Copyright © 1981 American Telephone and Telegraph Company
THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 60, No. 9, November 1981
Printed in U.S.A.

Frequency Scaling of Speech Signals by Transform Techniques

By D. MALAH* and J. L. FLANAGAN

(Manuscript received May 15, 1981)

The general framework of short-time Fourier analysis, modification, and synthesis is used to describe in a unified way several known techniques for frequency scaling of speech signals. Subsequently, a frequency domain harmonic scaling technique is studied in detail with emphasis on improving its performance and its implementation efficiency. This technique is particularly attractive for 2:1 scaling by use of a sign tracking algorithm which avoids the need for explicit phase computation and unwrapping. The implementation efficiency is achieved by using the fast Fourier transform algorithm, embedded decimation and interpolation, and an extended version of a recently developed weighted overlap-add synthesis scheme. The improvement in quality is achieved by improved sign tracking and elaborate design and selection of the analysis and synthesis prototype filters (data windows). Results of computer simulations, for a variety of adverse acoustical environment conditions, indicate that the system is highly robust but its quality for clean speech is lower than with a time domain harmonic scaling technique which uses pitch information. In applications which do not permit pitch transmission, a hybrid scheme which combines the two techniques is found to yield a better quality than either system alone.

I. INTRODUCTION

Frequency scaling of speech signals is a useful method for reducing the bandwidth requirements in analog and digital speech transmission systems.[1–5] In analog systems the frequency compressed signal is transmitted at reduced bandwidth. In digital systems the frequency

* On leave from the Electrical Engineering Department, Technion-Israel Institute of Technology, Haifa, Israel.

compressed signal is waveform coded to provide reduced bit-rate transmission.[4-7] A general block diagram of such a digital system is shown in Fig. 1. In this figure, the schematic spectral representation of the input speech signal is shown to consist of a spectral envelope with pronounced resonances (formant peaks) and of a fine structure because of pitch harmonics in voiced speech. The spectral envelope of the compressed signal is a scaled version of the input spectral envelope. However, different frequency scaling techniques may result in different fine structures.

Since we do not refer at this point to any specific technique, Fig. 1 does not show the fine structure of the compressed signal. This suggests that frequency scaling techniques can be classified according to the way the fine spectral structure is scaled. In particular, one can distinguish between narrow-band techniques, such as the phase vocoder[2] and time-domain harmonic scaling (TDHS)[4] techniques, which aim at separating and scaling the individual pitch harmonics, and wide-band techniques, such as the analytic signal rooting (ASR) technique[3] and the more recent constant Q transform (CQT) method,[8,9] which aim at directly scaling the spectral envelope. The much earlier Vobanc and CODIMEX systems also fall into the latter category.[10,11]

Another way of classification is to distinguish between time- and frequency-domain techniques. To provide useful quality, time-domain techniques require pitch tracking, as done in the TDHS technique.[4]

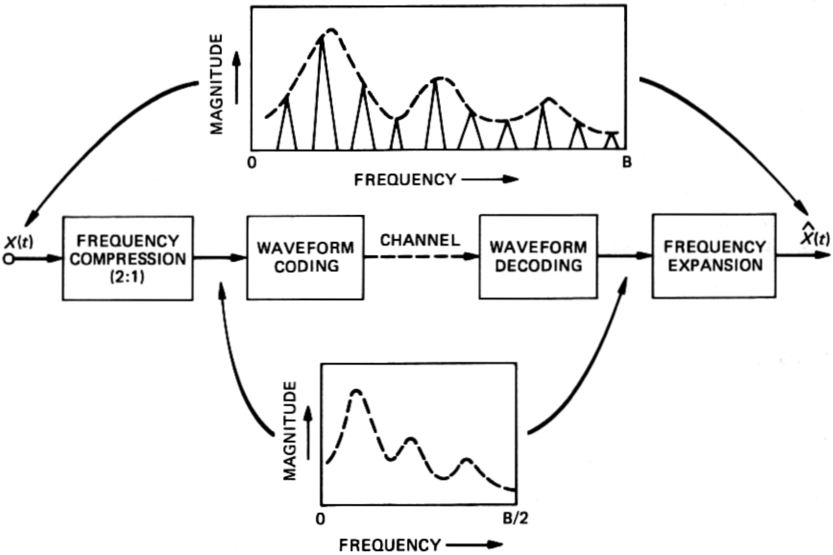

Fig. 1—General block diagram of a digital coding system which applies frequency scaling for bit rate reduction.

Once the pitch is known, the time-domain operations are simple and result in good quality scaled and reconstructed speech.[5–7] Frequency-domain techniques are typically much more complex but do not require explicit pitch tracking. However, they usually have a lower quality because of errors made in resolving phase ambiguity and from the need, in general, to scale both the phase and amplitude signals, as will be elaborated later on. For applications in which pitch tracking is not desired or not possible, because of adverse acoustical environment conditions, the use of an efficient frequency domain technique is of much interest.

In this work we present an efficient implementation of a frequency-domain harmonic scaling (FDHS) technique which is based on an improved version of the technique presented in Ref. 12. Frequency-domain harmonic scaling is a narrow-band technique which aims at scaling the individual pitch harmonics. It is particularly attractive for 2:1 scaling, since in this case a sign tracking algorithm avoids the need for explicit phase computation and unwrapping (that is, eliminating 2π phase ambiguities), which in general is a difficult and error-prone task.[13] The efficient implementation is based on the recently developed weighted overlap-add method for short-time Fourier analysis/synthesis, which allows block processing using the fast Fourier transform (FFT) algorithm.[14] It is extended here to include analysis and synthesis windows which are both longer than the FFT block, or the transform size.

The general framework of the short-time Fourier transform (STFT) as developed in several recent works[14–16] also provides a unified description of other known frequency scaling techniques, and helps to relate them to the FDHS technique. This unified description is given in the following section, and is followed by a detailed description of the FDHS technique. Section IV gives the details of the implementation scheme and Section V discusses design considerations and simulation results. Section VI presents a hybrid technique which combines TDHS and FDHS. This combination is designed for applications in which it is feasible to extract the pitch at the transmitter but for which the transmission of pitch data is either impossible or is to be avoided.

II. A UNIFIED DESCRIPTION OF FREQUENCY SCALING TECHNIQUES

A general scheme for frequency scaling is presented in this section. It is based on viewing the frequency scaling operation as a modification of the short-time spectrum of the speech signal. This scheme is then used to describe in a unified way several known frequency scaling techniques. Our attention in describing the different techniques will be mainly focused on the nature of the spectral modifications used by each technique, and not necessarily on the way they are implemented.

The relation between the short-time Fourier transform (STFT) and a filter-bank analysis is well established.[2,15] For the convenience of this presentation, the filter bank which is used to divide the two-sided speech spectrum into sub-bands is assumed to consist of N complex bandpass filters. The center frequency of the kth filter is denoted by ω_k and its complex (or analytic) output signal by $z_k(t)$. It is also assumed that each bandpass filter has a real low-pass filter prototype, which means that the complex impulse response $h_k(t)$, of the kth filter, is given by

$$h_k(t) = w_k(t)\exp(j\omega_k t), \tag{1}$$

where $w_k(t)$ is the impulse response of the low-pass prototype of $h_k(t)$. Note that in general the prototype filters need not be identical, but we assume that the bandpass filters are contiguous and are arranged symmetrically about $\omega = 0$, so that the filter centered at $\omega = -\omega_k$ has the same prototype filter as the one centered at $\omega = \omega_k$. This way the summation of the outputs from a pair of corresponding (conjugate) complex filters results in a real signal.

The output signal from the kth complex filter has the general form

$$z_k(t) = A_k(t)\exp[j\theta_k(t)], \tag{2}$$

where $A_k(t)$ is the amplitude, or envelope, function and $\theta_k(t)$ is the phase function. The phase function can be written as a sum of two components

$$\theta_k(t) = \omega_k t + \phi_k(t), \tag{3}$$

where the meaning of $\phi_k(t)$ is elaborated below. By substituting eq. (3) into eq. (2), we see that $z_k(t)$ can be interpreted as being the result of the simultaneous modulation of the amplitude and phase of the complex carrier signal $\exp(j\omega_k t)$ by the amplitude and phase signals $A_k(t)$ and $\phi_k(t)$, respectively. The instantaneous frequency of $z_k(t)$ is given by the phase derivative $\dot{\theta}_k(t) = \omega_k + \dot{\phi}_k(t)$, so that $\dot{\phi}_k(t)$ is seen to be the deviation of the instantaneous frequency from the center frequency ω_k. Frequency scaling of $z_k(t)$ by a factor q ($q < 1$ for compression and $q > 1$ for expansion), is achieved if the center frequency ω_k is scaled or shifted to $q\omega_k$ and the bandwidth of $z_k(t)$, about ω_k, is also scaled by q. It is well known that the bandwidth of a signal which is characterized by simultaneous amplitude and phase modulation of a carrier signal is a function of both modulating signals.[17] Hence, just scaling the instantaneous frequency deviation $\dot{\phi}_k(t)$ by a factor q does not result, in general, in the exact scaling of the bandwidth of $z_k(t)$ by q. The lack of adequate analytical models which describe the time variations of the amplitude and phase modulating signals—for an input speech signal—has resulted in a variety of frequency scaling techniques which use different analysis filter banks and different modifications of the modulating signals. In the following, the modifications applied to the modulation signals by different frequency scaling techniques are used to analyze and compare the different techniques. We first, however, show that the modification of $A_k(t)$ and $\phi_k(t)$ corresponds to the modification of the short-time spectrum of the speech signal.

Since $z_k(t)$ is the output signal from a bandpass filter having an impulse response $h_k(t)$ it can be expressed as the convolution between the input speech signal $x(t)$ and $h_k(t)$. From eq. (1) this results in

$$z_k(t) = X(\omega_k, t)\exp(j\omega_k t), \tag{4}$$

where

$$X(\omega_k, t) = \int_{-\infty}^{\infty} x(\tau)w_k(t-\tau)\exp(-j\omega_k\tau)d\tau. \tag{5}$$

Comparing eq. (4) with eq. (2) and using eq. (3), we have

$$X(\omega_k, t) = A_k(t)\exp[j\phi_k(t)]. \tag{6}$$

This shows that the amplitude and phase modulations of the carrier $\exp(j\omega_k t)$ are fully described by the composite modulation function $X(\omega_k, t)$. Additional understanding of these modulation functions can be gained from the studies in Refs. 18 and 19. The expression for $X(\omega_k, t)$ in eq. (5) shows that $X(\omega_k, t)$ is equal to the value of the STFT of $x(t)$ at the frequency $\omega = \omega_k$, if $w_k(t)$ is the window function used to weight the input signal.[2,15] It should be emphasized again that in the present discussion the different bandpass filters covering the speech band have, in general, different prototype filters. Hence, for each bandpass filter one can define an STFT which, if evaluated at the center frequency of that filter, gives the corresponding composite modulation function. If all bandpass filters have identical prototype low-pass filters, only a single STFT is needed to find the value of $X(\omega_k, t)$ for each k, by evaluating the STFT at each center frequency. With this understanding, we will refer to $X(\omega_k, t)$ as the STFT of $x(t)$ at $\omega = \omega_k$, even for the general case of nonidentical prototype filters.

Denoting the frequency scaled version of $z_k(t)$ by $z_{qk}(t)$ and the corresponding modified STFT by $X_q(\omega_k, t)$, we have

$$z_{qk}(t) = X_q(\omega_k, t)\exp(jq\omega_k t). \tag{7}$$

The magnitude and phase components of $X_q(\omega_k, t)$ are accordingly denoted by $A_{qk}(t)$ and $\phi_{qk}(t)$, respectively. As noted above, the modification of $X(\omega_k, t)$ needed for exact frequency scaling of speech signals is not known, and the bandwidth of the individual sub-band signals is usually only partially scaled by any given technique. Hence, to avoid excessive interband aliasing when the partially scaled sub-band signals are combined, additional filtering of $z_{qk}(t)$ may be needed. The filtering of $z_{qk}(t)$ can be performed either by bandpass filters having a bandwidth which is q-times the bandwidth of the analysis filters, or equivalently by low-pass filtering the modified STFT by the corresponding low-pass prototype filters. Figure 2 shows a general block diagram for frequency scaling which is based on modifying the STFT of the input signal as discussed above. The impulse response of the synthesis low-pass filters which are used to band-limit the output signals in each channel are denoted by $w_{qk}(t)$. These scaled-bandwidth synthesis filters can generally be obtained from $w_k(t)$ by the relation $w_{qk}(t) = w_k(qt)$. In the diagram of Fig. 2, only the details of the kth channel are given since all the other channels are similar (see solid line). The filtered modified STFT is denoted in Fig. 2 by $\hat{X}_q(\omega_k, t)$. The output scaled speech signal $\hat{y}_q(t)$ is given by

$$\hat{y}_q(t) = \sum_k \hat{z}_{qk}(t) = \sum_k \hat{X}_q(\omega_k, t)\exp(jq\omega_k t), \tag{8}$$

where, as seen in Fig. 2, $\hat{z}_{qk}(t)$ is the kth-channel scaled and filtered bandpass signal. The summation in eq. (8) is over the N sub-bands.

It is clear from the above discussion, and from the block diagram in Fig. 2, that the choice of the filter bank and the STFT modification are the key issues for any given technique. While the block diagram in Fig. 2 provides a basis for comparing different techniques, the actual implementations can differ, either because of historical reasons or the availability of more efficient or convenient ways for implementation.

We turn now to the description of several known frequency scaling techniques in terms of the STFT modification used by each technique. This will exemplify the above discussion and will provide a proper perspective for discussing the FDHS technique and its properties and implementation.

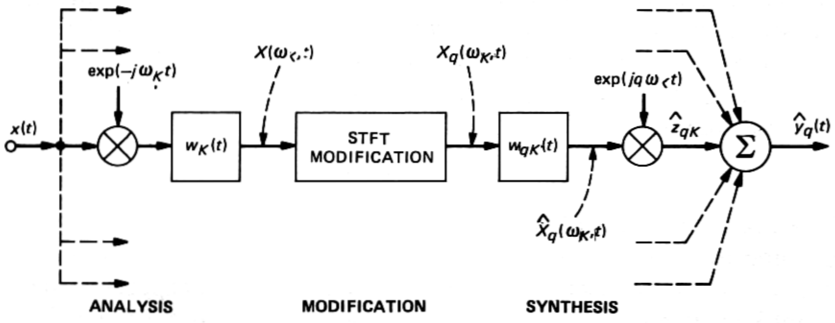

Fig. 2—General block diagram of a frequency-scaling system based on short-time spectral modification (kth channel shown in solid line).

2.1 Analytic signal rooting

In the analytic signal rooting technique[3] the number of bandpass filters is chosen to match the formant structure of speech signals so that, preferably, no more than one formant is present in each sub-band. The approach taken in Ref. 3, as well as in the earlier CODIMEX system,[11] is to obtain $z_{qk}(t)$ by raising the analytic signal $z_k(t)$ to the power of q. If $q < 1$, this corresponds to taking the $1/q$ root of $z_k(t)$ which is the origin for the name of this technique. Using the relations in eqs. (4) and (7), we find that the STFT modification performed by this technique is

$$X_q(\omega_k, t) = [X(\omega_k, t)]^q. \tag{9}$$

In terms of the modulating amplitude and phase signals, this modification corresponds to

$$A_{qk}(t) = [A_k(t)]^q, \tag{10a}$$

and

$$\phi_{qk}(t) = q\phi_k(t). \tag{10b}$$

To understand the effect of this modification, we note that since each sub-band is to contain no more than one formant, it can be expected that most often one-pitch harmonic, the one closest to the peak of the formant, is dominant to the other harmonics in that sub-band. From the analysis in Refs. 20 and 21 one can conclude that the phase-scaling operation in eq. (10b) scales the instantaneous frequency of the dominant harmonic in each band by q, but the other lower amplitude harmonics are shifted in such a way that their spacing from the dominant harmonic remains unchanged. The result of this translation is that the fine structure spectral components are not necessarily harmonic, although their spacing is equal to the pitch frequency. The scaling of the amplitude signals in the way given by eq. (10a) can be shown to scale the magnitude of the nondominant components (harmonics) relative to the amplitude of the dominant component. It also affects the intermodulation terms generated by the phase scaling. For $q < 1$, the effect is to reduce the magnitude of the nondominant components relative to the dominant one and hence, effectively, to reduce the bandwidth of the frequency-scaled formants. To avoid excessive interband aliasing, it is particularly important in this technique to use the band-limiting low-pass filters $w_{qk}(t)$ following the modification.

For more effective scaling of the amplitude signals, and with respect to the CQT which uses constant-Q bandpass filters, consider the approach by Ravindra.[9] He suggests that the $A_k(t)$ be spectrally analyzed for each k by an additional bank of filters and that the bandwidth be scaled by scaling the phase in each sub-band—this can be repeated in a tree-like structure.[9] The implementation complexity of this approach, however, appears to be exorbitant.

