THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 62, No. 6, July–August 1983
Printed in U.S.A.

Sample Reduction and Subsequent Adaptive Interpolation of Speech Signals

By R. STEELE* and F. BENJAMIN†

(Manuscript received December 15, 1982)

In this paper we investigate the effect of rejecting every nth speech sample and replacing it by means of adaptive interpolation. The interpolation procedure attempts to minimize the mean square interpolation error by recomputing the autocorrelation function of the speech sequence every W samples. We describe three methods of computing the correlation function. An iterative procedure is evaluated for estimating the correlation function of a speech sequence whose every nth sample has been discarded. For speech bandlimited to 3.2 kHz, sampled at 8 kHz, and $n = 4$, $W = 256$, the gain in signal-to-noise ratio (s/n) achieved by adaptive interpolation compared to nearest neighbor average interpolation was 14 and 8 dB, depending on whether the correlation function was computed from the original speech, or by using the iterative procedure, respectively. The effect of varying n from 2 to 6 was also investigated. Finally, we applied the interpolation procedures to 8-bit μ-law pulse code modulation (PCM), $\mu = 255$, reducing 64 kb/s transmission to 48 kb/s by rejecting one PCM word in four. The recovered speech after interpolation had a s/n that approximated that of conventional 56 kb/s μ-law PCM speech.

I. INTRODUCTION

Sampling of speech signals is usually performed at a rate high enough to prevent objectionable aliasing. Thus a speech signal whose bandwidth extends from 0.3 to 3.3 kHz is typically sampled at 8 kHz. After the speech signal has been sampled, the subsequent processing

* University of Southampton, Southampton, England. † Bell Laboratories.

of the samples depends on the application. In digital encoding the samples are converted into a digital (usually binary) sequence. Clearly, the lower the sampling rate the lower the bit rate of the digitally encoded speech, and the smaller the required channel capacity. There are numerous techniques for bit-rate reduction that have been compiled and categorized.[1,2] Our intention here is to dwell on a specific subset of bit-rate reduction, namely that arising from discarding speech samples at the transmitter or at a node in a network, and replacing them at the receiver by interpolation. The impetus for this approach arose from two unrelated interests, analog speech scramblers and bit-rate reduction in μ-law pulse code modulation (PCM). Scramblers that offer a reasonable amount of security, such as modulo masking scramblers, expand the bandwidth to half the sampling rate. Thus, operating on 8-kHz sampled speech, we wanted to reduce the sampling rate by discarding every fourth sample such that, after scrambling, the bandwidth would be 3 kHz, i.e., confined to the bandwidth of telephonic speech. The descrambled samples at the receiver would be brought back to the original 8-kHz rate by interpolating the missing samples. In the case of μ-law PCM our interest was to discard a percentage (typically 25 percent) of the words representing encoded speech, enabling other data to have a free ride. At the receiver the data would be removed and the missing speech samples recovered by interpolation.

Interpolation ideas are part of everyday life; we are always filling in the gaps in our perception by interpolation, extrapolation, and prediction. We will avoid the luxury of philosophizing, and briefly review some of the interesting aspects of interpolation that are related to speech encoding. Mathews,[3] in an attempt to make significant reductions in the sampling rate of speech, considered extremal encoding. With this technique the amplitudes of the speech signal at its extremes and the time intervals between these extremes are transmitted in an encoded format. At the receiver the extremes can be widely separated, and the interpolation of the missing samples can be thought of as curve fitting, i.e., passing a quadratic function through the extreme sample and two nearest received samples. In the 1960's there was considerable interest in interpolation, which Kortman[4] defined as "the process of after-the-fact polynomial curve fitting to eliminate redundant data samples." The interpolators[4–6] tended to be concerned with zero-order, first-order, and fan interpolators, and were closely related to aperture predictors.[4–7] Andrews et al.[6] considered straight-line optimum, optimum interpolation filter, $(\sin x)/x$, Lagrange, and Fourier reconstruction interpolators. The use of piecewise polynomials, called splines, have been extensively investigated (see Ref. 8 and its bibliography). More recently, there have been a spate of publications[9] on digital speech interpolation, although the interpolation there was mainly concerned with Time Assignment Speech Interpolation (TASI)-type systems. Interpolation techniques have also been applied to reduce distortion in packet switching of speech when a packet is lost or discarded.[10] For an understanding of the existing low-pass filtering procedures of interpolation and decimation of digital signals, the reader is directed to the in-depth review presented by Crochiere and Rabiner.[11]

Having commented on some of the existing interpolation methods let us now proceed to the issues to be addressed here. Consider the situation of being presented with the speech samples, or say μ-law PCM words, at some point in a communications network, together with a system control demand to eliminate J samples (or words) per W samples (or words). This could happen, for example, if there were a sudden increase in traffic. Rejection of these J samples might precipitate unacceptable degradation in the recovered speech unless we introduced at the receiver replacement samples that closely approximate those samples rejected. We must therefore decide at the outset on the means of reinserting the J samples, for example, by prediction or interpolation, using those samples not discarded. Having replaced the missing samples, we need to establish criteria for judging the quality of the recovered speech signal. Is a single criterion such as mean square error sufficient, or is a combination of objective and subjective measures required? Coupled with the issues of reinsertion and quality are the criteria of how to select which J samples are to be rejected in every W samples. Should we determine if speech is present as distinct from silence, and if so, whether it is voiced or unvoiced? Is it better to reject small groups of samples (e.g., in silence periods), leaving clusters of samples intact because rejection followed by subsequent interpolation of one of these samples might cause significant speech degradation, or should we always endeavor to retain samples on either side of a rejected sample, and so on?

Clearly, the choices are legion, and faced with this situation we have imposed a set of guidelines that are not concerned with what the samples (or words) represent in the speech signal. For example, we make no attempt to recognize if the talker is male or female, whether the speech is voiced or unvoiced, or in transition, and so on. Instead, we simply reject samples on a periodic basis, and reinsert them at their destination by means of adaptive interpolation. By making the interpolation adaptive we take cognizance of the local statistics of the speech signal, specifically, the utilization of the signal's correlative properties. This is the only characteristic of the speech signal that is exploited. Our approach is, therefore, oriented to ease implementation rather than to squeeze the maximum advantage from the properties of speech. Indeed, the method is applicable to other types of analog signals (e.g., modem-generated data signals), provided the signals possess correlative properties that are capable of exploitation in the interpolation processes. Our strategy is as follows:

1. Discard samples on a periodic basis, with no two consecutive samples being rejected.
2. Process blocks of W samples at a time.
3. Exploit the local statistics of the speech samples by determining their correlation function over each block. Approximate methods for determining the correlation function are required when the sequence has had J of its W samples rejected.
4. Apply adaptive interpolation.
5. Employ quality criteria based on the minimization of the mean square error, justified by the prior knowledge that the mean square error values achieved will conform to perceptual standards based on informal listening tests.

Having introduced the notion of sample or word rejection to reduce the baud or bit rate, and an outline of how to discard the samples or words and reintroduce them at their destination, we will now formulate the problem and its solution in detail.

II. THE PROBLEM

Consider a speech signal $x(t)$, bandlimited to $f_c H_z$ and sampled at $f_s H_z$ to yield the sequence $\{x_k\}$, where f_s satisfies

$$f_s \geq 2f_c. \tag{1}$$

This sequence $\{x_k\}$ could be encoded into binary words, or we could be given binary words at a node in the network. In such situations our description of the problem would be in data words. However, for ease of explanation we will confine our discussion to operations on samples unless otherwise stated.

To reduce the number of samples per second in $\{x_k\}$, we discard every nth sample. This is achieved by clocking the speech samples into a gear-down changing buffer under the auspices of a clock operating at f_s samples per second, such that, after every n samples are inserted into the buffer, the clock is inhibited for one clock period. The arrangement is shown in Fig. 1. The samples are clocked out of the buffer at a rate

$$F_s < 2f_c \tag{2}$$

to yield a sequence $\{y_k\}$ whose components are uniformly spaced in time by $1/F_s$. Figure 2 shows $\{x_k\}$ and $\{y_k\}$ for arbitrary segments of sampled speech. Observe that $\{y_k\}$ has its parameter F_s related to f_s by

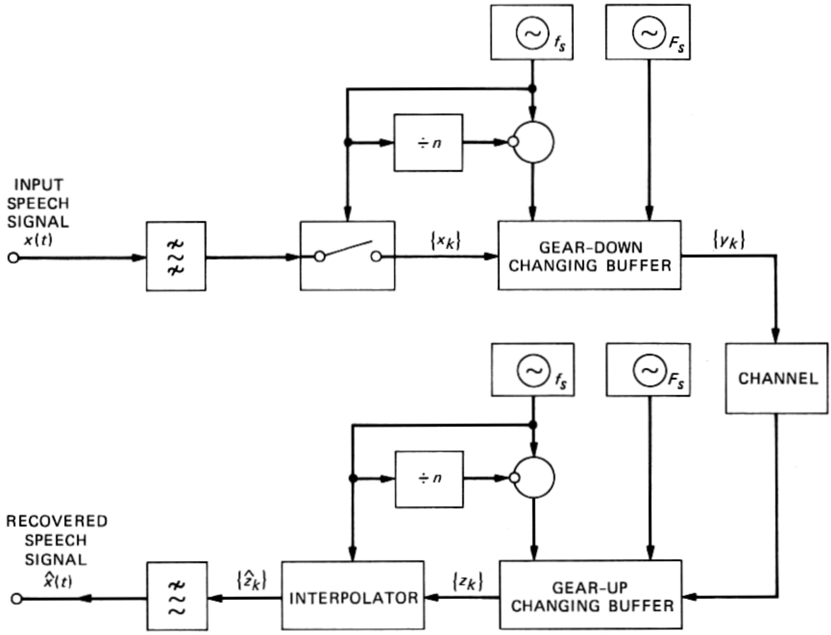


Fig. 1—Arrangement for decreasing the sample rate.

$$F_s = \left(\frac{n-1}{n}\right) f_s. \tag{3}$$

Suppose $\{y_k\}$ with its symbol rate lower than $\{x_k\}$ is transmitted over an ideal channel. Let us further assume that the receiver is able to return the samples in $\{y_k\}$ to the relative time positions they occupied in the original speech sequence $\{x_k\}$ using the gear-up changing buffer shown in Fig. 1. This newly formed sequence, $\{z_k\}$, generated at a rate $1/f_s$, has one out of every n samples absent owing to the sample reduction process at the transmitter. A small, arbitrary segment of this sequence $\{z_k\}$, corresponding to a particular $\{x_k\}$ and $\{y_k\}$, is displayed in Fig. 2. Our problem is to replace those samples rejected at the transmitter with substitutes of acceptable accuracy whose generation is not excessively complex. The approach employed is adaptive interpolation.

