The Chirp z-Transform Algorithm and Its Application

§ By LAWRENCE R. RABINER, RONALD W. SCHAFER, and CHARLES M. RADER*

(Manuscript received November 21, 1968)

We discuss a computational algorithm for numerically evaluating the z-transform of a sequence of N *samples. This algorithm has been named the chirp* z*-transform algorithm. Using this algorithm one can efficiently evaluate the* z*-transform at* M *points in the* z*-plane which lie on circular or spiral contours beginning at any arbitrary point in the* z*-plane. The angular spacing of the points is an arbitrary constant;* M *and* N *are arbitrary integers.*

The algorithm is based on the fact that the values of the z*-transform on a circular or spiral contour can be expressed as a discrete convolution. Thus one can use well-known high-speed convolution techniques to evaluate the transform efficiently. For* M *and* N *moderately large, the computation time is roughly proportional to* $(N + M) \log_2 (N + M)$ *as opposed to being proportional to* $N \cdot M$ *for direct evaluation of the* z*-transform at* M *points.*

Applications discussed include: enhancement of poles in spectral analysis, high resolution narrow-band frequency analysis, interpolation of band-limited waveforms, and the conversion of a base 2 fast Fourier transform program into an arbitrary radix fast Fourier transform program.

I. INTRODUCTION

In dealing with sampled data the z-transform plays the role which is played by the Laplace transform in continuous time systems. One example of its application is spectrum analysis. The computation of sampled z-transforms, which has been greatly facilitated by the fast Fourier transform algorithm, is further facilitated by the "chirp z-transform" algorithm described in this paper.[1,2]

* Mr. Rader is with Lincoln Laboratory, Massachusetts Institute of Technology, Lexington, Massachusetts. Lincoln Laboratory is operated with support from the U. S. Air Force.

The z-transform of a sequence of numbers x_n is defined as

$$X(z) = \sum_{n=-\infty}^{\infty} x_n z^{-n}, \tag{1}$$

a function of the complex variable z. In general both x_n and $X(z)$ could be complex. It is assumed that the sum on the right side of equation (1) converges for at least some values of z. We restrict ourselves to the z-transform of sequences with only a finite number N of nonzero points. Therefore, we can rewrite equation (1) without loss of generality as

$$X(z) = \sum_{n=0}^{N-1} x_n z^{-n} \tag{2}$$

where the sum in equation (2) converges for all z except $z = 0$.

Equations (1) and (2) are similar to the defining expressions for the Laplace transform of a train of equally spaced impulses of magnitudes x_n. Let the spacing of the impulses be T and let the train of impulses be

$$\sum_n x_n \, \delta(t - nT).$$

Then the Laplace transform is

$$\sum_n x_n e^{-snT}$$

which is the same as $X(z)$ if we let

$$z = e^{sT}. \tag{3}$$

For sampled waveforms the relation between the original waveform and the train of impulses is well understood in terms of the phenomenon of aliasing. Thus the z-transform of the sequence of samples of a time waveform is representative of the Laplace transform of the original waveform in a way which is well understood. The Laplace transform of a train of impulses repeats its values taken in a horizontal strip of the s-plane of width $2\pi/T$ in every other strip parallel to it. The z-transform maps each such strip into the entire z-plane or, conversely, the entire z-plane corresponds to any horizontal strip of the s-plane, for example, the region $-\infty < \sigma < \infty$, $-\pi/T \leqq \omega < \pi/T$, where $s = \sigma + j\omega$.

In the same correspondence, the $j\omega$ axis of the s-plane, along which we generally equate the Laplace transform with the Fourier transform, is the unit circle in the z-plane; the origin of the s-plane corresponds to $z = 1$. The interior of the z-plane unit circle corresponds to the left half of the s-plane; the exterior corresponds to the right half plane. Straight lines in the s-plane correspond to circles or spirals in the z-plane. Figure 1 shows the correspondence of a contour in the s-plane to a contour in the z-plane. To evaluate the Laplace transform of the impulse train along the linear contour is to evaluate the z-transform of the sequence along the spiral contour.

Values of the z-transform are usually computed along the path corresponding to the $j\omega$ axis, namely the unit circle. This gives the discrete equivalent of the Fourier transform and has many applications including the estimation of spectra, filtering, interpolation, and correlation. The applications of computing z-transforms off the unit circle are fewer, but one is presented in this paper, namely the enhancement of spectral resonances in systems for which one has some foreknowledge of the locations of poles and zeros.

Just as we can only compute equation (2) for a finite set of samples, so we can only compute equation (2) at a finite number of points, say z_k:

$$X_k = X(z_k) = \sum_{n=0}^{N-1} x_n z_k^{-n}. \tag{4}$$

The special case which has received the most attention is the set of points equally spaced around the unit circle,

$$z_k = \exp\left(j \frac{2\pi}{N} k\right), \qquad k = 0, 1, \cdots, N-1 \tag{5}$$

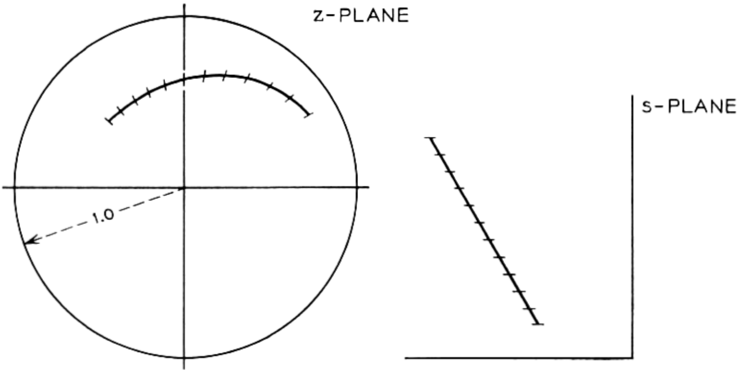

Fig. 1 — The correspondence of a z-plane contour to an s-plane contour through the relation $z = e^{sT}$.

for which

$$X_k = \sum_{n=0}^{N-1} x_n \exp\left(-j\frac{2\pi}{N}nk\right), \qquad k = 0, 1, \cdots, N-1. \tag{6}$$

Equation (6) is called the discrete Fourier transform. The reader may easily verify that, in equation (5), other values of k merely repeat the same N values of z_k, which are the Nth roots of unity. The discrete Fourier transform has assumed considerable importance, partly because of its nice properties but mainly because since 1965 it has become widely known that the computation of equation (6) can be achieved, not in the N^2 complex multiplications and additions called for by direct application of equation (6), but in something of the order of $N \log_2 N$ operations if N is a power of two, or $N \sum_i m_i$ operations if the integers m_i are the prime factors of N. Any algorithm which accomplishes this is called a fast Fourier transform. Much of the importance of the fast Fourier transform is that the discrete Fourier transform may be used as a stepping stone to computing lagged products such as convolutions, autocorrelations, and cross correlations more rapidly than before.[3,4] The discrete Fourier transform has some limitations which can be eliminated using the chirp z-transform algorithm which we describe. We also investigate the computation of the z-transform on a more general contour, of the form

$$z_k = AW^{-k}, \qquad k = 0, 1, \cdots, M-1 \tag{7a}$$

where M is an arbitrary integer and both A and W are arbitrary complex numbers of the form

$$A = A_o \exp(j2\pi\theta_o) \tag{7b}$$

and

$$W = W_o \exp(j2\pi\varphi_o). \tag{7c}$$

(See Fig. 2.) The case $A = 1$, $M = N$, and $W = \exp(-j2\pi/N)$ corresponds to the discrete Fourier transform. The general z-plane contour begins with the point $z = A$ and, depending on the value of W, spirals in or out with respect to the origin. If $W_o = 1$, the contour is an arc of a circle. The angular spacing of the samples is $2\pi\varphi_o$. The equivalent s-plane contour begins with the point

$$s_o = \sigma_o + j\omega_o = \frac{1}{T}\ln A \tag{8}$$

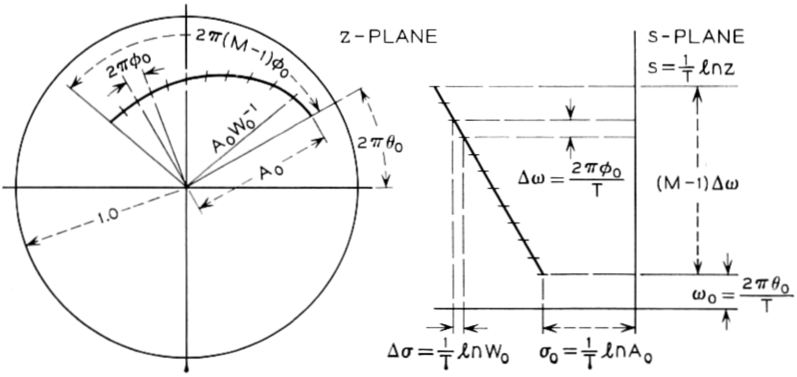

Fig. 2 — An illustration of the independent parameters of the chirp z-transform algorithm. The upper figure shows how the z-transform is evaluated on a spiral contour starting at the point $z = A$. The lower figure shows the corresponding straight line contour and independent parameters in the s-plane.

and the general point on the s-plane contour is

$$s_k = s_o + k(\Delta\sigma + j\,\Delta\omega) = \frac{1}{T}(\ln A - k \ln W), \qquad k = 0, 1, \cdots, M - 1. \tag{9}$$

Since A and W are arbitrary complex numbers we see that the points s_k lie on an arbitrary straight line segment of arbitrary length and sampling density. Clearly the contour indicated in equation (7a) is not the most general contour, but it is considerably more general than that for which the discrete Fourier transform applies. In Fig. 2, an example of this more general contour is shown in both the z-plane and the s-plane.

To compute the z-transform along this more general contour would seem to require NM multiplications and additions since the special symmetries of exp $(j2\pi k/N)$ which are exploited in the derivation of the fast Fourier transform are absent in the more general case. However, we shall see that by using the sequence $W^{n^2/2}$ in various roles we can apply the fast Fourier transform to the computation of the z-transform along the contour of equation (7a). Since for $W_o = 1$, the sequence $W^{n^2/2}$ is a complex sinusoid of linearly increasing frequency, and since a similar waveform used in some radar systems has the picturesque name "chirp", we call the algorithm we are about to present the chirp z-transform. Since this transform permits computing the z-transform on a more general contour than the fast Fourier transform permits, it is more flexible than the fast Fourier transform, although it is also considerably slower. The additional freedoms offered by the chirp z-transform include:

(*i*) the number of time samples does not have to equal the number of samples of the z-transform.

(*ii*) Neither M nor N need be a composite integer.

(*iii*) The angular spacing of the z_k is arbitrary.

(*iv*) The contour need not be a circle but can spiral in or out with respect to the origin. In addition, the point z_o is arbitrary, but this is also the case with the fast Fourier transform if the samples x_n are multiplied by z_o^{-n} before transforming.

II. DERIVATION OF THE CHIRP Z-TRANSFORM

Along the contour of equation (7a), equation (4) becomes

$$X_k = \sum_{n=0}^{N-1} x_n A^{-n} W^{nk}, \qquad k = 0, 1, \cdots, M-1 \tag{10}$$

which, at first appearance, seems to require NM complex multiplications and additions, as we have already observed. But, let us use Bluestein's ingenious substitution[5]

$$nk = \frac{n^2 + k^2 - (k-n)^2}{2} \tag{11}$$

for the exponent of W in equation (10). This produces an apparently more unwieldly expression

$$X_k = \sum_{n=0}^{N-1} x_n A^{-n} W^{(n^2/2)} W^{(k^2/2)} W^{-(k-n)^2/2}, \qquad k = 0, 1, \cdots, M-1 \tag{12}$$

but in fact equation (12) can be thought of as a three step process consisting of: (*i*) forming a new sequence y_n by weighting the x_n according to the equation

$$y_n = x_n A^{-n} W^{n^2/2}, \qquad n = 0, 1, \cdots, N-1, \tag{13}$$

(*ii*) convolving y_n with the sequence v_n defined as

$$v_n = W^{-n^2/2} \tag{14}$$

to give a sequence g_k

$$g_k = \sum_{n=0}^{N-1} y_n v_{k-n}, \qquad k = 0, 1, \cdots, M-1, \tag{15}$$

and (*iii*) multiplying g_k by $W^{k^2/2}$ to give X_k

$$X_k = g_k W^{k^2/2}, \qquad k = 0, 1, \cdots, M - 1. \tag{16}$$

The three step process is illustrated in Fig. 3. Steps *i* and *iii* require N and M multiplications respectively; step *ii* is convolution which may be computed by the high speed technique disclosed by Stockham, based on the use of the fast Fourier transform.[3] Step *ii* is the major part of the computational effort and requires a time roughly proportional to $(N + M) \log (N + M)$.

