*.

3.9.1 Boundedness

Given different numerical values for Z we can ask whether $X(Z)$ is finite or infinite. Numerical values of Z of particular interest are $Z = +1$, $Z = -1$, and all those complex values of Z which are unit magnitude, say, $|Z| = 1$ or $Z = e^{i\omega}$ where ω is the real Fourier transform variable. Then the Z-transform is a Fourier sum.

Our attention can be restricted to time functions u_t with a finite amount of energy by demanding that $U(Z)$ be finite for all values of Z on the unit circle $|Z| = 1$. Filter functions are always restricted to have finite energy.

3.9.2 Causality and the unit disk

The most straightforward way to say that a filter is causal is to say that its time domain coefficients vanish before zero lag, that is $u_t = 0$ for $t < 0$. Another way to say it is to say that $U(Z)$ is finite for $Z = 0$. At $Z = 0$ the Z-transform would be infinite if the coefficients u_{-1}, u_{-2}, etc. were not zero.

For a causal function, each term in $U(Z)$ will be smaller if Z is taken inside the disk $|Z| < 1$ rather than on the rim $|Z| = 1$. Thus convergence at $Z = 0$ and on the circle $|Z| = 1$ implies convergence everywhere inside the unit disk. So boundedness combined with causality means convergence in the unit disk.

Convergence at $Z = 0$ but not on the circle $|Z| = 1$ would refer to a causal function with infinite energy, a case of no practical interest. What kind of function converges on the circle, at $Z = \infty$, but not at $Z = 0$? What function converges at all three places, $Z = 0$, $Z = \infty$, and $|Z| = 1$?

We have seen that the filter $1/(1 - 2Z)$ can be expanded into powers of Z in (at least) two different ways. These are

$$\begin{aligned} \frac{1}{1-2Z} &= 1 + 2Z + 4Z^2 + 8Z^3 + \cdots \\ &= -\frac{1}{2Z}\frac{1}{1-\frac{1}{2Z}} = -\frac{1}{2Z}\left[1 + \frac{1}{2Z} + \frac{1}{4Z^2} + \cdots\right] \end{aligned}$$

Which of these two infinite series converges depends on the numerical value of Z. For $|Z| = 1$ the first series diverges, but the second converges. So the only acceptable filter is *anticausal*. Is a series expansion unique? It is if it converges. Complex-variable theory proves this.

3.9.3 Minimum phase defined

Let b_t denote a filter. Then a_t is its inverse filter if the convolution of a_t with b_t is an impulse function. Filters are said to be inverse to one another if their Fourier transforms are inverse to one another. Thus in terms of Z-transforms, an inverse is simple defined by $A(Z) = 1/B(Z)$. Whether the filter $A(Z)$ is causal depends on whether it is finite everywhere inside the unit disk, or really on whether $B(Z)$ vanishes *anywhere* inside the disk. For example, $B(Z) = 1 - 2Z$ vanishes at $Z = 1/2$. There $A(Z) = 1/B(Z)$ must be infinite, that is to say, the series $A(Z)$ must be nonconvergent at $Z = 1/2$. Thus, as we have just seen, a_t is noncausal. A most interesting case, called *minimum phase*, occurs when both a filter $B(Z)$ and its inverse are causal. In summary:

causal:	$\|B(Z)\| < \infty$	for $\|Z\| \leq 1$
causal inverse:	$\|1/B(Z)\| < \infty$	for $\|Z\| \leq 1$
minimum phase:	both above conditions	

The reason for the interesting words "minimum phase" is in the next chapter.

3.9.4 Laurent Expansion

In the general case, then, one must factor $B(Z)$ into two parts: $B(Z) = B_{out}(Z)B_{in}(Z)$ where B_{out} contains roots outside the unit circle and B_{in} contains the roots inside. Then the inverse of B_{out} is expressed as a Taylor series about the origin and the inverse of B_{in} is expressed as a Taylor series about infinity. The final expression for $1/B(Z)$ is called a Laurent expansion for $1/B(Z)$, and it converges on a ring surrounding the unit circle. Cases with zeros exactly on the unit circle present special problems. For example differentiation $(1 - Z)$ is the inverse of integration, but the converse is not true because of the additive constant of integration.

EXERCISES:

1 Find the filter which is inverse to $(2 - 5Z + 2Z^2)$. You may just drop higher-order powers of Z, but an exact expression for the coefficients of any power of Z is preferred. (Partial fractions is a useful, though not necessary, technique). Sketch the impulse response.

2 Describe a general method for determining $A(Z)$ and $B(Z)$ from a Taylor series of $B(Z)/A(Z) = C_0+C_1Z+C_2Z^2+\cdots+C_\infty Z^\infty$ where $B(Z)$ and $A(Z)$ are polynomials of unknown degree n and m, respectively. Work out the case $C(Z) = \frac{1}{2}-\frac{3}{4}Z-\frac{3}{8}Z^2-\frac{3}{16}Z^3-\frac{3}{32}Z^4-\cdots$. Don't try this problem unless you are quite familiar with determinants. [HINT: Identify coefficients of $B(Z) = A(Z)C(Z)$.]

3.10 MECHANICAL INTERPRETATION

Because of the stringent conditions on minimum phase wavelets you might wonder whether they can exist in nature. A simple mechanical example should convince you that minimum phase wavelets are plentiful: The stress (pressure) in a material may be given by x_t, and the strain (volume charge) may be represented by y_t. Physically you can specify the stress or the strain and nature gives you the other. So obviously the stress in a material may be expressed as a linear combination of present and past strains. Likewise, the strain may be deduced from present and past stresses. Mathematically this means that the filter which relates stress to strain and vice versa has all poles and zeros outside the unit circle.

3.11 INTRODUCTION TO ALL-PASS FILTERS

An all-pass filter is a filter whose spectral magnitude is unity. Given an input $X(Z)$ and an output $Y(Z)$, then the spectra of the two are the same, i.e. $\bar{X}(1/Z)X(Z) = \bar{Y}(1/Z)Y(Z)$. The existence of an infinitude of all-pass filters tells us that an infinitude of wavelets can have the same spectrum. Wave propagation without absorption is modeled by all-pass filters. All-pass filters are a kind of waveform distortion that we will seek to eliminate in a later section on spectral factorization.

The simplest example of an all-pass filter is the delay operator $Z = e^{i\omega}$ itself. Its phase as a function of ω is simply ω.

A less trivial example of phase distortion can be constructed from a single root Z_r where Z_r is an arbitrary complex number. Notice that the ratio of any complex number, say $\rho e^{i\phi} = x + iy$, to its complex conjugate $\rho e^{-i\phi} = x - iy$, has unit magnitude $|e^{i2\phi}|$. So, given a minimum-phase filter $B(\omega)$ we can take its conjugate and make an all-pass filter $P(Z)$ from the ratio $P(Z) = \overline{B(\omega)}/B(\omega)$. A simple case is

$$B(\omega) = 1 - \frac{Z}{Z_r} \tag{3.33}$$

$$\overline{B(\omega)} = 1 - \frac{1}{Z\overline{Z_r}} \tag{3.34}$$

The all-pass filter of this ratio is not causal because of the presence of $1/Z$ in $\overline{B}$. We can repair that by multiplying by another all-pass operator, namely Z. The resulting causal

all-pass filter is

$$P(Z) = Z\,\frac{1-\frac{1}{Z\overline{Z}_r}}{1-\frac{Z}{Z_r}} = \frac{Z-\frac{1}{\overline{Z}_r}}{1-\frac{Z}{Z_r}} \tag{3.35}$$

The denominator tells us we have a pole at Z_r. Let this be $Z_r = e^{i\omega_0}/\rho$. The numerator vanishes at

$$Z = Z_0 = \frac{1}{\overline{Z}_r} = \rho\, e^{i\omega_0} \tag{3.36}$$

In conclusion, the pole is outside the unit circle and the zero is inside. They face one another across the circle at the phase angle ω_0.

But this all-pass filter would output a complex time function unless we handle the negative frequencies the same as the positive ones. So, as on page 23 the filter (3.35) is multiplied by another like itself with Z_r replaced by $\overline{Z}_r$, i.e., with ω_0 replaced by $-\omega_0$. The result is shown in Figure 3.12.

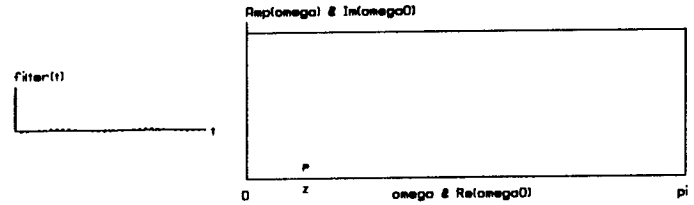

FIG. 3.12. (caus-allpass) All-pass filter. The pole and zero are at equal logarithmic distance from the unit circle.

A general form for an all-pass filter is $P(Z) = Z^N\overline{A}(1/Z)/A(Z)$ where $A(Z)$ is an arbitrary minimum phase filter. That this form is valid may be verified by checking that $\overline{P}(1/Z)P(Z) = 1$.

EXERCISES:

1 Verify that $\overline{P}(1/Z)P(Z) = 1$ for the general form of an all-pass filter $P(Z) = Z^N\overline{A}(1/Z)/A(Z)$

3.12 NOTCH FILTER

In some applications it is desired to reject a very narrow frequency band leaving the rest of the spectrum little changed. The most common example is 60-Hz noise from power lines. Another is low frequency ground roll. Such filters can easily be made by a slight variation fiel on the all-pass filter. In the all-pass filter the pole and zero have an equal (logarithmic) relative distance from the unit circle. All we need to do is to put the zero closer to the

circle. In fact, there is no reason why we should not put the zero right on the circle. Then the frequency at which the zero is located is exactly canceled from the spectrum of input data.

Narrowband filters and sharp cutoff filters should be used with caution. An ever-present floor penalty for such filters is that they do not decay rapidly in time. Although this may not present problems in some applications, it will do so in others. Obviously, if the data collection duration is shorter or comparable to the impulse response of the narrowband filter, then the transient effects of starting up the experiment will not have time to die out. Likewise, the notch should not be too narrow in a 60-Hz rejection filter. Even a bandpass filter (easier to implement with fast Fourier transform than with a few poles) has a certain decay rate in the time domain which may be too slow for some experiments. In radar and in reflection seismology the importance of a signal is not related to its strength. Late-arriving echoes may be very weak, but they contain information not found in earlier echoes. If too sharp a frequency characteristic is used, then filter resonance from early strong arrivals may not have decayed sufficiently by the time that the weak late echoes arrive.

A curious thing about narrow band reject filters is that when you look at their impulse responses you always see the frequency being rejected! For example, look at Figure 3.13. The filter consists of a large spike (which contains all frequencies) and then a sinusoidal tail of opposite polarity of the frequency being rejected.

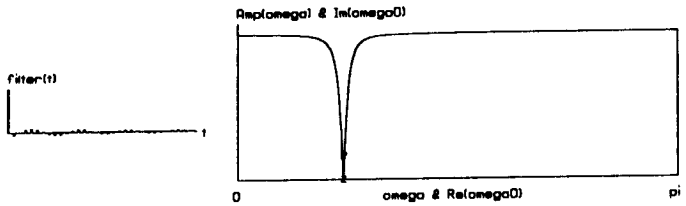

FIG. 3.13. (cause-notch) A zero on the real frequency axis and a pole just above it gives a notch filter, i.e. the zeroed frequency is rejected while other frequencies are little changed.

A filter with a wider notch is shown in Figure 3.14. The vertical axis in the complex frequency plane in the figure is not exactly $\Im\omega_0$. It is something like the logarithm of $\Im\omega_0$. The logarithm is not precisely appropriate either because zeros may be exactly on the unit circle. I couldn't devise an ideal theory for scaling $\Im\omega_0$, so after some experimentation, I chose $\Im\omega_0 = -(1+y^2)/(1-y^2)$ where y is the vertical position in a window of vertical range $0 < y < 1$. Because of the minus sign, the outside of the unit circle is above the $\Re\omega_0$ axis and the inside of the unit circle is below it.

EXERCISES:

1 Find a three-term real feedback filter to pass 59-61 Hz on data which is sampled at 500 points/sec. Where are the poles? What is the decay time of the filter?

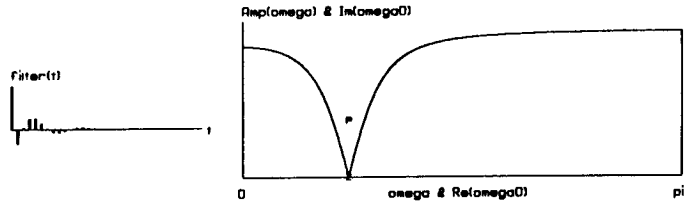

FIG. 3.14. (caus-bnotch) A notch filter where the notch has been broadened by moving the pole further away from the zero.

3.13 MY FAVORITE WAVELET

I'll describe my favorite wavelet for seismic modeling. Of course the ideal wavelet is an impulse, but the wavelet I describe is intended to mimic real life.

I use some zeros at high frequency to force continuity in the time domain. I use a zero at the origin to suppress d.c. I like to simulate the suppression of low frequency ground roll so I put another zero not at the origin, but at a low frequency. Theory demands a conjugate pair for this zero so effectively there are three roots to suppress low frequencies. I use some poles to skew the passband towards low frequencies. These poles also remove some of the oscillation caused by the three zeros. (Each zero is like a derivative and causes another lobe in the wavelet). There is a trade off between having a long low frequency tail and having a rapid spectral rise just above the ground roll. The trade off is adjustable by repositioning the lower pole. The time domain wavelet shows its high frequencies first and its low frequencies only later. See Figure 3.15. I like this wavelet better than the Ricker wavelet in Figure 2.9.

3.14 PATHOLOGICAL EXAMPLES

As we reach the end of this chapter on poles and feedback filtering you might be inclined to conclude that all is well if poles are outside the unit circle and that they may even come quite close. And, if we'll accept anticausal filtering, poles can be inside the unit circle too.

Reality is more difficult. We can have big trouble from just a modest clustering of poles at a moderate distance from the unit circle. This is shown in Figure 3.16

How can Figure 3.16 be explained? We have not considered the implications of finite precision machine arithmetic. When single word precision becomes a noticeable problem the obvious path is to choose double precision. But considering that most geophysical data has a precision less than one part in a hundred, and only rarely do we see precision of one part in a thousand, the failure of single word precision arithmetic, about one part in 10^6,

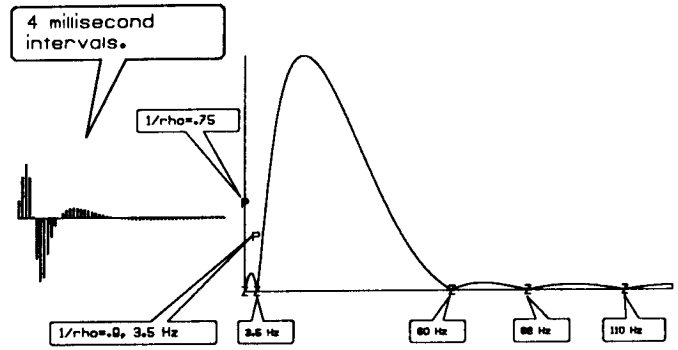

FIG. 3.15. (caus-favorite) My favorite wavelet for seismic modeling. I use some zeros at high frequency to force continuity in the time domain. I use a zero at the origin to suppress d.c. I like to simulate the suppression of low frequency ground roll so I put another zero not at the origin, but at a low frequency. Theory demands a conjugate pair for this zero so effectively there are three roots to suppress low frequencies. I used some poles to skew the passband towards low frequencies. They also remove some of the oscillation caused by the three zeros. (Each zero is like a derivative and causes another lobe in the wavelet). There is a trade off between having a long low frequency tail and having a rapid spectral rise just above the ground roll. The trade off is adjustable by repositioning the lower pole. The time domain wavelet shows its high frequencies first and its low frequencies only later. I like this wavelet better than a Ricker wavelet.

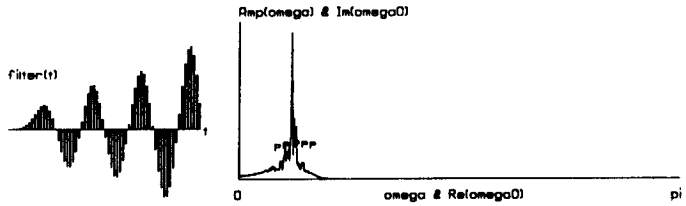

FIG. 3.16. (caus-path) A pathological failure when poles cluster too much requiring more precision than single-word precision.

is more sign of conceptual failure than of numerical precision inadequacy.

Recall that the spectrum in Figure 3.5 multiplied by itself about six or seven times, once for each pole, should give that in Figure 3.16. Such repetitive multiplication increases the dynamic range (range between largest and smallest amplitude as a function of frequency) above the 10^6 of single precision arithmetic. Notice that the peak spectral values in Figure 3.16 come from the *minimum* values of the denominator. But the denominator will be unable to get itself properly small, if the precision of its terms is not adequate to allow them to extinguish one another.

I might know a way partly around this precision problem based on a reflection coefficient modeling of a layered earth, described in a later chapter. Since this case doesn't arise commonly in practice, we'll set it aside for now. But take note of other pathologies coming up in the next chapter. Pathological cases often show most clearly things that can't work.

Chapter 4

SPECTRUM AND PHASE

In this chapter we will learn

- about 90° phase shift and the Hilbert transform
- about phase delay, group delay, and beating
- where the name "minimum phase" came from
- what it means about energy delay
- spectral factorization, i.e. finding a minimum phase wavelet to fit any spectrum

4.1 HILBERT TRANSFORM

Chapter 1 explains that many plots in this book have various interpretations. Nominally, the plot pairs represent cosine transforms of real even functions. Since the functions are even, their negative halves are not shown. An alternate interpretation of the plot pairs is that one signal is real and causal. This is illustrated in full detail in Figure 4.1. Half of the values in Figure 4.1 convey no information. These are the zero values at negative time, and the negative frequencies of the FT. In other words, the right half of Figure 4.1 is redundant. So it is generally not shown. Likewise the bottom plot which is the imaginary part is generally not shown because it is derivable in a simple way from given information. Computation of the unseen imaginary part is called *Hilbert transform*. Here we will investigate details and applications of the Hilbert transform. There are surprisingly many, including 90° phase-shift filtering, envelope functions, the instantaneous frequency function, and relating amplitude spectra to phase spectra.

Ordinarily a function is specified entirely in the time domain or entirely in the frequency domain. The Fourier transform then finds it in the other domain. In a mathematical view, the Hilbert transform arises when half the information is in the time domain and the other half in the frequency domain. (Algebraically speaking, any fractional part could be given in either domain).