III. ADAPTIVE INTERPOLATION

The speech sequence $\{z_k\}$ is divided into sequential blocks having W sample positions spaced $1/f_s$ seconds apart. The sequence $\{z_k\}$ is therefore composed of W samples from $\{x_k\}$ with every nth sample absent. Our purpose is to interpolate the missing samples to produce

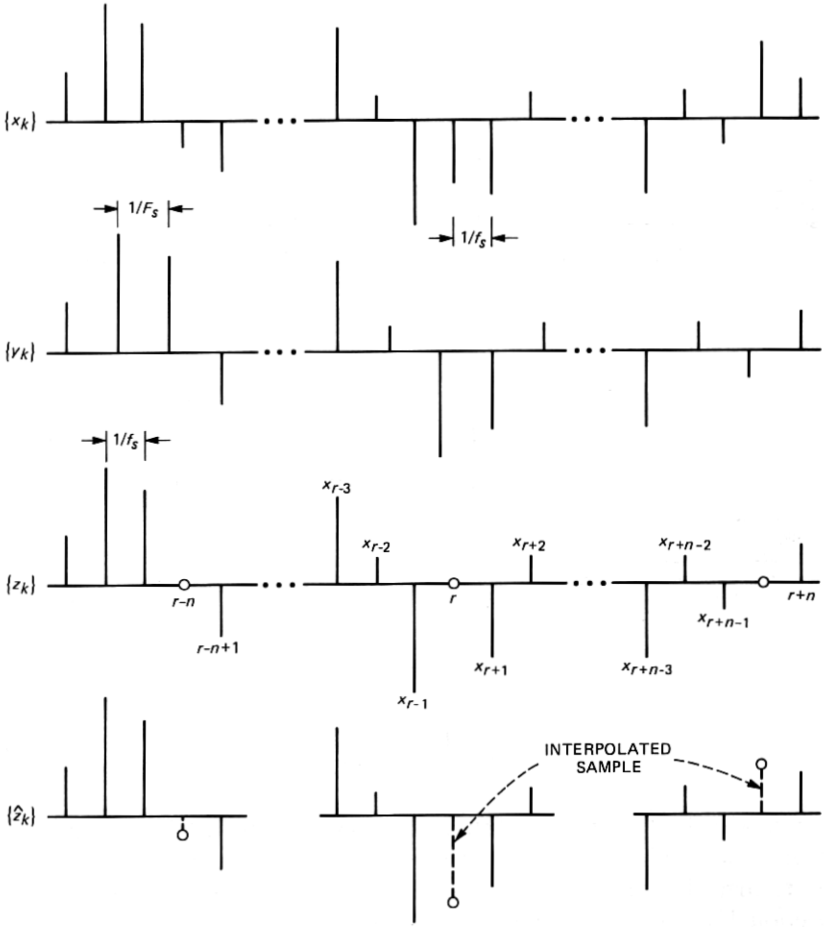


Fig. 2—Sampled sequences for arbitrary segment of speech; $\{x_k\}$ is the original speech sequence; $\{y_k\}$ is the sequence after sample reduction and gearing changing; $\{z\}$ is $\{x_k\}$ with every nth sample absent; $\{\hat{z}\}$ is the recovered speech sequence.

a sequence for the first block having components

$$x_1,\ \cdots,\ x_{n-1},\ \hat{z}_n,\ x_{n+1},\ \cdots,\ x_{W-2},\ x_{W-1}\hat{z}_W,$$

where

$$\hat{z}_r; \qquad r = n,\ 2n,\ \cdots,\ W - n,\ W$$

are the interpolated samples. When interpolating each speech sample, we will use λ past and λ previous samples, and restrict λ to be

$$\lambda \leq n - 1. \tag{4}$$

Thus, a missing sample at the rth sampling instant (see Fig. 2) is formed by interpolation according to

$$\hat{z}_r = \sum_{i=-\lambda}^{-1} a_i x_{r+i} + \sum_{i=1}^{\lambda} a_i x_{r+i}, \tag{5}$$

where a_i are the interpolation parameters. Equation (5) may be written as

$$\hat{z}_r = \sum_{i=-\lambda}^{\lambda} a_i x_{r+i}, \tag{6}$$

where $a_0 = 0$. The interpolation error of the rth sample is

$$e_r = x_r - \hat{z}_r \tag{7}$$

and the square of this error is

$$\begin{aligned} e_r^2 &= x_r^2 - 2x_r \sum_{i=-\lambda}^{\lambda} a_i x_{r+i} + \left(\sum_{i=-\lambda}^{\lambda} a_i x_{r+i} \right)^2 \\ &= x_r^2 - 2x_r \sum_{i=-\lambda}^{\lambda} a_i x_{r+i} + \sum_{i=-\lambda}^{\lambda} a_i^2 x_{r+i}^2 + 2 \sum_{i=-\lambda}^{\lambda} \sum_{j=-\lambda}^{\lambda} a_i x_{r+i} a_j x_{r+j} \end{aligned} \tag{8}$$

with $i \neq j$ and $j > i$. To determine the interpolation parameters we proceed as follows. The square of the error is summed over the block of samples

$$\begin{aligned} E_r^2 = \sum_{r=n}^{W} e_r^2 = \sum_{r=n}^{W} x_r^2 - 2 \sum_{r=n}^{W} \left\{ x_r \sum_{i=-\lambda}^{\lambda} a_i x_{r+i} \right\} \\ + \sum_{r=n}^{W} \sum_{i=-\lambda}^{\lambda} a_i^2 x_{r+i}^2 + 2 \sum_{r=n}^{W} \sum_{i=-\lambda}^{\lambda} \sum_{j=-\lambda}^{\lambda} a_i x_{r+i} a_j x_{r+j}, \end{aligned} \tag{9}$$

where $r = n, 2n, \cdots, W$. To select coefficients that minimize this summation, we partially differentiate E_r^2 with respect to each of the coefficients and set the result to zero. After rearranging the equations we have

$$\sum_{r=n}^{W} x_r x_{r+j} = \sum_{i=-\lambda}^{\lambda} \sum_{r=n}^{W} a_i x_{r+i} x_{r+j}, \tag{10}$$

where

$$\begin{aligned} j &= \pm 1, \pm 2, \cdots, \pm\lambda \\ i &= -\lambda, -\lambda + 1, \cdots, \lambda \\ r &= n, 2n, \cdots, W - n, W \\ W &\geq 2n. \end{aligned}$$

The interpolation coefficients in eq. (10) can be represented in vector form as

$$\alpha = A^{-1}C, \tag{11}$$

where

$$\alpha = [a_{-\lambda}, a_{-\lambda+1}, \cdots, a_{-1}, a_1, \cdots, a_{\lambda-1}, a_\lambda]^T, \tag{12}$$

$$C = [R(0, -\lambda), R(0, -\lambda + 1), \cdots, R(0, -1), R(0, 1), \cdots, R(0, \lambda - 1), R(0, \lambda)]^T, \tag{13}$$

and the superscripts −1 and T represent inverse and transpose operations, respectively. The rectangular matrix A is of order 2λ, and its elements are presented in Table I. The concentric dotted enclosures, commencing with the inner one, refer to the A matrix for $\lambda = 1, 2, 3, \cdots$. The elements are given by

$$R(k, j) = \frac{\sum_{r=n}^{W} x_{r+k}x_{r+j}}{\sum_{r=n}^{W} x_{r+j}^2}$$

$$k = \pm 1, \pm 2, \cdots, \pm\lambda$$

$$j = \pm 1, \pm 2, \cdots, \pm\lambda$$

$$r = n, 2n, \cdots, W, \tag{14}$$

and $R(k, j)$ will be referred to as the correlation function $R(k, j)$. The vector C has elements $R(0, j)$, i.e., the elements are given by eq. (13) with k always zero.

As an example of the application of eq. (11), consider the case of $W = 32$, $n = 4$, $\lambda = 2$, whence

$$\begin{bmatrix} a_{-2} \\ a_{-1} \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 1 & R(-1, -2) & R(1, -2) & R(2, -2) \\ R(-2, -1) & 1 & R(1, -1) & R(2, -1) \\ R(-2, 1) & R(-1, 1) & 1 & R(2, 1) \\ R(-2, 2) & R(-1, 2) & R(1, 2) & 1 \end{bmatrix}^{-1} \begin{bmatrix} R(0, -2) \\ R(0, -1) \\ R(0, 1) \\ R(0, 2) \end{bmatrix} \tag{15}$$

and eq. (14) becomes for the A and C matrices

$$R(k, j) = \frac{\sum_{r=4}^{32} x_{r+k}x_{r+j}}{\sum_{r=4}^{32} x_{r+j}^2}; \qquad \begin{aligned} k &= 0, \pm 1, \pm 2 \\ j &= \pm 1, \pm 2 \\ r &= 4, 8, 12, 16, 20, 24, 28, 32 \end{aligned}$$

Table I—The A matrix

1	$R(-\lambda + 1, -\lambda)$	· ·	$R(-5, -\lambda)$	$R(-4, -\lambda)$	$R(-3, -\lambda)$	$R(-2, -\lambda)$	$R(-1, -\lambda)$	$R(1, -\lambda)$	$R(2, -\lambda)$	$R(3, -\lambda)$	$R(4, -\lambda)$	$R(5, -\lambda)$	· ·	$R(\lambda - 1, -\lambda)$	$R(\lambda, -\lambda)$
$R(-\lambda, -\lambda, + 1)$	· ·	· ·	·	·	·	·	·	·	·	·	·	·	· ·	· ·	$R(\lambda, -\lambda + 1)$
·	· ·	· ·	·	·	·	·	·	·	·	·	·	·	· ·	· ·	·
·	· ·	· ·	·	·	·	·	·	·	·	·	·	·	· ·	· ·	·
$R(-\lambda, -5)$	· ·	· ·	1	$R(-4, -5)$	$R(-3, -5)$	$R(-2, -5)$	$R(-1, -5)$	$R(1, -5)$	$R(2, -5)$	$R(3, -5)$	$R(4, -5)$	$R(5, -5)$	· ·	· ·	$R(\lambda, -5)$
$R(-\lambda, -4)$	· ·	· ·	$R(-5, -4)$	1	$R(-3, -4)$	$R(-2, -4)$	$R(-1, -4)$	$R(1, -4)$	$R(2, -4)$	$R(3, -4)$	$R(4, -4)$	$R(5, -4)$	· ·	· ·	$R(\lambda, -4)$
$R(-\lambda, -3)$	· ·	· ·	$R(-5, -3)$	$R(-4, -3)$	1	$R(-2, -3)$	$R(-1, -3)$	$R(1, -3)$	$R(2, -3)$	$R(3, -3)$	$R(4, -3)$	$R(5, -3)$	· ·	· ·	$R(\lambda, -3)$
$R(-\lambda, -2)$	· ·	· ·	$R(-5, -2)$	$R(-4, -2)$	$R(-3, -2)$	1	$R(-1, -2)$	$R(1, -2)$	$R(2, -2)$	$R(3, -2)$	$R(4, -2)$	$R(5, -2)$	· ·	· ·	$R(\lambda, -2)$
$R(-\lambda, -1)$	· ·	· ·	$R(-5, -1)$	$R(-4, -1)$	$R(-3, -1)$	$R(-2, -1)$	1	$R(1, -1)$	$R(2, -1)$	$R(3, -1)$	$R(4, -1)$	$R(5, -1)$	· ·	· ·	$R(\lambda, -1)$
$R(-\lambda, 1)$	· ·	· ·	$R(-5, 1)$	$R(-4, 1)$	$R(-3, 1)$	$R(-2, 1)$	$R(-1, 1)$	1	$R(2, 1)$	$R(3, 1)$	$R(4, 1)$	$R(5, 1)$	· ·	· ·	$R(\lambda, 1)$
$R(-\lambda, 2)$	· ·	· ·	$R(-5, 2)$	$R(-4, 2)$	$R(-3, 2)$	$R(-2, 2)$	$R(-1, 2)$	$R(1, 2)$	1	$R(3, 2)$	$R(4, 2)$	$R(5, 2)$	· ·	· ·	$R(\lambda, 2)$
$R(-\lambda, 3)$	· ·	· ·	$R(-5, 3)$	$R(-4, 3)$	$R(-3, 3)$	$R(-2, 3)$	$R(-1, 3)$	$R(1, 3)$	$R(2, 3)$	1	$R(4, 3)$	$R(5, 3)$	· ·	· ·	$R(\lambda, 3)$
$R(-\lambda, 4)$	· ·	· ·	$R(-5, 4)$	$R(-4, 4)$	$R(-3, 4)$	$R(-2, 4)$	$R(-1, 4)$	$R(1, 4)$	$R(2, 4)$	$R(3, 4)$	1	$R(5, 4)$	· ·	· ·	$R(\lambda, 4)$
$R(-\lambda, 5)$	· ·	· ·	$R(-5, 5)$	$R(-4, 5)$	$R(-3, 5)$	$R(-2, 5)$	$R(-1, 5)$	$R(1, 5)$	$R(2, 5)$	$R(3, 5)$	$R(4, 5)$	1	· ·	· ·	$R(\lambda, 5)$
·	· ·	· ·	·	·	·	·	·	·	·	·	·	·	· ·	· ·	·
·	· ·	· ·	·	·	·	·	·	·	·	·	·	·	· ·	· ·	·
	· ·	· ·	·	·	·	·	·	·	·	·	·	·	· ·	· ·	
$R(-\lambda, \lambda - 1)$	· ·	· ·	·	·	·	·	·	·	·	·	·	·	· ·	· ·	$R(\lambda, \lambda - 1)$
$R(-\lambda, \lambda)$	$R(-\lambda + 1, \lambda)$	· ·	$R(-5, \lambda)$	$R(-4, \lambda)$	$R(-3, \lambda)$	$R(-2, \lambda)$	$R(-1, \lambda)$	$R(1, \lambda)$	$R(2, \lambda)$	$R(3, \lambda)$	$R(4, \lambda)$	$R(5, \lambda)$	· ·	$R(\lambda - 1, \lambda)$	1

When $r = 32$ and $|k|$ or $|j| \geq 1$, samples are used that reside in the subsequent block.

