XI. DIGITAL SIGNAL PROCESSING

Academic and Research Staff

Prof. Alan V. Oppenheim
Prof. Jonathan Allen
Dr. Tomonori Aoyama
Dr. Russell M. Mersereau

Graduate Students

Glenn R. Babecki
Ronald E. Crochiere
Dan E. Dudgeon
Harlan S. Hersey
Michael R. Portnoff
José M. Tribolet
Ernest Vincent
Robert J. Wenzel

A. USING A TWO-DIMENSIONAL DISCRETE HILBERT TRANSFORM TO FACTOR TWO-DIMENSIONAL POLYNOMIALS

U. S. Navy Office of Naval Research (Contract N00014-67-A-0204-0064)

Dan E. Dudgeon

In this report, a two-dimensional Hilbert transform is derived. This Hilbert transform can be used to construct the imaginary part of the Fourier transform from the real part of the Fourier transform for arrays of a particular form. An approximation to this Hilbert transform, termed a discrete Hilbert transform (DHT), is then derived. This DHT can be implemented as a computer program by using discrete Fourier transforms (DFTs).

This work was motivated by the need to split two-dimensional magnitude-squared frequency responses so that the resulting filters could be implemented. This is accomplished in one dimension by factoring the magnitude-squared polynomial and keeping suitable roots. Since the fundamental theorem of algebra does not apply in two-dimensional (or more) space, this factorization technique cannot be applied.

By applying the DHT to the log magnitude spectrum of a two-dimensional array, the analytic phase function is found. In this way, two-dimensional magnitude-squared functions may be factored approximately.

The idea for this DHT approach came from the work of Read and Treitel,[1] who used a two-dimensional DHT for stabilizing unstable filters. Although our derivation is slightly different, the underlying problems that we attack are essentially the same.

1. Two-Dimensional Hilbert Transform

The derivation presented here is a fairly straightforward extension of the one-dimensional Hilbert transform.[2-4] Let us define a one-sided array $b(m,n)$ such that $b(m,n) = 0$ for $m < 0$, and for $m = 0$, $n < 0$. The motivation for the term "one-sided" is that the nonzero points of $b(m,n)$ lie on or to one side of the line $m = 0$. Now define the even part of $b(m,n)$ as $b_e(m,n) = \frac{1}{2}[b(m,n) + b(-m,-n)]$, and the odd part of $b(m,n)$ as $b_o(m,n) = \frac{1}{2}[b(m,n) - b(-m,-n)]$. Notice that if $b(m,n)$ is a one-sided array, it is completely specified by its even part $b_e(m,n)$, and also completely specified (except for the point at the origin) by its odd part $b_o(m,n)$. In addition, the odd part can be calculated from the even part by use of a multiplicative window function. That is,

$$b_o(m,n) = b_e(m,n)\, w(m,n), \tag{1}$$

where

$$w(m,n) = \begin{cases} 0 \text{ for } m = n = 0 \\ +1 \text{ for } m > 0 \\ +1 \text{ for } m = 0,\ n > 0 \\ -1 \quad \text{elsewhere.} \end{cases}$$

We denote the Fourier transform of $b(m,n)$ as

$$B(\mu,\nu) = \sum_{m,n} b(m,n) \exp[-j\mu m - j\nu n].$$

It can be shown that the Fourier transform of $b_e(m,n)$ is the real part of $B(\mu,\nu)$, which we denote $B_r(\mu,\nu)$, and the Fourier transform of $b_o(m,n)$ is j times the imaginary part of $B(\mu,\nu)$, which we denote $B_i(\mu,\nu)$. Thus we have the transform pairs

$$b_e(m,n) \longleftrightarrow B_r(\mu,\nu)$$

$$b_o(m,n) \longleftrightarrow j\, B_i(\mu,\nu)$$

$$b(m,n) = b_e(m,n) + b_o(m,n) \longleftrightarrow B(\mu,\nu) = B_r(\mu,\nu) + j\, B_i(\mu,\nu).$$

Equation 1 may now be rewritten as a convolution in the frequency domain:

$$j\, B_i(\mu,\nu) = \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} W(\theta,\omega)\, B_r(\mu-\theta, \nu-\omega)\, d\theta d\omega, \tag{2}$$

where $W(\theta,\omega)$ is the Fourier transform of the window function $w(m,n)$.

$$W(\theta,\omega) = \sum_{m,n} w(m,n) \exp[-j\theta m - j\omega n]$$

$$= \left(\sum_{n=-\infty}^{\infty} \exp[-j\omega n] \right) \left(\sum_{m=1}^{\infty} \exp[-j\theta m] \right)$$

$$- \left(\sum_{n=-\infty}^{\infty} \exp[-j\omega n] \right) \left(\sum_{m=-\infty}^{-1} \exp[-j\theta m] \right)$$

$$+ \sum_{n=1}^{\infty} \exp[-j\omega n] - \sum_{n=-\infty}^{-1} \exp[-j\omega n].$$

The sum $\sum_{n=-\infty}^{\infty} \exp[-j\omega n]$ is operationally equal to $2\pi u_o(\omega)$ for $-\pi < \omega \leq \pi$, and the term

$$\sum_{m=1}^{\infty} \exp[-j\theta m] - \sum_{m=-\infty}^{-1} \exp[-j\theta m] = -j \cot \theta/2.$$

