THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 60, No. 5, May–June 1981
Printed in U.S.A.

Variable Bandwidth Adaptive Delta Modulation

By J. O. SMITH* and J. B. ALLEN

(Manuscript received December 3, 1980)

The ADM(adaptive delta modulation) speech coder is generally used with a time-invariant low-pass filter at the decoder output. The purpose of this low-pass filter is to reject coder noise at frequencies above the fixed speech passband. The speech spectrum, however, tends to occupy only the lower frequencies within the passband during voiced speech, and is somewhat "high pass" during unvoiced speech. In this paper, we show how the quality of ADM may be significantly improved by adaptively filtering the coder output such as to follow the natural bandwidth of the speech. This was found to reduce drastically the perceived noise in the output of the ADM coding system at low bit rates. The use of an adaptive low-pass filter realizes almost all of this quality gain. (An adaptive high-pass filter seems to reject less audible noise components and seems more prone to introducing objectionable artifacts.) We also discuss a method for reducing the bit rate with little or no sacrifice in quality (relative to normal ADM) by adapting the sampling rate along with the time-varying low-pass filter.

I. INTRODUCTION

In this paper, we explore two methods for better utilizing the time-varying bandwidth of speech in ADM (adaptive delta modulation) coders. In the first method, the ADM speech quality is shown to be improved by filtering the reconstructed (decoded) speech with a time-varying filter tailored to the natural speech bandwidth. In this case, adaptive bandpass filtering of the ADM output signal reduces coder noise by rejecting noise components at frequencies outside of the principal speech spectrum. Experimentally, we found that eliminating the upper 2 percent of the spectrum energy gave a reduction in average

* Presently a graduate student at the Information Systems Laboratory, Department of Electrical Engineering, Stanford University.

bandwidth on the order of a factor of two relative to an initial 3-kHz bandwidth for typical speech samples. If the coder noise is white, there is an average noise power reduction by a factor proportional to the bandwidth reduction. Furthermore, the remaining portion of the noise power lies entirely within the band of the speech so that for reasonably good signal-to-noise ratios, some masking of the noise by the speech can be expected.

The second case we explore is one of an adaptive sampling rate. In this case, the noise is again eliminated outside the principal speech bandwidth with a time-dependent low-pass filter. Then the average bit rate is reduced by a time-dependent decimation.

1.1 Block diagram

Figure 1 shows a block diagram of the system implemented in software for the tests to be presented. The system includes estimation of the short-time bandwidth (discussed in Section II), time-dependent filtering to this bandwidth, sampling rate conversion via decimation/interpolation,[1] and a 1-bit memory ADM coder with exponential step-size adaption.[2] For discursive purposes, we regard each of the two bandpass filters as a cascade of independently controlled low-pass and high-pass filters. The details on the implementation of this system are given in Appendix A.

1.2 Test cases studied

For clarity we define names for the following four cases studied:

Normal ADM (ADM)—Both bandpass filters in Fig. 1 are fixed at the full voice-channel bandwidth, and the sampling rate is fixed. For example, 24 kbps ADM is implemented with a constant sampling rate and both filters are set to pass frequencies from 200 to 3200 Hz at all times (see Ref. 2).

Post-filtered (ADM-PF)—Only the receiver reconstruction filter (at the far right in Fig. 1) varies to match the speech spectrum. The transmitter input filter is fixed at the channel bandwidth, and the sampling rate is fixed.

Pre- and Post-filtered (ADM-PPF)—Both the input and output filters are made to track the speech bandwidth, but the sampling rate remains fixed as in ADM-PF. The addition of adaptive prefiltering allows the ADM coder to track the speech waveform with less error.[3]

Adaptive Rate (ADM-AR)—The sampling rate varies at twice the upper cutoff frequency, and both the input and output filters track the speech bandwidth.

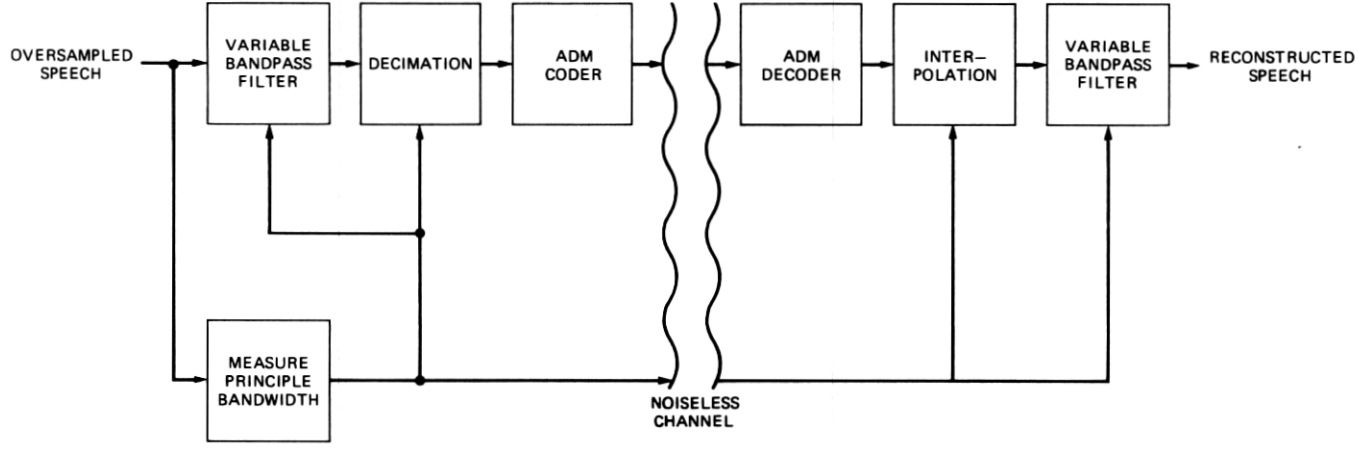

Fig. 1—Block diagram of the variable-rate ADM coding system with adaptive bandpass filtering. The speech bandwidth is measured in real time to control the variable bandpass filters and ampling-rate conversion (decimation/interpolation). In the simplest case, only the right-most filter is varied along with the speech spectrum to provide coder noise suppression. In the case of adaptive sampling rate, both filters and the sampling rate are tailored to the speech.

