Time-Domain Algorithms for Harmonic Bandwidth Reduction and Time Scaling of Speech Signals

DAVID MALAH, MEMBER, IEEE

Abstract—Frequency scaling of speech signals by methods based on short-time Fourier analysis (STFA), analytic rooting, and harmonic compression using a bank of filters, is a complex operation which requires a large amount of computation in a digital implementation.

It is shown in this paper that, by incorporating pitch frequency information into a frequency-scaling process based on STFA, it is possible, to a good approximation, to perform this operation in the time domain with very few arithmetic operations (one multiplication and two additions per output sample, in most applications). The derivation of the time-domain harmonic scaling (TDHS) algorithms, selection of parameters, and, in particular, the determination of an appropriate weighting function used in the algorithms, as well as several potential applications, are detailed in the paper. Two proposed applications are discussed in greater detail. These are 1) a vocoder system which incorporates waveform coding of the frequency divided signal (by a factor of up to 3), and 2) a computer-based isolated-word recognition system in which all input utterances are compressed to the same duration at the preprocessing phase effecting an overall computation reduction by a factor of up to 3. Computer simulation results which demonstrate the TDHS algorithms' performance are included.

I. Introduction

A BASIC property of human hearing is that the ear performs a short-time spectral analysis [1]-[3]. Different types of vocoders can be classified, therefore, according to the way spectral fine structure and envelope of the short-time spectrum are represented and coded for transmission. In particular, two broad categories can be distinguished: vocoders which include means for pitch extraction, and vocoders which do not extract pitch explicitly, such as frequency dividing vocoders (FDV's) [1], [3]-[5]. The avoidance of pitch extraction is considered to be an important advantage of FDV's. However, the complexity of the frequency-scaling operations (frequency division and multiplication) combined with the modest scaling factors (2 ÷ 3) have made these vocoders relatively unattractive.

Since real-time pitch extraction by digital techniques is now the state of the art [6], [7], we have attempted to reduce the complexity of the frequency- (and time-) scaling operations by incorporating pitch information into the frequency-scaling process. The frame chosen for introducing pitch information into the frequency-scaling process is harmonic scaling by means of short-time Fourier analysis. This type of analysis provided also the basis for the development of the phase vocoder [8], [9] which is probably the most flexible FDV.

It is shown in this paper that a good approximation of the desired frequency- (or time-) scaling operation can be performed directly in the time domain with simple and efficient algorithms. The time-domain harmonic scaling (TDHS) algorithms consist of properly weighting several adjacent input signal segments (with pitch dependent duration) by a suitable window function, to produce an output segment. In the frequency domain, the time-domain operations are equivalent to shifting the individual pitch harmonics of the quasi-periodic voiced-speech signal according to the center frequency of the subband in which each harmonic component is located. The number of subbands into which the speech band is divided is pitch dependent. The derivation of the TDHS algorithm is facilitated by formulating the short-time Fourier transform in a discrete Fourier transform (DFT) form (as was developed for the phase vocoder [9]) using pitch information for determining the number of analysis bands and performing an approximate harmonic scaling by frequency shifting. Time scaling is easily achieved by properly choosing input and output rates.

The simplicity of the proposed TDHS algorithms makes the proposed approach attractive for several applications. These include time-scale variation of recorded speech signals (with the well-known sampling method [10] being a particular case, of lowest performance, of the more general approach proposed) useful for language teaching and "speed reading" for the blind. Recently [11]-[13] the TDHS algorithms were applied for reducing computations in a computer-based isolated-word speech recognition system by time normalization (compression) of all input utterances to the same duration at the preprocessing phase, prior to the extraction of the parameters representing the input utterance. Frequency-scale variation utilizing the TDHS algorithms has potential for real-time speech bandwidth reduction and a particular system has been proposed recently [14].

The organization of this paper is as follows: frequency scaling by means of short-time Fourier analysis and a proposed approximation which utilizes pitch information is the subject of the next section. In Section III, the TDHS algorithms are derived, and in the following section, the selection of algorithm parameters and the determination of suitable window functions are discussed. In Section V, the application of the proposed algorithms for speech bandwidth reduction is considered, and in Section VI, its application to time-scale variation of speech signals, and, in particular, the proposed application for isolated-word recognition systems (IWRS) is discussed and the results obtained by simulations are reported.

Manuscript received April 10, 1978; revised August 30, 1978.

The author is with the Department of Electrical Engineering, Technion—Israel Institute of Technology, Haifa, Israel.

0096-3518/79/0400-0121$00.75 © 1979 IEEE

II. Frequency and Time Scaling via the Short-Time Fourier Transform

The relation between the short-time Fourier transform and multichannel signal filtering is well established [8], [9], [15]. This relation has been the basis for the development of the phase vocoder [8],]9] and is also the basis for the time-domain harmonic scaling (TDHS) algorithms to be presented. It is, therefore, briefly summarized below.

Since we are interested in a digital implementation, a discrete-time formulation is presented.

Consider a bank of L equally spaced causal bandpass digital filters having unit-sample responses

$$h_k(nT) = 2h(nT) \cos (\omega_k nT), \qquad k = 1, 2, \cdots, L \tag{1}$$

and a low-pass filter $h_o(nT) = h(nT)$ which is the unit-sample response of the basis (prototype) filter. The center frequencies of the passbands are $\omega_k = k\Delta\omega$, where $\Delta\omega$, the filter spacing, is chosen so that the bank of filters covers a frequency range which is contained in $[-\pi/T, \pi/T]$, with T being the sampling interval.

Using discrete convolution, the output signal from the kth filter ($k > 0$) is given by

$$\begin{aligned} y_k &= 2 \sum_{r=-\infty}^{n} x(rT)\, h(nT - rT) \cos [\omega_k(nT - rT)] \\ &= 2\text{Re}\{\exp [j\omega_k nT] X(\omega_k, nT)\}, \qquad k = 1, 2, \cdots, L \end{aligned} \tag{2}$$

where

$$X(\omega_k, nT) \triangleq \sum_{r=-\infty}^{n} x(rT)\, h(nT - rT) \exp [-j\omega_k rT]. \tag{3}$$

$X(\omega_k, nT)$ is called the discrete short-time Fourier transform [9], [15] of the input signal $x(nT)$. At time instant nT, it is the Fourier transform of the past input samples weighted by the data window $h(nT - rT)$, $-\infty < r \leqslant n$, and evaluated at frequency ω_k.

If $X(\omega_k, nT)$ is expressed in terms of its magnitude $|X(\omega_k, nT)|$ and phase $\phi(\omega_k, nT)$, (2) can be written as

$$y_k(nT) = 2|X(\omega_k, nT)| \cos [\omega_k nT + \phi(\omega_k, nT)]. \tag{4}$$

Frequency scaling the input signal by some factor q ($q < 1$ for frequency division and $q > 1$ for frequency multiplication), without changing its time duration, can be achieved from (4) by scaling the instantaneous frequency, i.e., multiplying by q each carrier frequency ω_k and phase derivative $\dot{\phi}(\omega_k, nT)$ (approximated by $\Delta\phi/T = [\phi(\omega_k, nT) - \phi(\omega_k, (n - 1)T)]/T$ in a digital implementation [3], [8]) and summing up all the frequency-scaled channel signals.[1]

This is the approach taken in the implementation of the phase vocoder when operated as an FDV in a "self-multiplexing" mode [3], [8]. In this mode of operation, the input speech signal is frequency divided at the transmitter and frequency multiplied at the receiver with the same factor.

A time-scaled signal which occupies the original frequency band can be obtained by replaying the frequency-scaled (by a factor q) signal at $1/q$ speed.

As discussed in the previous section, the TDHS algorithms are based on the assumption that the fundamental frequency F_o (the pitch) of the input voiced-speech signal is approximately known. To show how the pitch information is introduced into the frequency-scaling process discussed above, we assume, for the present discussion, that the input signal is a wide-band *periodic* signal with a fundamental frequency F_o. The fact that speech signals are quasi-stationary and contain voiced (quasi-periodic) and unvoiced (noiselike) portions will be considered at a later stage. Let F_p be the estimated fundamental frequency (as obtained, in the case of voiced speech, from the pitch extractor), and let the spacing between adjacent center frequencies Δf ($\Delta\omega = 2\pi\Delta f$) be chosen as equal to F_p. Let, also, each subband be of width $\Delta\omega$ and the number of subbands L (for $k > 0$) be greater than or equal to the number of harmonics present in the band-limited periodic input signal. With the above choice of $\Delta\omega$ and L, each spectral line of the signal harmonics will be located in a separate subband, provided that the fundamental frequency estimate error $\Delta F_p \triangleq |F_p - F_o|$ is less than $\Delta f/2L$, i.e., if

$$\epsilon_p \triangleq \Delta F_p/F_p < 1/2L. \tag{5}$$

Following a short-time Fourier analysis, and assuming that $h(nT)$ is a good approximation to an ideal low-pass filter, the phase derivative $\dot{\phi}(\omega_k, nT)$ essentially describes the frequency deviation of the kth harmonic from the center frequency ω_k of the subband in which the spectral line of this harmonic is located.