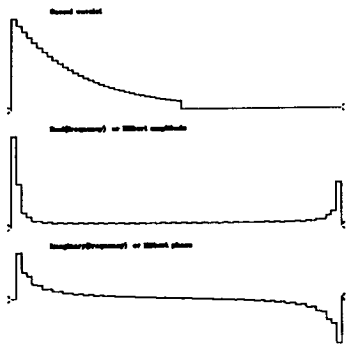

FIG. 4.1. (spec-intro) Both positive and negative times and frequencies of a real causal time function (top) and real (mid) and imaginary (bottom) parts of its FT.

4.1.1 A Z-transform view of Hilbert transformation

Let x_t be an even function of t. Recalling that $Z^{-n} + Z^n = 2\cos\omega n$, we have

$$X(Z) = \cdots + x_1 Z^{-1} + x_0 + x_1 Z + x_2 Z^2 + \cdots \quad (4.1)$$

$$X(Z) = x_0 + 2x_1\cos\omega + 2x_2\cos 2\omega + \cdots \quad (4.2)$$

Now make up a new function $Y(Z)$ by replacing cosine by sine in (4.2).

$$Y(Z) = 2x_1\sin\omega + 2x_2\sin 2\omega + \cdots \quad (4.3)$$

Recalling that $Z = \cos\omega + i\sin\omega$ we see that all the negative powers of Z cancel from $X(Z) + iY(Z)$ giving a causal $C(Z)$.

$$C(Z) = \frac{1}{2}[X(Z) + iY(Z)] = \frac{1}{2}x_0 + x_1 Z + x_2 Z^2 + \cdots \quad (4.4)$$

So for plot pairs the causal time function is c_t, the real part of the FT is equation (4.2), and the imaginary part not usually shown is given by equation (4.3).

4.1.2 The quadrature filter

In the last subsection we had a causal time function and we switched cosines and sines in the frequency domain. Here we do the same with time and frequency reversed, but the result will have a more physical interpretation.

A filter that converts sines into cosines is called a 90° phase shift filter or a *quadrature* filter. More specifically, if the input is $\cos(\omega t + \phi_1)$, then the output should be $\cos(\omega t + \phi_1 + \pi/2)$. An example is shown in Figure 4.2. Let $U(Z)$ denote the Z-transform of a real signal input and $Q(Z)$ denote a quadrature filter, then the output signal is

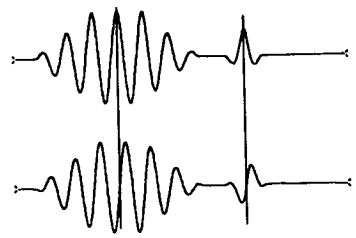

FIG. 4.2. (spec-hilb0) Input (top) filtered with quadrature filter yields phase shifted signal (bottom).

$$V(Z) = Q(Z)\,U(Z) \quad (4.5)$$

Let us find the numerical values of q_t. The time derivative operation has the desired 90° phase-shifting property we seek. The trouble with a differentiator is that higher frequencies are amplified with respect to lower frequencies. Recall the FT and take its time derivative.

$$b(t) = \int B(\omega)e^{-i\omega t}d\omega \quad (4.6)$$

$$\frac{db}{dt} = \int -i\omega B(\omega)e^{-i\omega t}d\omega \quad (4.7)$$

Thus we see that time differentiation corresponds to the weight factor $-i\omega$ in the frequency domain. The weight $-i\omega$ has the proper phase but the wrong amplitude. The desired weight factor is $Q(\omega) = -i\omega/|\omega| = -i\,\mathrm{sgn}\,\omega$ where "sgn" is called the "signum" or "sign" function.

Let us transform $Q(\omega)$ into the domain of sampled time $t = n$.

$$q_n = \frac{1}{2\pi}\int_{-\pi}^{\pi} Q(\omega)e^{-i\omega n}d\omega \quad (4.8)$$

$$= \frac{i}{2\pi}\int_{-\pi}^{0} e^{-i\omega n}d\omega - \frac{i}{2\pi}\int_{0}^{\pi} e^{-i\omega n}d\omega \quad (4.9)$$

$$= \frac{i}{2\pi}\left(\frac{e^{-i\omega n}}{-in}\Big|_{-\pi}^{0} - \frac{e^{-i\omega n}}{-in}\Big|_{0}^{\pi}\right) \quad (4.10)$$

$$= \frac{1}{2\pi n}(-1 + e^{+in\pi} + e^{-in\pi} - 1) \quad (4.11)$$

$$= \begin{cases} 0 & \text{for } n \text{ even} \\ \frac{-2}{\pi n} & \text{for } n \text{ odd} \end{cases} \quad (4.12)$$

An example of filtering with q_n is shown in Figure 4.3.

Since q_n does not vanish for negative n, the quadrature filter is nonrealizable (a filter that requires future inputs to create its present output). If the discussion were in continuous time rather than sampled time, the filter would be of the form $1/t$, a function that has a singularity at $t = 0$ and whose integral over positive t is divergent. Convolution with the filter coefficients q_n is therefore painful because the infinite sequence drops off slowly. Convolution with the filter q_t is called *Hilbert transformation*.

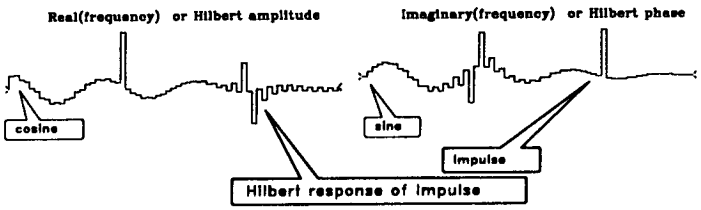

FIG. 4.3. (spec-hilb) A Hilbert transform pair.

4.1.3 The analytic signal

The so-called *analytic signal* can be constructed from a real valued time series u_t and itself 90° phase shifted, i.e., v_t from equation (4.5). The analytic signal is w_t where

$$W(Z) \quad = \quad X(Z) + iY(Z) \quad = \quad [1 + iQ(Z)]\, U(Z) \tag{4.13}$$

In the time domain the filter $[1 + iQ(Z)]$ is $\delta_t + iq_t$ where δ_t is an impulse function at time $t = 0$. The filter $1 + iQ(Z) = 1 + \omega/|\omega|$ vanishes for negative ω. Thus it is a real *step* function in the *frequency* domain.

We can guess where the name "analytic signal" comes from if we think back to Z-transforms and causal functions. They are free of poles inside the unit circle, so they are "analytic" there. Their causality is the Fourier dual to the one-sidedness we see in the frequency domain here. So, an "analytic" function is one that in the dual domain is one-sided.

4.1.4 Instantaneous envelope

The quadrature filter is often used to construct the envelope of a time function. The envelope time function may be defined by $e_t = (u_t^2 + v_t^2)^{1/2}$. Alternatively, with the analytic signal $w_t = u_t + iv_t$, the squared-envelope function is $w_t \bar{w}_t$.

A quick way to accomplish the 90° phase shift operation is to use Fourier transformation. Begin with $u_t + i \cdot 0$ and transform it to the frequency domain. Then multiply by the step function. Finally, inverse transformation getting $w_t = u_t + iv_t$. This is the time domain equivalent to $u_t + iv_t = (\delta_t + iq_t) * u_t$.

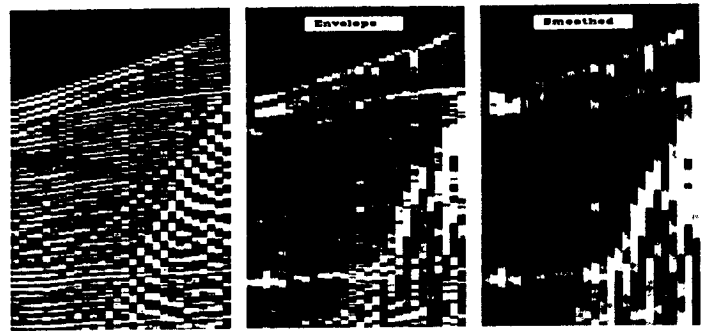

FIG. 4.4. (spec-envelope) Left is a field profile. Middle is the unsmoothed envelope function. Right is the smoothed envelope.

Sinusoids have smooth envelope functions, but that doesn't mean real seismograms do. Figure 4.4 shows an example of a field profile and unsmoothed and smoothed envelopes. Notice that before smoothing that the stepout (alignment) of the reverberatory reflections is quite clear. In the practical world, alignment is considered to be a manifestation of phase. In the theoretical world, an envelope is a smooth function such as might be used to scale data without altering its phase. Hence the reason for smoothing the envelope.

If you are interested in wave propagation you might recognise the possibility of using analytic signals. Energy stored as potential energy is 90° out of phase with kinetic energy, so u_t might represent scaled pressure while v_t represents scaled velocity. Then $\bar{w}_t w_t$ is the instantaneous energy. (The scales are the square root of compressibility and the square root of density).

4.1.5 Instantaneous frequency

The phase ϕ_t of a complex function of time $w_t = u_t + iv_t$ is defined by $\phi_t = \arctan(v_t/u_t)$. The *instantaneous frequency* is $d\phi/dt$. Before forming the derivative, recall the definition of complex logarithm of w

$$\begin{aligned} w &= re^{i\phi} \\ \ln w &= \ln|r| + \ln e^{i\phi} \\ &= \ln|r| + i\phi \end{aligned} \tag{4.14}$$

So $\phi = \Im \ln w$. The instantaneous frequency is

$$\omega_{\text{instantaneous}} \quad = \quad \frac{d\phi}{dt} \quad = \quad \Im \frac{d}{dt} \ln w(t) \quad = \quad \Im \frac{1}{w} \frac{dw}{dt} \tag{4.15}$$

For a signal that is a pure sinusoid, such as $w(t) = w_0 e^{i\omega t}$, equation (4.15) obviously gives the right answer. When a mixture of frequencies are simultaneously present, we can hope

that (4.15) gives a sensible average.

Trouble can arise in (4.15) when the denominator w gets small which happens whenever the envelope of the signal gets small. This difficulty can be overcome by careful averaging. Rationalize the denominator by multiplying by the conjugate signal, and then smooth locally a little (as indicated by the summation sign below).

$$\hat{\omega}_{\text{smoothed}} = \Im \frac{\sum \bar{w}(t) \frac{d}{dt} w(t)}{\sum \bar{w}(t)\, w(t)} \tag{4.16}$$

(Those of you who have had a class in quantum mechanics may recognize the notion of the "expectation of an operator." You also see why the probability wavefunction of quantum physics must be complex valued—it is a consequence of the analytic signal eliminating negative frequencies from the average. If the negative frequencies were not eliminated, then the average frequency would be zero.)

What range of times should be smoothed in equation (4.16)? Besides the nature of the data, the appropriate smoothing depends on the method of representing $\frac{d}{dt}$. To prepare a figure, I implemented $\frac{d}{dt}$ by multiplication by $-i\omega$. (This has the advantage of more accuracy than finite differences at high frequencies, but the disadvantage that the discontinuity in slope at the Nyquist frequency gives an extended transient in the time domain). The result is shown in Figure 4.5. Inspection of Figure 4.5 shows that smoothing is even more

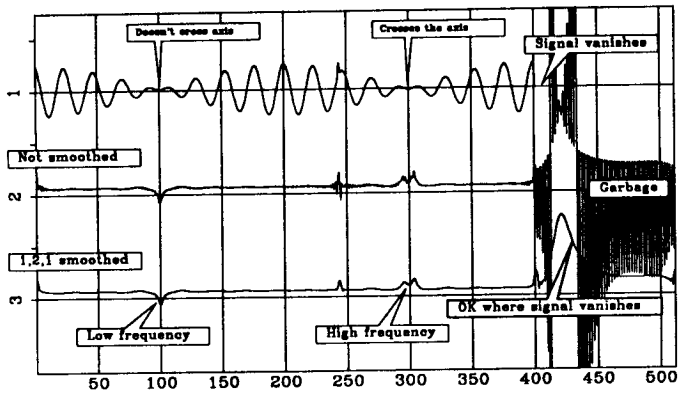

FIG. 4.5. (spec-node) A sum of three sinusoids (top), unsmoothed instantaneous frequency (middle), and smoothed instantaneous frequency (bottom).

necessary for instantaneous frequency than for envelopes, and this is not surprising because the presence of $\frac{d}{dt}$ makes it rougher. Particularly notice times in the range 400-512 where the sinusoids are truncated. There the unsmoothed instantaneous frequency becomes a large rapid oscillation near the Nyquist frequency. This roughness is nicely controlled by

$(1, 2, 1)$ smoothing.

It is also gratifying to see that a spike added to the sinusoids (at point 243) makes a burst of high frequency, and curious to notice the behavior where an oscillation approaches the axis and then turns away just before or just after crossing the axis.

An example of instantaneous frequency applied to field data is shown below in Figure 4.6. Although the instantaneous frequency formalism is appealing it has not found routine use

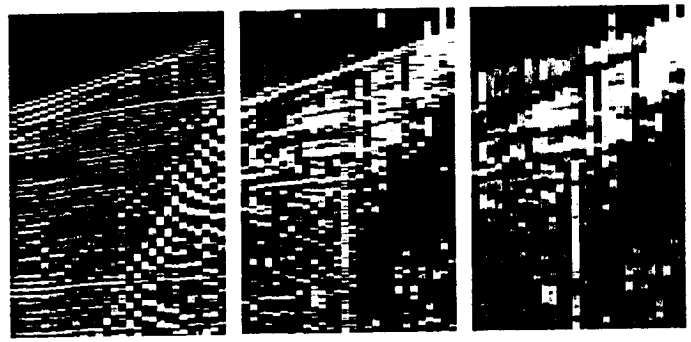

FIG. 4.6. (spec-frequency) A field profile (left), instantaneous frequency smoothed only with (1,2,1) (middle) and smoothed more heavily (right).

in practice. Part of the reason might be that the literature does not seem to include the stablized form (4.16). A more direct reason is that instantaneous frequency is not a part of the standard algorithms of earth imaging.

The instantaneous frequency idea can also be applied to the space axis. This will be more familiar to readers familiar with the methodology of imaging and migration. Instead of temporal frequency $\omega = d\phi/dt$ we compute the spatial frequency $k_x = d\phi/dx$. Figure 4.7 shows an example.

EXERCISES:

1 Let c_t be a *complex* causal function of time. How does $X(Z)$ change in equation (4.2), and how must $Y(Z)$ in equation (4.3), be deduced from $X(Z)$?

2 Figure 4.3 shows a Hilbert transform pair, the real and imaginary parts of the Fourier transform of a causal time function. Describe the causal time function.

3 Given $Y(Z) = Q(Z)X(Z)$ prove that the envelope of y_t is the same as the envelope of x_t.

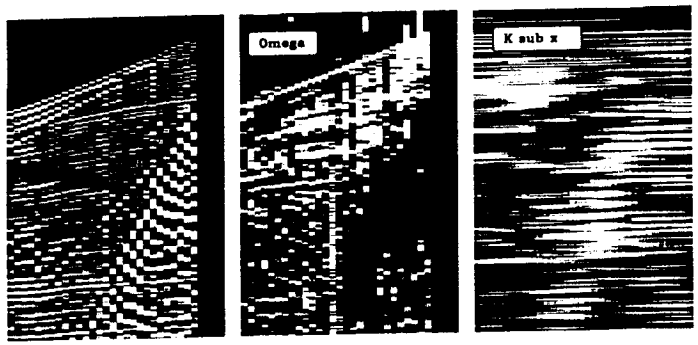

FIG. 4.7. (spec-kx) A field profile (left), unsmoothed instantaneous frequency (center) and instantaneous spatial frequency k_x (right).

4 By means of partial fractions convolve the waveform

$$(2/\pi)(\ldots,-\frac{1}{5},0,-\frac{1}{3},0,-1,0,1,0,\frac{1}{3},0,\frac{1}{5},\ldots)$$

with itself. What is the interpretation of the fact that the result is $(\ldots,0,0,-1,0,0,\ldots)$? (HINT: $\pi^2/8 = 1+\frac{1}{9}+\frac{1}{25}+\frac{1}{49}+\ldots$).

5 Using the fast Fourier transform matrix, the quadrature filter $Q(\omega)$ may be represented by the column vector

$$-i(0,1,1,1,\ldots,0,-1,-1,-1,\ldots,-1)^{\mathrm{T}}$$

Multiply this into the inverse transform matrix to show that the transform is proportional to $(\cos \pi k/N)/(\sin \pi k/N)$. What is the scale factor? Sketch it for $k \ll N$ indicating the limit $N \to \infty$. (HINT: $1+x+x^2+\ldots+x^N = (1-x^{N+1})/(1-x)$).

4.2 PHASE DELAY AND GROUP DELAY

The Fourier domain ratio of a wave seen at B divided by the wave seen at A may be regarded as a filter. Propagation velocity is the distance from A to B divided by the delay. But there are at least two ways to *define* the delay.

4.2.1 Phase delay

When you put a sinusoid into a filter then a sinusoid must come out. The only thing that can change is the amplitude and the phase. Comparing a zero crossing of the input to a zero crossing of the output measures the so-called ***phase delay***. To quantify this, define an input, $\sin \omega t$, and an output, $\sin(\omega t - \phi)$. Then the phase delay t_p is found by solving

$$\begin{aligned} \sin(\omega t - \phi) &= \sin \omega(t - t_p) \\ \omega t - \phi &= \omega t - \omega t_p \\ t_p &= \frac{\phi}{\omega} \end{aligned} \tag{4.17}$$

A problem with phase delay is that the phase can be ambiguous within an additive constant of $2\pi N$ where N is any integer. In wave propagation, the ambiguity is resolved if the input and the output are near enough that the phase is in the principle quadrant. In wave propagation theory, ***phase velocity*** is defined by the distance divided by the phase delay.

4.2.2 Group delay

A more interesting kind of delay is ***group delay*** corresponding to group velocity in wave propagation theory. In many cases the group delay is nothing more than the phase delay. This happens when the phase delay is independent of frequency. But when the phase delay depends on frequency, then a completely new velocity called the ***group velocity*** appears. Curiously, the group velocity is *not* an average of phase velocities.