When the block size W is large, typically in excess of 256 samples, the computations required to solve eq. (11) can be reduced at the expense of a small increase in interpolation noise power. This is accomplished by replacing $R(k, j)$ of eq. (14) by the correlation function

$$R(\tau) = \frac{(W-\tau)\sum_{i=1}^{W-\tau} x_i x_{i+\tau} - \left(\sum_{i=1}^{W-\tau} x_i\right)\left(\sum_{i=1}^{W-\tau} x_{i+\tau}\right)}{\left\{\left[(W-\tau)\sum_{i=1}^{W-\tau} x_i^2 - \left(\sum_{i=1}^{W-\tau} x_i\right)^2\right] \cdot \left[(W-\tau)\sum_{i=1}^{W-\tau} x_{i+\tau}^2 - \left(\sum_{i=1}^{W-\tau} x_{i+\tau}\right)^2\right]\right\}^{1/2}}, \tag{16}$$

where

$$\tau = |k - j| \tag{17}$$

and k and j have values prescribed in eq. (14). For a further small increase in interpolation noise, which typically reduces the recovered signal-to-noise ratio (s/n) by a couple of decibels compared to when eq. (16) is used, $R(\tau)$ can be simplified to

$$R(\tau) = \frac{\sum_{r=1}^{W-\tau} x_r x_{r+\tau}}{\sum_{r=1}^{W} x_r^2}, \tag{18}$$

where r is integer-valued. Observe that, in computing the interpolation coefficients with the aid of eqs. (16) or (18), only λ values need be determined as

$$a_{-p} = a_p; \qquad p = 1, 2, \cdots, \lambda. \tag{19}$$

By contrast, when $R(k, j)$ is used in preference to $R(\tau)$, eq. (19) does not apply and 2λ coefficients must be computed.

Thus, provided we can compute $R(k, j)$ or $R(\tau)$, over a duration of W/f_s, where W is the block length, we can determine the interpolation parameters contained in the vector α. Employing eq. (5) we can estimate the missing samples by means of interpolation. The mean squared error between $\{x_k\}$, and the recovered sequence $\{\hat{z}_k\}$ containing the interpolated samples (see the example shown in Fig. 2), is approximately minimized, provided W is sufficiently large. Practical values of W are determined in Section VI.

IV. INTERPOLATION PARAMETERS DERIVED FROM THE INPUT DATA

The interpolation parameters are computed by first finding the correlation function $R(\theta)$, where θ is k, j, or τ. Equation (14) shows $R(\theta)$ as dependent on the input sequence $\{x_k\}$. Clearly, as $\{x_k\}$ is only known at the transmitter, it follows that the interpolation vector α must be computed at the transmitter and multiplexed with the slowed-down speech samples $\{y_k\}$. Consequently,

$$Y = \left(\frac{n-1}{n}\right) W + \nu \tag{20}$$

samples are transmitted every W/f_s seconds, where Y is composed of $W(n-1)/n$ speech samples and ν interpolation parameter samples. The value of ν is λ or 2λ, depending on whether θ is τ or k, j, respectively. This means that F_s of eq. (3) is modified to

$$F_s' = F_s + \frac{\nu}{W} f_s. \tag{21}$$

For example, if $f_s = 8$ kHz, $n = 4$, $F_s = 6$ kHz, $W = 256$, $\lambda = 3$, $\theta = \tau$, then $F_s' = 6.093$ kHz. We also observe from eq. (20) that three interpolation samples are sent for every 192 speech samples. The values of λ and W as a function of s/n are present in Section VI.

V. INTERPOLATION PARAMETERS DERIVED FROM THE RECEIVED DATA

The receiver produces the sequence $\{z_k\}$ whose samples are spaced apart by $1/f_s$, and in every nth sample position one sample is missing, as shown in Fig. 2. The receiver has the task of estimating the interpolation coefficients from $\{z_k\}$, and must therefore commence by calculating the correlation function $R(\tau)$ without the full knowledge of the original speech sequence $\{x_k\}$. In this situation we proceed as follows. The missing samples are found as an average of adjacent speech samples

$$\bar{x}_r = (x_{r-1} + x_{r+1})/2 \tag{22}$$

until a sequence $\{\bar{z}_k\}$ consisting of $W(n-1)/n$ original speech samples and W/n interpolated samples is formed, where each sample is equally spaced from its neighbor by $1/f_s$ seconds. This sequence corresponds to a recovered speech signal that has considerable distortion, particularly for female speakers. Instead of accepting $\{\bar{z}_k\}$ as the recovered speech sequence, we use it solely for the purpose of computing $R(\tau)$, and hence the vector α containing a_{-1} and a_1. We now remove the samples introduced by eq. (22) and replace them by using adaptive interpolation based on the two nearest neighbors, viz:

$$_2x_r = a_{-1}x_{r-1} + a_1x_{r+1}, \tag{23}$$

where a_{-1} and a_1 are found using eq. (11) with $R(k, j)$ replaced by $R(\tau)$ of eq. (18). The new recovered speech sequence $\{_2z_k\}$ has interpolated samples formed according to eq. (23), and in general contains less distortion than $\{\bar{z}_k\}$. However, we can further reduce the distortion. The function $R(\tau)$ is again computed, this time from $\{_2z_k\}$. The interpolated samples in $\{_2z_k\}$, derived with the aid of eq. (23), are rejected and replaced by

$$_4x_r = a_{-2}x_{r-2} + a_{-1}x_{r-1} + a_1x_{r+1} + ax_{r+2} \tag{24}$$

to yield the speech sequence $\{_4z_k\}$. The correlation function $R(\tau)$ of sequence $\{_4z_k\}$ is found, and those interpolated samples previously formulated with the aid of eq. (24) are exchanged for

$$_6x_r = \sum_{i=-3}^{3} a_i x_{r-i}, \qquad a_0 = 0. \tag{25}$$

This process of using two more samples per iteration in the interpolation procedure continues until the limits of the summation in eq. (25) become $n - 1$, when the final recovered speech sequence $\{\hat{z}_k\}$ is obtained. It should be noted that more than $2(n - 1)$ samples can be used in each interpolation process, but the improvement in interpolation accuracy results in a significant increase in complexity.

Often it is sufficient to produce the sequence $\{\bar{z}_k\}$, compute $R(\tau)$ from $\{\bar{z}_k\}$, remove the samples formed by averaging the adjacent two samples, and with the aid of $\lambda \leq n - 1$ samples on either side of each discarded sample, insert the missing samples by adaptive interpolation. When this process is adopted, it will be referred to as λ-interpolation. However, when the same λ samples are used in the interpolation process, and the iteration procedure of eqs. (22) to (25) activated, we will refer to this interpolation scheme as λ-with-iteration.

Observe that $R(\tau)$ is used in the λ-with-iteration scheme. If $R(k, j)$ is employed instead of $R(\tau)$, the final interpolation is not better than that of the sample used to approximate x_r. For example, if we replaced x_r by $\bar{x}_r$, the iterative algorithm would merely attempt to minimize the error power between the $\bar{x}_r$ samples and the interpolated samples. Unlike the application of $R(k, j)$, the $R(\tau)$ function does not enable an exact minimization of interpolation error power to be achieved at the transmitter. However, it transpires that by computing $R(\tau)$ at the receiver, and using the iterative procedure to interpolate the missing samples, the interpolation noise is significantly reduced (see Section 6.2).

VI. RESULTS

The speech signal used in our experiments consisted of two concatenated sentences, "Live wires should be kept covered," and "To reach the end he needs much courage." These were spoken by a male and female, respectively. The signal was bandlimited from 0.3 to 3.2 kHz and sampled at 8 kHz to give the input speech sequence $\{x_k\}$. This sequence is displayed in Fig. 3, together with its spectrogram, where the higher frequencies have been preemphasized.

6.1 Interpolation parameters generated from the original speech

The gear-down changing procedure was invoked (see Figs. 1 and 2) whereby the sampling rate of 8 kHz was decreased to a uniform F_s-kHz rate by rejecting every nth sample, and adjusting the sample spacing to provide the output sequence $\{y_k\}$. This sequence was assumed to be transmitted through an ideal channel, and after passing through the receiver's gear-up changing buffer the sequence $\{z_k\}$ was formed. The sequence $\{z_k\}$ had a symbol rate of 8 kHz, with every nth sample absent. The absentee samples rejected at the transmitter were then formulated according to eq. (6). The interpolation parameters a_i were found with the aid of eq. (11), which used correlation functions $R(k, j)$ related to the input speech sequence $\{x_k\}$. Thus, in our first experiment we were concerned with how successfully we could discard every nth sample in $\{x_k\}$, and replace the discarded samples using interpolation parameters based on $\{x_k\}$.

We used as a performance criterion segmental signal-to-noise ratio[12] (SEG-s/n), computed using the input sequence $\{x_k\}$, and the error sequence $\{e_k\}$ whose components are given by eq. (7). In our initial experiment the variation of SEG-s/n as a function of block size W was found using practical values of W extending from 64 to 1024, and $n = 4$. This value of n is a compromise between providing adequate SEG-s/n and a reasonable reduction in the number of transmitted samples. Applying the correlation function $R(k,j)$ in the determination of the interpolation parameters, we obtained the solid curves in Fig. 4. Curves a, b, and c apply for the case of $\lambda = 3$, 2, and 1, respectively. Curve d is the SEG-s/n when the interpolation was performed using the average of adjacent samples. Increasing λ to $n - 1$, i.e., to 3, resulted in an increase in SEG-s/n compared to lower values of λ, and for a block size of 256 and $\lambda = 3$, a gain of 13.7 dB in SEG-s/n was obtained compared to the simple interpolation averaging method of eq. (22). When the correlation function $R(\tau)$ of eq. (18) was employed to compute the interpolation parameters, curves e, f, and g were obtained corresponding to $\lambda = 3$, 2, and 1, respectively. The improvement in SEG-s/n derived from employing $R(k, j)$ rather than $R(\tau)$ increased with increasing λ, being 0.1, 1, and 3 dB for $\lambda = 1$, 2, and 3, respectively, and with $W = 256$.