1.3 Results

The main conclusions are:

(*i*) Post-filtering gives a significant increase in quality. For example, 16 kbps ADM-PF gives a quality commensurate with 24 kbps ADM without post-filtering. The signal to noise ratio (s/n) is increased primarily in the low bandwidth segments such as back vowels, nasals, and voiced stops.[4] Almost all of the improvement arises from the adaptive low-pass component of the filtering. The adaptive high-pass filter contributes only slight noise suppression, and can introduce undesired side effects; for example, when there is a rapid transition from voiced to unvoiced, in which the speech band goes from low pass to high pass, an audible and objectionable change in the coder noise can occur even though there is an improved s/n due to the rejection of out-of-band low-frequency coder noise. Thus, adaptive low-pass filtering improves quality without serious side effects, while adaptive high-pass filtering only slightly improves s/n and causes significantly more audible noise modulation. For ADM, these effects are most pronounced at 24 kbps and below.

(*ii*) For the case of ADM-AR (prefiltering, adaptive sampling rate, and post-filtering) we found that adapting the sampling rate causes low-pass time frames (such as voiced segments) to have a degraded s/n compared to the ADM s/n; however, for these frames, the bit rate is substantially reduced. Furthermore, while the in-band coder noise is increased, the out-of-band coder noise is eliminated. Consequently, the quality of ADM-AR is different from ADM but not easily judged to be worse. Informal listening tests indicated no reliable preference for one over the other for the few samples of speech tested (base bit rates of 16, 24, and 32 kbps).

Summarizing positive practical results, our simulations indicate that time-dependent (adaptive) low-pass post-filtering yields a significant quality increase (for bit rates of 24 kbps and below) and adaptive low-pass pre- and post-filtering plus adaptive sampling rate yields reduced average bit rate with little change in quality.

In Section II, we discuss how the time-dependent filter cutoff frequencies are measured from the short-time speech spectrum. Section III presents simulation results for the four cases defined above.

II. MEASUREMENT OF THE TIME-VARYING SPEECH BANDWIDTH

Given the short-time spectrum of the speech at a given time, we wish to define the upper and lower cutoff frequencies of the spectrum in a way that minimizes bandwidth without introducing significant quality loss in the bandlimited speech. For this purpose, we define two constants T_L and T_U which may be thought of as the fractional energy

of the speech bandwidth to be removed.[5] T_L and T_U are taken to lie between 0 and 1. We call T_L the lower cutoff threshold and T_U the upper cutoff threshold. If $X(t, f)$ denotes the short-time spectrum of the speech at time t, then the time-varying high-pass and low-pass cutoff frequencies are found by solving

$$T_U = \frac{1}{E(t)} \int_{f_U(t)}^{\infty} |X(t, f)|^2 \, df,$$

$$T_L = \frac{1}{E(t)} \int_{0}^{f_L(t)} |X(t, f)|^2 \, df, \tag{1}$$

for $f_U(t)$ and $f_L(t)$, where $E(t)$ is the total spectrum energy at time t given by

$$E(t) = \int_{0}^{\infty} |X(t, f)|^2 \, df. \tag{2}$$

Note that $f_U(t)$, the high-frequency (or low-pass) cutoff, and $f_L(t)$, the low-frequency cutoff, vary to maintain constant T_U, T_L.

The discrete-time, discrete-frequency definitions that result from using the discrete Fourier transform (DFT) to generate short-time spectra are exactly analogous. When discussing sampled data, we write n in place of t, and the sampling rate will be denoted by $f_s = 1/T$.

The upper cutoff threshold T_U is the fixed fraction of the total energy that is rejected by the time-varying low-pass filter, and similarly, T_L controls the time-varying high-pass filter. These constants are chosen in accord with desired coder quality. Ideally, the values of T_L, T_U might be optimized to trade off bandwidth for suppressed coder noise. In the spirit of Wiener filter theory, we might define the optimum thresholds as the values for which a decrease of either results in more added coder noise than added signal in the reconstructed speech, and where an increase of either value causes more distortion loss due to bandlimiting than quality gain from noise excision. However, we do not know how to define objective measures of subjective degradations due to changes in bandwidth and coder noise. In our tests, the thresholds T_L and T_U were set such that they did not appreciably degrade the speech quality in the absence of coder noise. That is, rather than attempt to define optimal thresholds for each bit rate, we wish merely to estimate the benefits of variable bandwidth when no perceptually significant distortion results from the bandlimiting alone. Accordingly, in all ADM coder simulations, where the initial speech bandwidth is 0.2 to 3.2 kHz, the values $T_L = 2$ percent and $T_U = 1$ percent were used to specify the time-varying filters (and sampling rate when applicable).

An example of the passband behavior for these threshold values is

given in Fig. 2. The phrase analyzed was from an adult male speaker. Note that within the telephone passband, the speech is basically either low pass or high pass at any given time. It is these temporal speech bandwidth variations that we exploit for noise and sampling rate reduction in ADM.

Figure 3a gives a spectrogram of the same speech sample, and Fig. 3b shows the spectrogram after filtering the speech to the bandlimits shown in Fig. 2. We see that the 1 percent energy upper cutoff limit tends to follow the third formant during voiced regions, and the 2 percent lower cutoff limit has an almost unobservable effect. (The 2 percent high pass has an audible effect on unvoiced phonemes, however.) Figure 4a shows the output of a 16-kbps normal ADM coding system (time-invariant filters), and Fig. 4b shows the effect of post-filtering. The audible improvement due to post-filtering is much like Fig. 4b suggests, namely the out-of-band noise has been removed in Fig. 4b. This particular sample of post-filtered 16-kbps coded speech sounds about as good as when coded with normal 24-kbps ADM.

Figure 5 gives a plot of the *average* bandlimits (averaged over the entire utterance) as a function of thresholds. When $T_U = T_L = 0$, the passband is identical to the original speech passband; as the thresholds approach one half, the passband converges to zero. Comparing the two traces, we see that the speech is primarily low pass, which correlates with the fact that the utterance is predominantly voiced. Note that Fig. 5 implies an average bandwidth of only one half the maximum bandwidth using the values $T_U = T_L = 1$ percent. In other words, it is possible to reduce the average sampling rate by a factor of two while sacrificing only 2 percent of the spectral energy.

