CENTER FOR COMPUTER RESEARCH IN MUSIC AND ACOUSTICS OCTOBER

Department of Music
Report No. STAN-M-33

SPECTRUM ANALYSIS TUTORIAL

by

David A. Jaffe

This manuscript is an outgrowth of a course in signal processing given by Julius O. Smith at Stanfo University beginning in the fall of 1984. It provides an elementary introduction to spectrum analysis.

This research was supported (in part) by the System Development Foundation under Grant SDF #345. The views a conclusions contained in this document are those of the author and should not be interpreted as necessa representing the official policies, either expressed or implied, of Stanford University or of the sponsoring foundatio

David A. Jaffe
Center for Computer Research in
Music and Acoustics
Stanford University
Stanford, California 94305 USA

Spectrum Analysis Tutorial, Part 1: The Discrete Fourier Transform

Introduction

This tutorial is an outgrowth of a course in signal processing given by Julius O. Smith at Stanford University in the fall of 1984 (see Smith 1981, as well). It provides an elementary mathematical introduction to spectrum analysis. This is the first of two parts. In part one, the discrete Fourier transform is introduced and analyzed in depth. In part two, some fundamental spectrum analysis theorems and applications are discussed. The only mathematical background assumed is high school trigonometry, algebra, and geometry. No calculus is required. Familiarity with summation formulae, complex numbers, and vectors is helpful, although not essential.

Overview

Since the days of the mathematician Jean Baptiste Joseph Fourier (1768–1830), it has been recognized that any sound can be broken down into a set of sinusoidal functions in much the way that any colored light can be broken down into basic colors of the visual spectrum. Furthermore, if the sound is strictly periodic and has no energy above a certain frequency (i.e., it is *band-limited*), the set of sinusoids is finite. The *discrete Fourier transform* (DFT) is a mathematical function that performs the operation of breaking down a digitally represented signal, such as a digitally recorded sound, into its *spectrum*—a set of scalars on a set of sinusoidal components. Most readers are probably more familiar with the *fast Fourier transform* (FFT). The FFT is simply an efficient implementation of the DFT. The FFT runs in time proportional to $N \log N$ rather than in time proportional to N^2, where N is th number of input samples. (For more informati FFTs, see Rabiner and Gold 1975 or Aho, Hop and Ullman 1974.)

More precisely, the DFT takes a waveform (a tally sampled sound) as input and produces a s coefficients that can be used to scale a set of s waves equally spaced between 0 Hz and the sa pling rate f_s Hz. (The frequencies between $f_s/2$ and f_s Hz are equivalent to the frequencies bet $-f_s/2$ Hz and 0 Hz. This phenomenon is expla in part two of this tutorial.) If the scaled funct are added together, the original waveform is re structed. This two-stage process of spectrum a sis and waveform reconstruction is called *ana synthesis* in computer music terminology. Th of coefficients is called the *frequency domain* resentation of the waveform while the wavefo self is called the *time domain* representation. DFT and its inverse, the IDFT, are fundament operations that convert between these two do The nineteenth-century acoustician Helmhol ognized that the spectrum of a sound is strong correlated to what we perceive as the "timbre" the sound. Thus the frequency domain repres tion offers meaningful information to a music

The average musician thinks of the DFT as black box that displays the amplitude of the h monics of a sound. The musician can use it t amine the frequency content of a sound in or create a corresponding synthetic model. There however, a number of reasons why it is of valu a musician to open up this black box and see works. First, one can understand the subtletie sampling-rate conversion and other operation sound. Secondly, one can intelligently manipu compositional tools such as the *phase vocode* (Dolson 1983). Finally, a vast body of technica erature becomes accessible to the musician.

Computer Music Journal, Vol. 11, No. 2, Summer 1987,

The DFT is a single mathematical formula. Yet, packed into this one formula is some powerful and profound mathematics that merits a good deal of explanation. We shall examine the DFT in terms of a mathematical *vector space*. This requires that we provide some background in linear algebra and complex variables. A list of identities illustrating complex variable arithmetic is provided in Appendix B.

We first discuss the input and output of the DFT, looking at the DFT itself as a black box. Next the DFT equation is defined. The remainder (and bulk) of part one is concerned with explaining the DFT equation.

The Input and Output of the DFT

The DFT is a function that takes a waveform as input and produces as output the set of coefficients that determine the sinusoids present in the sound. Each of the output coefficients is a complex number whose *magnitude* specifies the amplitude of a particular sinusoidal component and whose *angle* specifies the phase of that component. (We explain more about complex numbers shortly.) The output of the DFT is called the spectrum of the waveform. In common usage, the term "spectrum" is often used to refer to the magnitude of the DFT coefficients. However, we define "spectrum" as the complex coefficients themselves. The inverse DFT reverses the effect of the DFT. It takes as input a spectrum and produces as output a (time domain) waveform.

Fourier's theorem implies that if a waveform is periodic and band-limited, it can be represented by a finite number of sinusoids. The DFT assumes that its input is one period of such a waveform. With this assumption, a finite number of sinusoids can be used to represent the waveform. (Further implications of this assumption are discussed in part two of this tutorial.) The frequencies of the sinusoids used in the DFT are equally spaced between 0 Hz and the f_s (or between $-f_s/2$ and $f_s/2$). The number of sinusoids in the set is the same as the number of samples in the waveform. Note that the input "waveform" can, in fact, be several periods of some other waveform. For example, if the waveform is of length N and it consists of four copies of a waveform of length $N/4$, the resulting spectrum will contain significant energy at only one out of every four frequencies.

In mathematical terms, both the input and output of the DFT are *sequences*. A sequence is a list of numbers indexed by an integer variable. For example $y = \{1, .5, .1, -.1\}$ is a sequence of length $N = 4$, with $y(0) = 1$, $y(1) = .5$, and so forth. A sequence can also be represented with a functional definition. For example, the functional definition $y(n) = \cos(\omega n)$, where ω is the radian frequency, generates a cosine sequence. ($\omega = 2\pi f/f_s$, where f is the frequency in Hz and f_s is the sampling rate in Hz.)

In sound processing, the input to the DFT is a sequence of samples of a digitized pressure function of time. By convention, we use the index n as the sample number where $n = 0, 1, 2, \ldots, (N - 1)$, and we use a lower case letter as the sequence name (e.g., y). Time can be converted from sample numbers to seconds by defining a function of time in seconds $y(t_n) = y(nT)$, where T is the sampling interval (that is, the reciprocal of the sample rate) in seconds. Note, however, that this is still a discrete-time function defined only at the points nT, $n = 0, 1, 2, \ldots, (N - 1)$.

The output of the DFT, the spectrum, is also a sequence. This means that frequency is quantized just as time is quantized. We follow the convention that k is the index variable of the output sequence, where $k = 0, 1, 2, \ldots, (N - 1)$. We use the uppercase version of the input sequence name for the output sequence name (e.g., Y). $Y(k)$ thus represents the coefficient of the kth sinusoidal component of the spectrum of the waveform y. The physical units of frequency (Hz) can be shown explicitly by defining a function of a continuous variable that has been sampled $Y(f_k) = Y(k/NT)$). This is, once again, a function of discrete values.

Definition of the DFT

We begin by simply stating the equation for the DFT and giving a brief explanation of its components. The DFT equation is:

$$\mathrm{DFT}_k(y) \triangleq Y(k) \triangleq \sum_{n=0}^{N-1} y(n)e^{-j\omega_k nT},$$

$$k = 0, 1, \ldots, N-1.$$

A number of possibly unfamiliar symbols appear in this formula. The subscript k is used in the notation $\mathrm{DFT}_k(y)$, to show that the output of the DFT is a spectrum indexed by k. The symbol $\triangleq$ means "is defined as." $\Sigma_{n=0}^{N-1} x(n)$ means "the sum of all values of $x(n)$ for n between 0 and $N-1$ inclusive." In our case, $x(n)$ is the multiplication of the input waveform $y(n)$ by a complex function $e^{-j\omega_k nT}$. n is the waveform sample index, e is the famous irrational number that serves as the base of the natural logarithm (Moore 1978a, 1978b). Its appearance in the DFT formula stems from *Euler's identity*, which expresses sines and cosines in terms of the exponential function, as we show. j is the square root of negative one (-1) and is further explained in the section on complex numbers. ω_k is the radian frequency of the kth sinusoidal component. It can be expanded to $\omega_k = 2\pi k f_s/N$, where f_s is the sampling rate in samples per second. $T = 1/f_s$ is the sampling period in seconds per sample.

The variable k is held fixed for each evaluation of the summation. That is, the summation is first evaluated for k equal to 0. This produces the Fourier coefficient of the first sinusoidal component. Then k is reset to 1, and the summation is reevaluated to give the second Fourier coefficient. The process is repeated until all N Fourier coefficients have been computed. Notice that as k goes from 0 to $N-1$, $\omega_k T$ goes from 0 to 2π in equal jumps. This turns out to be equivalent to measuring energy in N frequency bins equally spaced from 0 Hz to the sampling rate.

The inverse DFT, or IDFT, is defined to be:

$$\mathrm{IDFT}_n(Y) \triangleq y(n) \triangleq \frac{1}{N}\sum_{k=0}^{N-1} Y(k)e^{j\omega_k nT}.$$

Note that the IDFT and DFT equations are quite similar. Since the IDFT is the inverse of the DFT, taking the IDFT of the DFT of a waveform is an identity operation. That is, $\mathrm{IDFT}_n(\mathrm{DFT}(y)) = y(n)$. Similarly, $\mathrm{DFT}_k(\mathrm{IDFT}(Y)) = Y(k)$.

Having defined the DFT, we now explain how it works. The approach we take is to view the D as a simple *change of coordinate system*. This quires some concepts from linear algebra, incl the concepts of *vector, basis, vector projectio orthogonality*. First, however, we need to unde stand something about complex numbers and they are used in the DFT.

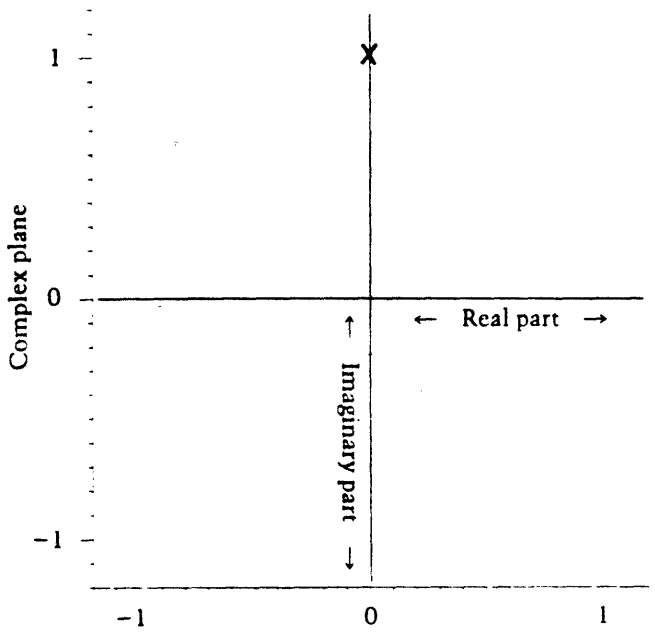
Why Complex Numbers Appear in the DFT Equation

The input to the DFT is a sequence of sample waveform. These samples are real numbers wh taken from the physical world, but they can be plex numbers in theoretical cases. Of more rel vance to the musician is the fact that the Fou coefficients generated by the DFT are generall complex. Complex numbers are used in the D for two reasons. First, Euler's identity (given i later), allows a sinusoid to be expressed as a c plex exponential. Since trigonometric calcula are often much more cumbersome than expon calculations, it is preferable to work with exp tials. (See Smith 1981 for a comparison of trig metric and exponential methods.) Second, it i essary that both a phase term and an amplitu term be associated with each spectral compor Complex numbers allow both to be packaged single algebraic quantity.

Complex Numbers

The set of real numbers is converted to the se complex numbers by the addition of a single number, j. (Appendix A gives a brief discussic the original motivation for the invention of c plex numbers.) We define j as $j \triangleq \sqrt{-1}$. (Som use i rather than j as $\sqrt{-1}$.) Then $j^2 = -1$, j^3 $j^4 = 1$, etc. For any negative number $x < 0$ an tive number $R = -x$, $\sqrt{x} = \sqrt{-R} = j\sqrt{R}$. A *plex number* z is defined as a sum of a *real p* an *imaginary part*, either of which can be 0. z can be represented as $x + jy$ where x and y real. We also sometimes use the notations Re ("real part of z equals x") and $\mathrm{Im}\{z\} = y$ ("ima

Fig. 1. Representation of the point (0, 1) = j on the complex plane.



part of z equals y"). The real numbers are the subset of the complex numbers for which $y = 0$. Appendix B is a list of identities that includes the rules for complex arithmetic.

We can plot complex numbers in a plane (called the *complex plane*) as ordered pairs (x, y). For example, the number j has coordinates (0, 1) as shown in Fig. 1. We can also express complex numbers in terms of polar coordinates as an ordered pair (R, θ) where $R = |z|$ is the magnitude of z and $\theta = \angle z$ is the angle of z. Using simple trigonometry, we can convert from rectangular coordinates (x, y) to polar coordinates (R, θ) and vice versa:

$$x = R\cos(\theta)$$

$$y = R\sin(\theta)$$

$$R = \sqrt{x^2 + y^2}$$

$$\theta = \tan^{-1}(y/x).$$

Note that $z = x + jy$ is an algebraic representation of z in terms of its rectangular coordinates. Similarly, there is an algebraic representation of z in terms of polar coordinates:

$$z = R(\cos(\theta) + j\sin(\theta)).$$

Thus any complex number can be broken down into a cosinusoidal and a sinusoidal component.

Another representation of z exists in terms of polar coordinates. In order to define it, we must introduce Euler's identity:

$$e^{j\theta} = \cos(\theta) + j\sin(\theta). \tag{1}$$

(A proof of Euler's identity is not given here, although the perplexed reader may be consoled to note that for $\theta = 0$, $e^{j0} = (\cos(0) + j\sin(0)) = 1 + j0 = 1$, as one would expect.) With Euler's identity, we gain the alternative algebraic representation of z in terms of polar coordinates:

$$z = Re^{j\theta} = R(\cos(\theta) + j\sin(\theta)).$$

This representation simplifies the mathematics of the DFT greatly, since simple rules of exponents can now be used in place of difficult trigonometric identities.

The *complex conjugate* of z is notated $\bar{z} = \overline{x + jy} \triangleq x - jy$. In polar coordinates, $\overline{Re^{j\theta}} \triangleq Re^{-j\theta}$. It is calculated by simply replacing j with $-j$. Note that $z + \bar{z} = 2\mathrm{Re}\{z\}$. Similarly, $z - \bar{z} = 2\mathrm{Im}\{z\}$. This fact, with Euler's identity, can be used to derive formulas for sine and cosine in terms of $e^{j\theta}$:

$$\begin{aligned} e^{j\theta} + \overline{e^{j\theta}} &= e^{j\theta} + e^{-j\theta} \\ &= (\cos(\theta) + j\sin(\theta)) + (\cos(\theta) - j\sin(\theta)) \\ &= 2\cos(\theta). \end{aligned}$$

Similarly, $e^{j\theta} - \overline{e^{j\theta}} = 2j\sin(\theta)$.

A complex number multiplied by its conjugate is equal to its magnitude squared:

$$z\bar{z} = (x + jy)(x - jy) = x^2 - (jy)^2 = x^2 + y^2 = |z|^2.$$

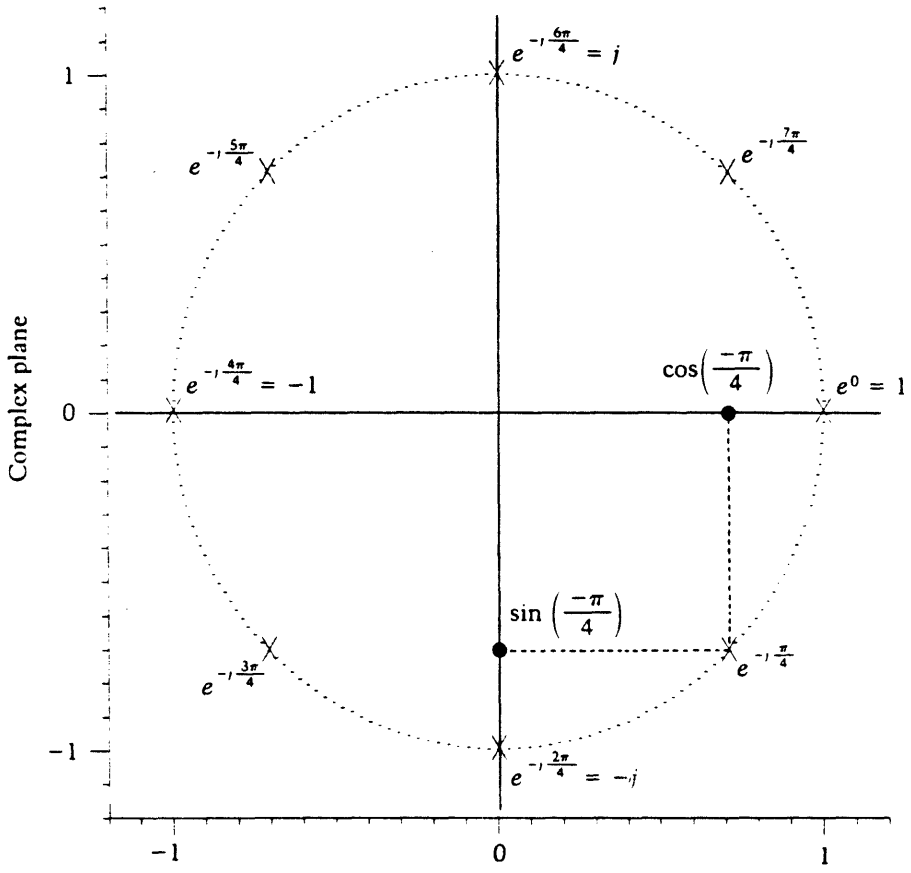
Or, in polar coordinates,

$$z\bar{z} = Re^{j\theta}Re^{-j\theta} = R^2e^0 = R^2 = |z|^2.$$

Discrete-Time Complex Sinusoids

The DFT *projects* a waveform onto a set of *discrete-time complex sinusoids*. Projection is explained later. In this section we define a complex sinusoid and generate it from the exponential function. We show that raising a complex number of unit magnitude $e^{j\theta}$ to successive powers generates succes-

Fig. 2. A sinusoid with $\omega T = \pi/4$ on the complex plane.



sive points along the circle of unit magnitude $y[n] = e^{j\theta n}$ (Fig. 2). A sequence of the form $e^{j\theta n}$ is called a discrete-time complex sinusoid of unit amplitude.

