Speech Data Compression Through Sparse Coding of Innovations
Tenkasi V. Ramabadrnan, Member, IEEE, and Deepen Sinha, Senior Member, IEEE

Abstract—A new scheme for coding speech at low bit rates (4.8–16 kb/s) but still maintaining high quality is described. Speech is regarded as a piecewise-stationary random signal and its synthesis is accomplished by means of a Kalman estimator at the decoder. The Kalman estimator requires for its operation a signal model and a sequence of measurements of the states of the model. A two-stage, time-varying, all-pole filter excited by white noise is used as the speech signal model. Linear combinations of speech samples taken at sparse but periodic intervals and provided in the form of innovations serve as measurements. The role of the encoder in the proposed scheme is seen as that of extracting the signal model parameters as well as forming the measurements and transmitting this information to the decoder. An optimum measurement strategy is developed for the estimator. A procedure for shaping the error spectrum of the synthesized speech is also described. Simulation studies show that coders based on the proposed scheme can provide high-quality speech at low bit rates. Important implementation details of such coders as well as their performance results for different choices of coder parameters are given.

I. INTRODUCTION

Coding speech at low bit rates (4.8–16 kb/s) but still maintaining high quality is of considerable interest in current speech research with potential application to satellite communication, cellular-radio communication, secure communication for both military and business, voice storage, voice response, and voice mail [1]. Among the different methods that have been proposed to achieve this end, perhaps the most successful ones belong to the class of linear predictive (LP) coders. These coders [2]–[4] employ the well-known excitation-modulation model of speech production [5]. The modulation filter typically consists of two stages: a short-delay filter modeling the spectral envelope of speech and a long-delay filter modeling the spectral fine structure. Both are time-varying, all-pole filters and are derived from the original speech through LP analysis. The excitation signal in these coders is usually selected by means of an analysis-by-synthesis procedure whereby a perceptually weighted error criterion is minimized. Depending on the particular method, the excitation signal takes different forms as follows. In the multipulse linear predictive (MPLP) coder [2], it is a sequence of appropriately located and scaled impulses. In the code-excited linear predictive (CELP) coder [3], it is a properly chosen entry from a codebook consisting, e.g., of white, Gaussian sequences. In the self-excited vocoder (SEV) [4], the selection of the excitation signal is made from the past history of the source excitation itself. High-quality, natural-sounding speech can be produced by these LP coders at low bit rates. In particular, extensive research has led the CELP coder to emerge as the standard for several different applications [6]–[8].

In this paper, we describe a new scheme for coding speech at low bit rates but still maintaining high quality. In this scheme, we regard speech as a piecewise-stationary random signal and treat its synthesis (or decoding) operation as an estimation procedure for this signal. Accordingly, we employ the Kalman estimator [9]–[12] for this purpose, which is known to provide (linear) optimal estimates, i.e., estimates with minimum mean-squared error. For its operation, the Kalman estimator requires a signal model described using state-space notation and a sequence of measurements of the states of the model. A two-stage, time-varying, all-pole filter excited by white noise is used as the speech signal model. In the state-space description of this model, speech samples at different time instants serve as states. Linear combinations of states (i.e., speech samples) taken at sparse but periodic intervals and provided in the form of innovations serve as measurements. The job of the encoder in the new scheme is seen as that of providing the necessary information, viz., the parameters of the signal model and the sparse measurements, to the decoder that then uses the information to synthesize speech by means of the Kalman estimator.

In a sense, the proposed coder can be viewed as belonging to the class of LP coders. It uses a two-stage, time-varying, all-pole filter derived from the original speech through LP analysis as the modulation filter. However, no excitation-signal information is provided except that it is assumed white and for an estimate of its variance. Instead, linear combinations of the differences between actual and predicted speech samples (i.e., innovations) are provided at sparse but periodic intervals. Yet another way of interpreting the proposed coder is as a vector-DPCM (differential pulse code modulation) coder [13]. A block of speech samples is first predicted and the difference between actual and predicted speech blocks (i.e., the error vector) is formed. However, instead of using a vector quantization (VQ) technique to quantize the error vector as in [13], the length of a projection of the error vector is scalar-quantized and transmitted to the decoder, which then uses it optimally to synthesize speech.

Manuscript received January 17, 1993; revised June 14, 1993. The associate editor coordinating the review of this paper and approving its publication was Dr. W. Bastian Klein. This work was supported through a Faculty Development Program at Iowa State University sponsored by the 3M Foundation of St. Paul, MN.

T. V. Ramabadrnan is with the Department of Electrical Engineering and Computer Engineering, Iowa State University, Ames, IA 50011.
D. Sinha is with AT&T Bell Labs, Murray Hill, NJ 07974.
IEEE Log Number 9215233.
In prior work, e.g., [14]–[20], the use of the Kalman estimator has been investigated extensively for speech-coding applications. In [14], Gunn and Sage propose a Kalman filter model for speech synthesis and discuss its potential advantages over conventional transversal filter models although they do not process any real speech. Pirani and Scaglioni [15] provide results of processing real speech in a DPCM system employing a Kalman filter predictor; they use several time-invariant, pole-zero models to represent the average vocal-tract behavior. In the survey paper [16], Gibson discusses the application of the Kalman filter for both parameter estimation and noise filtering in speech-waveform coders. The use of Kalman-type algorithms for filtering quantization noise in DPCM systems is considered in [17] by the same author. Gibson, et al. [18] and Gibson [19] investigate the use of sequential estimation algorithms such as the Kalman filter for parameter estimation in ADPCM systems with backward adaptive predictors. In [20], Asmuth and Gibson describe the use of the Kalman algorithm for noise spectral shaping in ADPCM systems.

Compared to prior work on signal filtering using the Kalman estimator, e.g., [14]–[17], the present work differs in the following respects: 1) the innovations are not coded at each time instant; instead they are coded at sparse but periodic intervals, the motivation being that it provides a way to achieve lower bit rates. For this situation, an optimum measurement strategy for the Kalman estimator is also developed in this work, and 2) both the long-delay and short-delay filters are incorporated in the speech signal model. This results in a more accurate model even though the complexity is greatly increased.

The objective of this paper is to describe the sparse-innovations coding approach to speech data compression and to present the results of our investigation using this approach in coding speech at low bit rates. Being at an early stage of development of a new approach, important implementation issues such as complexity, delay, and word length are not given due weight. Furthermore, simplifying assumptions are made in the formulation of the estimation problem, viz., the process and measurement noise sequences are assumed to be white and mutually uncorrelated, which may not be justifiable under all situations.

The organization of this paper is as follows. In Section II, we briefly describe the Kalman estimation algorithm. The proposed sparse-innovations coding approach using the Kalman estimator is then presented in Section III. In Section IV, some implementation details of coders based on the new scheme are discussed. Several simulation results indicative of the performance of the new scheme are next presented in Section V. Our conclusions are summarized in Section VI.