Therefore

$$W(\theta, \omega) = 2\pi\, u_o(\omega)[-j \cot \theta/2] + [-j \cot \omega/2] \qquad -\pi < \theta,\ \omega < \pi.$$

Substituting in Eq. 2, we get

$$B_i(\mu, \nu) = \frac{-1}{2\pi} \int_{-\pi}^{\pi} B_r(\theta, \nu) \cot\left[\frac{\mu-\theta}{2}\right] d\theta - \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} B_r(\theta, \omega) \cot\left[\frac{\nu-\omega}{2}\right] d\omega d\theta. \qquad (3)$$

We shall refer to Eq. 3 as a two-dimensional Hilbert transform.

2. Implementation as a Computer Program

Notice that Eq. 3 is really two one-dimensional Hilbert transforms in cotangent form.[2] If, on a digital computer, we wish to use Eq. 3 to find the imaginary part of $B(\mu, \nu)$, we must approximate the cotangent integral

$$G(\omega) = \frac{-1}{2\pi} \int_{-\pi}^{\pi} F(\theta) \cot\left[\frac{\omega-\theta}{2}\right] d\theta. \qquad (4)$$

We assume that $F(\theta)$ is real, but not necessarily even, and is the transform of an M-point sequence. Equation 4 is a circular convolution of $F(\theta)$ and $\cot \theta/2$, so we might consider using a DFT to approximate the integral. Dividing the unit circle into N equal parts (an even integer greater than or equal to M) and approximating the integral with a sum, we have

$$G(\omega) \approx \frac{-1}{2\pi} \frac{2\pi}{N} \sum_{k=\frac{-N}{2}+1}^{N/2} F\left(\frac{2\pi k}{N}\right) \cot\left[\frac{\omega - \frac{2\pi k}{N}}{2}\right].$$

Evaluating $G(\omega)$ for $\omega = 2\pi\ell/N$ gives

$$G\left(\frac{2\pi\ell}{N}\right) \approx \frac{-1}{N} \sum_k F\left(\frac{2\pi k}{N}\right) \cot\left[\frac{\pi}{N}(\ell-k)\right]$$

$$= \text{DFT}\left[\text{IDFT}\left[F\left(\frac{2\pi k}{N}\right)\right] \cdot \text{IDFT}\left[-\cot\frac{\pi\ell}{N}\right]\right].$$

This formulation is undesirable, however, because the cotangent function becomes infinite for $\ell = 0$.

Since the integrand in Eq. 4 is periodic, we may alter the limits of integration without affecting the answer.

$$G(\omega) = \frac{-1}{2\pi} \int_{-\pi+\frac{\pi}{N}}^{\pi+\frac{\pi}{N}} F(\theta) \cot\left[\frac{\theta-\omega}{2}\right] d\theta.$$

Going through the same approximation, we get

$$G\left(\frac{2\pi\ell}{N}\right) \approx \frac{-1}{N} \sum_k F\left(\frac{2\pi\left(k+\frac{1}{2}\right)}{N}\right) \cot\left[\frac{\pi}{N}\left(\ell-k-\frac{1}{2}\right)\right]$$

$$= \text{DFT}\left[\text{IDFT}\left[F\left(\frac{2\pi\left(k+\frac{1}{2}\right)}{N}\right)\right] \cdot \text{IDFT}\left[-\cot\frac{\pi}{N}\left(\ell-\frac{1}{2}\right)\right]\right]. \tag{5}$$

We have sampled $F(\theta)$ and $\cot \theta/2$ at $\theta = \pi/N, 3\pi/N, \ldots$ instead of $\theta = 0, 2\pi/N, 4\pi/N \ldots$.

If $f(n)$ is the inverse Fourier transform of $F(\theta)$, and $f'(n)$ is the inverse Fourier transform of $F\left(\theta+\frac{\pi}{N}\right)$, then

$$f'(n) = f(n) \exp\left[\frac{-j\pi n}{N}\right].$$

Since $N \geq M$, the $\text{IDFT}\left[F\left(\frac{2\pi k}{N}\right)\right]$ will be one period of the periodic extension of $f(n)$. The $\text{IDFT}\left[F\left(\frac{2\pi\left(k+\frac{1}{2}\right)}{N}\right)\right]$ will be one period of the periodic extension of $f'(n)$. Consequently if

$$p(n) = \text{IDFT}\left[F\left(\frac{2\pi k}{N}\right)\right]$$

and

$$p'(n) = \text{IDFT}\left[F\left(\frac{2\pi\left(k+\frac{1}{2}\right)}{N}\right)\right],$$

then

$$p'(n) = p(n)\exp\left[\frac{-j\pi n}{N}\right] \qquad 0 \leq n < N/2$$

$$= p(n)\exp\left[\frac{-j\pi(N-n)}{N}\right] \qquad N/2 < n < N$$

$$= 0 \qquad n = N/2.$$

The IDFT $\left[-\cot\left[\frac{\pi}{N}\left(\ell-\frac{1}{2}\right)\right]\right]$ can be shown to be

$$h'(n) = -j\exp\left[\frac{j\pi n}{N}\right] \qquad 1 \leq n \leq N-1$$

$$= 0 \qquad n = 0.$$

By substituting in Eq. 5, we get $G\left(\frac{2\pi\ell}{N}\right) \approx \text{DFT}[g(n)]$, where

$$g(n) = p'(n)\,h'(n) = -j\,p(n) \qquad 0 < n < N/2$$

$$= j\,p(n) \qquad N/2 < n < N$$

$$= 0 \qquad n = 0 \text{ and } N/2.$$

This yields a formulation in terms of the IDFT(F) directly, which we shall call the one-dimensional discrete Hilbert transform (DHT).