Bluestein used the substitution of equation (11) to convert a discrete Fourier transform to a convolution as in Fig. 3. The linear system to which the convolution is equivalent can be called a chirp filter which is, in fact, also sometimes used to resolve a spectrum. Bluestein showed that for N a perfect square, the chirp filter could be synthesized recursively with $N^{1/2}$ multipliers and the computation of a discrete Fourier transform could then be proportional to $N^{3/2}$ (see Ref. 5).

The flexibility and speed of the chirp z-transform algorithm are related to the flexibility and speed of the method of high-speed convolution using the fast Fourier transform. Recall that the product of the discrete Fourier transforms of two sequences is the discrete Fourier transform of the circular convolution of the two sequences; therefore, a circular convolution is computable as two discrete Fourier transforms, the multiplication of two arrays of complex numbers, and an inverse discrete Fourier transform, which can also be computed by the fast Fourier transform. Ordinary convolutions can be computed as circular convolutions by appending zeroes to the end of one or both sequences so that the correct numerical answers for the ordinary convolution can result from a circular convolution.

We now summarize the details of the chirp z-transform algorithm on the assumption that an already existing fast Fourier transform

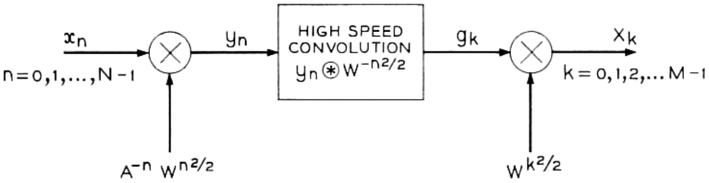

Fig. 3 — An illustration of the steps involved in computing values of the z-transform using the chirp z-transform algorithm.

program (or special purpose machine) is available to compute discrete and inverse discrete Fourier transforms.

We begin with a waveform in the form of N samples x_n and we seek M samples of X_k where A and W have also been chosen:

(*i*) We choose L, the smallest integer greater than or equal to $N + M - 1$ which is also compatible with our high speed fast Fourier transform program. For most users this will mean L is a power of two. Notice that while many fast Fourier transform programs will work for arbitrary L, they are not equally efficient for all L. At the very least, L should be highly composite.

(*ii*) We form an L point sequence y_n from x_n by the equation

$$y_n = \begin{cases} A^{-n}W^{n^2/2}x_n & n = 0, 1, 2, \cdots, N-1 \\ 0 & n = N, N+1, \cdots, L-1 \end{cases} . \tag{17}$$

(*iii*) We compute the L point discrete Fourier transform of y_n by the fast Fourier transform, calling it Y_r, $r = 0, 1, \ldots, L - 1$.

(*iv*) We define an L point sequence v_n by the relation

$$v_n = \begin{cases} W^{-n^2/2} & 0 \leqq n \leqq M-1 \\ W^{-(L-n)^2/2} & L - N + 1 \leqq n < L. \\ \text{arbitrary} & \text{other } n, \text{ if any} \end{cases} \tag{18}$$

If L is exactly equal to $M + N - 1$, the region in which v_n is arbitrary will not exist. If the region does exist an obvious possibility is to increase M, the desired number of points of the z-transform we compute, until the region does not exist.

Notice that v_n could be cut into two with a cut between $n = M - 1$ and $n = L - N + 1$; if the two pieces were abutted differently, the resulting sequence would be a slice out of the indefinite length sequence $W^{-n^2/2}$. This is illustrated in Fig. 4. The sequence v_n is defined the way it is in order to force the circular convolution to give us the desired numerical results of an ordinary convolution.

(*v*) We compute the discrete Fourier transform of v_n and call it V_r, $r = 0, 1, \ldots, L - 1$.

(*vi*) We multiply V_r and Y_r point by point, giving G_r:

$$G_r = V_r Y_r , \qquad r = 0, 1, \cdots, L-1.$$

(*vii*) We compute the L point inverse discrete Fourier transform g_k, of G_r.

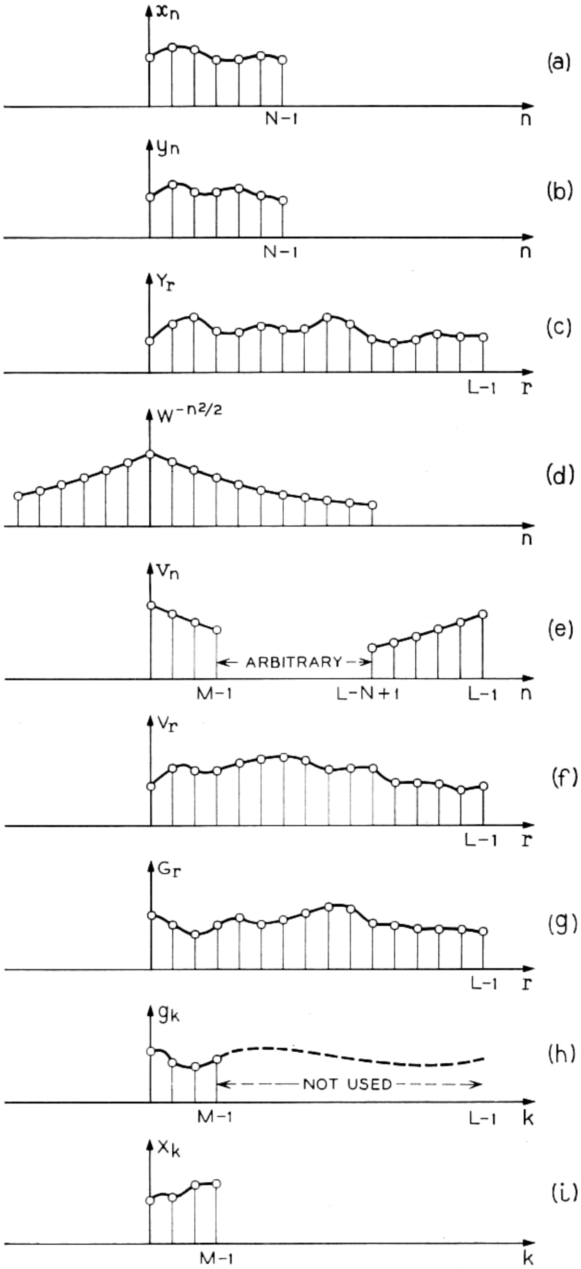

Fig. 4 — Schematic representation of the various sequences involved in the chirp z-transform algorithm: (a) input sequence x_n with N values. (b) weighted input sequence $y_n = A^{-n}W^{n^2/2}x_n$. (c) discrete Fourier transform of y_n. (d) values of the indefinite sequence $W^{-n^2/2}$. (e) sequence v_n formed appropriately from segments of $W^{-n^2/2}$. (f) discrete Fourier transform of v_n. (g) product $G_r = Y_r \cdot V_r$. (h) inverse discrete Fourier transform of G_r. (i) desired M values of the z-transform.

(*viii*) We multiply g_k by $W^{k^2/2}$ to give us the desired X_k :

$$X_k = W^{k^2/2} g_k , \qquad k = 0, 1, 2, \cdots , M - 1.$$

The g_k for $k \geqq M$ are discarded.

Figure 4 represents typical waveforms (magnitudes shown, phase omitted) involved in each step of the process.

III. FINE POINTS OF THE COMPUTATION

3.1 *Operation Count and Timing Considerations*

An operation count can be made, roughly, from the eight steps just presented:

(*i*) We assume that step *i*, that is, choosing L, is a negligible operation.

(*ii*) Forming y_n from x_n requires N complex multiplications, not counting the generation of the constants $A^{-n}W^{n^2/2}$. The constants may be prestored, computed as needed, or generated recursively as needed. The recursive computation would require two complex multiplications per point.

(*iii*) An L point discrete Fourier transform requires a time $k_{\text{FFT}}L \log_2 L$ for L a power of two, and a very simple fast Fourier transform program. More complicated (but faster) programs have more complicated computing time formulas.

(*iv*), (*v*) The value of v_n is computed for either M or N points, whichever is greater. The symmetry in $W^{-n^2/2}$ permits the other values of v_n to be obtained without computation. Again, v_n can be computed recursively. The fast Fourier transform takes the same time as that in step *iii*. If the same contour is used for many sets of data, V_r need only be computed once, and stored.

(*vi*) This step requires L complex multiplications.

(*vii*) This is another fast Fourier transform and requires the same time as step *iii*.

(*viii*) This step requires M complex multiplications.

As the number of samples of x_n or X_k grows large, the computation time for the chirp z-transform grows asymptotically as something proportional to $L \log_2 L$. This is the same sort of asymptotic dependence of the fast Fourier transform, but the constant of proportionality is bigger for the chirp z-transform because two or three fast Fourier transforms are required instead of one, and because L is greater than N or M. Still, the chirp z-transform is faster than the direct computation of equation (10) even for relatively modest values of M and N of the order of 50.

3.2 *Reduction in Storage*

The chirp z-transform can be put into a more useful form for computation by redefining the substitution of equation (11) to read

$$nk = \frac{(n - N_o)^2 + k^2 - (k - n + N_o)^2 + 2N_o k}{2}.$$

Equation (12) can now be rewritten as

$$X_k = W^{k^2/2} W^{N_o k} \sum_{n=0}^{N-1} x_n A^{-n} W^{(n-N_o)^2/2} W^{-(k-n+N_o)^2/2}.$$

The form of the new equation is similar to equation (12) in that the input data x_n are preweighted by a complex sequence $(A^{-n}W^{(n-N_o)^2/2})$, convolved with a second sequence $(W^{-(n-N_o)^2/2})$, and postweighted by a third sequence $(W^{k^2/2}W^{N_o k})$ to compute the output sequence X_k . However, there are differences in the detailed procedures for realizing the chirp z-transform. The input data x_n can be thought of as having been shifted by N_o samples to the left. For example, x_o is weighted by $W^{N_o{}^2/2}$ instead of W^0. The region over which $W^{-n^2/2}$ must be formed, in order to obtain correct results from the convolution, is

$$-N + 1 + N_o \leqq n \leqq M - 1 + N_o \,.$$

By choosing $N_o = (N - M)/2$ it can be seen that the limits over which $W^{-n^2/2}$ is evaluated are symmetric; that is, $W^{-n^2/2}$ is a symmetric function in both its real and imaginary parts. (Therefore, the transform of $W^{-n^2/2}$ is also symmetric in both its real and imaginary parts.) It can be shown that by using this special value of N_o only $(L/2 + 1)$ points of $W^{-n^2/2}$ need be calculated and stored, and these $(L/2 + 1)$ complex points can be transformed using an $L/2$ point transform*. Hence the total storage required for the transform of $W^{-n^2/2}$ is $L + 2$ locations.

The only other modifications to the detailed procedures for evaluating the chirp z-transform presented in Section II are:

(*i*) following the L point inverse discrete Fourier transform of step *vii*, the data of array g_k must be rotated to the left by N_o locations,

(*ii*) the weighting factor of the g_k is $W^{k^2/2}W^{N_o k}$ rather than $W^{k^2/2}$.

* The technique for transforming two symmetric L point sequences using one $L/2$ point fast fourier transform was demonstrated by J. Cooley at the fast Fourier transform workshop, Arden House, Harriman, New York, October 1968. The appendix summarizes this technique.

The additional factor $W^{N_o k}$ represents a data shift of N_o samples to the right, thus compensating for the initial shift and keeping the effective positions of the data invariant to the value of N_o used.

Now we can estimate the storage required to perform the chirp z-transform. Assuming that the entire process is to take place in core, storage is required for V_r which takes $L + 2$ locations, for y_n, which takes $2L$ locations, and perhaps for some other quantities which we wish to save, such as the input or values of $W^{n^2/2}$ or $A^{-n}W^{n^2/2}$.

3.3 *Additional Considerations*

Since the chirp z-transform permits $M \neq N$, it is possible that occasions will arise where $M \gg N$ or $N \gg M$. In these cases, if the smaller number is small enough, the direct method of equation (10) is called for. However, if even the smaller number is large it may be appropriate to use the methods of sectioning described by Stockham.[3] Either the lap-save or lap-add methods may be used. Sectioning may also be used when problems, too big to be handled in core memory, arise. We have not actually encountered any of these problems and have not programmed the chirp z-transform with provision for sectioning.