2.2 Phase vocoder

The phase vocoder, as its name indicates, can be used directly as a vocoder system in which the phase derivative and magnitude of the input signal STFT are coded and transmitted.[1,2,22,23] The phase vocoder technique can also be applied for frequency scaling and this aspect is considered here.[2]

In the phase vocoder, the number of bandpass filters is chosen to match the harmonic structure of voiced speech.[2] This means that a relatively large number of filters is used so that, preferably, no more than one-pitch harmonic is present in each sub-band. The fact that individual harmonics are separately scaled, allows us to infer the characteristics of the modulation signals in each band from known speech properties. In particular, since pitch and vocal tract variations are relatively slow, the bandwidth of each pitch harmonic is quite narrow, as shown by the "pitch teeth" in the input spectrum shown in Fig. 1. In view of this fact, one would expect that even if the pitch harmonics are only shifted to the proper frequencies, without scaling the bandwidth of each pitch tooth, acceptable compression can be achieved (i.e. only a small interharmonic aliasing is expected), provided that the compression ratio is limited to 2 or at most 3. Indeed, this finds support in the results obtained with the TDHS technique which we discuss later.[4,5] However, to shift the pitch harmonics to the proper locations requires knowledge of the pitch frequency or, equivalently, the deviation of each pitch harmonic from the center frequency of the sub-band in which it is located.

Let Ω_k be the pitch-harmonic frequency in the kth sub-band, with center frequency ω_k, and $\Delta\Omega_k$ the deviation of the pitch harmonic from the center frequency; i.e., $\Delta\Omega_k = \Omega_k - \omega_k$. Then, the phase derivative $\dot{\phi}_k(t)$ can be expressed as

$$\dot{\phi}_k(t) = \Delta\Omega_k + \dot{\Psi}_k(t), \qquad (11)$$

where $\dot{\Psi}_k(t)$ describes the contribution of the phase variations to the bandwidth of the pitch harmonic in the kth sub-band. In the phase vocoder technique, the phase derivative $\dot{\phi}_k(t)$ is scaled by q, so that in addition to shifting each pitch tooth to its proper location, a partial scaling of its bandwidth is obtained since $\dot{\Psi}_k(t)$ is scaled as well. The amplitude modulation signals are not modified in this technique. However, since individual harmonics are analyzed, the amplitude signal in each band varies slowly [see (a) of Fig. 5 in Ref. 19], and its contribution to the pitch-tooth bandwidth is expected to be small.

Accordingly, the modified amplitude and phase signals are given by

$$A_{qk}(t) = A_k(t), \tag{12a}$$

and

$$\phi_{qk}(t) = \int_{t_0}^{t} q\dot{\phi}_k(\tau)d\tau. \tag{12b}$$

It is observed from eq. (12b) that the constant phase term $\phi_k(t_0)$ is discarded [note: $\phi_k(t) = \int_{t_0}^{t} \dot{\phi}_k(\tau)d\tau + \phi_k(t_0)$]. This can have an effect on the shape of the scaled signal waveform, but because of the relative insensitivity of the ear to a fixed phase distortion, it was not judged to be perceptually significant.[2]

Since in this technique individual pitch harmonics are scaled and the interband aliasing is expected to be small, use of the output synthesis filters, denoted by $w_{qk}(t)$ in Fig. 2, is less compelling than for the ASR technique, but it can still be useful.

It should be noted that the phase vocoder technique can perform time-scale variations of speech signals simply by playing back the signal which has been frequency-scaled by a factor q at $(1/q)$-times the original speed. This restores the original frequency range but scales the signal's time duration by q. This useful property is not shared by the ASR technique because of the way the pitch harmonics are shifted and because of the nonlinear scaling of the amplitude signals. On the other hand, the ASR technique can be useful for restoring speech distorted by a helium atmosphere, where scaling of the formants without changing the perceived pitch of the signal is desired.[1,3]

We turn now to the more recently developed time-domain harmonic scaling (TDHS) technique.[4] Although this technique is most efficiently implemented in the time domain, it was formulated and derived within the STFT framework.

2.3 Time-domain harmonic scaling

As noted in the discussion on the phase vocoder technique, compression factors of up to 3 can possibly be obtained even if the bandwidth of each pitch harmonic is not scaled, provided that the pitch harmonics are shifted to the correct frequency locations. In the phase vocoder this necessitates scaling the phase derivative of the STFT so that $\Delta\Omega_k$, the frequency deviation of the pitch harmonic in the kth sub-band from the center frequency ω_k, is scaled by q. The approach taken by the TDHS technique is to incorporate pitch information which is obtained by a separate pitch detector, into the scaling process.[4] If the pitch frequency is known, the bandwidth of each bandpass filter can be made equal to the pitch frequency and the center frequency of each bandpass filter can be aligned with the corresponding pitch harmonic, so that $\Delta\Omega_k = 0$ for all the bandpass filters which cover the speech band. Here, in principle, the number of bandpass filters also varies with the pitch frequency and is equal to the number of pitch harmonics in the given speech band. Hence, if only shifting of the pitch harmonics is desired, as schematically shown in Fig. 3 (for $q = 1/2$ and $q = 2$), without scaling the pitch-teeth bandwidth, there is no need to modify the modulating amplitude and phase signals (i.e. the STFT), but only to scale the carrier, or center, frequencies. Thus,

$$X_q(\omega_k, t) = X(\omega_k, t), \tag{13}$$

with the understanding that ω_k is chosen to coincide with the pitch harmonic Ω_k in the kth sub-band. Using eq. (13) in eq. (8), and assuming that no synthesis filters are used, the output-scaled signal $y_q(t)$ is given by

$$y_q(t) = \sum_k X(\omega_k, t)\exp(jq\omega_k t). \tag{14}$$

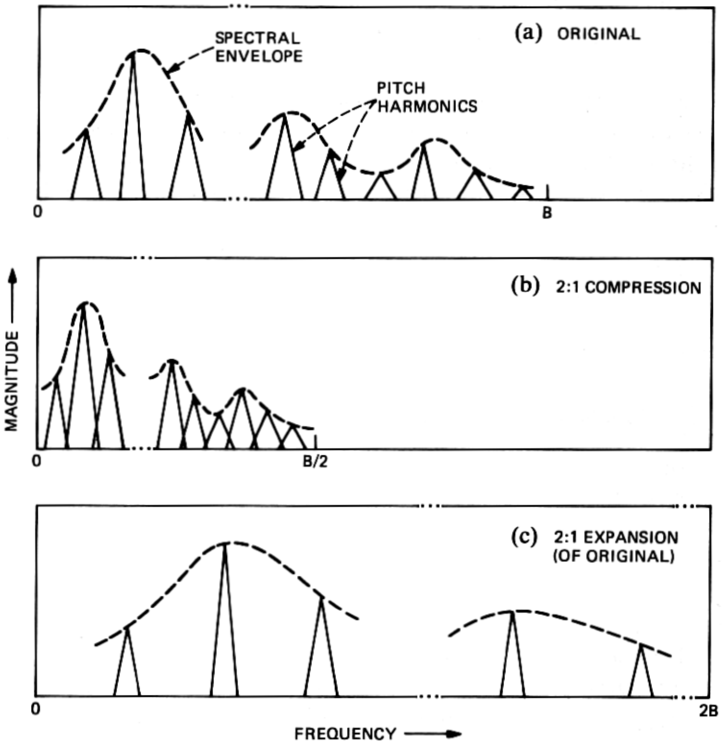

Fig. 3—Schematic spectral representation of frequency scaling by shifting of the pitch harmonics.[5]

Equation (14) can be used to derive an explicit linear relation between $y_q(t)$ and $x(t)$ by substituting the right-hand side of eq. (5) for $X(\omega_k, t)$ in eq. (14). Discretization of this relation yields the TDHS algorithms,[4-6] which were successfully applied in conjunction with several waveform coding systems for an effective reduction in the required transmission bit-rate of speech signals.[5-7] Like the phase vocoder, the TDHS technique can also be used for time scaling of speech signals by playing back the signal at a different speed or, equivalently, at a different sampling rate.

Before we conclude this section, we would also like to mention two additional time- or frequency-scaling systems. One is the scaling system devised in Ref. 24 to which we will refer in a later section; the other is the recent CQT method to which we already referred in Section 2.1.[8,9] In the CQT method a constant Q analysis filter bank is used (i.e., the $w_k(t)$'s in Fig. 2 are not identical) which resembles the type of spectral analysis performed by the ear. Originally the modification consisted of scaling only the unwrapped phase in each band.[8] However, since some bands may contain more than one-pitch harmonic, the amplitude signals may vary significantly and need to be scaled as well. [See (a) of Fig. 7 in Ref. 19.] As mentioned earlier, the approach proposed in Ref. 9 is to analyze the amplitude signal in each band with another bank of filters and to scale the phase signals.

III. FREQUENCY-DOMAIN HARMONIC SCALING TECHNIQUE

The basic FDHS technique is given in Ref. 12. In this section, we relate this technique to the techniques described in the previous section and give its details, including modifications which we introduce in the original sign tracking algorithm.[12]

As in the phase vocoder, the FDHS technique aims at scaling the individual pitch harmonics of voiced speech signals. However, in FDHS, the total phase is scaled (including the constant phase term) which is discarded in the phase vocoder technique. Also, as in other narrowband techniques the amplitude signals remain unmodified.

Therefore, the modified STFT amplitude and phase components are given by

$$A_{qk}(t) = A_k(t), \tag{15a}$$

and

$$\phi_{qk}(t) = q\phi_k(t). \tag{15b}$$

This type of modification has been the underlying modification of several early techniques which are described and analyzed in Ref. 21. The more recent techniques reported in Refs. 24 and 25 are also based on this modification, if only frequency compression or expansion is considered. However, the way the FDHS technique performs the phase modification is specific to this technique as is elaborated in the following.

For noninteger values of q, the phase modification in eq. (15b) must be performed on the unwrapped phase. Otherwise, phase ambiguity of a multiple of 2π, which results from computing the principal value of the phase, may give rise to an incorrect modified phase value. In the phase vocoder technique, the difficult task of explicit phase unwrapping is avoided by directly computing the phase derivative from the real and imaginary components of the STFT and their time derivatives.[2] In the ASR technique, the scaling factors are restricted to $q = 1/2$ for compression and $q = 2$ for expansion.[3] The explicit phase division by 2 for $q = 1/2$ is then avoided by expressing the scaled signal in each band in terms of the input signal and its envelope (amplitude function), with a sign which is determined by means of a simple sign tracking algorithm.[3]

Since compression and expansion by a factor of 2 is of most practical interest (speech quality at higher scaling factors degrades rapidly), the approach of using 2:1 scaling and avoiding explicit phase computation and unwrapping, by means of a proper sign tracking algorithm, is adopted also by the FDHS technique presented in Ref. 12. We now present the details of the FDHS technique and the modifications introduced in the original sign tracking algorithm.

Let $a_k(t)$ and $b_k(t)$ be the real and imaginary parts, respectively, of the input signal STFT, $X(\omega_k, t)$, and $a_{qk}(t)$, $b_{qk}(t)$ the corresponding real and imaginary parts of the modified STFT $X_q(\omega_k, t)$. From eqs. (6) and (15) we find,

$$a_k(t) = A_k(t)\cos\phi_k(t); \quad a_{qk}(t) = A_k(t)\cos[q\phi_k(t)]. \tag{16a}$$

$$b_k(t) = A_k(t)\sin\phi_k(t); \quad b_{qk}(t) = A_k(t)\sin[q\phi_k(t)]. \tag{16b}$$

By restricting q to be 1/2 or 2 and using basic trigonometric relations, the following algorithms were derived in Ref. 12.

3.1 Compression ($q = 1/2$)

Using half-angle trigonometric relations, one finds from eq. (16)

$$a_{k/2}(t) = \mathrm{SGN}[a_{k/2}(t)]\{[A_k(t)/2][A_k(t) + a_k(t)]\}^{1/2}, \tag{17a}$$

$$b_{k/2}(t) = \mathrm{SGN}[b_{k/2}(t)]\{[A_k(t)/2][A_k(t) - a_k(t)]\}^{1/2}, \tag{17b}$$

where the signs [SGN$(\cdot)$] are determined according to the quadrant in which the complex vector (or phasor) $[a_{k/2}(t) + jb_{k/2}(t)]$ is present at any given time instant t. Once initialized, a consistent sign, or quadrant, tracking algorithm is given by the following rules or logic.[12]

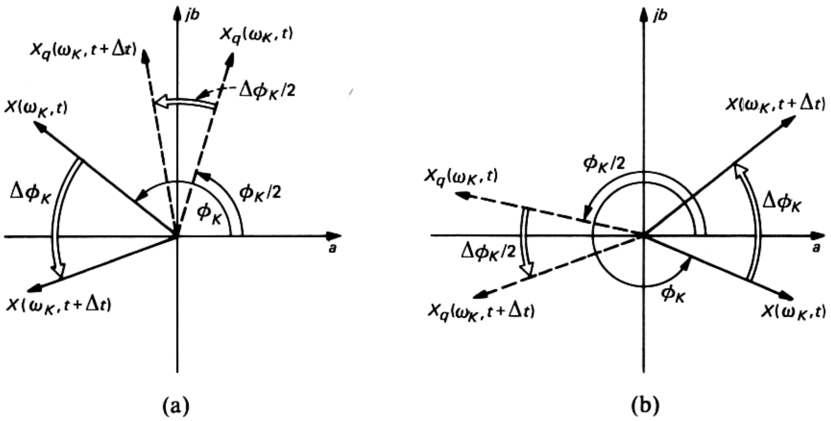

Fig. 4—Demonstration of the validity of the change of sign rules in eq. (18): (a) $\mathrm{SGN}[a_{k/2}(t+\Delta t)] = -\mathrm{SGN}[a_{k/2}(t)]$; (b) $\mathrm{SGN}[b_{k/2}(t+\Delta t)] = -\mathrm{SGN}[b_{k/2}(t)]$.

$$\mathrm{SGN}[a_{k/2}(t+\Delta t)] = \begin{cases} -\mathrm{SGN}[a_{k/2}(t)] & \text{if } A \cap B \text{ is true} \\ \mathrm{SGN}[a_{k/2}(t)], & \text{otherwise} \end{cases} \tag{18a}$$

$$\mathrm{SGN}[b_{k/2}(t+\Delta t)] = \begin{cases} -\mathrm{SGN}[b_{k/2}(t)] & \text{if } \bar{A} \cap B \text{ is true} \\ \mathrm{SGN}[b_{k/2}(t)], & \text{otherwise} \end{cases}, \tag{18b}$$

where A and B are logical variables such that

$$A \text{ is true if } a_k(t+\Delta t) < 0,$$

$$B \text{ is true if } \mathrm{SGN}[b_k(t+\Delta t)] = -\mathrm{SGN}[b_k(t)].$$

The validity of these rules is demonstrated in the phasor diagram shown in Fig. 4. In this figure, the solid-line vectors represent the input STFT at times t and $t + \Delta t$, with a corresponding phase change of $\Delta\phi_k = \Delta\Omega_k \Delta t$, where $\Delta\Omega_k$ is the deviation of the harmonic from the center frequency ω_k. The dashed-line vectors represent the modified STFT and rotate at half the angular velocity. In (a) of Fig. 4 we illustrate a situation in which the input vector crosses the real axis; that is, condition B is "true." Since the real part of the input vector is negative (i.e. condition A is also "true"), the rule in eq. (18a) states that the real part of the modified STFT changes its sign during the time interval Δt, as is indeed the case—the dashed-line vector crosses the imaginary axis. In the same way, (b) of Fig. 4 illustrates a situation in which the imaginary part of the modified STFT changes its sign in agreement with the rule in eq. (18b). Similar diagrams can show that the rules are valid even if the direction of rotation of the phasors shown in Fig. 4 is reversed.*

* The direction of rotation depends on the location of the pitch harmonic with respect

Originally, it was proposed to initialize the sign tracking algorithm described above by assigning positive values to the signs of $a_{k/2}$ and $b_{k/2}$ in all bands,[12] or for that matter, any arbitrary assignment will do as well. However, further analysis shows that such an initialization can result in shifting some pitch harmonics to incorrect frequencies and thereby breaking up the harmonic structure. This situation results when the signs of $a_{k/2}$ and $b_{k/2}$ in a given band happen to be initialized such that the sign of one is correct and the sign of the other is reversed. The sign tracking algorithm in eq. (18) will then cause the scaled signal vector (phasor) to reverse its direction of rotation. This can be verified with the help of a phasor diagram such as in Fig. 4. For simplicity, let the input signal be of the form $A\exp(j\Omega_0 t)$, such that $\Omega_0 = \omega_k + \Delta\Omega$, where ω_k is the center frequency of kth band, and $\Delta\Omega \leq \Delta\omega$ (the bandwidth of that band). Then, the inversion of rotation direction discussed above will result in a scaled signal of the form $A\exp[j(\omega_k/2 - \Delta\Omega/2)t] = A\exp[j(\Omega_0/2 - \Delta\Omega)t]$ instead of the desired signal $A\exp[j(\omega_k/2 + \Delta\Omega/2)t] = A\exp[j(\Omega_0/2)t]$. The situation can be even worse when two adjacent filters share a single harmonic;* that is, the harmonic lies in the transition bands of the two filters as schematically shown in (a) of Fig. 5. In this case, the following conditions may arise: (*i*) The sign initialization in both bands is correct so that the components from the two bands sum up to

$$A_1\exp[j(\omega_1/2)t]\exp[j(\Delta\Omega_1/2)t] + A_2\exp[j(\omega_2/2)t]\exp[-j(\Delta\Omega_2/2)t]$$
$$= (A_1 + A_2)\exp[j\Omega_0/2)t],$$

where ω_1 and ω_2 are assumed to be the center frequencies of the two bandpass filters under consideration, $\Delta\Omega_1$ and $\Delta\Omega_2$ are the deviations of the given harmonic, of frequency Ω_0, from ω_1 and ω_2, respectively, and A_1 and A_2 are the corresponding signal amplitudes in each band. For a uniform filter-bank design, $A_1 + A_2 = A$, where A is the input signal amplitude. This is the desired result as depicted in (b) of Fig. 5. (*ii*) The sign initialization in one band, say the second one, is incorrect but such that only one of the signs of $a_{k/2}$ or $b_{k/2}$ is reversed. As discussed earlier, if we use phasor description, this will reverse the direction of rotation of the scaled component in that band and will result in

$$A_1\exp[j(\omega_1/2)t]\exp[j(\Delta\Omega_1/2)t] + A_2\exp[j(\omega_2/2)t]\exp[j(\Delta\Omega_2/2)t]$$
$$= A_1\exp[j(\Omega_0/2)t] + A_2\exp[j(\Omega_0/2 + \Delta\Omega_2)t].$$

to the center frequency ω_k of the sub-band in which it is located (i.e. on the sign of $\Delta\Omega_k$). If $\Delta\Omega_k > 0$, the phasor rotates in a counterclockwise direction, as in Fig. 4, whereas if $\Delta\Omega_k < 0$, its direction of rotation is reversed.