II. THE KALMAN ESTIMATOR

The theory and applications of the Kalman estimation algorithm have been extensively covered in the literature [9]–[12]. In this section, we present the main elements of the discrete-time version of the algorithm [11], [12]. Let a random process to be estimated and the measurements of the process be modeled, respectively, as

\[ x_{k+1} = \Phi_k x_k + w_k \]  

and

\[ z_k = H_k x_k + v_k \]

where

- \( x_k \) is a \((n \times 1)\) process state vector at time \( t_k \);
- \( \Phi_k \) is a \((n \times n)\) state transition matrix at time \( t_k \) relating \( x_k \) to \( x_{k+1} \);
- \( w_k \) is a \((n \times 1)\) process noise vector at time \( t_k \);
- \( z_k \) is a scalar measurement at time \( t_k \);
- \( H_k \) is a \((1 \times n)\) measurement vector at time \( t_k \) relating to \( x_k \) to \( z_k \);
- \( v_k \) is a scalar measurement noise at time \( t_k \).

The process noise vector \( w_k \) and the scalar measurement noise \( v_k \) are assumed to be zero mean, white (time-uncorrelated) noise sequences with known covariance structures given by \( E[w_k w_k^T] = G_k \delta_{ik} \) and \( E[v_k v_k] = R_k \delta_{ik} \), where \( E[\cdot] \) denotes the expectation operator, the superscript \( T \) indicates a vector transpose, and \( \delta_{ik} \) is the Kronecker delta function. It is also assumed that \( w_k \) and \( v_k \) are uncorrelated, and both \( w_k \) and \( v_k \) are uncorrelated with the initial state vector \( x_0 \), i.e., \( E[w_k v_k] = 0 \), \( E[w_k x_0^T] = 0 \), and \( E[v_k x_0^T] = 0 \).

Let \( \hat{x}_k \) and \( \hat{x}_0 \) denote, respectively, the a priori (or one-step predicted) and the a posteriori (or filtered) estimates of \( x_k \). These estimates will be unbiased if \( \hat{x}_0 \) is chosen as \( E[x_0] \), which we assume to be the case henceforth. Corresponding to the two estimates, the estimation errors, viz., the a priori error and a posteriori error, are, respectively, given by \( e_k = x_k - \hat{x}_k \) and \( \xi_k = x_k - \hat{x}_k \). Since the estimates are assumed to be unbiased, the estimation errors have zero mean and we can define the corresponding error covariance matrices as \( P_k^- = E[e_k e_k^T] \) and \( P_k = E[\xi_k \xi_k^T] \). Typically, \( P_k^- \) is taken to be \( E[(x_0 - E[x_0])(x_0 - E[x_0])^T] \), if known.

The recursive equations defining the Kalman estimator are given below.

\[ \xi_k = z_k - H_k \hat{x}_k^- \]  

\[ E_k = H_k P_k^- H_k^T + R_k \]  

\[ K_k = P_k H_k^T E_k^{-1} \]  

\[ \hat{x}_k = \hat{x}_k^- + K_k \xi_k \]  

\[ P_k = (I - K_k H_k) P_k^- \]  

\[ \hat{x}_{k+1} = \Phi_k \hat{x}_k \]  

\[ P_{k+1}^- = \Phi_k P_k \Phi_k^T + G_k \cdot P_{0}^- \]  

The scalar quantity \( \xi_k \) in (3), which is formed as the difference between the measurement \( z_k \) and its predicted estimate \( H_k \hat{x}_k^- \), is referred to as the innovation and its variance is given by \( E_k \) in (4). The \((n \times 1)\) matrix \( K_k \) in (5) is referred to as the Kalman filter gain matrix. The above equations can be used recursively to estimate the state vector \( x_k \), \( k \geq 0 \) based on the measurements \( z_k \), \( k \geq 0 \), provided the system matrices \( \Phi_k \) and \( H_k \) as well as the noise covariance matrices \( G_k \) and \( R_k \) are known and suitable values are assumed for \( \hat{x}_0 \) and \( P_0^- \). Given
the model in (1) and (2), the Kalman estimator described by (3) through (9) is optimal in the sense that the trace of $P_k$ as well as that of $P_k^*$ is minimized.

The Kalman filter also provides optimum estimates of any signal that can be expressed as a linear combination of the states. Suppose $y_k$ represents a scalar output of the random process and is modeled as

$$ y_k = C_k x_k \tag{10} $$

where $C_k$ is a $(1 \times n)$ output vector relating $y_k$ to $x_k$. Then, the estimate $\hat{y}_k = C_k \hat{x}_k$ based on the measurements $z_0$ through $z_k$ is optimal in the sense that the variance of the estimation error $E[(y_k - \hat{y}_k)^2]$ is minimized.

III. PROPOSED CODING SCHEME

As mentioned in Section I, the decoder in the proposed scheme synthesizes speech using the Kalman state estimation algorithm. In order to do this, we need a model for the speech process and a sequence of measurements of the process (Ref. (1) and (2)). In this section, we first present a speech signal model and describe it using state-space notation. Then, we introduce the sparse coding approach for measurements.

An optimum measurement model for the Kalman estimator is next derived. The encoder and decoder structures of the proposed scheme are then presented. A method for shaping the reconstruction error spectrum is outlined next. Finally, the relationships between the proposed scheme and some of the other known coding schemes are pointed out.

A. Speech Process Model

A two-stage (short-delay and long-delay), time-varying, all-pole filter excited by white noise is used as the speech signal model in the proposed scheme. The parameters of the model are obtained through linear predictive analysis of the original speech. Such a model is implicit in several LP coders, especially the CELP coder. The justification for the model comes from the fact that the prediction residual (after both short-delay and long-delay prediction) has been found to be approximately white (and Gaussian) for steady speech sounds [3], [21], [22].

The speech signal model involving both short-delay and long-delay predictors is shown in Fig. 1(a) in which the quantities $r(k), d(k),$ and $s(k)$ denote, respectively, the white noise input, the output of the long-delay filter, and the speech signal output at time $t_k$. In $Z$-transform notation, the system function of the short-delay predictor can be expressed as

$$ P_s(z) = \sum_{i=1}^{p} \alpha_i^{(k)} z^{-i} \tag{11} $$

where $\alpha_i^{(k)}, i = 1, 2, \cdots, p$ denote the short-delay predictor coefficients. The superscript $(k)$ indicates that these coefficients are time-varying. Similarly, the system function of the long-delay predictor can be expressed as

$$ P_l(z) = \sum_{j=M(k)}^{M(k)+q-1} \beta_j^{(k)} z^{-j} \tag{12} $$

where $\beta_j^{(k)}, j = 1, 2, \cdots, q$ (odd) denote the long-delay predictor coefficients and $M(k) + (q - 1)/2$ is the pitch period. The time-varying parameters $\alpha_i^{(k)}, \beta_j^{(k)},$ and $M^{(k)}$ are computed by the encoder for consecutive segments of original speech through LP analysis, quantized, and transmitted to the decoder.