To demonstrate this fact, we consider a particular harmonic at frequency lF_o, $1 \leqslant l \leqslant L$. If (5) is satisfied, then the spectral line corresponding to this harmonic is located at the lth subband with center frequency $\omega_l = l2\pi F_p$. This harmonic is given in the time domain by

$$x_l(nT) = A_l \cos (\Omega_l nT + \phi_l) \tag{6}$$

where A_l and ϕ_l are the amplitude and phase of the lth harmonic and $\Omega_l = 2\pi l F_o$, F_o being the fundamental frequency of the periodic input signal.[2] Substituting (6) into (3) and assuming for simplicity that the Fourier transform of $h(nT)$ is $\exp(-j\omega\tau)$ for $|\omega| < \Delta\omega/2$ and zero for $|\omega| > \Delta\omega/2$ (i.e., an ideal low-pass filter with a time delay of τ), one obtains

$$X(\omega_l, nT) = (A_l/2) \exp [j(\Delta\Omega_l(nT - \tau) + \phi_l)] \tag{7}$$

where

$$\Delta\Omega_l = \Omega_l - \omega_l, \text{ i.e.,} \qquad |\Delta\Omega_l| = 2\pi l \Delta F_p.$$

From (7) we find that $\phi(\omega_l, nT) = \Delta\Omega_l(nT - \tau) + \phi_l$ and the

[1] It is assumed here that if $q > 1$, T is at least q times smaller than the Nyquist interval, so that the frequency multiplication does not cause aliasing.

[2] The wide-band harmonic input signal is given, therefore, in this example by $x(nT) = \sum_{l=1}^{L} x_l(nT)$.

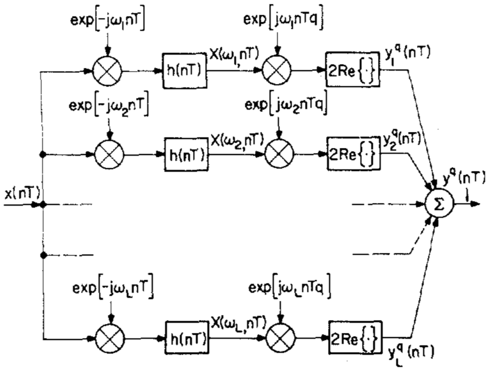

Fig. 1. A complex scheme for approximate frequency scaling of speech signals by a factor q via the short-time Fourier transform. (Prototype filter bandwidth and the number of channels depend on pitch information.) The channel centered at zero frequency is omitted.

phase "derivative" is given by the frequency deviation $\Delta\Omega_l$ from the center of the lth subband.

We conclude, therefore, that if this deviation is sufficiently small, a good approximation to exact frequency scaling is obtained by just scaling the center frequencies ω_k and neglecting the effect of the phase derivative.

The approximate frequency scaled signal (by a factor q) at the kth channel $y_k^q(nT)$ is given from (2) by

$$y_k^q(nT) = 2 \text{ Re } \{\exp(jq\omega_k nT)\, X(\omega_k, nT)\}, \quad k = 1, 2, \cdots, L. \tag{8}$$

Viewing (3) as a discrete convolution between $x(nT) \cdot \exp(-j\omega_k nT)$ and $h(nT)$, the scheme shown in Fig. 1 describes the operations required for approximate frequency scaling as of the present discussion. It will be shown in the next section that this rather complicated scheme can be further reduced to simple time-domain algorithms (we differentiate later between the two cases of $q < 1$ and $q > 1$) which require only few arithmetic operations per output sample. Before we turn to the derivation of these algorithms, we pursue further the above example and substitute (7) in (8) obtaining

$$y_l^q(nT) = A_l \text{ Re } \{\exp[j(\Omega_l' nT + \phi_l')]\} \tag{9}$$

where

$$\Omega_l' = q\omega_l + \Delta\Omega_l = q\Omega_l + \Delta\Omega_l(1 - q) \tag{10}$$

and

$$\phi_l' = \phi_l - \Delta\Omega_l \tau = \phi_l - 2\pi l(F_o - F_p)\, \tau. \tag{11}$$

Hence, the relative error in frequency scaling is given by

$$\epsilon_f \triangleq |\Omega_l' - q\Omega_l|/\Omega_l = |\Delta\Omega_l(1 - q)|/\Omega_l \simeq \epsilon_p |1 - q|. \tag{12}$$

The approximation at the right-hand side of (12) is under the assumption that $F_o \gg \Delta F_p$ so that $|\Delta\Omega_l|/\Omega_l \simeq \epsilon_p$.

In an FDV, the input signal is first frequency divided by a factor $C > 1$ ($q = 1/C$) and hence, from (10)

$$\Omega_l' = \omega_l/C + \Delta\Omega_l = \Omega_l/C + \Delta\Omega_l(C - 1)/C. \tag{13}$$

If the frequency-divided signal is replayed at C-times speed, the lth harmonic is restored to the original lth subband only if $C|\Delta\Omega_l| < \Delta\omega/2$, i.e., if

$$\epsilon_p = \Delta F_p / F_p < 1/(2Cl). \tag{14}$$

If the replayed signal (which is now compressed in time) is frequency multiplied using the same approach (but with $q = S > 1$), the frequency of the lth harmonic becomes

$$\Omega_l'' = S\omega_l + C\Delta\Omega_l = S\Omega_l + (C - S)\, \Delta\Omega_l. \tag{15}$$

If we let $S = C$, we see from (15) that $\Omega_l'' = S\Omega_l$ and following the replay of the time-compressed signal at $1/S$ speed the harmonics of the resulting signal are back to their original location at multiples of F_o. We conclude, therefore, that if the condition in (14) holds for all the harmonics, exact reconstruction is obtained (except for a time delay and provided that $h(nT)$ is an ideal low-pass filter as assumed above). In other words, if M signal harmonics are to be exactly reconstructed following frequency division and multiplication, with the same factors, by the proposed approximate frequency scaling method, the relative error in determining the fundamental frequency ϵ_p should satisfy

$$\epsilon_p \leqslant 1/(2CM). \tag{16}$$

We turn now to the derivation of the TDHS algorithms which greatly reduces the number of arithmetic operations in comparison to the scheme shown in Fig. 1.

III. Derivation of the TDHS Algorithms

Let q be the desired frequency-scaling factor (usually in the range[3] $\frac{1}{3} \leqslant q \leqslant 3$, $q \neq 1$) and assume that q is rational and is expressed as

$$q = \mu/\delta, \tag{17}$$

where μ and δ are relatively prime integers. For the purpose of the derivation, we assume initially that the sampling interval of the input signal is

$$T' = T/\mu, \tag{18}$$

where T is less or equal to the Nyquist interval for the input band-limited speech signal.

According to the approximate frequency-scaling operation discussed in the previous section, the output of the kth channel is given by

$$y_k^q(nT') = \begin{cases} 2 \text{ Re } \{\exp(jq\omega_k nT')\, X(\omega_k, nT')\}, & \\ & k = 1, 2, \cdots, L \\ X(0, nT') & k = 0. \end{cases} \tag{19}$$

In (19) we have repeated (8) but substituted T' for T and added the channel for $k = 0$. Utilizing the fact that all channel filters are frequency shifted versions of the prototype low-pass

[3] Due to speech nonstationarity and the quasi-periodic nature of voiced speech, the assumption of a discrete line spectrum is not exact. If frequency division by a factor of more than 3 is attempted, the "widened" spectral lines will usually overlap. This will degrade the reconstructed signal. In the case of time scaling, the perception limitation of speeded-up or slowed-down speech limits the scaling factors range to even a smaller one than stated.

filter $h(nT')$, the expression for $X(\omega_k, nT)$ given in (3) can be put into a DFT form [7]. In the particular case under consideration, let the frequency range $[-\pi/T', \pi/T']$ be divided into N equally spaced subbands with a spacing $\Delta\omega = 2\pi/(NT')$, so that the center frequencies ω_k, $k = 0, \pm 1, \cdots, \pm(N-1)/2$ (N is assumed to be odd[4]), are given by $\omega_k = 2\pi k/(NT')$. Rewriting (3) (with T' replacing T) as a sum of finite sums over N samples of the weighted signal and utilizing the periodicity in r of $\exp(-j\omega_k rT') = \exp(-j2\pi kr/N)$, with period N, we obtain [9]

$$X(\omega_k, nT') = W_N^{-kn} G(k, nT') \tag{20}$$

where

$$W_N \triangleq \exp(j2\pi/N) \tag{21}$$

and

$$G(k, nT') \triangleq \sum_{r=0}^{N-1} g(rT', nT')\, W_N^{kr} \tag{22}$$

with

$$g(rT', nT') = \sum_{i=0}^{\infty} x(nT' - rT' - iNT')\, h(rT' + iNT')$$
$$r = 0, 1, \cdots, N-1. \tag{23}$$

$G(k, nT')$ in (22) is observed to be the DFT of the N point sequence $g(rT', nT')$. Note that due to the periodicity of W_N^{kr} in k, with period N, one can use $G(k, nT') = G(N + k, nT')$ in the computation of $X(\omega_k, nT')$ in (20) for negative values of k. We further assume, as in [9], that the prototype low-pass filter has a finite duration unit-sample response (FIR), i.e., $h(nT') = 0$ for $n < 0$ and $n \geqslant M$. For convenience (and without loss of generality) we let $M = mN$, m a positive integer. In that case, $g(rT', nT')$ takes the form

$$g(rT', nT') = \sum_{i=0}^{m-1} x(nT' - rT' - iNT')\, h(rT' + iNT'),$$
$$r = 0, 1, \cdots, N-1. \tag{24}$$

Letting the number of channels L $(k > 0)$ be equal to $(N-1)/2$ and substituting (20) and (21) in (19) we get

$$y_k^q(nT') = \begin{cases} 2\,\mathrm{Re}\,\{W_N^{nk(q-1)} G(k, nT')\}, & \\ & k = 1, 2, \cdots, (N-1)/2 \\ G(0, nT') & k = 0. \end{cases} \tag{25}$$

Hence, by summing the outputs of the $L + 1$ channels (including $k = 0$), we obtain the signal $y^q(nT')$ given by

$$y^q(nT') = G(0, nT') + 2\,\mathrm{Re}\left\{\sum_{k=1}^{(N-1)/2} W_N^{nk(q-1)} G(k, nT')\right\} \tag{26}$$

which is the approximate frequency-scaled signal. We now show that (26) can be further reduced and simplified.