* It is assumed that the side lobes of the filters in the filter bank are sufficiently small so that only two adjacent filters which share the same harmonic are considered.

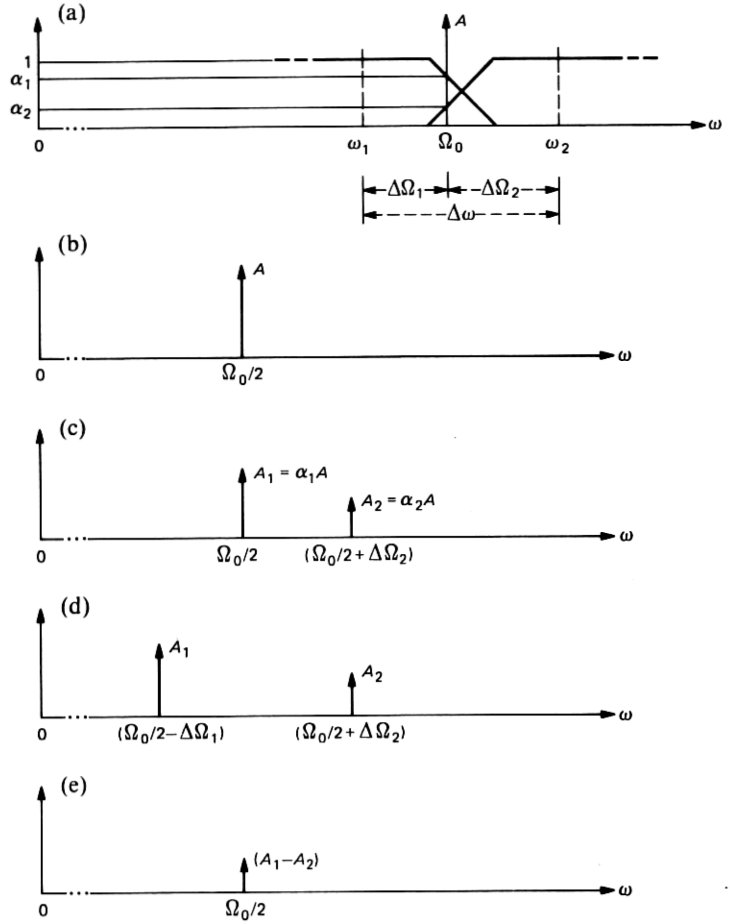

Fig. 5—Schematic representation of the effect of sign initialization on frequency compression ($q = 1/2$) of a harmonic shared by two adjacent filters. (a) Original harmonic. (b) Condition (*i*): correct sign initialization. (c) Condition (*ii*): sign inversion in one band of one of ($a_{k/2}$, $b_{k/2}$). (d) Condition (*iii*): sign inversion in both bands of one of ($a_{k/2}$, $b_{k/2}$). (e) Condition (*iv*): sign inversion in one band of both ($a_{k/2}$, $b_{k/2}$).

This result is shown in (c) of the figure. (*iii*) The sign initialization in both bands is incorrect, and in both bands it is such that only one of the signs of $a_{k/2}$ or $b_{k/2}$ is reversed. In this case, as shown in (d) both components are shifted to incorrect locations. (*iv*) The signs in one band, say the second one, are initialized incorrectly, such that both signs of $a_{k/2}$ and $b_{k/2}$ are reversed. In this case the rotation direction can be shown to remain unchanged, but there is a π phase shift in the phase of the scaled component in this band. Hence, when the two components from the two adjacent bands are recombined in the synthesis stage, the following scaled signal results

$$(A_1 - A_2)\exp[j(\Omega_0/2)t].$$

This is shown in (e). As will be discussed later, designing the filters for minimal overlap, or narrow transition bands, mitigates the amplitude interactions. (v) The signs in both bands are initialized incorrectly, but in both bands both signs of $a_{k/2}$ and $b_{k/2}$ are reversed. In this case there is a π shift in both bands and this results in a constant π phase shift in the recombined scaled signal as well, resulting in

$$-(A_1 + A_2)\exp[j(\Omega_0/2)t] = -A\exp[j(\Omega_0/2)t].$$

This situation is of no concern. It may affect the compressed signal waveform but is not found to have any perceptual effect (if we listen to the time compressed version of the signal). Furthermore, in the expansion process the phase is multiplied by 2 and this constant phase shift is cancelled. Similarly, if the harmonic is located within a single band, a sign inversion is produced in the scaled harmonic, which is of no concern.

We have seen that certain errors in sign initialization can have an adverse effect on the resulting scaled signal, and particularly so when a harmonic is shared by two filters. The best approach for initialization would appear to be the use of phase unwrapping along the frequency axis at the time of initialization, so that correct initial signs are assigned to each band. However, this is generally a difficult and error prone task, which typically requires a high-frequency resolution.[13] Also, since speech is nonstationary, reinitialization is needed quite often and a complex phase unwrapping technique will defeat the whole purpose of using a simple sign tracking algorithm. Therefore, we have examined the implications of using the principal value of the phase in each band for assigning the initial signs of $a_{k/2}$ and $b_{k/2}$. To do this, there is no need to actually compute the phase but only to examine the sign of $b_k(t)$ at the instant of initialization. Since the principal value of the phase is assumed to be in the range 0 to 2π, the initial position of the vector $[a_{k/2}(t) + jb_{k/2}(t)]$ must be assumed to be in the first or second quadrant. Hence, a sign initialization according to the principal value of the phase is given by the following simple rules:

$$\mathrm{SGN}[a_{k/2}(t_0)] = \begin{cases} 1 & \text{if } b_k(t_0) > 0 \\ -1 & \text{if } b_k(t_0) < 0 \end{cases}. \tag{19a}$$

$$\mathrm{SGN}[b_{k/2}(t_0)] = 1. \tag{19b}$$

An analysis of this sign initialization shows that either both signs of $a_{k/2}$ and $b_{k/2}$ are correct or both are reversed. If the harmonic is located within a single filter, the scaled harmonic will be shifted to the correct frequency with a possible phase shift of π – which is of no concern.

However, if the harmonic is shared by two adjacent filters, the sign initialization according to eq. (19) can give rise to conditions (*iv*) and (*v*) discussed above. In summary, we face a problem only when a harmonic is shared by two adjacent filters and condition (*iv*) exists. As shown in (e) of Fig. 5, under this condition the harmonic is shifted to the correct frequency location but its amplitude is generally attenuated and it can even be canceled, if $A_1 = A_2$. This possibility of attenuation, or even cancellation, is still of concern. After further study of the problem, we found it possible to devise a procedure for matching the signs in the two bands so that no attenuation of the scaled signal will occur. (A π phase shift of the recombined harmonic is still possible, but as explained earlier this is of no concern). We now explain this sign matching procedure.

For the purpose of simplifying the explanation, consider again a single input complex tone at frequency Ω_0, of the general form $A\exp[j(\Omega_0 t + \phi_0)]$, with Ω_0 satisfying $\omega_1 < \Omega_0 < \omega_2$, where ω_1 and ω_2 are again the center frequencies of two adjacent bands. Following phase scaling and sign initialization according to eq. (19), the modified STFT signals $X_1(t) = X_q(\omega_1, t)$ and $X_2(t) = X_q(\omega_2, t)$ (with $q = 1/2$) are given by

$$X_1(t) = S_{x_1} A_1 \exp[j(\Delta\Omega_1 t + \phi_0)/2],$$

$$X_2(t) = S_{x_2} A_2 \exp[-j(\Delta\Omega_2 t - \phi_0)/2],$$

where A_1 and A_2 are the magnitudes of the components in the two bands. Further, $A_1 + A_2 = A$, $\Delta\Omega_1 = \Omega_0 - \omega_1$, $\Delta\Omega_2 = \omega_2 - \Omega_0$, and S_{x_1}, S_{x_2} can only take the values of (+1) or (−1) and are used to denote a possible sign inversion because of an incorrect sign initialization. Therefore, the problem is to make S_{x_1} and S_{x_2} equal to each other so that the two components will recombine without attenuation, a sign inversion being inconsequential. To perform this matching of S_{x_1} and S_{x_2} we examine the function $R(t)$ given by

$$R(t) \triangleq X_1(t)/X_2(t) = S_R(A_1/A_2)\exp[j(\Delta\omega/2)t], \qquad (20)$$

where $\Delta\omega = \omega_2 - \omega_1 = \Delta\Omega_1 + \Delta\Omega_2$ is the fixed frequency separation between the centers of the two given bands, and

$$S_R = \begin{cases} 1 & \text{if } S_{x_1} = S_{x_2} \\ -1 & \text{if } S_{x_1} \neq S_{x_2} \end{cases}.$$

Our goal then is to have a sign initialization for which $S_R = 1$. Since $R(t)$ is independent of the unknown value of Ω_0, we can find the value of S_R by examining $R(t)$ at any of the specific time instants $t = t_n =$

$nT_0 = n(2\pi/\Delta\omega)$, (where n is an integer) which results in

$$R(t_n) = S_R(A_1/A_2)(-1)^n. \tag{21}$$

Hence, S_R can be determined from*

$$S_R = \text{SGN}[R(t_n)](-1)^n. \tag{22}$$

In practical situations where the filters are not ideal, such as when leakage from other bands is present, and the signal is not purely periodic, $R(t)$ at $t = t_n$ is not necessarily real as is implied in eq. (21). Therefore, we use the sign of the real part of $R(t_n)$; that is,

$$S_R = \text{SGN}[\text{Re}\{R(t_n)\}](-1)^n. \tag{23}$$

To evaluate the right-hand side of eq. (23), there is no need to fully compute $R(t_n)$, as shown below. Let $X_1(t_n) = \alpha_1 + j\beta_1$ and $X_2(t_n) = \alpha_2 + j\beta_2$—for convenience, the explicit time dependence of α_1, β_1, α_2, and β_2 is suppressed. Then, by the definition in eq. (20),

$$\text{SGN}[\text{Re}\{R(t_n)\}] = \text{SGN}(\alpha_1\alpha_2 + \beta_1\beta_2). \tag{24}$$

Hence, assuming that we know two bands which share a single harmonic, the complete initialization procedure (at an appropriate time instant $t_n = nT_0$) consists of first initializing the two signs of $a_{k/2}$ and $b_{k/2}$ according to eq. (19) and then evaluating S_R from eqs. (23) and (24). If $S_R = 1$, the two components are matched. If $S_R = -1$, the signs in one band should be inverted. If the signal is stationary, the above initialization procedure needs to be done only once as the correct sign will continue to be tracked by the sign tracking algorithm in eq. (18). However, even for the stationary case one has to first determine which are the two bands which share a single harmonic because, in general, both bands on the two sides of the band to be initialized may contain signal components. One way to find out which is the correct pair of bands is to compute the phase of $R(t_n)$ for each of the two bands and pick the band which has the smaller phase angle. Again, there is no need to explicitly compute the value of the phase of $R(t_n)$ for the two bands in question, but it is sufficient to compute the ratio $(\alpha_2\beta_1 - \alpha_1\beta_2)/(\alpha_1\alpha_2 + \beta_1\beta_2)$ for each of the two possible pairs, and pick the pair for which this ratio has the smaller magnitude. However, in simulations we have found that, given sufficient frequency resolution (relative to the separation between harmonics), it is also adequate to simply choose the band in which the signal magnitude is larger.

In addition to the above pairing issue, since speech is nonstationary

* By choosing t_n to be a multiple of $2T_0$, the term $(-1)^n$ can be avoided. However, for better tracking of pitch frequency variations we prefer to reduce the time interval between possible sign initializations to T_0.

one is also faced with the problem of detecting the following conditions: onset of speech, transitions from unvoiced to voiced, and the appearance of a pitch harmonic in a band because of pitch variation. In all of these cases, the signs should be reinitialized according to the procedure discussed above. We have found in simulations that satisfactory automatic tracking of the above conditions is obtained by reinitializing the signs in any band k for which

$$|X(\omega_k, t_n)|/|X(\omega_k, t_{n-1})| \geq E_T, \tag{25}$$

where E_T is a preset threshold (typically, as elaborated in Section V, it is set to correspond to an energy increase of 10 dB in the time interval $T_0 = 16$ ms). The mechanization of this initialization process is further detailed in Section 5.1.

3.2 Expansion ($q = 2$)

From eq. (16) we have

$$a_{2k}(t) = A_k(t)\cos[2\phi_k(t)], \tag{26a}$$

and

$$b_{2k}(t) = A_k(t)\sin[2\phi_k(t)]. \tag{26b}$$

Using double-angle trigonometric relations and eq. (26) one finds,

$$a_{2k}(t) = \{2[a_k(t)]^2 - [A_k(t)]^2\}/A_k(t), \tag{27a}$$

and

$$b_{2k}(t) = 2\ a_k(t)b_k(t)/A_k(t), \tag{27b}$$

where it is assumed that $A_k(t) \neq 0$. If $A_k(t) = 0$, then, directly from eq. (26), $a_{2k}(t) = b_{2k}(t) = 0$.

It is seen that because the phase is multiplied by an integer ($q = 2$), a phase ambiguity which is a multiple of 2π is of no concern in frequency expansion. Note also that if the compressed signal is expanded (for signal reconstruction), A_k, a_k, and b_k, should be replaced in eq. (27) by $A_{k/2}$, $a_{k/2}$, and $b_{k/2}$, respectively.

In comparing the STFT modifications, as expressed by eq. (15) for this technique and by eq. (10) for the ASR technique, one may question if it is useful to use eq. (10a) in place of eq. (15a). The answer is no. The reason is that, again, with practical filters a pitch harmonic may be shared by two adjacent filters. Hence, if a nonlinear modification, such as that in eq. (10a), is applied to the amplitude signal in each band, the two signals (which are components of the same harmonic) will in general not recombine to the correct magnitude. Again, since the analysis is narrow-band the scaling of the amplitude signal is generally of secondary importance.

IV. EFFICIENT DISCRETE-TIME IMPLEMENTATION

The basis for the discrete time implementation of the FDHS technique is the general block diagram for frequency scaling in Fig. 2. However, in this form (following discretization) it is highly inefficient because a large amount of computation is needed to perform the filtering analysis and synthesis operations. It has been demonstrated in several works[14–16,26] that a discrete Fourier transform (DFT) formulation, using the FFT algorithm, can be used for STFT analysis and synthesis with its accompanying large saving in computation. We particularly found the weighted overlap-add (WOLA) method given in Ref. 14, to be most suitable for our application. However, we had to extend the particular scheme given in Fig. 2 of Ref. 14 to accommodate situations in which both the analysis and synthesis windows (the prototype low-pass filter impulse responses) have durations longer than the transform size N. This arises from the need to use analysis filters with narrow transition bands, and, hence long duration, when performing frequency compression. This design minimizes filter overlap and lowers the probability of obtaining harmonics shared by adjacent filters, and, therefore, lessens the effect of possible incorrect sign initialization in such bands.

We begin by showing that a discrete-time version of the block diagram in Fig. 2 of this paper has the basic form considered in Ref. 14, so that the WOLA scheme developed is indeed suitable for our application.[14]

Figure 6 shows a discrete-time form of the block diagram in Fig. 2 which is made to match Fig. 3 in Ref. 14 (with somewhat different notation). This requires some clarification, which is given next.