As a means of providing further insight into the performance of the

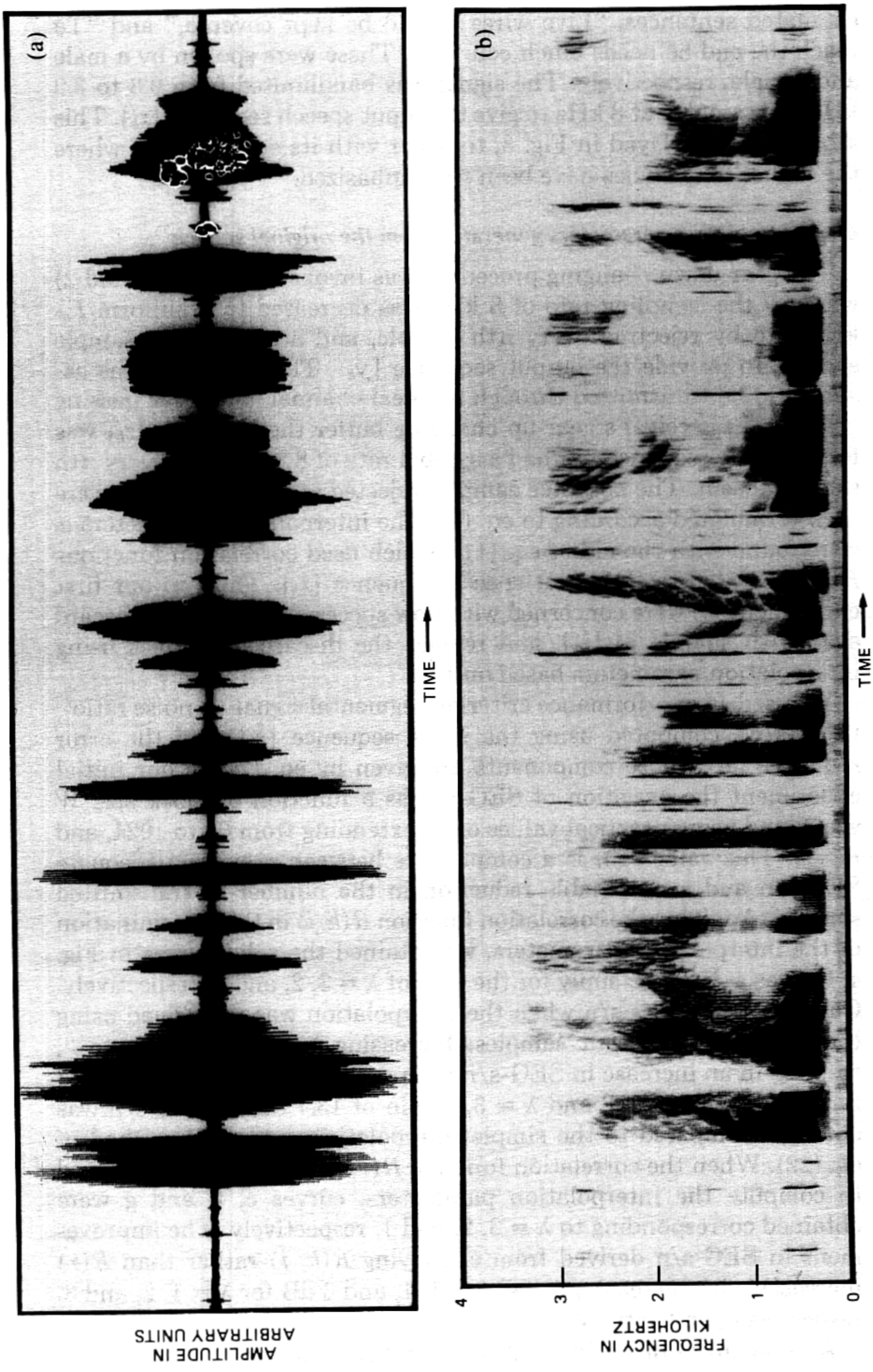


Fig. 3—(a) Input speech sequence $\{x_k\}$ and (b) its preemphasized spectrogram.

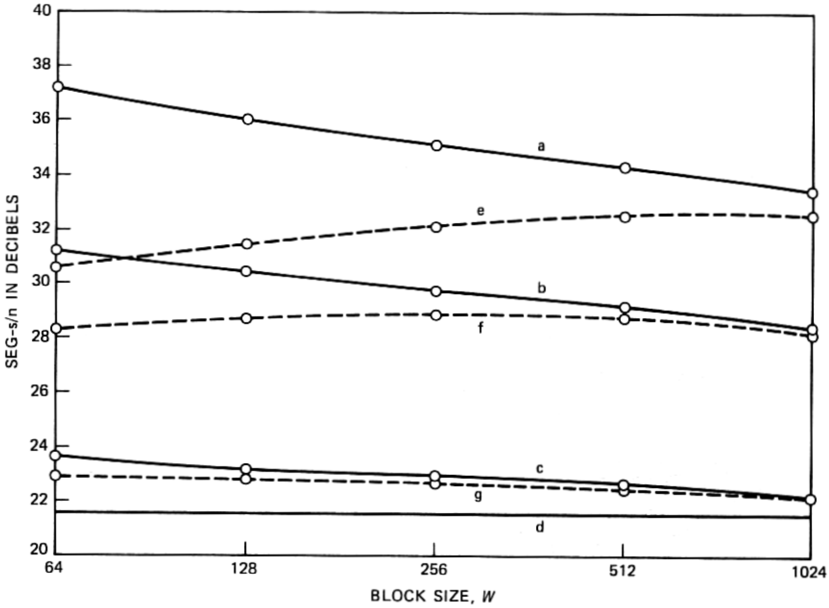


Fig. 4—SEG-s/n versus block size W. Curves a, b, and c apply for $\lambda = 3$, 2, and 1; and $\bar{R}(k, j)$ was used in calculating the interpolation parameters. Curve d relates to interpolation using nearest neighbor averaging. Curves e, f, and g apply for $\lambda = 3$, 2, and 1; and $R(\tau)$, given by eq. (18), was used in calculating the interpolation parameters.

interpolation system, we show in Fig. 5 the variation of the s/n of each block in the speech signal as a function of successive blocks for $W = 256$, $n = 4$. The average of these s/n values constitutes the SEG-s/n points in Fig. 4 for $W = 256$. As expected, we found that the s/n in every block was greater, if only by an infinitesimal amount, when more samples were used in the interpolation process, i.e., when larger values of λ were employed. Interpolation by adjacent sample averaging always provided the lowest s/n. In some blocks the advantage of using $\lambda = 3$ compared to $\lambda = 2$, $\lambda = 1$, and nearest neighbor averaging, provided s/n gains as large as 13, 31, and 33 dB, respectively.

Returning to Fig. 4, the SEG-s/n of over 35 dB, $\lambda = 3$, was found to be approximately 3 dB greater than the coventional s/n computed by measuring the mean square values of the components in $\{x_k\}$ and $\{e_k\}$ over the entire speech input signal. Our SEG-s/n measurements therefore indicate that the recovered speech is similar to toll quality speech.[2] Informal listening experiences for the recovered speech sequence $\{\hat{z}_k\}$ when $\lambda = 3$, $W = 256$, tended to confirm the SEG-s/n findings that the quality of the recovered speech sequence $\{\hat{z}_k\}$ was judged to be very similar to that of the original bandlimited sequence $\{x_k\}$. We observed that although the distortion in $\{\hat{z}_k\}$ for $\lambda = 1$ was

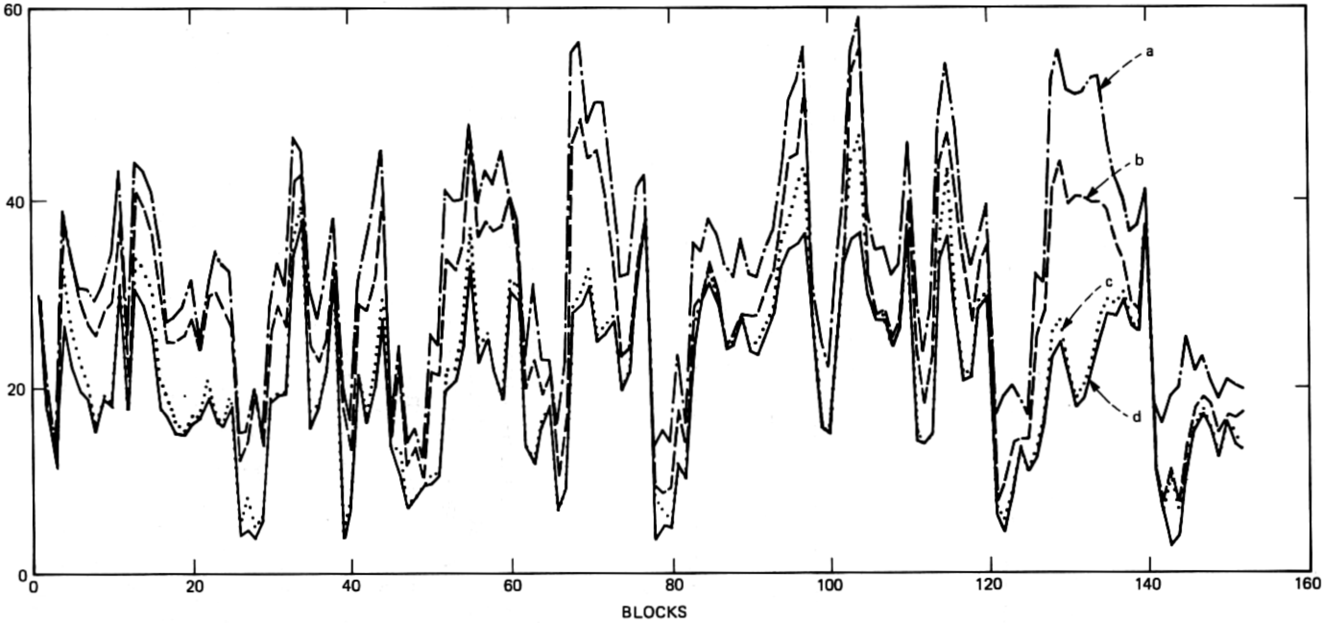


Fig. 5—Variation of block s/n as a function of block number, $W = 256$, $n = 4$. Interpolation procedure used (a) $\lambda = 3$, (b) $\lambda = 2$, (c) $\lambda = 1$, and (d) nearest neighbor averaging.

annoying, by making $\lambda = 2$ the interpolation noise was barely noticeable.

In Section III we present different expressions for the correlation function $R(\theta)$, namely eq. (14), where $\theta = k, j$, and eqs. (16) or (18) having $\theta = \tau$. The SEG-s/n and block s/n values shown in Figs. 4 and 5 were determined using $\theta = k, j$. We now demonstrate the loss in SEG-s/n due to using $\theta = \tau$ compared to when $\theta = k, j$ as a function of W, for the conditions of $n = 4$ and $\lambda = 3$. It will be recalled that $\theta = k, j$ enables the interpolation samples to be formulated that minimize E_r^2 over a block of W samples. When $\theta = \tau$ a low but not minimum value of E_r^2 is produced over the working range of W. The application of eq. (16) gives a more accurate measure of $R(\tau)$ than the simpler expression of eq. (18). Figure 6 shows the variation of SEG-s/n with block size when $R(\theta)$ is computed using eqs. (14), (16), and (18), $n = 4$, $\lambda = 3$. When $R(k, j)$ was used, the effect of increasing W was to decrease the SEG-s/n. This is to be expected because the interpolation parameters are fixed for a block, and we may think of a large block as composed of many smaller blocks, each with its optimum interpolation parameters. Thus, if one set of parameters is selected for the large block, these parameters are inevitably suboptimum for the smaller subblocks, and hence the SEG-s/n is lower for the larger