The simplest analysis of group delay begins by defining a filter input x_t as the sum of two frequencies.

$$x_t = \cos \omega_1 t + \cos \omega_2 t \tag{4.18}$$

By using a trigonometric identity

$$x_t = \underbrace{2 \cos(\frac{\omega_1 - \omega_2}{2} t)}_{\text{beat}} \cos(\frac{\omega_1 + \omega_2}{2} t) \tag{4.19}$$

we see that the sum of two cosines looks like a cosine of the average frequency multiplied by a cosine of half the difference frequency. Since the frequencies in Figure 4.8 are taken close together, the difference frequency factor in (4.19) represents a slowly variable amplitude

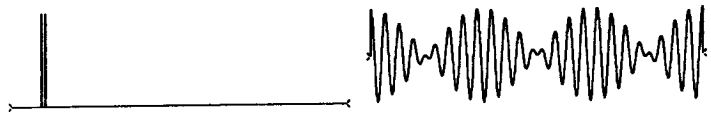

FIG. 4.8. (spec-beat) Two nearby frequencies beating.

multiplying the average frequency. The slow (difference frequency) modulation of the higher (average) frequency is called ***beating***.

The beating phenomena is also called "interference" though that word is deceptive. If the two sinusoids were two wave beams crossing one another, they would simply cross each other *without* interfering. Where they are present simultaneously, they simply add.

Each of the two frequencies could be delayed a different amount by a filter, so take the output of the filter y_t to be

$$y_t = \cos(\omega_1 t - \phi_1) + \cos(\omega_2 t - \phi_2) \tag{4.20}$$

In taking the output of the filter to be of the form of (4.20), we have assumed that neither frequency was attenuated. (The group velocity concept loses its simplicity and much of its utility in dissipative media.) Using the same trigonometric identity on (4.20) gives

$$y_t = \underbrace{2\cos(\frac{\omega_1-\omega_2}{2}t - \frac{\phi_1-\phi_2}{2})}_{\text{beat}}\cos(\frac{\omega_1+\omega_2}{2}t - \frac{\phi_1+\phi_2}{2}) \tag{4.21}$$

Rewriting the beat factor in terms of a time delay t_g, we have

$$\cos[\frac{\omega_1-\omega_2}{2}(t-t_g)] = \cos(\frac{\omega_1-\omega_2}{2}t - \frac{\phi_1-\phi_2}{2}) \tag{4.22}$$

$$(\omega_1-\omega_2)t_g = \phi_1-\phi_2 \tag{4.23}$$

$$t_g = \frac{\phi_1-\phi_2}{\omega_1-\omega_2} = \frac{\Delta\phi}{\Delta\omega} \tag{4.24}$$

For a continuum of frequencies the group delay is

$$t_g = \frac{d\phi}{d\omega} \tag{4.25}$$

4.2.3 Group delay as a function of the FT

We'll see that the group delay is a simple function of the Fourier transform of the filter. We'll assign the name P to this filter to denote its being an all-pass filter. It need not be a causal all-pass filter, and in practice a bit of energy absorption might be OK too. The phase angle ϕ could be computed as the arctangent of the ratio of imaginary to real parts of the Fourier transform, namely $\phi(\omega) = \arctan[\Im P(\omega)/\Re P(\omega)]$. As with (4.14) we use $\phi = \Im \ln P$ and from (4.25) we get

$$t_g = \frac{d\phi}{d\omega} = \Im\frac{d}{d\omega}\ln P(\omega) = \Im\frac{1}{P}\frac{dP}{d\omega} \tag{4.26}$$

which could be expressed like the Fourier dual to equation (4.16).

4.2.4 Observation of dispersive waves

There are various formulas relating energy delay to group delay, and this chapter should illuminate those that are one dimensional. In observational work, it is commonly said that "what you see is the group velocity." This means when you see an apparently sinusoidal wave train, its distance from the source divided by its travel time (group delay) is the group velocity. An interesting example of a dispersive wave is in FGDP Figure 1-11.

4.2.5 Group delay of all-pass filters

On page 46 we introduced all-pass filters, i.e., filters with constant unit spectra, that is, $P(Z)\overline{P}(1/Z) = 1$. In the frequency domain $P(Z)$ can be expressed as $e^{i\phi(\omega)}$ where ϕ is real and is called the *phase shift*. Clearly $P\overline{P} = 1$ for all real ϕ. It is an easy matter to make a filter with any desired phase shift—you merely Fourier transform $e^{i\phi(\omega)}$ into the time domain. If $\phi(\omega)$ is arbitrary, the resulting time function is likely to be two-sided. Since we are interested in physical processes which are causal, we may wonder what class of functions $\phi(\omega)$ corresponds to one-sided time functions. The answer is that the group delay $\tau_g = d\phi/d\omega$ of a causal all-pass filter must be positive.

Proof that $d\phi/d\omega > 0$ for a causal all-pass filter is found in Fundamentals of Geophysical Data Processingand there is no need to reproduce the algebra here. The proof begins from equation (3.35) and takes the imaginary part of the logarithm to get phase. Differentiation with respect to ω yields a form that is recognizable as a spectrum, hence always positive.

A single-pole, single-zero all-pass filter passes all frequency components with constant gain and a phase shift that can be adjusted by the placement of the pole. Taking Z_0 near the unit circle causes most of the phase shift to be concentrated near the frequency where the pole is located. Taking the pole further away causes the delay to be spread over more frequencies. Complicated phase shifts or group delays may be built up by cascading single-pole filters.

The above reasoning for a single-pole, single-zero all-pass filter also applies to many roots because the phase of each will add and the sum of $\tau_g = d\phi/d\omega > 0$ will be greater than zero.

The Fourier dual to the group delay of a causal all-pass filter is that the instantaneous frequency of a certain class of analytic signals must be positive. This class of analytic signals is all of those with a constant envelope function, as might be approximated by field data after the process of automatic gain control.

EXERCISES:

1 Let x_t be some real time function. Let $y_t = x_{t+3}$ be another real time function. Sketch the phase as a function of frequency of the cross spectrum $X(1/Z)Y(Z)$ as computed by a computer which put all arctangents in the principal quadrants $-\pi/2 < \arctan < \pi/2$. Label axis scales.

2 Sketch the amplitude, phase, and group delay of the all-pass filter $(1-\overline{Z_0}Z)/(Z_0-Z)$ where $Z_0 = (1+\epsilon)e^{i\omega_0}$ and ϵ is small. Label important parameters on the curve.

3 Show that the coefficients of an all-pass, phase-shifting filter made by cascading $(1-\overline{Z_0}Z)/(Z_0-Z)$ with $(1-Z_0Z)/(\overline{Z_0}-Z)$ are real.

4 A continuous time function is the impulse response of a continuous-time, all-pass filter. Describe the function in both time domain and frequency domain. Interchange the words *time* and *frequency* in your description of the function. What is a physical example of such a function? What happens to the statement: "The group delay of an all-pass filter is positive."?

5 A graph of the group delay $\tau_g(\omega)$ shows τ_g to be positive for all ω. What is the area under τ_g in the range $0 < \omega < 2\pi$. (HINT: This is a trick question you can solve in your head.)

4.3 MINIMUM PHASE

In Chapter 3 we learned that the inverse of a causal filter $B(Z)$ is causal if $B(Z)$ has no roots inside the unit circle. The terminology "minimum phase" was introduced there without motivation. Here we examine the phase, and learn why it is called *minimum*.

4.3.1 Phase of a single root

Let Z_0 be a complex number. It could be $x_0 + iy_0$ or $Z_0 = e^{i\omega_0}/\rho$ where ρ and ω_0 are fixed constants. Consider the phase shift of the two-term filter

$$B(Z) = 1 - \frac{Z}{Z_0} \tag{4.27}$$

$$B(Z(\omega)) = 1 - \rho e^{i(\omega-\omega_0)} \tag{4.28}$$

$$B(Z(\omega)) = 1 - \rho\cos(\omega-\omega_0) - i\rho\sin(\omega-\omega_0) \tag{4.29}$$

Real and imaginary parts of B are plotted in Figure 4.9. Arrows are at frequency ω intervals

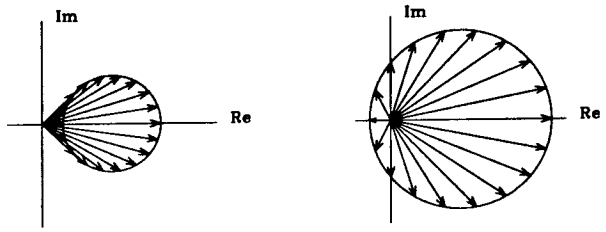

FIG. 4.9. (spec-origin) Left, complex B plane for $\rho < 1$. Right, for $\rho > 1$.

of 20°. Observe that for $\rho > 1$ the sequence of arrows has a sequence of angles that ranges over 360°, whereas for $\rho > 1$ the sequence of arrows has a sequence of angles between ±90°. Now let us replot equation (4.29) in a more conventional way, with ω as the horizontal axis. Whereas the phase is the angle of an arrow in Figure 4.9, in Figure 4.10 it is the arctangent of $\Im B/\Re B$. Notice how different is the phase curve in Figure 4.10 for $\rho < 1$ than for $\rho > 1$.

Real and imaginary parts of B are *periodic* functions of frequency ω since $B(\omega) = B(\omega + 2\pi)$. You might be tempted to conclude that the phase would be periodic too. But

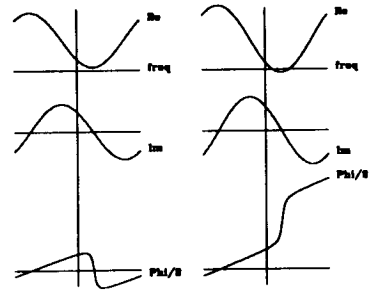

FIG. 4.10. (spec-phase) Left, shows real and imaginary parts and phase angle of equation (4.29), for $\rho < 1$. Right, for $\rho > 1$. Left is minimum phase and right is nonminimum phase.

Figure 4.10 shows that for a nonminimum phase filter, as ω ranges from zero to 2π the phase ϕ increases by 2π (because the circular path in Figure 4.10 surrounds the origin).

The word *minimum* arises because delaying a filter can always add more phase. For example, multiplying any polynomial by Z delays it and adds ω to its phase.

For the minimum phase filter the group delay $d\phi/d\omega$ applied to Figure 4.10 is positive for all ω. For the nonminimum phase filter it is negative for ω near ω_0.

Because group delay $d\phi/d\omega$ is the Fourier dual to instantaneous frequency, we can now go back to Figure 4.5 and explain the discontinuous behavior of instantaneous frequency where the signal amplitude is near to zero.

4.3.2 Phase of a rational filter

Now we are ready to consider a complicated filter like

$$B(Z) = \frac{(Z-c_1)(Z-c_3)\cdots}{(Z-a_1)(Z-a_2)\cdots} \tag{4.30}$$

By the rules of complex-number multiplication the phase of $B(Z)$ is the sum of the phases in the numerator minus the sum of the phases in the denominator. Since we are discussing realizable filters the denominator factors must all be minimum phase, and so the denominator phase curve is a sum of periodic phase curves like lower left in Figure 4.10.

The numerator factors may or may not be minimum phase. Thus the numerator phase curve is a sum of phase curves like either type in Figure 4.10. If any factors augment phase by 2π, then the phase is not periodic and the filter is nonminimum phase.

4.4 FILTERS IN PARALLEL

We have seen that in a cascade of filters the filter polynomials are multiplied together. When filters operate in parallel then their polynomials add. See Figure 4.11.

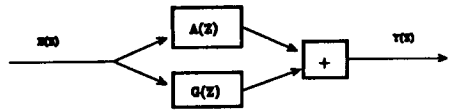

FIG. 4.11. (spec-2-10) Filters operating in parallel.

We have seen that a cascade of filters is minimum phase if, and only if, each element of the product is minimum phase. Now we will see a sufficient (but not necessary) condition that a sum $A(Z) + G(Z)$ be minimum phase. First, assume that $A(Z)$ is minimum phase. Then write

$$A(Z) + G(Z) = A(Z)\left[1 + \frac{G(Z)}{A(Z)}\right] \tag{4.31}$$

The question whether $A(Z) + G(Z)$ is minimum phase is now reduced to determining

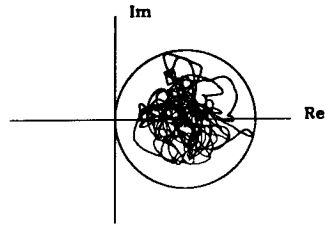

FIG. 4.12. (spec-garbage) A phase trajectory like in Figure 4.9 left, but more complicated.

whether $A(Z)$ and $1 + G(Z)/A(Z)$ are both minimum phase. We have assumed that $A(Z)$ is minimum phase. Before we ask whether $1 + G(Z)/A(Z)$ is minimum phase we need to be sure that it's causal. Since $1/A(Z)$ is expandable in positive powers of Z only, then $G(Z)/A(Z)$ is also causal. We will next see that a sufficient condition for $1 + G(Z)/A(Z)$ to be minimum phase is that the spectrum of A exceeds that of G at all frequencies. In other words, for any real ω, $\mid A \mid > \mid G \mid$. Thus, if we plot the curve of $G(Z)/A(Z)$ in the complex plane, for real $0 \le \omega \le 2\pi$ it lies everywhere inside the unit circle. Now if we add unity—getting $1 + G(Z)/A(Z)$, the curve will always have a positive real part as in Figure 4.12. Since the curve cannot enclose the origin, the phase must be that of a minimum-phase function. In words, "You can add garbage to a minimum-phase wavelet if you do not add too much.

This abstract theorem has an immediate physical consequence. Suppose a wave characterized by a minimum phase $A(Z)$ is emitted from a source and detected at a receiver some time later. At a still later time an echo bounces off a nearby object and is also detected at the receiver. The receiver sees the signal $Y(Z) = A(Z) + Z^n \alpha A(Z)$ where n measures the delay from the first arrival to the echo and α represents the amplitude attenuation of the echo. To see that $Y(Z)$ is minimum phase, we note that the magnitude of Z^n is unity and that the reflection coefficient α must be less than unity (to avoid perpetual motion) so that $Z^n \alpha A(Z)$ takes the role of $G(Z)$. Thus a minimum-phase wave along with its echo is minimum phase. We will later consider wave propagation with echoes of echoes ad infinitum.

EXERCISES:

1 Find two nonminimum-phase wavelets whose sum is minimum phase.

2 Let $A(Z)$ be a minimum-phase polynomial of degree N. Let $A'(Z) = Z^N \overline{A}(1/Z)$. Locate in the complex Z plane the roots of $A'(Z)$. $A'(Z)$ is called *maximum phase*. [HINT: Work the simple case $A(Z) = a_0 + a_1 Z$ first.]

3 Suppose $A(Z)$ is maximum phase and that the degree of $G(Z)$ is less than or equal to the degree of $A(Z)$. Assume $\mid A \mid > \mid G \mid$. Show that $A(Z) + G(Z)$ is maximum phase.

4 Let $A(Z)$ be minimum phase. Where are the roots of $A(Z) + cZ^N \overline{A}(1/Z)$ in the three cases $\mid c \mid < 1, \mid c \mid > 1, \mid c \mid = 1$? (HINT: The roots of a polynomial are continuous functions of the polynomial coefficients.)

4.5 ROBINSON'S ENERGY DELAY THEOREM

Here we will see that a minimum-phase wavelet has less energy delay than any other one-sided wavelet with the same spectrum. More precisely the energy summed from zero to any time t for the minimum-phase wavelet is greater than or equal to that of any other wavelet with the same spectrum. Refer to Figure 4.13.

FIG. 4.13. (FGDP fig 3-2) Percent of total energy in a filter between time 0 and time t.

Here is how I prove Robinson's energy delay theorem: Compare two wavelets F_{in} and F_{out} which are identical except for one zero, which is outside the unit circle for F_{out} and

inside for F_{in}. We may write this as

$$F_{\text{out}}(Z) = (b + sZ)\,F(Z) \tag{4.32}$$

$$F_{\text{in}}(Z) = (s + bZ)\,F(Z) \tag{4.33}$$

where b is bigger than s and F is arbitrary but of degree n. Complex valued b and s are left for an exercise. Notice that the spectrum of $b + sZ$ is the same as that of $s + bZ$. Next tabulate the terms in question.

t	F_{out}	F_{in}	$F_{\text{out}}^2 - F_{\text{in}}^2$	$\sum_{k=0}^{t}(F_{\text{out}}^2 - F_{\text{in}}^2)$
0	bf_0	sf_0	$(b^2 - s^2)\,{f_0}^2$	$(b^2 - s^2)\,{f_0}^2$
1	$bf_1 + sf_0$	$sf_1 + bf_0$	$(b^2 - s^2)\,({f_1}^2 - {f_0}^2)$	$(b^2 - s^2)\,{f_1}^2$
⋮	⋮			
k	$bf_k + sf_{k-1}$	$sf_k + bf_{k-1}$	$(b^2 - s^2)\,({f_k}^2 - {f_{k-1}}^2)$	$(b^2 - s^2)\,{f_k}^2$
⋮	⋮			
$n+1$	sf_n	bf_n	$(b^2 - s^2)(-{f_n}^2)$	0

The difference, which is given in the right-hand column, is clearly always positive.

To prove that the minimum-phase wavelet delays energy the least, the preceding argument is repeated with each of the roots until they are all outside the unit circle.

EXERCISES:

1 Repeat the proof of Robinson's minimum-energy-delay theorem for complex-valued b, s, and f_k. [HINT: Does $F_{\text{in}} = (s + bZ)\,F$ or $F_{\text{in}} = (\bar{s} + \bar{b}Z)F$?]

4.6 SPECTRAL FACTORIZATION

A problem that arises in a variety of physical contexts is this: Given a spectrum, find a minimum phase wavelet which has that spectrum. We will see how to construct this wavelet and see that it is unique (except for a trivial aspect, the negative of any wavelet has the same spectrum as the wavelet, and more generally any wavelet can be multiplied by any complex number of unit magnitude, such as $\pm i$, etc).

First consider the simpler problem where the wavelet need not be causal. You can easily find a symmetric wavelet with any spectrum. Just take the spectrum (which by definition is an energy or power) and take its square root getting what is called the *amplitude* spectrum. Inverse transform that to the time domain and you have a symmetric wavelet with the desired spectrum.

This section solves the *spectral factorization* problem, namely, finding a minimum phase wavelet to go along with any given spectrum. Thus, given any wavelet it shows how to find another wavelet that is minimum phase and has the same spectrum as the given wavelet. Some examples are shown in Figure 4.14. For all but the 4th signal it is obvious that the spectrum of the minimum phase wavelet matches that of the input. Wavelets are shifted to $t = 0$ and turned backwards. In the 4th case the wave shape changes to make a big pulse

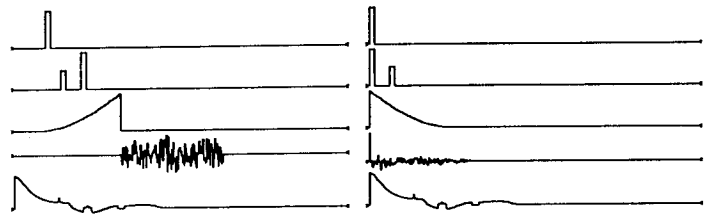

FIG. 4.14. (spec-mpsamples) Left are given wavelets and right are minimum phase equivalents.

at zero lag. As the Robinson theorem suggests, minimum-phase wavelets tend to decay rapidly after a strong onset. I imagined that hand drawn wavelets with a strong onset would rarely turn out to be perfectly minimum phase, but when I tried it, I was surprised at how easy it seemed to be to draw a minimum-phase wavelet. This is shown on the bottom of Figure 4.14.

