Amplitude Optimization and Pitch Prediction in Multipulse Coders

SHARAD SINGHAL, MEMBER, IEEE, AND BISHNU S. ATAL, FELLOW, IEEE

Abstract—Multipulse excitation is an attractive way of modeling the input to the LPC synthesis filter at medium bit rates. The excitation consists of a few pulses, defined by their locations and amplitudes, per frame of speech. The pulses are determined by minimizing a weighted mean-squared error between the original speech and the output of the LPC synthesis filter. Unfortunately, the minimization results in a combinatorial problem and a global solution is computationally expensive. Thus, the excitation is typically obtained in stages—one pulse per stage. At each stage, the amplitude and location of a pulse is obtained using an exhaustive search. Pulses obtained in previous stages are assumed constant during the search. For high quality synthesis, 6–8 pulses are required per pitch period of speech. However, this implies that for the same speech quality, the female speech requires a higher bit rate than male speech.

In this paper we review the basic multipulse model and describe an algorithm to obtain the excitation pulses with optimally adjusted pulse amplitudes. Within the framework of the algorithm, the excitation can be computed with varying degrees of optimization, and with a few modifications the procedure can search for more than one pulse at each stage and use nonexhaustive searches. We show that for high pitched speakers, the pitch dependence of multipulse excitation can be reduced and speech quality improved by including long delay predictors in the multipulse model.

I. Introduction

Linear prediction is widely used today for speech analysis, synthesis, and coding. In linear predictive voice coding (LPC), speech is modeled as the output of an all-pole filter, driven by an excitation function. The excitation consists of a sequence of pulses separated by the pitch period for voiced speech and pseudorandom noise for unvoiced speech. Although the output of this method is highly intelligible, it is not natural sounding even at high bit rates. The exact reasons why the output sounds unnatural are not well understood, but it is believed that the main problem is modeling the excitation in terms of pitch pulses and white noise. This model assumes that speech can be classified into only two categories—voiced and unvoiced. However, there are portions of speech where it is not clear whether the speech is voiced or unvoiced. Moreover, accurate determination of pitch period in voiced speech is sometimes difficult.

High quality speech can be obtained by using a different model for the excitation to the LPC synthesis filter. The model, first proposed by Atal and Remde [1], [2], has been dubbed multipulse excitation, and has received much attention among speech researchers [3]–[18]. In this model, instead of a single pulse per pitch period for voiced speech and white noise for unvoiced speech, the excitation consists of a few pulses per frame of speech regardless of whether the speech is voiced or unvoiced. Since all sounds are synthesized using the same form of excitation, the problems of pitch estimation and voiced–unvoiced analysis are avoided. The pulse amplitudes and locations are not specified *a priori*, but are computed on a frame-by-frame basis by minimizing a weighted mean-squared error between the original speech signal and the output of the LPC synthesis filter. The number of pulses needed in each frame of speech depends on the desired speech quality, the more pulses per frame, the better the speech quality. Generally, 6–8 pulses every 10 ms of speech are enough for synthesizing high quality speech.

Although the multipulse model is conceptually simple, the problem of locating the pulses is computationally complex. In this paper we discuss the basic multipulse model and describe a procedure to compute the excitation with optimally adjusted amplitudes. The algorithm provides a framework for computing multipulse excitation with varying degrees of optimization and computational complexity. As mentioned above, the speech quality depends on the pulse rate. We also find that for the same quality, female speech requires a higher pulse rate than male speech. The pitch dependence can be reduced and speech quality improved for high-pitched speakers by incorporating long delay prediction in the multipulse model.

II. The Multipulse Model

Fig. 1 shows the block diagram of the analysis procedure described by Atal [2]. The excitation generator produces a sequence of pulses as the input to an LPC all-pole filter. The resulting error signal between the output of the all-pole filter and the original speech is suitably weighted and fed back to the excitation generator. The excitation generator computes the amplitudes and locations of the pulses such that the error (distortion) between the original and synthetic speech is minimized. It is this "closed loop" structure that differentiates multipulse excitation from other pulse-like excitations such as a center-clipped or thinned-out LPC residual [19]. A frequency weighted mean-squared error criterion is usually used as the distor-

Manuscript received October 30, 1986; revised March 28, 1988.

S. Singhal is with Bell Communications Research, Morristown, NJ 07960.

B. S. Atal is with AT&T Bell Laboratories, Murray Hill, NJ 07974.

IEEE Log Number 8825658.

0096-3518/89/0300-0317$01.00 © 1989 IEEE

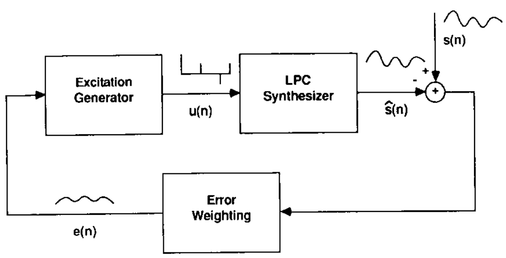

Fig. 1. Block diagram of a multiple coder.

tion measure. This criterion leads to a straightforward solution and has intuitive appeal since it minimizes the weighted segmental signal-to-noise ratio.

The frequency weighting on the error is perceptually motivated. The unweighted mean-squared error criterion minimizes the rms difference between the original and synthetic signals and results in an error spectrum that is "white." However, it is well recognized that the perceptual distortion is not based on the rms error alone. Due to auditory masking, we can tolerate much more noise in frequency regions where the speech has significant energy. Thus, the noise components in these regions can have higher energy relative to the components in low energy regions without increasing the perceptual distortion. A simple way of controlling the noise spectrum is by filtering the error signal through a weighting filter before minimization. We use a weighting filter of the form [20]

$$W(z) = [1 + P(z)]/[1 + P(z/\gamma)], \tag{2.1}$$

where

$$1 + P(z) = 1 + \sum_{k=1}^{p} a_k z^{-k} \tag{2.2}$$

is the LPC inverse filter with coefficients $a_1, \cdots, a_p$ and γ is a fraction between zero and one that determines the degree to which the error is deemphasized in any frequency region.[1] The deemphasis concentrates the error energy under the high energy portions of the speech spectrum thus achieving the desired spectral shaping of the error.

III. Computation of Multipulse Excitation

Multipulse excitation is computed by minimizing the weighted mean-squared error between the speech signal $s(n)$ and the synthetic speech signal $\hat{s}(n)$. Since the weighting filter is linear, the minimization is equivalent to minimizing the unweighted mean-squared error between a weighted speech signal $y(n)$ and the corresponding weighted synthetic signal $\hat{y}(n)$ as shown in Fig. 2. The speech signal $s(n)$ is filtered through $W(z)$ to obtain

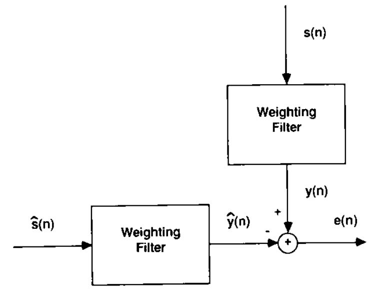

Fig. 2. Computation of the weighted error signal.

the weighted speech signal $y(n)$, i.e.,

$$y(n) = s(n) + \sum_{k=1}^{p} a_k[s(n-k) - \gamma^k y(n-k)]. \tag{3.1}$$

Similarly, the excitation signal $u(n)$ is filtered through a cascade of the LPC all-pole filter and the weighting filter to obtain the weighted synthetic speech signal $\hat{y}(n)$. Let us block speech in frames of N samples each and let $h(n)$ and $u(n)$, $0 \le n < N$, be the impulse response of the weighted all-pole filter and the input to this filter, respectively, in the current frame. Then $\hat{y}(n)$ for the present frame is given by

$$\hat{y}(n) = \sum_{i=0}^{n} h(i)u(n-i) + \hat{y}_0(n), \qquad 0 \le n < N, \tag{3.2}$$

where $\hat{y}_0(n)$ is the contribution to $\hat{y}(n)$ from the filter memory.