To obtain a state-space description of the speech signal model, it is convenient to combine the short-delay and long-delay predictors into a single predictor as shown in Fig. 1(b). The system function of the combined predictor can be expressed as

$$ P_c(z) = \sum_{m=1}^{M^{(k)}+p+q-1} \gamma_m^{(k)} z^{-m} \tag{13} $$

where $\gamma_m^{(k)}, m = 1, 2, \cdots, M^{(k)} + p + q - 1$ denote the time-varying coefficients of the combined predictor. The relationship between the coefficients $\gamma_m^{(k)}$ and the coefficients $\alpha_i^{(k)}$ and $\beta_j^{(k)}$ can be found from the easily-derived expression

$$ s(k) = r(k) + \sum_{i=1}^{p} \alpha_i^{(k)} s(k-i) + \sum_{j=M^{(k)}}^{M^{(k)}+q-1} \beta_j^{(k)} s(k-j) \tag{14} $$

Notice that out of the $M^{(k)} + p + q - 1$ coefficients, only the first $p$ and the last $p + q$ coefficients are nonzero, assuming that $M^{(k)} > p$, which is the usual case. Using the relationship implied in (14), the combined predictor coefficients $\gamma_m^{(k)}, m = 1, 2, \cdots, M^{(k)} + p + q - 1$ can be obtained from the short-delay and long-delay predictor coefficients.

Letting $R(z)$ and $S(z)$ denote, respectively, the $Z$-transforms of $r(k)$ and $s(k)$, the system function of the speech
signal model shown in Fig. 1(b) can be written as

\[
H(z) = \frac{S(z)}{R(z)} = \frac{1}{1 - P_c(z)} = \frac{1}{M^{(k)} + 1 + \sum_{m=1}^{M^{(k)}+p+q-1} \gamma_m^{(k)} z^{-m}}.
\]

(15)

The system function \(H(z)\) can be easily realized in state-space form [14]. Using Type 1 realization [23], the speech process can now be expressed as shown in (1), i.e.

\[
x_{k+1} = \Phi_k x_k + w_k
\]

where we have the equations that appear at the bottom of the page. The size \(n\) of the state vector in the above equation is clearly \(M^{(k)} + p + q - 1\). Notice that the states at time instant \(t_k\) are just the speech samples at the previous \(M^{(k)} + p + q - 1\) instants. The state transition matrix \(\Phi_k\) has a simple structure. The first row of \(\Phi_k\) is made up of the (time-varying) combined predictor coefficients with only \(2p + q\) nonzero entries. The last column of \(\Phi_k\) has only one nonzero entry, viz., \(\gamma^{(k)}_{M^{(k)}+p+q-1}\). If the first row and the last column are deleted, \(\Phi_k\) consists of only an identity matrix. The process noise vector \(w_k\) also has a simple structure with only one nonzero entry. In fact, both \(\Phi_k\) and \(w_k\) are in the so called controllable canonical form [24]. The speech signal itself can now be regarded as an output of the above process, i.e., as a linear combination of the states. For example, the speech signal with a single time delay and the signal with \((M^{(k)} + p + q - 1)\) time delays are, respectively, given by

\[
s(k-1) = [1 \ 0 \ 0 \ \ldots \ 0] x_k
\]

(16a)

\[
s(k - [M^{(k)} + p + q - 1]) = [0 \ \ldots \ 0 \ 0 \ 0] x_k.
\]

(16b)

B. Sparse Coding of Measurements

In addition to the speech process model, a sequence of measurements of the process is required at the decoder for synthesizing speech through the Kalman estimator. As expressed by (2), a scalar measurement at any time instant \(t_k\) is obtained essentially as a linear combination of the states, i.e., past speech samples, at that instant. Since actual speech samples are available at the encoder, it can easily form the necessary measurements, quantize, and transmit them to the decoder. The quantization error can be conveniently modeled as the measurement noise term \(v_k\). To reduce the number of bits needed to code the measurements, a sparse coding approach is employed in the proposed scheme, i.e., the measurements are coded at every \(L\)th time instant, where \(L > 1\), instead of every time instant. The effect of these sparse measurements on the estimation algorithm is that (8) and (9) (collectively referred to as the “project ahead” step [11]) are executed \(L\) times before proceeding to the next step in the recursion. That is, the \textit{a priori} estimate \(x_k\) is now a \(L\)-step predicted estimate instead of a one-step predicted estimate, and \(P_{x_k}^n\) is the corresponding error covariance matrix. The new measurement at every \(L\)th time instant is used as before to update the \textit{a priori} estimate into an \textit{a posteriori} estimate. Clearly, the choice of \(L\) can be expected to have a significant impact on the quality of estimates. Interestingly, the formation of the \(L\)-step predicted estimates in the proposed scheme can be related to the “vector prediction” employed in a vector-DPCM coder [13] and to the formation of the “zero-input response” in a LD-CELP coder [7].

C. Optimum Measurement Model

Unlike traditional applications of Kalman filtering [10], in which all the states of the process are in general not accessible, the states of the speech process that are past speech samples are all available at the encoder. This means that any linear combination of the states can be used as a measurement at a given time instant. In the proposed scheme, this degree of freedom is used to select the optimum measurement vector \(H_k\) every time a measurement is formed, i.e., at every \(L\)th time instant. A solution to the optimum measurement selection problem is given below.

Combining (4), (5), and (7), we can express \(P_k\) as a function of \(H_k\) as follows.

\[
P_k = P_k^\ast - P_k^\ast H_k^T (H_k P_k^\ast H_k^T + R_k)^{-1} H_k P_k^\ast.
\]

(17)
If we use the same notion of optimality as employed in the Kalman estimation algorithm, the optimum measurement vector \( \mathbf{H}_k \) is seen to be the one that minimizes the trace of \( \mathbf{P}_k \). Equivalently, the trace of the second term in (17) can be maximized. Realizing that \( (\mathbf{H}_k \mathbf{P}_k^{-1} \mathbf{H}_k^T + R_k) \) is a scalar and \( \mathbf{P}_k^{-1} \) is symmetric, the quantity to be maximized can be written as

\[
\text{tr}[\mathbf{P}_k^{-1} (\mathbf{H}_k \mathbf{P}_k^{-1} \mathbf{H}_k^T + R_k) \mathbf{H}_k \mathbf{P}_k^{-1} \mathbf{H}_k^T]
\]

\[
= \text{tr}[(\mathbf{H}_k \mathbf{P}_k^{-1} \mathbf{H}_k^T)^T (\mathbf{H}_k \mathbf{P}_k^{-1} \mathbf{H}_k^T) + R_k] = \mathbf{H}_k \mathbf{P}_k^{-1} \mathbf{H}_k^T + R_k.
\]

We now make the assumption to be justified shortly that \( R_k = \delta (\mathbf{H}_k \mathbf{P}_k^{-1} \mathbf{H}_k^T) \), where \( \delta \geq 0 \) is a scalar constant. Incorporating this assumption, the quantity to be maximized becomes

\[
\frac{1}{1 + \delta} \left( \mathbf{H}_k \mathbf{P}_k^{-1} \mathbf{H}_k^T \right)
\]