A discrete-time complex sinusoid (hereafter called simply a sinusoid) can be defined as the sequence:

$$y[n] \triangleq e^{j\omega t_n} = e^{j\omega nT} = \cos(\omega nT) + j\sin(\omega nT).$$

The real component of this sequence is a sampled cosine with unit amplitude and frequency ωT radians per sample. The imaginary component is a sampled sine with unit amplitude and frequency ωT radians per sample. The complex magnitude is interpreted as amplitude (in this case equal to 1). Figure 2 shows an example of a sinusoid with $\omega T = \pi/4$ radians per sample. A sinusoid $e^{j\omega nT}$ is periodic with a period of $P \triangleq 2\pi/\omega$ seconds. That is, we can add P to the initial phase without changing the trajectory:

$$e^{j\omega(nT + P)} = e^{j\omega nT}e^{j\omega P} = e^{j\omega nT}e^{j(2\pi/P)P} = e^{j\omega nT}e^{j2\pi} = e^{j\omega nT}.$$

Similarly, for all integers k, $e^{j\omega(nT + kP)} = e^{j\omega nT}$. Thu $2\pi/\omega$ is the period of the sinusoid.

A sinusoid can be scaled by a real amplitude A and can have a real phase offset ϕ:

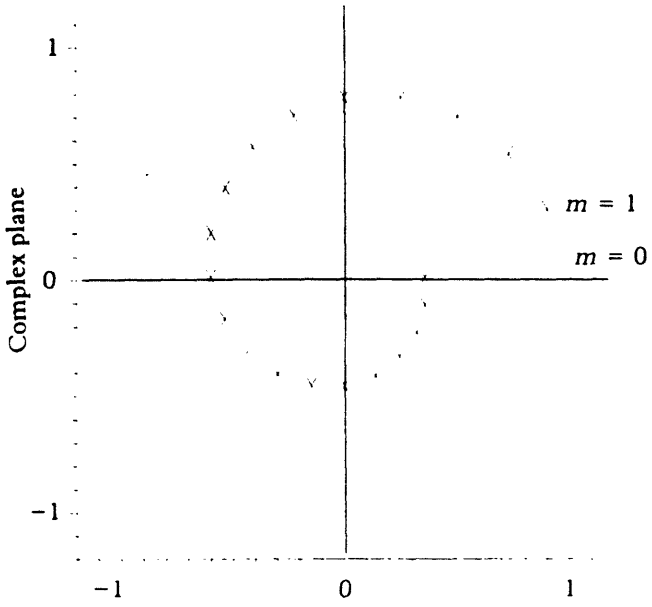
$$A[\cos(\omega nT + \phi) + j\sin(\omega nT + \phi)] = Ae^{j(\omega nT +}$$

Futhermore, ϕ can be "pulled out" and combine with A to form a convenient single complex coe cient Z that incorporates both amplitude and ph

$$\begin{aligned} y[n] &= A[\cos(\omega nT + \phi) + j\sin(\omega nT + \phi)] \\ &= Ae^{j(\omega nT + \phi)} \\ &= (Ae^{j\phi})e^{j\omega nT} \\ &= Ze^{j\omega nT}. \end{aligned}$$

The complex coefficients that make up the spec trum produced by the DFT can thus express bot phase and amplitude. The magnitude and angle the coefficient are, respectively, the amplitude a

Fig. 3. Representation on the complex plane of the spiral .95[cos(π/10) + j sin(π/10)]^m, m = 0, 1, . . . 20.



phase of the complex sinusoid corresponding to that coefficient.

Incidentally, we can easily represent a sinusoid with exponentially decaying or increasing magnitude. Consider an arbitrary point in the Z-plane and let $z = Re^{j\omega T}$. Then raising z to successive integer powers produces a spiral as shown in Fig. 3:

$$z^n = R^n e^{j\omega nT} = R^n[\cos(\omega nT) + j \sin(\omega nT)].$$

We have already seen that $e^{j\omega nT}$ produces a circular motion for $n = 0, 1, 2, \ldots, (N - 1)$. It is necessary only to show that R^n produces an exponential decay (for $R < 1$) or an exponential growth (for $R > 1$) or a constant (for $R = 1$).

The decay is customarily represented in terms of a *time constant* τ, where $R \triangleq e^{-T/\tau}$. Solving for τ, gives $\tau = -T/\ln(R)$ seconds, where ln is the log base e. A given point z in the complex plane can thus be represented as:

$$z = Re^{j\theta} \triangleq e^{-T/\tau}e^{j\omega T} = e^{-T/\tau + j\omega T}.$$

Raising z to successive integer powers,

$$z^n = R^n e^{jn\theta} = e^{-nT/\tau}e^{j\omega nT} = e^{-T/\tau}[\cos(\omega nT) + j \sin(\omega nT)].$$

We have shown that the relationship between complex numbers and sinusoids is unexpectedly strong. In particular, any complex number raised to successive integer powers generates a complex sinusoid of frequency ω with a magnitude that is either exponentially increasing, exponentially decreasing, or constant, depending on the sign of τ.

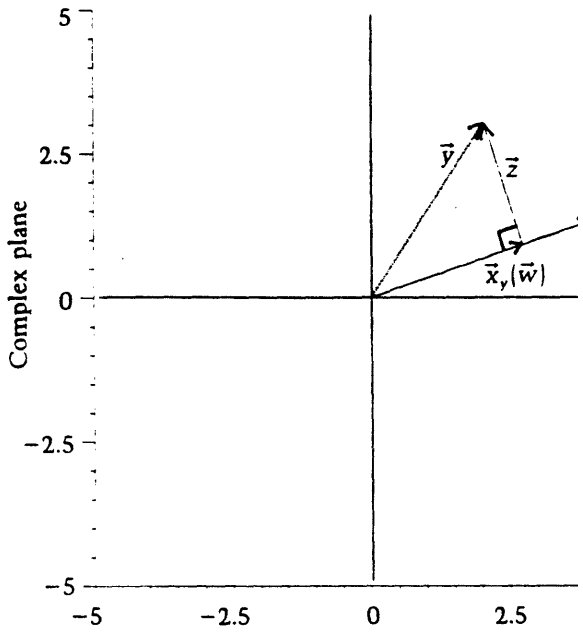
Vector Representation

The waveform y that serves as the input to the DFT can be viewed as a *vector* $\vec{y}$ in a multidimensional space where the *coordinates* of the vector are the successive samples of the waveform. Similarly, the spectrum Y produced by the DFT can be viewed as a vector whose coordinates are the spectral coefficients. We use vectors because they allow us to use simple rules of linear algebra. Everything gleaned from the vector viewpoint applies also to the sequence viewpoint. This section explains the vector viewpoint in detail.

The concept of a point in the Cartesian plane represented as an ordered pair $[x(1), x(2)]$ is familiar. Similarly, a point in space (or, more accurately, *3-space*) is an ordered triple $[x(1), x(2), x(3)]$. This idea can be generalized to N-dimensional space (or *N-space*) with a point represented as an ordered N-tuple. The nth coordinate of the point y in N-space is represented by the notation $y(n)$. Although this corresponds to our notation for a sequence, the notion is quite different. The coordinate $y(n)$ is not an element of a sequence but the nth coordinate of a single point y in N-dimensional space with the coordinates $[y(0), y(1), y(2), \ldots, y(N - 1)]$.

The point y can also be interpreted as a *vector*. (A vector is a line segment with a direction.) Thus the vector y extends from the origin (the point $x(n) = 0$ for all n) to the point y. If the sequence is complex, each coordinate in the corresponding vector is similarly complex, and each coordinate axis can be thought of as a complex plane rather than a line. The waveform y can be viewed as a vector y in N-space, where the coordinates of the vector are the successive samples of the waveform. We use the notation $\vec{y}$ for the vector y when speaking specifically in terms of vectors. We use the notation $y(n)$, both for the nth component of a sequence y and for the

Fig. 4. Orthogonal projection of the vector $\vec{y}$ onto the vector $\vec{x}$.



nth coordinate value of a vector $\vec{y}$. It should be clear from the context which interpretation is intended.

The rules of two-dimensional vector addition can be easily generalized to N dimensions. The sum of vectors $\vec{y}$ and $\vec{x}$ is defined as $\vec{y} + \vec{x} \triangleq [y(0) + x(0), y(1) + x(1), \ldots, y(N-1) + x(N-1)]$. In terms of sounds, this process is equivalent to *mixing*. Multiplication of a scalar α by a vector $\vec{y}$ is defined as $\alpha\vec{y} = [\alpha y(0), \alpha y(1), \ldots, \alpha y(N-1)]$, equivalent to adjusting the *gain* of a sound.

In the Cartesian coordinate plane, the distance from the point (a, b) to the origin is $\sqrt{a^2 + b^2}$ (or $\sqrt{|a|^2 + |b|^2}$ for complex numbers). This is called the *norm* of (a, b). (There are various other possible choices for the norm but this is the meaningful one for our purposes.) Similarly, in N-space, the norm of the vector $\vec{y}$ with complex coordinates is defined as

$$\|\vec{y}\| \triangleq \sqrt{|y(0)|^2 + |y(1)|^2 + \ldots + |y(N-1)|^2}$$

and is interpreted as the distance from the origin to the point y or as the length of the vector $\vec{y}$. If $\vec{y}_1$ and $\vec{y}_2$ are two vectors, then the distance between them is the norm of the difference of the two vectors. That is,

$$\|\vec{y}_1 - \vec{y}_2\| \triangleq$$
$$\sqrt{|y_1(0) - y_2(0)|^2 + |y_1(1) - y_2(1)|^2 + \ldots + |y_1(N-1) - y_2(N-1)|^2}.$$

The norm squared of the vector is interpreted as the *total energy* of the corresponding waveform. In the next section we show that the norm squared is equal to the *inner product* of the vector with itself.

A spectrum can be viewed as a vector. A spectrum vector, however, is in a different N-dimensional coordinate system from the corresponding time sequence. Indeed, the DFT can be viewed as a rotation of a vector from one coordinate system to another. To see how this works, it is necessary to introduce the idea of *projection*.

Vector Projection

We eventually show that the DFT produces each value of the spectrum by *projecting* the waveform vector onto a vector representing a particular sinusoidal component and taking the *coefficient of the projection*. In order to do this, we must generalize the notions of *projection* and *perpendicularity* an N-dimensional complex space.

Let us return, for the sake of simplicity, to a cussion of vectors of two real dimensions and projection from a geometric standpoint. Let $\vec{y}$ be two vectors of nonzero length (see Fig. 4). N choose a vector $\vec{w}$ lying along the same line wi such that $\vec{z}$, the vector from the end of $\vec{w}$ to th of $\vec{y}$, is perpendicular to the line *collinear* witl (Collinear means "lying along the same line.") the vector $\vec{w}$ is the *projection of* $\vec{y}$ *onto* $\vec{x}$. We the notation $\vec{x}_y$ for $\vec{w}$ in order to explicitly sho relationship between $\vec{x}$, $\vec{y}$, and $\vec{x}_y$.

Two vectors that are perpendicular are calle *thogonal*. We need to generalize the notion of thogonality to N complex dimensions so that can talk about projection of one N-dimension complex vector onto another. To do this, we in duce the idea of the *inner product*. Let $\vec{x}$ and $\vec{y}$ *real* N-dimensional vectors. Then $\langle\vec{x}, \vec{y}\rangle$ (the i product of $\vec{x}$ and $\vec{y}$) is defined as

$$\langle\vec{x}, \vec{y}\rangle \triangleq \sum_{n=0}^{N-1} x(n)y(n), \quad (x, y \text{ real}).$$

(We will have to redefine the inner product for complex vectors. But, for the moment, this definition is sufficient.) The inner product is not a vector but a scalar. In intuitive terms, the inner product is a measure of how "interdependent" two vectors are. If $\langle \vec{x}, \vec{y} \rangle = 0$, then $\vec{x}$ and $\vec{y}$ are orthogonal and "completely independent." For example, in the plane, the vectors $(1, -1)$ and $(1, 1)$ are orthogonal because

$$\langle \vec{x}, \vec{y} \rangle = x(0)y(0) + x(1)y(1) = 1 \cdot 1 + (-1) \cdot 1 = 1 - 1 = 0.$$

$\vec{x}$ is also orthogonal to $\vec{z} = (10, 10)$ because $\vec{z}$ lies along the same line as $\vec{y}$. In general, the orthogonality of two vectors is independent of their length.

Now let us extend the definition of inner product to include complex numbers. We would like to draw a graphic representation of two complex vectors that are obviously orthogonal and then see what we have to do to the inner product formula to make the inner product of those vectors equal to zero. Recall that a complex number can be represented graphically in two dimensions by assigning the real part to the horizontal axis and the imaginary part to the vertical axis. In general, we need four spatial dimensions to graph a two-dimensional complex vector $(\mathrm{Re}\{x(1)\}, \mathrm{Im}\{x(1)\}, \mathrm{Re}\{x(2)\}, \mathrm{Im}\{x(2)\})$. This causes trouble because a sheet of paper can represent only two dimensions well. However, if we constrain each vector to have a zero imaginary component along the first dimension and a zero real component along the second dimension, it can be represented in a graph of the complex plane. Consider two such vectors $\vec{x} = (1, j)$ and $\vec{y} = (1, -j)$. It is clear from the graphic representation (see Fig. 5) that $\vec{x}$ and $\vec{y}$ are perpendicular. The inner product, as defined above, of these vectors is

$$\langle \vec{x}, \vec{y} \rangle = x(0)y(0) + x(1)y(1) = 1 - j^2 = 1 + 1 = 2.$$

We have a problem. Since we know that the vectors are orthogonal, we would like the inner product to be equal to zero. A simple change to the inner product formula corrects this problem. We take the complex conjugate of the second vector:

$$\langle \vec{x}, \vec{y} \rangle = x(0)\overline{y(0)} + x(1)\overline{y(1)} = (1)\overline{(1)} + (j)\overline{(-j)} = 1 + j^2 = 1 - 1 = 0.$$

The process of conjugation does not affect the real part, so this definition works for real vectors as well. We choose, therefore, to redefine the inner product of two N-dimensional *complex* vectors $\vec{x}$ and $\vec{y}$ as

$$\langle \vec{x}, \vec{y} \rangle \triangleq \sum_{n=0}^{N-1} x(n)\overline{y(n)}, \quad (x, y \text{ complex}).$$

As a second example, consider the two complex one-dimensional vectors $(1 + j)$ and $(1 - j)$. These vectors have an inner product equal to $2j$ and so are not orthogonal. This makes sense if the complex plane is considered as a single "direction." That is, just as any two real numbers lie "along the same line" and are not orthogonal, two complex numbers lie "along the same plane" and are not orthogonal.

Finally, consider the inner product of a vector with itself, recalling that a complex number multiplied by its conjugate is equal to its magnitude squared:

$$\langle \vec{x}, \vec{x} \rangle \triangleq \sum_{n=0}^{N-1} x(n)\overline{x(n)} = \sum_{n=0}^{N-1} |x(n)|^2 \triangleq \|\vec{x}\|. \quad (2)$$

Thus the inner product of a vector with itself is equal to its norm squared.

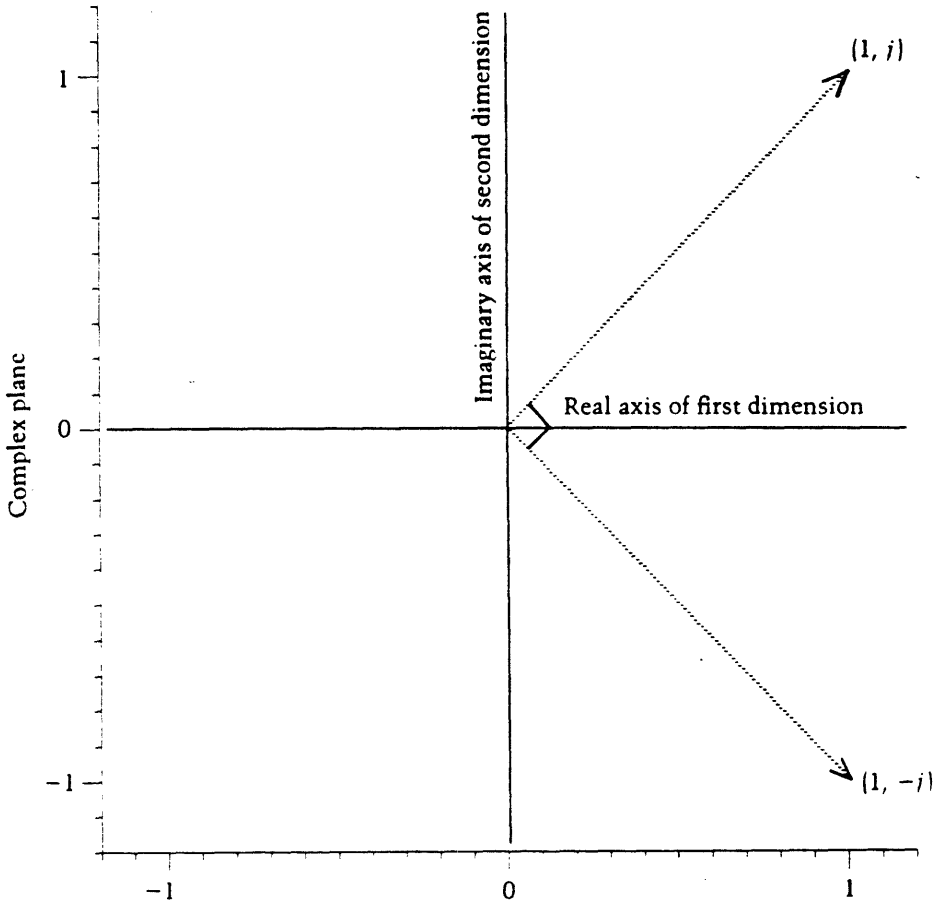
A useful property of the inner product is its *linearity*. An operator L is linear if $L(\alpha x) = \alpha L(x)$ and $L(x + y) = L(x) + L(y)$. For example, L could be a graphic equalizer such as is used in recording studios to color sound or remove unwanted color. A graphic equalizer is, ideally, linear. This implies that a sound can be amplified by a scalar α before or after entering the equalizer with the same effect. It also implies that two sounds x and y can be mixed together before or after entering the equalizer with the same effect. An example of a nonlinear operator is an automatic gain control, which boosts the gain when the input is soft and reduces the gain when the input is loud.