To begin understanding spectral factorization notice that the polar form of any complex number puts the phase into the exponential i.e. $x + iy = |r|e^{i\phi} = e^{\ln|r| + i\phi}$. So we look first into the behavior of exponentials and logarithms of Fourier transforms.

4.6.1 Exponential of a causal is causal

Begin with a causal time function c_t and its associated $C(Z)$. The Z-transform $C(Z)$ could be evaluated giving a complex value for each real ω. This complex value could be exponentiated to get another value, say

$$B(Z(\omega)) = e^{C(Z(\omega))} \tag{4.34}$$

Next we could inverse transform $B(Z(\omega))$ back to b_t. We will prove the amazing fact that b_t must be causal too.

First notice that when the Z-transform $C(Z)$ of any causal wavelet is squared, that the resulting wavelet is also causal because nothing happens to make negative powers of Z. Likewise for the third power or any positive integer power, or sum of positive integer powers. Now recall the basic power series definition of exponential

$$e^x = 1 + x + \frac{x^2}{2} + \frac{x^3}{2\cdot 3} + \frac{x^4}{2\cdot 3\cdot 4} + \frac{x^5}{2\cdot 3\cdot 4\cdot 5} + \cdots \tag{4.35}$$

Including equation (4.34) gives

$$B(Z) = e^{C(Z)} = 1 + C(Z) + \frac{C(Z)^2}{2} + \frac{C(Z)^3}{2\cdot 3} + \frac{C(Z)^4}{2\cdot 3\cdot 4} + \cdots \tag{4.36}$$

Each term in the infinite series corresponds to a causal time function, so the sum, b_t is causal.

(If you have forgotten the series for the exponential function, then recall that the solution to $dy/dx = y$ is the definition of the exponential function $y(x) = e^x$ and the power series satisfies the differential equation term by term, so it must be the exponential too. The factorials in the denominators assure us that the power series always converges, i.e. it is finite for any finite x.) Now you are ready for the big construction. Define $x = C(Z)$ and substitute into (4.35).

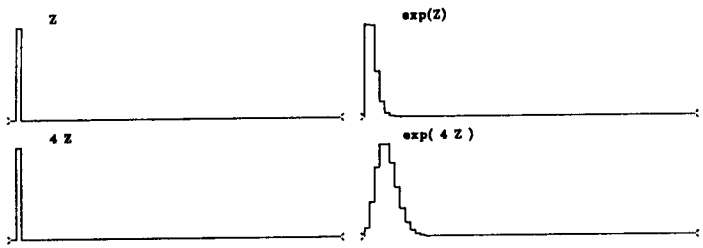

FIG. 4.15. (spec-eZ) Exponentials.

Figure 4.15 shows examples of equation (4.36) for $C = Z$ and $C = 4Z$. Unfortunately I don't have an analytic calculation to confirm the validity of that example. This suggests I should adapt the program for complex time series to look at examples like $e^{i\omega t_0}$ and $\exp((-i\omega)^{.7} t_0)$. Anyway, I SHOULD try $\exp((-i\omega)^{1/2} t_0)$.

4.6.2 Getting a causal wavelet from a prescribed spectrum

Take as given the spectrum of the causal wavelet $B(Z)$. That means that we are not given $B(Z)$ itself, but we are given $S(Z) = \overline{B}(1/Z)B(Z)$. Thus

$$S(Z) \quad = \quad \overline{B}(1/Z)\, B(Z) \quad = \quad e^{\overline{C}(1/Z)}\, e^{C(Z)} \quad = \quad e^{\overline{C}(1/Z)+C(Z)} \tag{4.37}$$

Given the spectrum $S(Z)$ for each value on the unit circle we could deduce the log spectrum $U(Z) = \ln S(Z)$ at each point on the unit circle.

$$U(Z) \quad = \quad \ln[S(Z)] \quad = \quad \overline{C}(1/Z) + C(Z) \tag{4.38}$$

This is the answer we have been looking for. Given $U(Z)$ for all real values of ω we could inverse transform it to the time domain getting the two sided function $u_t = c_{-t} + c_t$. Setting to zero the coefficients at negative times eliminates c_{-t} leaving just c_t hence $C(Z)$, and we already know that the exponential of $C(Z)$ gives $B(Z)$ with a causal b_t.

An example of log spectra is shown in Figure 4.16. On the infinite domain the FT of a box function is a sinc whose zeros become minus infinities in the logarithm. On the discrete domain exact zeros may occur or not. The transform of a triangle is a sinc squared, but since this triangle is imperfectly drawn, its transform does not go identically to zero.

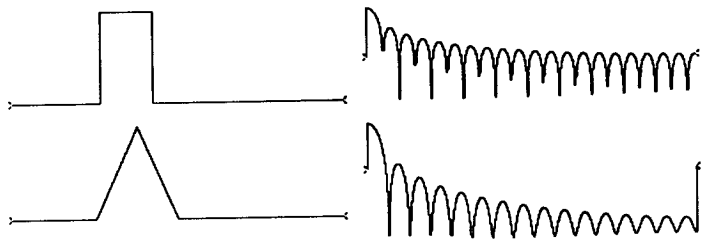

FIG. 4.16. (spec-logspec) Log spectra of a box function and a triangle function.

The sinc function drops off as ω and sinc squared drops off as ω^2. We confirm this on the logarithm plot by sinc squared dropping off *twice* as much.

An example of a truncated sinc approximation is shown in Figure 4.17. A flat top function in log frequency was tapered down rapidly with frequency. Then the resulting time-domain representation was tapered to zero under a triangle ramp.

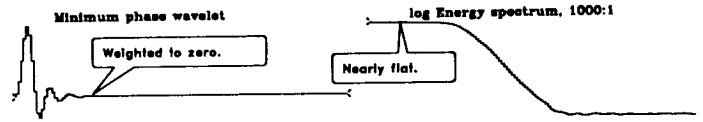

FIG. 4.17. (spec-causinc) Truncated sinc approximation.

4.6.3 Why the causal wavelet is minimum phase

Next we see why the causal wavelet $B(Z)$ we have constructed from the prescribed spectrum turns out to be minimum phase. First return to the original definition of minimum phase, a causal wavelet is minimum phase if and only if its inverse is causal. We have our wavelet in the form $B(Z) = e^{C(Z)}$. Consider another wavelet $A(Z) = e^{-C(Z)}$ constructed analogously. By the same reasoning, a_t is also causal. Since $A(Z)B(Z) = 1$ we have found a causal, inverse wavelet. Thus b_t is a minimum phase wavelet.

Since the phase is a Fourier series it must be periodic, it cannot increase indefinitely

with ω as it does for the nonminimum phase wavelet (see Figure 4.10).

4.6.4 Relation of amplitude to phase

As we learned from equation (4.38), a minimum phase function is determined completely from its spectrum. Thus its phase is determinable from its spectrum. Likewise, we will see that except for a scale, the spectrum is determinable from the phase.

It wasn't mentioned at the time, but spectral factorization implicitly uses Hilbert transformation. Somehow we generated a phase. To see how the phase arose, recall equation (4.37) and (4.38).

$$S_k = e^{\ln S_k} = e^{U_k} = e^{(U_k - i\Phi_k)/2}\, e^{(U_k + i\Phi_k)/2} = e^{\overline{C}_k} e^{C_k} = \overline{B}_k B_k \qquad (4.39)$$

Where did Φ_k come from? We took $U_k + i0$ to the time domain getting u_t. Then we multiplied u_t by a real valued step function of time. This multiplication in the time domain is what created the phase. It created the phase because multiplication in the time domain implies a convolution in the frequency domain. Recall the Fourier transform of a real valued step function that arises with Hilbert transform. Multiplying in time with a step means that in frequency U_k has been convolved with $\delta_{k=0} + i \times$ (90° phase shift filter). So U_k is unchanged and a phase, Φ_k has been generated. The explanation will be somewhat clearer if you review the Z-transform approach at the beginning of the chapter because there you can see both the frequency domain and the time domain in one expression.

To illustrate different classes of discontinuity, pulse, step, and slope, Figure 4.18 shows another Hilbert transform pair.

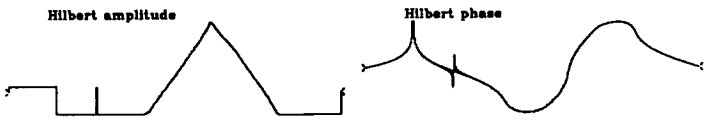

FIG. 4.18. (spec-hilb2) A Hilbert transform pair.

4.6.5 Computation

The task of putting one polynomial into another or one infinite series into another is an onerous task though it does lead to a wavelet that is exactly causal. In practice we do operations that are conceptually the same, but for speed we do them with discrete Fourier transforms. The disadvantage is periodicity, i.e. negative times are represented computationally like negative frequencies, they are the last half of the elements of a vector so there can be some blurring of late times into negative times.

The program below begins with a wavelet, forms its log spectrum, and then uncovers the minimum phase wavelet with that spectrum. (Between where negative lags are set to zero and positive lags are left untouched, are two points that are scaled by half.)


```
subroutine cfact( nt, cx)
integer nt, i
complex cx(1)
call fft( nt, cx, 1., 1.)
do i = 1,nt
        cx(i) = clog( cx(i) * conjg(cx(i)) )
call fft( nt, cx, -1., 1./nt)
cx(1)      = cx(1)      / 2.
cx(1+nt/2) = cx(1+nt/2) / 2.
do i = 1+nt/2+1, nt
        cx(i) = 0.
call fft( nt, cx, +1., 1.)
do i = 1,nt
        cx(i) = cexp(cx(i))
call fft( nt, cx, -1., 1./nt)
return; end
```

The first test I tried on this program was the wavelet (1,2,0,0). The desired result is obviously that the wavelet should time reverse itself to (2,1,0,0). The actual result was (1.9536, 1.0837, 0.0464, -0.0837). This result is imperfect because the four-point Fourier transform is a summation around the unit circle while theoretically an integration is called for. So, better results can be obtained by padding additional zeros after the input wavelet. Also, you might notice that the program is designed for complex valued wavelets. As typical of Fourier transform with single-word precision, the imaginary parts were about 10^{-6} of the real parts instead of being precisely zero.

4.6.6 Pathological examples

The spectral factorization algorithm fails when an infinity is encountered. This happens when the spectrum becomes zero so that its logarithm becomes minus infinity. This can happen in a rather benign way, for example with the spectrum of the wavelet (1,1), where the infinity occurs at the Nyquist frequency. You could obviously omit the Nyquist, or set its value to zero, or better yet, smooth off the spectrum in the vicinity of the Nyquist before you take the logarithm. On the other hand, the pathology can be more extreme. With the same wavelet, (1,1) convolved with itself N times we get the binomial coefficients with the origin at one end, so it doesn't look as bunched at the origin as the Robinson theorem would have you hope.

Figure 4.19 shows functions whose spectrum contains zeros, along with their minimum phase equivalents. When the logarithm of zero arises during the computation it is replaced by the log of 10^{-30}. It is surprising that the the triangle suffered so much less than the other two functions. It seems that minor imperfection in specifying the triangle resulted in a spectrum that did not have the theoretical zeros of sinc squared.

EXERCISES:

1 What is the meaning of *minimum-phase waveform* if the roles of time domain and frequency domain are interchanged?

2 Show how to do the inverse Hilbert transform, given ϕ find u. What is the interpretation of the fact that you cannot get u_0?

3 Consider a model of a portion of the earth where z is the north coordinate, $+z$ represents altitude above the earth, and magnetic bodies are distributed in the earth so as to

Chapter 5

IMPEDANCE

Classical physics gives much attention to energy conservation and dissipation. Engineering filter theory gives much attention to causality—that there can be no response before the excitation. In geophysics we often need to ensure both causality and energy loss. We need to incorporate both, not only in theoretical derivations, but also in computations, and sometimes in computations that are discretized in time. There is a special class of mathematical functions called *impedance* functions that describe causal, linear disturbances in physical objects that dissipate energy. An impedance function is the filter that characterizes a passive medium by relating its "flow" variable to its "pressure" variable. Examples are:

application	**pressure**	**flow**
mechanics	force	velocity
acoustic	pressure	parcel velocity
elastic	stress	strain rate
circuit	voltage	current
thermodynamics	temperature	heat flow
magnetic	B	H
electric	E	D
commerce	price	sales

In this chapter we examine the theory of impedance functions, their precise definition, their computation in the world of discretized time, and the rules for combining simple impedances to get more complicated ones. We will also examine the relationship of the *minimum-phase* filter to the impedance filter. Rocks are unlike "pure" substances because they contain irregularities at all scales. A particularly simple impedance function will be found that mimics the dissipation of energy in rocks, unlike the classical equations of Newtonian viscoelasticity. One of the most interesting impedance functions, wide-angle wave extrapolation, is found in my other book, "Imaging the Earth's Interior" .

Chapter 7

RESOLUTION

The accuracy of measurements on observed signals is limited not only by practical realities, but also by certain fundamental principles. A well-known example is the time-bandwidth product in Fourier transformation theory called the "uncertainty principle."

Observed signals often look random and often they are modeled by filtered random numbers. In this chapter we will see many examples of signals built from random numbers and see how the nomenclature of statistics applies to them. Fundamentally, this chapter characterizes *resolution*, resolution of frequency and arrival time, and the statistical resolution of signal amplitude and power as functions of time and frequency.

We will see $\sqrt{n}$ popping up everywhere, for example, signals that are theoretically uncorrelated generally appear to be somewhat correlated at a level derived from $\sqrt{n}$.

Measures of resolution (which are called variances, tolerances, uncertainties, bandwidths, durations, spreads, spans, etc.) often interact with one another so that experimental change to reduce one must necessarily increase another or some combination of the others. This chapter exhibits basic cases where such conflicting interactions occur.

We should begin with formal definitions of time duration ΔT and spectral bandwidth ΔF. A variety of defining equations are easy to write, and many are in general use. The time duration of a damped exponential function is infinite if by duration you mean the span of nonzero function values. However, for practical purposes the time span is generally chosen as the time required for the amplitude to decay to e^{-1} of its original value. For many functions the span is defined by the span between points on the time or frequency axis where the curve (or its envelope) drop to half of the maximum value. The main idea is that the time span ΔT or the frequency span ΔF should be able to include most of the energy but need not contain it all.

A problem we will need to grapple with is that no definition leads to easy quantitative analysis. Thus, this chapter is more descriptive than analytic.

7.1 TIME-FREQUENCY RESOLUTION

Scaling a function to be narrower in one domain also scales it to be wider in the other domain. This is a consequence of Fourier transforms being built from $e^{i\omega t}$. Scaling ω implies inverse scaling of t to keep the product ωt constant. For example the FT of a rectangle is a sinc. Making the rectangle narrower broadens the sinc in proportion because ωt is constant. Here we will quantify this relationship between the signal width ΔT and its spectral bandwidth ΔF.

A pure sinusoidal wave has a clearly defined frequency, but it is spread over the infinitely long time axis. At the other extreme is a delta function, which is nicely compressed to a point on the time axis but contains a mixture of all frequencies. Here we examine how the width of a function in one domain relates to that in the other. By the end of this section, we will conclude that for any function, the time duration ΔT and the spectral bandwidth ΔF are related by

$$\Delta F \, \Delta T \quad \geq \quad 1 \tag{7.1}$$

This relationship is often called the *uncertainty principle*. Equality in (7.1) is observed for many simple functions.

7.1.1 The time-bandwidth maximum is unbounded

The *inequality* arises from all-pass filters. An all-pass filter leaves the spectrum unchanged, and hence ΔF unchanged, but it can spread out the time function arbitrarily, increasing ΔT arbitrarily.

7.1.2 Counter example refuted

It is easy to misunderstand the uncertainty principle. An oversimplification of it is to say that it is "impossible to know the frequency at any particular time." This oversimplification leads you to think about a truncated sinusoid, such as in Figure 7.1. You know the frequency exactly, so ΔF seems zero, whereas ΔT is finite, and this seems to violate (7.1). But what the figure shows is that the truncation of the sinusoid has broadened the frequency band. More particularly the impulse function in the frequency domain has been convolved by the sinc function which is the Fourier transform of the truncating rectangle function.

FIG. 7.1. (rand-windcos) Windowed sinusoid and its Fourier transform.

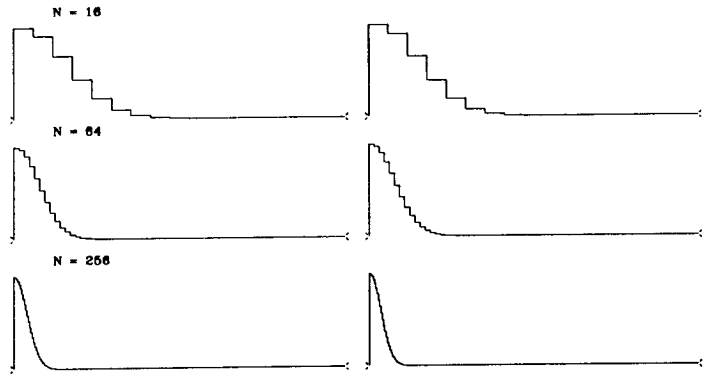

FIG. 7.2. (rand-uncertain) Sampled Gaussian functions and their Fourier transforms for in vectors of length n = 16, 64, and 256.

7.1.3 Measuring the time-bandwidth product

Now examine Figure 7.2 containing sampled Gaussian functions and their Fourier transforms. The Fourier transform of a Gaussian is well-known to be another Gaussian function as the plot confirms. I adjusted the width of each Gaussian so that it would be about equal in each of the two domains. The Gaussians were sampled at various values of n, increasing in steps by a factor of 4. You can measure the width dropping by a factor of 2 at each step. In real physical space as well as Fourier transform space, the object remains a constant size as the mesh is refined. Let us read from Figure 7.2 values for the widths ΔF and ΔT. On the top row with $n = 16$, I pick a width of about 4 points, and this seems to include about 90% of the area under the function. This suggests the rule $\Delta T = \sqrt{n}dt$ and $\Delta F = \sqrt{n}df$, which can be confirmed also on the plots with larger values of n. Using the relation between dt and df found in chapter 1 equation (1.25) that $dt\,df = 1/(2n)$, the product becomes $\Delta T \Delta F = \sqrt{n}\,dt\sqrt{n}df = 1/2$. It almost makes me want to replace the 1 in equation (7.1) by a half! I could increase my choices of Δ by $\sqrt{2}$ or I could redefine ΔF to include negative frequencies. But I do neither because small scale factors are not important here.

