THE USE OF DISTANT SAMPLE PREDICTION IN SPEECH CODERS

Ravi P. Ramachandran

Caip Center
Department of Electrical Engineering
Rutgers University
Piscataway, New Jersey, U.S.A.

Abstract

Distant sample or pitch prediction is an important aspect in predictive speech coders. The pitch predictor greatly enhances the efficiency of these coders by regenerating the periodicity in the decoded speech. This paper provides a review of the research done in pitch prediction. The significant issues considered are filter formulation, determination of the pitch parameters, stability, implementation as part of a speech coder and bit rate for transmission of the pitch parameters.

Introduction

Predictive speech coders make use of the correlations in the speech signal to enhance coding efficiency. Two major types of correlations are present, namely, near-sample redundancies and distant-sample redundancies. Distant-sample redundancies are due to the inherent periodicity of voiced speech and form the focus of this paper. In predictive speech coders, the cascade of two nonrecursive prediction error filters process the original speech signal. The formant filter removes near-sample redundancies. The pitch filter acts on distant-sample waveform similarities. The parameters that are quantized and coded for transmission include the filter coefficients and the resulting residual signal. From the coded parameters, the receiver decodes the speech by passing the quantized residual through a pitch synthesis filter and a formant synthesis filter. The filtering step at the receiver can be viewed in the frequency domain as inserting the fine pitch structure and shaping the spectral envelope to insert the formant structure. The formant and pitch filters are adaptive in that the analysis to determine the coefficients is carried out frame by frame. The frame update rate is chosen to be slow enough to keep the transmission rate required small, yet fast enough to allow the speech signal under consideration to be adequately described by a set of constant parameters.

Two examples of predictive coders include the adaptive predictive coder (APC) [1][2] and Code-Excited Linear Prediction (CELP) [3]. In APC, the formant and pitch predictors are placed in a feedback loop around the quantizer. In CELP, the residual is vector quantized by a stochastic codebook containing a repertoire of waveforms consisting of Gaussian random numbers with unit variance. The selection process uses an analysis-by-synthesis strategy in which each waveform is passed through the synthesis filters to allow for a comparison with the original speech. The waveform that leads to the closest resemblance to the original speech is chosen.

The aim of this paper is to present a review of the research done in the area of pitch prediction and discuss its impact on the efficiency of speech coders. The emphasis will be on the application to CELP. The major issues dealt with are (1) the transfer function of the pitch filter and parameter computation, (2) stability aspects, (3) coding strategies, (4) interaction with the formant filter and (5) parameter interpolation.

Pitch Predictors

A formant predictor has a transfer function $F(z) = \sum_{i=1}^{Q} a_i z^{-i}$ where Q is between 8 and 16 for 8 kHz sampled speech. The corresponding synthesis filter is $H_F(z) = 1/(1 - F(z))$. The simplest form of the pitch predictor has one tap and is given by $P(z) = \beta_1 z^{-M}$ where the integral delay M represents the pitch period. Since the sampling frequency is unrelated to the pitch period, a 3 tap predictor serves like an interpolation filter and is given by $P(z) = \beta_1 z^{-M+1} + \beta_2 z^{-M} + \beta_3 z^{-M-1}$. The pitch synthesis filter has a system function $H_P(z) = 1/(1 - P(z))$.

In computing the predictor coefficients and M, we consider the situation of a signal $s(n)$ passing through the prediction error filter $1 - P(z)$ to generate a residual $r(n)$. This is known as an open-loop analysis in that the parameters are determined by considering $s(n)$ to be the input speech or the residual formed after formant prediction. Assuming a given value of M, the coefficients of $P(z)$ are chosen to minimize the squared residual $E = \sum_{n=1}^{N} r^2(n)$ where N is the number of samples in one frame. The minimization of E leads to a system of equations which can be written in matrix form as $Ac = d$. For a 3 tap predictor, the entries of the matrix A are

$$A(i,j) = \phi(M+i, M+j) = \sum_{n=1}^{N} s(n-M-i)s(n-M-j) \quad (1)$$

for $-1 \le i,j \le 1$. The vector $c = [\beta_1 \;\; \beta_2 \;\; \beta_3]^T$ and the vector $d = [\phi(0, M-1) \;\; \phi(0, M) \;\; \phi(0, M+1)]^T$. Note that for the 1 tap case, the predictor coefficient is determined as $\beta_1 = \phi(0, M)/\phi(M, M)$.