Recapitulating, to approximate the integral (4), we form the sequence $p(n)$ which is the IDFT of $F(\theta)$ sampled at $\theta = 2\pi k/N$, $k = 0, N-1$. Then multiply $p(n)$ by $-j$ for $0 < n < N/2$, and by $+j$ for $N/2 < n < N$ to form the sequence $g(n)$. Note that $g(0) = g(N/2) = 0$. Then take the DFT of $g(n)$ as the approximation to $G(\omega)$ for $\omega = 2\pi\ell/N$, $\ell = 0, N-1$.

We can now use this approximation to help solve Eq. 3. We rewrite Eq. 3.

$$B_i(\mu,\nu) = \frac{-1}{2\pi}\int_{-\pi}^{\pi} B_r(\theta,\nu)\cot\left[\frac{\mu-\theta}{2}\right]d\theta - \frac{1}{2\pi}\int_{-\pi}^{\pi} B_s(\omega)\cot\left[\frac{\nu-\omega}{2}\right]d\omega, \tag{6}$$

where $B_s(\omega) = \frac{1}{2\pi}\int_{-\pi}^{\pi} B_r(\theta,\omega)\,d\theta$. We can approximate $B_s(\omega)$ by the sum

$$B_s\left(\frac{2\pi \ell}{N}\right) \approx \frac{1}{N} \sum_{k=0}^{N-1} B_r\left(\frac{2\pi k}{N}, \frac{2\pi \ell}{N}\right)$$

and then use the one-dimensional DHT to approximate the second integral in Eq. 6. By discrete representation of the parameter ν in the first integral to $2\pi\ell/N$, $\ell = 0, N-1$, and doing N discrete Hilbert transforms, we get an approximation to $B_i(\mu, \nu)$ for $\mu = 2\pi k/N$ and $\nu = 2\pi\ell/N$; $k, \ell = 0, N-1$. This approximation will be called a two-dimensional DHT.

3. Factoring Two-Dimensional Magnitude-Squared Functions

We shall now show how the two-dimensional DHT may be used to factor a two-dimensional magnitude-squared frequency response. This problem is equivalent to finding an array b(m, n) whose autocorrelation function q(m, n) is given. We wish, however, to add the restriction that b(m, n) be a one-sided array. If we denote the Fourier transform of q(m, n) by $Q(\mu, \nu)$ then the following relations hold.

$$q(m, n) = \sum_{r, s} b(r, s)\, b(r+m, s+n)$$

$$q(m, n) = q(-m, -n)$$

$$Q(\mu, \nu) = |B(\mu, \nu)|^2 \qquad Q(\mu, \nu) \geq 0.$$

It is important to realize that were it not for the restriction that b(m, n) be one-sided, it would be easy to generate a variety of arrays by setting $B(\mu, \nu) = \sqrt{Q(\mu, \nu)}\, \exp[j\, \phi(\mu, \nu)]$ and taking the inverse Fourier transform. To meet the requirement that b(m, n) be real, $\phi(\mu, \nu)$ is an arbitrary phase function such that

$$\exp[j\, \phi(\mu, \nu)] = \exp[-j\, \phi(-\mu, -\nu)].$$

We can now define the complex cepstrum of an array to be the inverse Fourier transform of the log of the Fourier transform of that array. That is,

$$\hat{b}(m, n) = F^{-1}[\ln [F[b(m, n)]]],$$

where F denotes the Fourier transform and F^{-1} its inverse. Equivalently, $\hat{B}(\mu, \nu) = \ln [B(\mu, \nu)]$.

If $\hat{b}(m, n)$ is one-sided, then we can show that b(m, n) must also be one-sided. Rewrite Eq. 7 as

$$\sum_{m=1}^{\infty} \sum_{n=-\infty}^{\infty} \hat{b}(m, n)\, w^{-m} z^{-n} + \sum_{n=0}^{\infty} \hat{b}(0, n)\, z^{-n} = \ln \left[\sum_{m, n} b(m, n)\, w^{-m} z^{-n}\right],$$

where we have used the definitions of the Fourier transform and one-sided arrays, and have made the substitutions $w = \exp(j\mu)$ and $z = \exp(j\nu)$ for convenience. Now we can use the power series for exponentials on both sides.

$$\sum_{m,n} b(m,n)\, w^{-m} z^{-n} = \sum_{k=0}^{\infty} \frac{1}{k!} \left[\sum_{m=1}^{\infty} \sum_{n=-\infty}^{\infty} \hat{b}(m,n)\, w^{-m} z^{-n} + \sum_{n=0}^{\infty} \hat{b}(0,n)\, z^{-n} \right]^{k}.$$