Since \( \mathbf{P}_k^{-1} \) is a positive semidefinite matrix, we can factor it as \( \mathbf{P}_k^{-1} = \mathbf{U}_k \mathbf{H}_k^T \mathbf{U}_k^T \), where \( \mathbf{U}_k \) is the matrix square root of \( \mathbf{P}_k^{-1} \). Substituting \( \mathbf{H}_k^T = \mathbf{U}_k \mathbf{H}_k^T \) in the above expression, we obtain

\[
\frac{1}{1 + \delta} \left( \mathbf{H}_k \mathbf{P}_k^{-1} \mathbf{H}_k^T \right)
\]

as the quantity to be maximized. The expression within parentheses in (18) is known as the Rayleigh quotient of the vector \( \mathbf{H}_k \) with respect to the matrix \( \mathbf{P}_k^{-1} \) [12]. It attains a maximum value of \( \lambda_{\text{max}} \), the largest eigenvalue of \( \mathbf{P}_k^{-1} \), and this happens when \( \mathbf{H}_k^T \) is an eigenvector of \( \mathbf{P}_k^{-1} \) corresponding to \( \lambda_{\text{max}} \). Since \( \mathbf{H}_k^T = \mathbf{U}_k \mathbf{H}_k^T \) and an eigenvector of \( \mathbf{P}_k^{-1} \) is also an eigenvector of \( \mathbf{U}_k \), it follows that the quantity in (18) will be maximized if we select \( \mathbf{H}_k^T \) to be an eigenvector corresponding to the largest eigenvalue of \( \mathbf{P}_k^{-1} \). For this choice of \( \mathbf{H}_k \), the reduction in the trace of \( \mathbf{P}_k \) can be expressed as

\[
\text{tr}[\mathbf{P}_k] = \text{tr}[\mathbf{P}_k^{-1}] - \frac{\lambda_{\text{max}}}{1 + \delta}.
\]

The above result for scalar measurements can also be extended to the case of vector measurements [25].

The optimum selection of the measurement vector \( \mathbf{H}_k \) as the eigenvector corresponding to the largest eigenvalue of \( \mathbf{P}_k^{-1} \) implies that the Kalman estimation algorithm has to be executed at the encoder in addition to the decoder. Before a measurement formed at the encoder is transmitted to the decoder, it has to be necessarily quantized to keep the bit-rate requirement low. The quantization operation introduces an error in the measurement that can be modeled by the measurement noise term \( v_k \) in (2). Additional bit-rate savings can be expected if instead of \( \mathbf{H}_k \mathbf{x}_k \), the quantity \( \mathbf{H}_k (\mathbf{x}_k - \tilde{\mathbf{x}}_k) = \mathbf{H}_k \mathbf{e}_k \) is quantized and transmitted. This is because a linear combination of prediction errors, viz., \( \mathbf{H}_k \mathbf{e}_k \), typically has a smaller variance than a linear combination of speech samples, viz., \( \mathbf{H}_k \mathbf{x}_k \). Clearly, \( \mathbf{H}_k \mathbf{e}_k \) can be recovered from \( \mathbf{H}_k \mathbf{e}_k \) if required at the decoder. In the proposed scheme, therefore, measurements are transmitted in the form of innovations [ref. (3)]. The measurement noise variance \( R_k \), which is the same as the quantization-error variance in the present context, depends on the design of the quantizer as well as the statistics of the input signal to the quantizer. Since \( \mathbf{H}_k \mathbf{e}_k \) is a weighted sum of \( (n) \) random variables, it is modeled in the proposed scheme as a zero-mean, Gaussian random variable with variance \( E[(\mathbf{H}_k \mathbf{e}_k)(\mathbf{H}_k \mathbf{e}_k)^T] = \mathbf{H}_k \mathbf{P}_k^{-1} \mathbf{H}_k^T \), and quantized using a pdf-optimized, nonlinear quantizer [26], [27]. The quantization-error variance in this case can be expressed [27] as \( R_k = \delta (\mathbf{H}_k \mathbf{P}_k^{-1} \mathbf{H}_k^T) \), where \( \delta \) is a scalar constant that depends on the number of bits \( B \) used in quantization, thus justifying the assumption made earlier about \( R_k \). If no quantization is used, then clearly \( v_k = 0 = R_k = \delta \). Also, for fairly uncorrelated input such as the (noiseless) innovations \( \mathbf{H}_k \mathbf{e}_k \) and reasonably fine quantization \( (B \geq 2) \), the quantization error \( v_k \) can be assumed to be white [27].

D. Encoder and Decoder Structures

As pointed out in Section II, execution of the Kalman state estimation algorithm requires the knowledge of the system matrices \( \Phi_k \) and \( \mathbf{H}_k \) and the noise covariance matrices \( \mathbf{G}_k \) and \( \mathbf{R}_k \) in addition to the measurements \( \mathbf{x}_k \) and suitable values for \( \mathbf{x}_0 \) and \( \mathbf{P}_0 \). In the proposed scheme, the job of the encoder is seen as that of extracting the required information from the original speech and transmitting it to the decoder, which then uses it to synthesize speech through Kalman's algorithm.

The state transition matrix \( \Phi_k \) is completely specified by the combined predictor coefficients \( a_{i,m}^{(k)} \), \( i = 1, 2, \ldots, p; m = 1, 2, \ldots, M^{(k)} \), and the long-delay predictor coefficients \( g_{j,i}^{(k)} \), \( j = 1, 2, \ldots, q \), \( i = 1, 2, \ldots, p \), and \( M^{(k)} \), which is related to the pitch period. These \( p + q + 1 \) parameters are computed for each segment of the original speech by the encoder using LP analysis techniques, coded, and transmitted.

The process noise covariance matrix \( \mathbf{G}_k = [g_{i,j}^{(k)}] = E[\mathbf{w}_k \mathbf{w}_k^T] \) has only one nonzero element, viz., \( g_{11}^{(k)} = E[\mathbf{r}(k) \mathbf{r}(k)^T] \). An estimate of \( g_{11}^{(k)} \) is obtained for each segment by the encoder through inverse filtering the original speech and computing the time average of the squared values of the residual samples within the segment. This information is then coded and transmitted. The inverse filtering operation can be expressed mathematically as follows.

\[
r(k) = s(k) - \sum_{m=1}^{M^{(k)}} \gamma_{m}^{(k)} \gamma_{m}^{(k)} \delta(k - m).
\]

There is a slight abuse of notation here since \( r(k) \) is used to indicate the output of the inverse filter (i.e., prediction residual) in addition to the white-noise input to the speech signal model [see Fig. 1(b)]. However, as pointed out earlier, the prediction residual is approximately white and the two quantities in question can be considered equivalent.