A few remarks are in order concerning practical issues associated with the measurement of the time-varying speech bandwidth. When tracking any spectrum over time, it is necessary to employ the proper balance of frequency resolution versus time resolution in the spectral analysis.[4] For speech, we wish to track bandwidth changes correspond-

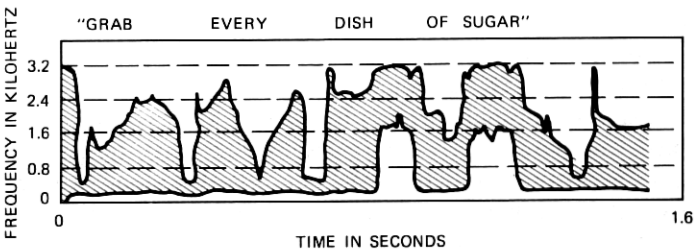

Fig. 2—Spectral band edges vs time for a male speaker, obtained by rejecting the upper 1 percent and the lower 2 percent of the 200–3200 Hz spectrum energy. Band-edge values are computed every 12.5 ms.

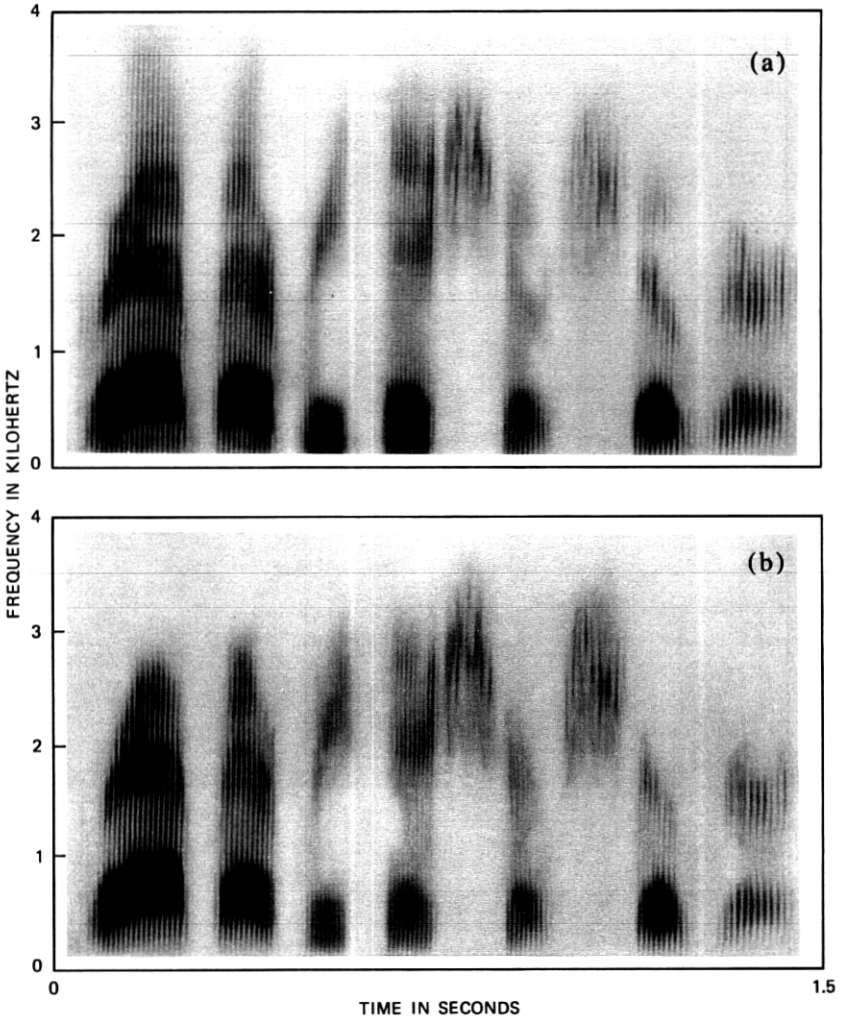

Fig. 3—Spectrograms of 8 kHz sampled speech, before and after filtering to the measured speech bandwidth. (a) Original speech spectrum within the fixed channel bandwidth of 200–3200 Hz. (b) Same speech after filtering with a time-varying bandpass, which rejects the upper 1 percent and lower 2 percent of the spectrum energy in the band. This filtering is approximately transparent.

ing to the articulation of phonemes. Tracking should be rapid so that there is little or no "smearing" of the estimated band-edges at the juncture of two dissimilar phonemes, and this implies using a small integration frame (window) in computing the short-time spectrum. However, when the spectrum is based on a frame length that is less than a pitch period, we obtain spectra that fluctuate excessively (at the pitch frequency) due to differing decay times of the vocal tract