Applying the binomial theorem to the right side, we get

$$= \sum_{k=0}^{\infty} \frac{1}{k!} \sum_{\ell=0}^{k} \left[\sum_{m=1}^{\infty} \sum_{n=-\infty}^{\infty} \hat{b}(m,n)\, w^{-m} z^{-n} \right]^{\ell} \left[\sum_{n=0}^{\infty} \hat{b}(0,n)\, z^{-n} \right]^{k-\ell} \frac{k!}{\ell!\,(\ell-k)!}.$$

By examining the right side closely and remembering that $k-\ell \geqslant 0$, we see that there are no coefficients of $w^{-m}z^{-n}$ for $m \leqslant -1$, and for $m = 0$, $n \leqslant -1$. After equating similar powers of w and z on both sides of the equation, the answer implies that $b(m,n) = 0$ for $m \leqslant -1$ and for $m = 0$, $n \leqslant -1$. Therefore, $b(m,n)$ is one-sided if $\hat{b}(m,n)$ is one-sided.

Now it is easy to see how to find $B(\mu,\nu)$ that satisfies our requirements. First, we take the logarithm of $Q(\mu,\nu)$ which is also real and symmetric, and multiply it by one-half. Since $|B(\mu,\nu)|^2 = Q(\mu,\nu)$, we have $\frac{1}{2}\ln[Q(\mu,\nu)] = \ln|B(\mu,\nu)|$. Applying the two-dimensional DHT to ln B gives a phase function $\phi(\mu,\nu)$. Note that the cepstrum $\hat{b}(m,n) = F^{-1}[\ln|B(\mu,\nu)| + j\,\phi(\mu,\nu)]$ is one-sided because the odd part of $\hat{b}(m,n)$ which corresponds to the transform of $\phi(\mu,\nu)$ was constructed to make it so. $B(\mu,\nu)$ is formed by

$$B(\mu,\nu) = \sqrt{Q(\mu,\nu)}\ (\cos[\phi(\mu,\nu)] + j\sin[\phi(\mu,\nu)])$$

and $b(m,n)$ is simply the inverse Fourier transform of $B(\mu,\nu)$. Since we constructed $\hat{b}(m,n)$ to be one-sided, $b(m,n)$ is also one-sided.

Some important points should be mentioned at this juncture. Although we have found (theoretically) the array $b(m,n)$ of the proper form, it may not be of finite extent even though its autocorrelation function is of finite extent. This is owing to the lack of the fundamental theorem of algebra. If we could carry out the procedure outlined above by using continuous Fourier transforms rather than DFTs, then we could calculate the infinite extent array whose autocorrelation would be the finite extent autocorrelation function with which we started. Using the computer, we must use DFTs, and so the result in this case is an aliased version of the infinite extent answer.

We now give a numerical example of using the two-dimensional DHT to find arrays from their autocorrelation functions. A program has been written using double-precision floating-point arithmetic to perform this procedure on an IBM 360/67 computer.

For the autocorrelation function q(m, n) shown in Fig. XI-1a the exact b(m, n) is shown in Fig. XI-1b. The calculated answer (done with 256-point DFTs) differed from the exact answer by 3×10^{-16} in the worst case. The same example done with 16-point DFTs differed from the exact answer by 10^{-4} in the worst case. Errors were somewhat larger when $Q(\mu, \nu)$ was very close to zero at some point. Several other examples were tried with similar results.

0.125	0.5	0.0
0.25	1.3125	0.25
0.0	0.5	0.125

(a)

	0.5	0
	1.0	0.25
		0

(b)

Fig. XI-1.
(a) Autocorrelation of our array.
(b) One-sided array used to test the two-dimensional DHT program.

In summary, a two-dimensional DHT has been described which will solve the problem of finding a one-sided b(m, n) from its autocorrelation function q(m, n). The derived b(m, n) may not be of finite extent, however, because of theoretical difficulties subject to the failure of the fundamental theorem of algebra in two-dimensional space.

References

1. B. B. Read and S. Treitel, "The Stabilization of Two-dimensional Recursive Filters via the Discrete Hilbert Transform," IEEE Trans., Vol. GE-11, No. 3, pp. 153-160, July 1973.

2. V. Cizek, "Discrete Hilbert Transform," IEEE Trans., Vol. AU-18, No. 4, pp. 340-343, December 1970.

3. B. Gold, A. V. Oppenheim, and C. M. Rader, "Theory and Implementation of Discrete Hilbert Transforms," Proceedings of the Symposium on Computer Processing in Communications, New York, April 8-10, 1969. Vol. XIX, Microwave Research Institute Symposia Series (Polytechnic Press, Polytechnic Institute of Brooklyn, New York, 1970), pp. 235-250.