The discrete-time input signal $x(nT)$ represents samples of $x(t)$ which is assumed to be sampled at or above its Nyquist rate, with T being the sampling interval. The signal band is divided into N equally spaced contiguous sub-bands with center frequencies $\omega_k = 2\pi k/(NT)$, $k = 0, 1, \cdots, N-1$. The discrete-time STFT, $X(\omega_k, nT)$, is obtained for each value of k, by modulating $x(nT)$ by $\exp(-j\omega_k nT) = W_N^{-nk}$, where

$$W_N \triangleq \exp(j2\pi/N), \tag{28}$$

and filtering the modulated signal by the low-pass filter $h(nT)$, which is the sampled version of $w_k(t)$. Here all N prototype filters $w_k(t)$ are identical. Since $h(nT)$ is approximately band-limited to $\Delta\omega/2 = \pi/NT$, its output signal can be decimated. To reduce frequency-domain aliasing, and because of considerations related to the sign tracking algorithm used by the FDHS technique, the decimation factor R, an integer, typically satisfies $R < N$. The decimated STFT, $X(\omega_k, nRT)$, is modified according to the modification algorithms of the FDHS technique, using discretized forms of eqs. (17) and (27) for compression

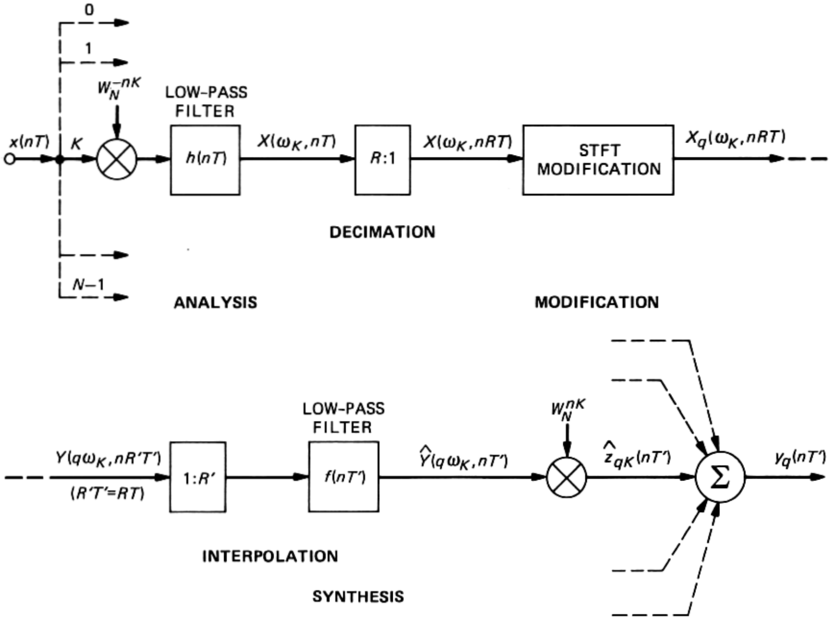

Fig. 6—Discrete-time form of the block diagram in Fig. 2, including decimation and interpolation of the sub-band signals.

and expansion, respectively. The modified STFT, $X_q(\omega_k, nRT)$, can be considered to be the STFT of the desired scaled signal at the scaled center frequency $q\omega_k$, and as being sampled at a rate which corresponds to the scaled signal bandwidth, i.e. with a sampling interval $T' = T/q$. It is suitable, therefore, to rename $X_q(\omega_k, nRT)$ as $Y(q\omega_k, nR'T')$, where R' is related to R through the requirement $R'T' = RT$ (i.e., $R' = qR$), so that the original time scale is maintained. For synthesizing the output scaled signal from the decimated and modified STFT signals, $Y(q\omega_k, nR'T')$, $k = 0, 1, \cdots, N-1$, these signals must first be interpolated by a factor R'. The interpolation is done in the scheme shown in Fig. 6 by inserting $(R'-1)$ zeroes between adjacent samples. This is represented by the box labeled $1{:}R'$ in Fig. 6. The result is processed with a low-pass filter $f(nT')$ having a nominal bandwidth of $\pi/(NT')$.

Since the interpolation is not ideal, we denote the interpolated STFT by $\hat{Y}(q\omega_k, nT')$. Modulation of the base-band signals with the complex sequence $\exp(jq\omega_k nT')$ results in the complex bandpass signals $\hat{z}_{qk}(nT')$, $k = 0, 1, \cdots, N-1$. Using $\omega_k = 2\pi k/NT$ and $T' = T/q$, we note that $\exp(jq\omega_k nT') = W_N^{nk}$, where W_N is defined in eq. (28). Thus, the input and output modulating sequences are complex conjugates of each other and are identical to the discrete transform kernels used in the DFT. Finally, by summing the N complex bandpass signals the frequency scaled output signal $\hat{y}_q(nT')$ is obtained.

Thus, we have seen that the general block diagram given in Fig. 2 can be implemented in the discrete-time form shown in Fig. 6. From the identity between this figure and Fig. 3 in Ref. 14, the more efficient WOLA scheme presented in Ref. 14 can be directly applied for the implementation of the block diagram in Fig. 6. However, as mentioned earlier, it is import t that frequency compression with the FDHS technique be performed with long duration window functions (longer than N). Hence, the scheme shown in Fig. 2 of Ref. 14 needs to be modified to accommodate this paticular situation. For clarity of presentation, and since the scheme shown in Fig. 2 of Ref. 14 can be used for expansion without any change, we briefly explain this scheme before we show its modification.

For convenience, we denote the analysis and synthesis window sequences by $h(n)$ and $f(n)$, respectively, dropping the explicit sampling intervals used in the previous notation. We now duplicate in Fig. 7 the scheme shown in Fig. 2 of Ref. 14, with some changes in notation to match the notation used in this paper.

According to this scheme, the input data samples are shifted into an input data buffer of length N, R samples at a time, corresponding to the R:1 decimation shown in Fig. 6. The input data block is weighted by $h(-n)$, which has its origin at the center of the block (see Fig. 7). The weighted data block is transformed using the FFT algorithm. The resulting STFT has its time reference at the beginning of the block (i.e., a sliding time reference); hence, a linear phase shift, corresponding to the time interval between the fixed time origin, say, $t_0 = 0$, and the beginning of the transformed data block, must be introduced. In addition, since the time origin of the analysis window is at the center of the block, a circular rotation of the weighted data by $N/2$ points is also needed.[14] Performing this rotation by phase modification in the frequency domain results in an overall phase modification by $(-1)^k W_N^{-nRk}$, $k = 0, 1, \cdots, N = 1$, as shown in Fig. 7. The result is the desired discrete-time STFT, $X(\omega_k, nT)$, in a fixed time reference, which is to be modified for frequency scaling. The modified STFT $Y(q\omega_k, nR'T')$ is translated back to the sliding time reference by applying the complementary phase modification $(-1)^k W_N^{nR'k}$. The output scaled signal is obtained by inverse transforming the modified STFT (in the sliding time reference), weighting the resulting data block by the synthesis window $f(n)$, and overlap-adding the weighted block to the output buffer. The output buffer shifts out R' samples for every R samples that are shifted into the input buffer. Also, to facilitate the overlap-add operation, R' zeroes are shifted into the output buffer as the processed signal samples are shifted out. Since the input and

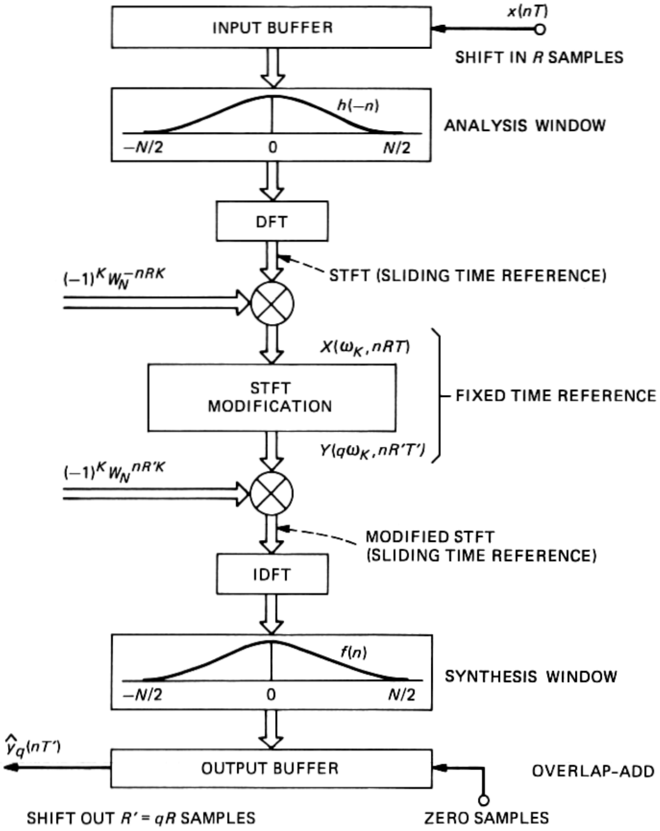

Fig. 7—A WOLA block implementation scheme for short-time Fourier analysis, modification, and synthesis.[14]

output sampling intervals are T and T', respectively, the time scale is not altered and the flow of data is uninterrupted. However, if time scaling is desired, the output data can be stored first and then replayed at the original sampling rate, resulting in a time-scaled signal (by the factor $q = R'/R$) which occupies the original frequency band.

We turn now to the more general situation in which $h(n)$ and $f(n)$ have longer durations than N samples. Let L_h and L_f denote the durations of $h(n)$ and $f(n)$, respectively, and let $L_h = m_h N$, $L_f = m_f N$, where m_h and m_f are positive integers. (If necessary, zeroes can be appended to the impulse responses to satisfy these conditions.) The appropriate scheme for this case is shown in Fig. 8 and is explained below.

It has been established in earlier works,[15,22,26] as well as repeated in Ref. 14, that if $L_h > N$, the N data points to be transformed are

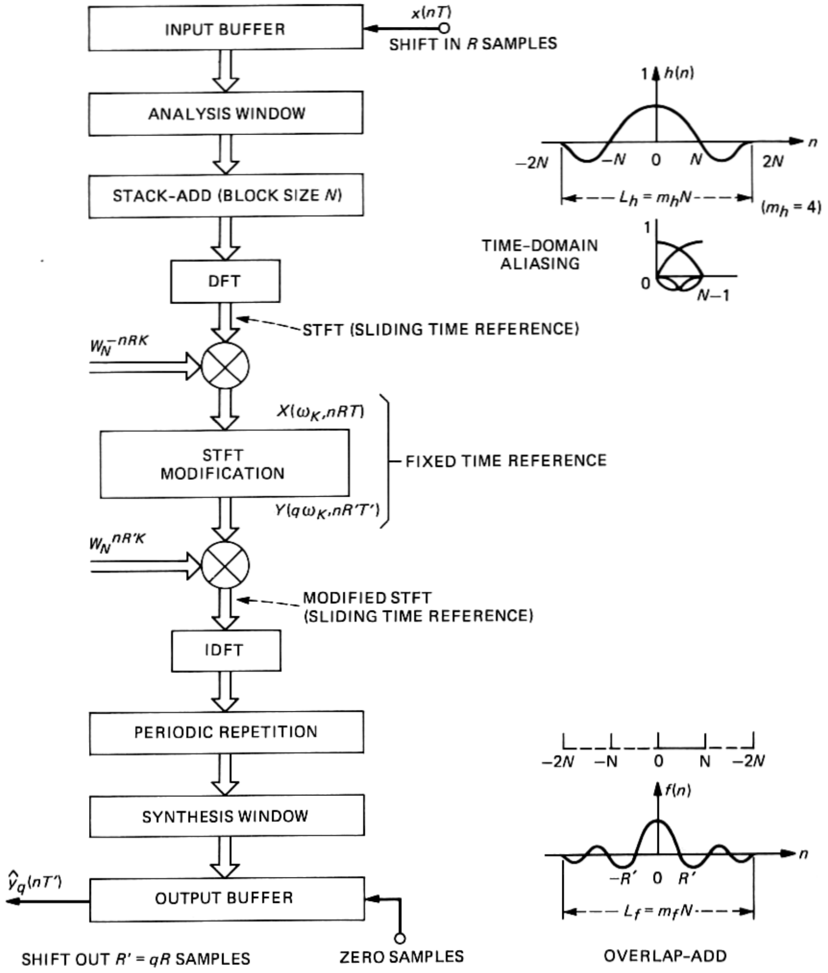

Fig. 8—Modified implementation scheme to accommodate analysis and synthesis windows which are longer than the transform block size N.

obtained by time aliasing the weighted L_h data points into N points. This can be seen as a stack-add operation of the m_h data segments, each of duration N, as shown in Fig. 8. Because of the stacking operation, the time origin of the transform is seen to be aligned with the time origin of the data window, and, hence, no circular rotation by N is needed; that is, there is no need to multiply the transformed data by $(-1)^k$. As before, the phase modification by W_N^{-nRk} is needed to obtain $X(\omega_k, nRT)$ in the fixed time reference. Following the STFT modification, the conversion of the sliding time reference, and the inverse transformation, N data points are obtained to which the synthesis window of duration $L_f = m_f N$ is to be applied. To do this, we periodically repeat the given block of N data points m_f times.* To be consistent with the analysis window weighting, the center of the synthesis window is aligned with the beginning of the given data block, as shown in Fig. 8. The weighted data is then overlapped-added to the output data buffer as before.

Before we conclude this section, several comments are in order. First, it should be noted that the scheme in Fig. 8 can be used also for the case considered in Fig. 7; that is, when the windows have a length which is less or equal to N. This is done by appending zeroes on each side of each window so as to make the duration of each $2N$ (i.e., $m_h = m_f = 2$). The stack-add operation will provide the circular shift by $N/2$ points, as needed in Fig. 7, and hence eliminate the need for multiplying by $(-1)^k$, as is the case in Fig. 8.

Second, it is observed that the implementation schemes presented above can, in principle, be used for scaling an input signal by any rational factor R'/R. However, for frequency compression with $q = 1/2$, the FDHS technique provides a particularly efficient realization of the STFT modification because the sign tracking algorithm avoids the need for explicit phase unwrapping. Expansion by integer factors is the simplest operation because the principal value of the phase can be used and the filter design considerations are simple. For other rational scaling factors it appears that one has to resort to phase unwrapping with its attendant complexity.

Third, in noncoding applications it may be desired to obtain a frequency compressed signal at the original sampling rate, i.e., oversampled. The needed interpolation can be embedded in the above implementation schemes. This is done simply by enlarging the modified STFT transform size ($1/q$ times) by padding with zeroes (at the center of the transform block). Following the inverse transform (IDFT) and the weighting by a suitable $f(n)$ of the longer data block, it is overlapped-added to the output data buffer. The data in both the input and output data buffers is accordingly shifted by R samples at a time.

Finally, it is of interest to point out that the scheme shown in Fig. 8 offers a generalization of the TDHS technique[4] as explained below.

Let us apply the principles of TDHS, as elaborated in Section III, to the scheme in Fig. 8. In principle, this means using a pitch-adaptive analysis filter bank—N is made equal to the pitch period and $h(n)$ is varied accordingly—and applying no modification to the STFT, except for scaling the center frequencies, as expressed by eq. (13) in Section III. Since the STFT is not modified, there is actually no need to use a

* This takes into account the underlying periodicity in the IDFT result because of the discretization in frequency. It can also be concluded from eq. (14) in Ref. 14.

transform—the phase shifts can be done by circular rotations in the time domain.[14] The part of the scheme in Fig. 8 between the stacking and the weighting by $f(n)$ collapses to a counter clock wise circular rotation in the time domain by $(R' - R)n$ Mod N_p samples, where N_p is the pitch period. The implementation scheme of the generalized TDHS technique is shown in Fig. 9.

The TDHS algorithms in Ref. 4 correspond to specific choices of R, R', and $f(n)$. For example, for 2:1 compression the values are $R = 2$, $R' = 1$ and $f(n)$ is a unit impulse; that is, $f(n) = \delta(n)$. The new generalized TDHS scheme above offers more flexibility through the possible use of more general synthesis windows, $f(n)$. It also could provide means for additional interband filtering such as $w_{qk}(t)$ in Fig. 2, which could improve the quality of the scaled speech signal, although at the expense of additional computation. This generalization of the TDHS technique is now under further investigation.

V. DESIGN CONSIDERATIONS AND SIMULATIONS

The implementation scheme in Fig. 8 applies both to compression and expansion. However, the type of frequency scaling performed affects the design requirements for the analysis/synthesis system. For this reason, we discuss below the design of the compression and expansion systems separately.

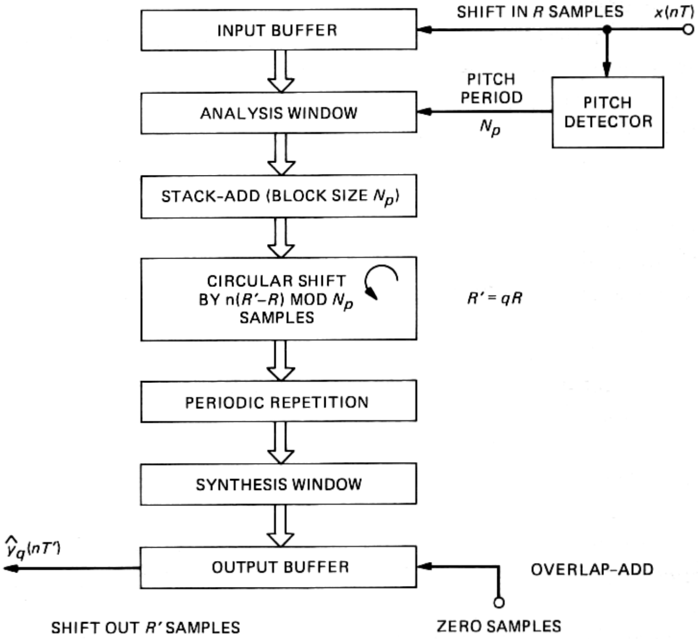

Fig. 9—Implementation scheme of the *generalized* TDHS technique.