Let us prove that the inner product is a linear operation by using the distributive property of multiplication over addition:

$$\langle \vec{x}, (\vec{y}_1 + \vec{y}_2) \rangle \triangleq \sum_{n=0}^{N-1} x(n)\overline{(y_1(n) + y_2(n))}$$

$$= \sum_{n=0}^{N-1} x(n)\overline{(y_1(n) + y_2(n))}$$

Fig. 5. Vectors (1, j) and (1, −j) on the complex plane.



$$= \sum_{n=0}^{N-1} x(n)\overline{y_1(n)} + \sum_{n=0}^{N-1} x(n)\overline{y_2(n)}$$

$$= \langle \vec{x}, \vec{y}_1 \rangle + \langle \vec{x}, \vec{y}_2 \rangle.$$

Similarly,

$$\langle (\vec{x}_1 + \vec{x}_2), \vec{y} \rangle = \langle \vec{x}_1, \vec{y} \rangle + \langle \vec{x}_2, \vec{y} \rangle$$

and

$$\langle \alpha\vec{x}, \vec{y} \rangle = \alpha \langle \vec{x}, \vec{y} \rangle$$

$$\langle \vec{x}, \beta\vec{y} \rangle = \bar{\beta} \langle \vec{x}, \vec{y} \rangle$$

Due to this linearity in each of its operands, the inner product is referred to as a *bilinear* operation.

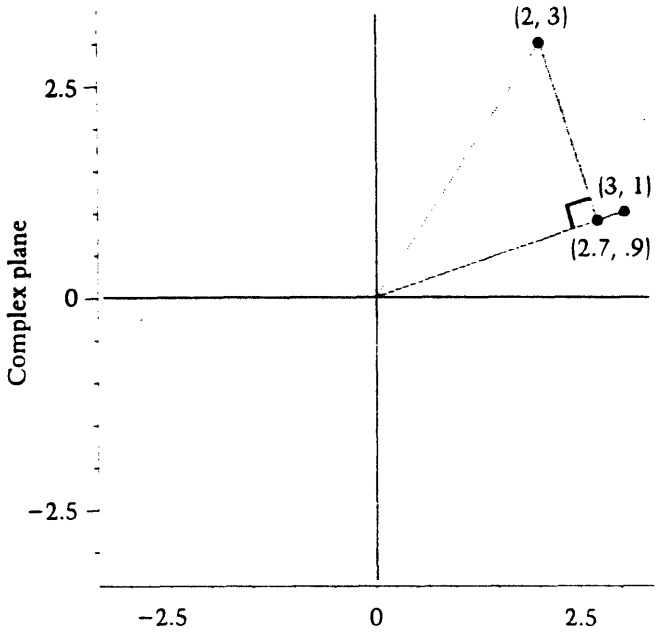
Using the inner product, we can define projection for two N-dimensional complex vectors. To make this easier to visualize, let us return to our or two-dimensional example (Fig. 4). We still con to speak in terms of the inner product for the of generality. Recall that the vectors $\vec{x}$ and $\vec{x}_y$ a fined to lie along the same line so $\vec{x}_y$ can be expr as $\alpha\vec{x}$, where α is real. Also, by definition, $(\vec{y}$ - is perpendicular to $\vec{x}$ so $\langle (\vec{y} - \vec{x}_y), \vec{x} \rangle = 0$. By t linearity of the inner product, $\langle \vec{y}, \vec{x} \rangle - \langle \vec{x}_y, \vec{x} \rangle$ and $\langle \vec{y}, \vec{x} \rangle = \langle \vec{x}_y, \vec{x} \rangle$. Substituting $\vec{x}_y = \alpha\vec{x}$,

$$\langle \vec{y}, \vec{x} \rangle = \langle \vec{x}_y, \vec{x} \rangle = \langle \alpha\vec{x}, \vec{x} \rangle = \alpha\langle \vec{x}, \vec{x} \rangle = \alpha\|\vec{x}$$

Thus, $\alpha = \langle \vec{y}, \vec{x} \rangle / \|\vec{x}\|^2$, and the projection of $\vec{y}$ is equal to

$$x_y = \alpha\vec{x} = \frac{\langle \vec{y}, \vec{x} \rangle}{\|\vec{x}\|^2} \vec{x}.$$

Fig. 6. Orthogonal projection of (2, 3) onto (3, 1).



The squared norm of the projection vector $\|\vec{x}_y\|^2$ is interpreted as a measure of how much energy in the direction of the vector $\vec{x}$ is present in the vector $\vec{y}$. The coefficient α is the *projection coefficient*.

For example, let $\vec{y} = (2, 3)$ and $\vec{x} = (3, 1)$ (see Fig. 6). Then the projection of y onto x is $(2 \cdot 3 + 3 \cdot 1)/(3 \cdot 3 + 1 \cdot 1)\vec{x} = (6 + 3)/(9 + 1)\vec{x} = .9\vec{x} = .9\,(3, 1) = (2.7, .9)$. If $\vec{x}$ and $\vec{y}$ are orthogonal, α is 0. Also, if $\vec{y} = \vec{x}$ then α is 1. As a final example, if $\vec{y}$ lies along the same line as $\vec{x}$, i.e. if $\vec{y} = \beta\vec{x}$, then

$$\alpha = \frac{\langle \beta\vec{x}, \vec{x} \rangle}{\|\vec{x}\|^2} = \beta$$

and the projection of $\vec{y}$ onto $\vec{x}$ is equal to $\alpha\vec{x} = \beta\vec{x} = \vec{y}$. Since the length of $\vec{x}$ has no effect on the result, we see that the projection onto $\vec{x}$ is really projection onto the line implied by $\alpha\vec{x}$ for all values of α. In more abstract terms, a vector is orthogonally projected onto a subspace (one-dimensional, in this case) spanned by $\alpha\vec{x}$.

The definitions of inner product and projection can easily be transferred from vectors to sequences. For example, the inner product of two waveforms $x(n)$ and $y(n)$ can be defined as $\langle x, y \rangle \triangleq \langle \vec{x}, \vec{y} \rangle$.

We will show that the spectrum produced by the DFT is actually a set of complex projection coefficients, (such as α above, but scaled by N), obtained by projecting the waveform (a specific vector $\vec{y}$) onto a set of complex sinusoids (each like $\vec{x}$ above), each of which is orthogonal to all the others. In this manner, the DFT measures the energy and phase of each sinusoidal component in the waveform.

Orthogonal Bases

Our goal is to view the DFT as a change of *orthogonal bases* from the standard coordinate system basis to a basis made up of complex sinusoids. In this section, we define and examine orthogonal bases and lay the groundwork for showing that the set of complex sinusoids used by the DFT forms such a basis.

We have seen that any two vectors are orthogonal if their inner product is 0. Similarly, if the inner product of each pair of vectors in a set of M N-dimensional nonzero vectors $\{\vec{x}_k\}$ is equal to zero, the set is called an *orthogonal set*. Furthermore, if M (the number of vectors in the set) is equal to N, then $\{\vec{x}_k\}$ is called a *basis* for N-space, and we can show that every vector $\vec{y}$ in N-space can be expressed as a weighted sum of the vectors comprising the set $\{\vec{x}_k\}$. First, let us consider a familiar example, in the form of a *coordinate system*. When we plot a vector (3, 2) in the plane, we are expressing the vector as a weighted sum of two orthogonal vectors (1, 0) and (0, 1) so that (3, 2) = 3(1, 0) + 2(0, 1). Note that we need two orthogonal vectors because we are in a two-dimensional space. The values α_k are simply the coordinates. Similarly, for N-space, the coordinate system basis is $\{\vec{x}_k\}$ with $k = 0, 1, 2, \ldots, (N - 1)$ defined by the condition that kth coordinate of $\vec{x}_k$ (counting the first coordinate as 0) is 1 and all the other coordinates are 0. For example, if $N = 3$, the basis is composed of the vectors $\vec{x}_0 = (1, 0, 0)$, $\vec{x}_1 = (0, 1, 0)$, and $\vec{x}_2 = (0, 0, 1)$. It is easy to see that the set $\{\vec{x}_k\}$ is orthogonal. Therefore, since there are N vectors in the set and since each has nonzero length, it forms an orthogonal basis for N-space. To see that any vector $\vec{y}$ can be expressed as a weighted sum of the vectors in the set $\{\vec{x}_k\}$ observe that, by the definition of a coordinate system,

$$\vec{y} = [y(0), y(1), y(2), \ldots, y(N-1)]$$
$$= y(0)\vec{x}_0 + y(1)\vec{x}_1 + y(2)\vec{x}_2 + \ldots + y(N-1)\vec{x}_{N-1}. \quad (4)$$

We have shown how to determine the α_k coefficients for the familiar coordinate system orthogonal basis. For an arbitrary basis, we need to solve for a set of complex scalars $\{\alpha_k\}$, $k = 0, 1, \ldots, (N-1)$ such that

$$\vec{y} = \sum_{k=0}^{N-1} \alpha_k \vec{x}_k. \quad (5).$$

We shall show that the complex coefficient α_k is actually the orthogonal projection of $\vec{y}$ onto $\vec{x}_k$.

Let us form a sequence $Y(k)$ of the inner products of $\vec{y}$ with each of the orthogonal basis vectors in the set $\{\vec{x}_k\}$.

$$Y(k) = \langle \vec{y}, \vec{x}_k \rangle = \left\langle \left(\sum_{l=0}^{N-1} \alpha_l \vec{x}_l \right), \vec{x}_k \right\rangle.$$

By recalling the definitions of the inner product and summation operations, it is possible to use the distributive property of multiplication over addition to move the summation outside of the inner product.

$$Y(k) = \sum_{l=0}^{N-1} \langle \alpha_l \vec{x}_l, \vec{x}_k \rangle = \sum_{l=0}^{N-1} \alpha_l \langle \vec{x}_l, \vec{x}_k \rangle.$$

The set $\{\vec{x}_k\}$ is defined to be orthogonal, which implies that the inner product of each pair of vectors in the set is zero. However, in Eq. (2), we showed that the inner product of a vector with itself is the norm squared of that vector. That is,

$$\langle \vec{x}_l, \vec{x}_k \rangle = \begin{cases} 0, & l \neq k \\ \|\vec{x}_k\|^2, & l = k. \end{cases}$$

So all the terms in the summation disappear except for the one where k equals l. Thus

$$\langle \vec{y}, \vec{x}_k \rangle = \alpha_k \|\vec{x}_k\|^2$$

and we can, at last, solve for α_k:

$$\alpha_k = \frac{\langle \vec{y}, \vec{x}_k \rangle}{\|\vec{x}_k\|^2}.$$

Substituting $\{\alpha_k\}$ into Eq. (5),

$$\vec{y} = \sum_{k=0}^{N-1} \frac{\langle \vec{y}, \vec{x}_k \rangle}{\|\vec{x}_k\|^2} \vec{x}_k.$$

But we recognize $\alpha_k \vec{x}_k$ from Eq. (3) as simply the orthogonal projection of $\vec{y}$ onto $\vec{x}_k$. Thus we have proven that for any orthogonal basis $\{x_k\}$, it is possible to represent $\vec{y}$ as a sum of orthogonal projections of $\vec{y}$ onto $\vec{x}_k$ and $\{\vec{x}_k\}$ is said to *span* N-space.

Returning to the familiar coordinate system example given in Eq. (4) we now see it is possible to reinterpret each coordinate value $y(n)$ as the projection of $\vec{y}$ on the line collinear with the x_nth basis vector. We can recover $\vec{y}$ from the sequence of its coordinates by multiplying each coordinate by the appropriate orthogonal basis vector and summing. The coordinate system is not, however, the only possible orthogonal basis for N-space; there are infinitely many orthogonal bases. For example, in 2 dimensions, the vectors $\vec{x}_0 = [\sqrt{2}/2, \sqrt{2}/2]$ and $\vec{x}_1 = [\sqrt{2}/2, -\sqrt{2}/2]$ form a perfectly adequate orthogonal basis. In this new coordinate system, we can find the coordinates of the vector $\vec{Y}$ that corresponds to the vector $\vec{y} = [3, 4]$ in the original coordinate system by projecting $\vec{y}$ onto each of the new orthogonal basis vectors:

$$Y(0) = \frac{\langle \vec{y}, \vec{x}_0 \rangle}{\|\vec{x}_0\|^2} = \frac{\langle [3, 4], [\sqrt{2}/2, \sqrt{2}/2] \rangle}{\|[\sqrt{2}/2, \sqrt{2}/2]\|^2}$$
$$= \frac{3\sqrt{2}/2 + 4\sqrt{2}/2}{(\sqrt{2}/2)^2 + (\sqrt{2}/2)^2} = 3.5\sqrt{2}$$
$$Y(1) = \frac{\langle \vec{y}, \vec{x}_1 \rangle}{\|\vec{x}_1\|^2} = \frac{\langle [3, 4], [\sqrt{2}/2, -\sqrt{2}/2] \rangle}{\|[\sqrt{2}/2, -\sqrt{2}/2]\|^2}$$
$$= \frac{3\sqrt{2}/2, -4\sqrt{2}/2}{(\sqrt{2}/2)^2 + (-\sqrt{2}/2)^2} = -.5\sqrt{}$$

Vector $\vec{Y}$ is thus equal to $[3.5\sqrt{2}, -.5\sqrt{2}]$. Note that $\vec{y}$ and $\vec{Y}$ are the same length:

$$\|\vec{y}\| = \sqrt{3^2 + 4^2} = \sqrt{25} = 5$$
$$\|\vec{Y}\| = \sqrt{(3.5\sqrt{2})^2 + (-.5\sqrt{2})^2} = \sqrt{24.5 + .5} =$$

The lengths are the same because the set $\{\vec{x}_0, \vec{x}$ and the original basis $\{(1, 0), (0, 1)\}$ are *orthonor*. An orthonormal basis is an orthogonal basis in which each vector in the set has a norm of 1. In our example, $\|\vec{x}_0\| = \|\vec{x}_1\| = 1$.

Similarly, for N-space, a translation of a vector $\vec{y}$ from the standard orthonormal coordinate system $\{[1, 0, 0, \ldots], [0, 1, 0, \ldots], \ldots\}$ to some other orthonormal basis $\{\vec{x}_k\}$ is given by

$$\vec{Y} = [\langle \vec{y}, \vec{x}_0 \rangle, \langle \vec{y}, \vec{x}_1 \rangle, \ldots, \langle \vec{y}, \vec{x}_{N-1} \rangle]$$

$$Y(k) = \langle \vec{y}, \vec{x}_k \rangle = \sum_{n=0}^{N-1} y(n)\overline{x_k(n)}.$$

Let us compare this equation with the equation for the DFT. If we make the assumption (a wild assumption, at the moment) that the set of sinusoids $\{e^{i\omega_k nT}\}$ forms an orthogonal basis in N-space, we need only set the basis $\{x_k\}$ equal to the set of sinusoids to produce the DFT equation:

$$Y(k) = N\alpha(k) = \sum_{n=0}^{N-1} y(n)\overline{x_k(n)} = \sum_{n=0}^{N-1} y(n)e^{-i\omega_k nT}.$$

The spectrum can then be seen as the coefficients resulting from an orthogonal projection of a waveform onto a sinusoidal basis set, where each of the sinusoids is a harmonic in the output spectrum. The IDFT is the sum of the sinusoidal basis functions, scaled by the corresponding spectral coefficients.

It remains to be shown that the set of vectors $\{\vec{x}_k\} = \{e^{i\omega_k nT}\}$ does indeed form an orthogonal (though not orthonormal) basis in N-space.

Orthogonality of Complex Sinusoids

We have almost reached the point where the DFT can be understood in terms of vector projection. In this section, we tie up the last loose end by showing that an appropriately chosen set of N complex sinusoids forms an orthogonal basis in N-space.

A complex sinusoid in N-space with unit amplitude is simply a vector $\vec{x}$ whose coordinates are given by the functional definition $x(n) = e^{i\omega nT}$. We will show that the set of vectors $\{\vec{x}_k\} = e^{i\omega_k nT}$, where $\omega_k = 2\pi(k/N)f_s$, forms an orthogonal basis in N-space. In order to prove this, we must show that the inner product of every pair of vectors $\langle \vec{x}_k, \vec{x}_l \rangle$ in $\{x_k\}$ is zero.

$$\langle \vec{x}_k, \vec{x}_l \rangle = \sum_{n=0}^{N-1} e^{i\omega_k nT}\overline{e^{i\omega_l nT}}$$

$$= \sum_{n=0}^{N-1} e^{i\omega_k nT}e^{-i\omega_l nT}$$

$$= \sum_{n=0}^{N-1} [e^{i(\omega_k - \omega_l)T}]^n.$$

Notice that the summation produces a geometric series. Thus, we can use the following well-known theorem (see, for example, Spitzbart 1975), which expresses a geometric series in closed form:

$$\sum_{n=0}^{N-1} z^n = \frac{1 - z^N}{1 - z}.$$

Using this theorem,

$$\sum_{n=0}^{N-1} [e^{i(\omega_k - \omega_l)T}]^n = \frac{1 - [e^{i(\omega_k - \omega_l)T}]^N}{1 - e^{i(\omega_k - \omega_l)T}} = \frac{1 - e^{i(\omega_k - \omega_l)NT}}{1 - e^{i(\omega_k - \omega_l)T}}.$$

We can simplify this expression by expanding ω_k and ω_l. Thus $(\omega_k - \omega_l)NT = (2\pi k/NT - 2\pi l/NT)NT = 2\pi(k - l)$ and furthermore:

$$\langle \vec{x}_k, \vec{x}_l \rangle = \frac{1 - e^{i2\pi(k-l)}}{1 - e^{i2\pi(k-l)/N}}.$$

If $k \neq l$, the numerator is 0 and the denominator is nonzero, so $\langle \vec{x}_k, \vec{x}_l \rangle = 0$ and the vectors $\vec{x}_k$ and $\vec{x}_l$ are orthogonal. If $k = l$, we have this expression:

$$\langle \vec{x}_k, \vec{x}_k \rangle = \|\vec{x}_k\|^2 = \sum_{n=0}^{N-1} e^{i\omega_k nT}\overline{e^{i\omega_k nT}}$$

$$= \sum_{n=0}^{N-1} e^{i\omega_k nT - i\omega_k nT} = \sum_{n=0}^{N-1} 1 = N.$$

The fact that we get an extra factor of N indicates that the set of complex sinusoidal vectors $\{\vec{x}_k\}$ is not an orthonormal set.