Since the contour for the chirp z-transform is a straight line segment in the s-plane, it is apparent that repeated application of the chirp z-transform can compute the z-transform along a contour which is piecewise spiral in the z-plane or piecewise linear in the s-plane.

Let us briefly consider the chirp z-transform algorithm for the case of z_k all on the unit circle. This means that the z-transform is like a Fourier transform. Unlike the discrete Fourier transform, which by definition gives N points of transform for N points of data, the chirp z-transform does not require $M = N$. Furthermore the z_k need not stretch over the entire unit circle but can be equally spaced along an arc. Let us assume, however, that we are really interested in computing the N point discrete Fourier transform of N data points. Still the chirp z-transform permits us to choose any value of N, highly composite, somewhat composite, or even prime, without strongly affecting the computation time. An important application of the chirp z-transform may be computing discrete Fourier transforms when N is not a power of two and when the program or special purpose device available for computing discrete Fourier transforms by fast Fourier transform is limited to when N is a power of two.

There is also no reason why the chirp z-transform cannot be extended to the case of transforms in two or more dimensions with similar considerations. The two dimensional discrete Fourier transform becomes a two dimensional convolution which can be computed by fast Fourier transform techniques.

Caution: For the ordinary fast Fourier transform the starting point of the contour is still arbitrary; merely multiply the waveform x_n by A^{-n} before using the fast Fourier transform and the first point on the contour is effectively moved from $z = 1$ to $z = A$. However, the contour is still restricted to a circle concentric with the origin. The angular spacing of z_k for the fast Fourier transform can also be controlled to some extent by appending zeros to the end of x_n before computing the discrete Fourier transform (to decrease the angular spacing of the z_k) or by choosing only P of the N points x_n and adding together all the x_n for which the n are congruent modulo P; that is, wrapping the waveform around a cylinder and adding together the pieces which overlap (to increase the angular spacing).

IV. APPLICATIONS OF THE ALGORITHM

Because of its flexibility, the chirp z-transform algorithm discussed in the Section III has many potential applications.

4.1 *Enhancement of Poles*

One advantage of the chirp z-transform algorithm over the fast Fourier transform is its ability to evaluate the z-transform at points both inside and outside the unit circle. This is important in the investigation of systems whose transfer functions can be represented as ratios of polynomials in z; that is, in finding poles and zeros of a linear system. By evaluating the transform off the unit circle, one can make the contour pass closer to the poles and zeros of the system, thus effectively reducing the bandwidths and sharpening the transfer function.

For example, a five-pole system was simulated at a 10 kHz sampling frequency. The poles were located at center frequencies of 270, 2290, 3010, 3500 and 4500 Hz with bandwidths of 30, 50, 60, 87 and 140 Hz, respectively. The z-plane pole positions are shown in Fig. 5. (Those familiar with speech will recognize these numbers as resonance positions appropriate for the vowel *i* in the word *beet.*) The input to the system was a periodic impulse train of period 100 samples; that is, 100 pulses per second. Impulse invariant techniques were used to simulate the system.[6] The z-transform of one period of steady state data (100 samples) was evaluated on two spirals outside the unit circle, one on the unit circle, and two spirals inside the unit circle. Figure 6 shows the five contours as they would appear in the s-plane and the s-plane pole positions. The contours are seen to be equi-

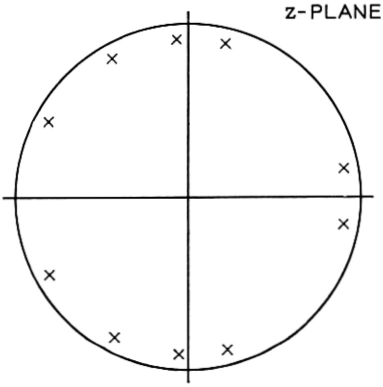

Fig. 5 — Representation of the z-plane locations of the poles of the linear system simulated in the text.

angularly spaced. The five sets of magnitude curves are shown in Fig. 7. The transform was evaluated at 50 equally spaced points from 0 to 4900 Hz, corresponding to $\varphi_o = - 1/100$. The sharpening of the magnitude response in the region of the poles is quite pronounced. Figure 6 indicates that contour 5 should be near optimum since it intersects three of the poles.

This example is a somewhat idealized case in that spectral samples were taken every 100 Hz; that is, at the harmonics of the fundamental frequency. Figure 8 shows the case for spectral data taken every 25 Hz on contour 5 of Fig. 6, along with the case where the spectral resolution is the same as shown in Fig. 7. This figure places in evidence the true nature of the z-transform of a finite number of samples. It is clear from equation (2) that $X(z)$ has no poles anywhere in the z-plane except at $z = 0$. There are instead $N-1$ zeros which manifest themselves in the ripples seen in the upper curve of Fig. 8. In many cases the poles of the original system which generated the samples are still in evidence because the zeros tend to be arrayed at approximately equal angular increments except at the locations of the original poles. Hence a pole usually manifests itself by an absence of zeros in the vicinity of that pole in the z-plane. Zeros of transmission are often masked by these effects when only a finite number of samples are transformed. Examples of this effect are given after equation (23).

It is of interest to examine the ability of the chirp z-transform algorithm to determine the bandwidth of a pole as well as its center frequency. To investigate this point, synthetic samples were generated with the bandwidth of the lowest pole (B_1) variable from 10 to 320 Hz by factors of 2; all other bandwidths and center frequencies were held at values used in the previous example. Again a fixed 100 pulse per second source excited each of the systems. Figure 9 illustrates the six sets of poles and three contours used in the investigation. Contour 3 extends into the right half-plane (spiral outside the unit circle) and is only close to the lowest pole. Contour 2 corresponds to the unit circle in the z-plane (that is, the discrete Fourier transform). Contour 1 is an appropriate left-half plane contour (spiral inside the unit circle) used for investigating this system. The resulting set of 18 magnitude curves (six different sets of poles and three contours) are shown in Fig. 10. The rows of Fig. 10 show magnitude curves with a fixed bandwidth and variable contour, whereas the columns show curves on the same contour with variable bandwidths. There are 801 points plotted in each curve in the range 0 to 5,000 Hz.

Looking down any column it is seen that as B_1 increases, the level of the first resonance decreases steadily. The variation in fine spectral detail resulting from the distribution of zeros in the neighborhood of the poles of the original system is seen clearly in column 1. For ex-

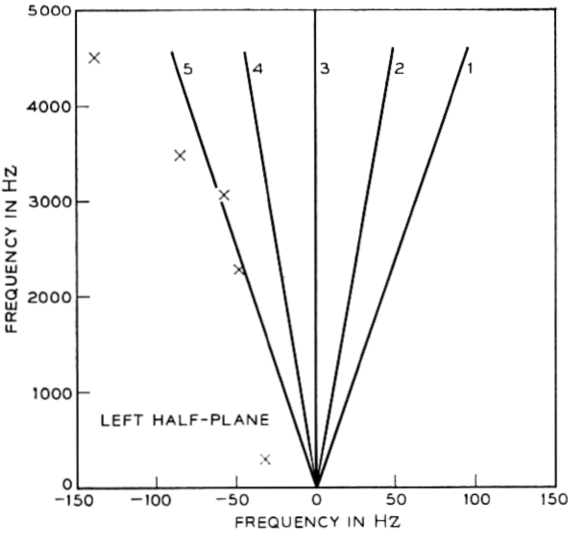

Fig. 6 — The s-plane locations of the poles of Fig. 5 and five contours for evaluation of the z-transform.

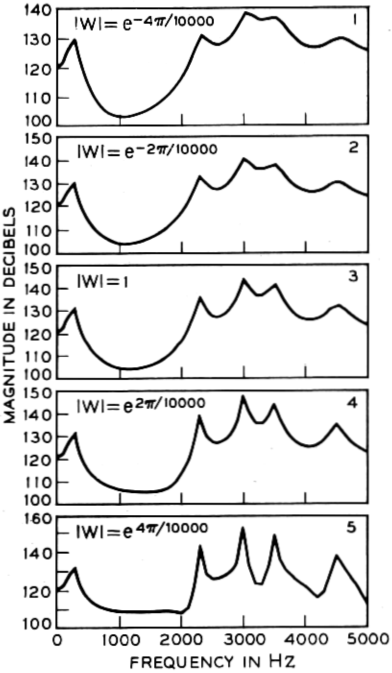

Fig. 7 — Magnitude curves corresponding to evaluation of the z-transform on the five contours of Fig. 6.

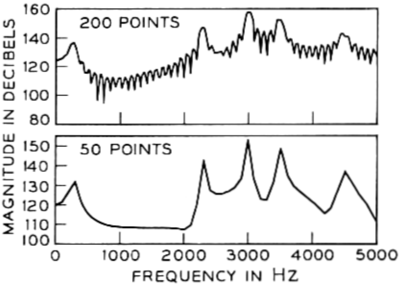

Fig. 8 — A comparison of high resolution and low resolution evaluations of the z-transform. The spacing of points is 25 Hz in the upper curve and 100 Hz in the lower curve.

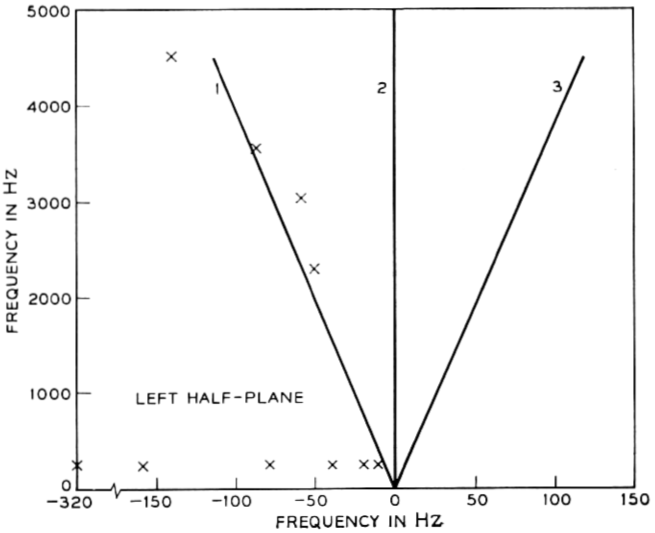

Fig. 9 — The pole locations and contours used to investigate the possibility of bandwidth determination using the chirp *z*-transform.

ample, the fifth resonance at 4,500 Hz is difficult to find in the upper plots and almost missing in the lower plots, because of the presence of a zero at the pole position. Furthermore, the frequency at which the magnitude is minimum, that is, the closest zero to contour 1 shifts from 2,500 to 2,700 to 800 to 1,100 Hz as bandwidth increases.

The plots in columns 2 and 3 show little or no variation from about 2,000 to 5,000 Hz where the appropriate contours are generally far away from the zeros of the distributions. The resonance at 4,500 Hz is always easy to locate on these plots, thus indicating the desirability of both detailed close-up examination of the transform (as on contour 1) and less detailed, further away looks at the magnitude curve (as on contours 2 and 3). The magnitude curves in the regions 0 to 2,000 Hz are still fairly sensitive to the exact zero distribution for contour 2, and slightly sensitive for contour 3. It would appear from Fig. 10 that there are cases when bandwidth can be determined either from the width or the magnitude of the resonance. Further investigation is necessary before quantitative techniques for determining bandwidths are available.