Since $g(rT', nT')$, as given in (23) or (24), is a real sequence, the complex conjugate of $G(k, nT')$ satisfies $G^*(k, nT') = G(N-k, nT')$ and, therefore, (26) can be put in the following form.

$$y^q(nT') = \sum_{k=0}^{N-1} G(k, nT')\, W_N^{nk(q-1)} + A_q(nT'), \tag{27}$$

where

$$A_q(nT') = [1 - W_N^{nNq}] \sum_{k=1}^{(N-1)/2} G^*(k, nT')\, W_N^{-nk(q-1)}. \tag{28}$$

Clearly, if q is an integer, $W^{nNq} = 1$, yielding $A_q(nT') = 0$. For other values of q, the term $A_q(nT')$ in (27) can be eliminated by sampling $y^q(nT')$ at time interval $\delta T'$, where δ is an integer and is given from (17). This is equivalent to replacing n in (27) and (28) by $n\delta$, so that W_N^{nNq} becomes $W_N^{nN\mu} = 1$ and, hence, $A_q(n\delta T') = 0$. This decimation of $y^q(nT')$ does not cause aliasing since the initial sampling interval T' is at least μ times smaller than the Nyquist interval [see (18)] and the scaling factor satisfies $q = \mu/\delta \leqslant \mu$.

We have, therefore,

$$y^q(n\delta T') = y^q(nT/q) = \sum_{k=0}^{N-1} G(k, nT/q)\, W_N^{nk(\mu-\delta)}. \tag{29}$$

We consider now, in turn, each of the two cases of frequency division (or time-scale compression) and frequency multiplication (or time-scale expansion).

A. Frequency Division

Let C denote the rational frequency division factor, i.e., $q = \mu/\delta = 1/C$ $(C > 1$ so that $\delta > \mu)$, and let p_c be an integer which is defined by

$$p_c \triangleq n(\delta - \mu) \bmod N, \qquad 0 \leqslant p_c \leqslant N-1. \tag{30}$$

Making use of the periodicity in l of W_N^l with period N, one may write (31) in place of (29):

$$y^{1/c}(nCT) = \sum_{k=0}^{N-1} G(k, nCT)\, W_N^{-kp_c}, \tag{31}$$

where $q = 1/C$ and p_c of (30) have also been used.

From the definition of $G(k, nT')$ given in (22), we observe that the right-hand side of (31) is the inverse DFT (scaled by N) of $G(k, nCT)$ evaluated at $r = p_c$. Hence,

$$y^{1/c}(nCT) = Ng(p_cT', nCT) = Ng(p_cT/\mu, nCT), \tag{32}$$

where $g(\cdot, \cdot)$ is given from (24).

Let T_p be the duration of the estimated pitch period, and assume $T_p = N_pT$, N_p an integer. According to the discussion in Section II, the speech band $[-\pi/T, \pi/T]$ is subdivided into N_p subbands so that the center frequencies ω_k are at multiples of the estimated pitch harmonics. Since by (18) $T' = T/\mu$ and the band $[-\pi/T', \pi/T']$ has been assumed to be subdivided into N subbands, we have that

[4] The derivation with N even is less convenient as it requires omitting the filter centered at zero frequency and taking in account that the center frequencies are not direct multiples of the first filter center frequency. It is possible to modify the derivation and obtain the same results as with N odd. For convenience we proceed with N odd.

$$N = \mu N_p. \tag{33}$$

Rewriting (30) as

$$p_c = n(\delta - \mu) - \alpha_c \mu N_p, \tag{34}$$

where α_c is the integer part of $n(\delta - \mu)/N$, i.e.,

$$\alpha_c \triangleq \text{int}\ [n(\delta - \mu)/N_p \mu] = \text{int}\ [n/N_c], \tag{35}$$

where[5]

$$N_c \triangleq \mu N_p/(\delta - \mu) = N_p/(C - 1). \tag{36}$$

Using (34) and (36) for evaluating $p_c T'$ which appears in (32), we find

$$p_c T' = p_c T/\mu = nT(C - 1) - \alpha_c N_p T = nT_c - \alpha_c N_c T_c, \tag{37}$$

where

$$T_c \triangleq T(C - 1) = TN_p/N_c, \tag{38}$$

and, hence, from (37) and (35)

$$p_c T' = (n \bmod N_c)\, T_c, \tag{39}$$

substituting in (32) and using (24), with $T' = T/\mu$ and $C = \delta/\mu$, we finally obtain[6]

$$y^{1/c}(nCT) = \sum_{i=0}^{m-1} x(nT + \alpha_c N_p T - iN_p T)\, h_N(iN_c T_c + (n \bmod N_c)\, T_c) \tag{40}$$

where

$$h_N(\cdot) \triangleq Nh(\cdot). \tag{41}$$

It is important to note that the final expression (40) for the frequency-divided signal is given in terms of the input samples $x(nT)$, i.e., T is the input sampling interval and no oversampling with T' of (18) is necessary in the evaluation of (40). The output signal is given in (40) with a sampling interval CT. This does not cause aliasing since the frequency-divided signal has a C-times reduced bandwidth. If the sequence $y^{1/c}(nCT)$ is output at the rate $1/(CT)$, the (approximate) frequency-divided signal which results is of the same duration as the input signal. If, however, this sequence is stored and then output at the rate $1/T$ (i.e., a C-times speedup), a time-compressed signal is obtained which occupies the original frequency band.

The arithmetic operations required in (40) are m multiplications and $(m - 1)$ additions per output sample. Since, as discussed in the following section, m is typically chosen in the range of 2 to 6, the computation load is quite small [typically, less than 1 percent of the operations required in the scheme of Fig. 1, even if the FFT algorithm is used for computing $X(\omega_k, nT)$]. Furthermore, in most applications to be considered later on, $m = 2$ is used and, due to the particular properties of the data window function $h_N(\cdot)$ (to be discussed), only one multiplication and two additions per output sample are required. It is assumed in this discussion that $h_N(nT_c)$ is obtained by sampling an analog prototype low-pass filter impulse response $\hat{h}(t)$.[7] In a digital implementation, these samples can be stored in a ROM, or computed on line if $\hat{h}(t)$ is given analytically. A more detailed discussion of implementation considerations is deferred to a later section.

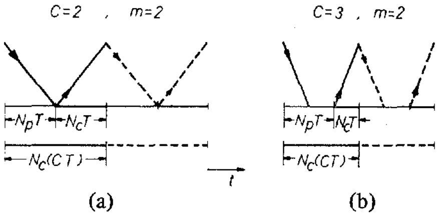

Fig. 2. Organization of the time-domain algorithm for frequency division on a segment by segment basis. The use of $m = 2$ and a triangular window is shown for scaling factors of (a) $C = 2$; (b) $C = 3$.

Let us examine now the way the input signal is processed. According to (40), an output sample is obtained from the weighted sum of m input samples taken at $N_p T$ (the estimated pitch period) time intervals apart. The role of α_c in the expression is to advance the input sequence index by N_p following the computation of N_c consecutive samples. This way, for every $N_c + N_p$ input samples, N_c output samples are computed. Since $(N_c + N_p)T = N_c CT$, the processing can be done continuously on line, requiring only a relatively small buffer memory. It is best organized on a segment by segment basis, as demonstrated in Fig. 2, for $m = 2$ and a triangular weighting function. It is observed from the figure that for this value of m, if $C > 2$ (and in general if $C > m$), not all input data are weighted and part of the data is discarded. Due to the redundancy present in the speech signal (except for fast transitions), this has a minor effect up to $C = 3$. The proper selection of m and the weighting function $\hat{h}(t)$ is explained in Section IV.