Reinterpreting $\{\vec{x}_k\}$ as a set of waveforms $x_k(n)$, each member of the set is a complex sinusoid resulting from raising $e^{i\omega_k T}$ to successive integer powers: $x_k(n) = e^{i\omega_k Tn}$, $n = 0, 1, 2, \ldots, (N - 1)$. This set of sinusoids is the orthogonal basis to which the DFT translates the time-domain waveform.

It is interesting to note that neither sines alone nor cosines alone form an orthogonal basis in N-space. Let us assume that the set of vectors $\{\vec{x}_k\} = \cos(\omega_k nT) = \cos(2\pi kn/N)$, $k = 0, 1, 2, \ldots, (N - 1)$

is an orthogonal basis. But two of the vectors in the set, $\cos(\omega_k nT)$ and $\cos(\omega_{N-k} nT)$ are equal. This means that their inner product is not equal to zero, contradicting our orthogonality hypothesis. Indeed, N cosines equally spaced in frequency between 0 Hz and the sampling rate provide only $N/2$ orthogonal vectors. A similar argument shows that sines alone are insufficient.

The DFT at Last

We have shown that the set of complex sinusoidal waveforms $\{x_k(n)\} = \{e^{j2\pi kn/N}\}$ corresponds to a set of orthogonal basis vectors $\{\vec{x}_k\}$. Therefore, any waveform $y(n)$ can be expressed as a linear combination of $\{e^{j2\pi kn/N}\}$. This is expressed as follows:

$$y(n) = \sum_{n=0}^{N-1} \alpha_k x_k(n), \quad \alpha_k = \frac{\langle y, x_k \rangle}{\|x_k\|^2}.$$

This is Fourier's theorem for waveforms of length N. The set of complex coefficients $\{\alpha_k\}$ is the spectrum of $y(n)$ at frequency ω_k and is usually viewed as a sequence $\alpha(k)$.

The DFT is usually defined as the sequence of projection coefficients multiplied by N (Rabiner and Gold 1975). This eliminates a division by N, the squared norm of the basis vector.

$$\begin{aligned} Y(k) &\triangleq N\alpha(k) \\ &= N \frac{\langle \vec{y}, \vec{x}_k \rangle}{\|\vec{x}_k\|^2} \\ &= \langle \vec{y}, \vec{x}_k \rangle \\ &= \sum_{n=0}^{N-1} y(n) e^{-j\omega_k nT} \\ &\triangleq \mathrm{DFT}_k(y). \end{aligned}$$

The extra factor of N requires a corresponding $1/N$ scaling term to appear in the inverse DFT:

$$y(n) = \frac{1}{N} \sum_{k}^{N-1} Y(k) e^{j\omega_k nT}.$$

To recapitulate, the DFT values are coefficients of projection onto a sinusoidal basis, and the inverse DFT is just the reconstruction of the original vector (or waveform) in terms of the sinusoidal basis functions.

To prove that the IDFT and DFT form an identi pair, let $y(n)$, $n = 0, 1, 2, \ldots, (N-1)$ be any sequence of N complex numbers and define the spe trum of $y(n)$ as $Y(k) = \mathrm{DFT}_k(y)$. We will show tha $\mathrm{IDFT}_n(Y) = y(n)$.

$$\begin{aligned} \mathrm{IDFT}_n(Y) &\triangleq \frac{1}{N} \sum_{k=0}^{N-1} Y(k) e^{j\omega_k nT} \\ &= \frac{1}{N} \sum_{k=0}^{N-1} \left[\sum_{m=0}^{N-1} y(m) e^{-j\omega_k mT} \right] e^{j\omega_k nT}. \end{aligned}$$

We can switch the order of the summation, due to the distributive property of multiplication ove addition.

$$\begin{aligned} &\frac{1}{N} \sum_{k=0}^{N-1} \left[\sum_{m=0}^{N-1} y(m) e^{-j\omega_k mT} \right] e^{j\omega_k nT} \\ &\quad = \frac{1}{N} \sum_{m=0}^{N-1} y(m) \sum_{k=0}^{N-1} e^{-j\omega_k mT} e^{j\omega_k} \\ &\quad = \frac{1}{N} \sum_{m=0}^{N-1} y(m) \sum_{k=0}^{N-1} e^{j\omega_k (n-m)T}. \end{aligned}$$

Notice that the rightmost summation in the fina equation is always equal to N or 0, depending on the relationship between the values of n and m.

$$\sum_{k=0}^{N-1} e^{j\omega_k (n-m)T} = \begin{cases} N, & n = m \\ 0, & n \neq m. \end{cases}$$

We can, therefore, replace this summation with a *delta function* $\delta_{m,n}$ which is equal to 1 when $n = m$, and 0 otherwise.

$$\mathrm{IDFT}_n(Y) = \frac{1}{N} \sum_{m=0}^{N-1} y(m) N \delta_{m,n} = \sum_{m=0}^{N-1} y(m) \delta_m$$

The effect of the delta function is to pick out th nth element of y so:

$$\sum_{m=0}^{N-1} y(m) \delta_{m,n} = y(n).$$

We have shown that $\mathrm{IDFT}_n(\mathrm{DFT}(y)) = y(n)$. It can l easily shown in a similar manner that $\mathrm{DFT}_k(\mathrm{IDFT} = Y(k)$.

Conclusion and Preview of Coming Attractions

The DFT takes a waveform as input and produces a spectrum. It does this by projecting the samples of the waveform, viewed as a vector, onto a set of complex sinusoids. Each element of the spectrum is a complex number that represents the amplitude and phase of the corresponding spectral component (harmonic). The spectral components are equally spaced in frequency. There are as many spectral components as there are samples in the original waveform. The DFT functions by producing the coefficients of the projection of the waveform onto each of a set of basis sinusoids. Each spectral coefficient is the inner product of the waveform with one of the basis sinusoids.

The IDFT undoes the effect of the DFT. It takes a spectrum as input and produces a time-domain representation of the sound, by multiplying the spectral coefficients by a set of sinusoids.

We are now in a position to examine the behavior of the DFT and IDFT under various types of inputs. In the second part of this tutorial, we examine several important properties of the DFT, and examine concepts useful for understanding operations on sound such as sampling-rate conversion and convolution. We also define the *Z-transform* and extend the DFT to the continuous domain. Since our primary interest is sound rather than N-dimensional vectors, we drop the vector notation and return to waveforms. It is hoped that the reader is now sufficiently familiar with the two representations to feel comfortable moving between them.

Acknowledgments

I would like to thank Julius Smith for his inspiring and patient tutelage. Thanks also to Bill Schottstaedt, Doug Keislar, and Ami Radunskaya for their helpful proofreading and suggestions.

References

Aho, A. V., J. E. Hopcroft, and J. D. Ullman. 1974. *The Design and Analysis of Computer Algorithms.* Reading, Massachusetts: Addison-Wesley.

Dolson, M. 1983. "A Tracking Phase Vocoder And Its Use In The Analysis Of Ensemble Sounds." Ph.d. thesis. Pasadena: California Institute of Technology.

Moore, F. R. 1978a. "An Introduction to the Mathematics of Digital Signal Processing. Part I: Algebra, Trigonometry, and the Most Beautiful Formula in Mathematics." *Computer Music Journal* 2(1):38–47. Reprinted in J. Strawn, ed. 1985. *Digital Audio Signal Processing: An Anthology.* Los Altos: Kaufmann, pp. 1–67.

Moore, F. R. 1978b. "An Introduction to the Mathematics of Digital Signal Processing. Part II: Sampling, Transforms, and Digital Filtering." *Computer Music Journal* 2(2):38–60. Reprinted in J. Strawn, ed. 1985. *Digital Audio Signal Processing: An Anthology.* Los Altos: Kaufmann, pp. 1–67.

Rabiner, L. R., and B. Gold. 1975. *Theory and Application of Digital Signal Processing.* Englewood Cliffs, New Jersey: Prentice-Hall, Inc.

Smith, J. O. 1981. "An Introduction to Digital Filter Theory." Stanford: Center for Computer Research in Music and Acoustics, Stanford University. Reprinted in John Strawn, ed. 1985. *Digital Audio Signal Processing: An Anthology.* Los Altos: Kaufmann, pp. 69–136.

Spitzbart, A. 1975. *Calculus with Analytic Geometry.* Glenview, Illinois: Scott, Foresman and Company.

Appendix A: Motivation behind the Invention of Complex Numbers

Historically, the need for complex numbers arose out of attempts to factor polynomials such as $f(x) = x^2 - 2x + 2$. To find the roots of $f(x)$, we use the quadratic formula. The quadratic formula gives the roots of a quadratic equation $ax^2 + bx + c$ as:

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}.$$

Applying the quadratic formula to determine the roots of $f(x)$ gives:

$$\frac{2 \pm \sqrt{4 - 8}}{2} = \frac{2 \pm \sqrt{-4}}{2}.$$

This equation cannot be solved because the expression $\sqrt{-4}$ is undefined. To remedy this situation, let us define a number $j \triangleq \sqrt{-1}$ such that $j^2 \triangleq -1$. This allows us to simplify any expression of the form $\sqrt{-n}$ where $n \geq 0$ by factoring out j and transforming the expression into the form $j\sqrt{n}$. The roots of $f(x)$ are thus $\pm 2j$. Having added a single number j to our number system, we can now find the roots of any quadratic equation.

Trigonometric Identities *(cont'd)*

$$\cos(A) + \cos(B) = 2\cos\left(\frac{A+B}{2}\right)\cos\left(\frac{A-B}{2}\right)$$

$$\sin(A) + \sin(B) = 2\sin\left(\frac{A+B}{2}\right)\cos\left(\frac{A-}{2}\right.$$

$$\cos(A) - \cos(B) = -2\sin\left(\frac{A+B}{2}\right)\sin\left(\frac{A-B}{2}\right)$$

$$\sin(A) - \sin(B) = 2\cos\left(\frac{A+B}{2}\right)\sin\left(\frac{A-}{2}\right.$$

$$\cos^2(A) - \cos^2(B) = -\sin(A+B)\sin(A-B)$$

$$\sin^2(A) - \sin^2(B) = \sin(A+B)\sin(A-B)$$

$$\cos^2(A) - \sin^2(B) = \cos(A+B)\cos(A-B)$$

$$\tan(A) + \tan(B) = \frac{\sin(A+B)}{\cos(A)\cos(B)}$$

$$\tan(\theta) \triangleq \frac{\sin(\theta)}{\cos(\theta)}$$

$$\tan^2\left(\frac{\theta}{2}\right) = \frac{1-\cos(\theta)}{1+\cos(\theta)}$$

$$\tan\left(\frac{\theta}{2}\right) = \frac{1-\cos(\theta)}{\sin(\theta)}$$

$$\tan\left(\frac{\theta}{2}\right) = \frac{\sin(\theta)}{1+\cos(\theta)}$$

$$\cos(\theta) = \frac{1-t^2}{1+t^2}$$

$$\sin(\theta) = \frac{2t}{1+t^2}, \qquad \left[t \triangleq \tan\left(\frac{\theta}{2}\right)\right]$$

$$\tan(\theta) = \frac{2t}{1-t^2}$$

$$\tan(A+B) = \frac{\tan(A)+\tan(B)}{1-\tan(A)\tan(B)}$$

Appendix B: Complex Arithmetic and Trigonometric Identities

The symbol $\triangleq$ means "is defined as"; z stands for a complex number; and r, θ, x, and y are real numbers.

Complex Number Identities

$$j \triangleq \sqrt{-1}$$

$$z \triangleq x + jy \triangleq re^{j\theta}$$

$$x = r\cos(\theta)$$

$$y = r\sin(\theta)$$

$$|z_1 z_2| = |z_1||z_2|$$

$$\left|\frac{z_1}{z_2}\right| = \frac{|z_1|}{|z_2|}$$

$$\angle z_1 z_2 = \angle z_1 + \angle z_2$$

$$\angle \frac{z_1}{z_2} = \angle z_1 - \angle z_2$$

$$r = |z| = \sqrt{x^2 + y^2}$$

$$\theta = \angle z = \tan^{-1}\left(\frac{y}{x}\right)$$

$$e^x \triangleq \lim_{n\to\infty}\left(1 + \frac{x}{n}\right)^n = \sum_{n=0}^{\infty}\frac{x^n}{n!}$$

$$e = 2.7\ 1828\ 1828\ 4\ldots$$

$$|e^{j\theta}| = 1$$

$$\angle r = 0$$

$$|z| = |r||e^{j\theta}| = r$$

$$\angle z = \angle r + \angle e^{j\theta} = \theta$$

$$z_1 z_2 = (x_1 + jy_1)(x_2 + jy_2)$$
$$= (x_1x_2 - y_1y_2) + j(x_1y_2 + x_2y_1)$$

$$z_1 z_2 = (r_1 e^{j\theta_1})(r_2 e^{j\theta_2})$$
$$= (r_1 r_2)e^{j(\theta_1 + \theta_2)}$$

$$\bar{z} \triangleq x - jy = re^{-j\theta}$$

$$z\bar{z} = |z|^2 = x^2 + y^2 = r^2$$

Trigonometric Identities

$$e^{j\theta} = \cos(\theta) + j\sin(\theta)$$

$$e^{jn\theta} = \cos(n\theta) + j\sin(n\theta)$$

$$\sqrt{e^{j\theta}} = e^{j\theta/2}$$

$$\cos^2\theta + \sin^2\theta = 1$$

$$\sin(\theta) = \frac{e^{j\theta} - e^{-j\theta}}{2j}$$

$$\cos(\theta) = \frac{e^{j\theta} + e^{-j\theta}}{2}$$

$$\sin(-\theta) = -\sin(\theta)$$

$$\cos(-\theta) = \cos(\theta)$$

$$\cos(A + B) = \cos(A)\cos(B) - \sin(A)\sin(B)$$

$$\sin(A + B) = \sin(A)\cos(B) + \cos(A)\sin(B)$$

$$\cos^2(\theta) = \frac{1}{2}[1 + \cos(2\theta)]$$

$$\sin^2(\theta) = \frac{1}{2}[1 - \cos(2\theta)]$$

$$\cos(A)\cos(B) = \frac{1}{2}[\cos(A + B) + \cos(A - B)]$$

$$\sin(A)\sin(B) = \frac{1}{2}[\cos(A - B) - \cos(A + B)]$$

$$\sin(A)\cos(B) = \frac{1}{2}[\sin(A + B) + \sin(A - B)]$$

$$\cos(A)\sin(B) = \frac{1}{2}[\sin(A + B) - \sin(A - B)]$$

David A. Jaffe
Center for Computer Research in Music and Acoustics
Stanford University
Stanford, California 94305 USA

Spectrum Analysis Tutorial, Part 2: Properties and Applications of the Discrete Fourier Transform

Review of Part One

In part one of this tutorial (Jaffe 1987), we introduced the discrete Fourier transform (DFT). To review, the DFT takes a waveform as input and produces as output the spectrum of that waveform. One way to understand this process is to consider the samples of the waveform as a vector and to see the DFT as the projection of this vector onto a set of complex sinusoidal basis vectors. In this manner, the DFT produces a sequence of spectral components equally spaced in frequency, with a length equal to that of the original waveform. Each element of the spectrum is a coefficient of the projection given by the inner product of the waveform with one of the basis sinusoids. This coefficient can be represented in polar coordinates to give the amplitude and phase of the corresponding sinusoid.

The equation for the DFT is:

$$\mathrm{DFT}_k\{y\} \triangleq Y(k) \triangleq \sum_{n=0}^{N-1} y(n)e^{-j\omega_k nT},$$
$$k = 0, 1, \ldots, N-1,$$

where $\omega_k = 2\pi k f_s/N$, f_s is the sampling rate, $T = 1/f_s$ is the sampling period, and the symbol $\triangleq$ means "is defined as." This equation was explained in depth in the first part of this tutorial.

The inverse discrete Fourier transform (IDFT) undoes the effect of the DFT. It takes a spectrum as input and produces a time-domain representation of the sound, by multiplying the spectral coefficien by a set of sinusoids. The equation for the IDFT

$$\mathrm{IDFT}_n\{Y\} \triangleq y(n) \triangleq \frac{1}{N}\sum_{k=0}^{N-1} Y(k)e^{j\omega_k nT},$$
$$k = 0, 1, \ldots, N-1,$$

We are now in a position to examine the relati ship between the time- and frequency-domain re sentations. We examine several important prope ties of the DFT and discuss their implications fo applications such as sampling-rate conversion ar linear digital filtering. We also discuss two relati of the DFT, the Z-transform and the continuous Fourier transform. We begin by reexamining how the input and output of the DFT can be interpre

Physical Interpretations of the Input and Output of the DFT

The input and output of the DFT are defined as quences of length N. More precisely, they are int preted as *N-sequences*. N-sequences have two al native physical interpretations. One possibility i interpret an N-sequence as N points of a sequen of finite duration, preceded and followed by infinitely many zeros. Alternatively, it is possible tc terpret an N-sequence as one period of an infini periodic sequence. For the second interpretation the value of the N-sequence y at the nth point i $y(n \text{ MOD } N)$, where x MOD y is the remainder an integer divide of x by y. Both interpretations equally valid. We choose the interpretation appr priate to a given context.

Computer Music Journal, Vol. 11, No. 3, Fall 1987,

Fig. 1. Complex conjugate pair.

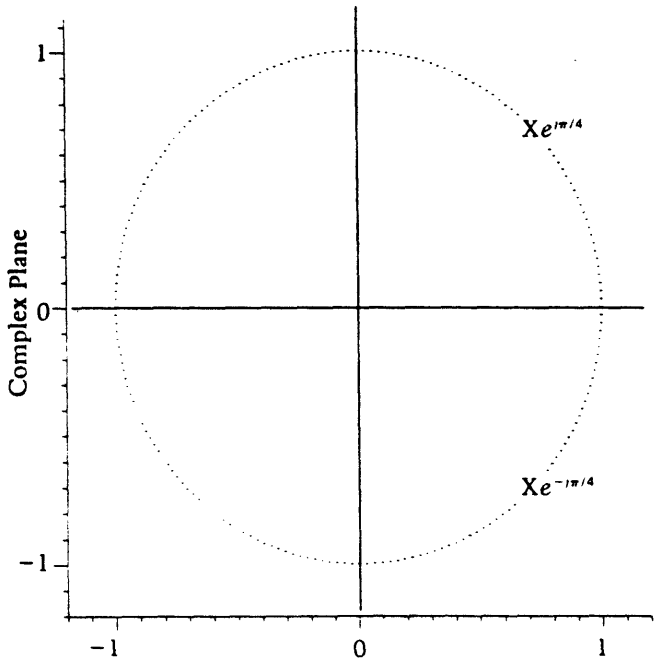
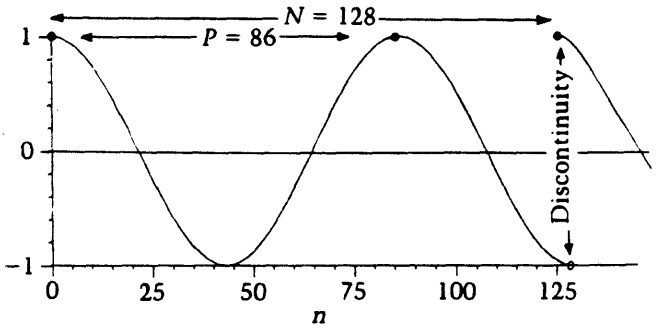


Fig. 2. Implied discontinuity of DFT periodicity assumption.