4. B. Gold and C. M. Rader, Digital Processing of Signals (McGraw-Hill Book Company, Inc., New York, 1969), p. 247.

B. RECURSIBILITY OF TWO-DIMENSIONAL DIFFERENCE EQUATIONS

U. S. Navy Office of Naval Research (Contract N00014-67-A-0204-0064)

Dan E. Dudgeon

Difference equations, as one means of implementing a linear, shift-invariant filter, have been studied in great detail in one dimension, and to a somewhat lesser extent in two dimensions. In this report, we shall try to answer the question, "What conditions are sufficient to guarantee that a given two-dimensional difference equation can be iterated so that every output point may be calculated, given the input and an appropriate set of initial (or boundary) conditions?" In doing so, we generalize the concept of "causality" and also present recursible difference equations in a more general form than those widely given heretofore.

A one-dimensional sequence, $h(n)$, is said to be causal if and only if $h(n) = 0$ for $n < 0$. A sequence will be termed anticausal if and only if $h(n) = 0$ for $n > 0$.

The concept of a "one-sided" two-dimensional array is a generalization of a causal (or anticausal) sequence. For example, an array $b(m,n)$ is one-sided if $b(m,n) = 0$ for $m < 0$ and for $m = 0$, $n < 0$. Such an array is shown in Fig. XI-2a. Note that, for the most part, $b(m,n)$ lies at one side of the line $m = 0$. We shall also consider arrays

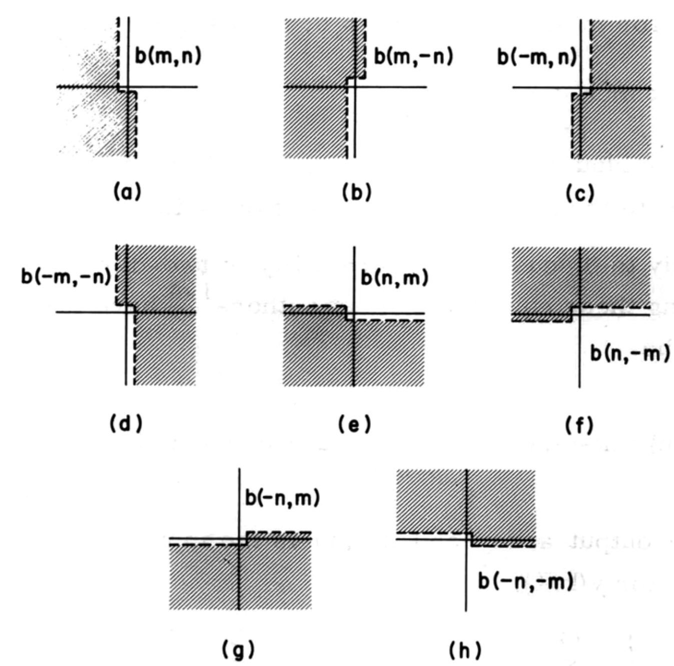

Fig. XI-2. One-sided arrays. Shaded portions enclose $b(m,n) \equiv 0$ regions.

generated by reflecting a one-sided array in either of both axes and rotating a one-sided array 90°, 180°, or 270° to be one-sided. In this manner, seven additional one-sided arrays are generated from the original b(m, n). They are b(m, -n), b(-m, n), b(-m, -n), b(n, m), b(n, -m), b(-n, m), and b(-n, -m) (see Fig. XI-2b through XI-2h).

We shall not stop here. Consider a line through the origin of the (m, n)-plane at any arbitrary angle θ with respect to the m axis. An equation describing such a line is $m \sin \theta - n \cos \theta = 0$. We can now define an array b(m, n) that is one-sided with respect to this line by requiring $b(m, n) = 0$ for $m \sin \theta - n \cos \theta < 0$, and for $m \sin \theta - n \cos \theta = 0$, $m \cos \theta + \sin \theta < 0$. For completeness, we again consider reflections with respect to the line at angle θ, and discover that a mirror-image set of one-sided arrays is generated by requiring $b(m, n) = 0$ for $m \sin \theta - n \cos \theta < 0$, and for $m \sin \theta - n \cos \theta = 0$, $m \cos \theta + n \sin \theta > 0$. Notice that the eight one-sided arrays shown in Fig. XI-2 are encompassed by our more general definitions. Four examples belong to one set, and the other four belong to the mirror-image set. We are now ready to formalize the definition of one-sidedness.

DEFINITION. A two-dimensional array b(m, n) is one-sided if and only if an angle θ can be found such that

$$b(m, n) = 0 \text{ for } m \sin \theta - n \cos \theta < 0,$$

$$\text{and for } m \sin \theta - n \cos \theta = 0, \; m \cos \theta + n \sin \theta > 0$$

$$(\text{or for } m \sin \theta - n \cos \theta = 0, \; m \cos \theta + n \sin \theta < 0)$$

$$-\infty < m, n < \infty.$$

Such an array is one-sided with respect to the line at angle θ. The parenthetical condition is used to generate the mirror-image set of one-sided arrays.