The choice of the optimum contours is highly dependent on the locations of the poles of the original system. In general there is no

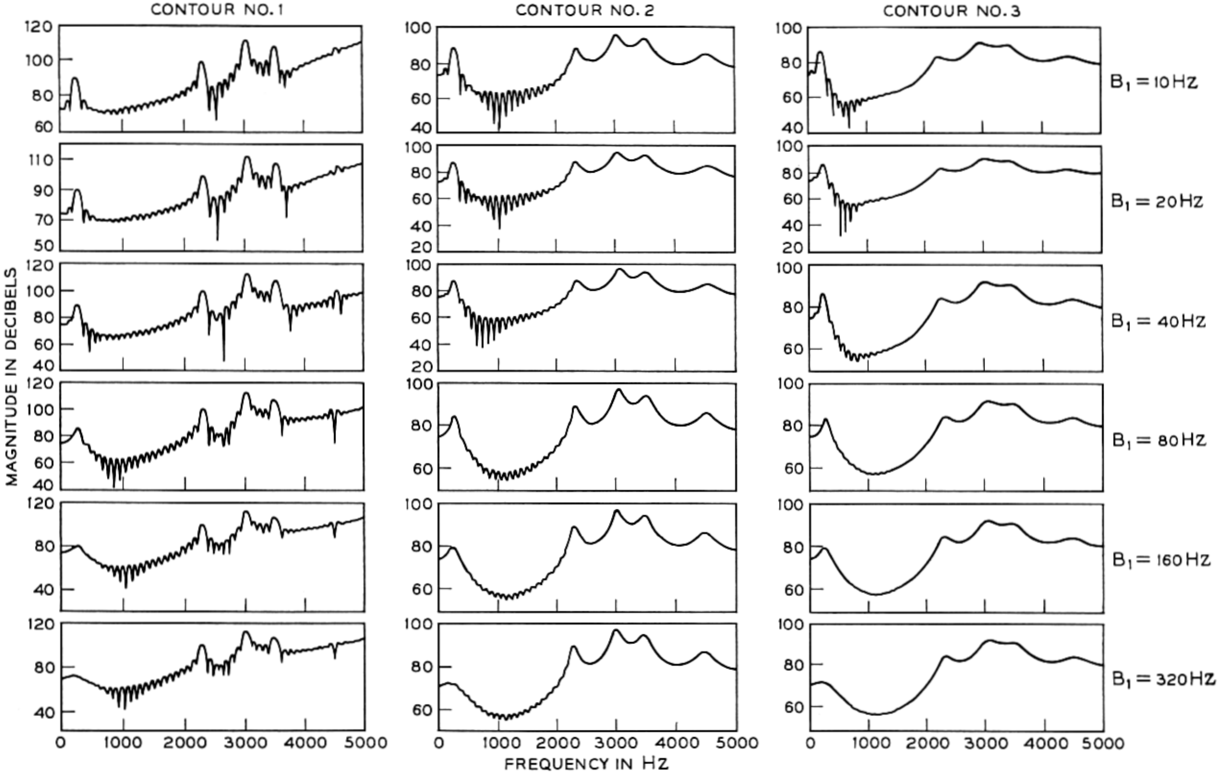

Fig. 10 — Magnitude curves corresponding to the 18 different situations of Fig. 9. The rows show magnitude curves for fixed bandwidths and the columns show magnitude curves for fixed contours.

single contour on which all the poles are located since these contours are essentially lines of constant $Q =$ (center frequency)/(bandwidth). Hence the choice of contour is highly dependent on which of the system poles is of most interest in the particular problem. However, some interesting observations can be made from studying magnitude curves for systems whose poles are constant Q poles. Such a system was simulated by keeping the pole center frequencies at the values used previously and setting the Q of each of the poles to 20. A 100 pulse per second impulse train was again used to excite the system and one period of steady state data was analyzed along contours 1 to 7 shown in Fig. 11. The pole positions of the system are shown in this figure and are seen to coincide with contour 6 exactly. The magnitude curves for these seven contours are shown in Fig. 12. These are high resolution spectra containing 801 points from 0 to 5,000 Hz. Notice that both magnitude curves 5 and 6 accentuate the poles equally except in the region of the fifth pole where curve 5 appears slightly better than curve 6. The fact that this occurs is not surprising in view of the fact that the pole is really manifested by an absence of a zero in an array of zeros of approximately the same magnitude and angular spacing.

Another anomaly which can be attributed to the way in which the

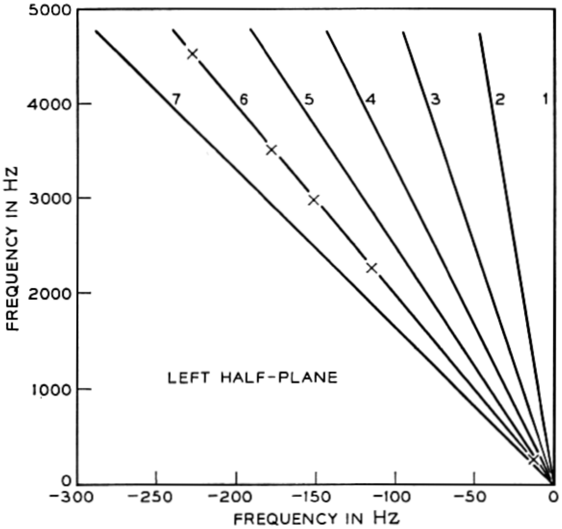

Fig. 11 — The s-plane locations of constant Q poles and contours on which the z-transform was evaluated.

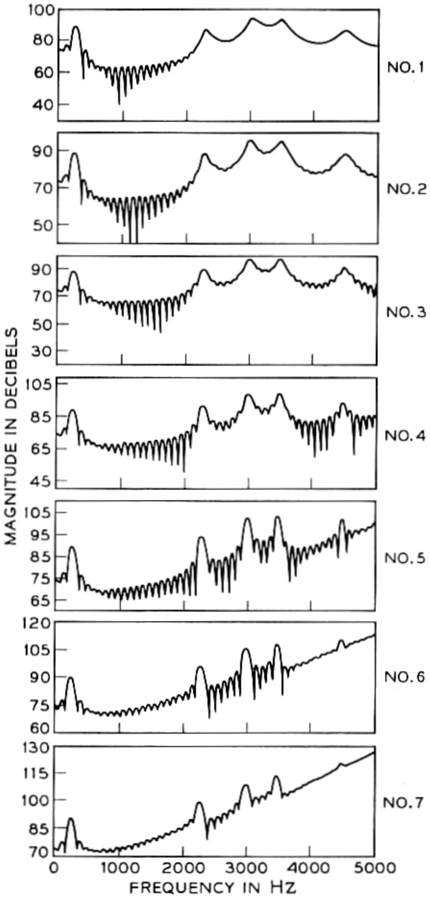

Fig. 12 — Magnitude curves for contours of Fig. 11 (constant Q poles).

zeros are distributed is evidenced by comparing curves 5 and 7. Based on the relative positions of the two contours with respect to the poles we would expect the magnitude curves for these contours to be identical, but the comparison shows that this is not the case in actual computation. This is the result of the fact that the zero distribution is not exactly symmetric so that contours which pass very close to the zeros may look considerably different from one another.

A final point of interest in Fig. 12 is the linear component in the last three curves which dominates at high frequencies. This effect is also shown in Fig. 14. Figure 13 shows the five contours used in obtaining the log magnitude plots in Fig. 14. It is clear that when the contour passes inside the original pole locations (and therefore inside the array of zeros in the z-plane), the log magnitude function exhibits a definite linear component. This effect is easily explained when $X(z)$ is written in the form

$$X(z) = Dz^{-(N-1)} \prod_{r=1}^{N-1} (1 - a_r^{-1}z). \tag{19}$$

where the a_r's are the zeros of $X(z)$. If we evaluate equation (19) at $z = W_o^{-k} \exp(-j2\pi\varphi_o k)$ we obtain for the magnitude

$$| X_k | = | D | W_o^{k(N-1)} \prod_{r=1}^{N-1} |[1 - a_r^{-1}W_o^{-k} \exp(-j2\pi\varphi_o k)]|. \tag{20}$$

For plotting in dB we define

$$\begin{aligned} 20 \log_{10} | X_k | &= 20 \log_{10} | D | + 20(N-1)k \log_{10} W_o \\ &\quad + \sum_{r=1}^{N-1} 20 \log_{10} |[1 - a_r^{-1}W_o^{-k} \exp(-j2\pi\varphi_o k)]|. \end{aligned} \tag{21}$$

In the examples we have shown, almost all of the zeros a_r have magnitudes slightly less than 1. Thus for contours inside these zeros (cor-

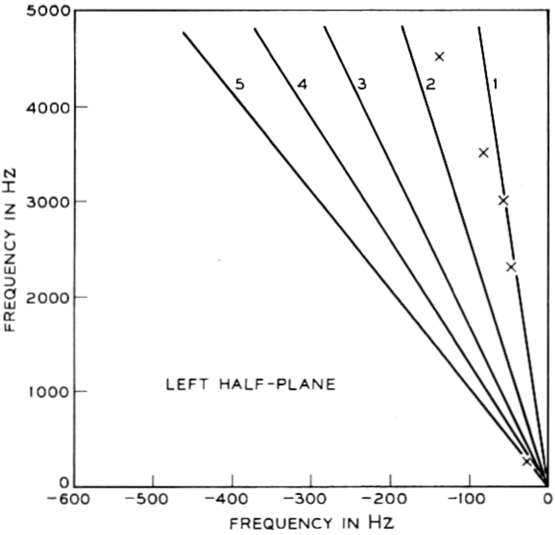

Fig. 13 — Contours and pole locations used to study the effect of passing inside the pole locations.

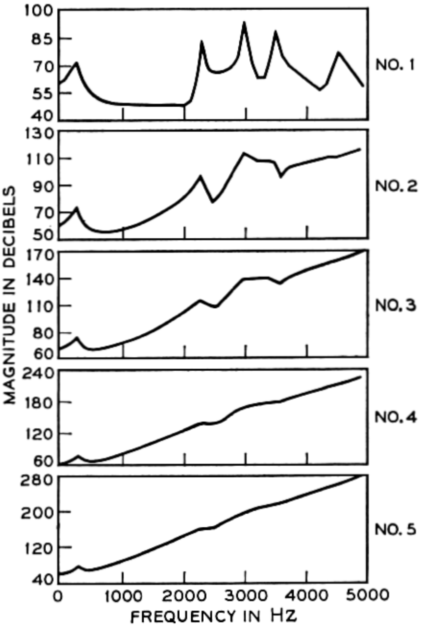

Fig. 14 — Magnitude curves for the contours of Fig. 13 showing a large linear component resulting from the $N - 1$ poles of $X(z)$ at $z = 0$.

responding to contours 2 through 5 in Fig. 13), W_o is greater than 1. Thus each term in the sum on the right side of equation (21) tends to decrease as k gets larger. In contrast, the second term on the right side of equation (21) represents a linear component with slope equal to $20\ (N-1)\log_{10} W_o$.

In Fig. 14, the mth curve corresponds to a value of $W_o = e^{4\pi m/10{,}000}$, $m = 1, 2, 3, 4, 5$. The value of φ_o is $-1/100$, N is 100, and the sampling rate is 10 kHz. Thus the frequency going from 0 to 5 kHz corresponds to k going from 0 to 50. For example, in the fifth curve $W_o = e^{20\pi/10{,}000}$ and the slope should be

$$20(N - 1) \log_{10} W_o = \frac{20(99)20\pi \log_{10} e}{10{,}000} = 5.4. \tag{22}$$

Thus the total dB change in going from 0 to 5 kHz should be on the order of $50(5.4) = 270$ dB. In Fig. 15a we show this case again. In Fig. 15b we show the result of evaluating

$$\sum_{n=-(N-1)}^{0} x_{n+N-1} z^{-n} = z^{(N-1)} X(z), \tag{23}$$

using the same value of W_o and φ_o. Notice that this should remove the second term in equation (21) leaving the other terms unaffected. This observation is substantiated by Fig. 15b since the value at 5 kHz is very nearly 270 dB less than in Fig. 15a. Notice also that some of the resonances are still in evidence although not as clearly defined because the contour is passed relatively far inside the zeros of $X(z)$.

Another interesting question was investigated using this technique. As shown above, when the response of a linear system is truncated by repetitively pulsing the system and transforming a finite number of samples, the z-transform has only zeros except for poles at $z = 0$. However, the poles of the original system function can still be located in magnitude response curves by the absence of zeros in the appropriate regions. The question arises about what happens when the system function contains zeros. Suppose $h(nT)$ is the impulse response of a linear system with both poles and zeros in its z-transform $H(z)$. Since $H(z)$ has poles, $h(nT)$ will be an infinite sequence. There is clearly no reason to expect that the transform of only N of these samples will have zeros at the same location as the zeros of $H(z)$. However, the system zeros can be expected to have an effect on the distribution of zeros of the truncated z-transform.

To illustrate this point, the system response used in the above examples was modified by passing the output waveform through a system whose transfer function consisted of a complex conjugate zero pair. A periodic 100 pulse per second source was again used to excite the system and one period of steady state data was analyzed. The system pole-zero pattern and the contours of analysis are shown in

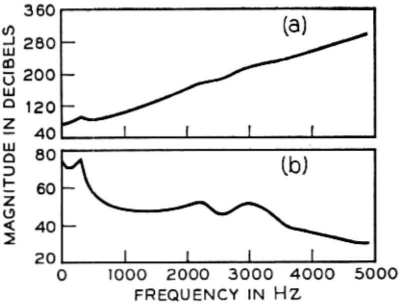

Fig. 15 — Magnitude curves obtained by evaluation of the z-transform on contour 5 of Fig. 13: (a) with the effect of the $N - 1$ poles at $z = 0$; (b) with the $N - 1$ poles removed by shifting the sequence x_n by N positions to the left.