When considering the output of the DFT, it is customary to use the periodic interpretation. The periodicity of the spectrum is a direct result of the sampling process, which maps sinusoids of the form $e^{j2\pi(lN + n)/N}$ to the sinusoid $e^{j2\pi n/N}$, with l equal to any integer. Because $e^{j2\pi l} = 1$,

$$e^{j2\pi(lN + k)/N} = e^{jl2\pi}e^{j2\pi k/N} = e^{j2\pi k/N},$$

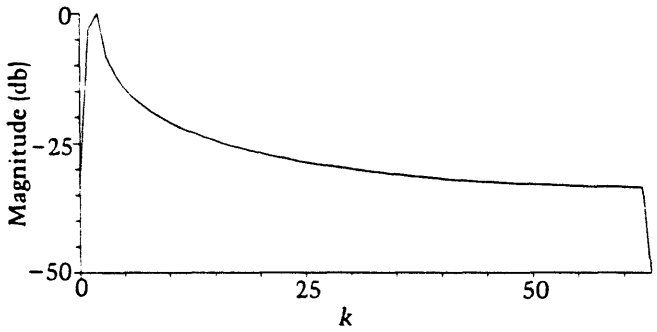
for all $l = -\infty, \ldots, -1, 0, 1, \ldots, \infty$.

That is, any integer multiple of N may be added to or subtracted from a frequency with no effect. In part one we defined the range of the spectrum as extending from 0 to f_s. Because the spectrum is periodic, it is just as valid to use any range that extends over a total of f_s Hz. We often choose to view the spectrum from $-f_s/2$ to $f_s/2$ (not including $f_s/2$, because it is equivalent to $-f_s/2$), which corresponds to a radian frequency $\omega_k T$ ranging from $-\pi$ to π (not including π). Notice that the point corresponding to a frequency of 0 lies in the middle of this range, which makes explicit the relationship between the sinusoids at ω_k and ω_{-k}. The two are symmetrically placed about the real axis. That is, the two are simply a complex conjugate pair: $e^{j\omega_{-k}} = e^{-j\omega_k} = \overline{e^{j\omega_k}}$. (See Fig. 1.) The periodicity of the spectrum requires that care be taken when doing sampling rate conversion, digital-to-analog conversion (DAC) and analog-to-digital conversion (ADC). For example, when digitizing a sound with an ADC, it is necessary to filter at half the sampling rate before sampling. Otherwise, all frequencies above $f_s/2$ and below $-f_s/2$ alias into the range $-f_s/2$ to $f_s/2$. A detailed discussion of this issue appears in subsequent sections of this tutorial.

Similarly, it is often convenient to place the origin of the *time* axis in the middle of the N-sequence. Although it may seem strange to move samples from the future to the past, it is perfectly reasonable in light of the fact that the DFT input can be assumed periodic with period N. Thus, for odd N, y can be represented as $y = [y((N - 1)/2 + 1), y((N - 1)/2 + 2), \ldots, y(N - 1), y(0), y(1), \ldots, y((N - 1)/2)]$.

This brings us to the question: What happens if the input is N samples of an aperiodic sound? A sound produced by a musical instrument is never truly periodic, although it is often quasi-periodic. The DFT simple assumes that its input is truly periodic, and the output samples of the DFT are then proportional to the "Fourier-series coefficients" of the periodic signal. (In this case, the spectrum is assumed to be zero between DFT samples.) This causes problems if the sequence is periodic with a period other than N, or is not periodic. In particular, if N does not equal a multiple of the period length, or if there is no period, then the DFT coefficients can no longer be interpreted as the amplitude and phase of the waveform harmonics. For example, suppose y has period $P = 86$ and we take a

Fig. 3. DFT of the waveform shown in Fig. 2.



DFT of size $N = 128$. When the sequence is viewed as an N-sequence, the first sample ($y(0)$) is assumed to follow the 128th sample ($y(128 \text{ MOD } 86) = y(42)$), forming a discontinuity in the resulting waveform (Fig. 2). The discontinuity shows up in the spectrum as a kind of broadband noise known as *crosstalk* or *spectral splatter*. Figure 3 shows the magnitude spectrum of this waveform. It has a peak near the original frequency, but the rest of the spectrum is not zero. Also notice that, since the original frequency is between two DFT points, the energy for that frequency is split between the two points. In practice, it is often difficult to determine the period length. Therefore, a technique known as *windowing* is used to smooth the ends of the sequence, reducing the spectral splatter.

Windowing

Windowing involves scaling the nth sample $y(n)$ of a waveform by the nth sample of an N-sequence $w(n)$, called a *window*. The windowed sequence is then $w(n)y(n)$, $n = 0, 1, \ldots, N-1$. The window usually begins and ends at or near zero and rises gradually to a peak between these points. It has the effect of smoothing the discontinuity that is produced by viewing the input as a periodic N-sequence. Windowing with an appropriately chosen window reduces the broadband energy caused by the discontinuity described previously. However, it has the disadvantage of spreading energy to neighboring points in the spectrum. These spreads are called *sidelobes* of each partial. For many applications, windowing is advantageous because such local spreading is considered less objectionable than broadband spreading.

To see why the sidelobes occur, windowing can be viewed as a linear digital filtering operation, where the filtering is done not in the time domain, but in the frequency domain. Multiplication in the time domain, such as is done in the process of windowing, is equivalent to filtering (or *convolution*) in the frequency domain. (This equivalence is explained in the section on "The Convolution Theorem.") Most useful windows are lowpass filters in the frequency domain. The local spreading of energy and the suppression of remote spreading can thus be seen as a smoothing caused by a lowpass filtering of the spectral samples.

There are two main classes of windows in common use. One class is designed to give maximal resolution of individual spectral components. That is, the worst-case sidelobe is minimized. An example of this type of window is the *Hamming window*. The other class is designed to minimize broadband spectral splatter. That is, the rolloff on either side of each spectral component is maximized. An example of this type of window is the *Hanning window*. The Hamming window has a rolloff of −6 db per octave, whereas the Hanning window has a rolloff of −12 db per octave. Nevertheless, the first sidelobe of the Hamming window is more attenuated than that of the Hanning window. Therefore, the Hamming window is considered preferable for musical applications. The Hamming window is defined (for odd N) as:

$$w_H(n) = .54 + .46 \cos\left(\frac{2\pi n}{N-1}\right), \quad |n| \leq \frac{N-1}{2},$$

where n is interpreted in terms of the convention of placing the time origin in the center. (Note that some authors use $(2\pi n/N)$ instead of $(2\pi n/(N-1))$.) Figure 4 illustrates a block of data multiplied by a Hamming window. Note how multiplication by the window smooths the discontinuity between the last and first samples in the original waveform. Figure 5 compares the spectrum of the cosine wave of Fig. 2, but windowed with a Hamming window, to the same data windowed with a Hanning window. (Note that, although these figures appear quite smooth, if we were to insert zeros before and after the window (i.e., *zero-pad*) and take a larger transform, we would see a *ripple* characteristic of win-

Fig. 4. An example of Hamming windowing: (a) original signal, (b) Hamming window, (c) windowed signal.

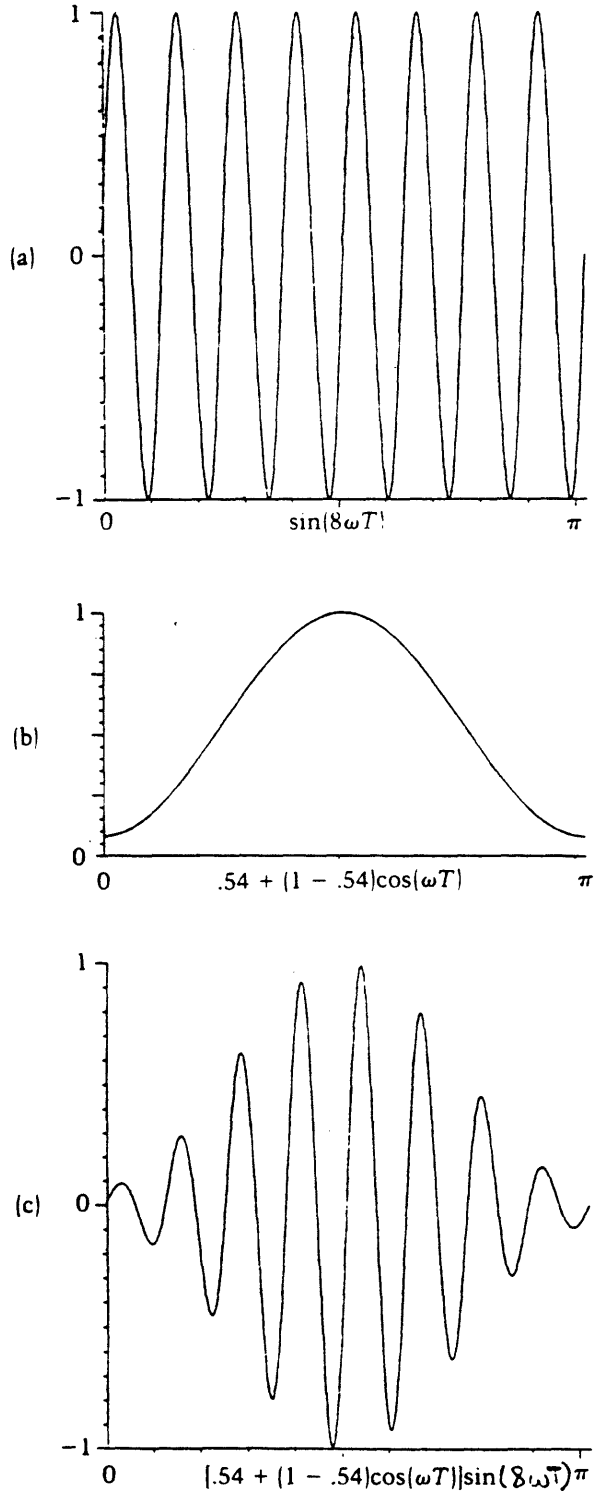
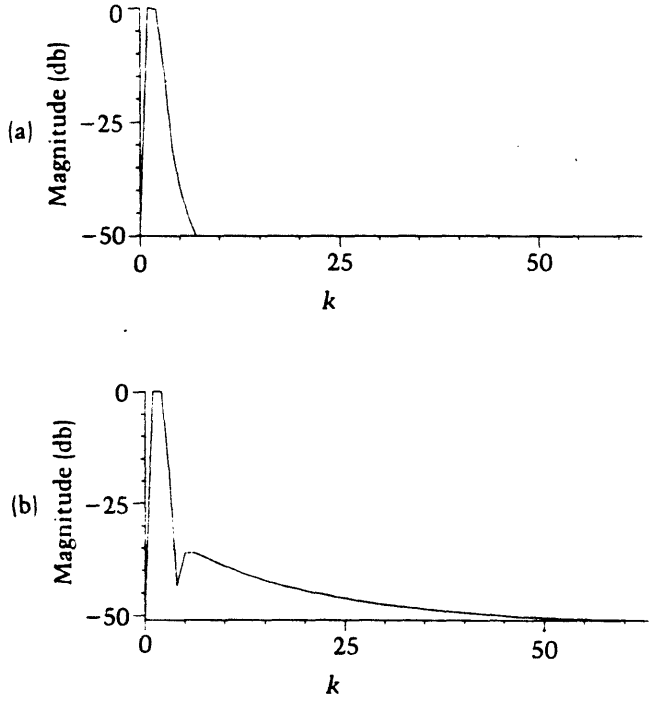


Fig. 5. Comparison of Hanning and Hamming windows using the waveform shown in Fig. 2: (a) DFT with a Hanning window, (b) DFT with a Hamming window.



dowing.) It is interesting that the Hanning window looks much better than the Hamming, except in the immediate vicinity of the significant spectral component.

For high-quality audio applications, a more sophisticated window such as the Blackman or Kaiser window is preferable, although a bit more expensive to implement. For more information on windows, see (Oppenheim and Schafer 1975) or (Rabiner and Gold 1975).

When windows are used, it is implicitly acknowledged that the DFT input is not periodic at the window length. In this case, the output samples of the DFT are interpreted as weightings for complicated functions, which do not concern us here.

Transform Pairs and the Conjugate Symmetry Theorem

In part one we explained that the DFT is simply an operation that converts from the time domain to the frequency domain, and the IDFT reverses this

process. This suggests the notion of a *transform pair.* A waveform y and a spectrum Y form a transform pair if $\mathrm{DFT}(y) = Y$, which implies $\mathrm{IDFT}(Y) = y$. The transform pair relationship is notated as $y \leftrightarrow Y$. In this section we examine several examples of transform pairs and conclude with a theorem about the DFT of real (that is, not complex) waveforms.

Consider the transform pair $y \leftrightarrow Y$ consisting of the unit impulse waveform $y = [1, 0, 0, 0, \ldots, 0]$ and the constant spectrum $Y = [1, 1, 1, 1, \ldots, 1]$. In terms of sound, this says that an impulse signal, a "click," transforms into a spectrum with equal energy everywhere. (In this example, we assume that the waveform N-sequence is a finite-duration waveform preceded and followed by zeros, and that the spectrum N-sequence is periodic.) Since a click sounds like a broadband signal, this transform pair corresponds well with our intuitive expectation. It is easy to see that $y \rightarrow Y$. The single impulse picks out only the sample of the sinusoid $e^{-j\omega_k nT}$ for which n equals zero. But this is equal to 1 for all ω_k (recalling that $e^0 = \cos(0) + j\sin(0) = 1$). That is,

$$\mathrm{DFT}_k(y) \triangleq Y(k) \triangleq \sum_{n=0}^{N-1} [1, 0, 0, \ldots, 0]_n e^{-j\omega_k nT}$$

$$= e^{-j\omega_k nT}\big|_{n=0} = 1 \text{ for all } \omega_k,$$

where $[1, 0, 0, \ldots, 0]_n$ means "the nth sample of the sequence $[1, 0, 0, \ldots, 0]$." Since the spectrum is real, the magnitude of each component is simply the absolute value of that component. The resulting magnitude vector is $Y(k) = 1$ for all k. We have shown that the impulse waveform transforms into a spectrum with equal energy everywhere. A similar process shows $y \leftarrow Y$. The spectrum Y is in this case a constant value of 1, so the IDFT is simply the sum over k of $e^{j\omega_k nT}$. For $n = 0$, this sums to N, but the factor of $1/N$ makes the result equal 1. For $n \neq 0$ the sum is zero because a sine wave summed over an integral number of periods always equals zero. If this seems unlikely, think of the sine wave in terms of the $\cos(\omega_k nT) + j\sin(\omega_k nT)$ representation. Both the cosine and sine functions sum to zero. (For a further explanation of this phenomenon, see Moorer 1978.) The IDFT of the constant spectrum is, therefore, a single impulse followed by zeros:

$$\mathrm{IDFT}_n(Y) \triangleq \frac{1}{N} \sum_{k=0}^{N-1} 1 e^{j\omega_k nT} = \begin{cases} 1, & n = 0 \\ 0, & n \neq 0. \end{cases}$$

As a second example, let us find the DFT of $y(n$ 1, a waveform that is a constant for all n. Since there is no variation in the waveform, our intuit tells us the spectrum can have energy only at 0 or DC. (Here we assume that the N-sequence is peated infinitely.) Using the same reasoning as b fore, the DFT is:

$$\mathrm{DFT}_k(y) \triangleq Y(k) \triangleq \sum_{n=0}^{N-1} 1 e^{-j\omega_k nT} = \begin{cases} N, & k = (\\ 0, & k \neq (\end{cases}$$

The IDFT retrieves the original waveform:

$$\mathrm{IDFT}_n(Y) \triangleq y(n) \triangleq \frac{1}{N} \sum_{k=0}^{N-1} [N, 0, 0, \ldots, 0]_k e^{j}$$

$$= \frac{N}{N} j^{0nT} = 1 \text{ for all } n.$$

Thus, a constant waveform corresponding to DC transforms into a spectrum with energy only at k (0 Hz), confirming our intuitive expectation.

These two examples illustrate an interesting sy metry (ignoring the factor of $1/N$ which is a resu of not using an orthonormal basis, as explained part one). In the first example, an impulse waveform with amplitude 1 transforms into a spectru equal to 1 for all k. In the second example, an ir pulse spectrum with amplitude N transforms in waveform equal to 1 for all n. Such symmetry is the rule, however, but is a special case that arise only when both the spectrum and the waveform real (and "even"—see later). A case such as this possible because the DFT and IDFT, although si lar, differ in the sign of the imaginary part. That they exhibit *conjugate symmetry* with $\mathrm{DFT}_m(\bar{Y}$ $N \cdot \overline{\mathrm{IDFT}}_m(Y)$. (We use the index m rather than n k purposely to blur the distinction between frequency and time domain.) Conjugate symmetry, while not a pure symmetry, is nevertheless pow ful enough to cause many theorems to have pai cases that differ only by a factor of N or by the of the power to which e is raised. For example, *shift theorem* (explained later) appears in two for

$$e^{j\omega_l nT} y \leftrightarrow \mathrm{Shift}_l(Y)$$

and

$$\text{Shift}_l(y) \leftrightarrow e^{-j\omega_l kT}Y.$$

(In this example, we follow the literature in being a bit sloppy about when to include and when not to include a sample counter. That is, we really should represent $e^{j\omega_l nT}y$ as either $e^{j\omega_l T}y$ or $e^{j\omega_l nT}y(n)$. However, the above convention is customary, and we hope it does not cause any confusion.) It turns out that a real spectrum, as in the previous examples occurs whenever the waveform is *even*. An even N-sequence y is one for which $y(n) = y(N - n)$. In other words, the graph of the sequence is symmetrical about the y-axis (assuming the sequence is real and using the convention that the origin is the middle of the N-sequence). A simple example of an even sequence is the cosine wave. For a real even waveform, there is no imaginary part in the spectrum and the factor of $1/N$ is the only difference between the DFT and IDFT.