If the input to the present frame is modeled as a series of m pulses of amplitude $\beta_0, \beta_1, \cdots, \beta_{m-1}$ at times $n_0, n_1, \cdots, n_{m-1}$, respectively, we have

$$u(n) = \sum_{k=0}^{m-1} \beta_k \delta(n - n_k) \tag{3.3}$$

where $\delta(n)$ is the Kronecker delta. Thus,

$$\hat{y}(n) = \sum_{i=0}^{n} h(i) \sum_{k=0}^{m-1} \beta_k \delta(n - i - n_k) + \hat{y}_0(n), \qquad 0 \le n < N, \tag{3.4}$$

or

$$\hat{y}(n) = \sum_{k=0}^{m-1} \beta_k h(n - n_k) + \hat{y}_0(n), \qquad 0 \le n < N. \tag{3.5}$$

The pulse amplitudes $\beta_0, \beta_1, \cdots, \beta_{m-1}$ and the corresponding locations $n_0, n_1, \cdots, n_{m-1}$ are now obtained by minimizing the squared error E defined by

$$E = \sum_{n=0}^{N-1} [y(n) - \hat{y}(n)]^2. \tag{3.6}$$

The error E has to be minimized with respect to the

[1] The parameter γ is chosen from perceptual considerations. A reasonable value of γ is around 0.8 or 0.9. For this weighting filter, γ should not be made unnecessarily small because the error spectrum stays flat and any perceptual advantage is lost for small values of γ. Further, as $\gamma \to 0$, $W(z) \to [1 + P(z)]$, and the transfer function of the cascade formed by the weighting filter and the LPC all-pole filter approaches unity. This in effect "breaks" the analysis-by-synthesis loop and leads to the trivial solution that the synthesis filter input should be simply the LPC prediction residual.

pulse amplitudes $\beta_0, \cdots, \beta_{m-1}$ as well as the pulse locations $n_0, \cdots, n_{m-1}$. Minimization with respect to the pulse amplitudes is straightforward. We substitute for $\hat{y}(n)$ from (3.5) in (3.6) and set the derivatives with respect to β_j, $0 \le j < m$, equal to zero to obtain the equations

$$\sum_{k=0}^{m-1} \beta_k \alpha_{n_k n_j} = c_{n_j}, \quad 0 \le j < m, \tag{3.7}$$

where α is the autocorrelation of the impulse response $h(n)$, i.e.,

$$\alpha_{ij} = \sum_{n=0}^{N-1} h(n-i)h(n-j), \quad 0 \le i, j < N, \tag{3.8}$$

and c is the cross correlation between the impulse response $h(n)$ and the signal $\bar{y}(n) = y(n) - \hat{y}_0(n)$, i.e.,

$$c_i = \sum_{n=0}^{N-1} \bar{y}(n)h(n-i), \quad 0 \le i < N. \tag{3.9}$$

The minimum value of E is given by

$$E_{\min} = \sum_{n=0}^{N-1} \bar{y}^2(n) - \sum_{k=0}^{m-1} \beta_k c_{n_k}. \tag{3.10}$$

Minimization of E with respect to the pulse locations $n_0, n_1, \cdots, n_{m-1}$ is a combinatorial problem and does not have a closed form solution. We need to choose m of the possible N locations such that the corresponding solution of (3.7) results in minimum error.

A. Amplitude Reoptimization Method

An exhaustive search for pulse locations requires $\binom{N}{m} = N!/m!(N-m)!$ solutions of (3.7) and quickly becomes impractical as the number of pulses m increases. The required computation can be kept within reasonable bounds by searching for pulse locations in stages; at each stage, the amplitude and location of only one pulse is allowed to vary. This reduces the computation required to m searches of order N [2], [8]. Once the pulse locations are found, (3.7) is used to reoptimize all pulse amplitudes (hence the name amplitude reoptimization method). We briefly summarize this procedure, first described by Atal in [1] and [2]. Let the input consist of only one pulse located at sample i. Equations (3.7) and (3.10) then reduce to

$$\beta_0(i) = c_i/\alpha_{ii}, \tag{3.11}$$

and

$$E_{\min}(i) = \sum_{n=0}^{N-1} \bar{y}^2(n) - c_i^2/\alpha_{ii}, \tag{3.12}$$

β_0 and $E_{\min}$ are shown as explicit functions of i to emphasize their dependence on the pulse location i during the search. From (3.12) we see that $E_{\min}(i)$ is minimum for that value of i where c_i^2/α_{ii} is maximum. This implies that the first pulse should be located at the maximum of the normalized cross-correlation function. Let this location be n_0 and the corresponding pulse amplitude from (3.11) be β_0. We now assume that β_0 and n_0 are known constants in (3.5) and minimize E with respect to $\beta_1, \cdots, \beta_{m-1}$ to obtain

$$\sum_{k=1}^{m-1} \beta_k \alpha_{n_k n_j} = c_{n_j} - \beta_0 \alpha_{n_0 n_j}, \quad 1 \le j < m. \tag{3.13}$$

If $m = 2$, (3.13) contains only two unknowns, β_1 and n_1. If the second pulse is located at i, we have

$$\beta_1(i) = [c_i - \beta_0 \alpha_{n_0 i}]/\alpha_{ii}, \tag{3.14}$$

and

$$E_{\min}(i) = \sum_{n=0}^{N-1} \bar{y}^2(n) - [c_i - \beta_0 \alpha_{n_0 i}]^2/\alpha_{ii}. \tag{3.15}$$

As before, we set $n_1 = i$ where the second term in (3.15) is maximum, and the corresponding amplitude is computed from (3.14). Note that (3.15) is the same form as (3.12) with the cross correlation c_i replaced by $c_i - \beta_0 \alpha_{n_0 i}$, and the pulse location corresponds to the maxima of the normalized value of this "updated" cross correlation. The procedure is repeated for each pulse, i.e., the $j + 1$th pulse is located at the maxima of

$$[c_i^{(j)}]^2/\alpha_{ii} = [c_i^{(j-1)} - \beta_{j-1}\alpha_{n_{j-1} i}]^2/\alpha_{ii}, \quad 0 \le i < N, \tag{3.16}$$

and its amplitude is

$$\beta_j = c_{nj}^{(j)}/\alpha_{n_j n_j}, \tag{3.17}$$

with $c_i^{(0)} = c_i$.

The method thus consists of three steps for each pulse. First, a cross-correlation function is computed. Next, the pulse location is obtained by locating the maximum of the normalized cross-correlation function. Finally, the pulse amplitude is computed. The pulse location and amplitude are then used to obtain the cross-correlation function for the next pulse. Once all pulses have been located, the pulse amplitudes are recomputed using (3.7)

This method works best when the pulses are widely spaced. The performance suffers, however, for closely spaced pulses. This is because the method implicitly assumes that amplitudes of past pulses remain constant during the search for the current pulse location. Thus, the pulse locations and amplitudes obtained during the search do not satisfy (3.7). Although the final reoptimization recomputes the amplitudes, the locations remain suboptimal. The resulting error is small if the pulses are widely spaced, but causes degradation in the output when the pulses are closely spaced.