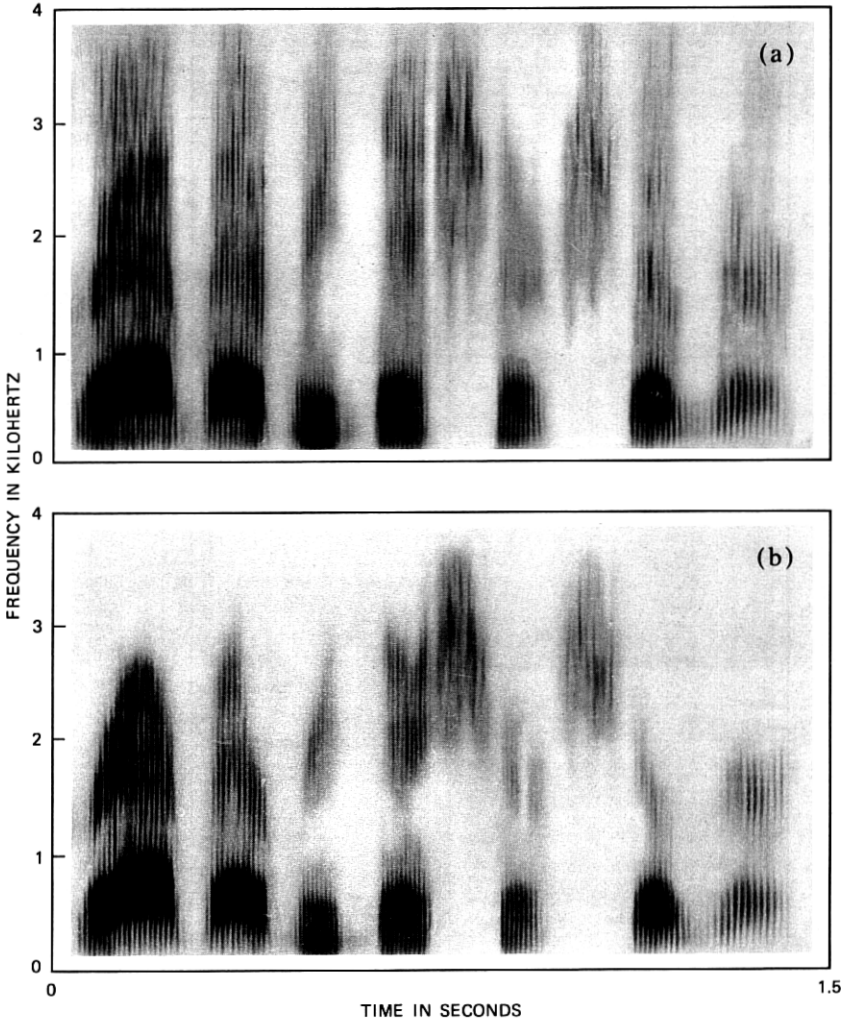

Fig. 4—Spectrograms of the same speech as in Fig. 3 at the output of a 16 kbps ADM system before and after time-varying filtering. (a) Fixed-bandwidth coder output. (b) Variable-bandwidth coder output. The time-varying filter is controlled by the same band edges as in Fig. 3b, i.e., the band edges are measured from the speech at the coder input. The effect of this filtering is to reduce significantly the ADM coder noise.

impulse response components. For this reason, it is desirable to include at least 20 ms of speech in each spectrum computation, corresponding to the observation that the pitch of voiced speech rarely, if ever, falls below 50 Hz. As Fig. 6a shows, when the spectrum analysis integration time is less than a pitch period, the time-varying low pass cutoff $f_U(n)$ can oscillate quite significantly (e.g., ±20 percent) at the pitch frequency. In Figs. 6b,c we show the effects of a seven-point median

smoother and a seven-point moving average smoother on the data of Fig. 6a.

Another implementation issue is that of using the time-varying cutoff frequencies to control the output filters. Care must be taken to match the spectral analysis integration time, sampling interval for the cutoff frequencies, and filter impulse-response duration. These three times should be comparable in magnitude. In Ref. 6, it is shown that for the case of FFT-based (fast Fourier transform) analysis and filtering, the minimum sampling rate for the filter cutoff frequencies is determined by the window used on the input to the FFT. For a length N FFT with a Hamming window, the band-edges may be sampled every $N/4$ samples (i.e., successive FFTs used for calculating f_U, f_L may be offset in time by $N/4$ samples). It is shown in Ref. 7 that the resulting time-varying filter will have properly bandlimited coefficients regardless of the spectral modifications made on each FFT.

Our formulation may be altered slightly to provide excellent performance during regions of silence. This is called the "idle channel" condition in the ADM literature. Inspection of (1) and (2) reveals that

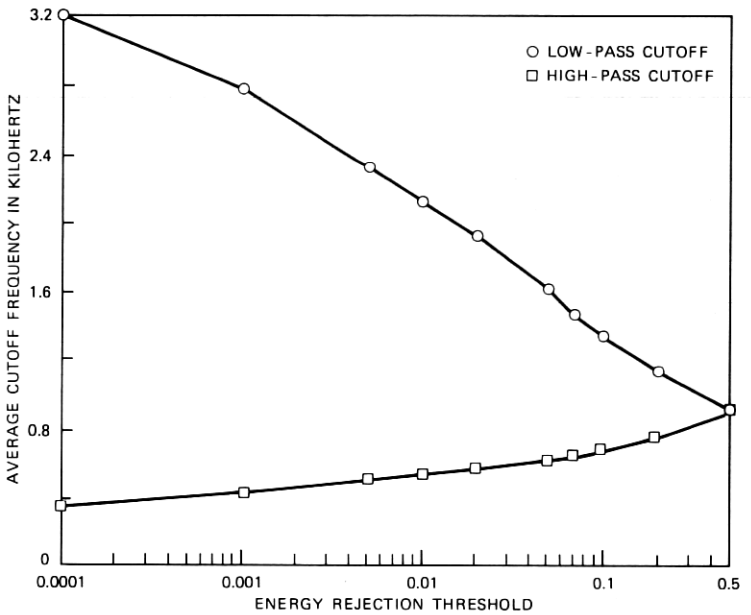

Fig. 5—Time-averages of the upper and lower band edges vs spectrum energy-rejection threshold. The average is taken over the entire utterance of Fig. 2 for each threshold. At the far left of the figure, the energy-rejection thresholds are near zero so that the band edges lie at the outer extremes of the true speech band. At the far right, the two thresholds are equal to 0.5 corresponding to rejection of the upper and lower 50 percent of the speech spectrum; consequently, the band edges meet at the median frequency.

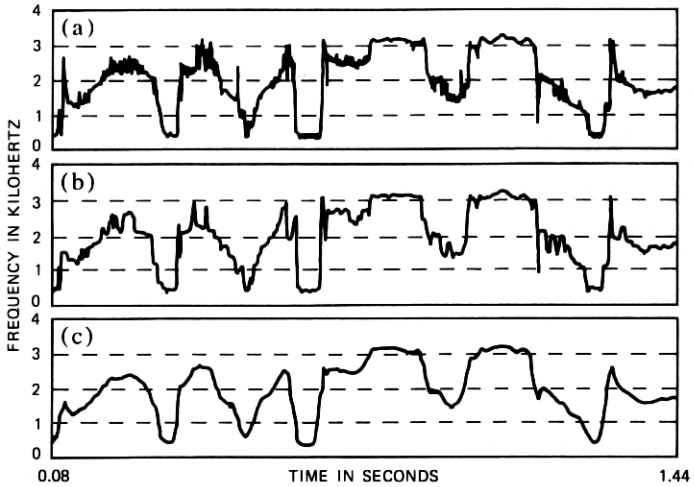

Fig. 6—Illustration of the effects of using an under-sized Fourier transform, and the possibility of compensation via filtering of the band-edge waveforms. (a) Upper 1 percent band edge for the same speech sample as in Fig. 2 using only a 10-ms time frame for spectrum computation. This causes oscillation of the measured band edge at the pitch frequency. (b) The same band-edge function of (a) after filtering with an order 7 median smoother. (c) The same curve of (a) after filtering with an order 7 moving average. Linear smoothing results in band-edge time behavior similar to that obtained when using a larger Fourier transform; however, the frequency resolution of the band-edge values is still sacrificed.