Most real waveforms are not even, however, and their spectra are therefore complex and may even be purely imaginary. As an example, consider the waveform $y = [0, 1, 0, -1]$, corresponding to a sine wave at a frequency of $\omega_{N/4}$. Let us take its DFT. Recall that the DFT is simply an inner product of the input waveform y with a set of sinusoidal basis vectors $\{x_k\}$. That is,

$$\text{DFT}_k(y) \triangleq \sum_{n=0}^{N-1} y(n)e^{-j\omega_k nT} = \langle y, x_k \rangle, \; x_k(n) \triangleq e^{j\omega_k nT}.$$

Let us expand out the vectors $\{x_k\}$ to produce the sequence of inner products:

$$[\langle y, [1, 1, 1, 1]\rangle, \langle y, [1, j, -1, -j]\rangle, \langle y, [1, -1, 1, -1]\rangle, \langle y, [1, -j, -1, j]\rangle].$$

Finally, evaluating each inner product gives the spectral samples

$$[(0 + 1 + 0 - 1), (0 - j + 0 - j), (0 - 1 + 0 + 1), (0 + j + 0 + j)] = [0, -2j, 0, 2j].$$

The spectrum is, therefore, purely imaginary. This transform pair is an example of another special case. The waveform is *odd* and transforms into a purely imaginary spectrum. An odd N-sequence y is one for which $y(n) = -y(N - n)$; its graph is anti-symmetrical about the y-axis. The sine wave is a simple example of an odd sequence.

Finally, consider a sinusoidal waveform of frequency $\omega_{N/4}$, which has an initial phase of $\omega_{N/8}$. Such a sinusoid is "in between" a cosine and a sine with respect to phase. One would expect that it would be neither even nor odd, and would have a spectrum with both a real and an imaginary part. Such a waveform, for $N = 4$ is, approximately $[.7, .7, -.7, -.7]$. Its spectrum works out to be $[0, 1.4 + 1.4j, 0, 1.4 - 1.4j]$ and has both a real and an imaginary part, as we suspected. Most waveforms are neither even nor odd but can be broken down into a sum of an even and an odd part (as proved in Appendix A).

Notice that the spectrum in the previous example, while not perfectly symmetrical as the prior example was, does exhibit a kind of symmetry. Precisely, it is conjugate symmetric or *Hermitian*. That is, $Y(k) = \overline{Y(N - k)}$. Our first major theorem is the *conjugate symmetry theorem*. It states that any real waveform has a conjugate symmetric spectrum. (A proof is given in Appendix A.) The conjugate symmetry theorem implies that the magnitude spectrum of a real waveform is even whereas the phase of the spectrum is odd. That is, $|Y(k)| = |Y(N - k)|$ and $\angle Y(k) = -\angle Y(N - k)$.

Since digitized and synthesized sound waveforms are ordinarily real, the conjugate symmetry theorem ensures that fast Fourier transform (FFT) programs can ignore the frequencies between $-f_s/2$ and 0. This is because these frequencies have the same magnitude and opposite phase as those between 0 and $f_s/2$ (due to conjugate symmetry) and thus offer no new information.

We now proceed to discuss a number of other important Fourier theorems.

The Linearity Theorem

We begin with a basic but essential theorem, the *linearity theorem*. Linearity makes it possible to do *additive synthesis*, the construction of waveforms by adding sine waves of different frequencies and amplitudes. This theorem has two parts. The first part states: The DFT of a sum of waveforms is equal

to the sum of the DFTs of each waveform. The second part states: Scaling a waveform scales its DFT by the same amount. Both parts can be represented, mathematically, by the single assertion: $\alpha y_1 + \beta y_2 \leftrightarrow \alpha Y_1 + \beta Y_2$ for any complex constants α and β. The proof follows directly from the fact that the DFT is a special case of the inner product which we showed to be bilinear in part one.

Linearity is why we can have several sounds at once and still recognize them individually, and why we can recognize the same timbre at different volume levels.

The Shift Theorem

The shift theorem describes a phenomenon familiar to anyone who has used a *ring modulator*. A ring modulator multiplies a waveform y by a sine wave. The effect is to shift the frequency of the waveform up and down by a certain amount. The shift theorem describes this in mathematical terms as follows (a proof is given in Appendix A):

$$e^{i\omega_l nT} y(n) \leftrightarrow \text{Shift}_l(Y),$$

where the Shift_l of an N-sequence is simply a circular rotation of the sequence. The nth component of an N-sequence y shifted by l is written as $\text{Shift}_{(l,n)}(y)$ and is equal to $y(n - l)$. The entire sequence is:

$$\text{Shift}_{(l,n)}(y) \triangleq [y(N - l), y(n - l + 1), \ldots, y(0), \ldots, y(-l + (N - 1))]_n$$

(see Fig. 6). If y is a time waveform, a shift by l samples is similar to a *delay* of l samples, except that in the case of a shift, samples that are shifted off the end of the waveform ($n > (-l + (N - 1))$) "wrap around" to the beginning. To make a shift behave as a true delay, a sequence of length M can be extended or *padded* with M or more samples of zero value, producing a sequence of length $N \geq 2M$. Precisely, a *zero-padded* waveform, y, is defined as $y(m) = 0$, $(N/2) \leq m \leq (N - 1)$. We are then assured that a delay of $l \leq M$ will not bring any nonzero samples into the beginning of the waveform.

If we change domains, we find that a shift in the time domain corresponds to a multiplication by a sinusoid in the frequency domain:

$$\text{Shift}_l(y) \leftrightarrow e^{-i\omega_l kT} Y(k).$$

(The term $e^{-i\omega_l kT}$ is often represented as $e^{-i\omega_k lT}$.) A shift in the time domain corresponds to an *additive linear phase shift* by the variable l. The higher the frequency of the component, the more severe the phase shift with respect to the period. For example, for an $N = 8$-point DFT, the shift theorem gives the shift factor of a waveform delayed by $l = 1$ sample as $e^{-i\omega_l kT} = e^{-i\pi lk/8} = e^{-ik\pi/4}$. Recalling that a multiplication by e^{ix} corresponds to a phase shift of x, the shift of the component $k = 1$ is thus $-\pi/4$. But for the component $k = 2$, the exponent of e is doubled and the phase shift is $-\pi/2$. In summary, if the input to the DFT is a delayed waveform with all harmonics in phase, the angle of the DFT, $\angle \text{DFT}_k(\text{Shift}_l(y))$, is an increasing linear function with a slope proportional to l. Note that if Y is a conjugate-symmetric spectrum corresponding to a real waveform y, the shifting process in the frequency domain can destroy the conjugate symmetry, introducing an imaginary component to the waveform.

The functioning of the phase vocoder (Dolson 1986) can be thought of as performing a shift in the frequency domain. *Heterodyning*, or multiplying a waveform by a complex sinusoid $e^{-i\omega_l kT}$, for $k = 0, 1, 2, \ldots, N - 1$, moves any partial near l to a point

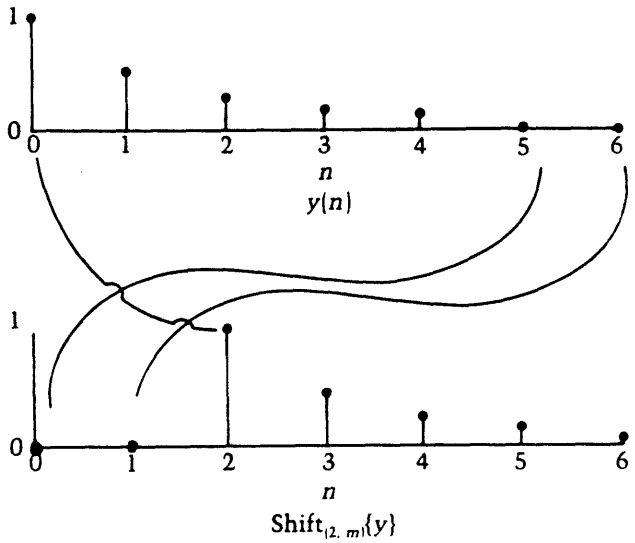
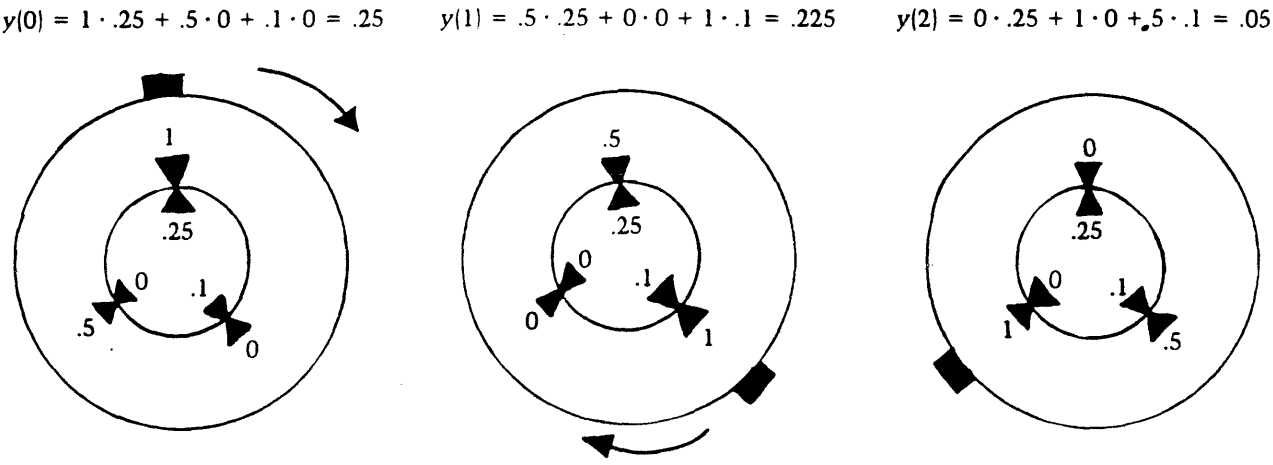


Fig. 6. Representation of the shift operation.

Fig. 7. Convolution of [1, .5, 0] and [.25, .1, 0] equals [.25, .225, .05].



Convolution of [1, .5, 0] and [.25, .1, 0] = [.25, .225, .05]

near 0 Hz. This partial is then "picked out" by a window functioning as a lowpass filter.

Another use of the shift theorem occurs in the proof of one of the most important theorems in spectrum analysis, the *convolution theorem*.

The Convolution Theorem

The convolution theorem forms the basis for much of digital filter theory. It explains the relationship between multiplication in one domain and *convolution* in the other. The operation known as circular convolution is denoted $x * y$, and is defined as

$$\mathrm{Conv}_n(x, y) = x * y(n) \triangleq \sum_{m=0}^{N-1} x(m)y(n - m).$$

Convolution is commutative. That is,

$$x * y(n) \triangleq \sum_{m=0}^{N-1} x(m)y(n - m) = \sum_{m=0}^{N-1} y(m)x(n - m) = y * x(n).$$

Convolution assumes that its inputs are N-sequence and it produces an N-sequence as output. In order to visualize convolution, think of two disks, one inside the other. Figure 7 illustrates this viewpoint. The two sequences to be convolved, [1, .5, 0] and [.25, .1, 0], are drawn around the circumference of the disks. To obtain each output sample, multiply the samples that line up and add the products. Then rotate the outer disk one sample and repeat the process. The result is the sequence [.25, .225, .05]. Note that, while the input sequences each have one zero sample, the output sequence has none. This is because convolution has the effect of "spreading" the waveform. If the input sequences do not have enough zeros at the end, this spreading has the effect of causing a "wrap around." Thus convolution is a *periodic* or *circular* operation. We have already discussed another circular operation, the shift. Recall that in order to yield a delay when applying the shift operator to a finite-duration sequence, the sequence should be zero-padded. Similarly, for convolution, it is often advantageous to arrange for the inputs to be zero-padded.

Convolution is what happens when an N-sequence is filtered with a finite impulse response (FIR) filter. For example, a simple lowpass FIR filter is given by $y(n) = (.4x(n) + .6x(n - 1))$, where $y(n)$ is the output and $x(n)$ is the input at time n. Consider what happens if $x(n)$ is an impulse. For $n = 0$, the impulse picks out the first term of the filter and the filter's output is .4. For $n = 1$, the first term contributes nothing, since the new input sample is 0. But the second term is multiplied by the original impulse and the filter's output is .6. On the third

sample, both terms contribute nothing and the output of the filter is 0. The sequence $[.4, .6, 0, \ldots]_n$ is thus the output of the filter. Since the input is an impulse, this is called the *impulse response* of the filter. Note that it is identical to the convolution of the impulse sequence $[1, 0, 0, \ldots]_n$ with the sequence $[.4, .6, 0, 0, 0, \ldots]_n$, formed by the coefficients of the two terms. It is also identical to the coefficient vector itself. Thus, the coefficients of an FIR filter can be obtained by exciting the filter with an impulse and reading off the output samples. If the input impulse is α instead of 1, the output is scaled by α, since we are considering only linear digital filters. A sequence can be thought of as a set of scaled, delayed impulses. For example, the sequence $[.5, .1, .2, 0, 0, 0, \ldots]_n$ is the sum of three impulse sequences, one scaled by .5, another delayed by one sample and scaled by .1 and, the third delayed by two samples and scaled by .2. An examination of the definition of convolution shows that linear time-invariant filtering is simply the convolution of the input with the impulse response of the filter. (See Smith 1981 for further information on convolution and digital filtering.)

Convolution is a computationally intensive process. Its compute time grows proportional to N^2. It is therefore impractical for most applications. But, this is not the case for the FFT implementation of the DFT. Here the compute time grows proportional to $N \log_2(N)$. Luckily, the convolution theorem gives a way to express convolution as a product of FFTs.

The convolution theorem states that the convolution of two waveforms x and y is the IDFT of the product of the corresponding spectra X and Y. That is, $x * y \leftrightarrow X \cdot Y$. (A proof of the convolution theorem is given in Appendix A.) The convolution theorem suggests an implementation of the convolution of x and y as follows: First take the FFTs of x and y. Then multiply the resulting spectra. Then obtain the result by taking the inverse FFT of the product.

The use of the FFT-based method of convolution provides a drastic improvement in computation speed over the direct implementation. For example, to convolve two 1-sec waveforms at a sample rate of 30000 Hz, it is necessary to perform $30000^2 = 9 \times 10^8$ multiplies. To take the FFT of each waveform requires only approximately $N \log_2(N) = 4.2 \times 10^5$ multiplies. Thus the total number of multiplies to perform convolution using the algorithm IFFT(FFT(x)FFT(y)), where IFFT is the inverse FFT, is only 1.29×10^6, an improvement of nearly three orders of magnitude over direct convolution.

A related theorem states that the spectrum of the product of two waveforms is the convolution of their spectra, with a factor of $1/N$ thrown in to remind us that we are not using an orthonormal basis: $x \cdot y \leftrightarrow (1/N)X * Y$. (The proof is left to the reader.) This form of the convolution theorem is useful in understanding the effect of a window function on a spectrum. The spectrum of the window is convolved with that of the original data.

The Stretch Theorem

The *stretch theorem* comes into play when sampling rate conversion from a lower to a higher sampling-rate is performed. If the ratio between the two rates is an integer, the conversion is ordinarily performed by inserting some number of zeros between adjacent samples of the original waveform and then lowpass filtering. The operation of inserting zeros is called the *stretch* of a waveform and produces the waveform at the desired sampling rate. It also produces artifacts in the frequency domain that can be removed with an appropriate filter.

The reader may be surprised that inserting zeros is the preferred method for upsampling by an integer multiple. Inserting samples interpolated in some other way may make more sense intuitively. However, if linear interpolation is used, significant distortion of the original waveform results. Quadratic interpolation also introduces some distortion. It is *bandlimited interpolation*, described in this section, which introduces the minimum amount of distortion. (The distortion depends on the filter chosen.)

We proceed in our quest toward understanding the effect of stretching by introducing a precise definition of stretching, followed by a definition of another important operation, the *repeat* of an N-sequence. The Stretch_L of an N-sequence (Fig. 8a) is a sequence of length $M = NL$ obtained by inserting $L - 1$ zeros

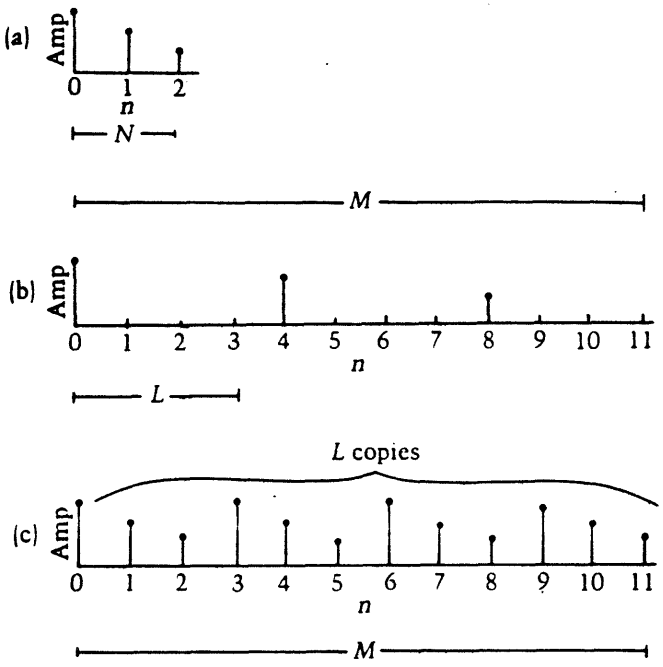


Fig. 8. Representations of signal processing operations. (a) original signal x(n), n = 0, 1, 2 *and Length* N = 3, *(b)* $Stretch_L\{x\}$ *with* L = 4, *(c)* $Repeat_L\{x\}$ *with* L = 4.

after each sample of the original sequence, as shown in Fig. 8b. Let $l \triangleq (n \text{ DIV } L)$, where DIV is an integer divide with the remainder discarded. Then

$$\text{Stretch}_{(L,n)}(y) \triangleq \begin{cases} y(l), & (n \text{ MOD } L) = 0 \\ 0, & (n \text{ MOD } L) \neq 0. \end{cases}$$

The $Repeat_L$ of an N-sequence is a sequence of length $M = NL$ consisting of L copies of the original sequence as shown in Fig. 8c.

$$\text{Repeat}_{(L,n)}(y) \triangleq y(n \text{ MOD } N) =$$

$$[y(0), y(1), \ldots, y(N-1), y(0), \ldots, y(N-1),$$

$$\ldots, y(0), \ldots, y(N-1)]_n, \; n = 0, 1, \ldots, LN-1.$$

The stretch theorem states that the DFT of the stretch of a waveform is a repeated version of the spectrum of the original waveform. That is, $\text{Stretch}_L(y) \leftrightarrow \text{Repeat}_L(Y)$. (A proof is given in Appendix A.) In terms of our upsampling example, the process of inserting zeros between the samples of the original waveform causes *spectral copies* to be created. These spectral copies are in most cases inharmonic and, in any case, are not part of the original waveform. They must therefore be filtered out using a lowpass filter whose cutoff is at or near the original Nyquist frequency ($f_s/2$, where f_s is the original sampling rate). It is this process, stretching followed by filtering at the original sampling rate over two, which constitutes bandlimited interpolation.