Many variations of this method are possible and have been discussed elsewhere in the literature [5]–[8], [11]–[18]. For example, it can be assumed that $\alpha_{ij} = \alpha_{|i-j|}$. Thus, the normalization of the cross-correlation function becomes unnecessary. Similarly, it is possible to reoptimize the pulse amplitudes after locating each pulse rather than only at the end of the procedure [6]–[8], [11], [18].

B. Optimal Amplitude Computation

The problem of locating the pulses can also be formulated in a slightly different way [3], [4], [6]–[8], [11], [15]. Instead of obtaining m location values, we stipulate that there is a pulse present at every location. However, only m of the amplitude values are nonzero. The problem of computing the pulse amplitudes and locations then reduces to the problem of choosing the best subset of m out of N variables; i.e., given the variables $\beta_0, \cdots, \beta_{N-1}$, we wish to minimize the error $E_{\min}^{(N-1)}$ subject to the constraint that only m of the variables are nonzero. Although many algorithms to compute the global optima for this problem exist [21], [22], they are computationally complex and, for the typical multipulse analysis problem with 80–100 variables, are impractical.

Suboptimal techniques for obtaining the best subset of variables take two approaches. In the first, variables are progressively added to the problem such that the reduction in the error at each stage is maximized; in the second, variables are dropped such that the increase in the error at each stage is minimized. Algorithms for obtaining the best subset usually use a mix of these two methods. However, in multipulse analysis, the number of significant variables is known to be a small fraction of the total and it is more efficient to start from a single variable and progressively add variables.

With this formulation in mind, we rewrite (3.7) and (3.10) as

$$\boldsymbol{\alpha}\boldsymbol{\beta} = \boldsymbol{c} \tag{3.18}$$

and

$$E_{\min}^{(N-1)} = \boldsymbol{y}^t\boldsymbol{y} - \boldsymbol{\beta}^t\boldsymbol{c}, \tag{3.19}$$

where $\boldsymbol{\alpha}$ is the $N \times N$ matrix[2] of autocorrelation terms

$$(\boldsymbol{\alpha})_{ij} = \alpha_{ij} = \sum_{n=0}^{N-1} h(n-i)h(n-j), \qquad 0 \le i, j < N, \tag{3.20}$$

$\boldsymbol{c}$ is the N vector containing the cross-correlation terms

$$(\boldsymbol{c})_j = c_j = \sum_{n=0}^{N-1} \bar{y}(n)h(n-j), \qquad 0 \le j < N, \tag{3.21}$$

$\boldsymbol{y}$ is the N vector with elements

$$(\boldsymbol{y})_j = \bar{y}(j), \qquad 0 \le j < N, \tag{3.22}$$

respectively, and $\boldsymbol{\beta}$ is the vector of pulse amplitudes with β_j corresponding to the pulse at location j. Since the matrix $\boldsymbol{\alpha}$ is positive semidefinite, there exists a lower triangular matrix $\boldsymbol{L}$ such that $\boldsymbol{\alpha} = \boldsymbol{L}\boldsymbol{L}^t$ and $\boldsymbol{L}$ can be determined by the Cholesky decomposition of $\boldsymbol{\alpha}$ [23]. Equation (3.18) can thus be written as two sets of equations

$$\boldsymbol{L}\boldsymbol{q} = \boldsymbol{c} \tag{3.23}$$

$$\boldsymbol{L}^t\boldsymbol{\beta} = \boldsymbol{q} \tag{3.24}$$

which can be solved[3] for the pulse amplitude vector $\boldsymbol{\beta}$. Here $\boldsymbol{q}$ is an N element vector. The error $E_{\min}^{(N-1)}$ is obtained by the expression

$$E_{\min}^{(N-1)} = \boldsymbol{y}^t\boldsymbol{y} - \boldsymbol{q}^t\boldsymbol{q}. \tag{3.25}$$

We now ask the question: if only m elements in $\boldsymbol{\beta}$ are nonzero, which elements should be retained to minimize the mean-squared error? In other words, we wish to pick m variables in (3.18) such that the resulting system of m equations minimizes the error defined in (3.19). As stated above, the global solution to this problem is computationally complex, but a suboptimal solution can be obtained by starting with one variable, and progressively adding variables such that the reduction in the error is maximized at each step. It is convenient to think of this procedure as a stepwise computation of the matrix $\boldsymbol{L}$ and the vector $\boldsymbol{q}$ [4], [6], [15]. At each step, every possible row/column of $\boldsymbol{\alpha}$ is used to compute the next row of $\boldsymbol{L}$ and the next element of $\boldsymbol{q}$. The row/column that maximizes the reduction in the resulting error from (3.25) is selected as the desired row/column, and the procedure repeated. Equation (3.24) need not be solved during this procedure since the error does not explicitly depend on $\boldsymbol{\beta}$. The pulse locations correspond to the indexes of the rows/columns selected from $\boldsymbol{\alpha}$ and the pulse amplitudes are computed using (3.24) once all pulses have been located.

Let us first assume that $m = 1$, i.e., we are only interested in retaining one of the diagonal elements of $\boldsymbol{\alpha}$, say, α_{ii}. Equation (3.18) then reduces to

$$\alpha_{ii}\beta_i = c_i. \tag{3.26}$$

Using Cholesky decomposition, the first (and only) elements of $\boldsymbol{L}$ and $\boldsymbol{q}$ are

$$l_{00}(i) = \sqrt{\alpha_{ii}}, \qquad q_0(i) = \frac{c_i}{l_{00}(i)}, \tag{3.27}$$

and the resulting error is given by

$$E_{\min}^{(0)}(i) = \boldsymbol{y}^t\boldsymbol{y} - q_0^2(i). \tag{3.28}$$

Again, the variables l_{00}, q_0, and $E_{\min}^{(0)}$ are shown as functions of i to make the dependence on the selected element of $\boldsymbol{\alpha}$ explicit. It is obvious from (3.28) that the error is minimized for i where $q_0^2(i)$ is maximum. Thus, (3.27) is computed for $0 \le i < N$ and the first pulse location n_0 is chosen where $q_0(i)$ has maximum magnitude. The values of $l_{00}(n_0)$ and $q_0(n_0)$ are retained as the first diagonal element of $\boldsymbol{L}$ and the first element of $\boldsymbol{q}$, respectively.

To obtain the second pulse, we need to select a second

[2]To maintain notational consistency, we label array subscripts starting at zero, i.e., the first diagonal element of α is labeled α_{00}.

[3]Note that if we attempt to compute all N elements of $\boldsymbol{\beta}$, the solution may become ill conditioned. This is usually not a problem since we are interested in only $m \ll N$ variables.

row/column from $\boldsymbol{\alpha}$. Once again, we do this by using each possible row/column of $\boldsymbol{\alpha}$ (corresponding to each possible pulse location for the second pulse) to obtain the mean-squared error and selecting the row/column that minimizes this error. The row/column corresponding to n_0 is ignored since it has already been selected.