We could also confirm the inequality (7.1) by simple functions for which we know the analytic transforms. For example, consider an impulse function in time. Then $\Delta T = dt$ and the Fourier transform occupies the entire frequency band up to the Nyquist, i.e. $\Delta F = .5/dt$. Thus again the product is $\Delta T \Delta F = 1/2$.

7.1.4 Uncertainty principle in physics

The inequality (7.1) gets its name, the "uncertainty principle," from its interpretation in quantum mechanics. Observations of subatomic particles show they behave like waves with wave frequency proportional to particle momentum. The classical laws of mechanics enable prediction of the future of a mechanical system by extrapolation from the currently known position and momentum. But because of the wave nature of matter with momentum proportional to frequency, such prediction requires simultaneous knowledge of both the location and the frequency of a wave. This is impossible, as we see from (7.1), hence the word, "uncertainty."

7.1.5 Gabor's proof of the uncertainty principle

Although it is easy to verify the uncertainty principle in many special cases, it is not easy to deduce it. The difficulty begins from finding a definition of the width of a function that leads to a tractable analysis. One possible definition is to use a second moment, that is, to define ΔT by

$$(\Delta T)^2 \quad = \quad \frac{\int t^2\, b(t)^2\, dt}{\int b(t)^2\, dt} \tag{7.2}$$

The spectral bandwidth ΔF is defined likewise. With these definitions, Dennis Gabor prepared a widely reproduced proof. I'll omit his proof here; it is not an easy proof; it is widely available; and the definition (7.2) seems inappropriate for a function we often use, the sinc function, i.e. the FT of a step function. Since the sinc function drops off as t^{-1}, its width ΔT defined with (7.2) is infinity, which is far from more natural measures, such as for example, the distance to the first axis crossing.

7.1.6 My risetime proof of the uncertainty principle

In Fundamentals of Geophysical Data ProcessingI came up with a proof of the uncertainty principle that is appropriate for causal functions. Since this book is oriented towards causal functions, the proof is not well known, and I wanted to clean it up anyway; it is found here.

A similar and possibly more basic concept than the product of time and frequency spreads is the relationship between spectral bandwidth and *rise time* of a system response function. The *rise time* ΔT of a system response is defined as follows: When you kick a physical system with an impulse function, it usually responds rapidly rising to a some maximum level, and then drops off more slowly towards zero. The quantitative value of the rise time is also somewhat arbitrarily, generally taken to be span between the time of excitation and the time at which the system response is more than half way up to its maximum.

Tightness in the inequality (7.1) is associated with minimum phase. *Slackness* in the inequality (7.1) would arise from a filter with an additional all-pass component. Slackness could also come from a decay time that is more rapid than the rise time. Slackness could also result from other combinations of rises and falls such as random combinations. Minimum phase systems generally respond rapidly compared to the rate at which they later decay. Focusing our attention on such systems, we can now seek to derive the inequality (7.1) applied to rise time and bandwidth.

The first step is to choose a definition for rise time. I have found a tractable definition of rise time to be

$$\frac{1}{\Delta T} \quad = \quad \frac{\int_0^\infty \frac{1}{t}\, b(t)^2\, dt}{\int_0^\infty b(t)^2\, dt} \tag{7.3}$$

where $b(t)$ is the response function under consideration. Equation (7.3) defines ΔT by the first negative moment. Since this is unfamiliar, consider two examples. Taking $b(t)$ to be a step function, the numerator integral diverges, giving the desired $\Delta T = 0$ rise time. For another example, take $b(t)^2$ to grow linearly from zero to t_0 and then vanish. Then the rise time ΔT is $t_0/2$, again reasonable. It is curious that $b(t)$ could grow as $\sqrt{t}$ which rises with infinite slope at $t = 0$.

Proof by way of the dual problem

Although the Z transform method is a great aid in studying cases where divergence (as $1/t$) plays a role, it has the disadvantage that it destroys the formal interchangeability between the time domain and the frequency domain. To take advantage of the analytic simplicity of the Z transform we consider instead the dual to the rise-time problem. Instead of a time function whose square vanishes identically at negative time we have a spectrum $\overline{B}(1/Z)B(Z)$ which vanishes at negative frequencies. We measure how fast this spectrum can rise after $\omega = 0$. We will find this to be related to the time duration ΔT of the complex time function b_t. More precisely, we will now define the lowest significant frequency component ΔF in the spectrum analogously to (7.3) to be

$$\frac{1}{\Delta F} \quad = \quad \int_{-\infty}^{\infty} \frac{1}{f}\, \overline{B} B\, df \quad = \quad \int_{-\infty}^{\infty} \overline{B} B\, \frac{d\omega}{\omega} \tag{7.4}$$

where we have assumed the spectrum is normalised, i.e. the zero lag of the auto-correlation of b_t is unity. Now recall the bilinear transform which gives various Z transform expressions for $(-i\omega)^{-1}$. The one we ordinarily use is the coefficients of $.5(1+Z)/(1-Z)$, namely, $(\ldots 0, 0, 0, 0.5, 1, 1, 1 \ldots)$. (For reference, see equation (5.13) as $\rho \to 0$.)

The pole right on the unit circle at $Z = 1$ causes some nonuniqueness. Because $1/i\omega$ is an imaginary odd frequency function, the expansion I want to insert into (7.4) is the odd function of time given by the coefficients of

$$\frac{1}{-i\omega} \quad = \quad \frac{(\cdots - Z^{-2} - Z^{-1} + 0 + Z + Z^2 + \cdots)}{2} \tag{7.5}$$

Inserting (7.5) into (7.4) gives

$$\frac{1}{\Delta F} \quad = \quad \frac{-i}{2\pi} \int_{-\pi}^{+\pi} \frac{1}{2} (\cdots - Z^{-2} - Z^{-1} + Z + Z^2 + \cdots)\, \overline{B}\left(\frac{1}{Z}\right) B(Z)\, d\omega \tag{7.6}$$

Since any integral around the unit circle of a Z-transform polynomial selects the coefficient of Z^0 of its integrand we have

$$\frac{1}{\Delta F} \quad = \quad \frac{-i}{2} \left[(s_{-1} - s_1) + (s_{-2} - s_2) + (s_{-3} - s_3) + \cdots \right] \tag{7.7}$$

where s_t is the autocorrelation function of b_t. This may be further expressed as a function of the imaginary part of the autocorrelation s_t as

$$\frac{1}{\Delta F} \quad = \quad \sum_{t=1}^{\infty} -\Im s_t \quad \leq \quad \sum_{t=1}^{\infty} |s_t| \tag{7.8}$$

The sum in (7.8) is like an integral representing area under the $|s_t|$ function. Imagine the $|s_t|$ function replaced by a rectangle function of equal area. This would define a ΔT_{auto} for the $|s_t|$ function. Any autocorrelation function satisfies $|s_t| < s_0$ and we have normalized $s_0 = 1$. Thus, we have

$$\frac{1}{\Delta F} \quad \leq \quad \sum_{t=1}^{\infty} |s_t| \quad = \quad \Delta T_{\text{auto}} \tag{7.9}$$

Finally, we must relate the duration of a time function ΔT to the duration of its autocorrelation ΔT_{auto}. Generally speaking, it is easy to find a long time function that has short autocorrelation. Just take an arbitrary short time function and convolve it by a lengthy all-pass filter. Conversely, you can't get a long autocorrelation function from a short time function. A good case would be the autocorrelation of a rectangle function that is a triangle. Nominally, the triangle seems to be twice as long, but considering that the triangle tapers down, it is reasonable to assert the ΔT's are the same. So we conclude

$$\Delta T_{\text{auto}} \quad \leq \quad \Delta T \tag{7.10}$$

Inserting into (7.9) we have the uncertainty relation

$$\Delta T \, \Delta F \quad \geq \quad 1 \tag{7.11}$$

A curious thing about this proof was the appearance of the imaginary part of a complex autocorrelation. In the Fourier dual we have a complex, one sided time function and its Hilbert phase.

EXERCISES:

1 Consider $B(Z) = [1 - (Z/Z_0)^n]/(1 - Z/Z_0)$ in the limit Z_0 goes to the unit circle. Sketch the time function and its squared amplitude. Sketch the frequency function and its squared amplitude. Choose ΔF and ΔT.

2 A time series made up of two frequencies may be written as

$$b_t = A\cos\omega_1 t + B\sin\omega_1 t + C\cos\omega_2 t + D\sin\omega_2 t$$

Given ω_1, ω_2, b_0, b_1, b_2, b_3 show how to calculate the amplitude and phase angles of the two sinusoidal components.

7.2 FOURIER TRANSFORMS OF RANDOM NUMBERS

Many real signals are complicated and barely comprehensible. In experimental work, we commonly transform such data. To better understand what this means, it will be worthwhile examining signals made from random numbers.

Figure 7.3 shows discrete Fourier transforms of random numbers. The basic conclusion from this figure is that transforms of random numbers look like more random numbers.

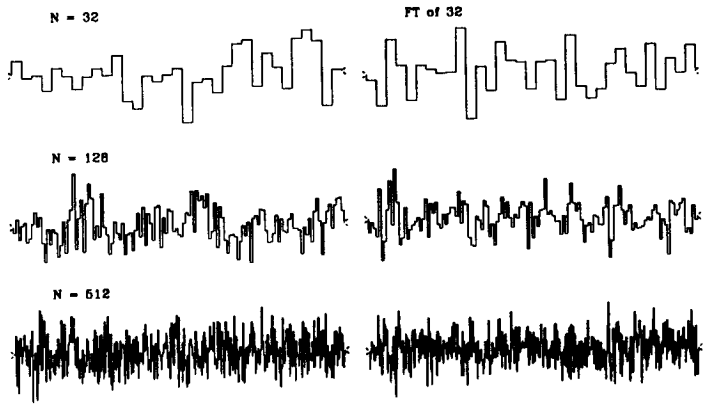

FIG. 7.3. (rand-nrand) Fourier cosine transforms of vectors containing random numbers. N is the number of components in the vector.

A random series containing all frequencies is called a *white* noise series because the color white is made from roughly equal amounts of all colors. Any series made by independently chosen random numbers is said to be an *independent* series. An independent series must be white, but a white series need not be independent.

Figure 7.4 shows Fourier transforms of random numbers surrounded by zeros, in other words, the random numbers have been *zero padded*. Since each vector of random numbers is the same length (1024 points including both sides of the even function with 513 points shown) the transforms are also the same length. The top time function has less randomness than the second trace (16 random numbers versus 64). Thus the top FT is smoother than the lower ones. The best way to understand this figure is to think of the left-hand signal as a frequency function. When higher frequencies are present, the right-hand signal oscillates faster.

7.2.1 Bandlimited noise

Figure 7.5 shows bursts of 25 random numbers at various shifts, and their Fourier transforms. You can think of either side of the figure as the time domain while the other side is the frequency domain. (See Chapter 1 page 8 for a description of the different ways of interpreting plots of one side of Fourier transform pairs of an even function.) I like to think of the left side as the Fourier domain and the right side as the time functions. Then the time functions seem to be sinusoids of a constant frequency (called the *center* frequency) and an amplitude that is modulated at a slower rate (called the beat frequency). Observe

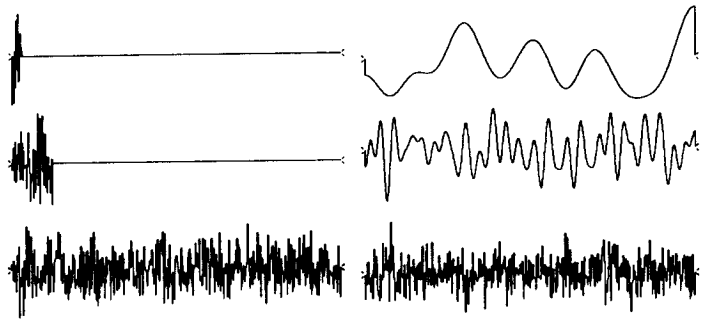

FIG. 7.4. (rand-pad) Zero padded random numbers and their Fourier transforms.

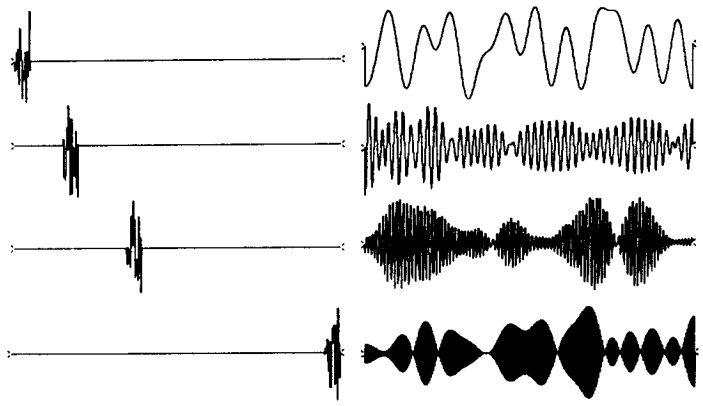

FIG. 7.5. (rand-shift) Shifted, zero padded random numbers in bursts of 25 numbers. (REMAKE WITH ALL BURSTS IDENTICAL.)

that the center frequency is related to the *location* of the random bursts and that the beat frequency is related to the *bandwidth* of the noise burst.

You can also think of Figure 7.5 as having one sided frequency functions on the left, and the right side as being the *real part* of the time function. The real parts are cosine like, whereas the imaginary parts (not shown) are sine like, with the same envelope function.

You might have noticed that the bottom plot in Figure 7.5 with the Nyquist frequency modulated beats, seems to have about twice as many beats as the two plots above it. This can be explained as an edge effect. The noise burst near the Nyquist frequency is really twice as wide as shown because it is mirrored about the Nyquist frequency into negative frequencies. Likewise, the top figure is not modulated at all, but the signal itself has a frequency that matches the beats on the bottom figure.

7.3 TIME-STATISTICAL RESOLUTION

If you flipped a fair coin 1000 times, it is unlikely that you would get exactly 500 "heads" and 500 "tails." More likely the head count would lie somewhere between 400 and 600. Or would it lie in another range? The theoretical value, called the *mean* or the *expectation* is 500. The value from your experiment is called the *sample mean*. How much difference Δm should we expect between the sample mean $\hat{m}$ and the true mean m? Both the coin flips x and your sample mean $\hat{m}$ are *random variables*. Your coin flip experiment could be repeated many times generally giving a different result each time. The expected squared difference between the mean and your sample mean, i.e. $(\Delta m)^2$ is called the *variance of the sample mean*. For a larger sample, the sample mean should be proportionately much closer to the true mean than for a smaller sample. This will lead us to an inequality constraining the product between Δm and the sample size $\Delta T = n\,\Delta t$.

7.3.1 Ensemble

The "true value" of the mean could be defined by flipping the coin n times and conceiving of n going to infinity. A more convenient definition of "true value" is that the experiment is imagined as having been done separately under identical conditions by an infinite number of people (an ensemble). The *ensemble* may seem a strange construction of the imagination, nonetheless much literature in statistics and in the natural sciences uses the idea of an ensemble. The ensemble idea enables us to define a time-variable mean for coins that change with time.

7.3.2 Expectation

A conceptual average over the ensemble, called an expectation, is denoted by the symbol E. The index for summation over the ensemble is never shown explicitly; every random variable is presumed to have one. Thus, the true mean at time t is defined as $m_t = E(x_t)$. When the mean does not vary with time, we have

$$m = E(x_t) \quad (\text{all } t) \tag{7.12}$$

7.3.3 Variance

Likewise, we may be interested in a property of x_t called its *variance* which is a measure of variability about the mean defined by $\sigma_t^2 = E[(x_t - m_t)^2]$. The x_t random numbers could be defined in such a way that σ or m or both is either time-variable or constant. If both are constant, we have

$$\sigma^2 = E[(x_t - m)^2] \tag{7.13}$$

When manipulating algebraic expressions the symbol E behaves like a summation sign, namely

$$E \equiv (\lim N \to \infty) \frac{1}{N} \sum_1^N \tag{7.14}$$

Note that the summation index is not given, since the sum is over the ensemble, not time.

7.3.4 Sample mean

Now let x_t be a time series made up from (identically distributed, independently chosen) random numbers in such a way that m and σ do not depend on time. Suppose we have a sample of n points of x_t and are trying to determine the value of m. We could make an estimate $\hat{m}$ of the mean m with the formula

$$\hat{m} = \frac{1}{n} \sum_{t=1}^{n} x_t \tag{7.15}$$

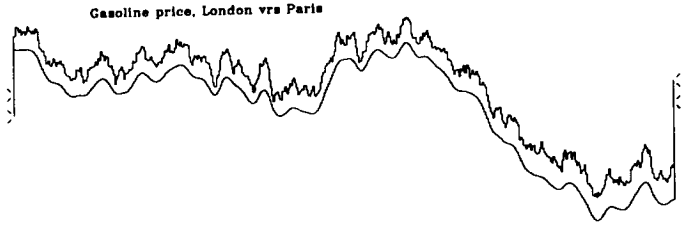

FIG. 7.6. (rand-walk) Random walk and itself smoothed (and shifted downward).

A somewhat more elaborate method of estimating the mean would be to take a weighted average. Let w_t define a set of weights normalized so that

$$\sum w_t = 1 \tag{7.16}$$

With these weights the more elaborate estimate $\hat{m}$ of the mean is

$$\hat{m} = \sum w_t x_t \tag{7.17}$$

Actually (7.15) is just a special case of (7.17) where the weights are $w_t = 1/n$.

Further, the weights could be *convolved* on the random time series. The weights are simply a filter response where the filter coefficients happen to be positive and cluster together. An example in Figure 7.6 shows a random walk function with itself smoothed locally.