5.1 Compression system design

A basic assumption in the development of the FDHS technique is that each filter in the analysis filter bank contains no more than one pitch harmonic. Hence, the number of filters in the filter bank should equal or exceed the largest number of harmonics expected in the given speech band. To accommodate low-pitch male speakers, we have chosen the number of filters to be $N = 128$. With a sampling rate of 8 kHz, which is typical for telephone bandwidth speech, this corresponds to a nominal frequency resolution of 62.5 Hz. This may not be sufficient resolution for very low-pitch speakers but was chosen as a compromise because of another conflicting requirement elaborated below. The prototype low-pass filter $h(n)$ has, therefore, a nominal bandwidth of 31.25 Hz. Besides the usual requirement that the transform of $h(n)$ have low side lobes to minimize interband leakage, it is extremely important that the transition band of $h(n)$ be as narrow as possible to minimize the overlap between the bandpass filters. The importance of minimizing the overlap stems from the difficulties in identifying bands which share a harmonic with an adjacent band and in matching the signs in those bands to avoid attenuation of the corresponding harmonic, as we discussed in detail in Section 3.1. By reducing the overlap between the filters, the probability of occurrence of this condition is lowered. In addition, it is also lowered by using the minimum number of filters necessary to separate the pitch harmonics. For this reason, we did not increase the transform size to 256 and have chosen N to be 128. For female voices, the best results are obtained with $N = 64$. However, since N is fixed the value of $N = 128$ was found to match both male and female voices. Other considerations in designing $h(n)$ are as follows.

In principle, one can reduce the transition-band width by increasing the filter length L_h. However, since speech is not a stationary signal and has a typical quasi-stationarity interval of several tens of ms, the length of $h(n)$ should be limited accordingly. Therefore, we have limited its length to 512 samples, i.e., to a duration of 64 ms for 8-kHz sampling rate, with the main lobe duration being 32 ms. With $L_h = 512$ we have $m_h = L_h/N = 4$ and, for this reason ($m_h > 1$), it was necessary to develop the scheme in Fig. 8, as an extension of the scheme in Fig. 7 from Ref. 14. To meet the requirements for low side lobes and a narrow transition band, we have chosen $h(n)$ to be an optimal equiripple filter[27] and designed it with the filter design package[28] available on our computer system.*

* The maximum filter length that can be designed on our system is 511. A 512-point filter is formally defined by appending a zero.

An additional requirement on the analysis filter bank is that its overall response be uniform. This way the two components of a harmonic which is shared by two filters will sum up to the correct magnitude. In terms of $h(n)$, a necessary and sufficient condition for the filter bank to be uniform is[22]

$$h(nN) = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases}. \tag{29}$$

While this condition is easily satisfied by windowing for example a $\sin(x)/x$ type response, the optimal equiripple filter does not necessarily satisfy this condition. Therefore, we had to use a trial and error approach and repeat the design several times until our design goals were met. Figure 10 presents the results of the final design which we denote by $h_0(n)$. The analysis window $h_0(n)$ is shown in (a) of Fig. 10, its frequency response in (b), and the composite filter bank frequency response in (c). The peak-to-peak ripple of 0.2 dB in the composite response is quite acceptable. The composite frequency response of a filter bank having a prototype low-pass filter $h(n)$ is simply found by transforming the sequence $d(n)$ defined by

$$d(n) = \begin{cases} h(n) & \text{if } n \text{ Mod } N = 0 \\ 0 & \text{otherwise} \end{cases}. \tag{30}$$

Before we consider issues related to the modification of the STFT signals, we would like first to consider the issues involved in designing a synthesis window $f(n)$ that provides an adequate reconstruction of the input signal when no spectral modifications are performed. Since we deal with the case where $L_h > N$ (i.e., $m_h > 1$), the synthesis filter has the difficult task of undoing the time aliasing, because of the stacking operation, as well as the frequency aliasing, because of the decimation of the signal in each band by the factor R.

It was shown in Refs. 16 and 24 that for exact reconstruction the following relation between $h(n)$ and $f(n)$ should hold

$$\sum_{s=-\infty}^{\infty} f(n - sR)h(pN - n + sR) = \delta(p), \quad \text{for all } n, \tag{31}$$

where $\delta(p) = 1$ for $p = 0$ and zero otherwise. If we assume that N is divisible by R, then eq. (31) is periodic in n with period R and constitutes a set of R conditions (for $n = 0, 1, \cdots, R-1$). In particular, for $R = N$, it is seen from eq. (31) that the design of $f(n)$ is equivalent to the problem of designing N inverse sub-filters.[16,24] Since $h(n)$ and $f(n)$ are both of finite duration (FIR), exact reconstruction cannot be obtained. However, if $R < N$ there is less aliasing in the frequency domain and the problem is relaxed. In view of the equivalent scheme in Fig. 6, we expect that for $R < N$ the use of a reasonably good

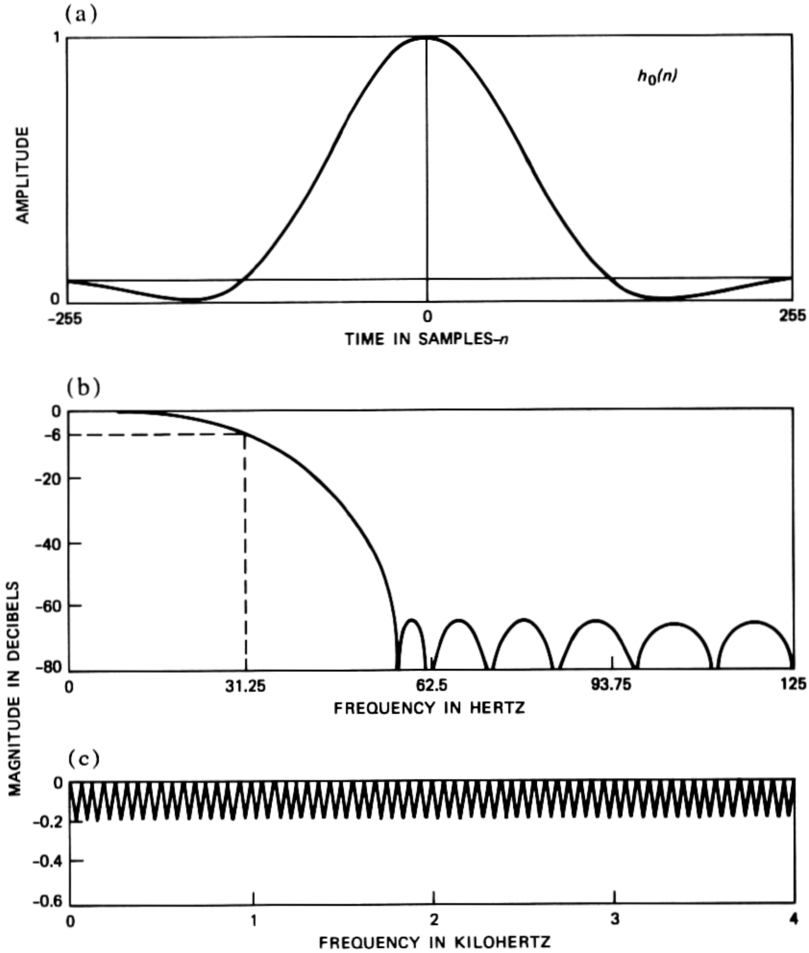

Fig. 10—Designed analysis prototype filter $h_0(n)$. (a) Impulse response. (b) Frequency response. (c) Composite filter-bank frequency response.

interpolation filter will also result in an acceptable reconstruction error. Therefore, we have initially selected four interpolation filters which have simple analytical respresentations. This facilitates the variation of their bandwidth by a change of a parameter. These windows are symmetrical and can also be made to have the usually desired property of interpolation filters, namely,

$$f(nR) = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases}. \tag{32}$$

The four synthesis window functions considered are a rectangular window $[w_R(n)]$, a Hanning window $[w_H(n)]$, a $\sin(x)/x$ function multiplied by a Hanning window $[w_{HS}(n)]$, and a special window function $[w_M(n)]$ which was derived in Ref. 4 and has some interesting properties.* The analytical representations of these windows are

$$w_R(n) = \begin{cases} 1 & |n| < L \\ 0 & |n| \geq L \end{cases};$$

$$w_H(n) = \begin{cases} \frac{1}{2}[1 + \cos(\pi n/L)] & |n| \leq L \\ 0 & |n| > L \end{cases}, \tag{33}$$

$$w_{HS}(n) = \begin{cases} \frac{1}{2}[1 + \cos(2\pi n/L_f)] \cdot [\sin(\pi n/L)/(\pi n/L)] & |n| \leq L_f/2 \\ 0 & |n| > L_f/2 \end{cases}, \tag{34}$$

and

$$w_M(n) = \begin{cases} \frac{L}{L_f} \sin(\pi n/L) \cot(\pi n/L_f) & |n| \leq L_f/2 \\ 0 & |n| > L_f/2 \end{cases}, \tag{35}$$

where L is an integer which determines the bandwidth of the synthesis window and L_f is its length, which is assumed to be an even multiple of L. In particular, the rectangular and Hanning windows in eq. (33) have a duration of $L_f = 2L$. Note that if L is chosen to be equal to R, then eq. (32) is satisfied. The above four window functions are shown in Fig. 11 for several values of L_f/L. The reconstruction error expected from using these synthesis windows, with the analysis window $h(n) = h_0(n)$ shown in Fig. 10, was found by evaluating the left-hand side of eq. (31) and computing its deviation from the desired value on the right $[\delta(p)]$. An average mean-square error (mse) was defined as follows. Let $V_n(p)$, n = 0, 1, $\cdots$, $R - 1$ be defined by the left-hand side of eq. (31) under the assumption that N is an integer multiple of R and let

$$\epsilon_n^2 \triangleq \sum_p [V_n(p) - G\,\delta(p)]^2 \quad n = 0, 1, \cdots, R-1, \tag{36}$$

* This window is described by eq. (53) in Ref. 4 (even case) and has the interesting property of satisfying an equation like (32) in both the time and frequency domains. Thus, as a prototype filter it results in a uniform filter-bank response, whereas in a WOLA-type operation it results in a uniform time response [see eq. (52) in Ref. 4]. For a particular choice of parameters, it becomes identical to the well-known Hanning window.[4]

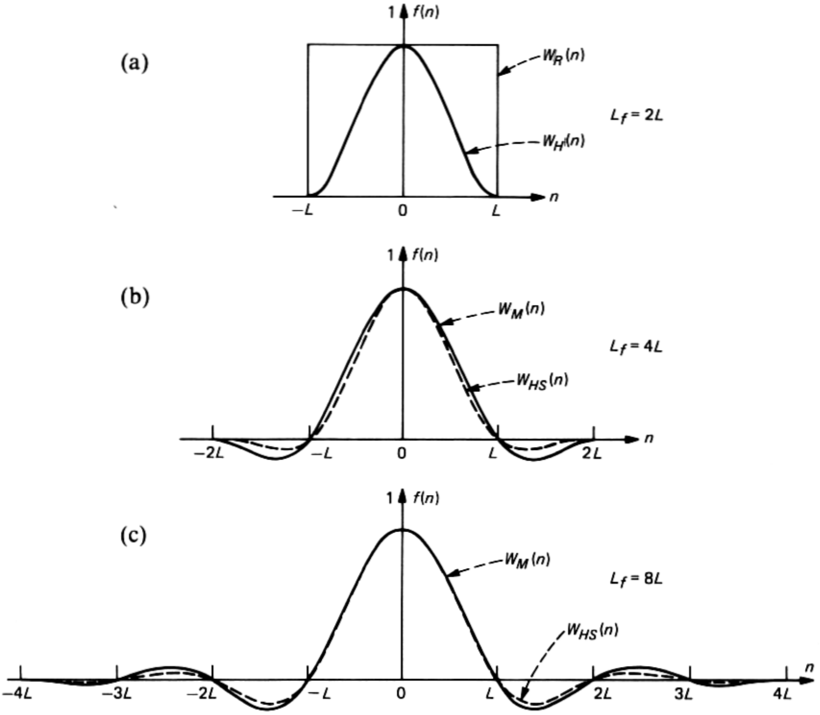

Fig. 11—Synthesis window functions. (a) Rectangular [$w_R(n)$] and Hanning [$w_H(n)$] windows. (b) $w_M(n)$ and $w_{HS}(n)$ windows for $L_f = 4L$. (c) w_M and w_{HS} windows for $L_f = 8L$.

where G is a constant given by

$$G = \frac{1}{R} \sum_{n=0}^{R-1} V_n(0), \tag{37}$$

and, in general, is not equal to 1 as assumed in eq. (31). We have introduced G since we consider a reconstruction which differs from the original signal by a constant gain term as being errorless. An average mse is now defined by

$$\epsilon^2 = \frac{1}{R} \sum_{n=0}^{R-1} \epsilon_n^2. \tag{38}$$

It can be shown that the choice of G according to eq. (37) minimizes ϵ^2; that is, $\partial\epsilon^2/\partial G = 0$. The values of ϵ^2, in dB, which were obtained for the above synthesis windows, using the analysis window $h_0(n)$ are shown in Fig. 12 as a function of R, which is given in terms of the transform size N. For each value of R, ϵ^2 was computed for different values of L. The minimum value of ϵ^2 so obtained is shown in Fig. 12.

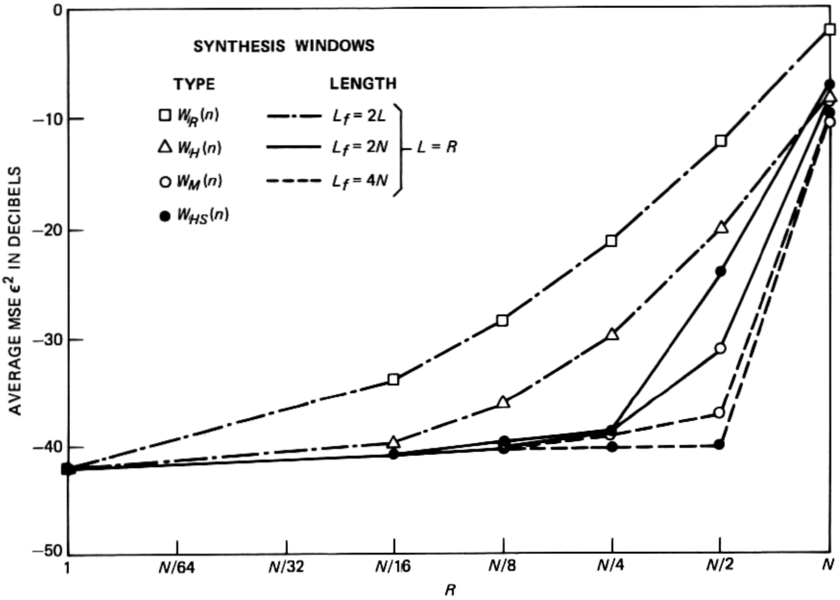

Fig. 12—Minimum values of the average mse ϵ^2 for different synthesis windows, paired with the analysis window $h_0(n)$ in (a) of Fig. 10, as a function of the decimation factor R, or input data shift. N is the transform block size.

In fact, for the above windows, the minimum is attained when $L = R$; that is, eq. (32) is satisfied.

Note also that in Fig. 12 the length of the synthesis windows $w_R(n)$ and $w_H(n)$ varies with R, since $L = R$ and $L_f = 2L$. However, the length of $w_{HS}(n)$ and $w_M(n)$ is assumed to be fixed—either $L_f = 2N$ or $L_f = 4N$—but the bandwidth still varies with R since $L = R$. It is seen from Fig. 12 that with all four windows R must be less than N. To save computation we wish, of course, to use the largest possible value of R. For $R = N/2$ and $L_f = 4N = 512$ the preferred window is $w_{HS}(n)$, whereas for $R = N/2$ and $L_f = 2N$ it is $w_M(n)$ (in both cases $L = R = 64$). To reduce the output buffer size and to save computation, we preferred using $L_f = 2N$. The resulting reconstruction error (below −30 dB) is quite acceptable. The other two windows [$w_R(n)$ and $w_H(n)$] result in reconstruction errors too high to be useful with this value of R.

The above results are for an analysis-synthesis system without the modification of the STFT signals. Since the frequency scaling affects the output sampling rate (the interpolation factor becomes $R' = qR$, where q is the scaling factor) and the sign tracking algorithm limits the maximum value of R (as elaborated later), the choice of R and the synthesis window function must be carefully reconsidered to account for the spectral modification.

For proper operation of the sign tracking algorithm, the phase in each frequency cell or band should not change by more than $\pi/2$ between observations in order that the crossing of the complex signal vector from one quadrant to another will be accounted for and the signs of its real and imaginary parts be tracked correctly. The maximum phase change occurs when a harmonic has the largest deviation $\Delta\Omega_{\max}$ from the center frequency. Taking in account the overlap between filters, we let $\Delta\Omega_{\max} = \Delta\omega$, which is the difference between adjacent center frequencies. During R samples, the phase can change by up to $\Delta\omega RT$ and, hence, with $\Delta\omega = 2\pi/(NT)$ we get the condition that $2\pi R/N \leq \pi/2$; that is, $R \leq N/4$. Therefore, with $N = 128$ we have $R \leq 32$, and to save computation we choose $R = 32$.