If we choose the ith row/column from $\boldsymbol{\alpha}$, (3.18) becomes

$$\begin{bmatrix} \alpha_{n_0 n_0} & \alpha_{n_0 i} \\ \alpha_{i n_0} & \alpha_{ii} \end{bmatrix} \begin{bmatrix} \beta_{n_0} \\ \beta_i \end{bmatrix} = \begin{bmatrix} c_{n_0} \\ c_i \end{bmatrix}, \tag{3.29}$$

and $\boldsymbol{L}$ now becomes a 2×2 matrix

$$\boldsymbol{L} = \begin{bmatrix} l_{00}(n_0) & 0 \\ l_{10}(i) & l_{11}(i) \end{bmatrix}. \tag{3.30}$$

Cholesky decomposition of (3.29) results in

$$l_{10}(i) = \frac{\alpha_{in_0}}{l_{00}(n_0)} \qquad l_{11}(i) = \sqrt{\alpha_{ii} - l_{10}^2(i)} \tag{3.31}$$

and

$$q_1(i) = \frac{c_i - l_{10}(i) q_0(n_0)}{l_{11}(i)}. \tag{3.32}$$

Once again, the error is minimized for i where $q_1(i)$ has maximum magnitude. Thus, (3.31) and (3.32) are computed for $0 \le i < N$; $i \ne n_0$, and the second pulse located at $n_1 = i$ for which (3.32) has maximum magnitude.

Similarly, to locate the third pulse, we add a third row and column to $\boldsymbol{L}$ and a third element to $\boldsymbol{q}$. If the third pulse location is i, the elements of the last row of $\boldsymbol{L}$ are

$$l_{20}(i) = \frac{\alpha_{in_0}}{l_{00}(n_0)}, \qquad l_{21}(i) = \frac{\alpha_{in_1} - l_{20}(i) l_{10}(n_1)}{l_{11}(n_1)},$$
$$l_{22}(i) = \sqrt{\alpha_{ii} - l_{20}^2(i) - l_{21}^2(i)}, \tag{3.33}$$

and the last element of $\boldsymbol{q}$ is

$$q_2(i) = \frac{c_i - l_{20}(i) q_0(n_0) - l_{21}(i) q_1(n_1)}{l_{22}(i)}. \tag{3.34}$$

As before, the third pulse location is chosen where $q_2(i)$ has maximum magnitude. Locations corresponding to previously picked pulses are omitted from the search. Note that $l_{20}(i) = l_{10}(i)$ and need not be recomputed. Indeed, it is easy to show that

$$l_{kj}(i) = l_{rj}(i), \quad j < k, r < N, \quad 0 \le i < N;$$
$$i \ne n_0, n_1, \cdots, n_{j-1}, \tag{3.35}$$

and thus, each of these elements has to be computed only once. This process can be repeated for each desired pulse, i.e., to find n_j we compute

$$l_{jj}(i) = \left[\alpha_{ii} - \sum_{k=0}^{j-1} l_{jk}^2(i) \right]^{1/2} \qquad 0 \le i < N;$$
$$i \ne n_0, n_1, \cdots, n_{j-1}, \tag{3.36}$$

and choose $n_j = i$ for which

$$q_j(i) = \frac{c_i - \sum_{k=0}^{j-1} l_{jk}(i) q_k(n_k)}{l_{jj}(i)} \tag{3.37}$$

has the maximum magnitude. The error after the pulse has been located is

$$E_{\min}^{(j)} = E_{\min}^{(j-1)} - q_j^2(n_j). \tag{3.38}$$

If more steps are required, we compute

$$l_{rj}(i) = \frac{\alpha_{in_j} - \sum_{k=0}^{j-1} l_{rk}(i) l_{jk}(n_j)}{l_{jj}(n_j)}, \qquad r = j + 1;$$
$$0 \le i < N; \qquad i \ne n_0, n_1, \cdots, n_j, \tag{3.39}$$

and repeat the process for the next pulse. Once all m pulses are located, the pulse amplitudes are computed using (3.20).

Note that $\boldsymbol{L}$ and $\boldsymbol{q}$ can overwrite $\boldsymbol{A}$ and $\boldsymbol{c}$, respectively. Also, if we keep two running vectors of partial sums

$$g_j(i) = \alpha_{ii} - \sum_{k=0}^{j-1} l_{jk}^2(i), \tag{3.40}$$

and

$$f_j(i) = c_i - \sum_{k=0}^{j-1} l_{jk}(i) q_k(n_k), \tag{3.41}$$

then the pulse search reduces to computing

$$f_j^2(i)/g_j(i) \qquad 0 \le i < N; \qquad i \ne n_0, n_1, \cdots, n_{j-1}, \tag{3.42}$$

and choosing the maximum value. Thus, only one square-root operation is required per pulse when $q_j(n_j)$ is computed after the pulse has been located. The complete algorithm is shown in the Appendix.

1) Comparison to Other Search Procedures: If we compare (3.42) to (3.16), we can make some interesting comments about the optimized amplitude algorithm. In the amplitude reoptimization method, pulses are located at the maxima of the cross-correlation function $c_i^{(j)}$ normalized by the autocorrelation function α_{ii}, and the pulse amplitude β_j is the normalized value of the cross correlation at the maxima. As more pulses are placed, the cross-correlation function $c_i^{(j)}$ is updated, but the previously found amplitudes β_j remain fixed. Similarly, in the optimal amplitude algorithm, pulses are located at the maxima of $f_j(i)$ normalized by $g_j(i)$, and q_j is the normalized value of $f_j(i)$ at the maxima. Once again, as more pulses are computed, $f_j(i)$ is updated and q_j remains fixed. However, unlike the amplitude reoptimization method, where β_j is assumed to remain constant, β_j implicitly varies as more pulses are placed in the optimal amplitude method. The algorithm thus keeps the amplitudes of all previously placed pulses as well as the current pulse optimal during the search for pulse locations. The performance does not suffer for closely spaced pulses in this method.

The real issue is to find an $m \times m$ subsystem of equations from (3.18) that minimizes (3.19) among all such subsystems. Within this framework, many modifications are possible in the algorithm; other efficient procedures to obtain multipulse excitation can be found, and different levels of optimization can be used. In the algorithm described above, only the row/column of $\boldsymbol{\alpha}$ that results in minimum error is retained for the next step. Instead of simply picking the best location at each stage, we could keep the best b locations as candidates and choose the best candidate set after all m stages. It is easily shown that the computation complexity increases only linearly with b. Similarly, we can pick more than one pulse per stage provided that the corresponding elements of $\boldsymbol{\alpha}$ are small implying that the pulse amplitudes are small. Other simplifications [6]–[8], [11]–[18] may be made to reduce complexity of the algorithm. For example, the amplitudes can be updated using Cholesky decomposition, but the pulses located by searching for the maxima of the updated cross-correlation function $c_i^{(j)}$. This is equivalent to assuming that the maxima of $f_j(i)$ determines the pulse location, and $g_j(i)$ remains approximately constant for all i. Similarly, the complexity can be reduced significantly by assuming that the pulses are regularly spaced in the frame. The pulse search then becomes equivalent to searching for the location of the first pulse in the sequence. The reader is directed to the references for more details on these and many other variations of the method.

2) Block Edge Effects: The elements of $\boldsymbol{\alpha}$ are estimates of the autocorrelation α_{ij} of the impulse response of the weighted LPC all-pole filter. However, for values of i or j close to N, these estimates include only a few terms in the sum and the solution of (3.18) can become ill conditioned resulting in artificially high pulse amplitudes for pulses located at the end of the frame [6], [11]. To avoid this problem, the end of each frame is overlapped with the beginning of the next frame, and pulses falling in the overlap region are recomputed in the next frame when the corresponding autocorrelation estimates are better. The length of the overlap region is made large enough to allow the weighted impulse response to decay to a negligible value in this region. In practice, a 2.5 ms overlap region was found adequate to prevent ill-conditioned solutions.