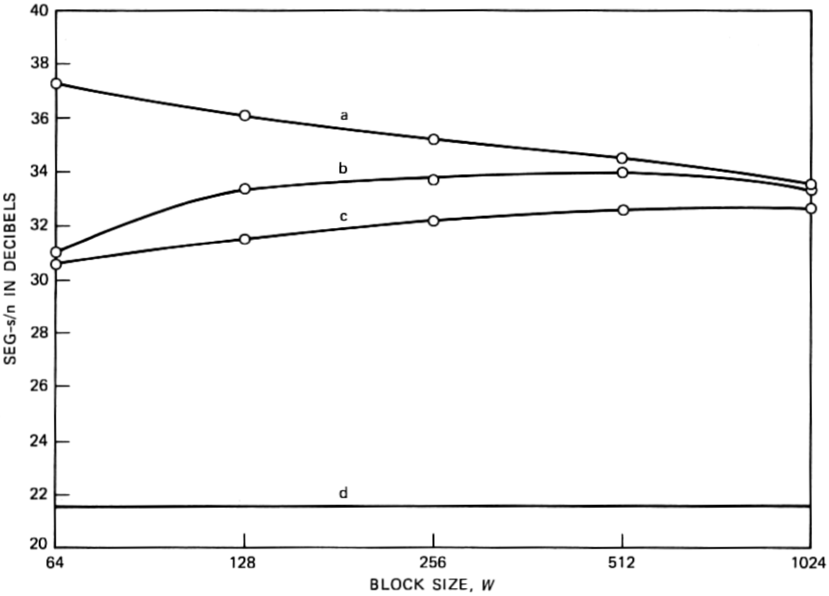


Fig. 6—Variation of SEG-s/n versus block size W for $\lambda = 3$, $n = 4$. The correlation function was computed for curves a, b, and c using eqs. (12), (16), and (18), respectively. Curve d applies to nearest neighbor averaging.

blocks. By contrast the SEG-s/n determined using $R(\tau)$ deteriorates with decreasing W. Now $R(\tau)$ is the conventional correlation function that assumes the speech signal to have stationary statistics. For the larger block sizes shown in Fig. 6, local statistical departures from stationarity tend to be smoothed by the $R(\tau)$ equations, but at low values of W a considerable number of interpolation errors occur in some blocks to yield a low SEG-s/n. When $W = 1024$ the SEG-s/n values computed using the different $R(\theta)$ expressions are very similar. We do not plot curves for $W < 64$ as the side information to transmit the interpolation parameters is unacceptably high [see eq. (20)], and for $W > 1024$ the delay is excessive (>250 ms). Thus, by using $R(k, j)$ we obtain higher and better interpolation performance compared with employing the conventional $R(\tau)$, but the computational complexity is greater.

6.2 Interpolation parameters generated from the received sequence

Having concluded that every fourth speech sample can be discarded and replaced by an interpolated sample to yield speech with negligible perceptual degradation, we next considered the performance of our scheme when the interpolation parameters were derived from the received data. The problem in this case was how to obtain a reliable estimate of the autocorrelation function $R(\tau)$. The procedures described in Section V for determining $R(\tau)$, and thence the interpolation parameters, were tried, and the variation of SEG-s/n with block size

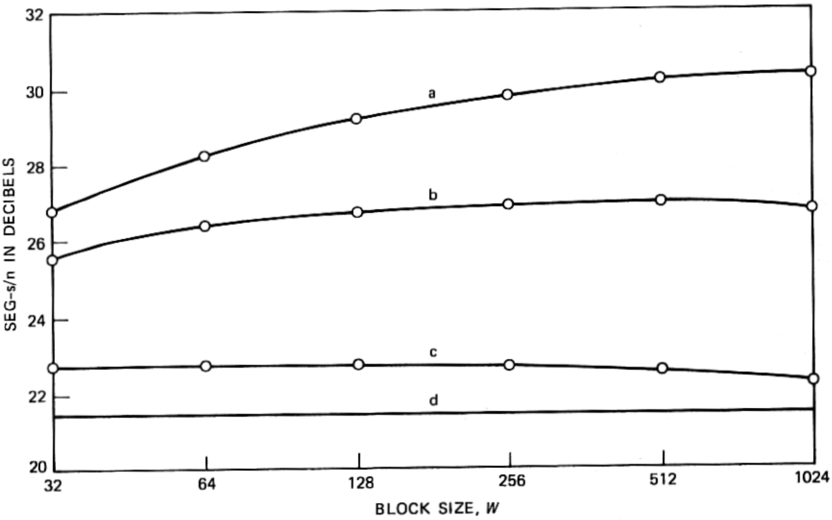


Fig. 7—SEG-s/n versus block size for interpolation parameters derived from the received sequence. Curves a, b, c, and d apply for λ-with-iteration, λ = 3; λ-interpolation, λ = 3; λ-interpolation, λ = 1; and average of adjacent sample interpolation, respectively.

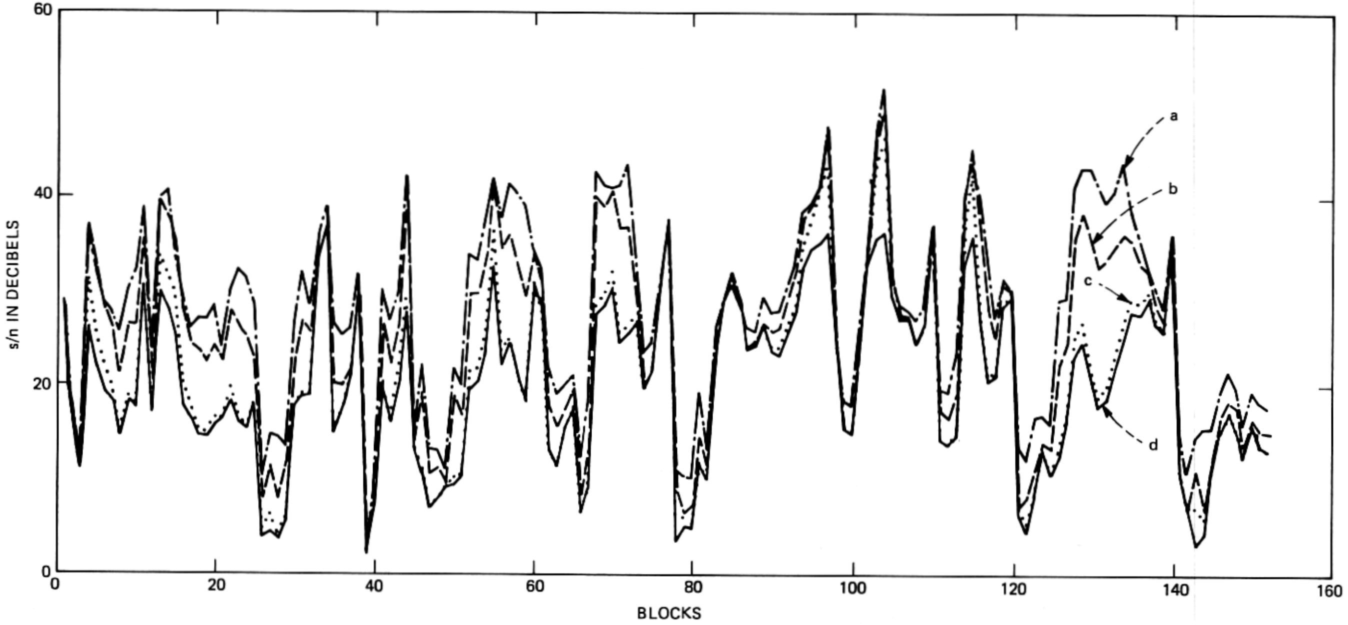


Fig. 8—Variation of block s/n versus block number for the iterative interpolation procedure, $W = 256$, $n = 4$. Curves a, b, c, and d apply for $\lambda = 3, 2, 1$, and nearest neighbor averaging, respectively.

W found for different values of λ. Curve d in Fig. 7 shows that a SEG-s/n of 21.5 dB was obtained when the interpolation process used nearest neighbor averaging, and was employed as a reference level. When the nearest neighbor interpolation was made adaptive, eq. (23) was used for which $\lambda = 1$, and curve c obtained. By using the iteration procedure where the interpolation was made according to eq. (25), $\lambda = 3$, a s/n > 30 dB was achieved for $N = 512$ (see curve a).

Also shown in Fig. 7 is curve b, which was obtained by formulating the sequence $\{\hat{z}_k\}$ based on first computing $\{\bar{z}_k\}$ according to eq. (22), formulating $R(\tau)$, and then removing samples $\bar{z}_r$ and replacing them with interpolated samples derived by using $\lambda = 3$. Thus, in this λ-interpolation method we do not progress from the average sequence to those derived with $\lambda = 1$ and 2, but proceed directly from the average sequence to compute the parameters with $\lambda = 3$. The result is a s/n of 27 dB for $W = 256$ to 1024, a 3-dB reduction compared to the λ-with-iteration ($\lambda = 3$) case, and a diminution in complexity. Informal listening experiences showed that the λ-interpolation and λ-with-iteration schemes, both with $\lambda = 3$, produced speech whose impairments were barely perceptible.

The variation of block s/n with block number for the iterative interpolation procedure is displayed in Fig. 8, $n = 4$, $W = 256$. The average values of these block s/n's give the segmental s/n in Fig. 7 for $W = 256$. Close inspection of the curves in Fig. 8 reveal that progressive iteration does not always give the highest block s/n. However, λ-with-iteration, $\lambda = 3$, achieved the highest s/n for most blocks with gains up to 25 dB compared to nearest neighbor averaging interpolation.

In Fig. 9, the variation of block s/n with block number is displayed for the $\lambda = 3$ condition, $n = 4$, $W = 256$. Curves a and b are those previously displayed in Figs. 5a and 8a, and refer to interpolation parameters computed from the original speech sequence, and by λ-with-iteration procedure, respectively. Curve c applies for the λ-interpolation method, while curve d corresponds to nearest neighbor averaging interpolation, and is included as a reference level. The curves in Fig. 9 illustrate that by deriving the interpolation parameters from the original speech instead of from the received data, the large dips in block s/n are mitigated.

6.2.1 Interpolation errors

We will now consider the interpolation error sequences and their spectra when the interpolation parameters are generated from the received sequence $\{z_k\}$. To illustrate the error performance we selected an arbitrary segment of our input speech signal (see Fig. 3) that had high-level and low-level voiced speech, and unvoiced speech. The speech segment and its spectrogram are displayed in Fig. 10. As before,

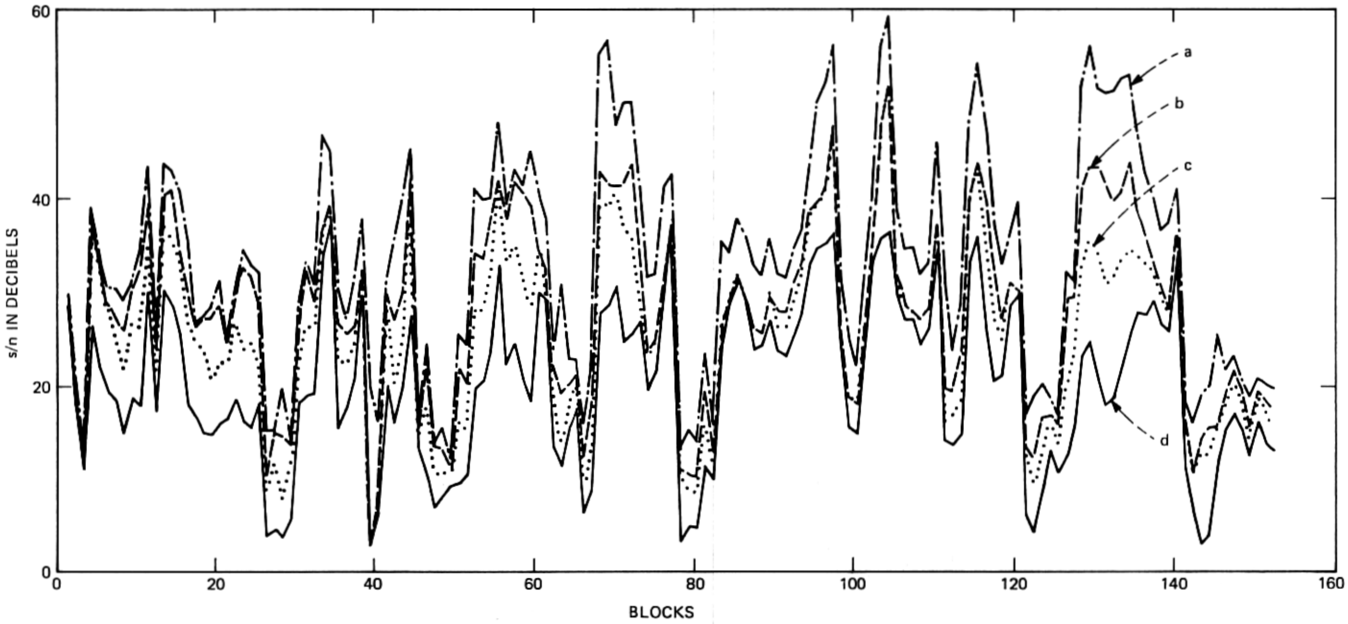


Fig. 9—Variation of block s/n as a function of block number for the $\lambda = 3$ condition, $W = 256$, $n = 4$. Curves a, b, c, and d are for interpolation parameters computed from original speech, λ-with-iteration, λ-interpolation, and nearest neighbor averaging, respectively.