Fig. 16. In one simulation a zero was placed at point *A* (500 Hz, 12.5 Hz), and in a second simulation the zero was at point *B*, (2,500 Hz, 60 Hz). The analysis was made at 801 points from 0 to 5,000 Hz along these contours. The resulting magnitude curves, along with a set of low resolution curves where the magnitude was computed every 100 Hz from 0 to 5,000 Hz, are shown in Figs. 17 and 18. The data of Fig. 17 are for the case where the transmission zero was at 500 Hz whereas for Fig. 18 the zero was at 2,500 Hz.

The high resolution data of Fig. 17 show no strong indication of the transmission zero; whereas the transmission poles are still very much in evidence. The low resolution data (evaluated at harmonics of the source) does indicate the presence of a zero along contour 1, but along the other contours the case is not so clear. The most unusual observation is that along contour 3, the contour closest to both the transmission zero and the poles, there is little or no indication of the zero; whereas the poles are still strongly in evidence. Along contour 4, at the high frequency end, there is noise in the magnitude spectrum. The source of this noise is discussed in Section V.

The indications from Fig. 17 are that a transmission zero can be more easily located on contours which are far from the zero than on

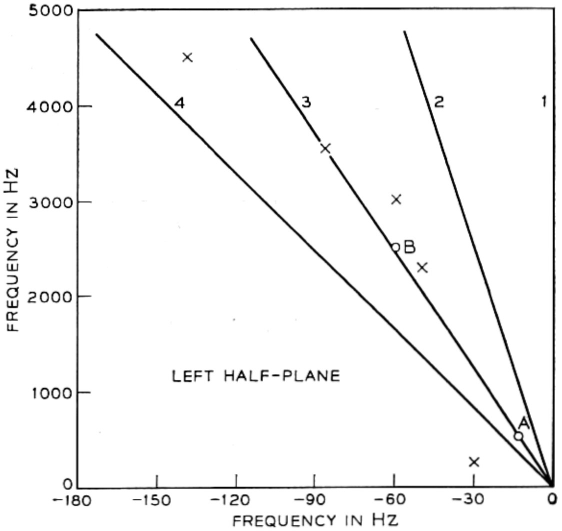

Fig. 16 — The *s*-plane locations of poles and zeros (at *A* and *B*) and contours used in studying the effect of zeroes on the magnitude curves.

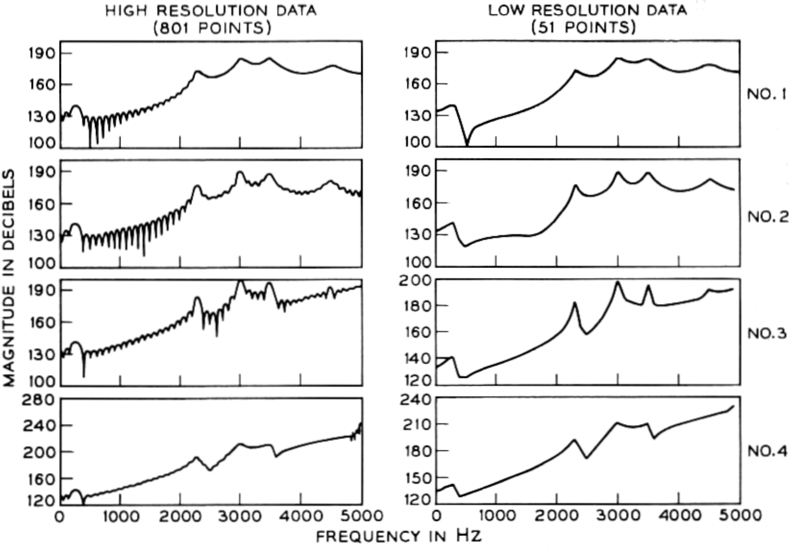

Fig. 17 — Magnitude curves for a zero at 500 Hz (position A in Fig. 16).

contours which traverse it. Furthermore it is much easier to locate on a low resolution spectrum than on a high resolution spectrum. Hence zeros, unlike poles, are not generally easy to locate from spectra.

The zero of Fig. 17 was at 500 Hz and in a region where the high resolution spectra displayed a large amount of ripple from the truncation zeros of the data. Figure 18 shows similar magnitude curves for the zero at 2,500 Hz, a region with much less ripple in the spectrum. The magnitude response curves show effects entirely similar to those of Fig. 17. The zero is most easily locatable for contour 1, the standard fast Fourier transform. In contour 3, which again passes through the zero, there is no indication of the zero. Also the low resolution data tends to show the zero better than the high resolution data. One important implication of these results is that one could not use these techniques to accurately find the position of complex transmission zeros. In many cases it would be difficult to differentiate between dips in the spectrum between poles and dips caused by complex zeros, thus indicating the difficulty of locating even the center frequency of a zero.

The chirp z-transform algorithm has been applied to the spectral analysis of speech in order to aid in automatic detection of the time varying resonances (poles or formants) of speech. Voiced speech can

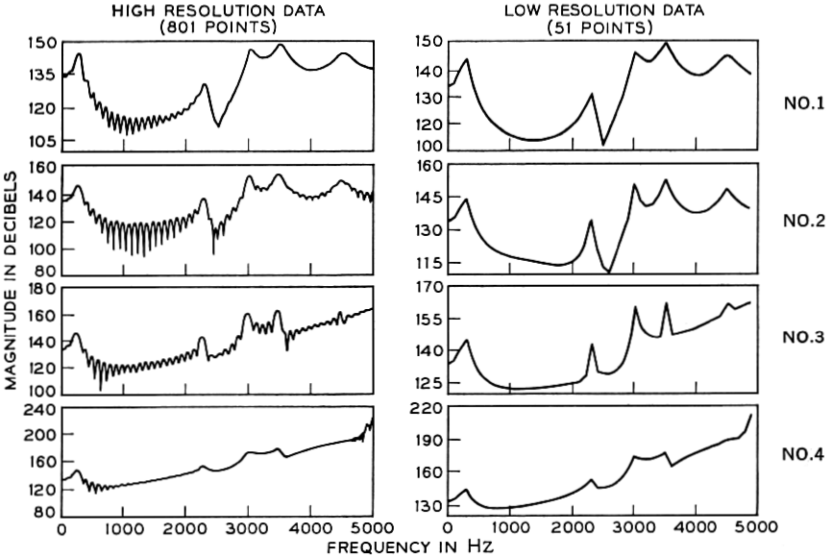

Fig. 18 — Magnitude curves for a zero at 2,500 Hz (position *B* in Fig. 16).

be modelled as the convolution of a source waveform with a vocal tract impulse response. The vocal tract impulse response is essentially a sum of damped exponentials, each exponential corresponding to a mode or pole of the vocal tract transfer function. It is of interest to speech researchers to detect these time varying resonances. The chirp z-transform algorithm has been applied to individual periods of voiced speech with a high degree of success. Figure 19 shows the result of applying the chirp z-transform algorithm along the two contours shown at the upper left of the figure, to a period of voiced speech. The upper contour corresponds to the standard fast Fourier transform contour; the lower to a suitably chosen spiral contour. The magnitude function along the upper contour indicates a single wide peak in the region 2,000 to 2,500 Hz, whereas the magnitude along the lower contour shows two isolated peaks in this region corresponding to the physical knowledge that there actually are supposed to be two peaks in this region. Variations on the chirp z-transform algorithm for spectral analyses of speech have been studied and will be reported on in a subsequent paper.[7]

4.2 *High Resolution, Narrow Band Frequency Analysis*

One very useful application of the chirp z-transform algorithm is the ability to efficiently evaluate high resolution, narrow frequency band spectra. Using standard fast Fourier transform techniques, in order to achieve a frequency resolution of $\leqq \Delta F$, with a sampling frequency of the data of $1/T$, requires $N \geqq 1/(T\cdot\Delta F)$ points. For very small ΔF, this implies very large values of N. The crucial issue is that what is often required is high resolution for a limited range of frequencies and low resolution for the remainder of the spectrum. An example of such a circumstance is the design of band-pass or low-pass filters. Usually what is desired is a microscopic look at details of the frequency response in the pass-band and only a gross look outside the pass-band.

The chirp z-transform algorithm is extremely well suited for such cases since it allows selection of initial frequency and frequency spacing, independent of the number of time samples. Hence high resolution data over a narrow frequency range can be attained at low cost.

To illustrate these points, simple rectangular band-pass filters were simulated by symmetrically truncating a delayed impulse response. The impulse response used was

$$h(nT) = \alpha \sin\left[\pi(F_2 - F_1)(n - \tfrac{1}{2} - m)T\right] \cdot \cos\left[\pi(F_2 + F_1)(n - \tfrac{1}{2} - m)T\right], \qquad 0 \leqq n \leqq 2m \tag{24}$$

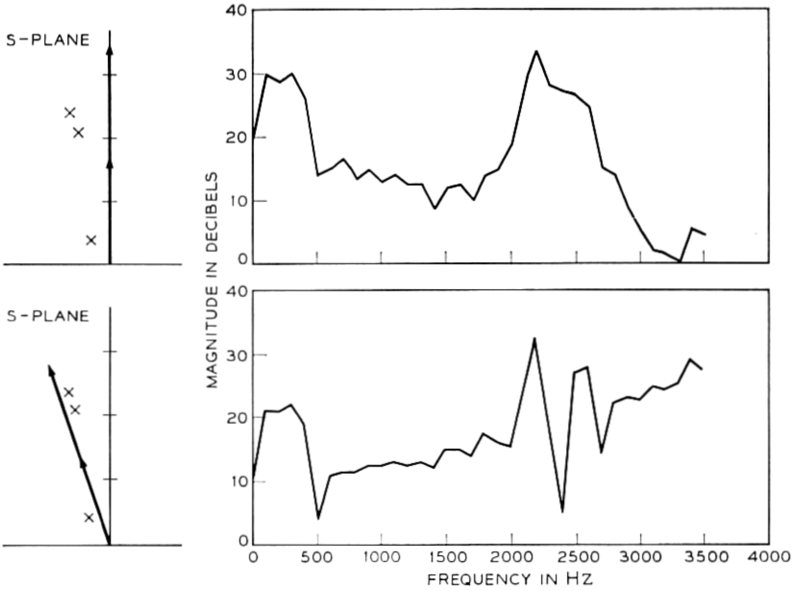

Fig. 19 — Magnitude curves from evaluation of the z-transform of one period of natural speech. The contour for the upper plot is the unit circle in the z-plane while the contour for the lower curve is a spiral inside the unit circle.

where

$2m$ = number of terms in the truncated impulse response
$1/T$ = sampling frequency = 10,000 Hz
F_1 = lower cutoff frequency in Hz
F_2 = upper cutoff frequency in Hz.

Values for m of 100 and 500 were used with $F_1 = 900$ Hz and $F_2 = 1{,}100$ Hz. Figure 20 shows plots of equation (24) for these two cases. A standard 1,600 point fast Fourier transform was calculated and the magnitude response for $m = 100$ is shown in the upper half of Fig. 21. In order to investigate the pass-band and transition region more carefully the chirp z-transform algorithm was used to give a 1.25 Hz resolution over the band from 500 to 1,500 Hz. The contour used was identical to the contour for the fast Fourier transform. The resulting magnitude response curve is shown in the lower half of Fig. 21. To achieve this high a resolution would have required an 8,000 point fast Fourier transform, instead of the 1,000 point transforms actually used. (Similar expansions of regions of the phase curve were made for this filter but are not shown.)