3) Autocorrelation Method: For small frame lengths, the speech signal and the LPC all-pole filter response may be considered stationary and we can approximate $\alpha_{ij} = \alpha_{|i-j|}$ and reduce the computation required. With this approximation, only the first row of $\boldsymbol{\alpha}$ needs to be computed and the entire $\boldsymbol{\alpha}$ matrix need not be kept in storage.

4) Search Pruning: The computation complexity of the multipulse algorithm described above can be reduced still further. We note that the search for a pulse requires only one multiplication and one division operator per location. Because of the exhaustive nature of the search for pulse locations, a large fraction of the computation is devoted to computing (3.39) at each step even though only a few of the elements are finally used to compute $\boldsymbol{\beta}$. If it were possible to eliminate some of the pulse locations from the search at each stage, substantial savings in computation would result. Fig. 3 shows the possible error reduction $q_j^2(i)$ as a function of pulse location for a typical voiced frame. We can see that $q_j^2(i)$ varies smoothly with the pulse location i and has only a few peaks. It is thus possible to find these peaks without computing the entire function.

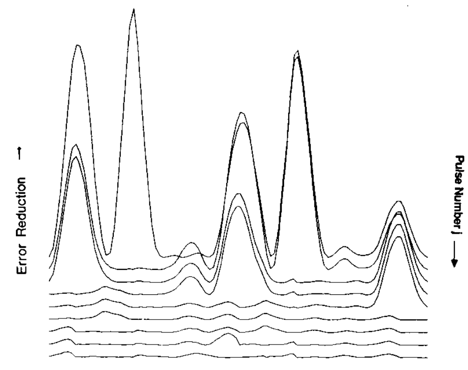

Fig. 3. Error reduction for the jth pulse as a function of the pulse location i for different values of j.

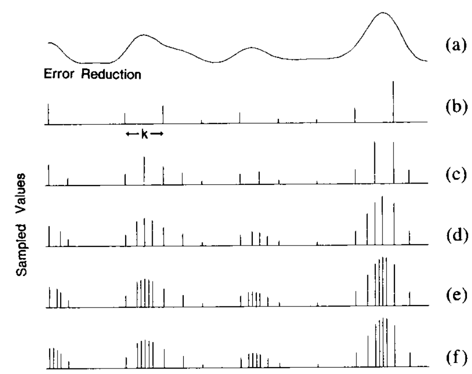

Fig. 4. Search procedure for a pulse location. (a) The error reduction function. (b)–(f) Computed samples of the function in (a) for successive steps.

One way to do this is by sampling the function at successively closer points until all peaks have been found [15]. Fig. 4 illustrates this procedure. First, the error reduction for the jth pulse $q_j^2(i)$ is computed for locations $i = 0, k, 2k, \cdots$. Then for each $i_0 \epsilon I = \{i \mid q_j^2(i) \geq q_j^2(i \pm k)\}$, we compute $q_j^2(i)$ for $i = i_0 + k/2$ and $i = i_0 - k/2$. k is then reduced by half and the process repeated until $k = 1$. At this point, we pick the largest computed value of $q_j^2(i)$ as the peak of the function and the corresponding value of i as the pulse location n_j. Locations that have already been selected are ignored in the procedure.

This procedure typically picks the global maxima in the error reduction function if the initial value of k is less than half the smallest distance between two consecutive peaks. The initial (before any pulses are placed) peaks in the reduction function are located at the extrema of c_i. Thus, we can start with an appropriate value of k, and reduce it as more pulses are placed to ensure that peaks are not missed in the procedure. The optimal amplitude algorithm can easily be modified to do this pruning. The computational complexity of this method is data dependent and it entails a certain overhead associated with searching for the peaks. The pruned algorithm reduces the computation by a factor of 2–4 over the unpruned algorithm with maximum reduction in complexity realized with voiced frames. Since it does not guarantee a global maxima, the excitation computed by the pruned algorithm may differ slightly from that computed by the unpruned algorithm for unvoiced speech, but the observed signal-to-noise ratio remains comparable.

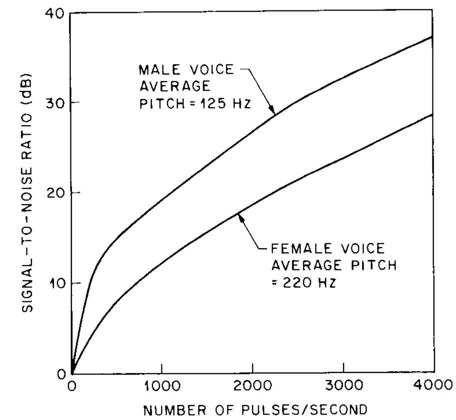

Fig. 5. Average SNR as a function of pulse rate for a female and male speaker.

IV. Voice Periodicity Prediction in Multipulse

It has been our experience that about 8 pulses are needed per pitch period in multipulse excitation for high quality standards. This implies that female speech requires more pulses per frame than male speech for comparable quality. Fig. 5 shows the observed SNR as a function of pulse rate for a segment of voiced speech spoken by two speakers. A difference of about 6–10 dB can be seen for the same pulse rate. Alternately, for the same SNR, female speech may require up to 1000 pulses more per second.

The multipulse model discussed so far did not assume anything about the speech signal; all classes of speech sounds were modeled with the same excitation signal. Our knowledge of the periodic nature of voiced speech was not utilized in the analysis-by-synthesis procedure. For voiced speech, multipulse excitation shows significant correlation from one pitch period to the next. It is possible to use this correlation to reduce the number of pulses in the excitation substantially by pitch prediction on the multipulse excitation [7], [8], [11]–[14], [16]–[18].

The multipulse model shown in Fig. 1 is now modified [8] to include a long delay predictor between the LPC all-pole filter and the excitation generator. Fig. 6 shows the block diagram of such a coder. In the modified model, multipulse excitation becomes the input to a long delay predictor which in turn computes the all-pole filter excitation by adding pulses predicted from past frames of the all-pole filter to the current multipulse excitation pulses. The all-pole filter input $v(n)$ for the current frame is now given by

$$v(n) = u(n) + \sum_{i=-T}^{T} g_i v(n - M + i), \qquad 0 \le n < N, \tag{4.1}$$

where $u(n)$ is the nth sample of the multipulse excitation in the frame, g_i, $-T \le i < T$ are the tap gains of a $2T + 1$ tap predictor, and M is the predictor delay. The predictor delay can be up to several pitch periods and, in

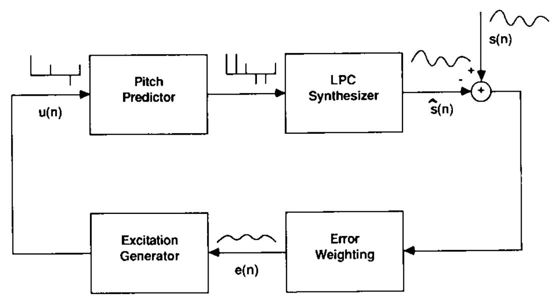

Fig. 6. Multipulse coder with pitch prediction.

general, is longer[4] than the frame length N. The weighted synthetic speech signal $\hat{y}(n)$ can now be expressed as

$$\hat{y}(n) = \sum_{k=0}^{m-1} \beta_k h(n - n_k) + \sum_{i=-T}^{T} g_i \sum_{k=0}^{N-1} h(k) \cdot v(n + i - M - k) + \hat{y}_0(n). \tag{4.2}$$

$\hat{y}(n)$ now consists of three terms: the first due to the multipulse excitation in the current frame, the second due to the excitation obtained by the pitch predictor from past frames, and the third due to the filter memory of the all-pole filter. Again, we wish to minimize the mean-squared error E between the original and synthetic speech signals as defined in (3.6). The last term in (4.2) remains fixed during the minimization and the problem is to obtain the pitch predictor delay M and the tap gains g_i, $-T \le i \le T$, as well as the pulse amplitudes and locations β_i, n_i, $0 \le i < m$.