Measurements are formed at every \( L_{th} \) time instant by the encoder, coded, and transmitted in the form of innovations \( \xi_k = \mathbf{H}_k \mathbf{e}_k + v_k \). The measurement vector \( \mathbf{H}_k \) is selected as
an eigenvector corresponding to the largest eigenvalue of \( P_k^- \). Since both the encoder and decoder can independently extract \( H_k \) from \( P_k^- \), no information regarding \( H_k \) needs to be transmitted by the encoder. Similarly, no information regarding the measurement noise variance \( R_k = \delta (H_k P_k^- H_k^T) \) needs to be transmitted since \( \delta \) is a function of \( B \), which is fixed, and the other quantities are available at both the encoder and decoder.

Both \( \hat{x}_0^- \) and \( P_0^- \) can be taken to be zero if we ensure that execution of Kalman's algorithm starts during a silent period of the given speech signal.

A schematic block diagram of the proposed coding scheme is shown in Fig. 2. The encoder extracts the following information for each segment of the original speech codes, and transmits to the decoder: 1) short-delay predictor coefficients \( \alpha_{i}^{(k)} \), \( i = 1, 2, \ldots, p \), 2) long-delay predictor coefficients \( \beta_{j}^{(k)} \), \( j = 1, 2, \ldots, q \), 3) value of \( M(k) \), which is related to the pitch period, 4) residual variance \( g_{i}^{(k)} \), and 5) innovations \( \xi_k \) at every \( L_{th} \) time instant. The Kalman estimator is executed at both the encoder and the decoder. At the encoder, the estimator converts the original speech signal into a sequence of innovations, an operation referred to sometimes as “whitening.” At the decoder, the innovations sequence is converted back to the speech signal—the “synthesis” operation. In obtaining the output speech signal \( \hat{y}_k \) from the estimated speech process \( \hat{x}_k \) at the decoder, the output vector \( C_k = [0 \ 0 \ 0 \ \ldots \ 1] \) is used. This introduces a time delay of \( M(k) + p + q - 1 \) units [ref. (16b)]; however, it provides the benefit of smoothing the estimate \( \hat{y}_k = \delta(1 - [M(k) + p + q - 1]) \) using measurements based on future speech samples.

\[ W(z) = \frac{1 - \sum_{i=1}^{p} \alpha_{i}^{(k)} z^{-i}}{1 - \sum_{i=1}^{p} \alpha_{i}^{(k)} \rho z^{-i}} \]  \hspace{1cm} (21)

where \( \rho \) is a fraction between 0 and 1. Notice that the effect of the weighting filter is to de-emphasize the error spectrum in the formant regions and the degree of de-emphasis is determined by the value of \( \rho \). The choice of an optimum value for \( \rho \) must be based on suitable listening tests. A typical value is 0.73 for speech sampled at 8 kHz [22].

In the proposed scheme, error spectrum shaping can be incorporated as follows. Since the object is to minimize the weighted error, it can be accomplished by synthesizing the best weighted speech (in the minimum mean-squared error sense) and then deweighting it to obtain the required output speech. Weighted speech can be synthesized by modifying the speech signal model in Fig. 1(a) to include the weighting filter \( W(z) \) in cascade with the long-term and short-term filters and also weighting the original speech before forming the measurements. The concatenation of the short-delay filter and the weighting filter [ref. (11), (21)] results in a filter with
system function

\[
\frac{1}{1 - P_0(z)} W(z) = \frac{1}{1 - \sum_{i=1}^{p}\alpha_i^{(k)} z^{-i}}.
\]  

(22)

Synthesis of weighted speech therefore can be performed by simply replacing the short-delay predictor coefficients \(\alpha_i^{(k)}\) by \(\alpha_i^{(k)} z_i^j\) for \(i = 1, 2, \ldots, p\), besides weighting the original speech by \(W(z)\) before forming the measurements. Deweighting the weighted speech can be achieved by means of the filter \(1/W(z)\). In Fig. 2, operations corresponding to error spectrum shaping are implemented by the blocks shown in dashed lines.

F. Relationship to Other Coding Schemes

The proposed coding scheme can be interpreted in different ways. It can be regarded as belonging to the class of LP coders since the signal model is obtained through linear predictive analysis of the original speech. However, instead of transmitting information about the excitation as in other LP codes [2]-[4], information about the innovations is transmitted as sparse but periodic intervals. But the innovations and the ideal excitation signal, viz., the prediction residual, are definitely related, as can be clearly seen when the measurement vector is selected to be \(H_k = [1 \ 0 \ 0 \ \cdots \ 0]\). Unlike the excitation signal, which is fed at the input of the signal model, the innovations are used directly to update the synthesized speech samples.

The proposed coder may also be regarded as a DPCM coder [14]. Clearly, if \(L = 1\) and \(H_k = [1 \ 0 \ 0 \ \cdots \ 0] = C_k\), then the coder described in this paper is essentially equivalent to a scalar DPCM coder (DPCM-APF-AQ). When \(L > 1\), i.e., when sparse measurements are used, the signal model (i.e., linear predictor) is used to estimate not only the next sample but the next \(L\) samples. This operation can be viewed as the “vector prediction” employed in a vector-DPCM coder [13]. It can also be viewed as the formation of the “zero-input response” in a LD-CELP coder [7]. Instead of quantizing the prediction error vector using VQ techniques as in [13] or [7], the proposed coder quantizes (using scalar techniques) the length of the projection of the error vector along \(H_k\) [ref. (2), (3)]. This projection is then used to update the predicted speech samples [ref. (4)-(6)]. The optimum choice of \(H_k\) as the eigenvector of \(P_k^*\) corresponding to its largest eigenvalue allows the projection to be taken in a direction along which the variance of the prediction error is maximum.

The idea of decimating the innovations to reduce the bit rate is in a sense similar to that contained in the residual excited linear predictive (RELDP) coder [28], where the prediction residual is low-pass-filtered and subsampled before being coded for transmission. In the RELDP coder, a spectral flattening procedure involving a nonlinear device is used to generate the excitation signal at the decoder from the low-pass-filtered residual; whereas, in the proposed coder, the Kalman estimator uses the innovations and the model information to synthesize speech using only linear operations.

IV. IMPLEMENTATION DETAILS

The coding scheme described in Section III was implemented in software, and several experiments were conducted to study its operation and evaluate its performance for different choices of coder parameters. The results of these studies will be presented in the next section, but first we discuss some of the important implementation details in this section.

In performing linear predictive analysis of the original speech to extract the signal model parameters, the short-delay predictor coefficients were computed first. The method used for this purpose is the “stabilized covariance method with high-frequency correction.” Details of this method can be found in [21]. In the different experiments conducted, the order \(p\) of the short-delay predictor was chosen to be between 8 and 16. The predictor coefficients were updated every 20 ms (corresponding to 160 samples at 8 KHz sampling) and transmitted in the form of the partial correlations. The partial correlations were coded by taking their inverse sines and quantizing them uniformly with bit resolutions of 5, 4, 4, 3, 3, 3, 3, 3, 3, 2, 2, 2, 1, and 1 [22] respectively for the first 16 values.