Methods to determine M are described in [4]. The simplest approach is to consider E as a function of M for the 1 tap case and determine its smallest value. With β_1 as given above, the resulting squared residual is

$$E = \phi(0,0) - \frac{\phi^2(0,M)}{\phi(M,M)} \quad (2)$$

We find the value of M (over the range 20 to 147) that leads to the largest value of $\phi^2(0,M)/\phi(M,M)$ which in turn gives the smallest value of E in the 1 tap case. This method of considering all possible values of M to minimize E is known as an exhaustive search. Since an exhaustive search for the best value of M is more complex for the 3 tap case, the optimal value of M for 1 tap predictors is taken as the pitch delay. The prediction gain measures the extent to which a predictor removes redundancies in a signal. It is the ratio of the average energy of the input signal to the average energy of the residual. Experiments have shown that processing speech first through the formant filter and then through the pitch filter results in a higher overall prediction gain than the reverse arrangement [4].

The receiver of a speech coder resynthesizes speech by passing the quantized residual through $H_P(z)$ and $H_F(z)$. An unstable $H_P(z)$ can accentuate the quantization noise. The method of determining the pitch predictor coefficients as described above does not assure a stable $H_P(z)$. Given the predictor coefficients and M, the stability in the 1 tap case is easy to check ($|\beta_1| < 1$). However, for 3 tap filters, conventional stability tests involve the checking of about M conditions which can be quite high for pitch predictors. Furthermore, known tests do not lend themselves to a technique that can stabilize an unstable $H_P(z)$. To overcome these problems, a stability test for 3 tap filters based on a tight sufficient condition was formulated in [5] and is as follows. Let $a = \beta_1 + \beta_3$ and $b = \beta_1 - \beta_3$.

- 1. If $|a| \ge |b|$, then satisfying the condition $|\beta_1| + |\beta_2| + \beta_3| < 1$ is sufficient for stability.
- 2. If $|a| < |b|$, two conditions must be satisfied.
  - (i) $|\beta_2| + |a| < 1$.
  - (ii) Either $b^2 \le |a|$ or $b^2\beta^2 - (1-b^2)(b^2 - a^2) < 0$.

The proposed test is independent of the order M, is computationally much simpler than known tests based on necessary and sufficient conditions and naturally leads to a stabilization technique. It is also shown in [5] that the set of test conditions, although only sufficient, is tight for finite M and comprises a set of necessary and sufficient conditions in the limit of large M.

When stabilizing a pitch filter, we deviate from an optimal filter $1 - P(z)$ that maximizes the prediction gain but is not minimum phase

CH 3381-1/93/$01.00 ©1993 IEEE

to a minimum phase pitch filter that is suboptimal. The stabilization technique is formulated such that $H_P(z)$ becomes stable and the reduction in prediction gain offered by $1-P(z)$ is minimized. The technique is based on scaling β_1, β_2 and β_3 by a common factor t [5]. If $|a| \geq |b|$, the value of t is

$$t = \frac{1}{|\beta_1| + |\beta_2| + |\beta_3|} \tag{3}$$

If $|a| < |b|$ and $b^2 \leq |a|$, then

$$t = \frac{1}{|a| + |\beta_2|} \tag{4}$$

If $|a| < |b|$ and $b^2 > |a|$, then

$$t = \sqrt{\frac{b^2 - a^2}{b^4 + b^2\beta_2^2 - b^2a^2}} \tag{5}$$

The method is computationally simple and noniterative. Experiments have shown that stabilization results in a negligible reduction in prediction gain and in speech of better perceptual quality.

For coding the pitch parameters, first consider the delay. There are 128 different integer values of M between 20 and 147. This takes up 7 bits. For a 1 tap filter, 3 bits are usually used to quantize β_1. Note that if the training data for the quantizer for β_1 is restricted to the interval $(-1, 1)$, the resulting codebook entries ensure stability. For a 3 tap filter, either scalar quantization or vector quantization of the coefficients are possible. One of the earliest known methods [6] used a transformation of the predictor coefficients to a new parameter set and designed a scalar quantizer for each transformed parameter. However, a total of 13 bits are required. This is not only expensive but does not assure that $H_P(z)$ is stable after quantization. A vector quantization of the predictor coefficients can bring the number of bits down to 3 [7]. Moreover, the codebook entries can always correspond to a stable $H_P(z)$. For a 5 ms update, 10 bits for the pitch parameters corresponds to 2000 bits per second.