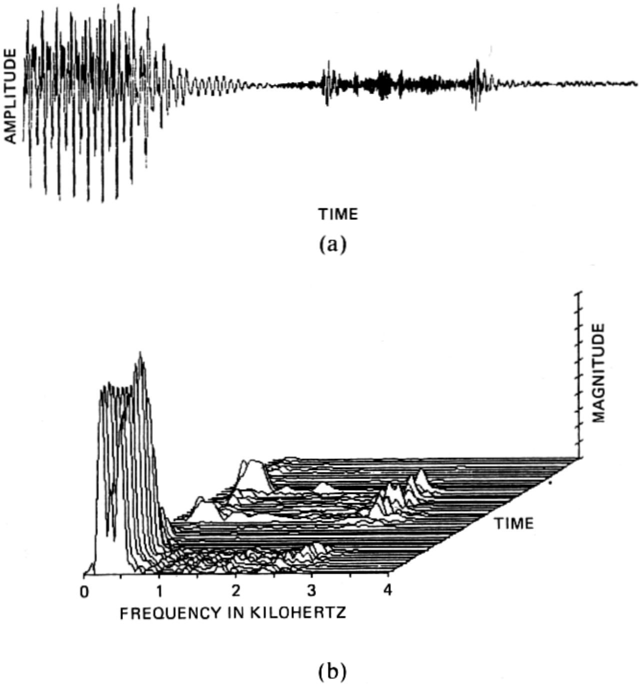


Fig. 10—Speech segment: (a) time waveform, and (b) spectrogram.

every fourth sample in the segment was removed and replaced by a sample produced by interpolation from neighboring samples, where the interpolation parameters were derived from the received data as described in Section V. We avoid displaying the spectrograms of the recovered speech segments associated with the four interpolation conditions used in Fig. 7 because of their similarity. Instead we show in Fig. 11a, b, c, and d the error signals determined as the difference between the input speech segment shown in Fig. 10a and the recovered waveforms produced using interpolation methods of: average of adjacent samples; λ-interpolation with $\lambda = 1$ and 3; λ-with-iteration, $\lambda = 3$; respectively. The error signals in Fig. 11 are small for the low-level highly correlated voiced speech section, and are much greater in the high-level voiced section and for the unvoiced speech. The block s/n values for the speech signal in Fig. 10a are therefore higher for the voiced speech than for the unvoiced speech. Inspection of Fig. 11 shows that the smallest error signal amplitudes generally occurred when the interpolation procedure used three samples on either side of each missing sample. For this segment of speech the advantage of using the full iteration procedure is small compared to $\lambda = 3$, no

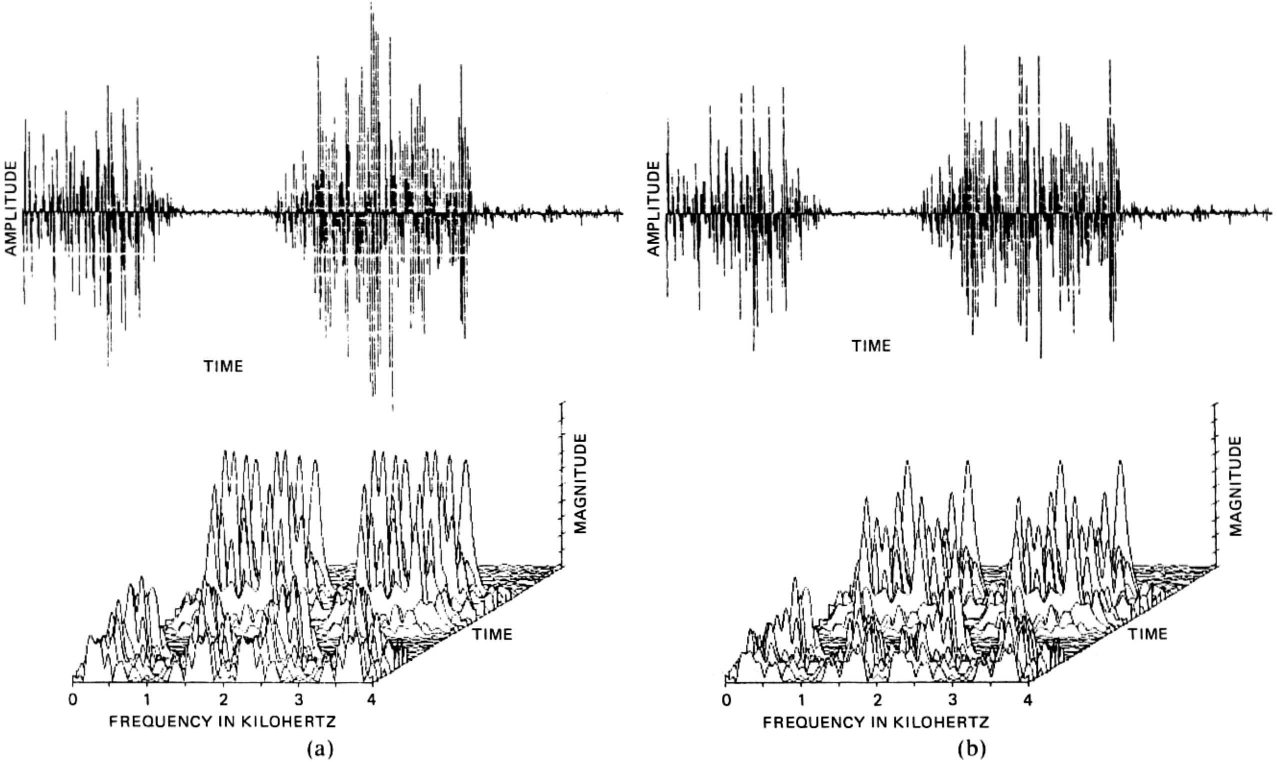
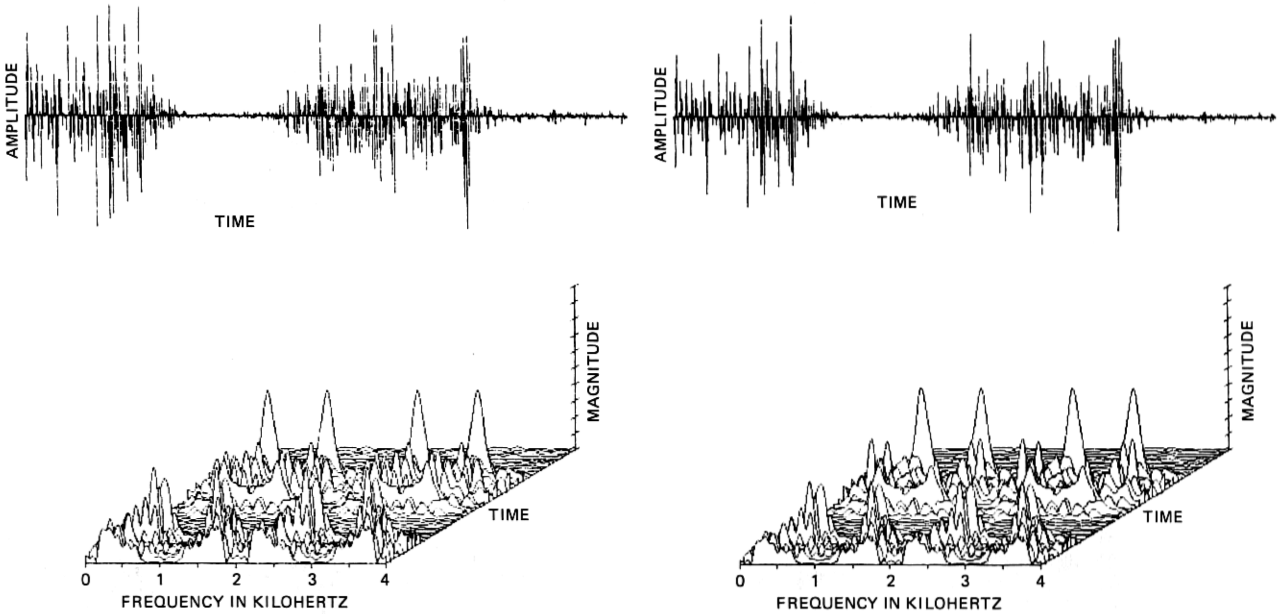


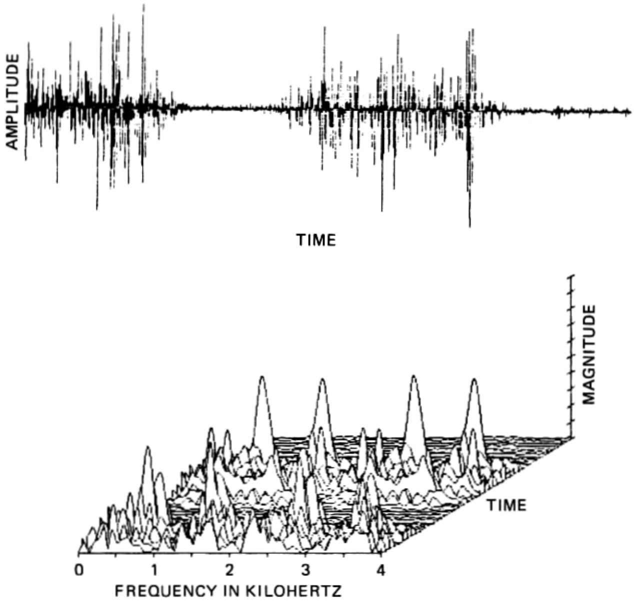
Fig. 11 (a) and (b)—Error waveforms and their spectrograms.
(Figure is continued on next page.)



(c)

(d)

Fig. 11 (c) and (d)—Error waveforms and their spectrograms.
(Figure is continued on next page.)