After computing the short-delay predictor coefficients as above, the speech signal was inverse-filtered to obtain the difference signal \(d(k)\), which was then analyzed to compute the long-delay predictor coefficients \(\beta_j^{(k)}, j = 1, 2, \ldots, q\) and the value of \(M^{(k)}\). The order \(q\) of the long-delay predictor was chosen as 3 and the pitch period \(M^{(k)} + (q - 1)/2 = M^{(k)} + 1\) was computed as the sample delay \(m\), which maximized the normalized correlation coefficient between \(d(k)\) and \(d(k - m)\) in the range between 5 and 12.875 ms (corresponding to 40 and 103 samples, respectively, at 8 KHz sampling). Notice that the value of \(M^{(k)} + 1\) thus obtained need not necessarily be the actual pitch period. The predictor coefficients were then obtained as the solution of a set of simultaneous linear equations that minimized the mean-squared prediction error between \(d(k)\) and its predicted value [21]. In the different experiments, the coefficients \(\beta_1^{(k)}, \beta_2^{(k)}, \text{and } \beta_3^{(k)}, \text{and the value of } M^{(k)}\) were updated every 10 ms. The coefficients were coded by first transforming them into

\[
\begin{align*}
&b_1^{(k)} = \beta_1^{(k)} + \beta_2^{(k)} + \beta_3^{(k)} \\
&b_2^{(k)} = \beta_1^{(k)} \beta_2^{(k)} \\
&b_3^{(k)} = \beta_1^{(k)} + \beta_3^{(k)}
\end{align*}
\]

(23)

and then quantizing them uniformly with 5, 4, and 4 bits, respectively. The value of \(M^{(k)}\) was coded using 6 bits.

After obtaining the short-delay and long-delay predictor coefficients and the value of \(M^{(k)}\), the residual signal \(r(k)\) was computed [ref. (20)]. An estimate of \(\delta_{11}^{(k)}\) was then obtained as the time average of the squared value of the residual signal. In the different experiments, the variance of the residual signal was updated every 10 ms and its value was coded using a logarithmic scale (i.e., decibels) by means of a uniform quantizer employing 6 bits.

As noted in the preceding section, the size \(n\) of the state vector \(x_k\) is given by \(n = M^{(k)} + p + q - 1\). Using the maximum values of \(M^{(k)}\) (102) and \(p\) (16), it was fixed at
\( n = 102 + 16 + 3 - 1 = 120 \) in the different experiments. Even though this size is relatively large, the simple structures of \( \Phi_k \) and \( G_k \) facilitate the implementation of the Kalman algorithm.

The measurements, as described earlier, were transmitted in the form of innovations. The measurement interval \( L \) ranged from 2 to 10 in the different experiments. Each measurement was coded using a Gaussian pdf-optimized, nonuniform quantizer with number of bits \( B \) ranging from 1 to 6. The step sizes of the quantizer were made proportional to the variance \( H_k P_k^{-1} H_k' \).

The measurement vector \( H_k \) was typically selected in the experiments to be the eigenvector corresponding to the largest eigenvalue of \( P_k \). The power method [29] was used for this purpose. In this method, we start with an arbitrary vector (or the previous value of \( H_k \)) and multiply it repeatedly with \( P_k \), normalizing the resulting vector each time with respect to its last element. After several iterations, the vector converges to the eigenvector corresponding to the largest eigenvalue and its last element to the largest eigenvalue. In the experiments, the iterations were stopped after 20 steps, or even earlier if the change in eigenvalue corresponding to two consecutive steps was not more than 5%. The coder performance was evaluated in the experiments primarily through the objective SEGSNR measure [27]. In computing the SEGSNR values, segments (blocks of 20 ms duration or 160 samples at 8 kHz sampling) with very low energy (\( \geq 40 \) dB below the average energy) were ignored. Informal listening tests were also used occasionally for subjective evaluation of the coder performance.

V. EXPERIMENTAL RESULTS

The results of several experiments that were conducted to study the operation and evaluate the performance of the proposed coding scheme for different choices of coder parameters are reported in this section. The results are presented for five representative speech sentences sampled at 8 kHz and quantized using 12 bits. The first three sentences are spoken by males and the last two by females. Additional details about these sentences can be found in Table I.

The effect of the measurement interval \( L \) (or equivalently, the frequency of measurements \( f_m = 8000/L \)) on the coder performance was studied through an experiment and the results obtained are shown in Table II. Clearly, the performance of the coder is strongly dependent on the value of \( L \) as one would expect. In this experiment, the order \( p \) of the short-delay predictor was chosen as 12. Optimum \( H_k \) vector was used and neither the measurements nor the signal model parameters were quantized. The performance figures shown in Table II, therefore, indicate the highest values attainable for the chosen value of \( p \).

The effect of quantizing the measurements on the performance was next studied by using different numbers of bits \( B \) for the quantizer of the innovations. The results of this experiment are shown in Table III. It is seen from the table that the performance of the coder does not deteriorate too much until the number of bits per quantized innovation sample is reduced to about 2. In this experiment, \( p \) was chosen as 12 and \( L \) was chosen as 4. Optimum \( H_k \) vector was used and no quantization of the signal model parameters was done. Quantization errors in measurements are handled easily by the Kalman estimator since measurement noise is included as part of the original problem formulation.

The number of bits per second allocated to measurements depends both on the frequency of measurements \( f_m \) and the number of bits \( B \) used to quantize each innovation sample. In order to study the role of the quantity \( f_mB \) versus the quality \( (B) \) of the measurements in influencing the coder performance, an experiment was conducted in which the performance was evaluated by changing the number of bits \( B \) to different values, but keeping the number of bits per second allocated to measurements \( (f_mB) \) constant. The results of this experiment are shown in graphical form in Fig. 3. The SEGSNR values shown in this figure are averages of the values obtained for all the five sentences listed in Table I. It is seen that it is better to have more approximate measurements than fewer accurate measurements, provided \( B \) is at least 2. In this experiment, \( p \) was chosen as 12. Optimum \( H_k \) vector was used and no quantization of the signal model parameters was done.
was established through an experiment. The coder performance was measured for the optimum selection of the $H_k$ vector as well as for two fixed selections, viz., $[1 \ 0 \ 0 \ \cdots \ 0]$ and $[0 \ 0 \ 0 \ \cdots \ 1]$. The results obtained are shown in Table IV. Besides yielding the best performance, the optimal $H_k$ vector selection generally resulted in a reconstruction error spectrum that was nearly white. In this experiment, $p$ was chosen as 12 and $L$ was chosen as 4. No quantization of the measurements or the signal model parameters was done.

The effect of changing the order $p$ of the short-delay predictor on the coder performance was also studied for the range of $p$ between 8 and 16. In this experiment, $L$ was chosen as 4 and optimum $H_k$ vector was used. No quantization of the measurements or the model parameters was done. It was found that the performance was fairly constant (around the values given in Table II for $L = 4$) within this range except for a slight drop when $p = 8$.