Minimization with respect to the tap gains g_i results in a linear problem if $M > N - 1 + T$. Minimization with respect to the pitch delay M once again results in a nonlinear problem. Although the pitch predictor and the excitation pulses can be obtained by exhaustively searching for the pulse locations $n_0, \cdots, n_{m-1}$ as well as the pitch delay M, the procedure becomes computationally very expensive and suboptimal solutions have to be used.

One way of reducing the complexity is by obtaining the

[4]We assume $M > N - 1 + T$ in our discussion.

pitch predictor and the multipulse excitation sequentially in two steps [8], [11]. First, we assume that the multipulse excitation is zero and search for a delay value M and the predictor tap gains g_i such that E is minimized. Next, the pitch predictor is held constant and multipulse excitation is computed as described in the last section. Let $\beta_k = 0, 0 \leq k < m$ and

$$z_M(n) = \sum_{k=0}^{N-1} h(k)v(n - M - k), \qquad 0 \leq n < N. \tag{4.3}$$

Minimization of E with respect to g_k results in

$$\sum_{i=-T}^{T} g_i \sum_{n=0}^{N-1} z_M(n + i)z_M(n + k) = \sum_{n=0}^{N-1} \bar{y}(n)z_M(n + k), \qquad -T \leq k \leq T, \tag{4.4}$$

and

$$E_{\min}(M) = \sum_{n=0}^{N} \bar{y}^2(n) - \sum_{i=-T}^{T} g_i \sum_{n=0}^{N-1} \bar{y}(n)z_M(n + i). \tag{4.5}$$

Equations (4.4) are used to solve for the tap gains g_i for all possible values of M, and the resulting error is computed from (4.5). The value of M that results in a minimum error is used as the predictor delay and the corresponding values of g_i as the tap gains. Next, the pitch predictor is held constant, and the multipulse excitation is found as described in the last section. Multipulse analysis thus computes a few additional excitation pulses to generate only that portion of speech which could not be generated by the *a priori* excitation. Hence, multipulse excitation only has to represent the uncorrelated part of the LPC filter excitation. For voiced speech, the excitation is highly correlated, and only a few pulses are needed in the multipulse excitation.

For voiced speech, the predictor delay M is usually some multiple of the pitch period. Instead of obtaining M by an exhaustive search, we can significantly reduce computations by using a pitch estimate obtained from the speech signal as the predictor delay [7]. We do not need the real "pitch" for this procedure, only a delay value where the speech signal shows significant correlation. Thus, complex pitch detection algorithms are not required; a simple correlation detector suffices. The tap gains g_i can then be computed from (4.4), followed by the multipulse excitation.

As mentioned above, computation of the pitch predictor tap gains is a linear problem for a known pitch delay provided $M > N - 1 + T$. Thus, instead of assuming that the tap gains remain fixed during multipulse analysis, we can include the tap gains as variables in the optimal amplitude multipulse analysis procedure [11]. In this case, (3.18) and (3.19) are replaced by

$$\boldsymbol{\alpha}'\boldsymbol{\beta}' = \boldsymbol{c}', \tag{4.6}$$

and

$$E_{\min} = \boldsymbol{y}^t\boldsymbol{y} - (\boldsymbol{\beta}')^t\boldsymbol{c}', \tag{4.7}$$

where

$$\boldsymbol{\alpha}' = \begin{bmatrix} \boldsymbol{\varphi} & \boldsymbol{\psi} \\ \boldsymbol{\psi}^t & \boldsymbol{\alpha} \end{bmatrix}, \quad \boldsymbol{\beta}' = \begin{bmatrix} \boldsymbol{g} \\ \boldsymbol{\beta} \end{bmatrix}, \quad \boldsymbol{c}' = \begin{bmatrix} \boldsymbol{\xi} \\ \boldsymbol{c} \end{bmatrix}. \tag{4.8}$$

Here $\boldsymbol{\alpha}$ and $\boldsymbol{c}$ are as defined in (3.18) and (3.19), respectively, $\boldsymbol{\varphi}$ is a $2T + 1 \times 2T + 1$ matrix with elements

$$\phi_{ij} = \sum_{n=0}^{N-1} z_M(n + i - T)z_M(n + j - T), \qquad 0 \leq i, j \leq 2T, \tag{4.9}$$

$\boldsymbol{\psi}$ is an $N \times 2T + 1$ matrix with elements

$$\psi_{ij} = \sum_{n=0}^{N-1} h(n - i)z_M(n + j - T), \qquad 0 \leq i < N, \quad 0 \leq j \leq 2T, \tag{4.10}$$

$\boldsymbol{\xi}$ is a $2T + 1$ element vector with elements

$$\xi_j = \sum_{n=0}^{N-1} \bar{y}(n)z_M(n + j - T), \qquad 0 \leq j \leq 2T, \tag{4.11}$$

and $\boldsymbol{g}$ is the vector of $2T + 1$ tap gains. During the solution of (4.6), the tap gains are computed first, then the multipulse excitation is computed as before.

Unlike LPC vocoders, pitch errors do not significantly degrade the performance of multipulse coders. The aim here is not to obtain the real "pitch," but simply a delay value for which the synthetic speech generated from the past excitation is highly correlated with the current samples of speech. Thus, there are many ways of taking voice periodicity into account in multipulse analysis. Although the pitch predictor can be found by an exhaustive search for the delay, the procedure becomes computationally impractical. Computation can be reduced by computing the pitch predictor and the excitation in two steps: first, the predictor is found, assuming the excitation remains zero; next, the excitation is found for that pitch predictor. Computation can be reduced significantly if the pitch delay is obtained directly from the speech signal rather than through a location-by-location search. Once the pitch delay is known, it is also possible to compute the predictor taps along with the excitation in the multipulse procedure. It is also unnecessary to make a voiced/unvoiced decision and the pitch predictor can be computed for all speech sounds. For unvoiced speech, the tap gains are small and mapped to zero during quantization. Even if some tap gains remain nonzero, their influence is small.

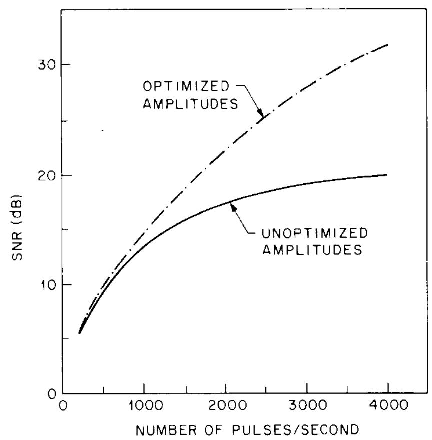

Fig. 7. Average SNR as a function of pulse rate for multipulse coders.