Figure 22 shows similar effects for the case $m = 500$. The applicability of the chirp z-transform algorithm for such frequency expansions is a powerful tool for close examination of small frequency bands, as well as for debugging implementations of digital filters. For example, one could easily check if a desired filter met its design specification of in-band ripple, transition ratio, and so on.[8]

One situation where the chirp z-transform algorithm may be quite useful is when we are confronted with an extremely long sequence for which we desire a fine grained spectrum over a narrow band of frequencies. Suppose we have a sequence of P samples and desire M spectral samples where $M \ll P$. That is, we wish to evaluate

$$X_k = \sum_{n=0}^{P-1} x_n A^{-n} W^{nk}, \qquad k = 0, 1, \cdots, M-1. \tag{25}$$

The sum in equation (25) can be broken up into r sums over N points as follows

$$X_k = \sum_{q=0}^{r-1} A^{-qN} W^{kqN} \left[\sum_{n=0}^{N-1} x_{n+qN} A^{-n} W^{nk} \right], \qquad k = 0, 1, \cdots, M-1 \tag{26}$$

where $rN \geqq P$. Each of the r sums in the brackets can be evaluated using the chirp z-transform algorithm, requiring storage on the order of

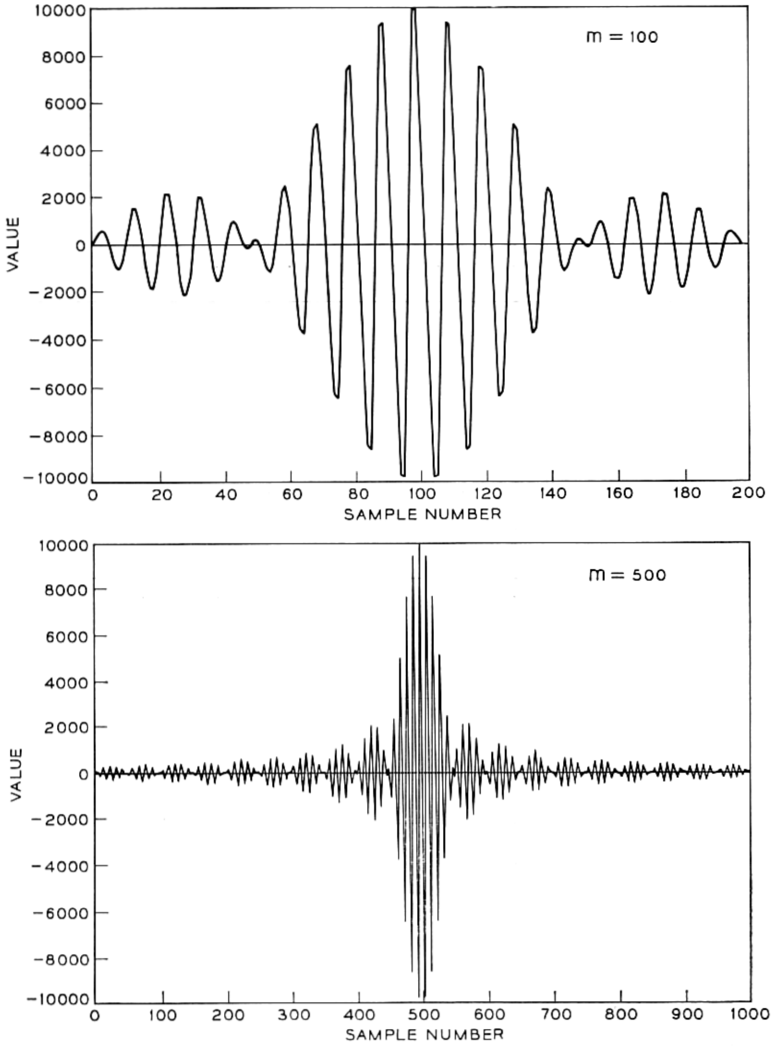

Fig. 20 — The impulse responses of simple band-pass filters.

$3(N + M - 1)$ locations. In addition we require $2M$ locations in which to accumulate the M complex values of the transform. Although 2 fast Fourier transforms and $2M$ complex multiplications are required for each of the r transforms, it is quite possible that a saving in total time may result from this method as opposed to evaluation of a P point transform using auxiliary storage such as drum, disk or tape.

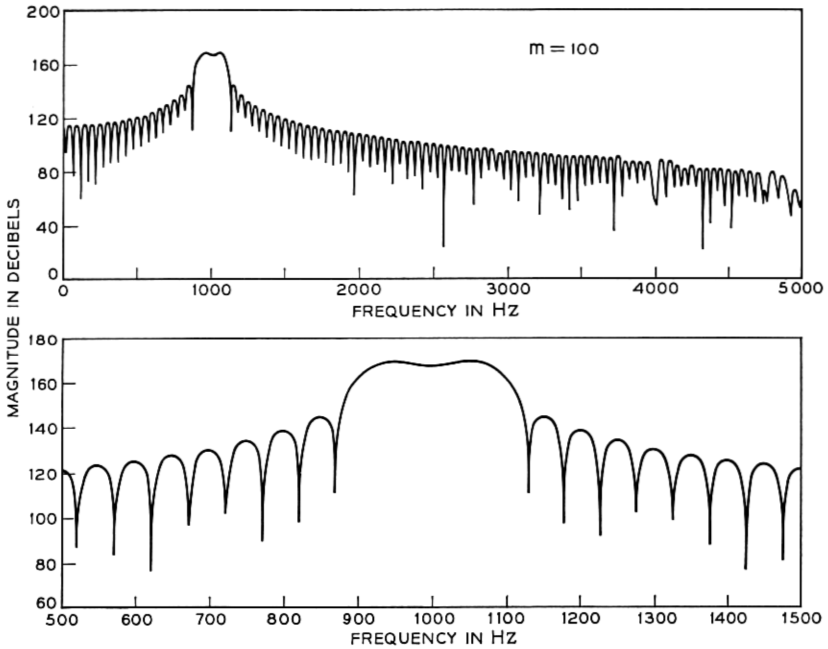

Fig. 21 — Frequency response curves for upper impulse response (200 samples) in Fig. 20. Upper curve obtained with 1,600 point fast Fourier transform (resolution 6.25 Hz). Lower curve obtained with chirp z-transform algorithm (1.25 Hz resolution).

4.3 *Time Interpolation or Sampling Rate Changing*

The flexibility of the chirp z-transform algorithm for obtaining high resolution in frequency has been explained and illustrated in Section 4.2. A similar procedure applies to interpolation between samples of a bandlimited time function using samples of the frequency spectrum.[9] In this section, we discuss how the discrete Fourier transform can be used to perform interpolation on a set of samples and the advantages and disadvantages of using the chirp z-transform algorithm for this.

4.3.1 *Bandlimited Interpolation Using the discrete Fourier transform*

Assume that we have available N samples $x(nT)$, $n = 0, 1, 2, \ldots,$ $N - 1$, of a bandlimited waveform $x(t)$. The sampling interval T is assumed less than or equal to the Nyquist interval. The total time interval spanned by these samples is therefore NT seconds. We wish to obtain equally spaced samples of $x(t)$ at a sampling interval T', where

T' is less than or equal to the Nyquist interval. These samples are denoted by $x(mT')$, $m = 0, 1, \ldots, N' - 1$, where $N'T' = NT$. (Notice that we are assuming N' is an integer. This assumption will be dropped later.)

If all the samples $x(nT)$ are available, the samples $x(mT')$ can be obtained from

$$x(mT') = \sum_{n=-\infty}^{\infty} x(nT) \frac{\sin \frac{\pi}{T}(mT' - nT)}{\frac{\pi}{T}(mT' - nT)}. \tag{27}$$

Thus the interpolation can be viewed as the result of convolving the interpolation function $[\sin(\pi t/T)]/[(\pi/T)\,t]$ with the samples $x(nT)$ and then resampling with period T'. It is well known that convolution may be done using the discrete Fourier transform, and we will show how the resampling can also be affected by properly augmenting the

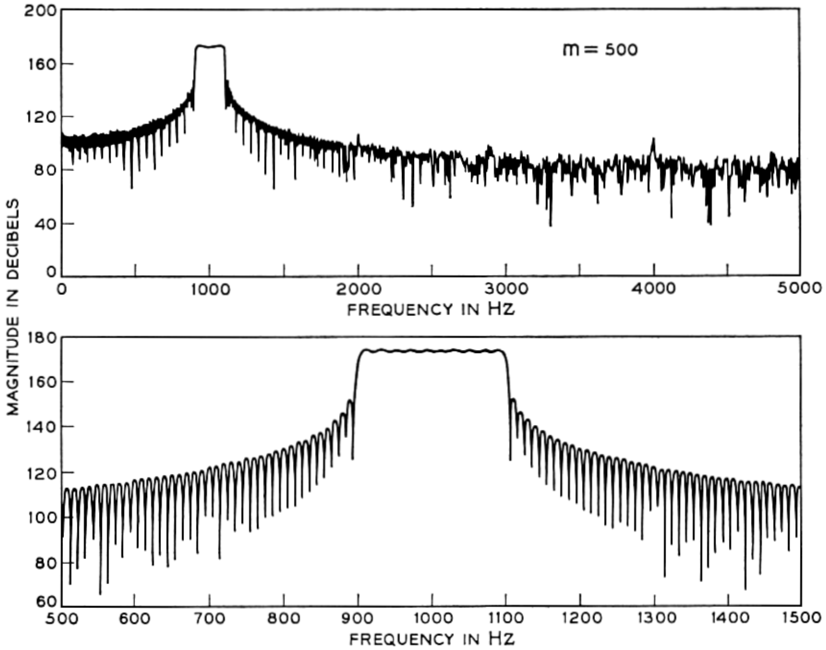

Fig. 22 — Frequency response curves for lower impulse response (1,000 samples) of Fig. 20. Upper curve obtained with 1,600 point fast Fourier transform (resolution 6.25 Hz). Lower curve obtained with chirp z-transform algorithm (1.25 Hz resolution).

transform with zeros. Because the discrete Fourier transform uses only a finite number of samples, we shall encounter errors similar (but not identical) to using only N terms in equation (27).

The discrete Fourier transform of the given samples is

$$X_N(k) = \sum_{n=0}^{N-1} x(nT) \exp\left(-j\frac{2\pi}{NT}knT\right), \qquad k = 0, 1, \cdots, N-1. \tag{28}$$

(Notice that we have changed notation in this section in order to make explicit the number of samples and the sampling period.) We define

$$X'_N(k) = X_N(k)H_N(k), \tag{29}$$

where $H_N(k)$ is the N point discrete Fourier transform of the interpolation function to be convolved with the samples $x(nT)$. [Notice that this convolution is equivalent to cyclic convolution of a periodic impulse response $h(nT)$ with the samples $x(nT)$.]

In order to change the sampling to period T', we split $X'_N(k)$ about $k = N/2$ and expand (by inserting zeros) or contract (by discarding zeros) the transform according to the following equations

$$X'_{N'}(k) = X'_N(k) \qquad \begin{cases} 0 \leqq k < N'/2 & N' < N \\ 0 \leqq k < N/2 & N' > N \end{cases} \tag{30a}$$

$$= 0 \qquad k = N'/2 \qquad N' < N \tag{30b}$$

$$= \tfrac{1}{2}X'_N(k) \qquad k = N/2 \qquad N' > N \tag{30c}$$

$$= 0 \qquad N/2 < k < N' - N/2 \qquad N' > N \tag{30d}$$

$$= \tfrac{1}{2}X'_N(k - N' + N) \qquad k = N' - N/2 \qquad N' < N \tag{30e}$$

$$= X'_N(k - N' + N) \begin{cases} N'/2 < k < N' & N' < N \\ N' - N/2 < k < N' & N' > N \end{cases}. \tag{30f}$$

Equations (30b), (30c), and (30e) are required only when N' and N are even integers and equation (30d) is required only when $N' > N$.