B. Frequency Multiplication

In this case, we let S denote the rational frequency multiplication factor, i.e., $q = \mu/\delta = S$ $(S > 1,\ \mu > \delta)$. The derivation for this case follows closely the derivation for frequency division, except for a slight modification stemming from the reversal of the sign of $(\mu - \delta)$ and its effect on the right-hand side of (29). To accommodate for this change of sign, we define the integer p_s in (42) in place of p_c in (30):

$$p_s \triangleq n(\mu - \delta) \bmod N, \qquad 0 \leqslant p_s \leqslant N - 1. \tag{42}$$

Substitution in (29) yields the counterpart expression to (32)

[5] It is assumed here that $N = \mu N_p$ is divisible by $(\delta - \mu)$ so that N_c, as defined in (36), is an integer. In the general case, this assumption does not hold and N_c is rounded to the closest integer $\tilde{N}_c$. This means that a slightly different scaling factor $C' = (N_p + \tilde{N}_c)/\tilde{N}_c$ is actually being realized. A similar situation occurs for frequency multiplication, discussed in the sequel. In this case, N_s, as defined in (45), is rounded to $\tilde{N}_s$ (if μN_p is not divisible by $(\mu - \delta)$), and $S' = \tilde{N}_s/(\tilde{N}_s - N_p)$ is the factor actually being realized.

[6] In this derivation N_p was assumed to be a fixed parameter. The variation of N_p with time (due to pitch variation) may cause a problem of output signal discontinuity. This problem is considered in the following section. An earlier less concise derivation is presented in [27].

[7] Aliasing due to sampling $\hat{h}(t)$ is negligible since the sampling interval T_c [and for frequency multiplication T_s of (46)] is sufficiently small relative to $N_p T = 2\pi/\Delta\omega$, where $\Delta\omega$ is the filter bandwidth.

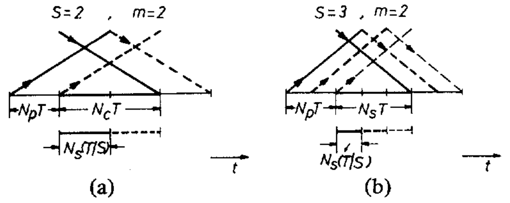

Fig. 3. Organization of the time-domain algorithm for frequency multiplication on a segment by segment basis. The use of $m = 2$ and a triangular window is shown for scaling factors of (a) $S = 2$; (b) $S = 3$.

$$y^s(nT/S) = Ng((N - p_s)T/\mu, nT/S). \tag{43}$$

Following further the previous steps and using α_s, N_s, and T_s in place of α_c, N_c, and T_c, respectively, where

$$\alpha_s \triangleq \text{int } [n/N_s], \tag{44}$$

$$N_s \triangleq N_p\mu/(\mu - \delta) = N_pS/(S - 1), \tag{45}$$

$$T_s \triangleq TN_p/N_s = T(S - 1)/S, \tag{46}$$

we finally arrive at the following expression for computing the samples of the (approximate) frequency-multiplied signal

$$y^s(nT/S) = \sum_{i=1}^{m} x(nT - \alpha_s N_p T - iN_p T) \cdot h_N(iN_sT_s - (n \bmod N_s)T_s). \tag{47}$$

If the sequence $y^s(nT/S)$ is output at the rate S/T, the resulting output signal is of the same duration as the input signal but occupies an S-times bandwidth. If the sequence is stored and then output at the rate $1/T$ (corresponding to a replay at $1/S$ speed), a signal which is stretched in time but occupies the original frequency band is obtained.

The organization of the algorithm on a segment by segment basis is demonstrated in Fig. 3 for $m = 2$ and a triangular window function. In this case, an output segment of N_s samples is produced from N_s consecutive input samples [weighted each with the corresponding past $(m - 1)$ samples]. The input index is then decremented by N_p (due to the increase of α_s by one for every N_s samples) so that the net effect is that, for every $N_s - N_p$ input samples, N_s output samples are produced. Since $(N_s - N_p)T = N_sT/S$, the processing can be done continuously on line.

The previous remarks with regard to the amount of computations apply also to this case.

IV. Determination of Algorithms' Parameters and Window Function

Given a desired scaling factor q, the algorithms expressed by (40) and (47) require the proper determination of the parameters N_p, m, and the window function $h_N(\cdot)$. The sampling interval T is chosen to comply with the input signal bandwidth.

As discussed earlier, N_p should be made equal to the number of samples in the estimated pitch period. If pitch tracking is performed, N_p varies with pitch period variation. This puts a requirement that adjacent segments processed with different values of N_p should maintain output signal continuity at the interface between segments. For $m = 2$ this is easily assured, as can be observed in Figs. 2 and 3 by assuming that the dashed window lines have a different slope than the solid lines corresponding to a different value of N_p (and hence N_c or N_s) than in the previous segment. If higher values of m are used, this property is not assured if N_p is varied from one segment to the other. In some applications, such as time normalization of speech utterances input to an isolated word recognition system, no signal reconstruction is required and hence a fixed N_p (e.g., corresponding to the average value for the particular speaker) can be used so that mainly spectral envelope information is retained. A bandwidth reduction or a time-scaling system can also be operated with fixed N_p but the output signal might be quite degraded. In such cases, however, a higher value of m can be used to obtain a better approximation of the prototype filter to the desired ideal low-pass filter assumed in the earlier discussion. The value of m is actually limited by two factors. One is the linear increase in the number of computations with m. The other stems from the quasi-stationary nature of the speech signal which, for a typical quasi-stationarity interval of 20–40 ms, limits m to 4, or at most 6, in order to avoid weighting of data outside this interval.

To determine the proper window function to be used for different values of m, we state first the requirements and constraints that should be satisfied by this function. It is convenient to discuss this subject with respect to an analog filter having an impulse response $\hat{h}(t)$ from which $h_N(\cdot)$ is obtained by sampling at the proper time instants.

A basic requirement from the underlying bank of filters is that it has a linear phase and uniform amplitude response in the speech band when all its outputs are summed up. Phase linearity is easily achieved by requiring $\hat{h}(t)$, $t\epsilon[0, mN_pT]$ to be symmetrical, i.e.,

$$\hat{h}(t) = \hat{h}(mN_pT - t), \qquad 0 \leqslant t \leqslant mN_pT/2. \tag{48}$$

If we set $q = 1$ (i.e., $\mu = \delta = 1$) in (29), we obtain

$$y^1(nT) = \sum_{k=0}^{N-1} G(k, nT) = Ng(0, nT), \tag{49}$$

and from (24), and since here $N = N_p$, (also $T' = T$)

$$y^1(nT) = N_p \sum_{i=0}^{m-1} x(nT - iN_pT)\, h(iN_pT). \tag{50}$$

Hence, $y^1(nT)$ will become a delayed version of the input signal, if $h(iN_pT) = 0$, for all i except one. To accommodate with (48) this requirement becomes

$$h((l - m/2)N_pT) = \begin{cases} 1 & \text{for } l = m \\ 0 & \text{for } l \neq m \end{cases}, \tag{51}$$

where l takes integer values.

A similar result is obtained in [9], [15] by considering the overall unit-sample response of the bank of filters with summed outputs.

Another important requirement is that, if the input signal is periodic, with period N_oT, and N_p is set equal to N_o, exact frequency scaling should result. In the frequency domain this can be expressed by the requirement that $\hat{H}(l\Omega_p) = 0$, $l \neq 0$ (l any integer and $\Omega_p = 2\pi/(N_pT)$).

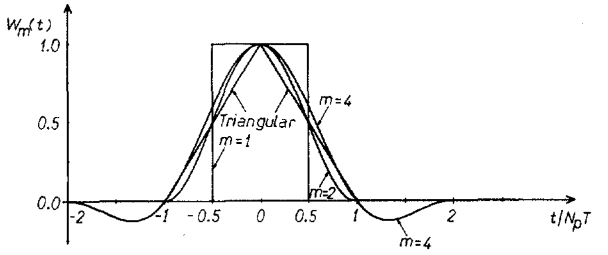

Fig. 4. The window functions $w_m(t)$ for $m = 1, 2, 4$ and the triangular window function $w_T(t)$.

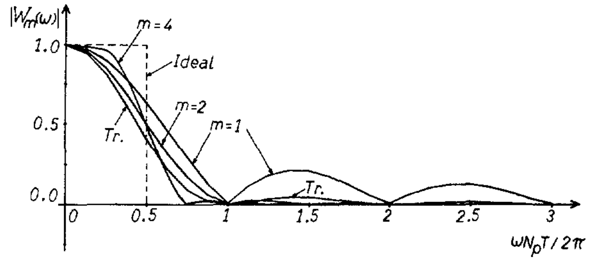

Fig. 5. Frequency response of the window functions shown in Fig. 4 and the desired ideal response (dashed line).

To find the equivalent condition in the time domain, the periodicity of the input, i.e., $x(nT) = x(nT - iN_pT)$, is used in (40) [or (47)] which results in the condition

$$\sum_{i=0}^{m-1} \hat{h}(t + iN_pT) = 1, \qquad 0 \leqslant t \leqslant N_pT. \tag{52}$$

This result can also be obtained from the frequency-domain condition stated above by using Poisson's sum formula [16]. It is noted that if (52) is satisfied, (51) is not contradicted.

A particular family of window functions $w_m(t)$, which satisfies all the above requirements by setting $\hat{h}(t) = w_m(t - mN_pT/2)$, is derived in Appendix I and is given by

$$w_m(t') = \begin{cases} \left.\begin{array}{ll} (\sin \pi t')/(m \sin \pi t'/m), & m \text{ odd} \\ (\sin \pi t')(\cot \pi t'/m)/m, & m \text{ even} \end{array}\right\} & |t'| < m/2 \\ 0 & |t'| > m/2 \end{cases} \tag{53}$$

where t' is a normalized time variable given by $t' \triangleq t/(N_pT)$. For $t' = m/2$, $w_m(t') = 0$ if m is even, and is set to half of the value of the right-hand side of (53) if m is odd (due to the discontinuity at window ends). For $m = 1$, the rectangular window is obtained (with end points set to 0.5), whereas for $m = 2$, the well-known Hanning window results.