7.3.5 Variance of the sample mean

Our objective here is to calculate how far the estimated mean $\hat{m}$ is likely to be from the true mean m for a sample of length n. This departure is called the *variance of the sample mean*, and it is given by $(\Delta m)^2$, where

$$(\Delta m)^2 = E[(\hat{m} - m)^2] \tag{7.18}$$

$$= E\left\{[(\sum w_t x_t) - m]^2\right\} \tag{7.19}$$

Now use the fact that $m = m \sum w_t = \sum w_t m$

$$(\Delta m)^2 = E\left\{\left[\sum_t w_t (x_t - m)\right]^2\right\} \tag{7.20}$$

$$= E\left\{\left[\sum_t w_t (x_t - m)\right]\left[\sum_s w_s (x_s - m)\right]\right\} \tag{7.21}$$

$$= E\left[\sum_t \sum_s w_t w_s (x_t - m)(x_s - m)\right] \tag{7.22}$$

The step from (7.21) to (7.22) follows because generally

$$(a_1 + a_2 + a_3)(a_1 + a_2 + a_3) = \text{sum of} \begin{bmatrix} a_1a_1 & a_1a_2 & a_1a_3 \\ a_2a_1 & a_2a_2 & a_2a_3 \\ a_3a_1 & a_3a_2 & a_3a_3 \end{bmatrix} \tag{7.23}$$

The expectation symbol E may be regarded as another summation that can be done after, as well as before, the sums on t and s, so

$$(\Delta m)^2 = \sum_t \sum_s w_t w_s E[(x_t - m)(x_s - m)] \tag{7.24}$$

By the randomness of x_t and x_s the expectation on the right, that is, the sum over the ensemble, gives zero unless $s = t$. If $s = t$, then the expectation is the variance defined by (7.13). Expressing the result in terms of the Kronnecker delta, δ_{ts} (which equals unity if $t = s$, and vanishes otherwise) gives

$$(\Delta m)^2 = \sum_t \sum_s w_t w_s \sigma^2 \delta_{ts} \tag{7.25}$$

$$(\Delta m)^2 = \sum_t w_t^2 \sigma^2 \tag{7.26}$$

$$\Delta m = \sigma \sqrt{\sum_t w_t^2} \tag{7.27}$$

For n weights, each of size $1/n$, the variance of the sample mean is Then

$$\Delta m = \sigma \sqrt{\sum_{t=1}^{n} \left(\frac{1}{n}\right)^2} = \frac{\sigma}{\sqrt{n}} \tag{7.28}$$

This is the most important property of random numbers that is not intuitively obvious. Let us reexpress the result (7.28) in informal language. Given a sum s of random terms

$$s = \pm 1 \pm 1 \pm 1 \cdots \quad (n \text{ terms}) \tag{7.29}$$

You are likely to find $|s| \approx \sqrt{n}$. Technically, the expectation of s is zero, and the *variance of the sample mean* of s is $\sqrt{n}$.

7.3.6 Dilemma in estimating a time variable mean

If you are trying to estimate the mean of a random series that has a time-variable mean then you faces a basic dilemma. Including many numbers in the sum to get Δm to be small conflicts with the possibility of seeing m_t change during the measurement. This is the dilemma faced by a stockbroker when a client tells him, "Since the market fluctuates a lot I'd like you to buy me some stock when the market is below its mean and then sell the stock when the market is above its mean."

Taking a time series to be sampled every dt seconds and letting $\Delta T = n\,dt$ denote the length of the sample, then (7.28) becomes

$$\frac{(\Delta m)^2}{\sigma^2}\frac{\Delta T}{dt} = 1 \tag{7.30}$$

It is desirable to have both Δm and ΔT as small as possible. As with the other uncertainty principle, things could be worse. If the original random numbers x_t were correlated with one another, for example, if x_t were an approximation to a continuous function, then the sum of the n numbers would not cancel as $\sqrt{n}$. Thus the inequality

$$\frac{(\Delta m)^2}{\sigma^2}\frac{\Delta T}{dt} \geq 1 \tag{7.31}$$

The inequality (7.31) may be called an uncertainty relation between accuracy and time resolution.

7.3.7 Variance of the sample variance

The *variance of the sample variance* arises in many contexts. Suppose you want to measure the storminess of the ocean. You measure water level as a function of time and subtract the mean. The storminess is the variance about the mean. You measure the storminess in one minute and call it a sample storminess. You compare it to other minutes and other locations and you find they are not all the same. To characterize these differences, you need the *variance of the sample variance*.

Given a sample of a zero-mean random time series x_t, we define another series y_t by $y_t = x_t^2$. The problem of estimating the variance $\sigma^2 = p$ of x_t is identical to the problem of estimating the mean $m = p$ of y_t. As always, we anticipate an error Δp in our estimate of the variance. Recalling equation (7.28) we see the variance of the sample variance is defined by $(\Delta p)^2$ where

$$\Delta p = \frac{p}{\sqrt{n}} \tag{7.32}$$

Since $\Delta T = n\,dt$, by squaring we have

$$\left(\frac{\Delta p}{p}\right)^2 \frac{\Delta T}{dt} \geq 1 \tag{7.33}$$

The *inequality* could come into effect if the random numbers x_t were not totally unpredictable. If x_t were an approximation to a continuous function, then it would be predictable and there would be slack in the inequality.

Variance of the sample variance ad infinitum

Another more esoteric occurrence of the variance of the sample variance arises when you get fussy about error bounds. In a scientific paper may find

$$m = \hat{m} \pm \Delta m \tag{7.34}$$

$$= \hat{m} \pm \sigma/\sqrt{n} \tag{7.35}$$

but since the variance σ^2 often is not known either, it is necessary to use the estimated $\hat{\sigma}$, that is

$$m = \hat{m} \pm \hat{\sigma}/\sqrt{n} \tag{7.36}$$

Of course (7.36) really is not right because we really should add something to show the additional uncertainty from error in $\hat{\sigma}$. This estimated error would again have an error called the *variance of the sample variance*. People often neglect this, or it can be calculated from a presumed probability function.

EXERCISES:

1 Suppose the mean of a sample of random numbers is estimated by a triangle weighting function, i.e.,

$$\hat{m} = s\sum_{i=0}^{n}(n-i)\,x_i$$

Find the scale factor s so that $E(\hat{m}) = m$. Calculate Δm. Define a reasonable ΔT. Examine the uncertainty relation.

2 A random series x_t with a possibly time-variable mean may have the mean estimated by the feedback equation

$$\hat{m}_t = (1-\epsilon)\hat{m}_{t-1} + bx_t$$

a. Express $\hat{m}_t$ as a function of $x_t, x_{t-1}, \ldots$, and not $\hat{m}_{t-1}$.

b. What is ΔT, the effective averaging time?

c. Find the scale factor b so that if $m_t = m$, then $E(\hat{m}_t) = m$.

d. Compute the random error $\Delta m = \sqrt{E(\hat{m}-m)^2}$. [HINT: Δm goes to $\sigma\sqrt{\epsilon/2}$ as $\epsilon \to 0$].

e. What is $(\Delta m)^2 \Delta T$ in this case?

3 Show that

$$(\Delta P)^2 = \frac{1}{n}[E(x^4) - \sigma^4]$$

7.4 SPECTRAL FLUCTUATIONS

Observations of sea level for a long period of time can be summarized in terms of a few statistical averages such as the mean height m and the variance σ^2. Another important kind of statistical average for use on such geophysical time series is the *power spectrum*. Many mathematical models explain only such statistical averages of data and not the data themselves. To recognize certain pitfalls and understand certain fundamental limitations on work with power spectra, we first consider the idealized example of random numbers.

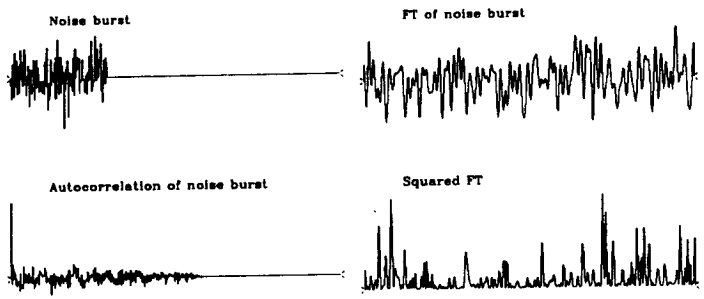

FIG. 7.7. (rand-auto) Autocorrelation and spectrum of random numbers.

Figure 7.7 shows a signal that is a burst of noise, its Fourier transform, and the transform squared, and its inverse transform, the autocorrelation. Here the FT squared is the same as the more usual FT times its complex conjugate—because the noise burst signal is even so its FT is real.

Notice that the autocorrelation has a big spike at zero lag. This spike represents the correlation of the random numbers with themselves. The other lags are much smaller. They represent the correlation of the noise burst with itself shifted. Theoretically, the noise burst is not correlated with itself shifted and these small fluctuations result from the finite extent of the noise sample.

Imagine many other copies of Figure 7.7. Ensemble averaging would be adding these other autocorrelations or equivalently adding these other spectra. The fluctuations aside the central lobe of the autocorrelation would be destroyed by ensemble averaging. The fluctuations in the spectrum would be smoothed out by ensemble averaging. The *expectation* of the autocorrelation is that it is an impulse at zero lag. The *expectation* of the spectrum is that it is a constant, namely

$$E[\hat{S}(Z)] \quad = \quad S(Z) \quad = \quad \text{const} \tag{7.37}$$

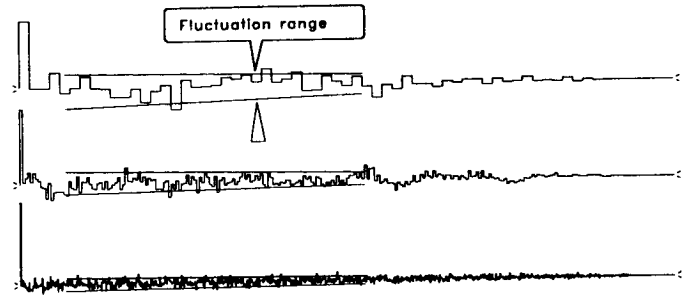

FIG. 7.8. (rand-fluct) Autocorrelation as a function of number of data points. The random noise series (even) lengths are 60, 240, 960.

7.4.1 Paradox: $N \to \infty$ vrs the ensemble average

Now for the paradox. Imagine $n \to \infty$ in Figure 7.7. Will we see the same limit as the ensemble average? Here are two contradictory points of view:

- For increasing n, the spectrum does not get any smoother because the FT's do not get any smoother as Figure 7.3 on page 7 shows.
- For increasing n, the fluctuations on the non-zero autocorrelation lags get smaller and smaller, so the spectrum = s_0 + (vanishing terms) becomes a constant.

To investigate this further, Figure 7.8 shows the autocorrelation samples as a function of n in steps of n increasing by factors of four. Thus $\sqrt{n}$ increases by factors of two. It looks as though the sample variance for the near lags of the autocorrelation compared to the zeroth lag, drop off by about $s_k = s_0/\sqrt{n}$. This is predicted by the *variance of sample variance* equation (7.32), assuming that the numbers $s_k = \sum_1^n x_i x_{i+k}$ are independently chosen random numbers. (They are random but who knows if they are independent?)

Also, we notice that the fluctuations drop off with lag. The drop off goes to zero at a lag equal the sample length. This is because the number of terms in the crosscorrelation drops as lag increases. A first impression is that the autocorrelation fits a triangular envelope. More careful inspection shows that the triangle bulges upward at wide offsets (clearer on Figure 7.7). The count of the nonzero terms in the autocorrelation at lag k, namely s_k, is $n-k$. So, presuming the s_k are random and they add as random numbers do, the expected size as a function of lag is proportional to $\sqrt{n-k}$. So s_k is expected to be in the range

$$|s_k| \quad \approx \quad s_0 \frac{1}{\sqrt{n}} \frac{\sqrt{n-k}}{\sqrt{n}} \quad = \quad s_0 \frac{\sqrt{n-k}}{n} \tag{7.38}$$

Technically, (7.38) states the variance of the sample variance, *Deltap'* of equation (7.32). So the fluctuation does go to zero as the sample length increases.

But the paradox only heightens, how can the autocorrelation go to an impulse without the spectrum going to a constant?

Recall the delta function, a rectangle of width ϵ and height $1/\epsilon$, so as $\epsilon \to 0$, the integral of the delta function is unity. Something the opposite is happening with the fluctuation flanks of the autocorrelation. The fluctuation size is dropping with $\sqrt{n}$ but the *count of the terms* in $S(\omega) = s_0 + s_1 \cos\omega + s_2 \cos 2\omega + \cdots$ is increasing with n. Indeed the *area* seems to *increase* as $\sqrt{n}$ until we account for the randomness of the values being summed, which drops the area in the fluctuations to the area of the impulse at zero lag. Thus $S(\omega) = s_0 \pm s_0$. This resolves the paradox. The spectrum does remain forever fuzzy while the autocorrelation tends to an impulse. In conclusion, the expectation (ensemble average) of the spectrum is not properly estimated by letting $n \to \infty$ in a sample.

Letting n go to infinity does not take us to the expectation $\hat{S} = \sigma^2$. The problem is, as we increase n, we increase the frequency resolution but not the statistical resolution. To increase the statistical resolution we need to simulate ensemble averaging. There are two ways to do this:

1. Take the sample of n points and break it into k equal-length segments of n/k points each. Compute an $S(\omega)$ for each segment and then add all k of the $S(\omega)$ together.
2. Form $S(\omega)$ from the n-point sample. Of the $n/2$ independent amplitudes, replace each one by an average over its k nearest neighbors. This could also be done by tapering the autocorrelation.

The second method is shown in Figure 7.9. This figure shows a noise burst of 240 points. Since the time function is even, the noise burst is effectively 480 points wide, so the autocorrelation is 480 points from center to end. The spectrum is very rough. Multiplying the autocorrelation by a triangle function effectively smooths the spectrum by a sinc squared function. The first taper takes the autocorrelation width from 480 lags to 120 lags. Thus the spectral fluctuations should drop by a factor of 2. The next weighted autocorrelation has its width dropped from 480 to 30 lags. Thus spectral roughness should drop by another factor of 2. As you can see, the reduction in spectral fluctuation $\Delta p/p$ is accompanied by a reduction in spectral resolution ΔF.

Convergence seems slow, but the results would look twice as smooth if *amplitudes* were plotted instead of *power*. The reason is this: Let power P equal squared amplitude $A^2 = P$. Thus $dP/dA = 2A\,dA$, or

$$\frac{\Delta P}{P} = 2\frac{\Delta A}{A} \tag{7.39}$$

7.4.2 Spectral estimation

Luckily in Figure 7.9 we didn't care about spectral resolution, since we knew theoretically that the spectrum was white. But in practice we don't have such foreknowledge. Indeed, the random factors we deal with in nature rarely are white. A widely used model for naturally occurring random functions, such as microseism, or sometimes reflection seismograms, is

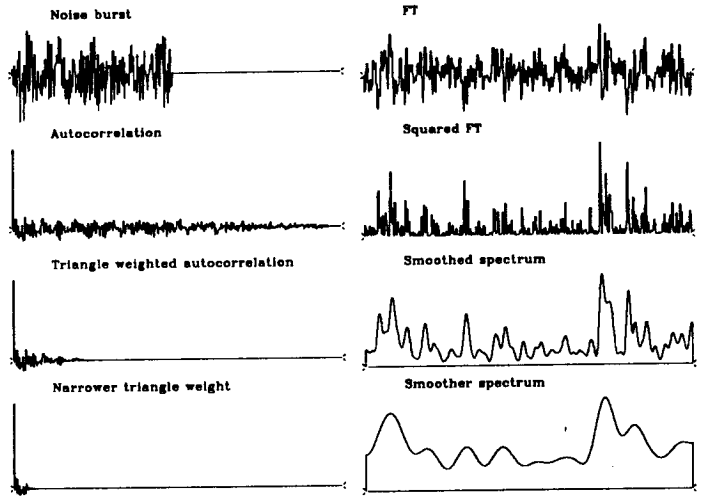

FIG. 7.9. (rand-taper) Spectral smoothing by tapering the autocorrelation.

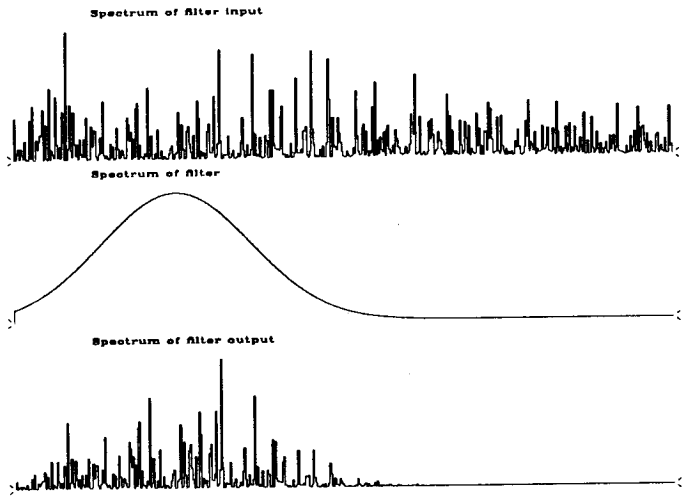

FIG. 7.10. (rand-filter) Random numbers into a filter. WILL MIMIC FGDP p 81, Fig 4-5, BUT WITH FAVORITE WAVELET.

that they may be modeled by putting white noise into a filter. The spectra for an example of this type is shown in Figure 7.10. You can see that smoothing the envelope of the power spectrum of the output gives an estimate of the spectrum of the filter. But you also see that the estimate may need even more smoothing.

I PLAN TO MAKE SEISMIC EXAMPLE WITH MY FAVORITE WAVELET AND n=1024 AND amplitude spectra.

7.4.3 Time-frequency-statistical resolution

Equation (7.33) applied to a single frequency, namely the zero frequency. But the principles underlying equation (7.33) apply to all frequencies. Therefore

$$\Delta F \, \Delta T \left(\frac{\Delta p}{p} \right)^2 \quad > \quad 1 \tag{7.40}$$

7.5 CROSSCORRELATION AND COHERENCY

With two time series we can see how they are related.

7.5.1 Correlation

Correlation is a concept similar to cosine. A cosine measures the angle between two vectors. It is given by the dot product of the two vectors divided by their magnitudes

$$c \quad = \quad \frac{(\mathbf{x} \cdot \mathbf{y})}{\sqrt{(\mathbf{x} \cdot \mathbf{x})(\mathbf{y} \cdot \mathbf{y})}} \tag{7.41}$$

Correlation is similar, but x and y are scalar random variables instead of vectors so the summation is an expectation instead of a dot product.

$$c \quad = \quad \frac{E(xy)}{\sqrt{E(x^2)\, E(y^2)}} \tag{7.42}$$

There is a practical difficulty when the ensemble averaging is simulated over a sample. The problem arises with small samples and is most dramatically illustrated for a sample with only one element. Then the sample correlation is

$$\hat{c} \quad = \quad \frac{xy}{|x|\,|y|} \quad = \quad \pm 1 \tag{7.43}$$

regardless of what value the random number x or the random number y should take. For any n, the sample correlation $\hat{c}$ scatters away from zero. Such scatter is called *bias*. The topic of bias and variance of coherency estimates is a complicated one, but a rule of thumb seems to be to expect bias and variance of $\hat{c}$ about $1/\sqrt{n}$ for samples of size n. Bias, no doubt, accounts for many false "discoveries" since cause-and-effect is often inferred from correlation.