The N' point inverse discrete Fourier transform of $X'_{N'}(k)$ is defined to be

$$x'(mT') = \frac{1}{N'}\sum_{k=0}^{N'-1} X'_{N'}(k) \exp\left(j\frac{2\pi}{N'T'}kmT'\right), \qquad m = 0, 1, \cdots, N'-1. \tag{31}$$

For example, if N is even and $N' > N$, we can show using equations (28), (29), and (30) that

$$x'(mT') = \frac{1}{N'} \sum_{k=-N/2}^{N/2} H_N(k) \left[\sum_{n=0}^{N-1} x(n) \exp\left(-j\frac{2\pi}{NT} knT\right)\right] \cdot \exp\left(j \frac{2\pi}{N'T'} kmT'\right), \tag{32}$$

where $m = 0, 1, \cdots, N' - 1$, and the terms corresponding to $k = \pm N/2$ are understood to be multiplied by $1/2$ since N is even. By interchanging the order of summation and using the fact that $N'T' = NT$, we obtain

$$x'(mT') = \frac{N}{N'} \sum_{n=0}^{N-1} x(nT)h(mT' - nT) \tag{33}$$

where

$$h(mT' - nT) = \frac{1}{N} \sum_{k=-N/2}^{N/2} H_N(k) \exp\left[j \frac{2\pi}{NT} k(mT' - nT)\right] \cdot \tag{34}$$

Notice that equation (33) has the desired form for interpolation [see equation (27)], however the values of $x'(mT')$ are clearly not exactly equal to the desired interpolated values $x(mT')$. This is so because only N samples are used and because of the form of $h(mT - nT)$. As an example, suppose

$$H_N(k) = \begin{cases} 1 & -N/2 < k < N/2. \\ \frac{1}{2} & k = \pm N/2 \end{cases} \tag{35}$$

(This is equivalent to splitting $X_N(k)$ at $k = N/2$ and inserting $N' - N$ zeros between the two halves of the transform). If we evaluate equation (34) for this case, we obtain

$$h(mT' - nT) = \frac{\sin \frac{\pi}{T}(mT' - nT)}{N \tan \frac{\pi}{NT}(mT' - nT)} \cdot \tag{36}$$

This function is plotted in Fig. 23a where $\theta = (mT' - nT)/T$, and $N = 8$. Clearly

$$h(mT' - nT) \approx \frac{\sin \frac{\pi}{T}(mT' - nT)}{\frac{\pi}{T}(mT' - nT)} \tag{37}$$

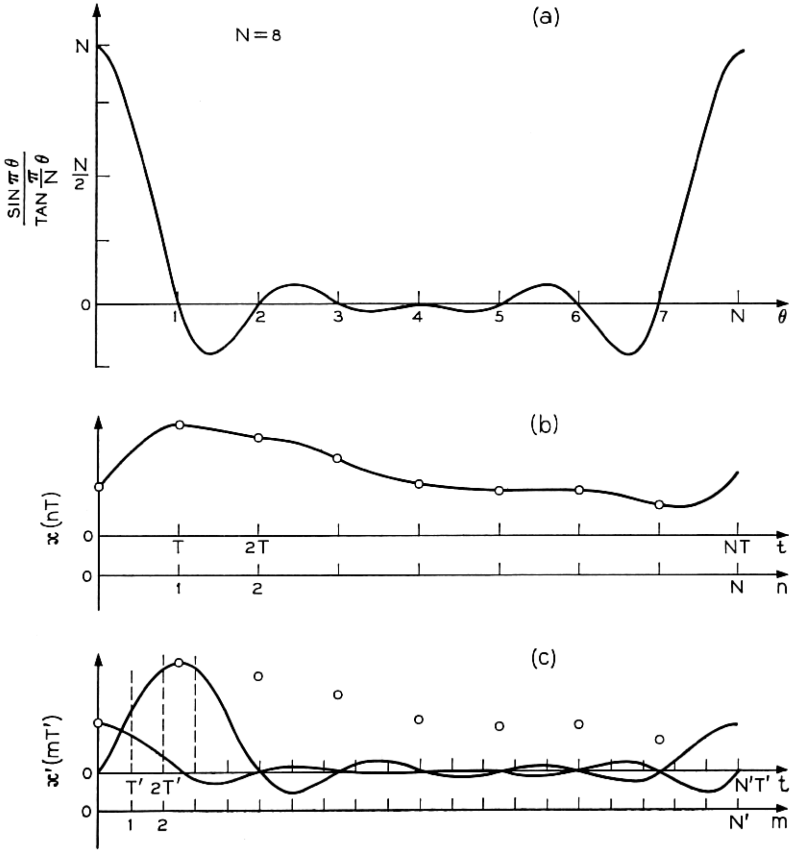

Fig. 23 — An illustration of bandlimited interpolation using the discrete Fourier transform: (a) the periodic function which is convolved with the original samples; (b) a bandlimited time function showing samples with spacing T; (c) How the interpolated values $x'(mT')$ are formed.

when $\pi(mT' - nT)/NT$ is small, that is, in a region where $h(mT' - nT)$ is significantly different from zero. Figure 23b shows a segment of a waveform $x(t)$ and samples $x(nT)$. In Fig. 23c, we have shown just two of the terms in equation (33). This figure places in evidence the nature of the interpolation which is performed. The errors are likely to be greatest at either end of the segment, since the interpolated values at one end depend on the samples at the other end in a way which is not at all consistent with equation (27). The error caused by this effect will be most significant in the regions $0 \leqq mT' < 2T$ and

$T(N - 2) \leqq mT' \leqq TN$. The remainder of the values will have essentially the same error associated with using only N terms in equation (27).

Notice that equation (36) is not the only interpolation function which can be used. Other choices of $H_N(k)$ may lead to interpolation functions which are in some sense more desirable. For example, Fig. 24 shows four different choices for $H_N(k)$ and their associated interpolation functions or impulse responses. (The impulse responses were shifted modulo N' as an aid in plotting.) It can be seen from Fig. 24, that removing the sharp cutoff in $H_N(k)$ greatly shortens the effective duration of the impulse response, thus tending to minimize the end effects discussed previously. Clearly the approximation to $(\sin \pi t)/\pi t$ interpolation is not as good as equation (36), but in many cases such smoothing of the interpolated values may not be objectionable.

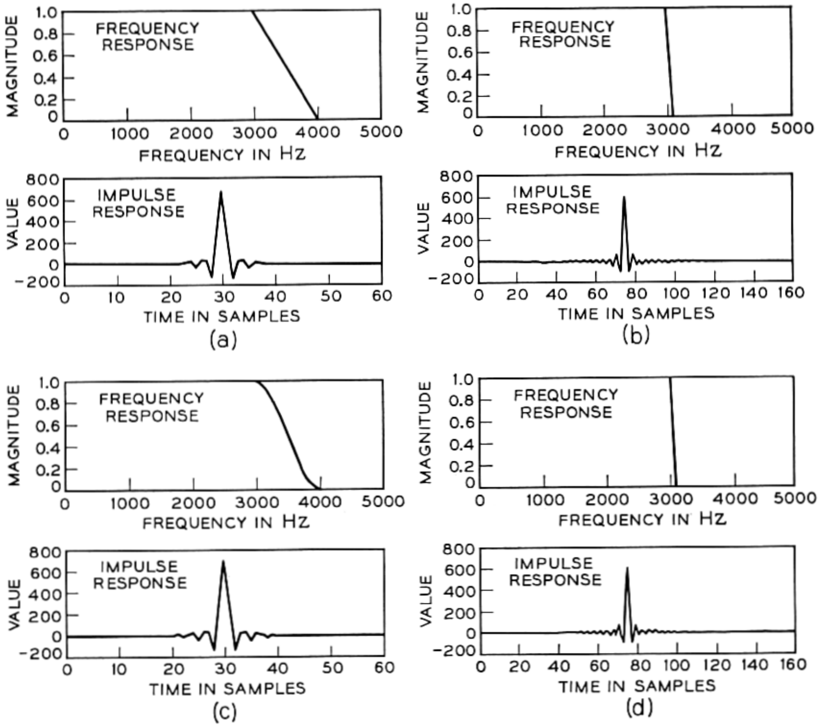

Fig. 24 — A set of four simple frequency responses and corresponding impulse responses which could be used for interpolation.

4.3.2 *Computational Considerations in Bandlimited Interpolation*

In Section 4.3.1 we discussed a method of bandlimited interpolation based on the discrete Fourier transform. The operations involved are summarized in Fig. 25. The sequence $\{X_N(k)\}$ may be evaluated using the fast Fourier transform. In this case, N will be restricted to a value compatible with the available fast Fourier transform routine, for example, N would be highly composite. The transform is then multipled by $H_N(k)$ and expanded or contracted according to equation (30). Then we must compute the inverse discrete Fourier transform with N' points. This can be done using the fast Fourier transform provided that

(*i*) $N' = NT/T'$ is an integer and compatible with the available fast Fourier transform routine.

(*ii*) Enough high speed storage (a minimum of N' locations) is available.

[Notice that *i* applies for either $N' > N$ or $N' < N$ while *ii* will probably not be a problem except when $N' \gg N$.]

In many cases it may not be possible to meet one or both of the above conditions; then the chirp z-transform algorithm can be very useful. The N' point inverse discrete Fourier transform may be computed using $W_o = 1$ and $\varphi_o = +\ 1/N'$, where N' need not even be an integer. Thus we can compute M interpolated values using

$$x'(mT) = \frac{1}{N'} W^{m^2/2} \sum_{k=-N/2}^{N/2} [X'_{N'}(k) W^{k^2/2}] W^{-(m-k)^2/2} \tag{38}$$

where $X'_{N'}(k)$ is determined by equation (30) and

$$X'_{N'}(-k) = X'_{N'}(N' - k), \qquad k = 1, 2, \cdots, N/2. \tag{39}$$

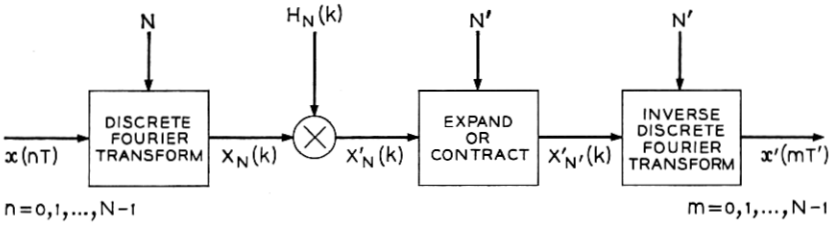

Fig. 25 — Illustration of the steps involved in bandlimited interpolation using the discrete Fourier transform.

Assuming that the transform of $W^{-k^2/2}$ is available, equation (38) can be evaluated using two L point fast Fourier transforms where L is the smallest integer which is greater than $N + M - 1$ and which is compatible with an available fast Fourier transform routine.

Alternatively we can evaluate

$$y'(mT') = \frac{1}{N'} W^{m^2/2} \sum_{n=0}^{N/2} [Y'_{N'}(k) W^{k^2/2}] W^{-(m-k)^2/2} \tag{40}$$

where

$$Y'_{N'}(k) = \begin{cases} X'_{N'}(k) & k = 0 \\ 2X'_{N'}(k) & 0 < k \leqq N/2 \end{cases} . \tag{41}$$

It can be shown that

$$y'(mT') = x'(mT') - j\hat{x}'(mT') \tag{42}$$

where $\hat{x}'(mT')$ is the inverse discrete Fourier transform of

$$\hat{X}'_{N'}(k) = j \operatorname{sgn}_{N'} (k) \cdot X'_{N'}(k) \tag{43}$$

and

$$\operatorname{sgn}_{N'} (k) = \begin{cases} 0 & k = 0, N'/2 \\ 1 & 0 < k < N'/2. \\ -1 & N'/2 < k < N' - 1 \end{cases} \tag{44}$$

From equation (41) and (44) it can be shown that $\hat{x}'(mT')$ is an approximation to the Hilbert transform of $x'(mT)$. In this case we require at least $(N/2 + M - 1)$ point transforms to compute M interpolated values. This is at the expense of not being able to do two interpolations at once as is possible with equation (38) (obtaining one interpolation as a real output and one as an imaginary output); however we do obtain an approximation to the Hilbert transform of $x'(mT')$ which may be of value in some applications.

If sufficient core storage is not available to compute an N' point fast Fourier transform, we can compute the interpolated values in sections and piece these sections together as is commonly done in high speed convolution. The chirp z-transform algorithm allows us to compute as many as $2M$ interpolated points at a time, where M can be chosen so that the fast Fourier transforms can be done using only core storage. Probably the most significant advantage, though, is the ability to efficiently interpolate to arbitrary sampling intervals.

As an example of the ideas discussed in this section, consider the waveforms shown in Fig. 26. Figure 26a shows 500 samples of a speech waveform where the sampling rate was 20 kHz. ($T = 5 \times 10^{-5}$ second). The samples are connected by straight lines in the figure. Figure 26b shows the 500 samples of the waveform in (a) after filtering with a nonrecursive filter of the type shown in Fig. 24, whose gain was zero after 3.2 kHz. Figure 26c shows 160 samples of the result of a change of sampling rate from 20 kHz to 6.4 kHz. The value of N was 700 and $N' = (6{,}400)(700)/20{,}000 = 224$. It is difficult to judge quantitatively from such a figure the accuracy of the interpolation. It does seem safe to conclude that the error is not extreme. Our experience has been that there is significant error only in the first and last few samples of the N' output samples. Using the chirp z-transform algorithm, these "bad" samples need not ever be computed, for example, only M "good" values need be computed.