Fig. 4 shows the window functions $w_m(t)$, $m = 1$, 2, and 4, and Fig. 5 the corresponding frequency responses. Due to the discontinuity of the window function at the end points (when m is odd), these windows are not very useful for odd values of m. The rectangular window ($m = 1$) is shown since the "sampling method" [10] which is used for time scaling of speech signals actually uses a rectangular window (usually with some tapering at the edges to avoid the discontinuity). The "sampling method" is very simple since it avoids pitch extraction and uses a rectangular window function. However, in view of the rectangular window frequency response shown in Fig. 5 and the lack of pitch tracking, the resulting signal is expected to be quite degraded. The performance can be improved, even without pitch tracking, by the use of better window functions corresponding to higher values of m (taken to be even), since they provide a better approximation to the desired ideal lowpass filter. Since increasing m increases the amount of computations, we have found in simulations that a good compromise is to use $m = 2$. In particular, for $m = 2$, the triangular window function, also shown in Fig. 4, satisfies all the required conditions and is simpler to implement than the Hanning window. This is especially important if pitch tracking is performed, so that N_p varies from one segment to the other, or if different values of q are to be used (e.g., compressing different utterances to the same duration). In such cases, the triangular window eliminates the need for storing, or interpolating, different values of the window function, corresponding to different sampling instants of $\hat{h}(t)$, according to different values of N_p or q. The triangular window $w_T(t)$ makes possible simple on-line computation of its samples and is particularly suitable for real-time applications. Its frequency response is also shown in Fig. 5. Its performance has been found in simulations to be only slightly less (on the average) than the Hanning window.

The use of $m = 2$ also has the advantage of changing the number of computations from two multiplications and one addition to one multiplication and two additions per output sample. This is based on the property (52) which, when used (with $m = 2$) in (40), enables one to write, for a typical computation cycle,

$$\begin{aligned} y^{1/c}(l) &= x(l)h_N(l) + x(l - N_p)h_N(l + N_c) \\ &= x(l - N_p) + h_N(l)[x(l) - x(l - N_p)], \\ &\qquad l = 0, 1, \cdots, N_c - 1 \end{aligned} \tag{54}$$

or, if used in (47),

$$\begin{aligned} y^s(l) &= x(l - N_p)h_N(N_s - l) + x(l - 2N_p)h_N(2N_s - l) \\ &= x(l - 2N_p) + h_N(N_s - l)[x(l - N_p) - x(l - 2N_p)], \\ &\qquad l = 0, 1, \cdots, N_s - 1. \end{aligned} \tag{55}$$

V. Applications of the TDHS Algorithms to Speech Bandwidth Reduction

From previously reported results on the performance of FDV'S [3]-[5], we expect that the proposed algorithms can be effectively utilized for achieving bandwidth reduction factors in the range of 2 to 3. Higher factors can be achieved by applying additional data compression methods to the frequency-divided signal. In this section, we first present and discuss computer simulation results by which the performance of the proposed algorithms were examined. We next consider a proposal for a vocoder system which applies waveform coding to the frequency-divided signal.

The computer simulations were performed on a Nova-2 minicomputer system at the signal processing laboratory of the

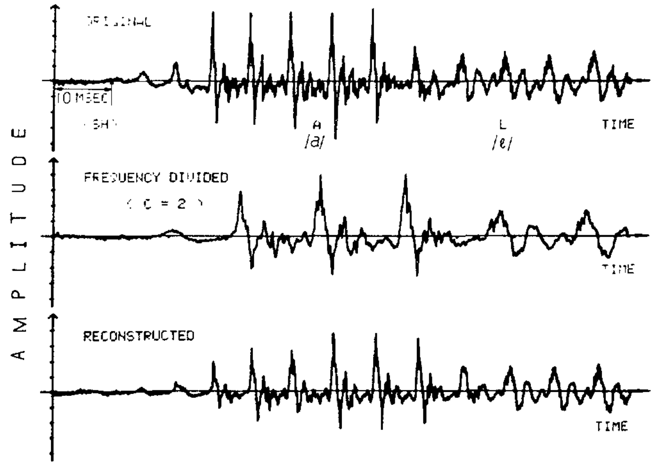

Fig. 6. Original, frequency divided by a factor $C = 2$, and reconstructed signals as obtained for $m = 2$ and a triangular window function.

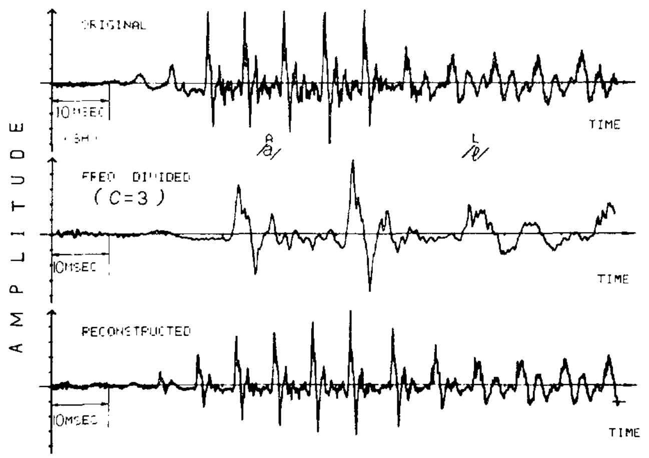

Fig. 7. Original, frequency divided by a factor $C = 3$, and reconstructed signals as obtained for $m = 2$ and a triangular window function.

Technion. The data base for the simulation was prepared from several recorded sentences spoken by different speakers in two languages, Hebrew and English. The input speech signal was bandlimited to the frequency range of 200-3200 Hz and sampled at a 10 kHz rate with a 12 bit A/D converter. Pitch extraction was done by the computer using a modified (simplified) version of the algorithm presented in [6]. Informal listening tests have shown that, with a bandwidth reduction factor of 2, very little, if any, degradation is present in the reconstructed speech signal. A slight degradation in some speech segments was discerned when a factor 3 was used, mainly due to the more stringent requirement on pitch tracking accuracy [see (16)] or extremely fast transitions. Most simulations were performed with $m = 2$ and a triangular window function, since the performance with this simple window was only slightly lower (on the average) than with the Hanning window. As expected from the discussion in the previous section, the use of higher values of m with pitch tracking actually reduced performance, whereas without pitch tracking the resulting signal sounds metallic and is usually of unacceptable quality. The results so far do not appear to be noticeably dependent on speaker or language (for the above two languages and few sample sentences, and bandwidth reduction factors of up to 3). When higher scaling factors than 3 were attempted, a higher degree of degradation was obtained for fast speakers.

Typical waveforms obtained in the simulations are shown in Figs. 6-8. The speech segment shown in these figures is part of the Hebrew word "shalom" (meaning peace). The short segment shown (of 100 ms duration) is a relatively difficult one since it includes a transition from unvoiced (/ʃ/) to voiced (the vowel /a/), and in a relatively short time (30 ms) another transition to a voiced semivowel (/ℓ/). The transitions in the reconstructed signal are longer (especially with the triangular window function), but this did not have a noticeable effect in listening. The first two figures (6 and 7) show the results obtained for factors of 2 and 3, respectively. In Fig. 8 a comparison is made between the triangular and Hanning windows. Computer-generated narrow-band spectrograms of a 1 s duration sentence (original and reconstructed), which includes the segment examined in the last three figures, are shown in Fig. 9.

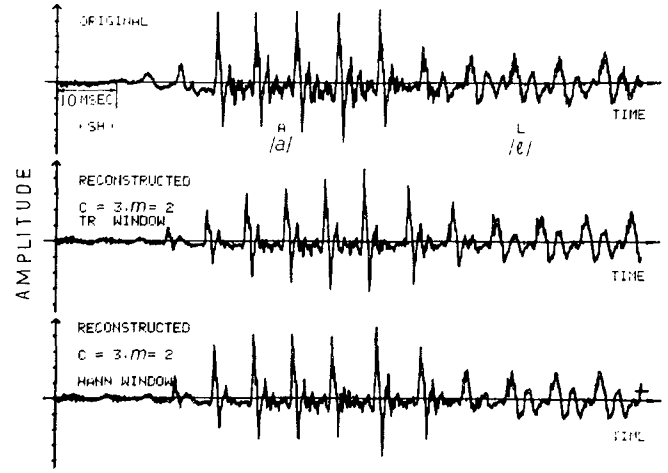

Fig. 8. Original and reconstructed signals as obtained for a scaling factor of $C = 3$ with triangular window and a Hanning window $[w_2(t)]$.

It should be noted that here the pitch detector is required only to track pitch and not to perform voiced/unvoiced (V/U) or silence decisions. The value of N_p during unvoiced or silent segments can be set arbitrarily (within a limited range). For example, N_p can be set to the value obtained from the pitch extractor which operates continuously, or if V/U and silence decisions are made (with much relaxed accuracy requirements) N_p can be set to a fixed predetermined value, such as to the number of samples in an average pitch period for a given class of speakers.