the cutoff frequencies are indeterminate in this situation [$|X(\omega, t)| = 0$]. If there is any amount of white noise present in the input signal, then the time-varying bandwidth, $f_U - f_L$, will open to full bandwidth as if the speech itself were spectrally flat. This undesirable behavior may be suppressed by various ad hoc schemes. In the simulations, we added a small positive value to $E(t)$. That is, (2) is replaced by

$$\tilde{E}(t) = \int_0^\infty |X(t, f)|^2 \, df + \sigma_{\min}^2, \tag{3}$$

where $\sigma_{\min}^2$ may be thought of as noise energy, or as a lower bound on the acceptable speech level. As the speech energy falls to zero, the band edges cross, corresponding to disjoint low-pass and high-pass filters, and we must therefore define all negative values of $f_U - f_L$ to be zero bandwidth. Also, there exists the possibility that no solution to (1) exists, for $\sigma_{\min}^2 > 0$, in which case the bandwidth is again set to zero. Thus, when the channel is idle, there will be zero bandwidth and subsequently no output signal. This fact can be used to advantage when optimizing the step-size adaption algorithm in the ADM coder.[3]

Our simulations assume that the band edges f_U and f_L are transmitted as side information in the variable bandwidth coding system. However,

it is worth considering filter adaption based only on the received speech data. Increasing the receiver energy rejection thresholds T_U and T_L relative to the transmitter thresholds will contract the (estimated) band edges so as to compensate for the artificial band expansion that occurs because of coder noise in the received spectrum.

If the bandlimits are transmitted as side information, then the increase in data rate is relatively small. As a practical example, if the FFT length is 512, a Hamming window is used, and the speech sampling rate is 8 kHz, then we have a pair of band edge values every 16 ms. Furthermore, the band edge values are quite smoothly behaved, and can be coded more efficiently. It appears that the band-edge waveform signals $f_L(t)$ and $f_U(t)$ have a bandwidth on the order of 30 Hz for speech.[5]

III. RESULTS OF ADM SIMULATIONS

In this section, we present s/n evaluations of the four ADM coder configurations (described in Section I) ADM, ADM-PF, ADM-PPF, and ADM-AR. The comparisons are made using the s/n and segmental s/n[8] measures defined in Appendix B. Detailed parameter information may be found in Appendix C.

The degree to which quality is enhanced by adaptive post-filtering depends on the character of the coder noise. If the coder noise is known to be stationary additive white noise, uncorrelated with the speech, then the gain in s/n may be predicted in advance from f_U and f_L. Given that bandlimiting the speech causes no distortion, the s/n of each segment will increase by

$$\text{s/n increase (dB)} = 10 \log \frac{\text{(maximum bandwidth)}}{\text{(short-time bandwidth)}}$$

$$= 10 \log \left(\frac{f_{\max}}{f_U - f_L} \right), \tag{4}$$

where $f_{\max}$ is the full channel bandwidth. For a "typical" frame (f_U = 2 kHz, f_L = 400 Hz, C = 3 kHz), this is about 2 dB. The gain in quality at lower bit rates is dramatically greater than indicated by the s/n. This is perhaps due to the high perceptual significance of out-of-band coder noise and/or auditory masking of in-band noise by the speech.

To anticipate the improvement of ADM due to post-filtering, we need to know the spectral distribution of ADM coder noise. While some theoretical work along these lines has been done,[9] it is difficult analytically to derive general estimates of the short-time noise power spectral density. Some intuition may be obtained, however, from simulations on isolated, quasi-stationary speech segments.

Figure 7a shows a tenth-order LPC spectral envelope[10] for the front

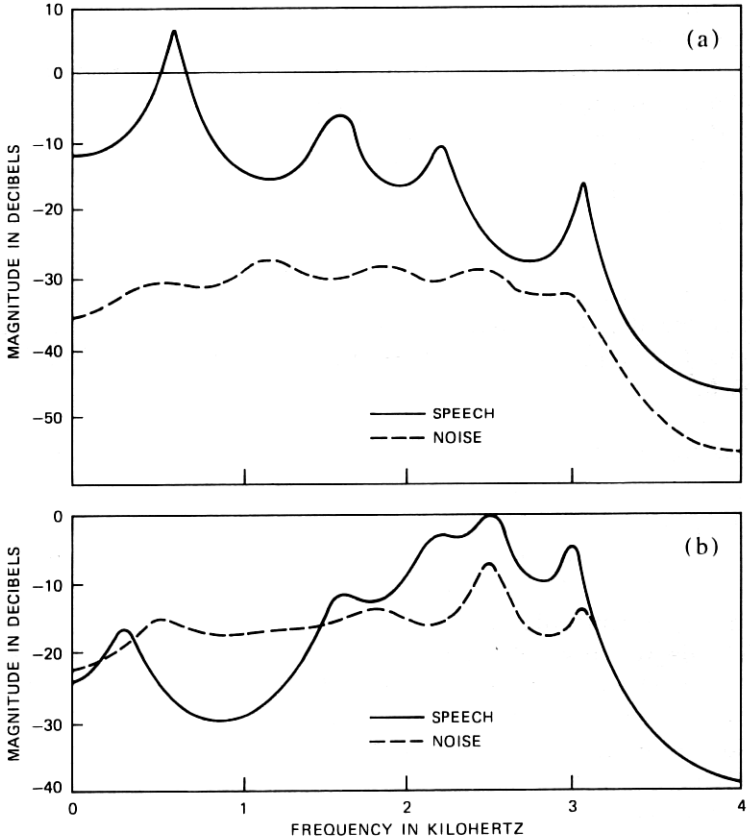

Fig. 7—Spectral envelopes of signal and noise for two representative sounds. All spectral envelopes were calculated using a tenth-order linear prediction on 1024 samples of data. The noise was isolated from the speech by subtracting the noiseless (precoder) speech from the ADM coder output (which used time-invariant filtering). (a) Spectral envelope for the vowel "a" from "grab" superimposed with the spectral envelope of its associated ADM coder noise. (b) Spectral envelopes for the "s" sibilant in "sugar" and its corresponding coder noise.

vowel "a" (as in "grab") superimposed with the tenth-order spectral envelope of the normal 24-kbps ADM coder noise generated by this vowel. (Appendix C gives detailed analysis parameters.) The measured s/n is 15 dB, and the noise spectrum within the passband is relatively flat. It should be noted that the slight ripple in the spectral envelope of the error signal depends on the order of the linear predictor.