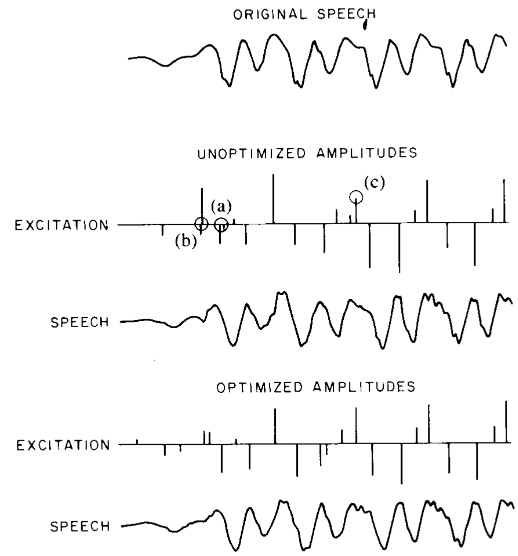

Fig. 8. Illustration of the differences in the excitation obtained with and without amplitude optimization. (a) A superfluous pulse caused by amplitude inaccuracy. (b) A location error. (c) An irregular pitch pulse.

V. Results

A. Results Using Amplitude Optimization

Fig. 7 shows the signal-to-noise ratio (SNR) obtained as a function of the pulse rate using the amplitude reoptimization method and the optimal amplitude method without pitch prediction for the sentence "Both teams started from zero." The sentence, spoken by a female speaker, was digitized at 8 kHz and analyzed in 12.5 ms wide frames where the last 2.5 ms of each frame was overlapped with the next frame. The excitation pulses falling in the overlap region at the end of the frames were discarded and recomputed in the next frame to avoid block edge effects (see Section III-B-2). If the pulse amplitudes are not optimized, the SNR saturates after a few pulses have been placed per frame (Fig. 7). The saturation effect is not observed for the optimized amplitude method. If amplitudes are reoptimized after each pulse is found, the SNR falls between the two extremes. It should be noted that the gain obtained by amplitude optimization is very small when the pulse rate is small. The figure is representative of the bounds on the performance of different algorithms since the parameters are unquantized. The advantage of amplitude optimization may be reduced significantly even for higher pulse rates if the parameters are quantized subsequent to optimization [18].

Fig. 8 illustrates the differences in the excitation obtained with and without amplitude optimization. A small section of speech (50 ms long) is shown in the figure along with the excitation obtained with the two methods. The synthetic speech signals are also shown. It can be seen that amplitude optimization removes some superfluous pulses and corrects some pulse locations.

Fig. 9 shows the signal energy and the SNR obtained with and without amplitude optimization for a sentence at a pulse rate of 1600 pulses/s. The improvement is consistent throughout the utterance and not limited to any one section of speech. The degree of improvement varies from 2 to 10 dB with an average of slightly over 3 dB.

B. Results Using Pitch Prediction

Fig. 10 illustrates the improvement obtained by a single tap pitch predictor for a segment of speech uttered by a female speaker. The improvement varies from about 2 dB to 5 dB. A short segment of speech is shown in Fig. 11 along with the multipulse excitation, the all-pole filter excitation, and the synthetic speech. Note that with pitch

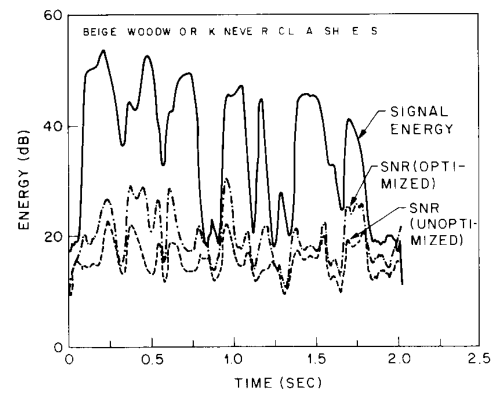

Fig. 9. The signal energy and the SNR obtained with and without amplitude optimization for the utterance "Beige woodwork never clashes."

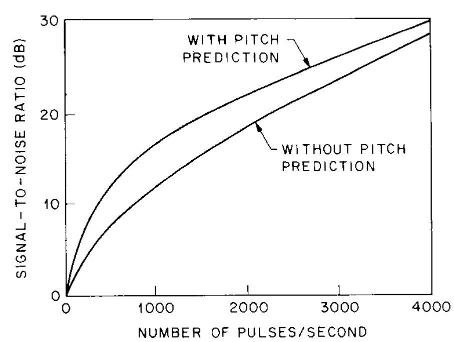

Fig. 10. Average SNR as a function of pulse rate with and without pitch prediction of the excitation for a 1 s segment of speech by a female speaker.

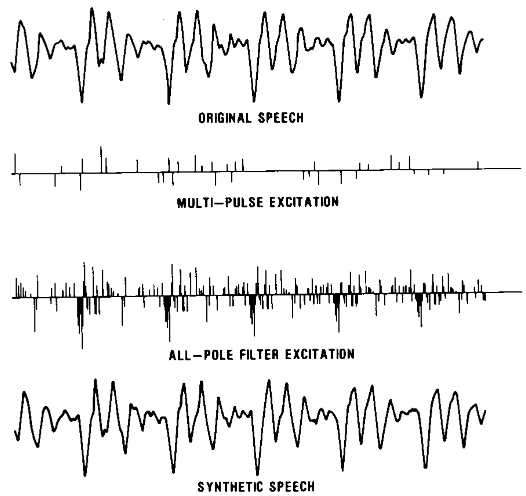

Fig. 11. The original speech, multipulse excitation, all-pole-filter excitation, and the synthetic speech obtained using the pitch loop.

prediction, multipulse excitation is much less periodic than the all-pole filter excitation.

VI. Discussion

Multipulse excitation is a promising way of encoding speech at medium bit rates (9.6–16 kbits/s). At these bit rates, coders such as LPC vocoders do not obtain the required speech quality because of the simplistic excitation model used in such coders. Waveform coders such as APC coders also do not perform well in this range. Multipulse coders offer better performance because the flexible excitation model allows waveform coding of the speech signal with only a few pulses per frame. In this paper we reviewed the basic multipulse model and discussed a procedure to compute the excitation with optimally adjusted amplitudes. The algorithm provides a framework for computing multipulse excitation with varying degrees of optimization and computational complexity. We also discussed some methods of improving the performance of these coders, especially for high pitched voices, by including a long delay "pitch" predictor in the coder.

Appendix

The algorithm to compute optimized multipulse excitation follows.

1) Compute

$$c_i = \sum_{n=i}^{N-1} y(n)h(n-i), \qquad 0 \le i < N,$$

$$\alpha_i = \sum_{n=i}^{N-1} h(n)h(n-i), \qquad 0 \le i < N.$$

2) Set

$$p(i) = i, \qquad 1 \le i \le N,$$

$$f(i) = c_{i-1}, \qquad 1 \le i \le N,$$

$$g(i) = \alpha_0, \qquad 1 \le i \le N.$$

3) Choose $n_1 = i$ such that $f^2(i)$ is maximum.

4) Interchange $p(n_1)$ with $p(1)$.