Fractional Delay Predictors

The original motivation of going from a 1 tap to a 3 tap predictor was to provide interpolation between the samples since the pitch period is unrelated to the sampling frequency. Also, 3 tap predictors provide a higher prediction gain than 1 tap filters. However, the main drawback of 3 tap predictors is in the coding effort. Extra bits are needed to encode three predictor coefficients if a scalar quantizer is used. Also, the only way the number of bits can be lowered is by using a vector quantizer. Hence, a fractional delay pitch predictor was proposed in [8]. A fractional delay predictor has 1 tap but allows for better temporal resolution by allowing the delay M to be expressed as an integer plus a fraction l/D where $0 \leq l < D$ and l and D are integers. By using this fractional delay, the sampling frequency effectively increases by a factor D thereby providing a better match between the pitch delay M and the pitch period of the underlying continuous time speech signal. The implementation of a fractional delay is done by using a nonrecursive interpolation filter with linear phase. An efficient implementation results by using a polyphase structure. Although many design approaches for interpolation filters exist, a $\sin x/x$ function weighted by a Hamming window was used in [8].

Various experiments with fractional delay predictors revealed the following [8]. When using the formant predicted residual as the input to $1-P(z)$, the prediction gain improves with the degree of resolution D but virtually saturates for $D > 8$. One tap filters with $D = 2$ and $D = 4$ result in about the same prediction gain as for a 3 tap filter with integer delays. An exhaustive search for the best M to minimize the squared residual is more complex as D increases. Fast search procedures by a judicious sampling of the possible values of M leads to a negligible decrease in the prediction gain. Another way of reducing the complexity is to do an exhaustive search of a subset of the allowable fractional delay values that are nonuniformly distributed between 20 and 147. By using this subset, the coding effort can be kept at 7 bits even with fractional delays. The coding efficiency of the fractional delay predictor is the same as for a 1 tap or 3 tap pitch predictor with integer delays. However, the coding effort is facilitated for the 1 tap filter by using a scalar quantizer. Also, a 1 tap filter provides for an easy stability check and possible stabilization. Subjective tests in a CELP coder reveal that the speech quality is indeed the best with fractional delay predictors [8].

Use in CELP Coding System

The pitch predictor is a significant component of a CELP system in that it regenerates the periodicity in the decoded speech. Figure 1 shows the decoder structure of a CELP system with a 1 tap pitch predictor. The input to the formant synthesis filter $v(n)$ consists of two parts, namely, the contribution of the pitch predictor $\beta_1 v(n-M)$ and the excitation to the pitch synthesis filter $ge(n)$. Rather than do an open-loop analysis on the input signal to determine the pitch predictor parameters, an alternative closed-loop search is now described [9]. For a closed-loop search, β_1 and M are determined such that the mean-square error between the original speech and the pitch predictor contribution to the decoded speech is minimized. This is synonymous to an analysis-by-synthesis strategy [10] in which the decoder structure forms a major component of the encoder structure for the purposes of parameter computation. The closed-loop algorithm results in the pitch predictor optimally contributing to the minimization of the error and hence, leads to better speech quality than its open-loop counterpart.

The closed-loop method can be interpreted to be a search through an adaptive codebook whose entries correspond to the vectors $v(n-M)$ for different values of M [11]. This leads to an alternative structure of CELP as shown in Figure 2. Note that this single adaptive codebook formulation is restricted to 1 tap pitch predictors. For the CELP algorithm, the formant predictor parameters are first determined from the original speech and the residual $r(n)$ generated. Since the parameters are updated frame by frame, the initial error signal $f_0(n)$ is $r(n) \star h(n)$ minus the zero input response of the weighted formant synthesis filter $H_F(z/\alpha)$. Note that $h(n)$ is the impulse response of $H_F(z/\alpha)$ and α is usually around 0.8. The weighting is done to deemphasize the formant regions where more noise can be tolerated. The best adaptive codebook entry (or equivalently, the optimal value of M) is chosen to minimize

$$E = \sum_{n=1}^{N} [f_0(n) - \beta_1 v(n-M) \star h(n)]^2 \tag{6}$$

The derivative of E with respect to β_1 is set equal to zero to derive an optimal β_1 and get an expression for E explicitly in terms of M. This results in

$$\beta_1 = \frac{\sum_{n=1}^{N} f_0(n)[v(n-M) \star h(n)]}{\sum_{n=1}^{N} [v(n-M) \star h(n)]^2} \tag{7}$$

and

$$E = \sum_{n=1}^{N} f_0^2(n) - \frac{[\sum_{n=1}^{N} f_0(n)[v(n-M) \star h(n)]]^2}{\sum_{n=1}^{N} [v(n-M) \star h(n)]^2} \tag{8}$$