Figure 7b shows the same comparison of signal and noise spectral envelopes for the "sh" sound in "sugar." Note that in this case, the noise is fairly flat out to 2.4 kHz after which it begins to follow the speech spectrum. The noise has a significant peak near 2.6 kHz indicating that these spectral components could not be properly

tracked. In this example, it was evident from the time-domain waveform that the coder was tracking dominant high-frequency components with a large positive error in the amplitude difference estimate (adaptive step-size inside the ADM coder).[2] The measured s/n for this sibilant is only 1.6 dB, and the primary character of the noise is that of rough loud "static."

Generalizing from Fig. 7, we might expect low-pass signals to generate coder noise that may be approximately modeled as white, and high-pass speech segments to correspond to relatively strong correlated noise. Such heuristics, while over-simplified, serve to point out the more generally observed differences in ADM noise characteristics for voiced vs unvoiced speech. Awareness of these two contrasting cases aids in the interpretation of the segmental s/n in which the s/n for individual phonemes is evident.

Figure 8b gives a plot of the segmental s/n (defined in Appendix B) versus time for the three cases ADM, ADM-PF, and ADM-AR. The bit rates of ADM and ADM-PF are 32 kbps. ADM-AR has 32 kbps as its maximum instantaneous bit-rate while the average rate for this particular phrase is 23 kbps. Figure 8a shows the segmental input rms level from which the various phonemes may be located. The segment size is

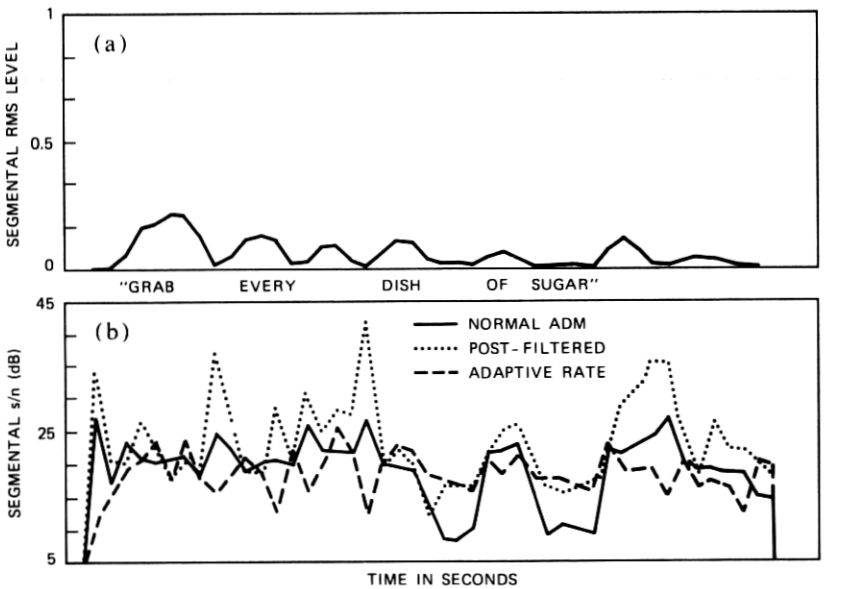

Fig. 8—Time behavior of ADM noise for three cases as defined by the s/n of each 32-ms time frame (segmental s/n). (a) Segmental rms amplitude of speech utterance vs time indicating phoneme locations. (b) Segmental s/n vs time for normal ADM and post-filtered ADM (ADM-PF) at a bit rate of 32 kbps, and adaptive-rate ADM (ADM-AR) having a peak bit rate of 32 kbps.

32 ms in both Figs. 8a and 8b. We may observe several features in the behavior of the segmental s/n due to post-filtering and adaptive rate:

(*i*) There is little difference among the three cases for front vowels such as "a" in "grab" and "e" at the beginning of "every." From Fig. 2, we see that during these segments, the speech bandwidth is wide and almost fully occupies the channel bandwidth. Consequently, the adaptive low pass is almost the same as the fixed low pass, and ADM-AR is running at maximum sampling rate during the greater portion of these vowels.

(*ii*) When the low-pass cutoff $f_U(n)$ is small, ADM-PF realizes large quality gains due to rejection of much out-of-band coder noise. In contrast, ADM-AR exchanges these gains in return for reduced sampling rate. Examples of this may be seen at the phonemes corresponding to "b," "v," "d," and "u."

(*iii*) When the high-pass cutoff $f_L(n)$ is large [at which time $f_U(n)$ is maximum], ADM-AR reduces to the case ADM-PPF, and its performance is close to that of ADM-PF. Both exhibit higher segmental s/n than normal ADM due to elimination of low-frequency noise. This condition may be observed at the two unvoiced regions "sh" and "s."

We now turn to plots of segmental s/n averaged over the entire utterance, and we denote the average segmental s/n by s/n_{seg}. Figure 9 shows s/n_{seg} vs bit rate for all four test cases. The post-filtered case, ADM-PF, is 2.8 dB better than normal ADM on the average. Note that the prefiltering in ADM-PPF, which reduces ADM tracking error, adds

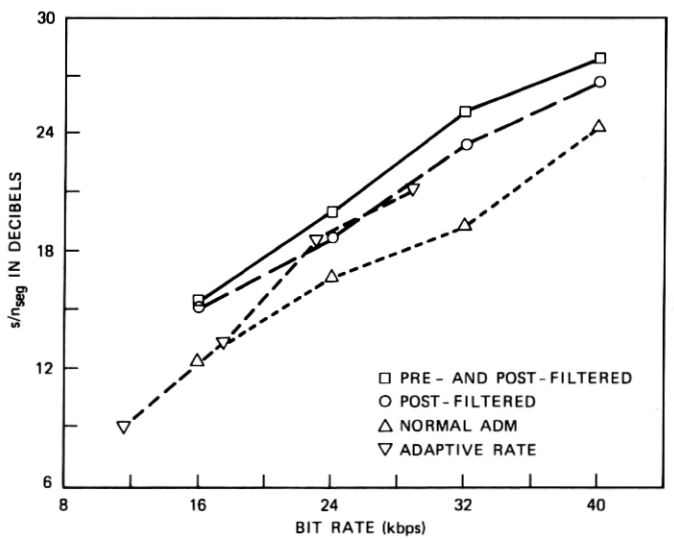

Fig. 9—Segmental s/n averaged over the entire utterance for four cases, plotted as a function of bit rate. For the case of adaptive rate, the average bit rate is used as abscissa.