It is noted that this choice of R was dictated by the FDHS modification technique used and not by the analysis-synthesis system. However, while a transform of the input data must be taken every 32 samples for proper sign tracking, the modified STFT can actually be computed at a lower rate corresponding to a decimation factor $R_M > R$. This obtains because the analysis-synthesis system is capable of acceptable reconstruction for larger values of the decimation factor. Furthermore, since $q = 1/2$ (2:1 compression), the interpolation factor, or the output data shift, is $R' = R_M/2$ so that the modified STFT needs to be computed only every $N = 128$ samples. This corresponds to $R' = N/2 = 64$ which, as shown earlier, can be accommodated by the analysis-synthesis system with an acceptable reconstruction error.

In summary, the following windows and parameters are used. The transform size is $N = 128$; the analysis window is $h(n) = h_0(n)$ of Fig. 10 ($L_h = 512$); the input data shift is $R = N/4 = 32$; the STFT modification is computed every $R_M = N = 128$ samples; the output data shift is $R' = R_M/2 = N/2 = 64$; and the synthesis window chosen is $f(n) = w_M(n)$ of eq. (35), with $L_f = 2N = 256$, and $L = R' = 64$. This selection of parameters and window functions is supported by simulation results with both synthetic and natural speech signals as will be illustrated subsequently.

Before we present the design considerations for the expansion system, we wish to explain an additional issue related to the selection of the synthesis window and give details of the STFT modification. In Fig. 2, the synthesis filters were assumed to have a bandwidth which is q times the bandwidth of the analysis filters. In the implementation described above, this corresponds to using synthesis filters with $L = N$ and not $L = R'$ as we have chosen, which widens the synthesis filter bandwidth since $R' < N$. However, with practical synthesis filters the use of $L = N$ was found to cause inband attenuation and an increase in the reconstruction error. For $R' = N/2 = 64$, the use of the above-selected window functions, $w_M(n)$ for $L_f = 256$ or $w_{HS}(n)$ for $L_f = 512$

with $L = R'$ gave better or as good results as several other window functions having bandwidth in the normalized frequency range of π/R' to π/N. However, for smaller values of R' an advantage was found in using $L = N/2 > R'$ because of the additional filtering provided by the synthesis window.

We discuss now the details of the STFT modification for frequency compression. The modification is performed in each band according to the expressions in eq. (17) at time instants $t = nR_MT$, where R_M is the modification decimation factor, chosen to be $R_M = N = 128$. The sign tracking is done according to eq. (18) at a higher rate, namely at time instants $t = nRT$, with $R = N/4 = 32$. The condition for sign initialization in each band is determined by eq. (25). The ratio on the left-hand side of eq. (25) is computed at time instants $t = t_n = nT_0$, where $T_0 = NT$, which is also the STFT modification update interval since $R_M = N$. If the ratio exceeds the threshold value E_T in a given band, the signs in that band are initialized. On the basis of simulations, E_T was chosen to correspond to an energy increase of 10 dB during the time interval $T_0 = 16$ ms for 8-kHz sampling rate. This choice of E_T was found to indicate quite reliably the onsets of speech, the transitions from unvoiced to voiced, and the crossing of a pitch harmonic into a given band from a neighboring band. At the same time, this value was found to be sufficiently high to prevent reinitialization because of the normal amplitude fluctuations in each band during sustained voiced intervals.

If the initialization condition is met, the initialization process consists, on the basis of the discussion in Section 3.1, of the following operations. First, the signs in all the cells, or bands, which need to be initialized are set according to eq. (19); that is, according to the principal value of the phase in each cell. Then, starting from the lowest frequency cell to be initialized, a "pairing" process is performed. That is, for each cell to be initialized one of its adjacent cells is picked for sign matching to provide for the situation in which a pitch harmonic is shared by two adjacent bands. This is done either by picking the band which gives an $R(t_n)$ [see eq. (20)] with a smaller phase angle (see Section 3.1) or, as preferred in our simulations and to save computation, by picking the band in which the signal magnitude is larger. Note that if cell k_0 is to be initialized and cell $k_0 + 1$ is picked for sign matching, then, even if cell $k_0 + 1$ is also to be initialized, this cell is skipped in the pairing and matching process since it was already chosen to be matched to cell k_0. Finally, the sign matching of the chosen pairs is performed by computing S_R for each pair, using eqs.(23) and (24), and inverting the signs in one of the bands of those pairs for which S_R is found to have a negative value.

5.2 Expansion system design

Expansion by an integer factor avoids the phase ambiguity problems and is, therefore, a relatively easy task. The main requirement is sufficient frequency resolution in the spectrum analysis. Since there is no concern if the filters overlap, the resolution requirement is easily fulfilled by using even the simple Hanning window $h(n) = [1 + \cos(\pi n/N)]/2$, $n \in [-N/2, N/2]$, with $L_h = N = 256$. The Hanning window also satisfies eq. (29) and, therefore, the resulting filter bank is uniform. Without the STFT modification, the analysis-synthesis system can now be made a unity system—no reconstruction error—since eq. (31) can be exactly satisfied. In particular, if $f(n)$ is a rectangular window of length N, eq. (31) becomes

$$\sum_{s=-L_R}^{L_R-1} h(sR + n) = 1 \qquad n = 0, 1, \cdots, R - 1, \tag{39}$$

where $L_R = N/R$, with R being assumed again to divide N. The condition in eq. (39) is satisfied within a constant gain factor by the Hanning window for any $R \leq N/2$ which divides N. The gain factor is $L_R/2$. The analysis-synthesis implementation then becomes the well-known overlap-add (OLA) implementation.[14,15,29]

Let us now consider the effect of STFT modification for frequency expansion on the analysis-synthesis system design. Since in this case $q = 2$, so that $R' = 2R$, limiting R' to $N/2$ requires limiting R to $N/4$. However, since the rectangular synthesis window has high sidelobes in its frequency response, it did not provide sufficiently good speech quality when frequency expansion was performed. To improve the filtering provided by $f(n)$, we have considered again using the window functions in eqs. (34) and (35). Very good quality expanded speech was obtained with $R = N/8 = 32$ and $w_M(n)$ of eq. (35), using $L_f = N = 256$ and $L = N/2 = 128$. For this choice of parameters, $w_M(n)$ becomes identical to the Hanning window $w_H(n)$. Note that the output data shift or interpolation factor is $R' = 2R = N/4$, but $L = N/2$, i.e., $L > R'$, which is a condition that provides for additional filtering as discussed in Section 5.1.

To summarize, the expansion system is implemented with a transform size of $N = 256$ and analysis and synthesis windows that are both Hanning windows of length N. The input and output data shifts are given by $R = N/8 = 32$ and $R' = 2R = 64$, respectively, and the STFT modification is done according to eq. (27) at time instants $t = nRT$, i.e., every $R = 32$ samples.

5.3 Simulations

The FDHS system was simulated on a laboratory computer which is equipped with an integral array processor (Data General-Eclipse AP/130). The array processor facilitated fast computation of the needed array operations, such as transforms, windowing, stack-adding, and overlap-adding. The input signal was generally telephone bandwidth speech (200–3200 Hz) sampled at 8 kHz. In addition to natural speech input, a synthetic vowel with fixed pitch was used. It was synthesized by periodically repeating a single pitch period (51 samples) from the speech of a male speaker. This synthetic signal was valuable in the development of the system, in checking assumptions, and in selecting parameters and window functions. By way of illustration, and to support our selection of parameters, Fig. 13 shows in part (a) the spectrum of the synthetic vowel, and in parts (b) to (l) the spectra obtained for different windows and parameter values, as detailed in the figure caption and explained below. For reference, an ideally

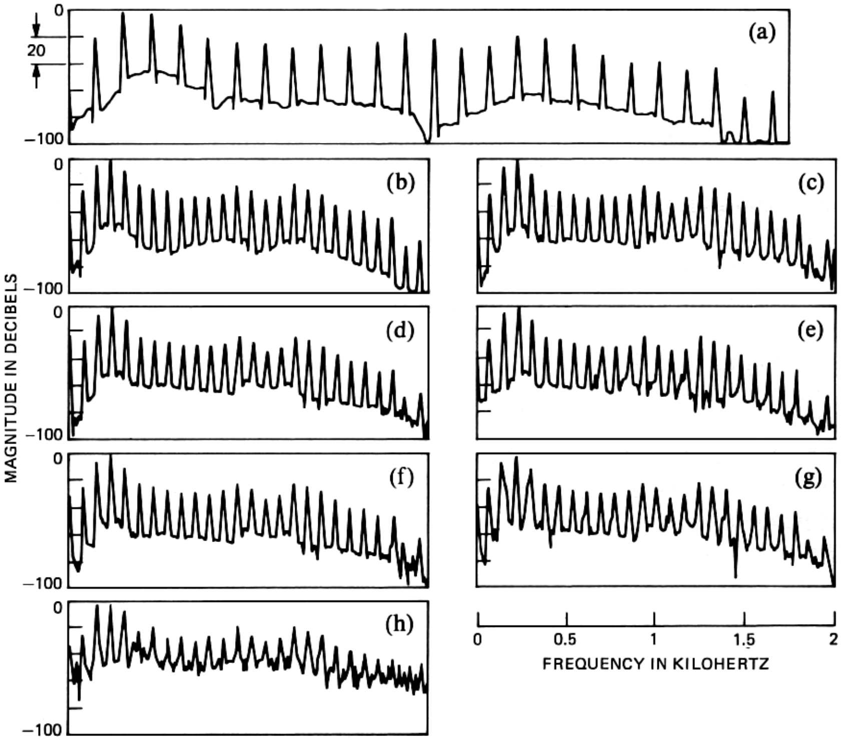

Fig. 13—Spectral representation of original and processed synthetic vowel for different system parameters and window functions. (a) Original. (b) Ideal 2:1 compression. (c) Compression with the selected system parameters: $N = 128$, $h(n) = h_0(n)$, $R = 32$, $R_M = 128$, $R' = 64$, $f(n) = w_M(n)$ with $L_f = 256$, $L = 64$. (d) Same as in (c), except $R_M = R = 32$. (e) Same as in (c), except $R_M = R = 64$. (f) Same as in (c) but $f(n) = w_{HS}(n)$ with $L_f = 512$, $L = 64$. (g) Same as in (c) but $f(n) = w_H(n)$ with $L = 64$ ($L_f = 128$). (h) Same as in (c) but $f(n) = w_R(n)$ with $L = 64$ ($L_f = 128$).

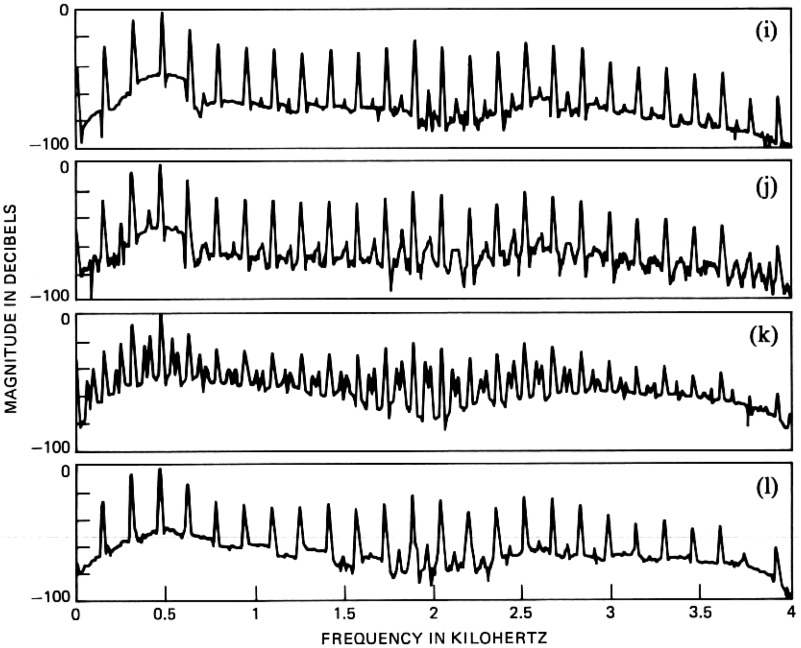

Fig. 13—(Contd.) (i) Expansion of the ideally compressed signal in (b) using the selected expansion system parameters: $h(n) = w_H(n)$ with $L = 128$, $(L_f = N = 256)$, $R = 32$, $R' = 64$, $f(n) = w_H(n)$ with $L = 128$, $(L_f = 256)$. (j) Same as in (i) but with $f(n) = w_R(n)$ $(L = 128)$ and $R = 64$. (k) Same as (i) except $R = 64$. (l) Expansion of the compressed signal in (c) using the selected system parameters as in (i).

compressed spectrum is shown in part (b). Since the synthetic signal is periodic with a known period, ideal 2:1 compression can be obtained by discarding every other period and reducing the sampling rate by a factor of 2. Practically, (b) is obtained by windowing (we used a Hamming window for the spectral analysis) a segment of speech with half the number of samples than in the segment analyzed for producing the spectra in (a). Since the input signal is exactly periodic, the shape of the spectral teeth is determined only by the frequency response of the window. In natural speech, which is only quasiperiodic, the amplitude and phase modulations of the pitch harmonics, as discussed in Section II, widen further the spectral teeth. This can be observed in later illustrations.

In (c) of Fig. 13 the spectrum of the compressed signal, using the selected system parameters detailed in Section 5.1, is shown. Because the signal is purely periodic, the results are particularly good. A frequency domain signal-to-distortion ratio (SDR) computation* results

* The frequency-domain distortion measure used is based on the mse between the spectral envelopes of the input and processed signals, as measured in sub-bands with a

here in SDR = 42 dB. To illustrate that no harm is done by using $R_M > R$ (see Section 5.1), part (d) shows, for comparison with (c), the result of using $R_M = R = 32$. The difference is minute—only 0.1 dB higher SDR between the processed and ideal spectrum. Part (e) shows the spectrum obtained when $R = R_M = 64$ is used. The problem in this case is that for some harmonics the sign tracking is inadequate because R is too large. The effect on the spectra is clearly seen. Part (f) shows the result obtained when a longer, $L_f = 512$, synthesis window is used. On the basis of Fig. 12, $w_{HS}(n)$ is now preferred and used. The results are somewhat better than in (c), an improvement of 2 dB in SDR, but this does not seem to justify the doubling of the output buffer size and the additional computation. Parts (g) and (h) reaffirm the unsuitability of the simple Hanning and rectangular windows, respectively, as synthesis windows in the given compression system. The loss in SDR amounts to 18 and 28 dB, respectively, as compared to (c).

The remaining parts of Fig. 13 illustrate some of the results obtained with the expansion system. Part (i) shows the spectrum that results from expanding the ideally compressed signal in (b), using the selected system parameters as detailed in Section 5.2. As expected, the results are better than for compression and the frequency-domain SDR is above 60 dB. Parts (j) and (k) show the results obtained with rectangular and Hanning synthesis windows, respectively, using $R = 64$. It is clearly seen that this value of R is too large and its use results in significant SDR reductions for both windows—up to 35-dB reduction for the Hanning window, as compared to (i). For $R = 32$, the selected Hanning window performs best, as seen in (i), offering an 8-dB advantage in SDR over the rectangular window.

Finally, part (l) shows the resulting spectrum when the compressed signal in (c) is expanded back, using the selected system parameters as in (i). The reduction in SDR because of the expansion process was found to be only about 0.5 dB; that is, a 41.5-dB SDR is obtained when the spectrum in (l) is compared to the spectrum in (a).

The importance of correct sign initialization is demonstrated in Fig. 14. In addition to showing the ideally compressed spectrum of the synthetic vowel in (a)—identical to part (b) of Fig. 13—it also shows the spectrum of the frequency compressed signal obtained with three different sign initialization approaches. Part (b) of Fig. 14 shows the result obtained when all the signs are initialized to be positive; part (c), when the signs are initialized according to eq. (19) (i.e., using only

bandwidth that increases with frequency, similar to articulation index measurements. The actual band allocation used was taken from Ref. 31. This measure was found useful for the synthetic signal, as it correlated well with our observations. For natural speech, however, this was not so and we had to rely on subjective listening and visual examination of the spectral representations.