The choice of M is that which minimizes E. As in the open-loop case, M can be found by an exhaustive search of a subset of nonuniformly distributed fractional values to ensure a 7 bit representation. The CELP algorithm continues by determining the best stochastic codebook (populated by Gaussian random numbers with unit variance) entry and the optimal gain factor g such that the mean-square difference between the original and decoded speech is further reduced. The stochastic codebook injects a random component in the decoded speech to refine its description.

Pitch Parameter Interpolation

Although the use of a fractional delay predictor in conjunction with a closed-loop algorithm improves speech quality, the bit rate required to achieve this quality is rather high at 10 bits/5 ms or 2000 bits/second. The pitch delay takes up 7 out of the 10 bits. This is rather unusual since the pitch period varies smoothly over time. However, merely changing the update rate to 10 ms or more to decrease the bit rate has been shown to degrade speech quality. Also, a more coarse representation of the delay to less than 7 bits diminishes speech

quality. Both these strategies increase the mean-square error between the original and decoded speech due to time misalignment. The issue of reducing the bit rate to transmit the pitch parameters without compromising speech quality has been addressed in [12].

In [12], the pitch predictor is first reformulated to allow for the pitch delay to vary linearly on a sample by sample basis rather than assume a constant value in each frame as is conventionally the case. This allows for continuity of the delay contour across frame boundaries resulting in a more smooth evolution of the pitch cycle waveform which is crucial to getting high quality. The adaptive codebook entries for a particular frame are given by $v(n - M(n))$ where $M(n) = M_0 + \alpha n$ for $n = 1$ to N. Different values of the slope α specify different linear contours with M_0 being the endpoint delay of the previous frame. Now, the best adaptive codebook entry is chosen to minimize

$$E = \sum_{n=1}^{N} [f_0(n) - \beta_1 v(n - M(n)) \star h(n)]^2 \tag{9}$$

The best adaptive codebook index is transmitted. The codebook indices have a one-to-one correspondence with the value of the endpoint delay in each frame. These endpoint delays assume one of $2^7 = 128$ nonuniformly distributed values for a 7 bit representation. The pitch gain β_1 continues to assume a constant value throughout any particular frame.

With this new pitch predictor, the bit rate is lowered by linearly interpolating the pitch parameters over 20 ms intervals. We first concentrate only on the pitch delay interpolation method. In this method, the delay at the endpoint of the 20 ms interpolation interval is estimated from an open-loop analysis. A linear delay contour is fixed over the entire interval thereby fixing the adaptive codebook contribution given by $u(n) = v(n - M(n))$. Consider the subintervals of the larger 20 ms interval. For these subintervals, $u(n)$ is again known. However, $u(n)$ is not an optimal adaptive codebook vector in that there will be time misalignment with the input residual $r(n)$. Therefore, $r(n)$ is time warped to match $u(n)$ and correct for the time misalignment. Since warping results in time asynchrony, we use n to denote the original time base and m for the warped time domain. The simplest form of warping is when the warped signal is given by $r_w(m) = r(m/A)$. Note that A represents the different time scale modifications. Although m is an integer, m/A is a real number and the value of $r(m/A)$ is obtained by interpolating the samples of the residual. The best warped signal is determined by searching different values of A. For each value of A, $f_0(m)$ is $r_w(m) \star h(m)$ minus the zero input response of $H_F(z/\alpha)$. The optimal warp is that which minimizes the mean-square error between $f_0(m)$ and $u(m) \star h(m)$ or which maximizes

$$T = \frac{[\sum_m f_0(m)[u(m) \star h(m)]]^2}{[\sum_m f_0^2(m)][\sum_m [u(m) \star h(m)]^2]} \tag{10}$$

The pitch gain β_1 is still updated every subinterval. With this approach, information about the delay value at the endpoint of every relatively long 20 ms interval is transmitted to substantially lower the bit rate to 7 bits/20 ms. Note that time warping is still performed in each subinterval at no cost in bits.