As mentioned earlier, higher bandwidth reduction factors or equivalently lower transmission bit rate can be achieved by further operating on the frequency-divided signal with other known data compression techniques. In particular, waveform coders such as adaptive delta modulation (ADM) [17] or other adaptive predictive coders (APC) [17], [18] can be effectively used. Such a combination can result in the reduction by a factor of 2-3 of the waveform coders typical transmission bit rate.

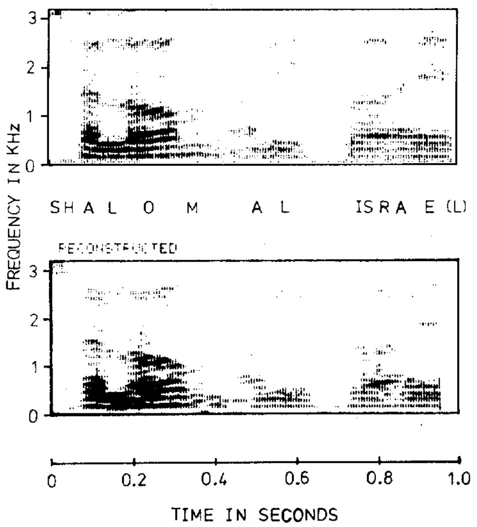

Fig. 9. Computer generated spectrograms of original and reconstructed sentences (in Hebrew), following frequency division and multiplication with the TDHS algorithms, by a factor of 3. A triangular window function was used.

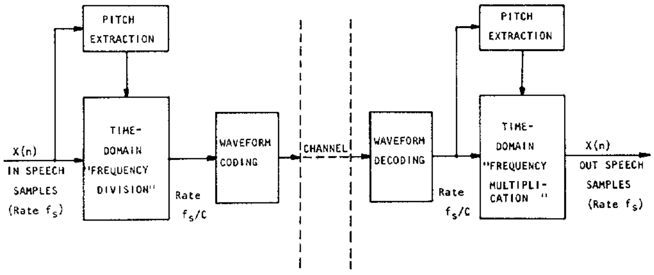
PITCH EXTRACTION
X(n)
IN SPEECH SAMPLES (Rate f_s)
TIME-DOMAIN "FREQUENCY DIVISION"
Rate f_s/C
WAVEFORM CODING
CHANNEL
WAVEFORM DECODING
Rate f_s/C
PITCH EXTRACTION
TIME-DOMAIN "FREQUENCY MULTIPLICATION"
X(n)
OUT SPEECH SAMPLES (Rate f_s)

Fig. 10. Block diagram of proposed vocoder system which combines time-domain harmonic scaling and waveform coding.

A particular proposition for a vocoder system which incorporates the TDHS algorithms and waveform coding has been recently presented by this author [14]. A general block diagram of the proposed system is shown in Fig. 10. In this system, pitch information is not transmitted, thus yielding a reduction in bit rate and possibly simplifying data transmission. Pitch information is reextracted at the receiver from the frequency-divided signal.[8]

Such a scheme is particularly appealing if a coder such as ADM is used since only bit (and not frame) synchronization is required. On the other hand, pitch extraction from the frequency-divided and noisy signal at the receiver is more difficult. This problem is now under study with preliminary results obtained with halving the bit rate of a 16 kbit ADM coder being encouraging.

It is also observed that, in contrast to other pitch-dependent vocoders, the reconstruction process is not performed by exciting the system with pitch pulses so that it is expected that the proposed system might be less sensitive to pitch errors. In particular, an error of the type of a double pitch period will usually have a small effect on the reconstructed signal in the proposed system but could be very disturbing in "pitch-excited" vocoders.

Due to the relatively small number of computations required by the TDHS algorithms (one multiplication and three additions per *output* sample, with $m = 2$, including the computation of the triangular window samples), an effective implementation for real-time applications could be based on an MOS microprocessor with a fast (STTL) serial multiplier as a peripheral device. Both functions of frequency division and multiplication can be performed by the same system, just by switching software. A hardware solution for a real-time pitch extractor was reported in the literature [6], but other more simple solutions can be thought of in view of the relaxed requirements from the pitch detector, as discussed above.

VI. Application of the TDHS Algorithms to Time-Scale Variation of Speech Signals

Two types of applications are considered in this section. The first is continuous speech processing for on-line time scaling of recorded speech signals. The second type is time-scale normalization of discrete speech utterances which is proposed here for reducing the amount of computations in computer-based isolated-word recognition systems (IWRS).

A. On-Line Time Scaling of Recorded Speech

Among the many applications of time scaling of recorded speech are "speeded speech" for the blind and slowed down speech for language teaching.

To implement a system which performs on-line time scaling of recorded speech by a desired factor q, the recorded input signal should be replayed at a $1/q$ speed and then be frequency scaled by the factor q. This frequency scaling can be effectively done by the TDHS algorithms which should in this case be performed in real time. Following frequency scaling with the TDHS algorithms (which does not change the time scale) the output signal occupies the original frequency band but is time scaled by a factor q since the input signal was replayed at $1/q$ speed.

Due to perception limitation, only scaling factors of up to 2 are practical. In extreme cases of very slow or very fast speakers, scaling factors of up to 3 (for slowing down the fast speaker and speeding up the slow speaker) can be applied.

Simulation results with a scaling factor of 2 for different speakers and texts have been informally judged to be very good when pitch tracking and a triangular window were used. Using a fixed value for N_p, to avoid pitch tracking, resulted in intelligible speech but of bad quality and an unpleasent metallic sound. The results are somewhat improved in this case if a higher value of m ($m = 4$) is used. Using m with a higher value than 1 results in an improvement of the "sampling method" which is equivalent to using $m = 1$ (or slightly better, due to tapering of rectangular window edges). If high quality speech is sought, pitch tracking should be implemented and a window function with $m = 2$ should be used.

Spectrograms which show typical simulation results with

[8] In implementing this system in real time, care must be taken that the scaling factors at the transmitter and receiver be exactly the same. This can be assured by constraining N_p to take values which are multiples of $|\mu - \delta|$ where $q = \mu/\delta$. For $q = 3$ or $\frac{1}{3}$ this means that N_p should be chosen even.

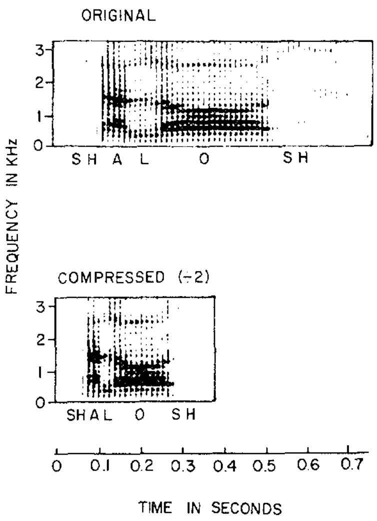

Fig. 11. Computer generated spectrograms of original and time-scaled sentences, as obtained by the TDHS algorithms with a triangular window function.

pitch tracking and a triangular window are presented in Fig. 11, for the two cases of time-scale compression and time-scale expansion (stretching) by a factor of 2.

B. Time-Scale Variation of Discrete Utterances with Application to IWRS

In typical isolated-word recognition system (IWRS) the recognition process of an unknown input utterance is divided into three main processing phases. These are [19]: preprocessing, feature or parameter extraction, and classification.

In an all-digital (computer-based) IWRS the principal computation load is the extraction of parameters representing the input utterance, and the classification of the unknown utterance by comparing the new parameter vector with stored vectors of the reference library.

Due to time duration variability of the speech utterances, time normalization and alignment of the new input vector must be performed when it is compared with any reference vector. Linear time scaling [20] and nonlinear time warping based on dynamic programming [20]-[23] are commonly used.

A noticeable reduction in the overall amount of computations could be achieved, however, if all input utterances were time scaled at the preprocessing phase, to the same short time duration. This will reduce the total number of parameters needed to be computed at the parameter extraction phase and will simplify classification, since all parameter vectors have the same low dimension and can be compared directly without any additional time-scale normalization.

This approach, based on using the TDHS algorithms for time scaling all input utterances at the preprocessing (preextraction) phase, was recently proposed by this author and tested in simulations [11]-[13].

In this particular application, pitch tracking can be avoided since the time-scaled signal need not be replayed and listened to, and the parameters extracted for utterance representation are required to present spectral envelope (and not fine structure).

With pitch tracking eliminated, the application of the TDHS algorithms with a fixed N_p (chosen to be larger than the number of samples in the longest pitch period expected for the given speaker, or class of speakers, using the system) requires relatively few computations. The simulations conducted so far have shown that an average reduction by a factor of more than 3 in the overall amount of computations can be achieved, without causing any degradation in system performance when compared to a similar system which uses dynamic programming (DP) and no preextraction time-scale compression (TSC).