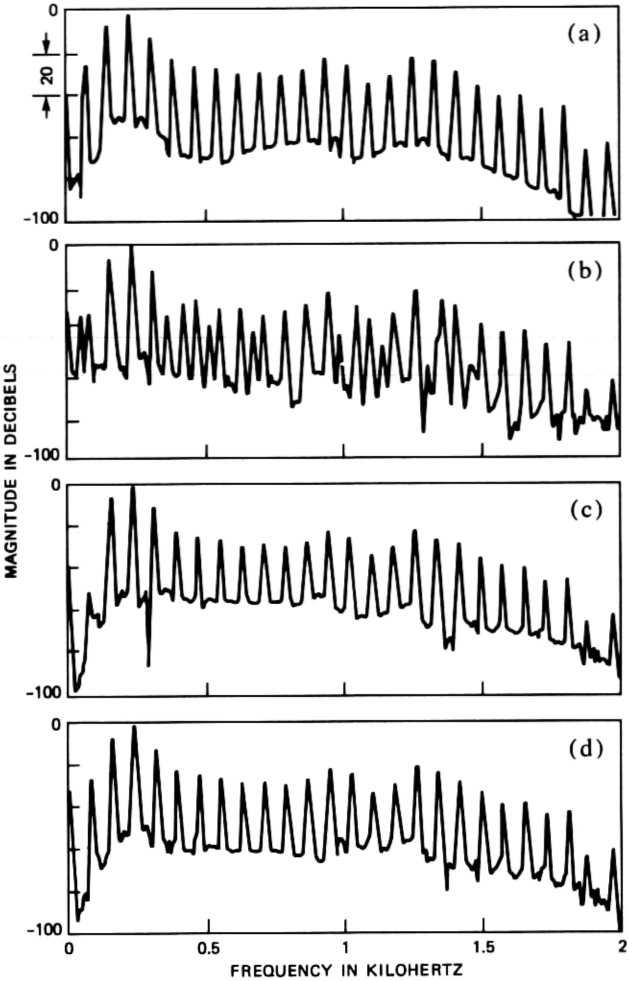

Fig. 14—Demonstration of the effect of different sign initialization approaches. (a) Spectrum of the ideally compressed synthetic vowel. (b) All-positive sign initialization. (c) Sign initialization according to principal value of the phase in each band. (d) Sign initialization according to the algorithm developed in Section 3.1.

the principal value of the phase); and part (d), according to the final initialization algorithm which includes the pairing and matching operations described earlier and also shown in (c) of Fig. 13. The results indeed validate the assumptions and analysis performed in Section III—namely, the possible generation of undesired spectral components, in addition to the attenuation of the desired components. Part (b) shows the result when a random or all-positive sign initialization is used; part (c), the possible cancellation of harmonics because of incorrect matching of the signs in two adjacent bands which share the same harmonic; and part (d), the improved results obtained by using the developed sign initialization algorithm.

Similar results were obtained with natural speech, although errors in the initialization do occur due to speech nonstationarity and deviations from the harmonic model. Yet, in all the simulations performed, the developed sign initialization algorithm always resulted in better speech quality. As an illustration, Fig. 15 shows the spectra obtained for a segment of voiced speech. Parts (a) to (c) show the spectra of the original, compressed, and reconstructed signal using the developed FDHS system. For comparison, parts (d) and (e) show the corresponding

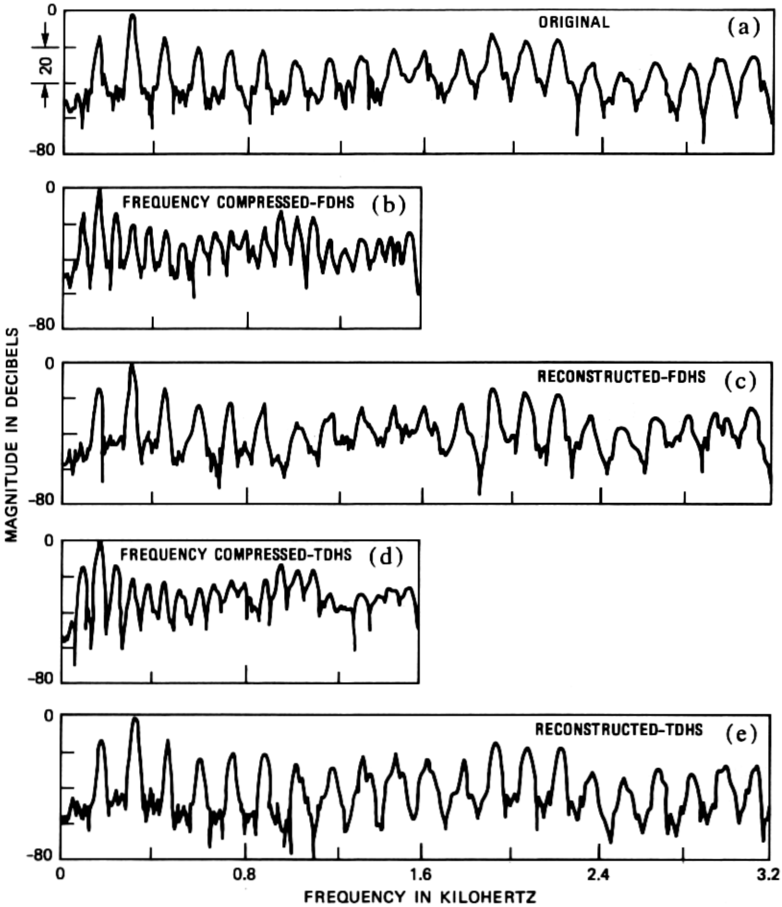

Fig. 15—Spectral representation of original and processed voiced speech segment for male speaker. (a) Original. (b) Compressed 2:1 using FDHS. (c) Expansion of compressed signal (reconstructed) using FDHS. (d), (e) Same as (b) and (c), respectively, but using TDHS.

results obtained with the time-domain harmonic scaling (TDHS) technique[4] using a cepstral pitch detector[30] implemented on the same array processor. The TDHS system was judged to give higher quality speech. Further discussion on the comparison between the two systems is given later. For an additional demonstration of the results obtained by the two systems, Fig. 16 shows the corresponding time waveforms, and Fig. 17 presents spectrograms of the complete processed sentence. To

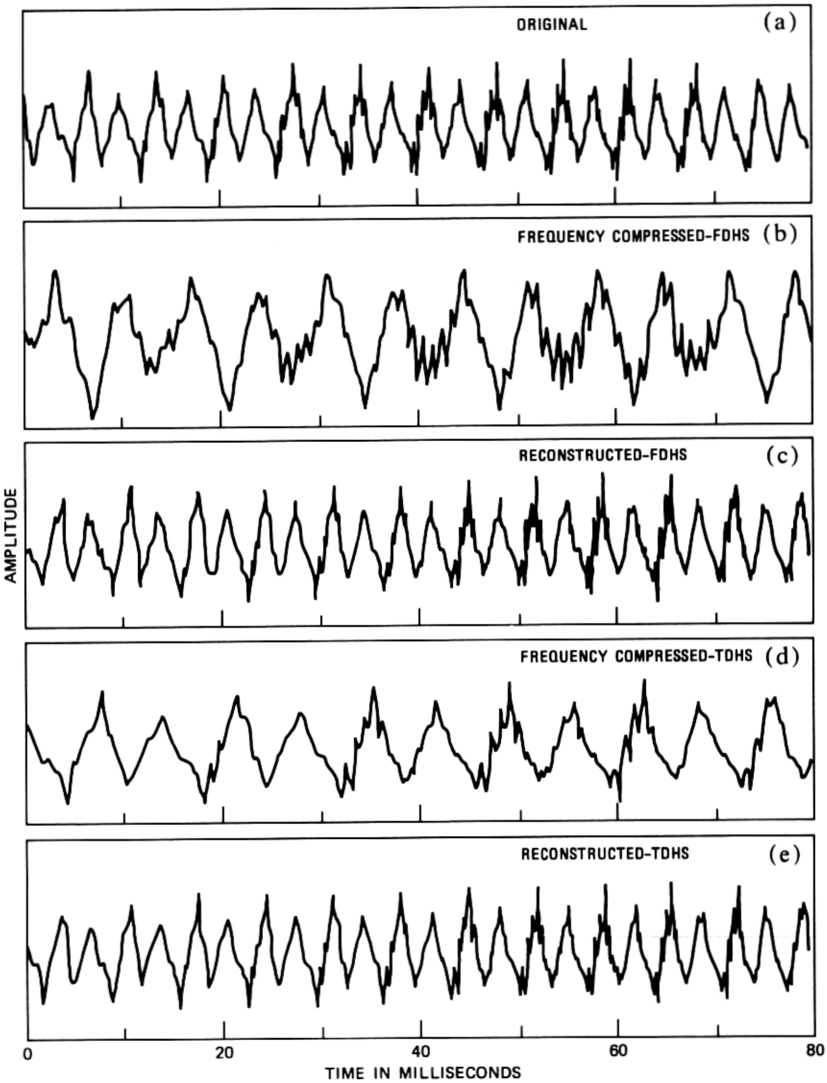

Fig. 16—Time-domain waveforms corresponding to the spectral representations given in Fig. 15.

illustrate the capability of the two systems to change the time scale, without changing the frequency scale, parts (f) and (g) of Fig. 17 show spectrograms of the time-compressed signals.

5.3.1 Performance with degraded inputs

To examine the robustness of the FDHS technique to adverse acoustical environment conditions, we ran simulations with noisy speech signals (down to 0-dB s/n), speech with severe room reverberation, and speech from three speakers speaking simultaneously. In simulations with noisy speech, the FDHS system appeared to be quite robust

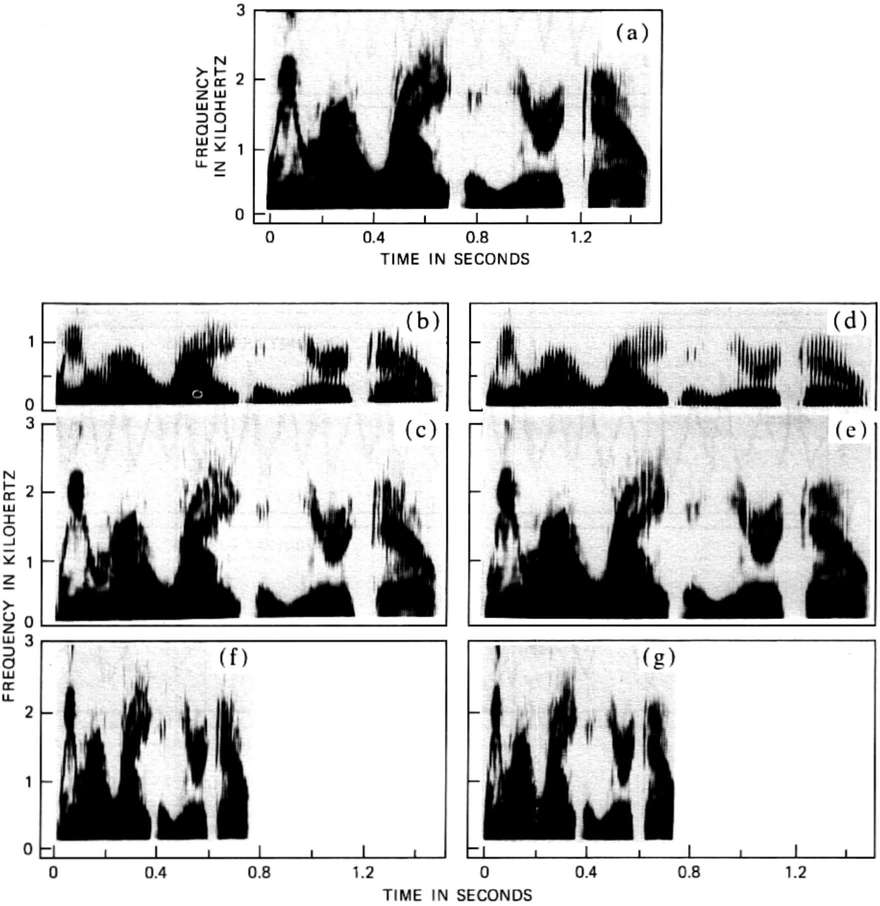

Fig. 17—Spectrograms of original and processed speech by FDHS and TDHS systems ("We were away a year ago," male speaker.) (a) Original. (b) Frequency compressed—FDHS. (c) Frequency-expanded (reconstructed)—FDHS. (d) Frequency-compressed—TDHS. (e) Frequency-expanded (reconstructed)—TDHS. (f) Time-compressed—FDHS. (g) Time-compressed—TDHS.

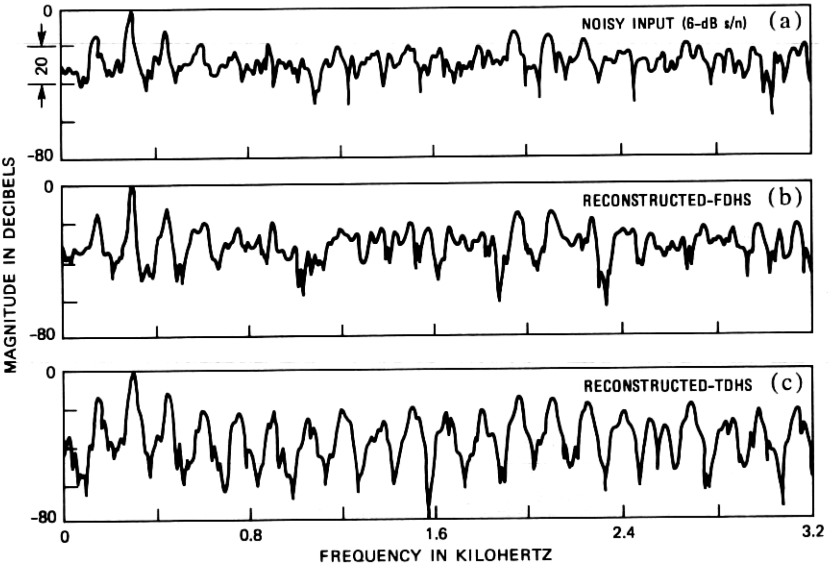

Fig. 18—Spectral representation of original and processed noisy voiced speech segment [white noise at 6-dB s/n added to signal in (a) of Fig. 16]. (a) Original. (b) Reconstructed—FDHS. (c) Reconstructed—TDHS.

with only some reduction in signal crispness as compared to processing clean speech. No change in the general nature of the noise was observed. This is in contrast to results obtained with the TDHS technique which tends to structure the noise caused by the pitch-dependent, time-domain weighting process. This is illustrated in Fig. 18 which shows the spectra of the original noisy signal (white noise at 6-dB s/n) and the reconstructed signals by the FDHS and TDHS systems. The structuring of the spectra, according to the speech pitch, caused by the TDHS technique is evident. Although this structuring provides a filtering effect similar to comb filtering, the structured noise is usually more annoying to listen to. For further illustration, Fig. 19 presents spectrograms of the complete processed noisy sentence by the two systems.

The results of processing speech with severe room reverberation, and multiple speakers speech, indicate that the FDHS system is highly robust to these conditions. In fact, these conditions tend to mask processing artifacts and the reconstructed signal sounds very similar to the original. In the TDHS system, the structuring of the uncorrelated reverberation components is perceivable. On the other hand, the TDHS system was found to be quite robust to multiple speakers speech.[5] Spectrograms of a processed sentence, recorded with severe room reverberation, are shown in Fig. 20, for the two systems.

To summarize, by the above simulation results we have demonstrated the validity of our assumptions in deriving the sign initialization algorithm and the selection of the analysis-synthesis window functions and parameters. The FDHS system was found to be robust to environment conditions, although its quality for clean speech signals is judged to be lower than with the TDHS. On the other hand, the robustness of the TDHS system is largely dependent on the type of pitch detector used (particularly with noisy signals), and it suffers from artifacts

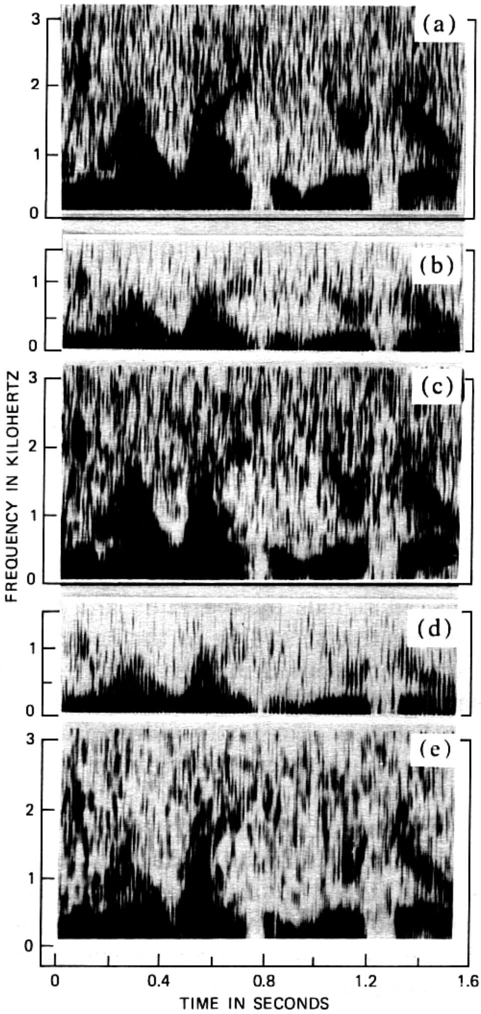

Fig. 19—Spectrograms of original and processed noisy speech by FDHS and TDHS systems. Parts (a) through (e) as in Fig. 17, but with white noise at 6-dB s/n added to original speech signal.