We are now ready to discuss the recursibility of two-dimensional difference equations. In considering these equations, most authors[1-4] have taken the form of the difference equation to be

$$\sum_{m=0}^{M} \sum_{n=0}^{N} b(m, n)\, y(k-m, \ell-n) = \sum_{p=0}^{P} \sum_{q=0}^{Q} a(p, q)\, x(k-p, \ell-q),$$

where $y(k, \ell)$ is the output array and $x(k, \ell)$ is the input array. This equation may be rewritten by solving for $y(k, \ell)$.

$$b(0, 0)\, y(k, \ell) = \sum_{p=0}^{P} \sum_{q=0}^{Q} a(p, q)\, x(k-p, \ell-q) - \sum_{\substack{m=0 \\ m+n\neq 0}}^{M} \sum_{n=0}^{N} b(m, n)\, y(k-m, \ell-n). \qquad (1)$$

Because the sums start at zero, this equation may be iterated; that is, after we solve for $y(k,\ell)$ we may then proceed to solve for $y(k,\ell+1)$, or, in this case, $y(k+1,\ell)$. By computing output points in the proper order, it is possible to generate all output points $y(k,\ell)$ from a set of initial conditions and the input array, $x(k,\ell)$. We define recursibility (equivalently, being recursible, or having a recursible form) as a property of certain two-dimensional difference equations that allow iteration of the equation, that is, choosing an indexing scheme such that any output point $y(k,\ell)$ may be calculated. As we shall see, Eq. 1 is not the most general form for a recursible two-dimensional difference equation.

Imagine two masks that are swept over the input and output arrays, picking up input and output values, multiplying them by the proper coefficients, and adding them in accordance with a general difference equation. This is illustrated in Fig. XI-3.

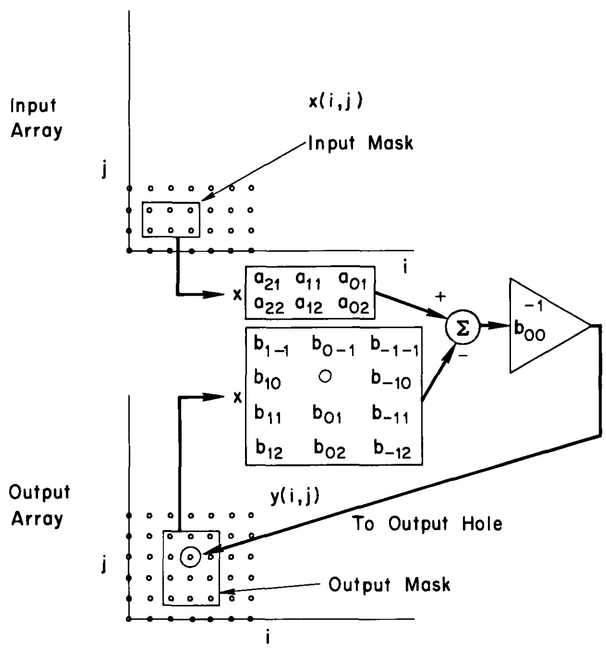

Fig. XI-3. Input and output masks to calculate $y(4,4)$.

Clearly, we must pick carefully the way in which the output mask is moved around so that an output point that has not yet been calculated will not be covered. This is completely equivalent to the requirement that we compute the output points in the proper order from the difference equation. Just as clearly, the input mask imposes no such constraint if we have the entire input before starting. Reasoning intuitively, if the hole in the output mask, which indicates the output point that is being calculated at the moment, is in the middle or even on the edge of the mask, we cannot calculate $y(k,\ell)$ for all values of (k, ℓ). Therefore, difference equations with such output masks are not recursible (see Fig. XI-4). If, however, the hole is on the corner of the mask, then there is no problem in iterating the difference equation, as indicated in Fig. XI-5 (see also Huang[2]).

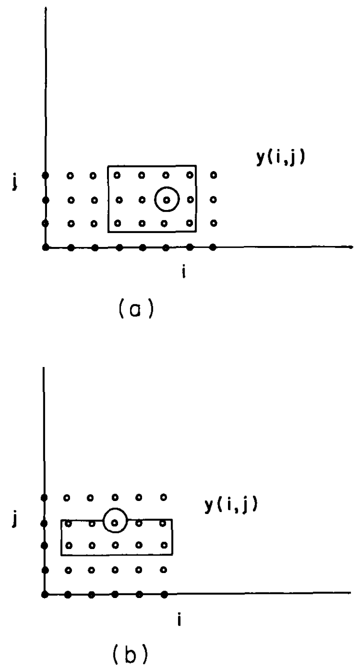

Fig. XI-4.
Nonrecursible output masks:
(a) with its hole in the middle;
(b) with its hole on an edge.

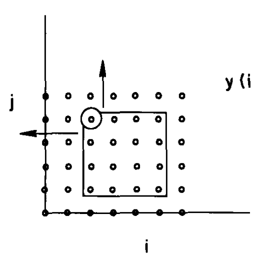

Fig. XI-5.
Output mask with its hole at a corner.

Figure XI-6 gives a simple example of iterating an equation whose output mask has its hole at a corner. This difference equation is of the form of Eq. 1. Such a mask can be swept across the output array, in this case column by column (Fig. XI-6a). After completing the first column out to all points in which we are interested, we start on the second column (Fig. XI-6b). We continue this iteration scheme until we have calculated all desired output points. Note that the recursion could also have proceeded in the horizontal direction by calculating y(k, ℓ) for a single row before proceeding to the next row.