7.5.2 Coherency

The concept of *coherency* in time-series analysis is analogous to correlation. Taking x_t and y_t to be time series, they may have a mutual relationship that depends on time-delay, scaling, or even filtering. For example, perhaps $Y(Z) = F(Z)X(Z) + N(Z)$ where $F(Z)$ is a filter and n_t is unrelated noise. The generalization of the correlation concept is to define *coherency* by

$$C = \frac{E\left[X\left(\frac{1}{Z}\right)Y(Z)\right]}{\sqrt{E(\overline{X}X)\,E(\overline{Y}Y)}} \tag{7.44}$$

Correlation is a real scalar. *Coherency* is a complex function of frequency and it expresses the frequency dependence of correlation. In forming an estimate of coherency it is always essential to simulate ensemble averaging. Note that if the ensemble averaging were to be omitted, the coherency (squared) calculation would give

$$|C|^2 = \overline{C}C = \frac{(\overline{\overline{X}Y})(\overline{X}Y)}{(\overline{X}X)(\overline{\overline{Y}Y})} = 1 \tag{7.45}$$

which states that the coherency squared is unity independent of the data. Because correlation scatters away from zero we find that coherency squared is biased away from zero.

POSSIBLE BI-SPECTRUM EXAMPLE....

7.6 PROBABILITY AND THE CENTRAL-LIMIT THEOREM

One way to obtain random integers from a known probability function is to write integers on slips of paper and place them in a hat. Draw one slip at a time. After each drawing replace the slip in the hat. The probability of drawing the integer i is given by the ratio a_i of the number of slips containing the integer i divided by the total number of slips. Obviously the sum over i of a_i must be unity. Another way to get random integers is to throw one of a pair of dice. Then all a_i equal zero except $a_1 = a_2 = a_3 = a_4 = a_5 = a_6 = \frac{1}{6}$. The probability that the integer i will occur on the first drawing and the integer j will occur on the second drawing is $a_i a_j$. If you draw two slips or throw a pair of dice, then the probability that the sum of i and j equals k is readily seen to be

$$c_k = \sum_i a_i a_{k-i} \tag{7.46}$$

Since this equation is a convolution, we may look into the meaning of the Z transform

$$A(Z) = \cdots a_{-1}Z^{-1} + a_0 + a_1 Z + a_2 Z^2 + \cdots \tag{7.47}$$

In terms of Z transforms the probability that i plus j equals k is simply the coefficient of Z^k in

$$C(Z) = A(Z)\,A(Z) \tag{7.48}$$

7.6.1 The central-limit theorem

The central-limit theorem of probability and statistics is perhaps the most important theorem in the fields of probability and statistics. A derivation of the central limit theorem explains why the Gaussian probability function is so frequently encountered in nature; not just in physics but also in the biological and social sciences. No experimental scientist should be unaware of the basic ideas behind this theorem. Although the result is deep and is even today the topic of active research, we can quickly go to the basic idea.

From equation (7.48), if we add n random numbers, the probability that the sum of them equals k is given by the coefficient of Z^k in

$$G(Z) = |A(Z)|^n \tag{7.49}$$

The central-limit theorem of probability says that as n does to infinity the polynomial $G(Z)$ goes to a special form, almost regardless of the specific polynomial $A(Z)$. The specific form is such that a graph of the coefficients of $G(Z)$ comes closer and closer to fitting under the envelope of the bell-shaped Gaussian function. Let us see why this happens. Our development will lack a mathematical rigor because the theorem is not always true. There are pathological A functions which do not result in G tending to Gaussian. Although the pathological cases do arise in applications, space limits us here to the usual case.

FIG. 7.11. (FGDP fig 4-8) The complex numbers $a_k e^{i\omega k}$ added together.

Consider the size of $A(Z)$ for real ω. If $\omega = 0$, the sum of the terms of $A(Z)$ may be visualized in the complex plane as a sum of vectors $a_k e^{i\omega k}$ all pointing in the positive real direction. If $\omega \neq 0$ the vectors point in different directions. This is shown in Fig. 7.11.

In raising $A(e^{i\omega})$ to the nth power, the values of ω of greatest concern are those near $\omega = 0$ where A is largest—because in any region where A is small A^n will be extremely small. Near $\omega = 0$ or $Z = 1$ we may expand $A(Z)$ in a power series in ω

$$A(e^{i\omega}) = A|_0 + \left.\frac{\partial A}{\partial \omega}\right|_0 \omega + \left.\frac{\partial^2 A}{\partial \omega^2}\right|_0 \frac{\omega^2}{2!} + \cdots \tag{7.50}$$

Note that the coefficients of this power series are proportional to the moments m_i of the probability function; that is

$$A(e^{i\omega}) = \sum_k a_k e^{ik\omega} \tag{7.51}$$

$$A(1) = \sum_k a_k = m_0 = 1 \tag{7.52}$$

$$\frac{\partial A}{\partial \omega} = \sum_k ika_k e^{ik\omega} \tag{7.53}$$

$$\left.\frac{\partial A}{\partial \omega}\right|_0 = \sum_k ika_k = im_1 \tag{7.54}$$

$$\left.\frac{\partial^2 A}{\partial \omega^2}\right|_0 = -\sum k^2 a_k = -m_2 \tag{7.55}$$

After raising $A(Z)$ to the nth power we conjecture only the first three terms of the power series expansion are important. (This fails if any of the moments of the probability function are infinite.) Thus, as far as G is concerned, the only important things about A are its mean value $m = m_1$ and its second moment m_2. If this is really so, we may calculate G by replacing A with any function B having the same mean and same second moment as A. We may use the simplest function we can find. A good choice is the so called binomial probability function given by

$$B = \frac{Z^m(Z^\sigma + Z^{-\sigma})}{2} \tag{7.56}$$

$$= \frac{e^{i(m+\sigma)\omega} + e^{i(m-\sigma)\omega}}{2} \tag{7.57}$$

Let us verify its first moment

$$\frac{\partial B}{\partial \omega} = \frac{i\left[(m+\sigma)e^{i(m+\sigma)\omega} + (m-\sigma)e^{i(m-\sigma)\omega}\right]}{2} \tag{7.58}$$

$$\left.\frac{\partial B}{\partial \omega}\right|_0 = im \tag{7.59}$$

Now let us verify its second moment

$$\left.\frac{\partial^2 B}{\partial \omega^2}\right|_0 = -\frac{(m+\sigma)^2 + (m-\sigma)^2}{2} \tag{7.60}$$

$$\left.\frac{\partial^2 B}{\partial \omega^2}\right|_0 = -(m^2 + \sigma^2) \tag{7.61}$$

Hence, σ should be chosen so that

$$m_2 = m^2 + \sigma^2 \tag{7.62}$$

Of course, we cannot expect that m and σ will necessarily turn out to be integers; therefore (7.56) will not necessarily be a Z transform in the usual sense. It does not really matter; we simply interpret (7.56) as saying:

1. The probability of drawing the number $m + \sigma$ is one-half.
2. The probability of $m - \sigma$ is one-half.
3. The probability of any other number is zero.

Now, raising $(Z^\sigma + Z^{-\sigma})$ to the nth power gives a series in powers of Z^σ whose coefficients are symmetrically distributed about Z to the zero power and whose magnitudes are given by the binomial coefficients. A sketch of the coefficients of $B(Z)^n$ is given in Figure 2.8. That the function and its Fourier transform tend to the Gaussian function $e^{-.5\sigma t^2}$ seems visually apparent from the figure, and it is shown mathematically in Fundamentals of Geophysical Data Processingas well as many textbooks.

Chapter 9

THE CONJUGATE TRANSFORMATION

Geophysical modeling calculations generally use linear operators that predict data from models. Our usual task is finding the inverse to these calculations, i.e., finding models (or making maps) from the data. Logically, the *conjugate operator* is the first step and a part of all subsequent steps of this inversion process. Surprisingly, in practice the conjugate sometimes does a better job than the inverse! This is because the conjugate operator tolerates imperfections in the data and doesn't demand the data provide full information.

Using the methods of this chapter, you will find that once you grasp the relationship between operators in general and their conjugates, you can have the conjugate just as soon as you have learned how to code the modeling operator.

When the conjugate is *not* an adequate approximation to the inverse, then you apply the techniques of fitting and optimization in the previous chapter which require iterative use of the modeling operator and its conjugate.

If you will offer me a poet's license with words, I will offer you this table of operators and their conjugates.

matrix multiply	conjugate-transpose matrix multiply
convolution	crosscorrelation
stretching	squeezing
zero padding	truncation
causal integration	anticausal integration
plane wave superposition	slant stack
summing along curve	adding onto a curve
downward continuation	upward continuation
diffraction modeling	imaging by migration
hyperbola modeling	CDP stacking
ray tracing	tomography

Continuing on with my poet's license, I'll say that conjugate operators *undo* the time and phase shifts of modelling operators. The inverse operator does this, but it also divides out the color. For example, with linear interpolation, high frequencies are smoothed out,

so inverse interpolation must restore them. You can imagine the possibilities for noise amplification. That is why conjugates are safer than inverses. But nature determines in each application what is the best operator to use, to stop after the conjugate, to go the whole way to the inverse, or to stop part way.

We'll see that computation of the conjugate is a straightforward adjunct to the computation itself, and the computed conjugate should be, and generally can be, exact (within machine precision). If the application's operator is computed in an approximate way, we'll see that it is natural and best to compute the conjugate with conjugate approximations.

9.1 FAMILIAR OPERATORS

To establish concepts we'll first see how to write programs for familiar operators, matrix multiply, filtering and correlation, zero padding with Fourier transform and causal integration. Then we will see how to test the conjugacy of the programs.

9.1.1 Matrix multiply

The conjugate operation to multiplying by a matrix is multiplying by the transposed matrix (unless the matrix has complex elements, in which case you need the complex-conjugated transpose). The following pseudocode does matrix multiplication $\mathbf{y} = \mathbf{A}\mathbf{x}$ and multiplication by the transpose matrix $\tilde{\mathbf{x}} = \mathbf{A}'\mathbf{y}$.

```
if not transpose
        then erase y
if transpose
        then erase x
do iy = 1, ny {
do ix = 1, nx {
        if not transpose
                y(iy) = y(iy) + b(iy,ix) × x(ix)
        if transpose
                x(ix) = x(ix) + b(iy,ix) × y(iy)
}}
```

The above example is a prototype of many to follow, so observe carefully the similarities and differences between the operation and its conjugate.

9.1.2 Transient convolution

When the matrix has a special form, such as

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} = \begin{bmatrix} b_0 & 0 & 0 \\ b_1 & b_0 & 0 \\ b_2 & b_1 & b_0 \\ b_3 & b_2 & b_1 \\ 0 & b_3 & b_2 \\ 0 & 0 & b_3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \tag{9.1}$$

then the matrix multiplication and transpose multiplication still fit easily in the same computational framework. The operation $\mathbf{B}\mathbf{x}$ *convolves* b_t with x_t whereas the operation $\mathbf{B}'\mathbf{y}$ *crosscorrelates* b_t with y_t. I'll leave it to you to verify that

```
do ix = 1, nx {
do ib = 1, nb {
        iy = ix + ib - 1
        if not transpose
                y(iy) = y(iy) + b(ib) × x(ix)
        if transpose
                x(ix) = x(ix) + b(ib) × y(iy)
}}
```

Notice the "bottom line" in the program is that x and y are simply interchanged.

9.1.3 Zero padding is the conjugate of truncation

Here we will see that surrounding a data set by zeros (*zero padding*) is conjugate to throwing away the extended data (truncation). Consider the example of Fourier transformation. The Fourier transform uses *complex* numbers so its conjugate is the complex conjugate of its matrix transpose. Henceforth, we denote the conjugate transpose with a prime, i.e. $\mathbf{F}'$. With Fourier transforms, the inverse is the conjugate, i.e. the complex conjugate of the transpose. This is seen in equation (1.13). Thus $\mathbf{F}'\mathbf{F} = \mathbf{I}$.

With Fourier transforms, padding and truncation are particularly prevalent. Most programs transform a data set of length of 2^n whereas data set lengths are often of length $m \times 100$. So the practical approach is to pad given data with zeros. Notice that padding data with zeros is the same as matrix multiplication by a rectangular matrix with an identity matrix $\mathbf{I}$ on top and and a zero matrix $\mathbf{0}$ on the bottom. Padding followed by Fourier transformation $\mathbf{F}$ can be expressed in matrix algebra as

$$\text{Program} = \mathbf{F} \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \end{bmatrix} \tag{9.2}$$

The conjugate transpose program is now

$$\text{Program}' = \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix} \mathbf{F}' \tag{9.3}$$

So we see the conjugate program *truncates* the data *after* the inverse Fourier transform. Likewise, if a program pads real data values to complex values (with zero imaginary parts), then the conjugate program truncates the complex values back to reals. So we see the mathematical conjugate suggests a program that does some of the friendly things you might want a program to do anyway.

9.1.4 Conjugate of causal integration is anticausal

The causal integration operator is like a matrix with ones below the diagonal and zeros above. So the conjugate to causal integration is anti-causal integration. In chapter 3 we wrote the Z-transform expression for causal integration as $(1 + \rho Z)/(1 - \rho Z)$. Thus the conjugate operator is $(1 + \rho/Z)/(1 - \rho/Z)$. So replacing Z by $1/Z$ is like taking the complex conjugate of $e^{i\omega}$.

9.1.5 Inner product tests

Programs for solving big matrices do not store the matrix in memory. Instead, the user supplies two subroutines, one to multiply $\mathbf{Ax}$ and another to do $\mathbf{A'y}$. (The solving program chooses various vectors for $\mathbf{x}$ and $\mathbf{y}$.) If you erroneously supply routines in which the matrices are not really transposes, then convergence is unlikely. How can you be assured you have supplied a consistent pair of routines?

The associative property of linear algebra says $\mathbf{y'(Ax)} = \mathbf{(y'A)x}$ which may be written

$$\mathbf{y'(Ax)} \quad = \quad \mathbf{(A'y)'x} \tag{9.4}$$

(In general, the matrix is not square.) For the test, you take the vectors $\mathbf{x}$ and $\mathbf{y}$ to contain random numbers. Form the vectors $\mathbf{\tilde{y}} = \mathbf{Ax}$ and $\mathbf{\tilde{x}} = \mathbf{A'y}$. Check whether $\mathbf{y'\tilde{y}} = \mathbf{\tilde{x}'x}$.

I examined (9.4) on many operators and was surprised and delighted to find that it is normally satisfied to near the computing accuracy. More amazing is that on some computers, equation (9.4) was often satisfied down to and including the *the least significant bit*. I don't doubt that larger rounding errors could occur, but so far, every time I encountered a relative discrepancy of 10^{-5} or more, I was later able to uncover a conceptual error or a programming error.

Don't be alarmed if the operator you have defined has truncation errors. Such errors in the definition of the original operator should be identically matched by truncation errors in the conjugate. If your code passes the dot product test, then you should be able to pass back and forth between data space and model space as many times as you like without fear of growth of truncation errors.

9.2 NORMAL MOVEOUT AND OTHER MAPPINGS

Many times we simply deform or stretch a wavefield or a map. Two examples we'll examine in detail are (1) resampling data along the space axis, and (2) normal moveout correction.

9.2.1 Interpolation, nearest neighbor

Let k range along a survey line and let data values x_k be packed into a vector $\mathbf{x}$. (Each data point x_k could also be a seismogram.) We plan to resample the data more densely, say from 4 to 6 points with a crude nearest neighbor scheme given by

$$\mathbf{y} \quad = \quad \mathbf{Bx} \tag{9.5}$$

where

$$\mathbf{B} \quad = \quad \begin{bmatrix} 1&0&0&0 \\ 0&1&0&0 \\ 0&1&0&0 \\ 0&0&1&0 \\ 0&0&0&1 \\ 0&0&0&1 \end{bmatrix} \tag{9.6}$$

In this highly simplified example, the interpolated data is obviously $\mathbf{y} = (x_1, x_2, x_2, x_3, x_4, x_4)$, but we carry though this easily comprehensible interpolation task to shed light on the less comprehensible matrix algebra. Multiply the resampled data $\mathbf{y}$ by the transposed resampling operator $\mathbf{B'}$ and see what we get.

$$\mathbf{\tilde{x}} \quad = \quad \mathbf{B'y} \quad = \quad \mathbf{B'Bx} \tag{9.7}$$

We see that the result $\mathbf{\tilde{x}}$ has the same number of components as the original data $\mathbf{x}$, but it is not exactly equal to the original data. The matrix $\mathbf{B'B}$ is the diagonal matrix

$$\mathbf{B'B} \quad = \quad \begin{bmatrix} 1&0&0&0 \\ 0&2&0&0 \\ 0&0&1&0 \\ 0&0&0&2 \end{bmatrix} \tag{9.8}$$

So, $\tilde{x}_1 = x_1$, but $\tilde{x}_2 = 2x_2$. To recover the original data we need to divide $\mathbf{\tilde{x}}$ by this diagonal matrix.

$$\mathbf{x} \quad = \quad \mathbf{(B'B)^{-1}\tilde{x}} \tag{9.9}$$

$$\mathbf{x} \quad = \quad \mathbf{(B'B)^{-1}B'y} \tag{9.10}$$

Curiously, (9.10) looks like a familiar equation from least squares theory [equation (8.13)]. Indeed we now understand a theoretical basis for why a conjugate operator is often an approximate inverse. It happens in proportion to the diagonal dominance of $\mathbf{B'B}$.

Recovering $\mathbf{x}$ from $\mathbf{y}$ with equation (9.10) presumes the existence of the inverse of $\mathbf{B'B}$. As you might expect, this matrix is nonsingular when $\mathbf{B}$ *stretches* the data, because then a few data values are distributed among a greater number of locations. Where the transformation *squeezes* the data it must become singular, since it must be impossible to return uniquely to the uncompressed condition. Chapter 8 should explain how to handle singular matrices, enabling you to decompress losing only the high frequencies. WOULD BE NICE TO HAVE AN EXAMPLE.

9.2.2 Nearest neighbor NMO

Normal moveout correction (NMO) is a geometrical correction of reflection data that stretches the time axis so that data recorded at nonzero separation x_0 of shot and receiver, after stretching appears to be at zero offset. Crudely speaking, it is like time to

depth conversion with the equation $v^2t^2 = x^2 + x_0^2$. If the data at x_0 is stretched from t to x, it should look like data at any other x (assuming plane horizontal reflectors, etc.). In practice x is not used, rather *travel time depth* τ is used, where $\tau = vx$ so $\tau^2 = t^2 + x_0^2/v^2$.