(e)

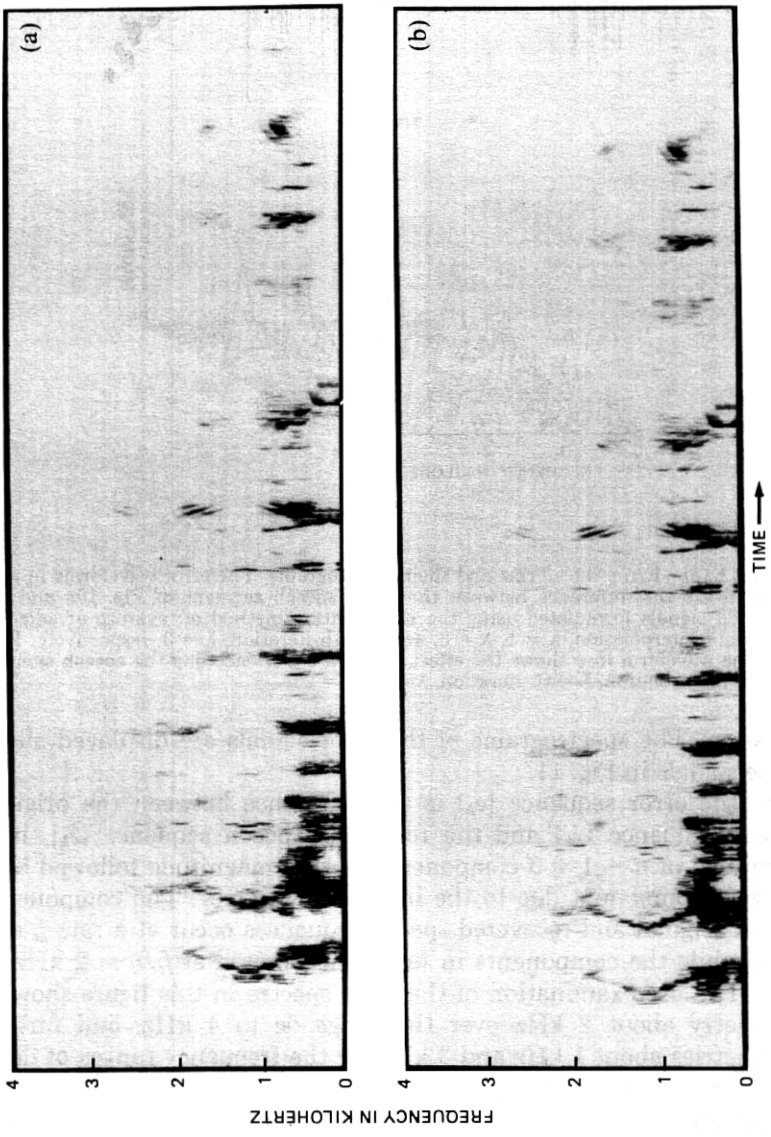
Fig. 11 (e)—Error waveforms and their spectrograms. The error waveforms in a, b, c, and d are the difference between the input speech segment of Fig. 10a and the recovered signals formulated using the interpolation method of: average of adjacent samples, λ-interpolation, $\lambda = 1$, $\lambda = 3$, and λ-with-iteration, $\lambda = 3$, respectively, $W = 256$. The waveform in e shows the effect of μ-law PCM encoding the speech samples prior to interpolation, λ-with-iteration, $\lambda = 3$, $W = 256$.

iteration. The spectrograms of the error signals are displayed above these signals in Fig. 11.

As the error sequence $\{e_k\}$ is the difference between the original speech sequence $\{x_k\}$ and the recovered speech sequence $\{\hat{z}_k\}$, it is composed of $n - 1 = 3$ components of zero magnitude followed by a nonzero component due to the interpolation error. The components in the original and recovered speech sequences occur at a rate $f_s = 8$ kHz, while the components in $\{e_k\}$ are generated at $f_s/n = 2$ kHz in Fig. 11. Close examination of the error spectra in this figure shows a symmetry about 2 kHz over the range dc to 4 kHz, and further symmetries about 1 kHz and 3 kHz for the frequency ranges of dc to 2 kHz, and 2 kHz to 4 kHz, respectively.

The noise components above 3.3 kHz were removed by the output filter shown in Fig. 1. When this was done the improvement in speech quality was minimal, as can be anticipated from the spectrograms of Fig. 11. Also the SEG-s/n values shown in Fig. 7 were only increased by a fraction of a decibel.

Figures 12a through d show the error spectrograms obtained when


(a)
(b)
TIME
FREQUENCY IN KILOHERTZ
0
1
2
3
4

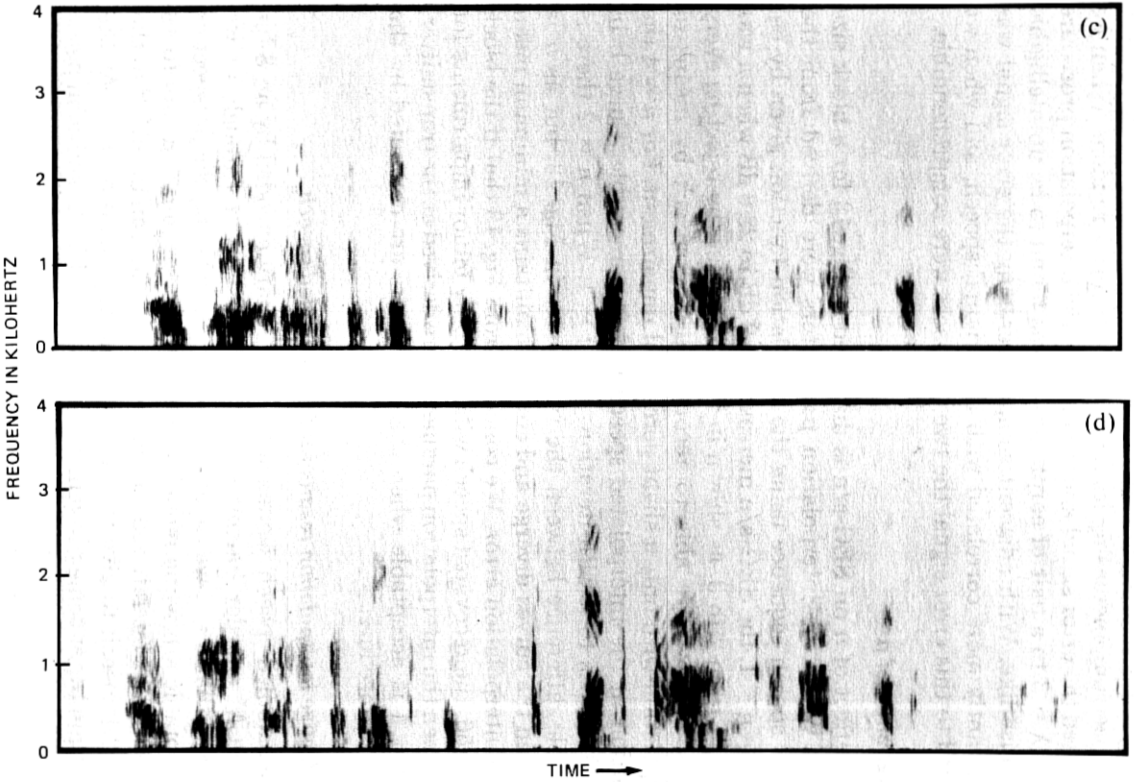


Fig. 12—Error spectograms for the two sentences shown in Fig. 3. The spectrograms a, b, c, and d relate to the conditions a, b, c, and d in Fig. 11, respectively.

the interpolation procedure used adjacent sample averaging, λ-interpolation with $\lambda = 1$, $\lambda = 3$, and λ-with-iteration, $\lambda = 3$ for the entire speech signal of Fig. 3. The spectrograms are not preemphasized. The effect of using $\lambda = 3$ compared to $\lambda = 1$ is to alter the nature of the noise spectrum, increasing its tendency to be more random. When we connected the error signal appertaining to the interpolation procedure having $\lambda = 3$ to a pair of earphones, it was found to be unintelligible and noise-like. With adjacent sample averaging the error signal was significantly more correlated with the original speech, and when we listened to this error signal the two sentences were comprehendible.

6.3 Variation of n

The effect of n on SEG-s/n is displayed in Fig. 13 for a block size of 256. When the interpolation parameters were derived from the original speech sequence using the correlation function given by eq. (14), $\lambda = n - 1$, the SEG-s/n increased by 9 dB to 28.6 dB when n was increased from 2 to 3, as shown in Fig. 13a. Thus, by rejecting every third sample we are able to reconstitute the speech by means of interpolation with only a slight perceptual impairment. For $n = 4$ the distortion in the interpolated speech was imperceptible. Curve b in Fig. 13 applies to λ-with-iteration, $\lambda = n - 1$. When $n = 2$ there is negligible difference between the two curves in Fig. 13, but as n is increased the curves diverge and curve a maintains a minimum mean square interpolation error. We conclude from Fig. 13 that if the block size is 256, the recovered speech will have only minor impairments for $n = 3$ when the interpolation parameters are derived at the transmitter, and $n = 4$ is acceptable when the parameters are computed by the iterative procedure.

6.4 Adaptive interpolation results for PCM encoded speech

The sampled speech sequence $\{x_k\}$ was binary encoded by an 8-bit μ-law PCM encoder having $\mu = 255$. From Figs. 4 through 13 we concluded that $n = 4$ was a good compromise, offering the prospect of an acceptable s/n while reducing the 64-kb/s transmission rate to 48 kb/s. With this bit-rate reduction, 16 kb/s of data can be added to the 48 kb/s of speech to give the conventional transmission rate. At the destination (or at some convenient point along the transmission path) the data can be removed, and each of the 6 k-words/s comprising the μ-law PCM signal decoded. In our experiments we assumed that the bits would be regenerated without error, and the 8-kHz sampling rate established by reinserting the missing samples by the interpolation techniques previously described. However, the accuracy of the interpolation process was reduced by the quantization noise produced in the digital encoder.

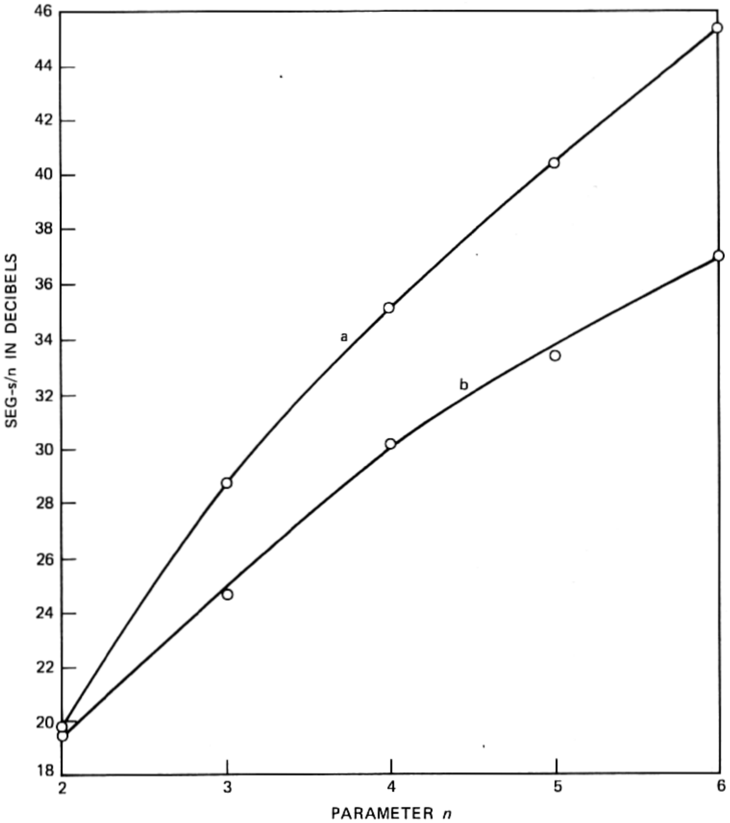


Fig. 13—Variation of SEG-s/n versus parameter n, for $\lambda = n - 1$, and the interpolation parameters derived from the original speech sequence, and by the λ-with-iteration scheme at the receiver. The block size is 256.