Digital-to-analog conversion can be thought of as upsampling to a sampling rate of infinity. A DAC performs two operations. First it converts a sequence of samples to a sequence of voltages. This is similar to inserting an infinite number of zeros between adjacent samples and convolving with a one-sample-wide rectangular pulse. Then it filters the result at the original Nyquist limit, producing a bandlimited analog signal.

Similar to the stretch theorem is the *repeat theorem*. This states that taking the DFT of $\text{Repeat}_L(y)$ has the effect of inserting $L - 1$ zeros between each point in Y, the spectrum of y. This theorem helps motivate one of the two interpretations of the input to the DFT. Recall that one interpretation assumes that the DFT's input is periodic. This is the same as $\text{Repeat}_L(y)$, where L is infinitely large. It corresponds to a DFT spectrum having an infinite number of zeros between the sample points.

The alternative view of the input to the DFT is that it is a waveform preceded and followed by an infinite number of zeros. This is the same as saying that an infinitely long waveform is windowed with a rectangular window that sets all samples outside the window to zero, while leaving all samples inside the window alone. Using the convolution theorem, we know that windowing, which is a multiplication in the time domain, corresponds to a convolution in the frequency domain. Thus, the spectrum, using this interpretation, consists of the convolution of the transform of the window with the transform of the original waveform. The transform of the rectangular window turns out to be a function of the form $\sin(x)/x$, called a *sinc* function (see Fig. 9).

We next consider the case of dividing the sampling rate by an integer divisor, M.

Fig. 9. The sinc *function, sin(x)/x.*

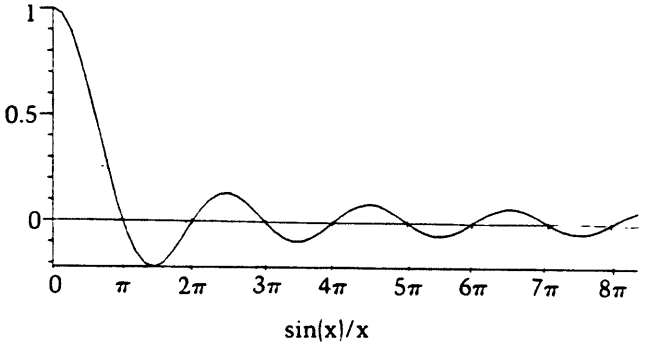
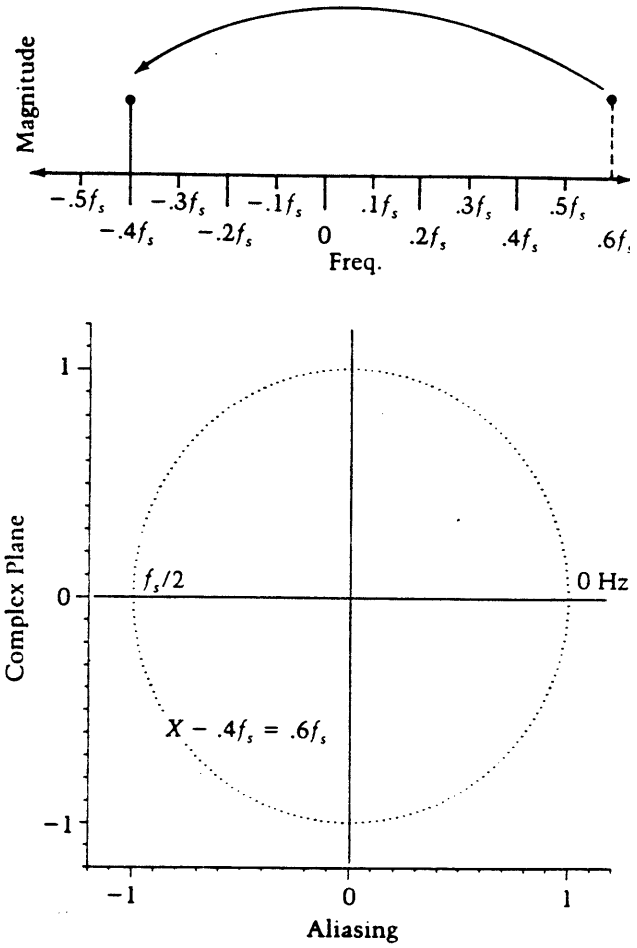


Fig. 10. The effect of aliasing, shown in the linear spectrum and on the complex plane.



The Decimation Theorem

Downsampling is analogous to upsampling. Instead of inserting $L - 1$ zeros between adjacent samples, all samples whose position is not a multiple of M are removed. This is called the *decimation* of a waveform. The decimation Dec_M of an N-sequence is of length $L = N/M$. It is obtained by taking every Mth sample of the original sequence, beginning with the first sample.

$$\text{Dec}_{(M,\,n)}(y) \triangleq y(nM)$$

where $n = 0, 1, \ldots, L$. As with stretching, decimation has ramifications in the frequency domain which must be accounted for in order to avoid unwanted artifacts.

We will show that decimating a waveform causes an *aliasing* of its spectrum. The Alias_M of an N-sequence is a sequence of length $L = N/M$, where M divides N evenly, obtained by partitioning the sequence into M pieces of length L, and then adding up the pieces. Then

$$\text{Alias}_{(M,\,n)}(Y) \triangleq \sum_{m=0}^{M-1} Y(n + mL)$$

where $n = 0, 1, \ldots, L - 1$. For example, a partial present at $.6f_s$ will be aliased down to $-.4f_s$ (see Fig. 10). Notice that Repeat_M lengthens a sequence by a factor of M while Alias_M shortens it by a factor of $1/M$.

The *decimation theorem* states that the DFT of a waveform decimated by M is the Alias_M of the spectrum of the original waveform scaled by a factor of $1/M$. That is, $\text{Dec}_M(y) \leftrightarrow (1/M)\text{Alias}_M(Y)$. (A proof is given in Appendix A.) Aliasing is almost always an undesired effect because, in general, it cannot be undone. The remedy is to lowpass filter the waveform, before performing the decimation, with a filter whose cutoff is at the new (lower) Nyquist rate. This ensures that the aliased frequencies will have no significant energy and thus will cause no audible distortion. Notice that the filtering operation is done before the decimation, whereas in the case of upsampling, it is done after the stretching.

The decimation theorem helps us understand analog-to-digital conversion (ADC) as analogous to a downsampling from a sampling rate of infinity. An ADC performs two operations. First it filters at

Fig. 11. The flip of the sequence [1, 0.5, 0.25, 0.125], which is [1, 0.125, 0.25, 0.5].

half the new sampling rate to remove any spectral components that would otherwise alias. The filter is appropriately called an *anti-aliasing filter.* Then it does a sample-and-hold process and generates a sequence of samples from the sequence of voltages. This can be thought of as a decimation of the original signal.

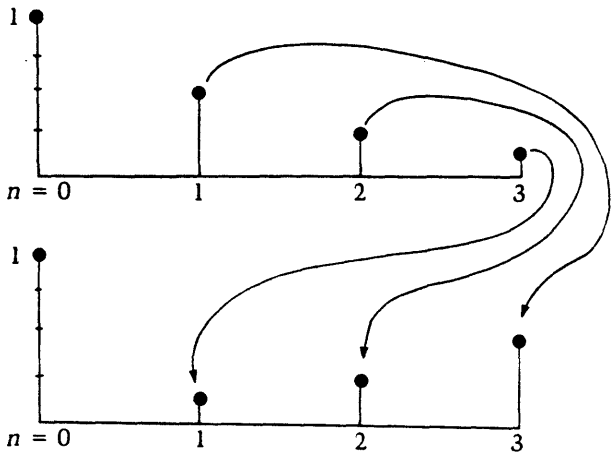
Up to this point we have considered resampling only by integer conversion factors. It is easy to implement arbitrary resampling by combining an upsampling operation followed by a downsampling operation. For example, to get a sampling rate change of .75, the sampling-rate conversion is accomplished by upsampling by a factor of 3 followed by downsampling by a factor of 4.

A related theorem states that decimation in the frequency domain causes aliasing in the time domain. This phenomenon, called *time-aliasing,* is usually undesirable. It can be avoided by lowpass filtering the spectrum before performing the decimation. It can also be avoided by first zero-padding in the time domain before performing time-spreading operations such as convolution.

The Flip Theorem

We now move to what may seem like a rather odd operation, the *flip* of an N-sequence. The flip is the complex conjugate of the reverse of the sequence, with the first component held in its original position. Precisely, the flip of y is $[y(0), y(N-1), y(N-2), \ldots, y(1)]$. By the assumed periodicity of the N-sequence y, $y(N+n) = y(n)$ so $\text{Flip}_n(y) \triangleq y(N-n) = y(-n)$. Figure 11 shows the flip of the sequence [1, .5, .25, .125], which is [1, .125, .25, .5].

Flip is used to eliminate phase distortion in digital filters. The convolution theorem tells us that a convolution (filtering) in the time domain is a multiplication in the frequency domain. Filtering a signal is the convolution of that signal with the impulse response of the filter, and corresponds, in the frequency domain, to the multiplication of the DFT of the input signal by the DFT of the impulse response. (The DFT of the impulse response is called the *frequency response* of the filter.) The filtering operation produces no phase distortion only if the



The sequence $[1, .5, .25, .125]_n$ (above) and its flip $[1, .125, .25, .5]_n$ (below).

filter's frequency response has a phase of zero for all frequencies. This is the same as saying the complex phase term is $e^0 = 1$ and is only true if the spectrum is real and nonnegative. Recall from the section "Transform Pairs and the Conjugate Symmetry Theorem" that a real spectrum results from an even waveform. Therefore, in order to eliminate phase distortion, we must design a filter whose impulse response is even. The flip comes in handy in this context.

In order to design a filter whose impulse response is even, we first design the filter with the frequency response we desire, using any method. Then we flip this impulse response and convolve it with the original impulse response, producing a filter with an impulse response whose length is approximately twice that of the original. Since this is a symmetrical real impulse response, its DFT is real. Thus the filter can introduce no phase distortion, except by multiples of π. The resulting filter has the square of the desired magnitude frequency response. This double filter can be thought of as two single-length filters in series. In order to see how the magnitude frequency response must be adjusted, we need a theorem that tells us the spectral effect of flipping the samples of a waveform.

The *flip theorem* states that the DFT of the flip of y is the complex conjugate of the spectrum of y.

That is, $\text{Flip}_n(y) \leftrightarrow \bar{Y}$. (A proof is given in Appendix A.)

In our example, the effect of flipping the impulse response is to conjugate its spectrum. But conjugating the spectrum has no effect on the magnitude of the spectrum. The effect of doubling the filter is the same as having two filters in series. In other words, it is the same as filtering the filtered signal. But filtering is convolution. Let y_1 be the original impulse response, y_2 be the flipped impulse response, and x be the input to the filter. Then the effect of filtering by both filters is to convolve x with y_1, and then convolve the result with y_2. But since convolution is commutative, this is the same as first convolving y_1 with y_2 and then convolving with x. The effect of convolving y_1 with y_2 is, by the convolution theorem, to multiply the magnitude frequency response by itself. Therefore, in order to come out with the desired magnitude frequency response and zero-phase, we must first design a filter with the square root of the desired magnitude frequency response, and then convolve with another filter whose frequency response is the conjugate of that of the first filter. The combined filter is zero-phase and has the desired frequency response.

In summary, an even waveform can be produced by convolving any waveform y with its flip. This, scaled by $1/N$ is called the *autocorrelation* of y. (The autocorrelation function used by statisticians has a different definition, but is related in spirit.) That is,

$$\text{Autocorrelation}_n(y) \triangleq 1/N \sum_{m=0}^{N-1} y(n+m)\overline{y(m)}$$

$$= \langle \text{Shift}_{-n}(y),\ y \rangle$$

Its spectrum differs from the spectrum of y in that the phase of each partial is zero, because the autocorrelation is an even function. In addition, by the convolution theorem, the magnitude has been squared. Finally, the factor of $1/N$ scales the spectrum.

The Correlation Theorem

The autocorrelation is important in statistical spectrum analysis (the study of statistically defined or *stochastic* waveforms), a subject that is beyond the scope of this tutorial. We give only a brief introduction here to familiarize the reader with the concepts of autocorrelation, power spectral density, and correlation.

Consider taking the DFT of white noise. We commonly think of white noise as having a flat "spectrum," and one might expect the magnitude spectrum of white noise to be a constant at all frequencies. It turns out that what we are actually imagining is not the spectrum but what is called the *power spectral density*, a measure of the statistical variance per unit frequency interval. Taking a DFT of white noise produces a random magnitude spectrum, no matter how large a time window is taken. The larger window introduces more frequency points but each still has random magnitude and phase. If, on the other hand, we take a series of DFTs, moving along in time one window-size for each DFT, and then average the successive windows, the spectrum approaches zero as the number of windows goes to infinity. This happens because the phases are random and, therefore, the set of spectral coefficients for each value of k averages to zero. In statistical terms, each spectral coefficient i a random variable whose standard deviation over successive windows can be greater than its mean. Thus the power spectral density is not estimated very well by using DFTs alone. On the other hand, it is possible to obtain a good power spectral density estimate by averaging the *squared magnitude* of DFTs of successive windows.

The expected flat spectrum can, equivalently, be obtained by taking the DFT of the autocorrelation of the waveform. By setting all phases to zero, the autocorrelation function has the effect of producing a deterministic waveform from a stochastic one. In this manner, it can be thought of as an information-destroying process. (But phase carries no information in a stochastic process, so this is not a problem. It does mean, however, that there is no "de-autocorrelation" function. Once an autocorrelation has been performed, it can not be undone.) The autocorrelation maps all possible phase combinations onto a single, zero-phase, representation. In the case of white noise, the resulting autocorrelation waveform, as N goes to infinity is an im-

pulse of height equal to 1 because only at $n = 0$ do all sinusoids of zero-phase "line-up." In the limit, as N goes to infinity, this value swamps all other values and the function goes to an impulse at $n = 0$. We showed previously that taking the DFT of the impulse sequence produces a flat spectrum. Thus, as N goes to infinity, the DFT of the autocorrelation of white noise approaches an estimate of the flat power spectrum we expect.

Autocorrelation is a special case of the *cross-correlation* or *correlation* of two sequences. Correlation is defined as:

$$\text{Corr}_n(x, y) = \sum_{m=0}^{N-1} x(n + m)\overline{y(m)} = \langle \text{Shift}_{-n}(x), y \rangle$$

where the angle brackets $\langle\ \rangle$ denote the *inner product* (see part one of this tutorial). Note that the autocorrelation is the correlation of a waveform with itself. This equation for correlation differs from convolution by a mere flip:

$$\begin{aligned}\text{Conv}_n(x, y) &= \sum_{m=0}^{N-1} x(m)y(n - m) \\ &= \sum_{k=-n}^{N-1-n} x(n + k)y(-k) \\ &= \sum_{m=0}^{N-1} x(n + m)y(-m) \\ &= \langle \text{Shift}_{-n}(x), \text{Flip}(y) \rangle = \text{Corr}_n(x, \text{Flip}(y)).\end{aligned}$$

The correlation theorem is similar to the convolution theorem. It states that the DFT of the correlation of a sequence x with a sequence y is equal to the conjugate of the DFT of x multiplied by the DFT of y. That is, $\text{Corr}(x, y) \leftrightarrow \bar{X}Y$. (A proof is given in Appendix A.) The correlation theorem helps us prove the energy theorem.

The Energy Theorem

The energy theorem makes it possible to construct devices such as graphic equalizers by ensuring conservation of energy over the DFT operation. The energy theorem states that the inner product of two waveforms of length N is equal to the inner product of the transforms of the two waveforms scaled by $1/N$, and defines the *cross-energy* of the transform pair as that value. That is, $\text{Energy}(x, y) \triangleq \langle x, y \rangle = 1/N \langle X, Y \rangle$. The *average power* is defined as the cross-energy divided by N. Let us return to the vector notation used in part one of this tutorial. Taking the DFT of two N-dimensional vectors $\vec{x}$ and $\vec{y}$ is equivalent to finding two corresponding vectors $\vec{x}'$ and $\vec{y}'$ in a different coordinate system. This operation does not change the relative angle in space of one vector with respect to the other. Thus, the inner product of the two vectors will be the same if the coordinate system is merely stretched and rotated, except for a constant factor D. D is defined as the square of the ratio between the norm of a vector of the original coordinate system and the norm of that vector in the new coordinate system. In the case of the DFT, the norm of the spectrum over the norm of the time-domain waveform is $\sqrt{N}$. Thus a factor $1/N$ must be included in the energy theorem. (A proof of the energy theorem is given in Appendix A.) Note that the $\text{Energy}(x, x)$ or *total energy* of a waveform x is its norm squared. Therefore, $\|x\|^2 = (1/N)\|X\|^2$. This is called *Parseval's theorem* or the *Rayleigh energy theorem*.

Generalization of the DFT: The Z-transform

The DFT can be generalized into a form known as the *Z-transform* that is particularly useful in digital filter theory. The Z-transform of a filter is a function of a single complex variable z. The Z-transform helps in locating the filter's *poles* (values of z at which the filter's magnitude frequency response is infinite) and *zeros* (values of z at which the filter's magnitude frequency response is zero). The Z-transform of an FIR filter can be obtained by taking the samples of the filter's impulse response as coefficients for increasing powers of z^{-1}. For example, the impulse response [1, 1, .25, 0, . . .] has the Z-transform of $1 + z^{-1} + .25z^{-2}$. This polynomial can be factored into $(1 + .5z^{-1})(1 + .5z^{-1})$. The zeros of this polynomial are determined by setting each term to zero and solving for z. In this case there are two zeros at $z = -.5$. In this example, there are no poles because this is

an FIR filter and poles only occur for infinite impulse response (IIR) filters. For IIR filters, the Z-transform is obtained from the difference equation of the filter. See (Smith 1981) for details.