The effect of quantizing the signal model parameters, viz., $\alpha_i^{(k)}$, $i = 1, 2, \ldots, p$, $\beta_j^{(k)}$, $j = 1, 2, \ldots, q$, $M_k$, and $q_{1i}^{(k)}$, on the coder performance was studied next. The value of $p$ was varied between 8 and 16 and the bit allocations specified in Section IV were employed. Once again, $L$ was chosen as 4 and the optimum $H_k$ vector was used. No quantization of the measurements was done. Compared to the results of the previous experiment, the performance was found to reduce only slightly (an average of 0.4 dB) due to parameter quantization. Similar results were obtained for $L = 2$.

Based on the knowledge gained through the above experiments about the performance of the proposed coding scheme for different choices of coder parameters, three coders operating respectively at 16 000, 9533, and 7200 b/s were designed. The specifications of the 16 000 b/s coder are as follows: number of bits assigned to measurements: 12 000 b/s with $L = 2$ ($f_m = 4000$) and $B = 3$; number of bits assigned to model parameters: 4000 b/s with $p = 10$ and $q = 3$; the short-delay predictor coefficients were updated every 20 ms and quantized, respectively, using the following numbers of bits: 5, 5, 4, 4, 3, 3, 3, 3, 2, and 2 (total: 34); the long-delay predictor coefficients were updated every 10 ms and quantized, respectively, using the following numbers of bits: 5, 3, and 3 (total: 11); the approximate pitch period and the residual variance were updated every 10 ms and were quantized using 6 bits each (total: 12). The specifications of the 9533 b/s coder are as follows: number of bits assigned to measurements: 5333 b/s with $L = 3$ and $B = 2$; number of bits assigned to model parameters: 4200 b/s; the model used was the same as that of the 16 000 b/s coder except that the long-delay predictor coefficients were quantized using the following numbers of bits: 5, 4, and 4 (total: 13). The specifications of the 7200 b/s coder are as follows: number of bits assigned to measurements: 4000 b/s with $L = 4$ and $B = 2$; number of bits assigned to model parameters: 3200 b/s with $p = 8$ and $q = 3$; the short-delay predictor coefficients were updated every 20 ms and quantized, respectively, using the following numbers of bits: 5, 5, 4, 4, 3, 3, 2, and 2 (total: 28); the long-delay predictor coefficients were updated every 10 ms and quantized, respectively, using the following numbers of bits: 4, 2, and 2 (total: 8); the approximate pitch period and the residual variance were updated every 10 ms and were quantized using 5 bits each (total: 10).

The performances of the above coders without and with error spectrum shaping were evaluated for the five speech sentences listed in Table I. The results obtained are shown in Table V. In shaping the error spectrum, a value of 0.73 was chosen for $\rho$ [ref. (21)]. Notice the significant drop in SEGSNR values caused by error spectrum shaping. Informal listening tests, however, reveal that the subjective performance improves with error spectrum shaping, albeit slightly. For comparison purposes, a 16 000 b/s LD-CELP coder [7] was also simulated and tested using the same five sentences. It was found that the objective performance of the proposed 16 000 b/s coder was slightly better than that of the CELP coder (about 1.6 dB better on the average without error spectrum shaping and 0.3 dB better on the average with error spectrum shaping). In informal listening tests, the subjective performance of the proposed 16 000 b/s coder was judged to be comparable to that of the LD-CELP coder by untrained listeners and slightly inferior by trained listeners. These tests involved four untrained listeners and three trained listeners. The listeners were asked to rank the performances of three different coders at a time. In addition to the proposed and LD-CELP coders, $\mu$-law coders with 5, 6, 7, and 8 bits were used in the tests. The performance of the proposed coder was

![Graph showing the effect of quality versus quantity of measurements on coder performance](image-url)

**Table IV**

<table>
<thead>
<tr>
<th>ID</th>
<th>Optimum $H_k$</th>
<th>$H_k = 10 \vdots 0$</th>
<th>$H_k = 0 \vdots 01$</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>14.4</td>
<td>11.1</td>
<td>8.4</td>
</tr>
<tr>
<td>M2</td>
<td>15.4</td>
<td>11.5</td>
<td>8.6</td>
</tr>
<tr>
<td>M3</td>
<td>14.7</td>
<td>12.3</td>
<td>9.7</td>
</tr>
<tr>
<td>F1</td>
<td>13.7</td>
<td>11.3</td>
<td>8.4</td>
</tr>
<tr>
<td>F2</td>
<td>16.8</td>
<td>13.5</td>
<td>10.2</td>
</tr>
</tbody>
</table>

reduce only slightly (an average of 0.4 dB) due to parameter quantization. Similar results were obtained for $L = 2$. Based on the knowledge gained through the above experiments about the performance of the proposed coding scheme for different choices of coder parameters, three coders operating respectively at 16 000, 9533, and 7200 b/s were designed. The specifications of the 16 000 b/s coder are as follows: number of bits assigned to measurements: 12 000 b/s with $L = 2$ ($f_m = 4000$) and $B = 3$; number of bits assigned to model parameters: 4000 b/s with $p = 10$ and $q = 3$; the short-delay predictor coefficients were updated every 20 ms and quantized, respectively, using the following numbers of bits: 5, 5, 4, 4, 3, 3, 3, 3, 2, and 2 (total: 34); the long-delay predictor coefficients were updated every 10 ms and quantized, respectively, using the following numbers of bits: 5, 3, and 3 (total: 11); the approximate pitch period and the residual variance were updated every 10 ms and were quantized using 6 bits each (total: 12). The specifications of the 9533 b/s coder are as follows: number of bits assigned to measurements: 5333 b/s with $L = 3$ and $B = 2$; number of bits assigned to model parameters: 4200 b/s; the model used was the same as that of the 16 000 b/s coder except that the long-delay predictor coefficients were quantized using the following numbers of bits: 5, 4, and 4 (total: 13). The specifications of the 7200 b/s coder are as follows: number of bits assigned to measurements: 4000 b/s with $L = 4$ and $B = 2$; number of bits assigned to model parameters: 3200 b/s with $p = 8$ and $q = 3$; the short-delay predictor coefficients were updated every 20 ms and quantized, respectively, using the following numbers of bits: 5, 5, 4, 4, 3, 3, 2, and 2 (total: 28); the long-delay predictor coefficients were updated every 10 ms and quantized, respectively, using the following numbers of bits: 4, 2, and 2 (total: 8); the approximate pitch period and the residual variance were updated every 10 ms and were quantized using 5 bits each (total: 10).