Deriving the interpolation parameters from the received data, as described in Section V, yielded the results shown in Fig. 14. The interpolation procedures applicable to curves a, b, c, and d were identical to those employed for curves a, b, c, and d, respectively, in Fig. 7. The respective curves in Fig. 14 are lower than those in Fig. 7 owing to the effect of quantization noise encountered in μ-law PCM encoding. We observe that the quantization noise causes a loss in SEG-s/n of 0.5 dB when the interpolation is performed by adjacent sample averaging. For the cases of $\lambda = 1$ and $\lambda = 3$ the losses in SEG-s/n owing to the effect of quantization noise become approximately 1 and 1.7 dB, respectively, when $W = 256$. As the interpolation process improves, the SEG-s/n becomes relatively more affected by the quantization noise. We observe that the greatest loss in SEG-s/n relative to when there is no quantization noise occurs for the case of λ-with-

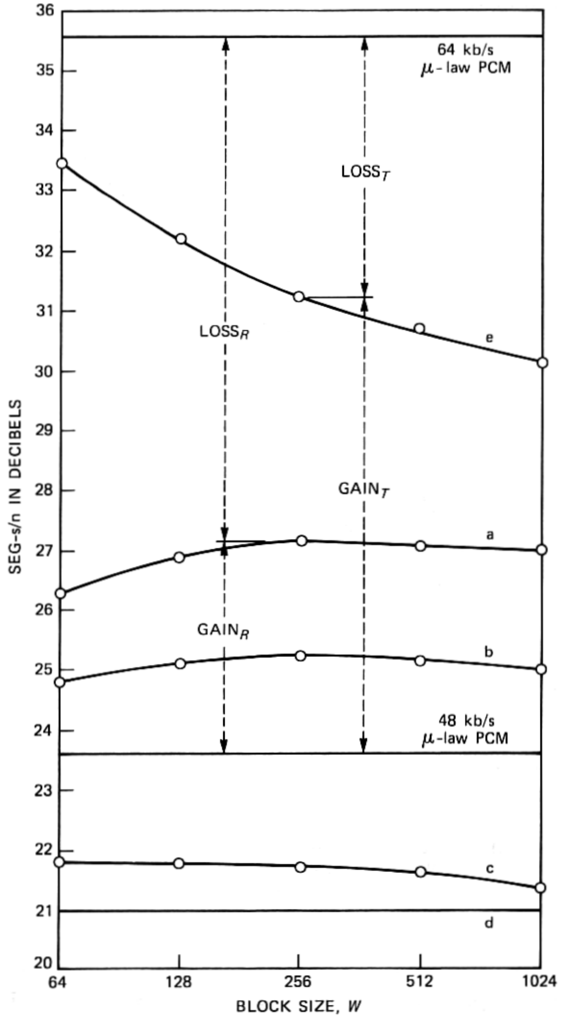


Fig. 14—8-bit μ-law PCM encoding, $\mu = 255$. The effect of block size on SEG-s/n for different conditions. Curves a, b, c, and d apply for λ-with-iteration, $\lambda = 3$; λ-interpolation, $\lambda = 3$; λ-interpolation, $\lambda = 1$; and average of adjacent sample interpolation, respectively. Curve e relates to the interpolation parameters derived at the transmitter using $\lambda = 3$ and $R(k, j)$.

iteration, $\lambda = 3$, and is 2.7 dB for $W = 256$. Also shown in Fig. 14 are the SEG-s/n values for the encoded speech when the sampling rate is 8 kHz, $\mu = 255$, and the number of bits per code word is 8 and 6, i.e., the transmission bit rates are 64 kb/s and 48 kb/s, respectively. The loss in SEG-s/n due to transmitting at only 48 kb/s and reinserting the discarded samples using λ-with-iteration, $\lambda = 3$, compared to not discarding samples and transmitting at 64 kb/s is given by $LOSS_R$ in

Fig. 14. The subscript R indicates that the interpolation parameters were derived from the received data. The gain in SEG-s/n, $GAIN_R$, is due to interpolating the missing samples compared to the 48 kb/s μ-law PCM encoding where no samples are discarded. Although the 48 kb/s has the 8-kHz word rate, the quantization noise is higher because there are 6 bits per code word. The values of $LOSS_R$ and $GAIN_R$ for $W = 256$ are 8.4 and 3.6 dB, respectively.

The interpolation performance can be enhanced if the interpolation parameters are derived from the locally decoded μ-law PCM signal at the transmitter. The interpolation parameters must be conveyed to the receiver in a binary format as side information. This is not a difficult task in a digital transmission system as fewer than 30 bits per W PCM words are required to be transmitted. When W is 256, the side information increases the bit rate by 2 percent. The bit rate can be maintained at 48 kb/s if the side information replaces the least significant bit in every sixth word transmitted. This will have only a marginal effect on the quality of the recovered speech. When the interpolation parameters for $\lambda = 3$ were determined at the transmitter using eqs. (11) through (14) and subsequently transmitted as side information, curve e in Fig. 14 was obtained. Comparing curve a in Fig. 6 with this curve shows that the presence of quantization noise reduces the SEG-s/n of the interpolated speech signal by 4 dB. Nevertheless, curve e is significantly higher than curve a, and the disparity increases to 7 dB for $W = 64$. The $LOSS_T$ and $GAIN_T$ factors, where the subscript T implies the generation of the parameters at the transmitter, had values of 4.3 and 7.7 dB, respectively, for $W = 256$. The effect of interpolation is equivalent to saving more than one bit per word.

For μ-law PCM encoding and the adaptive interpolation procedure of λ-with-iteration, $\lambda = 3$, $W = 256$, resulted in the error waveform and its spectrogram displayed in Fig. 11e for the speech segments shown in Fig. 10. Figures 11e and d show that the effect of quantization on the error spectrum is small.

VII. DISCUSSION

Our intention at the outset of this investigation was to discard one speech sample (or PCM word) in every four, $n = 4$, and to replace the missing samples or words by an interpolation process such that the degradation in speech quality was virtually imperceptible. Further, the implementation algorithm was to be inherently simple. These goals have been reached in good measure.

The central issue in any interpolation process is determining the interpolation parameters. Our approach is to attempt to minimize the mean square error, a nonoptimum procedure for speech signals where the perception of interpolation noise may be modified by the spectral composition of the speech signal over some 20 ms interval, and temporal effects lasting approximately 200 ms. The justification of the mean square error is based on simplicity, and for $n = 4$ gives good results. In deriving the interpolation parameters of eq. (11) based on $R(k, j)$, we made no assumptions concerning the statistic of the speech signal. The selection of block size W depends upon an acceptable s/n, the need to avoid excessive signal delays resulting from too high a value of W, and the amount of side information permitted, where appropriate, when W is small. Our suggested range of W is from 64 to 1024 (see Figs. 4 and 7). These values of W correspond to durations of 8 to 128 ms, i.e., ranging from approximately a pitch to a syllable period. Computing the interpolation parameters using eqs. (11) and (14), we were able to achieve gains in s/n of 16, 14, and 12 dB compared to interpolation using nearest neighbor linear interpolation for block sizes of 64, 256, and 1024, respectively.

When the estimate of the autocorrelation function could not be based on the original speech sequence, but had to be estimated from the received speech samples where every nth sample was missing, an iterative estimation procedure was employed. By making a crude estimation of the missing samples, the autocorrelation function $R(\tau)$ was computed, and in general a more accurate set of interpolated samples were found. The autocorrelation function was again determined, and the accuracy of the interpolated samples nearly always improved. By this iterative approach, for $n = 4$, λ-with-iteration, $\lambda = 3$ had an interpolation gain over nearest neighbor averaging of 8 dB for $W = 256$, as displayed in Fig. 7.

The effect of discarding every nth sample, $n = 2, 3, 4$, and 5, showed that it is advisable to maintain $n \geq 4$ for imperceptible perceptual degradation. For highly correlated sounds and where some masking of the interpolation noise occurs, we can satisfactorily deploy $n = 3$ and even $n = 2$. However, in general there is considerable distortion power when $n = 2$, which is hardly surprising as half the samples had been rejected. Nevertheless, for $n = 2$ adaptive interpolation yielded a s/n of approximately 19.5 dB, which is noisy but intelligible speech.

Finally, we found that when conventional 8-bit μ-law PCM encoded speech, $\mu = 255$, had its bit rate reduced from 64 kb/s to 48 kb/s to enable 16 kb/s of other data to be transmitted, the recovered speech after decoding and interpolation had a s/n that approximated that of 56-kb/s μ-law PCM.

VIII. ACKNOWLEDGMENTS

The authors thank D. Vitello for her advice with computer simulation, and D. J. Goodman for his constructive criticism of this work.

REFERENCES

1. B. G. Haskell and R. Steele, "Audio and Video Bit-Rate Reduction," Proc. IEEE, *69,* No. 2 (February 1981), pp. 252–62.
2. J. L. Flanagan, M. R. Schroeder, B. S. Atal, R. E. Crochiere, N. S. Jayant, and J. M. Tribolet, "Speech Coding," IEEE Trans. Commun., *COM-27* (April 1979), pp. 710–37.
3. M. V. Mathews, "Extremal Coding for Speech Transmission," IRE Trans. Inform. Theory, *IT-5* (September 1959), pp. 129–36.
4. C. M. Kortman, "Redundancy Reduction—A Practical Method of Data Compression," Proc. IEEE, *55,* No. 3 (March 1967), pp. 253–63.
5. L. Ehrman, "Analysis of Some Redundancy Removal Bandwidth Compression Techniques," Proc. IEEE, *55,* No. 3 (March 1967), pp. 275–87.
6. C. A. Andrews, J. M. Davis, and G. R. Schwarz, "Adaptive Data Compression," Proc. IEEE, *55,* No. 3 (March 1967), pp. 267–77.
7. N. S. Jayant, "Adaptive Aperture Coding for Speech Waveforms-1," B.S.T.J., *58,* No. 7 (September 1979), pp. 1631–45.
8. H. S. Hou and H. C. Andrews, "Cubic Splines for Image Interpolation and Digital Filtering," IEEE Trans. ASSP, *ASSP-26,* No. 6 (December 1978), pp. 508–17.
9. "Special Issue on Bit-Rate Reduction and Speech Interpolation," IEEE Trans. Commun., *COM-30,* No. 4 (April 1982), pp. 728–80.
10. N. S. Jayant, "Effect of Packet Losses in Waveform Coded Speech and Improvements Due to an Odd-Even Sample-Interpolation Procedure," IEEE Trans. Commun., *COM-29,* No. 2 (February 1981), pp. 101–9.
11. R. E. Crochiere and L. R. Rabiner, "Interpolation and Decimation of Digital Signals—A Tutorial Review," Proc. IEEE, *69,* No. 3 (March 1981), pp. 300–31.
12. P. Noll, "Adaptive Quantization in Speech Coding Systems," Int. Zurich Seminar, Zurich, Switzerland, April 1974.

AUTHORS

Frank Benjamin, B.A. (Music Education), 1980, B.S. (Electronic Engineering), 1983, Valedictorian (both cum laude), Monmouth College, West Long Branch, NJ; Monmouth College, 1980-1981; United Telecontrol Electronics, 1981; Bell Laboratories, 1981—. While Mr. Benjamin was attending college, he worked as a recording engineer and studio musician on albums for RCA-Columbia and EPIC studios, and on promotional sound tracks for the motion picture *STAR TREK.*® He taught privately in the fields of music, engineering, mathematics and physics. At Monmouth College he worked as a laboratory instructor and acoustical consultant. While at United Telecontrol Electronics, he helped develop a missile guidance control system. In 1981 he joined Bell Laboratories as a member of the Radio Communications Research Department, designing and writing software simulation systems for speech compression and interpolation techniques. Past President, Lambda Sigma Tau; member, Eta Kappa Nu; nominee, Who's Who Among Students in American Universities and Colleges; New Jersey State Teacher's Certificate.

Raymond Steele, (SM '80), B.S. (Electrical Engineering) from Durham University, Durham, England, in 1959, and the Ph.D. degree in 1975. Prior to his enrollment at Durham University, he was an indentured apprenticed Radio Engineer. After research and development posts at E. K. Cole Ltd., Cossor Radar and Electronics, Ltd., and The Marconi Company, all in Essex, England, he joined the lecturing staff at the Royal Naval College, Greenwich, London, England. Here he lectured in telecommunications to NATO and the External London University degree courses. His next post was as Senior Lecturer in the Electronic and Electrical Engineering Department of Loughborough University, Loughborough, Leics., England, where he directed a research group in digital encoding of speech and television signals. In 1975 his book, *Delta*

Modulation Systems (New York: Halsted), was published. He was a consultant to the Acoustics Research Department at Bell Laboratories in the summers of 1975, 1977, and 1978, and in 1979 he joined the company's Communications Methods Research Department, Crawford Hill Laboratory, Holmdel, NJ. In 1981 Mr. Steele was given *The Bell System Technical Journal* Best Paper Award in the category of Mathematics, Communication Techniques, Computing and Software, and Social Sciences. In 1983 he joined The University of Southampton, Southampton, England, as a Professor of Communications.