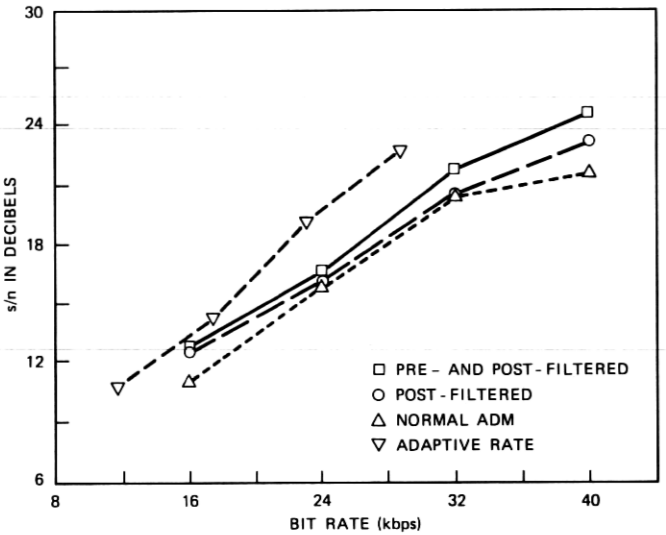

Fig. 10—Normal s/n computed on entire utterance for four cases, plotted as a function of bit rate. For the case of adaptive rate, the average bit rate is used as abscissa.

still another dB or so to the s/n_{seg} for ADM-PF, and the improvement is always within the short-time speech band. The adaptive rate coder, ADM-AR, exhibits s/n_{seg} between normal and post-filtered ADM. Overall, the s/n_{seg} measure corresponds well with subjective quality ratings. From informal listening to the speech samples represented in Fig. 9, we feel that ADM-PPF is not noticeably better than ADM-PF; ADM-PF is somewhat "cleaner" than ADM-AR (comparing where the maximum ADM-AR rate equals the ADM-PF rate), and normal ADM is definitely inferior due to the audible high-frequency noise which is allowed to pass.

Figure 10 gives s/n (as opposed to s/n_{seg}) for the same four cases (cf. Appendix B). All deviations from Fig. 9 are due to the fact that the s/n_{seg} measure is an average of the s/n's (in dB) obtained from disjoint 256-point frames while the s/n measure treats the entire speech sample as one frame. Since the regions of quality gain in ADM-PF and ADM-PPF are of low relative energy, they contribute little to the s/n. ADM-AR appears in this figure to be significantly superior, but this is misleading; ADM-AR has high distortion in the relatively low-energy low-bandwidth regions due to the large reduction in sampling rate, and the s/n measure does not adequately penalize it. For example, the consonants "b" and "d" might be least distinguishable in the ADM-AR case, relative to the other three, even though it scores the highest s/n. Thus, the s/n measure is overly insensitive to low-amplitude intelligibility loss, especially in the case of ADM-AR.

IV. CONCLUSIONS

It has been shown that the quality of ADM coded speech can be significantly improved by employing a time-dependent low-pass filter matched to the short-time speech bandwidth. A time-varying high-pass cutoff may be added with little additional computational cost, but its contribution to quality is small and sometimes perceptually distracting due to audible noise modulation at bit rates below 24 kbps. Transmission of the slowly varying cutoff frequencies adds only slightly to the transmission bit rate. Two uses of the adaptive low-pass cutoff were discussed. First, time-varying low-pass filtering of the ADM decoded signal was found to add quality commensurate with a large increase in ADM bit rate (e.g., 24 kbps quality at 16 kbps). Secondly, time-varying low-pass filtering before and after the ADM coder, coupled with a time-varying sampling rate, gave nearly the same quality as normal ADM but with a large reduction in the average bit rate (e.g., 24 kbps from 32 kbps). The gains cited are for continuous speech, and better relative performance is to be expected for speech containing regions of silence. The final conclusions concerning quality are based on casual listening tests and are only indirectly supported by the s/n measures employed.

V. ACKNOWLEDGMENTS

The authors wish to thank N. S. Jayant for sharing his expertise on ADM coding, and for recurrent help throughout the project. We also thank J. L. Flanagan for numerous ideas and suggestions which were incorporated into this investigation.

APPENDIX A

Implementation of Variable Bandwidth ADM Simulation

Referring again to Fig. 1, the software implementation is as follows. The input speech is sampled at 8 kHz, bandlimited to the typical channel bandwidth for telephone communication (200–3200 Hz), and is then resampled at 16, 24, 32, or 40 kHz. The data is partitioned into overlapping frames of 512 samples, a Hamming window is applied,[11] and the FFT of each frame is taken. The speech cutoff frequencies f_U and f_L are computed for each frame, as discussed in Section II.

If prefiltering is included or if the sampling rate is to be lowered, the spectrum values outside the cutoff frequencies are tapered to zero using a precomputed filter band edge. The filter band edge is computed using a window design method based on a Kaiser window.[11] Next, an inverse FFT is taken on each frame, and the time-domain waveform is reconstructed by adding the frames back together, partially overlapped in time (overlap-add synthesis[6]).

The decimation stage is only active during sampling rate reduction, and it operates by selecting every mth sample, where $m = \lceil 0.5\, f_s/f_U \rceil$ is the sampling rate reduction factor. $\lceil x \rceil$ denotes the smallest integer $\geq x$, namely, m is the greatest integer such that m times the low-pass cutoff frequency for the current frame does not exceed the upper channel band edge. Note that the integer decimation method of varying the sampling rate does not take full advantage of the unused bandwidth; however, it has the advantage that it is quite simple to implement.

The coder is a one-bit ADM coder with exponential step-size adaptation as described in Ref. 2. The coder output and the time-varying bandwidth information are assumed to be transmitted through a noiseless channel.

The ADM decoder is followed by a sample interpolator to restore the original sampling rate (when applicable), and the interpolator is followed by a time-varying filter. This filter is also implemented via short-time spectrum analysis, modification, and synthesis; it restricts the decoded speech spectrum to its original natural bandwidth, when post-filtering is employed, thus removing out-of-band coder noise. This filter is also part of the interpolation process as the interpolator merely inserts $m = \lceil 0.5\, f_s/f_U \rceil$ zeros between each sample.