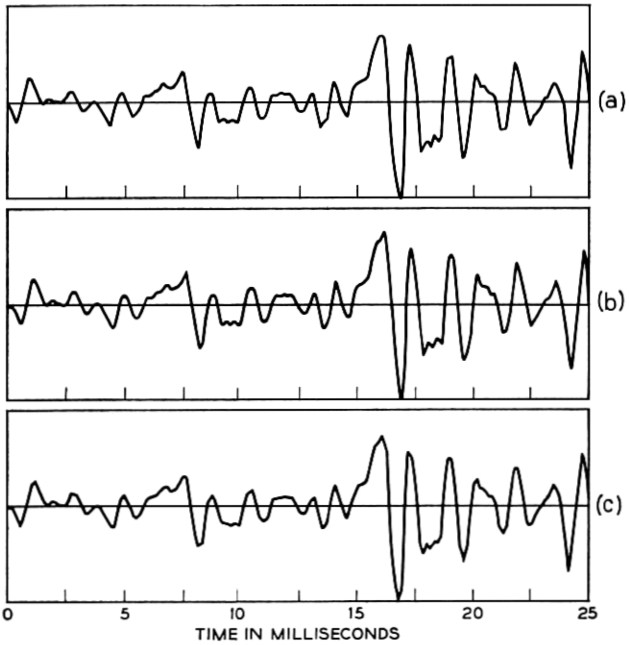

Fig. 26 — An example of interpolation for the purpose of changing the sampling rate: (a) 500 samples of speech at 20 kHz sampling rate; (b) 500 samples of (a) after low pass filtering to 3.2 kHz; (c) 160 samples of (a) after changing the sampling rate to 6.4 kHz using the chirp z-transform. (In all cases the samples are connected by straight lines).

If one wishes to low-pass filter a waveform and then go to a lower sampling rate, the filtering and interpolation can be combined if we use a nonrecursive filter. That is, the discrete Fourier transform of the filter impulse response can be simply combined with $H_N(k)$.

V. LIMITATIONS

Several times we have pointed out shortcomings of the chirp z-transform algorithm. One limitation in using it to evaluate the z-transform off the unit circle stems from the fact that we may be required to compute $W_o^{\pm n^2/2}$ for large n. If W_o differs very much from 1.0, $W_o^{\pm n^2/2}$ can become very large or very small when n becomes large. (We require a large n when either M or N become large, since we need to evaluate $W^{n^2/2}$ for n in the range $-N < n < M$.) For example, if $W_o = e^{-2{\cdot}5/10,000} \approx 0.999749$, and $n = 1,000$, $W_o^{\pm n^2/2} = e^{\pm 125}$ which exceeds the single precision floating point capability of most computers by a large amount. Hence the tails of the functions $W^{\pm n^2/2}$ can be greatly in error, thus causing the tails of the convolution (the high frequency terms) to be grossly inaccurate. The low frequency terms of the convolution will also be slightly in error, but these errors generally are negligible.

An example of this effect is shown in Fig. 27. The contour for the five curves in this figure was held fixed (contour 5 in Fig. 6) and the number of frequency points in the range 0 to 5,000 Hz was increased in steps of 2 from 50 to 800. Spectral samples are plotted every 100 Hz for comparison. (This example was programmed using single-precision floating-point arithmetic on a GE 635 computer with a 36 bit word length.) It is seen that as the number of output points increases, errors in the high frequency region become large and completely mask the fifth resonance for the 800 point case. The effects of the inaccuracy in $W^{\pm n^2/2}$ can also be seen at low frequencies. For example, the spectral magnitude at 0 Hz goes from about 120 dB to 134 dB as the number of points goes from 50 to 800. These small errors generally do not affect the gross spectral characteristics as seen in Fig. 27. The resonances are easy to locate in all cases until the errors get exceedingly large. One can push the maximum point limit higher than 800 (in this case) by using double precision arithmetic.

The limitation of contour distance in or out from the unit circle is again the result of computation of $W^{\pm n^2/2}$. As W_o deviates significantly from 1.0, the number of points for which $W^{\pm n^2/2}$ can be accurately computed decreases. It is of importance to stress, however, that for

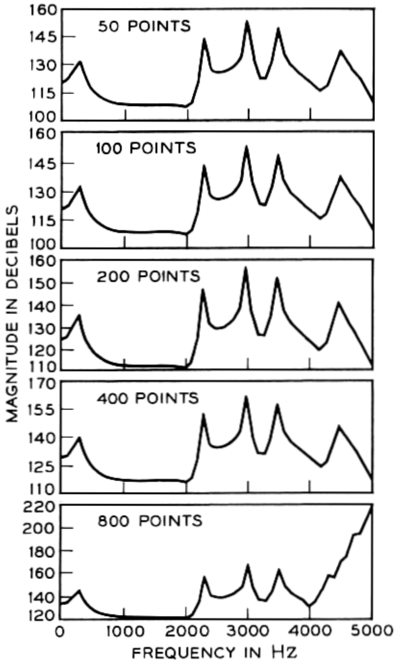

Fig. 27 — A comparison of magnitude plots for varying number of points on the same spiral contour. The fifth plot shows the effect of errors in evaluating $W^{n^2/2}$ for large n. (Points are plotted every 100 Hz in each curve to aid in comparison.)

$W_o = 1$ there is no limitation of this type since $W^{\pm n^2/2}$ is always of magnitude 1.

The other main limitation of the chirp z-transform algorithm stems from the fact that two L point fast Fourier transforms and one $L/2$ point fast Fourier transform must be evaluated where L is the smallest convenient integer greater than $N + M - 1$ as previously mentioned. We need one fast Fourier transform and $2L$ storage locations for the transform of $x_n A^{-n} W^{n^2/2}$; one fast Fourier transform and $L+2$ storage locations for the transform of $W^{-n^2/2}$; and one fast Fourier transform for the inverse transform of the product of these two transforms. We do not know a way of computing the transform of $W^{-n^2/2}$ either recursively or by a specific formula (except in some trivial cases.) Thus we must compute this transform and store it in an extra $L + 2$ storage locations. Of course, if many transforms are to be done with the same value of L we need not compute the transform of $W^{-n^2/2}$ each time.

We can compute the quantities $A^{-n} W^{n^2/2}$ recursively, as they are needed, to save computation and storage. This is easily seen from the fact that

$$A^{-(n+1)}W^{(n+1)^2/2} = (A^{-n}W^{n^2/2})\cdot W^nW^{\frac{1}{2}}A^{-1}. \tag{45}$$

If we define

$$C_n = A^{-n}W^{n^2/2} \tag{46}$$

and

$$D_n = W^nW^{\frac{1}{2}}A^{-1} \tag{47}$$

then

$$D_{n+1} = W\cdot D_n \tag{48}$$

and

$$C_{n+1} = C_n\cdot D_n\ . \tag{49}$$

Setting $A = 1$ in equations (45) through (49) provides an algorithm for the coefficients required for the output sequence. A similar recursion formula can be obtained for generating the sequence $A^{-n}W^{(n-N_o)^2/2}$. The user is cautioned that recursive computation of these coefficients may be a major source of numerical error, especially when $W_o \approx 1$ or $\varphi_o \approx 0$.

VI. SUMMARY

We give a computational algorithm for numerically evaluating the z-transform of a sequence of N time samples. This algorithm, the chirp z-transform algorithm, enables the evaluation of the z-transform at M equiangularly spaced points on contours which spiral in or out (circles being a special case) from an arbitrary starting point in the z-plane. In the s-plane the equivalent contour is an arbitrary straight line.

The chirp z-transform algorithm has great flexibility in that neither N or M need be composite numbers; the output point spacing is arbitrary; the contour is fairly general and N need not be the same as M. The flexibility of the chirp z-transform algorithm comes from being able to express the z-transform on the above contours as a convolution, permitting the use of well-known high speed convolution techniques to evaluate the convolution.

Applications of the chirp z-transform algorithm include enhancement of poles for use in spectral analysis, high resolution narrowband frequency analysis, and time interpolation of data from one sampling rate to any other sampling rate. These applications are explained in detail. The chirp z-transform algorithm also permits use of a radix 2 fast Fourier transform program or device to compute the discrete Fourier transform of an arbitrary number of samples. Examples were presented illustrating how the chirp z-transform algorithm was used in specific cases. It is anticipated that other applications will be found.

VII. ACKNOWLEDGMENT

The authors would like to acknowledge the help and assistance of Miss Carol McGonegal in programming many of the examples shown in this paper and Miss Debbie Hougak for valuable clerical help in preparing the manuscript.

APPENDIX

Fast Fourier Transforms for Two Real L *Point Sequences*

The purpose of this appendix is to show how the fast Fourier transforms of two real, symmetric L point sequences can be obtained using one $L/2$ point fast Fourier transform.

Let x_n and y_n be two real, symmetric L point sequences with corresponding discrete Fourier transforms X_k and Y_k. By definition,

$$\begin{aligned} x_n &= x_{L-n} \qquad n = 0, 1, 2, \cdots, L-1; \\ y_n &= y_{L-n} \end{aligned}$$

it is easily shown that X_k and Y_k are real, symmetric L point sequences, so that

$$\begin{aligned} X_k &= X_{L-k} \qquad k = 0, 1, 2, \cdots, L-1. \\ Y_k &= Y_{L-k} \end{aligned}$$

Define a complex, $L/2$ point sequence u_n whose real and imaginary parts are

$$\left.\begin{aligned} \mathrm{Re}\,[u_n] &= x_{2n} - y_{2n+1} + y_{2n-1} \\ \mathrm{Im}\,[u_n] &= y_{2n} + x_{2n+1} - x_{2n-1} \end{aligned}\right\} \qquad n = 0, 1, \cdots, L/2 - 1.$$

The $L/2$ point discrete Fourier transform of u_n is denoted U_k and is calculated by the fast Fourier transform. The values of X_k and Y_k may be computed from U_k using the relations

$$X_k = \tfrac{1}{2}\{\mathrm{Re}\,[U_k] + \mathrm{Re}\,[U_{L/2-k}]\} - \frac{1}{4\sin\frac{2\pi}{L}k}\{\mathrm{Re}\,[U_k] - \mathrm{Re}\,[U_{L/2-k}]\}$$

$$Y_k = \tfrac{1}{2}\{\mathrm{Im}\,[U_k] + \mathrm{Im}\,[U_{L/2-k}]\} - \frac{1}{4\sin\frac{2\pi}{L}k}\{\mathrm{Im}\,[U_k] - \mathrm{Im}\,[U_{L/2-k}]\}$$

$$\text{for} \quad k = 1, 2, \cdots, L/2 - 1.$$

The remaining values of X_k and Y_k are obtained from the relations

$$X_o = \sum_{n=0}^{L-1} x_n$$

$$Y_o = \sum_{n=0}^{L-1} y_n$$

$$X_{L/2} = \sum_{n=0}^{L-1} (-1)^n x_n$$

$$Y_{L/2} = \sum_{n=1}^{L-1} (-1)^n y_n .$$

REFERENCES

1. Cooley, J. W. and Tukey, J. W., "An Algorithm for the Machine Calculation of Complex Fourier Series," Mathematics of Computation, *19,* No. 90 (April 1965), pp. 297–301.
2. "What is the Fast Fourier Transform?," G-AE Subcommittee on Measurement Concepts, IEEE Tran. Audio and Electroacoustics, *AU-15,* No. 2, (June 1967), pp. 45–55.
3. Stockham, T. G., "High Speed Convolution and Correlation," 1966 Spring Joint Computer Conference, Amer. Federation of Inform. Processing Soc. Proc., *28,* Washington, D. C., April 1966, pp. 229–233.
4. Helms, H. D. "Fast Fourier Transform Method of Computing Difference Equations and Simulating Filters," IEEE Trans. Audio and Electroacoustics, *AU-15,* No. 2 (June 1967), pp. 85–90.
5. Bluestein, L. I., "A Linear Filtering Approach to the Computation of the Discrete Fourier Transform," 1968 Northeast Electronics Research and Engineering Meeting Record, *10,* November 1968, pp. 218–219.
6. Rader, C. M. and Gold, B., "Digital Filter Design Techniques in the Frequency Domain," Proc. IEEE, *55,* No. 2, (February 1967), pp. 149–171.

7. Schafer, R. W. and Rabiner, L. R., Automatic Formant Analysis of Speech Using the Chirp z-Transform Algorithm" to be presented at the 1969 IEEE International Conference on Communications, Boulder, Colorado, June, 1969.
8. Gold, B. and Jordan, K. L., "A Direct Search Procedure for Designing Finite Duration Impulse Response Filters," IEEE Trans. on Audio Electroacoustics, *AU-17*, No. 1 (March 1969) pp. 33–36.
9. Gentleman, W. M. and Sande, G., "Fast Fourier Transforms for Fun and Profit," 1966 Fall Joint Computer Conference, American Federation of Information Processing Societies Proc., *29*, Washington, D. C., November 1966, pp. 563–578.