In order to show that the Z-transform is a generalization of the DFT, recall that the DFT is simply an inner product $\langle y, x_k \rangle$ where $x_k(n) \triangleq e^{j\omega_k nT} \triangleq z_k^n$, with $z_k = e^{j2\pi k/N}$. We can denote the DFT as a function of z_k:

$$Y(z_k) = \langle y, z_k \rangle = \sum_{n=0}^{N-1} y(n) z_k^{-n}.$$

This is the Z-transform $Y(z)$ evaluated at z_k. The DFT is the Z-transform evaluated on the unit circle $z_k = e^{j2\pi k/N}$. The Z-transform evaluated at an arbitrary point on the complex plane z, rather than at a set of N points $\{z_k\}$, is simply

$$Y(z) = \langle y, z \rangle = \sum_{n=0}^{N-1} y(n) z^{-n}.$$

How can we interpret the Z-transform of an arbitrary point on the z-plane? We know that raising any complex number to successive integer powers produces a spiral. Thus the Z-transform of a waveform evaluated at z can be thought of as a measure of the exponentially increasing (if $|z| > 1$), decreasing (if $|z| < 1$), or constant (if $|z| = 1$) complex sinusoidal energy at a frequency $\angle z$ radians per sample. A circle (used as the domain of the DFT) is a special case of a spiral (used as the domain of the Z-transform), one for which the radius remains constant.

The Fourier Transform

Finally, we extend the DFT from the discrete, or digital, to the continuous domain. We shall do this in two steps. First, we allow the number of time-domain samples to go to infinity. Then we allow the sampling rate to go to infinity. In order to do so, we need to introduce the *integral* of calculus as a replacement for the summation that we have been using for the DFT. The definite integral of the function $f(x)$ with x ranging from $-\pi$ to π is notated $\int_{-\pi}^{\pi} f(x)dx$, where dx is the *differential* of the integration. An integral can be thought of as a summation over an infinite set of points comprising an interval. The integral adds the value of the function at every point in the interval and the differential multiplies this value by the distance between adjacent points in the interval. Since the points are infinitely close, the differential goes to 0 in the limit. The definite integral can be thought of as the area under the curve $f(x)$.

Recall that the spectrum produced by the DFT is a function over N points $\omega_k = 2\pi f_s k/N$ equally spaced around the unit circle. As N gets larger, these points get closer together. Let $\Delta\omega = 2\pi f_s/N$ equal the difference in frequency between successive points. The points can be made arbitrarily close by selecting an appropriate value of N. If N is allowed to go to infinity, the unit circle is divided into an infinite number of divisions, and $\Delta\omega$ goes to 0. In this case, the variable $\omega_k \triangleq 2\pi k f_s/N$ is replaced with a continuous variable ω. Let the waveform $y(n)$ be defined over all positive time. Then the DFT_ω of continuous ω and infinite discrete positive time becomes:

$$\text{DFT}_\omega(y) \triangleq Y(\omega) \triangleq \sum_{n=0}^{\infty} y(n) e^{-j\omega nT}.$$

This is a continuous function of an infinite sequence. Notice that the sampling rate has not been changed and the spectrum is still a periodic function of ω with period $2\pi f_s$. As N goes to infinity, the IDFT becomes an integral.

$$\text{IDFT}_n(Y) \triangleq y(n) \triangleq \frac{1}{2\pi} \int_{-\pi}^{\pi} Y(\omega) e^{j\omega nT} d\omega,$$

(Note that many texts include a factor of T in the DFT definition. That is, their DFT is our DFT multiplied by T. This makes the extension to the continuous domain a bit cleaner.)

Now consider the case where the sampling rate increases along with N. As the sampling rate goes to infinity, the period of the spectrum becomes infinite. That is, the spectrum ceases to be a periodic function and ω is redefined as a continuous variable on the interval $(-\infty, \infty)$.

What must we alter in the DFT equation to accommodate the infinite sampling rate? First, it no no longer makes sense to think of frequency as a fraction of the sampling rate. Second, as the sam-

pling rate goes to infinity, the sampling interval goes to 0 and it no longer makes sense to speak of the nth sample. The index variable n is replaced with a continuous variable t and the summation is replaced with an integration over t. Thus the basis set becomes $\{e^{i\omega t}\}$, which contains an infinite number of basis functions, one for each value of ω. The continuous Fourier transform of a function $y(t)$ is therefore defined as:

$$\mathrm{FT}_\omega(y) \triangleq Y(\omega) \triangleq \int_{-\infty}^{\infty} y(t)e^{-i\omega t}dt.$$

Similarly, the inverse Fourier transform is given by:

$$\mathrm{IFT}_t(Y) \triangleq y(t) \triangleq \frac{1}{2\pi}\int_{-\infty}^{\infty} Y(\omega)e^{i\omega t}d\omega.$$

Many of the theorems we have covered have analogs in the continuous domain. One major difference, however, is that in the continuous domain, frequency is no longer cyclical. There is, for example, no aliasing.

Conclusion

The spectrum analysis perspective provides insight into the relationship between the frequency and time domains. It also allows a deep understanding of such familiar operations as sampling-rate conversion, filtering, and modulation. One of the beauties of spectrum analysis is that it stems from one simple formula, the Fourier transform. (To pursue the study of spectrum analysis in more depth, we recommend Kesler 1986.)

Acknowledgments

I would like to thank Julius Smith for his inspiring and patient tutelage and Ken Shoemake, who first introduced me to the vector projection viewpoint. Thanks also to Andy Schloss and his computer music class at Brown University for putting the tutorial through a beta test. Finally, thanks to Bill Schottstaedt, Doug Keislar, and Xavier Serra for their helpful proofreading and suggestions.

References

Dolson, Mark. 1986. "The Phase Vocoder: A Tutorial." *Computer Music Journal* 10(4): 14–27.

Jaffe, D. 1987. "Spectrum Analysis Tutorial, Part 1: The Discrete Fourier Transform." *Computer Music Journal* 11(2): 9–24.

Kesler, S. B., ed. 1986. *Modern Spectrum Analysis II.* New York: IEEE Press.

Moorer, J. A. 1978. "The Use of the Phase Vocoder in Computer Music Applications." *Journal of the Audio Engineering Society* 26: 42–45.

Oppenheim, A. V., and R. W. Schafer. 1975. *Digital Signal Processing.* Englewood Cliffs, New Jersey: Prentice-Hall.

Rabiner, L. R., and B. Gold. 1975. *Theory and Application of Digital Signal Processing.* Englewood Cliffs, New Jersey: Prentice-Hall.

Smith, J. O. 1981. *"An Introduction to Digital Filter Theory."* Stanford: CCRMA. Reprinted in John Strawn, ed. 1985. *Digital Audio Signal Processing: An Anthology.* Los Altos: Kaufmann.

Appendix A: Proof of Theorems

The Conjugate Symmetry Theorem

The conjugate symmetry theorem states: if y is real, $Y = \mathrm{DFT}(y)$ is conjugate symmetric. In order to prove this, we must delve further into the notion of symmetry. Recall that a real sequence y is said to be *even* or *symmetric* if $y(n) = y(N - n)$. In contrast, a real sequence is said to be *odd* or *antisymmetric* if $y(n) = -y(N - n)$. Although many sequences are neither even nor odd, an arbitrary N-length sequence y can be expressed as a sum of a unique even and a unique odd part. In order to prove this, we guess a solution and then show that it works.

$$\text{Let } y_e(n) = \frac{y(n) + y(N - n)}{2} \text{ and}$$

$$\text{let } y_o(n) = \frac{y(n) - y(N - n)}{2}.$$

The sequence y_e is even because $y_e(N - n) = y_e(n)$. The sequence y_o is odd because $y_o(N - n) = -y_e(n)$. Since $y(n) = y_e(n) + y_o(n)$ and $y(n)$ was chosen arbitrarily, it has been shown that any sequence can be represented as a sum of an even and an odd sequence.

An example of an even sequence is $y(n) = \cos(2\pi kn/N)$. The evenness of this sequence is clear from looking at the exponential form of the cosine function and comparing it with the definition of y_e just given:

$$\cos(\theta) = \frac{e^{i\theta} + e^{-i\theta}}{2}.$$

Alternatively, one can recall the identity $\cos(-\theta) = \cos(\theta)$. Similarly, the sequence $y(n) = \sin(2\pi kn/N)$ can be shown to be odd, since $\sin(-\theta) = -\sin(\theta)$.

Armed with this insight, we are in a position to prove the conjugate symmetry theorem. We need only show that the real part of the spectrum is symmetrical (even) while the imaginary part is antisymmetrical (odd). That is, $\mathrm{Re}\{Y(n)\} = \mathrm{Re}\{Y(N-n)\}$ and $\mathrm{Im}\{Y(n)\} = -\mathrm{Im}\{Y(N-n)\}$. The real and imaginary parts of the DFT are obtained, for real y, by using Euler's identity (introduced in part one):

$$\mathrm{DFT}_k(y) = \sum_{n=0}^{N-1} y(n)\cos(\omega_k nT) + i\sum_{n=0}^{N-1} y(n)\sin(\omega_k nT)$$

It can be shown that a linear combination of odd sequences is an odd sequence and a linear combination of even sequences is an even sequence. (The proof of this is left to the reader.) Therefore, since the DFT of y is a sequence in k, and $y(n)$ serves simply as a real coefficient on the cosine and sine functions, $y(n)$ does not affect the symmetry of these functions. Similarly, since neither of the summations affect the symmetry,

$$\mathrm{Re}\{Y\} = \sum_{n=0}^{N-1} y(n)\cos(\omega_k nT) \quad \text{is even, and}$$

$$\mathrm{Im}\{Y\} = \sum_{n=0}^{N-1} y(n)\sin(\omega_k nT) \quad \text{is odd.}$$

The spectrum $Y(k)$ is, therefore, conjugate symmetric.

The Shift Theorem

The shift theorem states $e^{i\omega_l nT}y(n) \leftrightarrow \mathrm{Shift}_l(Y)$. To prove it, we use the variable substitution $m \triangleq k - l$,

$$\begin{aligned}\mathrm{IDFT}_n(\mathrm{Shift}_l(Y)) &= \sum_{k=0}^{N-1} Y(k-l)e^{i\omega_k nT} \\ &= \sum_{m=-l}^{N-1-l} Y(m)e^{i\omega_{(m+l)}nT} \\ &= e^{i\omega_l nT}\sum_{m=-l}^{N-1-l} Y(m)e^{i\omega_m nT} \\ &= e^{i\omega_l nT}y(n).\end{aligned}$$

The related theorem states $\mathrm{Shift}_l(y) \leftrightarrow e^{-i\omega_l kT}Y(k)$ and is proved in a similar manner.

The Convolution Theorem

The convolution theorem states $\mathrm{IDFT}_n(XY) = \mathrm{Conv}_n(x, y)$. Proof:

$$\begin{aligned}&\mathrm{IDFT}_n(XY) \\ &\quad = \mathrm{IDFT}_n\left(\sum_{m=0}^{N-1} x(m)e^{-i\omega_k mT}\sum_{l=0}^{N-1} y(l)e^{-i\omega_k lT}\right) \\ &\quad = \mathrm{IDFT}_n\left(\sum_{m=0}^{N-1} x(m)\left[e^{-i\omega_k mT}\sum_{l=0}^{N-1} y(l)e^{-i\omega_k lT}\right]\right)\end{aligned}$$

The key to simplifying this expression is to recognize that the expression in square brackets is the transform of y shifted by m. Using the shift theorem

$$\begin{aligned}&\mathrm{IDFT}_n\left(\sum_{m=0}^{N-1} x(m)\left[e^{-i\omega_k mT}\sum_{l=0}^{N-1} y(l)e^{-i\omega_k lT}\right]\right) \\ &\quad = \mathrm{IDFT}_n\left(\sum_{m=0}^{N-1} x(m)\sum_{l=0}^{N-1} y(l-m)e^{-i\omega_k lT}\right) \\ &\quad = \mathrm{IDFT}_n\left(\sum_{m=0}^{N-1}\sum_{l=0}^{N-1} x(m)y(l-m)e^{-i\omega_k lT}\right) \\ &\quad = \mathrm{IDFT}_n\left(\sum_{l=0}^{N-1}\left[\sum_{m=0}^{N-1} x(m)y(l-m)\right]e^{-i\omega_k l}\right. \\ &\quad = \mathrm{IDFT}_n\left(\sum_{l=0}^{N-1}(x*y)e^{-i\omega_k lT}\right) \\ &\quad = \mathrm{IDFT}_n(DFT(x*y)) \\ &\quad = x*y.\end{aligned}$$

The Stretch Theorem

The stretch theorem states: $\text{Stretch}_L(y) \leftrightarrow \text{Repeat}_L(Y)$. To prove it, let $y(n) = \text{Stretch}_{L,n}(x)$, where x is length M and y is length $N = LM$. Since $x(m)$ consists of all the nonzero points of $y(n)$, we can omit all the other points from the summation and sum over only M points.

$$Y(k) = \sum_{n=0}^{N-1} y(n)e^{-i\omega_k nT} = \sum_{m=0}^{M-1} x(m)e^{-i\omega_k mLT},$$

where $m \triangleq n/L$. But $\omega_k L = (2\pi k f_s/N)L = 2\pi k f_s/M$. Thus

$$\sum_{m=0}^{M-1} x(m)e^{-i\omega_k mLT} = \sum_{m=0}^{M-1} x(m)e^{-i2\pi k/M} = \text{DFT}_k(x)$$
$$= X(k),\ k = 0, 1, 2, \ldots, (M-1).$$

However, we started with $Y(k)$ where $k = 0, 1, 2, \ldots, (N-1)$. So $X(k) = Y(k)$ for $k = 0, 1, 2, \ldots, (M-1)$. What about the values of $X(k)$ for $k \geq M$? Since the spectrum $X(k)$ is a periodic function with a period of M, $Y(k)$ is equivalent to traveling around the unit circle L times, evaluating $X(k)$ at $k = 0, 1, 2, \ldots, (M-1), 0, 1, 2, \ldots, (M-1), \ldots, 0, 1, 2, \ldots, (M-1)$. Thus the spectrum $Y(k)$ is equal to repeat $_L(\text{DFT}(x))$.

The Decimation Theorem

The decimation theorem states: $\text{Dec}_M(y) \leftrightarrow (1/M)\text{Alias}_M(Y)$. To demonstrate this, let $X \triangleq \text{Alias}_M(Y)$, where X is length L, Y is length N, M is the decimation factor and $N = LM$. Then

$$X(l) \triangleq \sum_{k=0}^{M-1} Y(l + kL)$$
$$= \sum_{k=0}^{M-1}\sum_{n=0}^{N-1} y(n)e^{-i\omega_{(l+kL)}nT}$$
$$= \sum_{n=0}^{N-1} y(n) \sum_{k=0}^{M-1} e^{-i\omega_l nT}e^{-i\omega_{kL}nT}$$
$$= \sum_{n=0}^{N-1} y(n) \sum_{k=0}^{M-1} e^{-i2\pi ln/N}e^{-i2\pi kLn/N}$$
$$= \sum_{n=0}^{N-1} y(n)e^{-i2\pi ln/N} \sum_{k=0}^{M-1} e^{-i2\pi kn/M}.$$

We again invoke the theorem that a sinusoid summed over an integral number of periods is equal to zero.

$$\sum_{k=0}^{M-1} e^{-i2\pi kn/M} \begin{cases} M, & n = Mp,\ p = 0, 1, 2, \ldots, \infty \\ 0, & n \neq Mp. \end{cases}$$

Since only every Mth point is nonzero, let $i = nM$. Then

$$\sum_{n=0}^{N-1} y(n)e^{-i2\pi ln/N} \sum_{k=0}^{M-1} e^{-i2\pi kn/M} = M \sum_{i=0}^{L-1} y(i)e^{-i2\pi liM/N}$$
$$= M \sum_{i=0}^{L-1} y(i)e^{-i2\pi li/L}$$
$$= M \cdot \text{DFT}_l(\text{Dec}_M(y)).$$

The result of decimating a waveform is, therefore, to alias the spectrum.

The Flip Theorem

The flip theorem states $\text{Flip}_n(y) \leftrightarrow \bar{Y}$. Proof:

$$\text{DFT}_k(\text{Flip}(y)) = \sum_{n=0}^{N-1} \overline{y(N-n)}e^{-i\omega_k nT}.$$

Let $m \triangleq N - n$. Then

$$\text{DFT}_k(\text{Flip}(\bar{y})) = \sum_{m=N}^{1} \overline{y(m)}e^{-i\omega_k(N-m)T}$$
$$= \sum_{m=0}^{N-1} \overline{y(m)}e^{-i\omega_k(N-m)T}$$
$$= \overline{\sum_{m=0}^{N-1} y(m)e^{-i\omega_k(m-N)T}}$$
$$= \overline{\sum_{m=0}^{N-1} y(m)e^{-i\omega_k mT}}$$
$$= \overline{Y(k)}.$$

The Correlation Theorem

The correlation theorem states $\text{Corr}(x, y) \leftrightarrow \bar{X}Y$. To prove it, we invoke the convolution and flip theorems:

$$\text{DFT}_k(\text{Corr}(x, y)) = \text{DFT}_k(x * \text{Flip}(y)) = X(k)\overline{Y(k)}$$

The Energy Theorem

The energy theorem states Energy$(x, y) \triangleq \langle x, y \rangle = 1/N \langle X, Y \rangle$. It can be proven using the definition of correlation:

$$\langle x, y \rangle \triangleq \sum_{n=0}^{N-1} x(n)\overline{y(n)} = \mathrm{Corr}_0(x, y) = \mathrm{IDFT}_0(X\bar{Y})$$

$$= \frac{1}{N} \sum_{k=0}^{N-1} X(k)\overline{Y(k)}$$

$$= \frac{1}{N} \langle X, Y \rangle.$$

The factor of $1/N$ is, as usual, a result of our using a nonorthonormal basis.

Appendix B: Correction to Part One

The inner product of $(1 + j)$ and $(1 - j)$ was erroneously given as 2, rather than the correct value of $2j$.