We shall show now that a more general form of two-dimensional difference equations is also recursible. Assume that a difference equation has the form

$$\sum_{m=1}^{M} \sum_{n=-N}^{N} b(m,n)\, y(k-m, \ell-n) + \sum_{n=0}^{N} b(0,n)\, y(k, \ell-n) = x(k, \ell). \tag{2}$$

In this case we have made the input mask extremely simple, since its form is not important if the entire input is known beforehand. The recursion coefficients $b(m,n)$ are nonzero only in the regions $1 \leq m \leq M$, $-N \leq n \leq N$ and $m = 0$, $0 \leq n \leq N$. The difference equation (2) can be rewritten

$$b(0,0)\, y(k,\ell) = x(k,\ell) - \sum_{m=1}^{M} \sum_{n=-N}^{N} b(m,n)\, y(k-m, \ell-n) - \sum_{n=1}^{N} b(0,n)\, y(k, \ell-n). \tag{3}$$

The output mask for this equation has a jog in it, and the output hole is at the jog. (An example is given in Fig. XI-7 for M=2, N=2.) Therefore we may sweep the output mask along successive columns until we have generated the output at all points in which we are interested. In this way, any desired output point $y(k,\ell)$ may be obtained, and hence Eq. 3 is recursible.

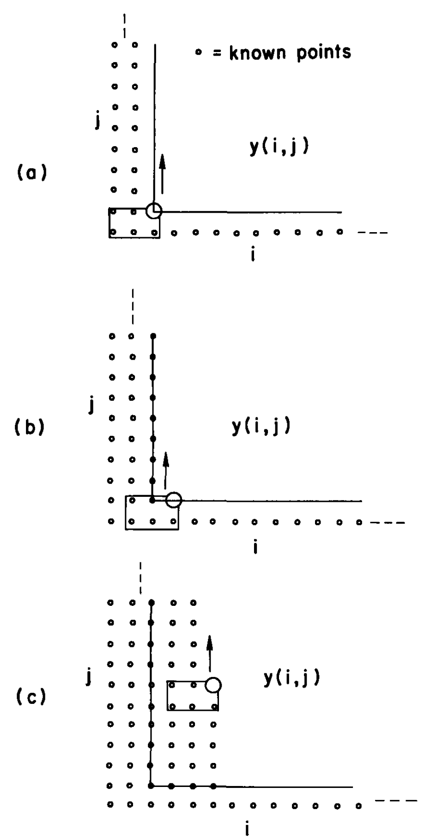

Fig. XI-6.

Iterating an output mask with its hole at a corner.

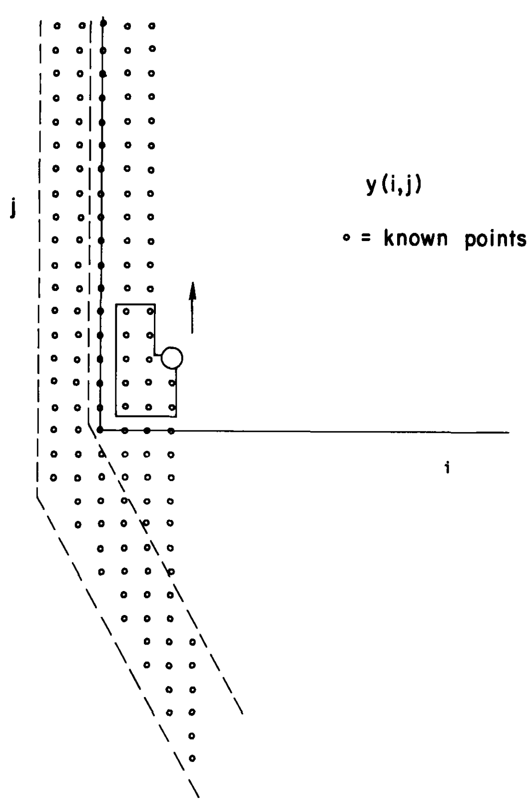

Fig. XI-7.

Output mask with its hole at a jog. Known points (∘) enclosed by dashed lines are necessary initial conditions.

We have previously defined a one-sided array. Note that if we consider the recursion coefficients b(m, n) of Eq. 3 to be an array, it is one-sided according to our definition, and is also of finite extent. This statement may also be made about the recursion coefficients of Eq. 1. This observation leads us to a statement relating one-sided arrays and recursible difference equations.

Consider the difference equation where the arrays a(p, q) and b(m, n) are of finite extent.

$$\sum_{m,n} b(m,n)\, y(k-m, \ell-n) = \sum_{p,q} a(p,q)\, x(k-p, \ell-q),$$

or equivalently

$$b(0,0)\, y(k,\ell) = \sum_{p,q} a(p,q)\, x(k-p, \ell-q) - \sum_{\substack{m,n \\ m \text{ and } n \text{ not both zero}}} b(m,n)\, y(k-m, \ell-n).$$

If the finite extent array b(m, n) is one-sided with respect to the line $m \sin\theta - n \cos\theta = 0$, then this difference equation is recursible, and any output point may be generated if the input and the appropriate initial conditions are known.