5) Set

$$g(n_1) = \sqrt{g(n_1)},$$

$$q_1 = f(n_1)/g(n_1),$$

$$E_1 = \sum_{n=0}^{N-1} y^2(n) - q_1^2,$$

$$j = 1.$$

6) $r = p(j)$; for $j < i \le N$ compute

$$k = p(i).$$

$$l_{kj} = \frac{\alpha(|k-r|) - \sum_{n=1}^{j-1} l_{kn} l_{rn}}{g(r)}.$$

7) For $j < i \le N$, compute

$$k = p(i)$$

$$g(k) = g(k) - l_{kj}^2$$

$$f(k) = f(k) - l_{kj} q_j.$$

8) $j = j + 1$; choose $n_j = i$ such that $f^2(p(i))/g(p(i))$ is maximized.

9) Interchange $p(j)$ with $p(n_j)$.

10) Set

$$g(n_j) = \sqrt{g(n_j)}$$

$$q_j = f(n_j)/g(n_j)$$

$$E_j = E_{j-1} - q_j^2.$$

11) Repeat steps 6–10 until enough pulses have been found. Let the number of pulses found be m.

12) Set $j = m$; while $j > 0$ compute

$$\beta_j = \frac{q_j - \sum_{k=j+1}^{m} l_{p(k)j}\beta_k}{g(p(j))}$$

$$n_j = p(j) - 1; \qquad j = j - 1.$$

Acknowledgment

We would like to thank one of the reviewers of this paper for many helpful suggestions and comments.

References

[1] B. S. Atal and J. R. Remde, "A new model of LPC excitation for producing natural-sounding speech at low bit rates," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, Paris, France, 1980, pp. 614–617.

[2] B. S. Atal, "Linear predictive coding of speech," in *Computer Speech Processing*, F. Fallside and W. A. Woods, Eds. Englewood Cliffs, NJ: Prentice-Hall, 1985.

[3] S. Singhal and B. S. Atal, "Optimizing LPC filter parameters for multi-pulse excitation," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, Boston, MA, 1983, pp. 781–784.

[4] S. Singhal, "Optimizing pulse amplitudes in multi-pulse excitation," *J. Acoust. Soc. Amer. Suppl. 1*, vol. 74, p. S51, Fall 1983.

[5] P. Kroon and E. F. Deprettere, "On the design of LPC vocoders with multi-pulse excitation," in *Proc. Euro. Conf. Circuit Theory, Design*, 1983, pp. 390-394.
[6] M. Berouti, H. Garten, P. Kabal, and P. Mermelstein, "Efficient computation and encoding of the multipulse excitation for LPC," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, San Diego, CA, 1984, Paper 10.1.
[7] P. Kroon and E. F. Deprettere, "Experimental evaluation of different approaches to the multi-pulse coder," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, San Diego, CA, 1984, Paper 10.4.
[8] S. Singhal and B. S. Atal, "Improving performance of multi-pulse coders at low bit rates," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, San Diego, CA, 1984, Paper 1.3.
[9] I. M. Trancoso, R. Garcia-Gomez, and J. M. Tribolet, "A study on short-time phase and multipulse LPC," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, San Diego, CA, 1984, Paper 10.3.
[10] I. M. Trancoso, L. B. Almeida, and J. M. Tribolet, "Pole-zero multipulse speech representation using harmonic modelling in the frequency domain," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, Tampa, FL, 1985, pp. 260-263.
[11] J. Lefevre and O. Passien, "Efficient algorithms for obtaining multipulse excitation for LPC coders," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, Tampa, FL, 1985, pp. 957-960.
[12] Y. Wake, S. Tanaka, K. Ozawa, and T. Araseki, "A multi-pulse LPC codec using digital signal processors," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, Tampa, FL, 1985, pp. 1429-1432.
[13] H. Koyama and A. Gersho, "Fully vector-quantized multipulse LPC at 4800 bps," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, Tokyo, Japan, 1986, pp. 445-448.
[14] K. Ozawa and T. Araseki, "Low bit rate multi-pulse speech coder with natural speech quality," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, Tokyo, Japan, 1986, pp. 457-460.
[15] S. Singhal, "Reducing computation in optimal amplitude multipulse coders," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, Tokyo, Japan, 1986, pp. 2363-2366.
[16] E. F. Deprettere and P. Kroon, "Regular excitation reduction for effective and efficient LP-coding of speech," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, Tampa, FL, 1985, pp. 965-968.
[17] P. Kroon, E. F. Deprettere, and R. Sluyter, "Regular-pulse excitation—A novel approach to effective and efficient multipulse coding of speech," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 1054-1063, 1986.
[18] P. Kroon, "Time-domain coding of (near) toll quality speech at rates below 16 kb/s," Ph.D. dissertation, Dep. Elec. Eng., Delft Univ. Technol., Delft, The Netherlands.
[19] A. Ichikawa, S. Takeda, and Y. Asakawa, "A speech coding method using thinned out residual," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, Tampa, FL, 1985, pp. 961-964.
[20] B. S. Atal and M. R. Schroeder, "Predictive coding of speech signals and subjective error criteria," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-27, pp. 247-254, 1979.
[21] M. J. Garside, "Some computational procedures for the best subset problem," *Appl. Stat.*, vol. 20, pp. 8-15, 1971.
[22] G. M. Furnival and R. W. Wilson, "Regressions by leaps and bounds," *Technometrics*, vol. 16, no. 4, pp. 499-511, 1974.
[23] A. Ralston, *A First Course in Numerical Analysis*. New York: McGraw-Hill, 1965.

Sharad Singhal (S'78-M'82) received the B.S. degree in electrical engineering from the Indian Institute of Technology, Kanpur, India, in 1977, and the M.S. and Ph.D. degrees from Yale University, New Haven, CT, in 1978 and 1982, respectively, both in electrical engineering.

In 1982 he joined the Acoustics Research Department at AT&T Bell Laboratories, Murray Hill, NJ, where he worked on speech coding. He joined Bell Communications Research, Inc. in 1984, where he is a member of the Speech and Image Processing Research Division. His primary research interests are in the areas of speech analysis and coding, speech recognition, and digital signal processing.

Dr. Singhal is a member of the Acoustical Society of America.

Bishnu S. Atal (M'76-SM'78—F'82) was born in Kanpur, India, on May 10, 1933. He received the B.Sc.(honors) degree in physics from the University of Lucknow, India, in 1952, the Diploma in electrical communication engineering from the Indian Institute of Science, Bangalore, India, in 1955, and the Ph.D. degree in electrical engineering from the Polytechnic Institute of Brooklyn, NY, in 1968.

From 1957 to 1960 he was a Lecturer in Acoustics at the Department of Electrical Communication Engineering, Indian Institute of Bangalore. In 1961 he came to the United States to join the Acoustics Research Department of Bell Telephone Laboratories, Murray Hill, NJ. He is the Head of the Acoustics Research Department at AT&T Bell Laboratories. At Bell Laboratories, his work has covered a wide range of topics in acoustics and speech. He is well known for his work on linear predictive coding of speech signals. He holds many patents in the fields of speech coding, mobile radio communication, and stereophonic sound reproduction. He has also published widely in architectural acoustics and speech processing.

Dr. Atal is a Fellow of the Acoustical Society of America and a member of the National Academy of Engineering. He was awarded the Technical Achievement Award of the IEEE ASSP Society in 1975. In 1980 he received, jointly with M. R. Schroeder, the IEEE ASSP Senior Award for a paper on predictive coding of speech signals and subjective error criteria. He is the recipient of the IEEE Cenntenial Medal in 1984, and the IEEE Morris N. Liebman Memorial Field Award in 1986 for his pioneering contributions to linear predictive coding for speech processing.