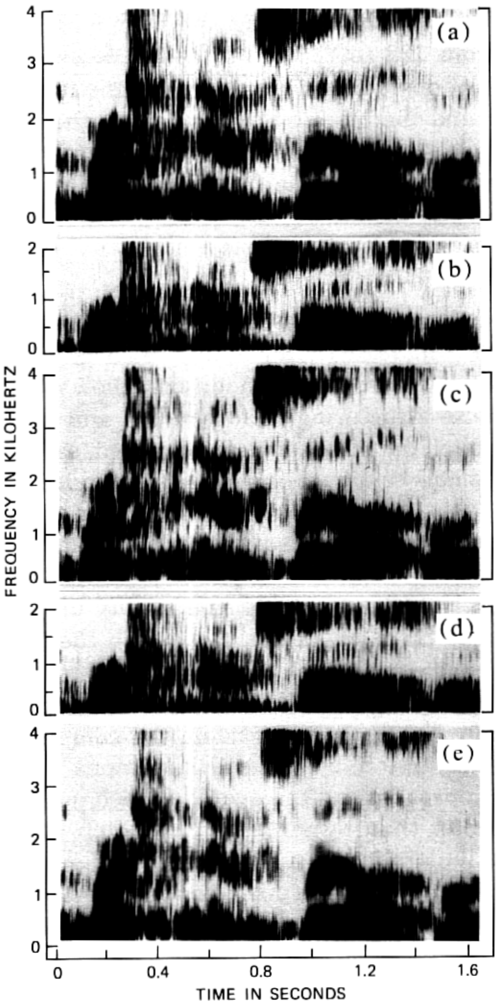

Fig. 20—Spectrogram of processed speech with severe room reverberation by FDHS and TDHS systems ["The birch canoe slid on the smooth (plank)," male speaker]. (a) Input signal. (b) Compressed—FDHS. (c) Reconstructed—FDHS. (d) Compressed—TDHS. (e) Reconstructed—TDHS.

introduced by structuring wide-band noise or uncorrelated reverberation.

It should be mentioned, however, that while the FDHS system, in general, provided good communications speech quality, an increase in reconstructed signal degradation was found for some low-pitch male speakers, typically below 80 Hz. One possible reason is insufficient frequency resolution of the analysis filter bank. At the expense of a slight input bandwidth reduction, this condition can be corrected somewhat by reducing the sampling rate from 8 kHz to 6.4 kHz since increasing N from 128 to 256 is not desirable as explained earlier. Another source of degradation appears to be the fact that if the pitch teeth are wide, and the pitch frequency is low, the original speech is characterized by a small separation of the pitch teeth, and the model assumed in the derivation of the FDHS technique is not as applicable. Further study of this problem is needed.

VI. A HYBRID TDHS-FDHS SYSTEM

The simulation results presented in Section V and our earlier experience with TDHS[4-6] indicate the advantage of TDHS over FDHS, provided that the acoustical environment conditions allow adequate pitch extraction, the noise structuring is acceptable, and pitch data can be transmitted. If pitch data cannot be transmitted, as would be the case with analog channels or digital channels with tandeming of waveform coders, or if it is not desirable to transmit the data, use of TDHS requires reextraction of the pitch at the receiver. Since pitch extraction from the compressed signal is more difficult because fewer pitch periods per unit time are available, the quality of the reconstructed signal is degraded. Since expansion alone with the FDHS system provides almost transparent speech quality without the need for explicit pitch extraction, we examined the possibility of using a hybrid system, such as shown in Fig. 21. In this system the compression is done by TDHS and the expansion by FDHS. Simulations using this system supported this approach, and the overall speech quality obtained was judged to be better than by TDHS without pitch transmission or by FDHS alone. For illustration, Fig. 22 shows spectrograms of a processed sentence by the different systems. It should also be noted that the proposed hybrid system has an advantage with noisy signals as well, as long as pitch extraction at the transmitter is feasible. The advantage is that the structuring of the noise in the reconstructed signal is avoided, since this structuring occurs mainly in the expansion stage of the TDHS, which is replaced by FDHS in the hybrid system. Furthermore, the hybrid system should also be more tolerant to channel errors than TDHS alone since these errors appear as noise in the compressed signal which does not affect the FDHS expansion system much, but could obviously affect the TDHS expansion process.

VII. CONCLUSION

A unified description of several frequency scaling techniques has been given in terms of the short-time spectral modifications which they produce. This description helps in understanding the properties and limitations of these techniques and their relation to FDHS.

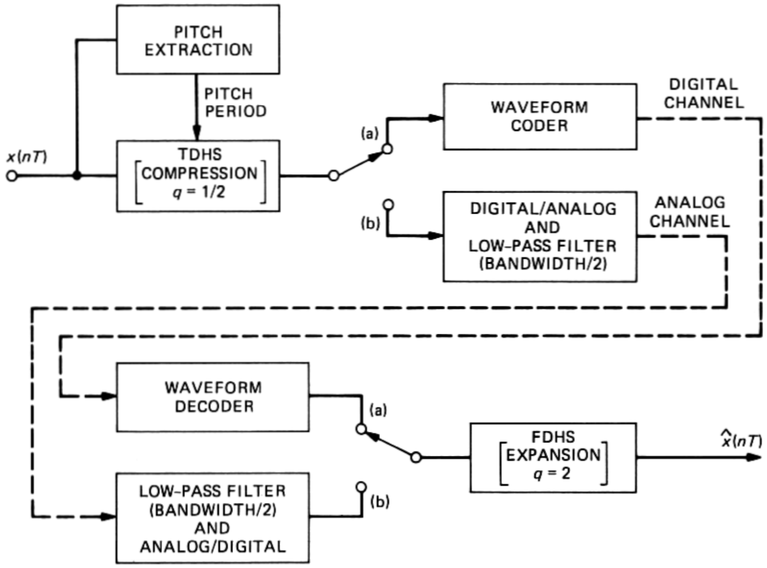

Fig. 21—Block diagram of hybrid frequency scaling system TDHS-FDHS. (a) Digital channel. (b) Analog channel.

The results of this work, with respect to the FDHS technique,* provide a substantial improvement of the earlier version of this technique as reported in Ref. 12. The improvement is manifested in both the quality achieved, and in implementation efficiency. The improvement in quality is the result of using a filter bank with higher frequency resolution and less overlap between filters, as well as the use of the dynamic sign initialization and matching algorithm developed in the present work. The improvement in implementation efficiency is achieved by the use of a block implementation of the short-time Fourier analysis-synthesis system. In this system, the FFT, the embedded decimation and interpolation, and the WOLA synthesis scheme—described in Ref. 14 and extended here to include the case of analysis and synthesis window having longer duration than the transform size—provide a large saving in computation.

It was seen that the introduction of STFT modifications can greatly affect the characteristics of the analysis-synthesis system and its design. The detailed design considerations of the window functions and the selection of the decimation and interpolation factors given here can be useful also in other applications involving spectral modi-

* An early presentation of the results was given in a talk that included an audio tape demonstration.[32]

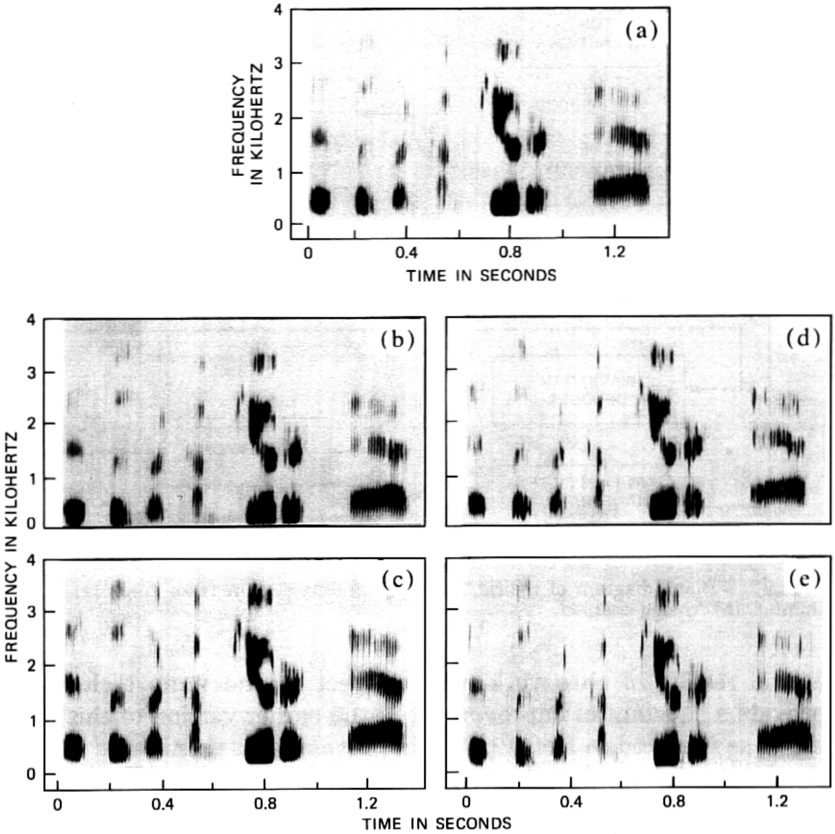

Fig. 22—Spectrograms of original and processed speech by FDHS, TDHS, and hybrid TDHS-FDHS systems ["This is a computer test (of a digital speech coder)," male speaker.] (a) Original. (b) Reconstructed—FDHS. (c) Reconstructed—TDHS. (d) Reconstructed—TDHS with pitch reextraction from compressed signal. (e) Reconstructed—hybrid TDHS-FDHS system.

fications of signals, such as in some of the techniques described in Section II.

The developed FDHS system is particularly amenable to an array-processor implementation. From simulations of the system for a variety of adverse acoustical environment conditions, the FDHS technique appears to be robust and provides reconstructed speech of good communications quality. The main degradation, as mentioned earlier, is introduced in the compression stage, since the expansion operation provides almost transparent quality. Unlike the simpler TDHS technique, the FDHS is not explicitly dependent on pitch extraction and is, hence, more robust. However, for clean speech, compression with TDHS results in better speech quality than with FDHS. On the other hand, for noisy speech signals, in addition to possible failure of the pitch detector at high-noise levels, the TDHS expansion process tends to structure the noise which can be perceptually annoying.

In applications where pitch extraction at the transmitter is feasible but where pitch data transmission is to be avoided, the hybrid TDHS-FDHS system, in which compression is performed by TDHS and expansion by FDHS, provides better overall speech quality than the TDHS or FDHS systems alone. The additional advantages of the hybrid system, such as reduction of noise structuring and higher immunity to channel errors, as compared to TDHS alone, and the lower complexity, as well as higher quality, as compared to FDHS alone, singles out the hybrid system as the best solution for a variety of applications.

An interesting outcome of the general implementation scheme, shown in Fig. 8, is the generalization of the TDHS technique to include both analysis and synthesis windows. This generalization has potential for further improving the TDHS performance.

The FDHS technique was developed on the basis of the quasiharmonic nature of voiced speech. Deviations from this model cause a reduction in processed speech quality. To achieve higher than communications quality with the FDHS technique, further study and ununderstanding are needed of the simultaneous amplitude and phase modulation processes of the individual pitch harmonics, and of the nonstationary characteristics of speech signals.

REFERENCES

1. J. L. Flanagan, *Speech Analysis, Synthesis and Perception.* New York: Springer Verlag, 1972.
2. J. L. Flanagan and R. M. Golden, "Phase Vocoder," B.S.T.J., *45* No. 9 (November 1966), pp. 1493–509.
3. M. R. Schroeder, J. L. Flanagan, and E. A. Lundry, "Bandwidth Compression of Speech by Analytic-Signal Rooting," Proc. IEEE, *55* (March 1967), pp. 396–401.
4. D. Malah, "Time Domain Algorithms for Harmonic Bandwidth Reduction and Time Scaling of Speech Signals," IEEE Trans. Acoust., Speech, Signal Processing, *ASSP-27,* No. 2 (April 1979), pp. 121–33.
5. D. Malah, R. E. Crochiere, and R. V. Cox, "Performance of Transform and Subband Coding Systems Combined with Harmonic Scaling of Speech," IEEE Trans. Acoust., Speech, Signal Processing, *ASSP-29* (April 1981), pp. 273–83.
6. D. Malah, "Combined Time Domain Harmonic Compression and CVSD for 7.2 kb/s Transmission of Speech Signals," Proc. 1980 IEEE Int. Conf. Acoust., Speech, Signal Processing (April 1980), pp. 504–7.
7. J. L. Melsa and A. K. Pande, "Medium-Band Speech Encoding Using Time Domain Harmonic Scaling and Adaptive Residual Coding," Proc. 1981 IEEE Int. Conf. Acoust., Speech, Signal Processing (April 1981), pp. 603–6.
8. J. E. Youngberg, "Rate/Pitch Modification Using the Constant-Q Transform," Proc. 1979 IEEE Int. Conf. Acoust., Speech, Signal Processing (1979), pp. 748–51.
9. H. Ravindra, "Speech Articulation Rate Change Using Recursive Bandwidth Scaling," Proc. 1980 IEEE Int. Conf. Acoust., Speech, Signal Processing (April 1980), pp. 352–5.
10. B. P. Bogert, "The Vobanc – A Two-to-One Speech Bandwidth Reduction System," J. Acoust. Soc. Am. *28,* No. 3 (May 1956), pp. 399–404.
11. J. L. Daguet, "Speech Compression CODIMEX System," IEEE Trans. Audio, *AU-11,* No. 2 (March–April 1963), pp. 63–71.
12. J. L. Flanagan and S. W. Christensen, "Technique for Frequency Division/Multi-

plication of Speech Signals," J. Acoust. Soc. Am. *60,* No. 4 (October 1980), pp. 1061–8.
13. J. M. Tribolet, "A New Phase Unwrapping-Algorithm," IEEE Trans Acoust., Speech, Signal Processing, *ASSP-25,* No. 2 (April 1977), pp. 170–7.
14. R. E. Crochiere, "A Weighted Overlap-Add Method of Short-Time Fourier Analysis/ Synthesis," IEEE Trans. Acoust., Speech, Signal Processing, *ASSP-28,* No. 1 (February 1980), pp. 99–102.
15. J. B. Allen and L. R. Rabiner, "A Unified Approach to Short-Time Fourier Analysis and Synthesis," Proc. IEEE, *65* (November 1977), pp. 1558–64.
16. M. Portnoff, "Time-Frequency Representation of Digital Signals and Systems Based on Short-Time Fourier Analysis," IEEE Trans. Acoust., Speech, Signal Processing, *ASSP-28,* No. 1 (February 1980), pp. 55–69.
17. R. E. Kahn and J. B. Thomas, "Some Bandwidth Properties of Simultaneous Amplitude and Angle Modulation," IEEE Trans. Inf. Theory, *IT-11,* No. 4 (October 1965), pp. 516–20.
18. J. L. Flanagan, "Parametric Coding of Speech Spectra," J. Acoust. Soc. Am., *68,* No. 2 (August 1980), pp. 412–9.
19. J. L. Flanagan and S. W. Christensen, "Computer Studies on Parametric Coding of Speech Spectra," J. Acoust. Soc. Am. *68,* No. 2 (August 1980), pp. 420–30.
20. R. E. Bogner and J. L. Flanagan, "Frequency Multiplication of Speech Signals," IEEE Trans. Audio Electroacoust., *AU-17* (September 1969), pp. 202–8.
21. R. E. Bogner, "Frequency Division in Speech Bandwidth Reduction," IEEE Trans. Comm. Tech., *COM-13,* No. 4 (December 1965), pp. 438–51.
22. R. W. Schafer and L. R. Rabiner, "Design and Simulation of a Speech Analysis-Synthesis System Based on Short-Time Fourier Analysis," IEEE Trans. Audio Electroacoust., *AU-21* (June 1973), pp. 165–74.
23. L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*, Englewood Cliffs, New Jersey: Prentice Hall, 1978, Chapter 6.
24. M. R. Portnoff, Time Scale Modification of Speech Based on Short-Time Fourier Analysis, Ph.D Dissertation, Massachusetts Inst. Tech., Cambridge, April 1978.
25. S. Seneff, "High Quality System for Speech Transformations," (Abstract) 98th Meeting, Acoust. Soc. Am., J. Acoust. Soc. Am., Suppl. 1, *66* (Fall 1979), p. S22.
26. M. R. Portnoff, "Implementation of the Digital Phase Vocoder Using the Fast Fourier Transform," IEEE Trans. Acoust., Speech, Signal Processing, *ASSP-24* (June 1976), pp. 243–8.
27. L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing*, Englewood Cliffs, New Jersey: Prentice Hall, 1975, Chapter 3.
28. J. H. McClellan, J. W. Parks, and L. R. Rabiner, "FIR Linear Phase Filter Design Program," in *Programs for Digital Signal Processing*, New York: IEEE Press, 1979, Chapter 5.
29. J. B. Allen, "Short-Term Spectral Analysis Synthesis, and Modification by Discrete Fourier Transform," IEEE Trans. Acoust., Speech, Signal Processing, *ASSP-25* (June 1977), pp. 235–8.
30. A. M. Noll, "Cepstrum Pitch Determination," J. Acoust. Soc. Am., *41,* No. 2 (February 1967), pp. 293–309.
31. D. H. Klatt, "A Digital Filter Bank for Spectral Matching," Proc. IEEE Int. Conf. Acoust., Speech Signal Processing (April 1976), pp. 573–6.
32. D. Malah and J. L. Flanagan, "Efficient Implementation of a Frequency Domain Technique for Frequency Scaling of Speech Signals," (Abstract) 100th Meeting, Acoust. Soc. Am., November 1980, J. Acoust. Soc. Am., Suppl. 1, *68* (Fall 1980), p. S87.