Normal moveout is a linear operation. This means that data can be decomposed into any two parts, early and late, high frequency and low, etc, and the two parts can be NMO'ed either separately or together and the result is the same, i.e. $\mathbf{N(a+b) = Na + Nb}$.

The NMO transformation $\mathbf{N}$ is representable as a square matrix. The matrix $\mathbf{N}$ is a (x,t)-plane containing all zeros except an interpolation operator centered along the hyperbola. The dots in the matrix below are zeros. The vector $\mathbf{x}$ represents a seismogram before NMO. The output vector $\mathbf{y}$, i.e. the NMOed data is simply $(x_6, x_5, x_5, x_6, x_6, x_7, x_7, x_8, x_9, x_{10})$. In real life, the subscript would go up to about a thousand instead of merely to ten.

$$\mathbf{y} = \mathbf{Nx} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \end{bmatrix} = \begin{bmatrix} . & . & . & . & 1 & . & . & . & . & . \\ . & . & . & . & 1 & . & . & . & . & . \\ . & . & . & . & 1 & . & . & . & . & . \\ . & . & . & . & . & 1 & . & . & . & . \\ . & . & . & . & . & 1 & . & . & . & . \\ . & . & . & . & . & . & 1 & . & . & . \\ . & . & . & . & . & . & 1 & . & . & . \\ . & . & . & . & . & . & . & 1 & . & . \\ . & . & . & . & . & . & . & . & 1 & . \\ . & . & . & . & . & . & . & . & . & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \end{bmatrix} \tag{9.11}$$

You can think of the matrix as having a horizontal t-axis and a vertical τ-axis. The transpose matrix defining an $\tilde{\mathbf{x}}$ from $\mathbf{y}$ gives pseudodata $\tilde{\mathbf{x}}$ from the zero offset (or stack) model $\mathbf{y}$, namely

$$\tilde{\mathbf{x}} = \mathbf{N'y} = \begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \\ \tilde{x}_3 \\ \tilde{x}_4 \\ \tilde{x}_5 \\ \tilde{x}_6 \\ \tilde{x}_7 \\ \tilde{x}_8 \\ \tilde{x}_9 \\ \tilde{x}_{10} \end{bmatrix} = \begin{bmatrix} . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . \\ 1 & 1 & 1 & . & . & . & . & . & . & . \\ . & . & . & 1 & 1 & . & . & . & . & . \\ . & . & . & . & . & 1 & 1 & . & . & . \\ . & . & . & . & . & . & . & 1 & . & . \\ . & . & . & . & . & . & . & . & 1 & . \\ . & . & . & . & . & . & . & . & . & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \end{bmatrix} \tag{9.12}$$

Next we look at the matrix $\mathbf{N'N}$.

$$\mathbf{N'N} = \begin{bmatrix} . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . \\ . & . & . & 3 & . & . & . & . & . & . \\ . & . & . & . & 2 & . & . & . & . & . \\ . & . & . & . & . & 2 & . & . & . & . \\ . & . & . & . & . & . & 1 & . & . & . \\ . & . & . & . & . & . & . & 1 & . & . \\ . & . & . & . & . & . & . & . & 1 & . \\ . & . & . & . & . & . & . & . & . & 1 \end{bmatrix} \tag{9.13}$$

Any mathematician will tell you that equation (9.13) is not invertible because the zeros on the diagonal make it singular. But as a geophysicist, you know better. Our inverse is

$$(\mathbf{N'N})^{-1} = \begin{bmatrix} . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . \\ . & . & . & \frac{1}{3} & . & . & . & . & . & . \\ . & . & . & . & \frac{1}{2} & . & . & . & . & . \\ . & . & . & . & . & \frac{1}{2} & . & . & . & . \\ . & . & . & . & . & . & 1 & . & . & . \\ . & . & . & . & . & . & . & 1 & . & . \\ . & . & . & . & . & . & . & . & 1 & . \\ . & . & . & . & . & . & . & . & . & 1 \end{bmatrix} \tag{9.14}$$

The implementation of the program is straightforward. Because of the limited alphabet of programming languages, I used the variable z to designate τ.

```
# NMO, transpose NMO, and inverse NMO by nearest neighbor.
# transp=0:     zz(iz) = [NMO]          tt(it)
# transp=1:     tt(it) = [NMO transpose] zz(iz)
subroutine nmoiix( inverse, transp, slow, x, t0, dt, nt, tt, zz )
real slow(nt),  t0, dt, tt(nt), zz(nt), count(4096)
integer  inverse, transp, it, nt, iz
real t, x, z, xs, arg
if( transp == 0 )
        do iz = 1,nt
                zz(iz) = 0.0
else
        do it = 1,nt {
                tt   (it) = 0.0
                count(it) = 0.0
                }
z = t0 + nt * dt
t = z
do iz = nt, 1, -1  {
        xs = x * slow(iz)
        arg = z * z + xs * xs
        # next line replaceable by:     t = sqrt ( arg )
        t = (arg + t * t) / (t + t)
        it = 0.5 + (t - t0) / dt        # Define nearest neighbor.
        if ( it <= nt  ) {
                if( transp == 0 )
                        zz(iz) = zz(iz) + tt(it)
                else {
                        tt(it) = tt(it) + zz(iz)
                        count(it) = count(it) + 1.
                        }
```

```
                }
        z = z - dt
        }
if( inverse > 0 )
        do it = 1, nt
                if( count(it) != 0.0 )
                        tt(it) = tt(it) / count(it)
return; end
```

(Parenthetically, you might notice the line where the square root is computed more rapidly by Newton's iteration, i.e. for any starting value of t, iterating with $t \leftarrow .5(\alpha + t^2)/t$ converges to $t = \sqrt{\alpha}$. It is not so easy to prove that the iteration converges, but it is easy to see what it converges to, if it converges. Just replace the "$\leftarrow$" by an "$=$" and solve for t.)

9.2.3 NMO with linear interpolation

Linear interpolation implies that the matrix **N** is a two-band matrix. Each row has exactly two elements that interpolate between two elements on the input. I'll sketch the appearance of the matrix, using the letters a and b for the elements. Each a and b is different numerically, but on a given row, $a + b = 1$.

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \end{bmatrix} = \begin{bmatrix} . & . & . & . & a & b & . & . & . & . \\ . & . & . & . & a & b & . & . & . & . \\ . & . & . & . & a & b & . & . & . & . \\ . & . & . & . & . & a & b & . & . & . \\ . & . & . & . & . & a & b & . & . & . \\ . & . & . & . & . & . & a & b & . & . \\ . & . & . & . & . & . & a & b & . & . \\ . & . & . & . & . & . & . & a & b & . \\ . & . & . & . & . & . & . & . & a & b \\ . & . & . & . & . & . & . & . & . & a \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \end{bmatrix} \tag{9.15}$$

In this case the matrix $\mathbf{B'B}$ is tridiagonal, but I am going to let you work out the details by yourself. The original data can be recovered by solving the tridiagonal system. This idea can be used to program an invertible NMO or trace interpolation.

Basing a program on the above principles, a field profile was NMOed and then inverse NMOed. The result was then plotted upon the original profile in Figure 9.1. The processed profile is not distinguishable from the original except near the direct arrival. The direct arrival was evidently moved out to before $t = 0$ so it was not recoverable.

Program for NMO and its inverse including linear interpolation

Some day this book will be sold on a laser disk. Then there will be room to include all the programs, test examples, and data. Meanwhile, you need to refer to SEP-42 p. 119-120 for the program. That program constructs tables of interpolation coefficients. The construction amounts to about 80% of the computational effort. So when many traces of the same offset will be (inverse)-(transpose)-NMOed these tables my be reused, thereby saving a factor of five.

FIG. 9.1. (conj-nmo) Field profile **d** plotted on top of $\mathbf{N^{-1}NMO\,d}$. The two overlay except near the first arrival. I SHOULD BE ABLE TO RECONSTRUCT THIS FROM THE CANON LIBRARY.

9.3 UNITARY AND PSEUDOUNITARY TRANSFORMATION

A so-called *unitary* transformation **U** conserves energy. In other words, if $\mathbf{y} = \mathbf{Ux}$ then $\mathbf{x'x} = \mathbf{y'y}$, which requires $\mathbf{U'U} = \mathbf{I}$. Imagine an application where the transformation seems it shouldn't destroy information. Can we arrange it to conserve energy? Starting from our transform pair

$$\mathbf{y} = \mathbf{Bx} \tag{9.16}$$

$$\hat{\mathbf{x}} = \mathbf{(B'B)^{-1}B'y} \tag{9.17}$$

define

$$\mathbf{U} = \mathbf{B(B'B)^{-1/2}} \tag{9.18}$$

For examples like nearest neighbor NMO, the new factor $\mathbf{(B'B)^{-1/2}}$ is a prescaling of the data **x**. The proposed transformations between data space and model space are:

$$\mathbf{y} = \mathbf{Ux} \tag{9.19}$$

$$\hat{\mathbf{x}} = \mathbf{U'y} \tag{9.20}$$

Direct substitution shows that $\hat{\mathbf{x}} = \mathbf{x}$.

Is the operator **U** unitary? It wouldn't be for NMO because (9.13) is not invertible. But in practice it is almost as good. So we call it *pseudounitary*. Certain values are lost such as (x_1, x_2, x_3, and x_4) in (9.11), but it is nice to be able to recover all the rest. So although $\hat{\mathbf{x}} \neq \mathbf{U'Ux}$ we do find by substitution that $\hat{\mathbf{x}} = \mathbf{U'U}\hat{\mathbf{x}}$. So, everything that is going to be lost, is lost on the *first* time through the transformation.

Mathematically such a pseudounitary operator is called an *idempotent* operator, and the mathematical condition for idempotence is that $\mathbf{(U'U)^2 = U'U}$ but $\mathbf{U'U \neq I}$.

A trip into and back from the space of a pseudounitary operator is like a pass through a bandpass filter. When done a second time, nothing changes.

9.3.1 Pseudounitary NMO with linear interpolation

It is often desirable to work with transformations that are as near as possible to being unitary, i.e. their transpose is their pseudoinverse. Such transformations will be called *pseudounitary*. The useful feature is that repetitive transformation does yield rapid degradation.

Let us make NMO with linear interpolation into a pseudounitary transformation. We need to factor the tridiagonal matrix $\mathbf{N'N} = \mathbf{T}$ into bidiagonal parts, $\mathbf{T} = \mathbf{B'B}$. One such factorization is the well-known Cholesky decomposition. It is reminiscent of spectral factorization. Then we'll define *pseudounitary* NMO as $\mathbf{U} = \mathbf{NB^{-1}}$. To confirm the unitary property we check that $\mathbf{U'U} = \mathbf{B'^{-1}N'NB^{-1}} = \mathbf{B'^{-1}B'BB^{-1}} = \mathbf{I}$. Time series experts will recall that an all-pass filter is a ratio of two terms, both with the same color, the denominator minimum phase and the numerator not. Analogously in $\mathbf{U} = \mathbf{NB^{-1}}$, the numerator time shifts, and the denominator corrects the numerator's color.

9.4 TOMOGRAPHY

Tomography is the reconstruction of a function from line integrals through it. Tomography has become a routine part of medicine, and an experimental part of earth sciences. A simple arrangement for illustration is well-to-well tomography. A sound source can be placed at any depth in one well, and receivers placed at any depth in another well. At the sender well, we have sender depths s, and at the receiver well, we have receiver depths g. Our data is a table $t(s,g)$ of travel times from s to g. The idea is to try to map the area between the wells. We divide the area between wells into cells in (x,z)-space. The map could be one of material velocities or absorptivity. The travel time of a ray increases by adding the *slownesses* of cells traversed by the ray. Our model is a table $s(x,z)$ of slownesses in the plane between wells. (Alternately, the logarithm of the amplitude of the ray is a summation of absorptivities of the cells traversed.) The pseudocode follows

```
do s = range of sender locations
do g = range of receiver locations
        z = z(s)                    # depth of sender.
        θ = θ(s,g)                  # ray take-off angle.
        do x = range from senders to receivers.
                z = z + dx tan θ    #ray tracing
                if modeling
                        t_sg  =  t_sg + s_zz
                else tomography
                        s_zz  =  s_zz + t_sg
```

Above we assumed the rays were straight lines. The problem remains one of linear operators even if the rays curve making the ray tracing more complicated. The problem becomes nonlinear requiring the complexities of nonlinear optimization theory if the solution $s(x,z)$ is used to modify the ray tracing.

It is well known that the speed of convergence of the conjugate-gradient method depends on the preconditioning of the matrix. I should say something about such issues as trying to make the transformation pseudounitary. It isn't as easy as with deformations, as I have learned with velocity transformations.

If you want to ray trace more precisely through blocks of well defined shape, then you may wish to study the way NMO was handled earlier by linear interpolation.

9.5 MIGRATION

When reflectors in the earth are dipping, or broken into point scatterers, time-to-depth conversion is not simply a stretching of the time axis. Modeling is done in a variety of ways, one of which is to model each point in the depth (x,z)-plane by an hyperbola in the data (x,t)-plane. The conjugate operation consumes much computer power in the petroleum prospecting industry and is called *migration*. There are a large variety of methods, many of which are the subject of my second book "Imaging the Earth's Interior" . But that book does not describe the conjugacy properties which are described below.

9.5.1 Kirchhoff modeling and migration

The linear interpolation within a deformation transformation is a weighted sum over two points. In Kirchhoff migration and diffraction, the summation is over more than two points.

The transpose to summation is taking a point and spreading it around, actually, adding it atop of what is already out there. I'll refer to the transpose of summation as "spraying".

Ignoring velocity and "if index off data" tests, Kirchhoff modeling and migration is:

```
do is = 1,ns
        do ih = -25, 25
                it = sqrt( is*is + ih*ih )
                do iy = 1,ny
                        ig = iy + ih
                        if not transpose
                                zz(is,iy) = zz(is,iy) + tt(it,ig) # summing
                        if transpose
                                tt(it,ig) = tt(it,ig) + zz(is,iy) # spraying
```

The trick to making the Kirchhoff fast is to move the y-loop to the inside of the square root and interpolation overheads.

9.5.2 Stolt migration and diffraction

NMO is based on the quadratic equation $v^2t^2 = z^2 + x^2$ (as explained in IEI). Stolt's migration is based on the quadratic equation $\omega^2/v^2 = k_z^2 + k_x^2$. Stolt migration is NMO

in the Fourier domain. Denote the Fourier transform operator by **F** and the Stolt operator by **S** where

$$\mathbf{S} = \mathbf{F}'\mathbf{N}\mathbf{F} \tag{9.21}$$

A property of matrix conjugates is $(\mathbf{A}\,\mathbf{B}\,\mathbf{C})' = \mathbf{C}'\mathbf{B}'\mathbf{A}'$. We know the transpose of NMO, and we know that the (conjugate) transpose of Fourier transformation is inverse Fourier transformation. So

$$\mathbf{S}' = \mathbf{F}'\mathbf{N}'\mathbf{F} \tag{9.22}$$

So we see the transpose to Stolt modeling is Stolt migration. (There are a few more details with Stolt's Jacobian).

9.5.3 Gazdag migration and diffraction

The Gazdag algorithm is the most complicated one considered so far. Below you need to check two things, is it really the Gazdag modeling and migration program? Are the operations really the conjugates?

A property of matrix conjugates is $(\mathbf{A}\,\mathbf{B}\,\mathbf{C})' = \mathbf{C}'\mathbf{B}'\mathbf{A}'$. Interpreted in terms of layers, this means if the operator goes from the earth's surface to its interior, then the conjugate operator goes from the interior to the surface. Observe in the program sketch below that the loops run in opposite directions, the inverse Fourier transform is its conjugate, and the order of `FT`, `layer`, and `source` are reversed in the conjugate.

```
if not conjugate i.e. migration
        erase all of Image (z, k_x)
        U(ω, k_x)   =   FT2D[u(t, x)]
        for z = 0, z < z_max, z = z + Δz
                call source
                call layer
                image(z, x)   =   FT^{-1}[Image(z, k_x)]
if conjugate i.e. diffraction
        erase all of U(ω, k_x)
        for z = z_max, z > 0, z = z − Δz
                Image(z, k_x)   =   FT[ image(z, x)]
                call layer
                call source
        u(t, x)   =   IFT2D[U(ω, k_x)]
return; end
```

The `layer` subroutine resembles a diagonal matrix multiply.

```
subroutine layer
for all ω and all k_x
        C   =   exp(−Δz √((−iω)²/v² + k_x²))
        if not conjugate
                U(ω, k_x)   =   U(ω, k_x) × C
        if conjugate
                U(ω, k_x)   =   U(ω, k_x) × C̄
return; end
```


The *source* subroutine next displays a typical conjugate character. In modeling, the image, being a point in time, is sprayed out into all frequencies. In migration, the image point is created by summing all frequencies.

```
subroutine source
for all ω and all k_x
        if not conjugate
                Image(z, k_x)   =   Image(z, k_x) + U(ω, k_x)
        if conjugate
                U(ω, k_x)   =   U(ω, k_x) + Image(z, k_x)
return; end
```

The above program sketch looks correct, but no program for it has yet gone through the $(yA)x = y(Ax)$ test. (CHECK THIS WITH JOS OR IN CANON LIBRARY).

9.6 GENERAL CASE OF MAPPING OR DEFORMATION

Now let us look at the general case of deformation and mapping with nearest-neighbor interpolation. Commonly, you know a coordinate transformation, but you do not always know its inverse directly. Luckily, that is enough to transform data both ways as you can see from the code, which assumes that given values in the output space it can compute values in the input space, but not vice versa. (If your application runs the other way, then you interchange your definition of operator and transpose.)

I need to generalise this code a bit more. More examples found in the canon library. Maybe pseudo unitary? Maybe omit this whole section? or condense to a few comments following the NMO program?

```
if not transpose
        then erase y
if transpose
        then erase x
Loops over output space i-indices {
        Find input j-indices as functions of output i-indices.
        if j-index on data {
                value = any function of indices
                if not transpose
                        y(i1, i2...) = y(i1, i2...) + value × x(j1, j2...)
                if transpose
                        x(j1, j2...) = x(j1, j2...) + value × y(i1, i2...)
                        sum(j1, j2...) = sum(j1, j2) + value
                } } }
```

```
if want pseudo inverse
        loop over j-space
                if sum(j1, j2, ...) ≠ 0.0
                        x(j1, j2, ...) = x(j1, j2, ...)/sum(j1, j2, ...)
```

LOTS OF REFERENCES COMMENTED OUT HERE