It should be noted that the use of TSC does not exclude the possibility of applying DP at the classification phase for effecting a nonlinear time warping of the time scale which could be important in some multisyllabic (or multiphonemic) words. Since each utterance is time scaled to a short duration (scaling factors of up to 3 are practical), the use of DP at the classification phase does not change the large saving in computations at the parameter extraction phase, and having to classify low dimension vectors with DP also considerably reduces the number of computations at this phase. Thus, the proposed approach is computationally appealing but should be carefully examined with respect to the effect on recognition performance. To examine whether or not TSC degrades performance, two recognition systems have been implemented on a Nova-2 minicomputer. One uses TSC at the preprocessing phase, as proposed above, and the other system uses no TSC and applies DP for classification. The vocabulary consisted of the first ten Hebrew digits[9] (all but one have two syllables and typically range in duration from 0.5 to 0.9 s). The parameters chosen for representing the speech utterances were, as in [23], the partial correlation (PARCOR) coefficients which are based on a linear prediction model of the speech signal [24], [25]. Two different data bases have been used in the experiments. One contained the utterances of three speakers which were recorded in a quiet environment, each speaker recording four sets of ten digits at different sessions, several days apart. The second data base contained the utterances of two of the three speakers recorded in a noisy computer room environment. The input speech signal was bandlimited to the range of 200-3200 Hz and sampled at 10 kHz with a 12 bit A/D converter.

In the preprocessing phase, automatic utterance boundary determination was performed. For the system with TSC this phase also included the compression of all input utterances to the same duration of 0.3 s. In the parameter extraction phase 6 PARCOR parameters were extracted for each 15 ms speech segment, using the algorithm described in [26] and a Hanning window of 30 ms duration.

An example of the waveform obtained following TSC by a factor of 2 is presented in Fig. 12. The corresponding spectrograms are shown in Fig. 13. The PARCOR parameters extracted from the original word and used in the system with DP, and those extracted from the compressed word are shown in Fig. 14.

[9] The transliteration of the Hebrew digits is given in Appendix II.

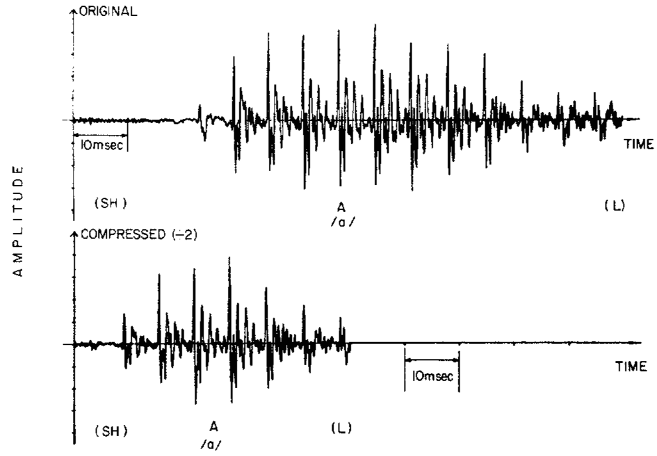

Fig. 12. Original and time-compressed waveforms of the voiced portion in the first syllable of the Hebrew word "Shalosh"–/ʃaloʃ/ (meaning "three").

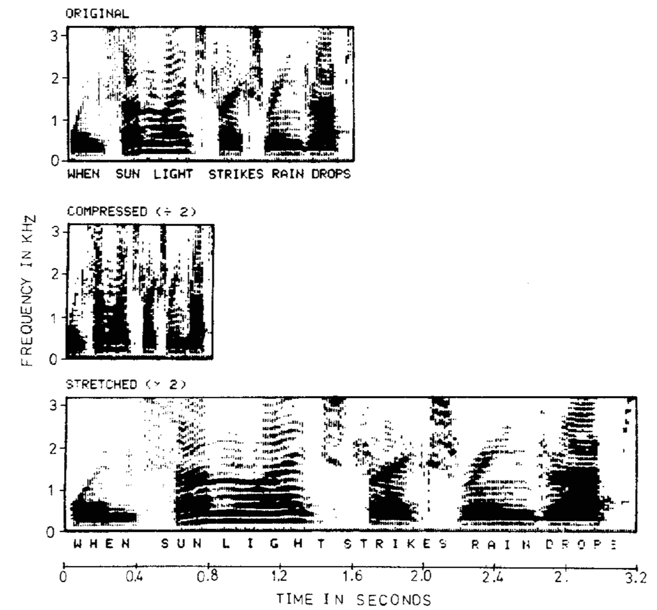

Fig. 13. Computer-generated spectrograms of the original and time-compressed Hebrew word "Shalosh."

Using the first ("quiet") data base, 360 tests have been performed in each of the system. Two hundred and forty tests have been performed with the "noisy" data base. The recognition scores are summarized in Tables I and II. Table III summarizes the processing steps and the amount of computations required for performing a recognition test in each system. More details on the performed simulations are given in the above-mentioned references [11]–[13].

It is noted that the system with TSC outperformed the system with DP under the quiet environment condition (a score of 99.1 percent in comparison to 98.3 percent) and achieved the same score (97.9 percent) under the noisy condition. The fact that TSC did not cause any degradation in performance is perhaps due to the way compression is performed, i.e., weighting of most input data and not just discarding segments, and effectively keeping spectral envelope information.

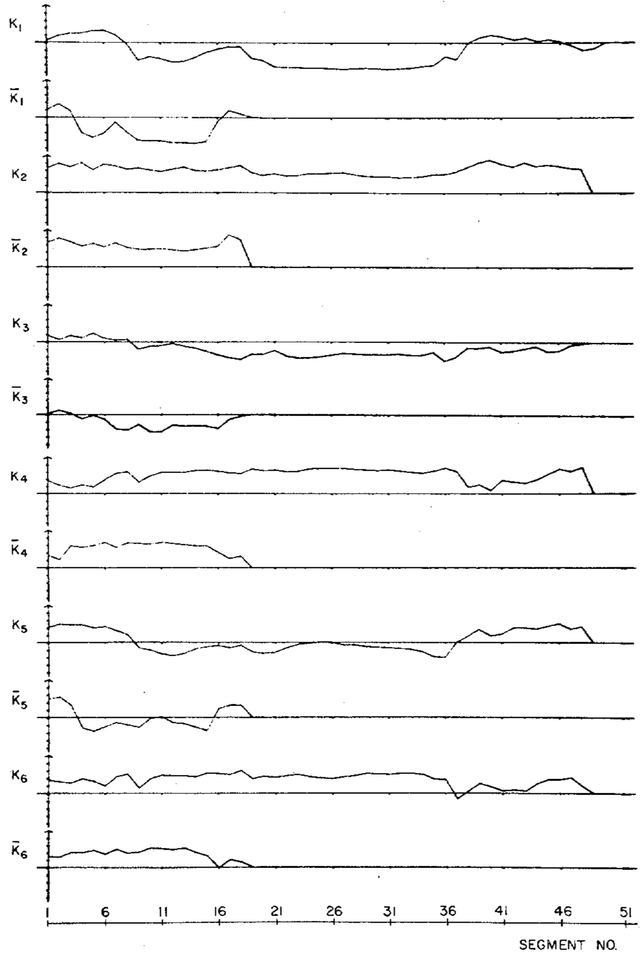

Fig. 14. Partial correlation (PARCOR) coefficients of the original ($k_i, i = 1, 2, \ldots 6$) and time-compressed ($\bar{k}_i, i = 1, 2, \ldots, 6$) signals obtained from the Hebrew word "Shalosh."

It is realized that the number of tests performed, the size and type of vocabulary used, and the number of speakers involved are not sufficient for reaching a final conclusion with respect to the use of the proposed TDHS algorithms for TSC in any general all-digital IWRS. Yet, it is believed that the results obtained support the proposed approach and point out its potential.

VII. Conclusions

It was shown that, by introducing pitch information into the frequency-scaling operation based on short-time Fourier analysis, it is possible to derive simple time-domain algorithms for frequency and time scaling of speech signals. An important factor in the realization of the new TDHS algorithms is the proper choice of the window function. Constraints to be satisfied by this function and a family of functions which satisfy these constraints were presented. For real time applications the triangular window is found, however, to be particularly suitable. Several applications were considered and propositions for effectively utilizing the new algorithms for implementing a vocoder system and for reducing computations in an all-digital isolated-word recognition system were presented. The results of preliminary simulations support the

TABLE I
SUMMARY OF RECOGNITION TEST RESULTS FOR IWRS WITH TSC

Type of Environment	Number of Tests	Number of Errors Per Speaker			Total Number of errors	Recognition Score (%)
		IE	DM	DA		
Quiet	360	3	0	0	3	99.1
Noisy (S/N = 20dB)	240	3	2	-	5	97.9

TABLE II
SUMMARY OF RECOGNITION TEST RESULTS FOR IWRS WITH DP

Type of Environment	Number of Tests	Number of Errors per Speaker			Total Number of Errors	Recognition Score (%)
		IE	DM	DA		
Quiet	360	6	0	0	6	98.3
Noisy (S/N = 20dB)	240	5	0	-	5	97.9

TABLE III
COMPARISON OF COMPUTATION LOAD IN TSC AND DP SYSTEMS

Type of System / Operation	TSC		DP	
	Adds	Mult.	Adds	Mult.
Detector of utterance boundaries and peak amplitude	42000	-	42000	-
Time Scale Compression (TSC)	9000	3000	-	-
Differencing (Pre-empahsis)	5700	-	14200	-
Extraction of PARCOR coefficients	39900	45600	99300	113500
Classification	9180	-	249900	11900
TOTAL:	105780	48600	405400	125400

above propositions and point to the potential of the TDHS algorithms for these and other applications involving frequency or time scaling of speech signals or perhaps other quasi-periodic wide-band signals.