The performances of the above coders without and with error spectrum shaping were evaluated for the five speech sentences listed in Table I. The results obtained are shown in Table V. In shaping the error spectrum, a value of 0.73 was chosen for $\rho$ [ref. (21)]. Notice the significant drop in SEGSNR values caused by error spectrum shaping. Informal listening tests, however, reveal that the subjective performance improves with error spectrum shaping, albeit slightly. For comparison purposes, a 16 000 b/s LD-CELP coder [7] was also simulated and tested using the same five sentences. It was found that the objective performance of the proposed 16 000 b/s coder was slightly better than that of the CELP coder (about 1.6 dB better on the average without error spectrum shaping and 0.3 dB better on the average with error spectrum shaping). In informal listening tests, the subjective performance of the proposed 16 000 b/s coder was judged to be comparable to that of the LD-CELP coder by untrained listeners and slightly inferior by trained listeners. These tests involved four untrained listeners and three trained listeners. The listeners were asked to rank the performances of three different coders at a time. In addition to the proposed and LD-CELP coders, $\mu$-law coders with 5, 6, 7, and 8 bits were used in the tests. The performance of the proposed coder was
TABLE V
PERFORMANCES OF THE 16000-b/s, 9533-b/s, AND 7200-b/s CODERS

<table>
<thead>
<tr>
<th>ID</th>
<th>16000 b/s coder</th>
<th>9533 b/s coder</th>
<th>7200 b/s coder</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Without Error</td>
<td>With Error</td>
<td>Without Error</td>
</tr>
<tr>
<td></td>
<td>Shaping</td>
<td>Shaping</td>
<td>Shaping</td>
</tr>
<tr>
<td>M1</td>
<td>16.6</td>
<td>13.5</td>
<td>11.8</td>
</tr>
<tr>
<td>M2</td>
<td>17.5</td>
<td>13.9</td>
<td>11.7</td>
</tr>
<tr>
<td>M3</td>
<td>17.6</td>
<td>14.9</td>
<td>12.2</td>
</tr>
<tr>
<td>F1</td>
<td>17.0</td>
<td>14.5</td>
<td>11.6</td>
</tr>
<tr>
<td>F2</td>
<td>19.4</td>
<td>16.0</td>
<td>13.2</td>
</tr>
</tbody>
</table>

Fig. 4. SNR versus time plot for the second sentence, ID: M2 (16 000 b/s coder, no error spectrum shaping).

also judged to be close to that of 7-bit μ-law coder. To further illustrate the performance of this coder, a plot of SNR versus time (or segment number) for the second sentence (ID: M2) is shown in Fig. 4. No error spectrum shaping was done in this case. It can be seen that the coder performance is quite good for voiced sounds and poor to medium for unvoiced sounds. In informal listening tests, slight modeling noise was observed during unvoiced sounds by the trained listeners. The small prediction gains typically associated with unvoiced speech sounds is perhaps the reason.

The computational complexity of the proposed scheme is rather high. For example, the 16000 b/s coder simulation program took about 180 s to process 1 s of speech on a HDS AS/9180 computer (15 Mflops). It was observed that a considerable amount of time was being spent on updating the error covariance matrix in the Kalman estimation algorithm. The program, however, was not optimized to run faster. Furthermore, many of the computations were done using floating-point numbers. This coder also introduced a delay of about 35 ms.

VI. CONCLUSIONS

A sparse-innovations coding approach to speech data compression using the Kalman state estimation algorithm was described. The approach uses the conventional two-stage, time-varying, all-pole filter model for the speech signal; but instead of some excitation information, measurements of linear combinations of actual speech samples taken at sparse but periodic intervals and provided in the form of innovations are used. An optimum measurement strategy for the Kalman estimator was described. A procedure for shaping the error spectrum of the synthesized speech was also outlined. Results of simulation experiments were provided for different choices of coder parameters. These results indicate that a coder based on the proposed approach has the potential to provide high-quality speech at low bit rates. The computational complexity of such a coder and the coding delay are, however, rather high.

At its current state of development, the proposed coder is not expected to replace the well-established CEPH coder; however, it provides a new way of looking at the "excitation selection" problem that may perhaps be incorporated in a conventional CEPH coder. The proposed coder can also be configured easily as a variable-bit-rate coder, for example, to maintain the quality of the synthesized speech above a certain level. To achieve this, the error covariance matrix can be monitored at both the encoder and decoder to determine the instants at which the measurements are to be taken or even the number of measurements to be taken at any particular instant. In the latter case, the first measurement will be based on the eigenvector corresponding to the largest eigenvalue, the second measurement will be based on the eigenvector corresponding to the second largest eigenvalue, and so on.

As mentioned in the introduction, important implementation issues such as complexity, delay, and word length were not given their due importance given the early stages of development of the new approach. Clearly, future investigation of these issues is absolutely necessary. One approach to reducing the complexity of the proposed scheme is to keep the long-delay filter outside the Kalman filter loop, but still incorporating its effect on synthesized speech. Coding delay can be reduced by using backward adaptation techniques for the estimation of the signal model parameters. The effect of channel errors on coder performance is another important issue to be studied. The errors can affect the bit rate used to code the signal model parameters as well as the measurements. If forward adaptation is used as in the proposed scheme, the effect of errors on the signal model can be contained to short durations. However, errors in both signal model and measurements would affect the error covariance matrix on which the estimates are based. The silence periods of the speech signal can be used to advantage here to bring both the encoder and decoder to synchronization [16]. Furthermore, it is known that the Kalman estimator tends to forget the past in favor of new measurements, especially if the measurements are of reasonable quality. This property may prove useful in recovering from error conditions. Yet another issue to be investigated is the use of improved signal models, e.g., one that uses colored noise input for nonsteady speech sounds.
ACKNOWLEDGMENT
The authors would like to thank Mr. S. Satyamurti, a former graduate student at ISU, for his help in executing the simulation programs and obtaining the results. We would also like to thank Dr. J. Mills of Tellabs, Inc., for his help in the subjective evaluation of the coder performance. Thanks are also due to the anonymous referees for their valuable comments and suggestions.

REFERENCES

Tenkasi V. Ramabadran (S’82–M’86) was born in Kallidakkurichi, India, on June 1, 1949. He received the B.E. degree from REC, Trichy, India, in 1970, the M.Tech. degree from IIT, Madras, India, in 1972, and the M.S. and Ph.D. degrees from the University of Notre Dame, Notre Dame, IN, in 1984 and 1986, respectively, all in electrical engineering. He worked as an R&D Engineer from 1973 to 1980 at both in Best and Crompton Engineering Ltd., Madras, India, in the area of industrial electronics. He was with NTBF, Bangalore, India, as a Project Manager from 1980 to 1982. Since August 1986, he has been with Iowa State University, Ames, as an Assistant Professor of Electrical Engineering and Computer Engineering. His research interests lie in the areas of data compression, speech coding, and image coding.

Deepen Sinha (S’82) was born in Gonda, India. He was awarded the B.Tech. (Honors) degree in electronics and electrical communication engineering from the Indian Institute of Technology, Kharagpur, in 1986, the M.S. degree in electrical engineering from Iowa State University, Ames, in 1989, and the Ph.D. degree in electrical engineering from the University of Minnesota in 1993. During 1986–1987 he worked as a Computer Research Engineer at UPTRON in Lucknow, India. He is currently with the Signal Processing Research Department at AT&T Bell Labs, Murray Hill, NJ. The focus of his present work is in the area of multichannel audio compression. His other research interests are in the areas of very low bit-rate coding of speech and music signals, video compression, time-frequency signal representation, and filterbanks. He is a member of Phi Kappa Phi.