APPENDIX B

Signal-to-Noise Ratio Calculation

Two types of signal-to-noise ratio are defined. The most common form is

$$\text{s/n} \triangleq 10 \log \left| \frac{\sum_{m=0}^{N-1} (x(m) - \mu_x)^2}{\sum_{m=0}^{N-1} (e(m) - \mu_e)^2} \right|,$$

where $x(m)$ is the signal with sample mean

$$\mu_x \triangleq \frac{1}{N} \sum_{m=0}^{N-1} x(m),$$

$e(m)$ is the noise with sample mean μ_e, and N is the total number of samples available for the s/n measurement.

The definition of s/n diverges from subjective quality ratings for large N due to the fact that high-amplitude signal regions dominate the influence of low-amplitude signal regions during the s/n calculation. This insensitivity may be partially circumvented by computing s/n values over segments of some reasonably small size M (e.g.,

spanning 20 ms), and averaging the s/n (dB) values of the segments. Accordingly we define

$$\text{segmental s/n}(k) \triangleq 10 \log \left| \frac{\sum_{m=0}^{M-1} [x_k(m) - \mu_x(k)]^2}{\sum_{m=0}^{M-1} [e_k(m) - \mu_e(k)]^2} \right|,$$

$$\text{s/n}_{\text{seg}} \triangleq \frac{1}{N} \sum_{k=0}^{N-1} \text{segmental s/n}(k)$$

$$= \frac{1}{N} \sum_{k=0}^{N-1} 10 \log \left| \frac{\sum_{m=0}^{M-1} [x_k(m) - \mu_x(k)]^2}{\sum_{m=0}^{M-1} [e_k(m) - \mu_e(k)]^2} \right|,$$

where N is the number of segments of length M, $x_k(\cdot)$ is the kth segment of the signal, and $\mu_x(k)$ is the sample mean of the kth segment.[8] In the case of s/n_{seg}, the measure is vulnerable to domination by segments having insignificant signal energy (i.e., the s/n can approach $-\infty$ in a time frame where the signal is silent and where there is any amount of noise). Consequently, if the total energy (sum of samples squared) in a given segment is below a prescribed energy threshold, the segment is eliminated from the computation of s/n_{seg}. (This feature was not needed for the continuous speech samples used in the ADM simulations.)

In all ADM tests, the noise $e(m)$ is calculated as the point-wise difference between the noisy coded signal and a signal which was generated in precisely the same way but bypassing the ADM coder. In this way, all side effects of bandlimiting, processing delay, etc., are eliminated from the calculated error. Measurement of s/n in an ADM coding system is facilitated by the fact that it is a waveform coder (as opposed to source coder), and thus does not have the inherent delay, phase-dispersion, or level-offset characteristics that commonly impede the objective measurement of subjective signal quality.

APPENDIX C

System Parameters Used in Generating s/n Curves

Coder input: Phrase = "*Grab every dish of sugar*" from an adult male speaker, sampled at 8 kHz, and bandlimited to 200–3200 Hz with a 256-point FIR bandpass.

Time-varying filters: In all runs, the filters were implemented via modified FFTs of length $N = 512$. To prevent time-aliasing, the number of data points N_x brought into the FFT input buffer plus the length N_h

of the Kaiser window (used as the basis of the time-varying filter) cannot exceed N. Furthermore, short-time spectral modification theory requires that the step-size through the data (time offset between successive FFTs) not exceed $N_x/4$ for the case of a Hamming window on the FFT input.[6] The table below gives the employed data frame size N_x and time-varying filter length N_h as a function of sampling rate f_s for all ADM simulations.

f_s(kHz)	N_x	N_h
16	304	208
24	456	56
32	456	56
40	400	112

The filter controls $f_U(n)$ and $f_L(n)$ are each eight-bit values at a sampling rate of $4f_s/N_x$.

ADM coder: The step-size multipliers were experimentally found to give good results with $P = 1.2$, $Q = 0.9$.[2] These values did better than $P = 1/Q = 1.5$, $P = 1/Q = 1.2$, and a few other trial settings, in terms of the s/n and s/n_{seg} measures.

LPC spectral envelopes: The short data segments "a" and "s" were each processed with $N = 512$, $f_s = 24$ kHz, $N_x = 456$, $N_h = 56$, fixed filters, and nonadaptive sampling rate. The tenth-order LPC spectral envelopes were calculated using 1024 data samples.

REFERENCES

1. R. W. Schafer and L. R. Rabiner, "A Digital Signal Processing Approach to Interpolation," *Digital Signal Processing II*, New York: IEEE Press, 1976. (Also Proc. IEEE, June 1973.)
2. N. S. Jayant, "Adaptive Delta Modulation with a One-Bit Memory," B.S.T.J., *49*, No. 3 (March 1970), pp. 321–42.
3. N. S. Jayant, "Digital Coding of Speech Waveforms: PCM, DPCM, and DM Quantizers," Proc. IEEE, *62*, No. 5 (May 1974), pp. 611–32.
4. L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*, Englewood Cliffs, N.J.: Prentice-Hall, 1978.
5. J. B. Allen, "A Method for Simultaneous Transmission of Data Over Voice," unpublished work.
6. J. B. Allen, "Short Term Spectral Analysis, Synthesis, and Modification by Discrete Fourier Transform," IEEE Trans. on Acoustics, Speech, and Signal Proc., *ASSP-25*, No. 3 (June 1977), pp. 235–8.
7. J. B. Allen and L. R. Rabiner, "A Unified Theory of Short-Time Spectrum Analysis and Synthesis," Proc. IEEE, *65*, No. 11 (November 1977), pp. 1558–64.
8. P. Noll, "Nonadaptive and Adaptive DPCM of Speech Signals," Polytech. Tiijdschr. Ed. Elektrotech/Elektron (The Netherlands), No. 19 (1972).
9. N. S. Jayant, "A First-Order Markov Model for Understanding Delta Modulation Noise Spectra," IEEE Trans. Comm., *COM-26*, No. 8 (August 1978), pp. 1316–8.
10. J. D. Markel and A. H. Gray, *Linear Prediction of Speech*, Berlin: Springer-Verlag, 1976.
11. L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing*, Englewood Cliffs, N.J.: Prentice-Hall, 1975.
12. J. L. Flanagan et al., "Speech Coding," IEEE Trans. Comm., *COM-27*, No. 4 (April 1979), pp. 710–37.