Finally, it should be mentioned that in view of a recent work on the short-time Fourier transform [28], one can conclude that the proposed frequency-scaling approach, as given by (8), is a time-varying multiplicative modification of the short-time Fourier transform of the speech signal (as discussed in [28, Sect. V]). A presentation of the modifying function characteristics and the derivation of the above time-domain algorithms using this approach is now in preparation.

APPENDIX I

DERIVATION OF THE WINDOW FUNCTIONS $w_m(t)$

According to the requirements stated in Section IV, the window functions to be derived have finite duration ($w_m(t) = 0$ for $|t| > mN_pT/2$), are symmetrical ($w_m(t) = w_m(-t)$), and have the following additional properties:

$$w_m(lN_pT) = \begin{cases} 1, & l = 0 \\ 0, & l \neq 0 \end{cases} \tag{A1}$$

$$W_m(\omega) = 0 \quad \text{for} \quad \omega = 2\pi l/(N_pT); \quad l \neq 0, \tag{A2}$$

where $W_m(\omega)$ is the Fourier transform of $w_m(t)$.

Since $w_m(t)$ is of finite duration, we can write

$$w_m(t) = r(t) \sum_{k=-\infty}^{\infty} c_k e^{jk\omega_0 t} \tag{A3}$$

where $\omega_0 = 2\pi/(mN_pT)$ and

$$r(t) = \begin{cases} 1, & |t| < mN_pT/2 \\ 0, & |t| > mN_pT/2 \end{cases} \tag{A4}$$

and, hence,

$$W_m(\omega) = \sum_{k=-\infty}^{\infty} c_k R(\omega - k\omega_0), \tag{A5}$$

where $R(\omega)$ is the Fourier transform of $r(t)$, i.e.,

$$R(\omega) = 2 \sin (\omega mN_pT/2)/\omega. \tag{A6}$$

From (A6) we observe that for any nonzero integer l,

$$R(l\omega_0) = 0, \quad l \neq 0. \tag{A7}$$

Hence, a necessary and sufficient condition that (A2) be satisfied is that

$$c_{lm} = 0, \quad l \neq 0. \tag{A8}$$

The frequency response of the window functions that we are seeking should approximate the ideal low-pass filter response $H(\omega)$ where

$$H(\omega) = \begin{cases} H(0), & |\omega| < \pi/(N_pT) \\ H(0)/2, & \omega = \pi/(N_pT). \\ 0, & |\omega| > \pi/(N_pT) \end{cases} \tag{A9}$$

We choose here a particular approximation by setting c_k to be related to $H(\omega)$ by

$$c_k = H(k\omega_0), \quad k = 0, \pm 1, \pm 2, \cdots. \tag{A10}$$

Hence, $c_k = 0$ for all $k > \text{int}\ [m/2]$ and (A8) is certainly satisfied. We obtain therefore the following expression for $w_m(t)$, in place of (A3),

$$w_m(t) = \begin{cases} r(t)H(0) \displaystyle\sum_{k=-(m-1)/2}^{(m-1)/2} e^{jk\omega_0 t}, & m \text{ odd} \\ r(t)H(0) \left[\displaystyle\sum_{k=-(m-2)/2}^{(m-2)/2} e^{jk\omega_0 t} \right. \\ \left. + \frac{1}{2} (e^{-jm\omega_0 t/2} + e^{jm\omega_0 t/2}) \right], & m \text{ even} \end{cases} \tag{A11}$$

To satisfy $w_m(0) = 1$ [see (A1)] we set $H(0) = 1/m$ and obtain from (A11) the results stated in (53). It can easily be verified that the resulting window function also satisfies the second half of (A1), i.e., $w_m(lN_pT) = 0$ for $l \neq 0$.

Note also that the triangular window does not satisfy (A10) but does satisfy (A8) (with $m = 2$).

Appendix II

Transliteration of the First Ten Hebrew Digits

Zero—"Effess"
One—"Ahchat"
Two—"Shtayim"
Three—"Shalosh"
Four—"Arbah"
Five—"Chamesh"
Six—"Shesh"
Seven—"Shevah"
Eight—"Shemoneh"
Nine—"Teshah"

Acknowledgment

The author gratefully acknowledges the assistance of I. Engel in programming the algorithms and performing the speech recognition tests. The careful reading and helpful comments of the reviewers are greatly appreciated.

References

[1] M. R. Schroeder, "Vocoders: Analysis and synthesis of speech," *Proc. IEEE*, vol. 54, pp. 720-734, May 1966.

[2] J. L. Flanagan, "Spectrum analysis in speech coding," *IEEE Trans. Audio Electroacoust.*, vol. AU-15, pp. 66-69, June 1967.

[3] —, *Speech Analysis, Synthesis and Perception.* New York: Springer-Verlag, 1972.

[4] M. R. Schroeder, J. L. Flanagan, and E. A. Lundry, "Bandwidth compression of speech by analytic-signal rooting," *Proc. IEEE*, vol. 55, pp. 396-401, Mar. 1967.

[5] M. R. Schroeder, B. F. Logan, and A. J. Prestigiacomo, "New methods for speech analysis-synthesis and bandwidth compression," in *Proc. Stockholm Speech Commun. Seminar*, Stockholm, Sweden, Sept. 1962; also, see [3].

[6] J. J. Dubnowski, R. W. Schafer, and L. R. Rabiner, "Real-time digital hardware pitch detector," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-24, pp. 2-8, Feb. 1976.

[7] L. R. Rabiner *et al.*, "A comparative performance study of several pitch detection algorithms," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-24, pp. 399-418, Oct. 1976.

[8] J. L. Flanagan and R. M. Golden, "Phase vocoder," *Bell Syst. Tech. J.*, vol. 45, pp. 1493-1509, Nov. 1966.

[9] 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.*, vol. AU-21, pp. 165-174, June 1973.

[10] F. F. Lee, "Time compression and expansion of speech by the sampling method," *J. Audio Eng. Soc.*, vol. 20, pp. 738-742, Nov. 1972.

[11] D. Malah and I. Engel, "Computation reduction in all-digital isolated word recognition systems by efficient preextraction time normalization," Technion I.I.T., E.E. pub. 292, Nov. 1976.

[12] I. Engel, "A minicomputer implementation of a speech recognition system," M. Sc. thesis (in Hebrew), Technion – I.I.T., Jan. 1977.

[13] I. Engel and D. Malah, "A minicomputer implementation of an isolated-word recognition system," in *Proc. 10th Conv. Elec. Electron. Eng. in Israel*, Oct. 1977, paper A/2/4.

[14] D. Malah, "Digital harmonic compression of speech signals in the time domain," in *Proc. 10th Conv. Elec. Electron. Eng. in Israel*, Oct. 1977, paper A/2/3.

[15] R. W. Schafer and L. R. Rabiner, "Design of digital filter banks for speech analysis," *Bell Syst. Tech. J.*, vol. 50, pp. 3097-3115, Dec. 1971.

[16] A. Papoulis, *The Fourier Integral and Its Applications.* New York: McGraw-Hill, 1962, p. 47.

[17] N. S. Jayant, "Digital coding of speech waveforms: PCM, DPCM and DM quantizers," *Proc. IEEE*, vol. 62, pp. 611-632, May 1974.

[18] B. S. Atal and M. R. Schroeder, "Adaptive predictive coding of speech signals," *Bell Syst. Tech. J.*, vol. 49, pp. 1973-1986, Dec. 1970.

[19] D. R. Reddy, "Speech recognition by machine: A review," *Proc. IEEE*, vol. 64, pp. 501-531, Apr. 1976.

[20] G. M. White and R. B. Neely, "Speech recognition experiments with linear prediction, bandpass, filtering and dynamic programming," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-24, pp. 183-188, Apr. 1976.

[21] V. M. Velichko and N. G. Zagoruiko, "Automatic recognition of 200 words," *Int. J. Man-Machine Studies*, vol. 2, pp. 223-234, 1970.

[22] F. Itakura, "Minimum prediction residual applied to speech recognition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 62-72, Feb. 1975.

[23] A. Ichikawa, Y. Nakano, and K. Nakata, "Evaluation of various parameter sets in spoken digits recognition," *IEEE Trans. Audio Electroacoust.*, vol. AU-21, pp. 202-209, June 1973.

[24] B. S. Atal and S. L. Hanauer, "Speech analysis and synthesis by linear prediction of the speech wave," *J. Acoust. Soc. Amer.*, vol. 50, no. 2, pp. 637-655, 1971.

[25] J. Makhoul, "Linear prediction: A tutorial review," *Proc. IEEE*, vol. 63, no. 4, pp. 561-580, Apr. 1975.

[26] J. D. Markel and A. H. Gray, Jr., "On autocorrelation equations with application to speech analysis," *IEEE Trans. Audio Electroacoust.*, vol. AU-21, pp. 69-79, Apr. 1973.

[27] D. Malah, "Time-domain algorithms for time-scale variation of speech signals," Technion I.I.T., E.E. publ. 280, May 1976.

[28] J. B. Allen and L. R. Rabiner, "A unified approach to short-time Fourier analysis and synthesis," *Proc. IEEE*, vol. 65, pp. 1558-1564, Nov. 1977.