This statement is a direct consequence of the definition of one-sidedness. For $\theta = 0$, $\pm\pi/2$, and π, we have b(m, n) in the form of one of our eight original examples of

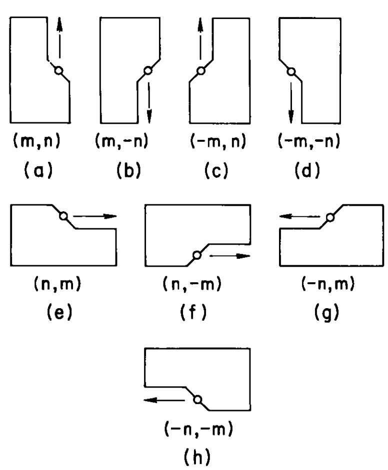

Fig. XI-8.

Eight forms for recursible output masks. Arrows indicate the direction of recursion.

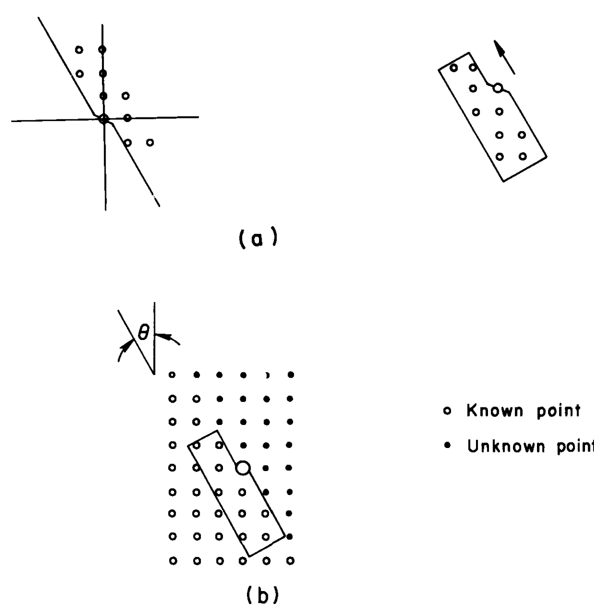

Fig. XI-9.

(a) "One-sided" b(m, n) for $\theta = \tan^{-1} \frac{1}{2}$ (left) and output mask (right).
(b) Iterating the output mask for $\theta = \tan^{-1} \frac{1}{2}$.

one-sided arrays (Fig. XI-2). Corresponding to each of these is an output mask, as shown in Fig. XI-8. The reader can verify that these output masks can be swept over the output plane in such a way that any desired output point $y(k, \ell)$ is generated.

We shall give some examples for the more general case. Consider that the array $b(m,n)$ is one-sided with respect to the line at angle $\theta = \tan^{-1} \frac{1}{2}$. Then the output mask will be of the form shown in Fig. XI-9a, and may be swept across the output array at angle θ from the horizontal axis, as indicated in Fig. XI-9b.

As another example, consider the set of recursion coefficients:

$$b(0,0) = 1$$

$$b(1,0) = \frac{1}{2}$$

$$b(0,1) = \frac{1}{4}$$

$$b(m,n) = 0 \qquad \text{elsewhere.}$$

The corresponding difference equation is

$$y(k,\ell) = x(k,\ell) - \frac{1}{2} y(k-1,\ell) - \frac{1}{4} y(k,\ell-1) \tag{4}$$

which has the output mask shown in Fig. XI-10.

Using our general definition of one-sidedness for two-dimensional arrays, we note that this particular $b(m,n)$ is one-sided for many values of the parameter θ. Consequently, Eq. 4 may be iterated in an infinite number of ways. The four most obvious

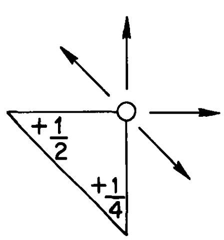

Fig. XI-10.
Output mask for Eq. 4. Arrows indicate four possible directions of recursion.

ways are indicated in Fig. XI-10. Clearly, these examples generalize to arbitrary θ with rational arc tangent, and therefore the statement is indeed true. For most practical applications, however, we expect that only the eight original examples (Fig. XI-2) need be considered.

In discussing these recursible forms for two-dimensional difference equations, no mention has been made of stability. Any two-dimensional difference equation may be forced to have a recursible form by substituting other variables for the indices. But if this alters the unit sample response (the output of the difference equation when the input is a unit sample at the origin), then possibly a stable filter would be changed into an unstable one. The approach taken here is that a given difference equation is stable, and we may not substitute in the indices. Then we can check the form of the equation to determine whether it is recursible.

References

1. B. D. O. Anderson and E. I. Jury, "Stability Tests for Two-Dimensional Recursive Filters," IEEE Trans., Vol. AU-21, No. 4, pp. 366-371, August 1973.
2. T. S. Huang, "Stability of Two-Dimensional Recursive Filters," IEEE Trans., Vol. AU-20, No. 2, pp. 158-163, June 1972.
3. G. A. Maria and M. M. Fahmy, "On the Stability of Two-Dimensional Digital Filters," IEEE Trans., Vol. AU-21, No. 5, pp. 470-471, October 1973.
4. J. L. Shanks, S. Treitel, and J. H. Justice, "Stability and Synthesis of Two-Dimensional Recursive Filters," IEEE Trans., Vol. AU-20, No. 3, pp. 115-128, June 1972.