The next step is to reduce the bit rate for the pitch gain by means of interpolation. This is accomplished by interpolating the root mean-square energy of the adaptive codebook contribution which in turn can be converted to a pitch gain. Note that the pitch gain is still held constant for every subinterval. In this case, the warped original speech is amplitude scaled by a factor λ to match the known scaled adaptive codebook contribution $\beta_1 u(m)$ in a mean-square sense. In finding λ, suppose $\beta_1 u(m)$ is indeed optimally matched to $f_0(m) = \lambda r_w(m)$ minus the zero input response of $H_F(z/\alpha)$. Then, the optimal scaling factor for $\beta_1 u(m)$ is unity. Hence,

$$1 = \frac{\sum_m f_0(m)[\beta_1 u(m) \star h(m)]}{\sum_m [\beta_1 u(m) \star h(m)]^2} \tag{11}$$

from which λ can be obtained.

In this algorithm, the decoded speech is optimally matched to the amplitude scaled and warped original speech. In practice, the extent of time warping will only slightly alter the original pitch contour. Also, amplitude scaling will slightly modify the original energy contour. Hence, the perceptual quality of the original speech is preserved. Increasing the update interval to 20 ms in conjunction with modifying the original speech lowers the bit rate by a factor of four without compromising perceptual quality. Subjective tests confirm this result.

Summary

This paper presents an overview of the important issues regarding pitch filters like system function formulation, implementation, stability and coding considerations. We discuss both the open-loop and closed-loop procedures for computing the pitch delay and predictor coefficient(s). A simple algorithm to ensure stability is described. A fractional delay predictor improves speech quality, allows for simple stabilization and requires only a scalar quantizer for coding. Recently, an interpolation method for the pitch parameters based on a modification of the analysis-by-synthesis paradigm leads to a substantial decrease in the bit rate.

References

1. B. S. Atal and M. R. Schroeder, "Adaptive predictive coding of speech signals", *Bell System Technical Journal,* vol. 49, pp. 1973-1986, Oct. 1970.

2. B. S. Atal and M. R. Schroeder, "Predictive coding of speech signals and subjective error criteria", *IEEE Trans. on Acoust., Speech and Signal Proc.,* vol. ASSP-27, pp. 247-254, June 1979.

3. M. R. Schroeder and B. S. Atal, "Code-excited linear prediction (CELP): High-quality speech at low bit rates", *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Proc.,* Tampa, Florida, pp. 25.1.1-25.1.4, March 1985.

4. R. P. Ramachandran and P. Kabal, "Pitch prediction filters in speech coding", *IEEE Trans. on Acoust., Speech and Signal Proc.,* vol. 37, pp. 467-478, April 1989.

5. R. P. Ramachandran and P. Kabal, "Stability and performance analysis of pitch filters in speech coders", *IEEE Trans. on Acoust., Speech and Signal Proc.,* vol. ASSP-35, pp. 937-946, July 1987.

6. B. S. Atal, "Predictive coding of speech at low bit rates", *IEEE Trans. on Commun.,* vol. COM-30, pp. 600-614, April 1982.

7. P. Kroon and B. S. Atal, "Quantization procedures for the excitation in CELP coders" *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Proc.,* Dallas, Texas, pp. 36.8.1-36.8.4, April 1987.

8. P. Kroon and B. S. Atal, "On improving the performance of pitch predictors in speech coding systems", in *Advances in Speech Coding*, edited by B. S. Atal, V. Cuperman and A. Gersho, Kluwer Academic Publishers, pp. 321–327, 1991.

9. S. Singhal and B. S. Atal, "Improving performance of multi-pulse LPC coders at low bit rates" *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Proc.,* San Diego, California, pp. 1.3.1-1.3.4, March 1984.

10. P. Kroon and E. F. Deprettere, "A class of analysis-by-synthesis predictive coders for high quality speech coding at rates between 4.8 and 16 kbits/s", *IEEE Jour. on Selec. Areas in Commun.,* vol. 6, pp. 353-363, Feb. 1988.

11. W. B. Kleijn, D. J. Krasinski and R. H. Ketchum, "An efficient stochastically excited linear predictive coding algorithm for high quality low bit rate transmission of speech", *Speech Communication,* vol. 7, pp. 305-316, 1988.

12. W. B. Kleijn, R. P. Ramachandran and P. Kroon, "Generalized analysis-by-synthesis coding and its application to pitch prediction", *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Proc.,* San Francisco, California, pp. I337-I340, March 1992.

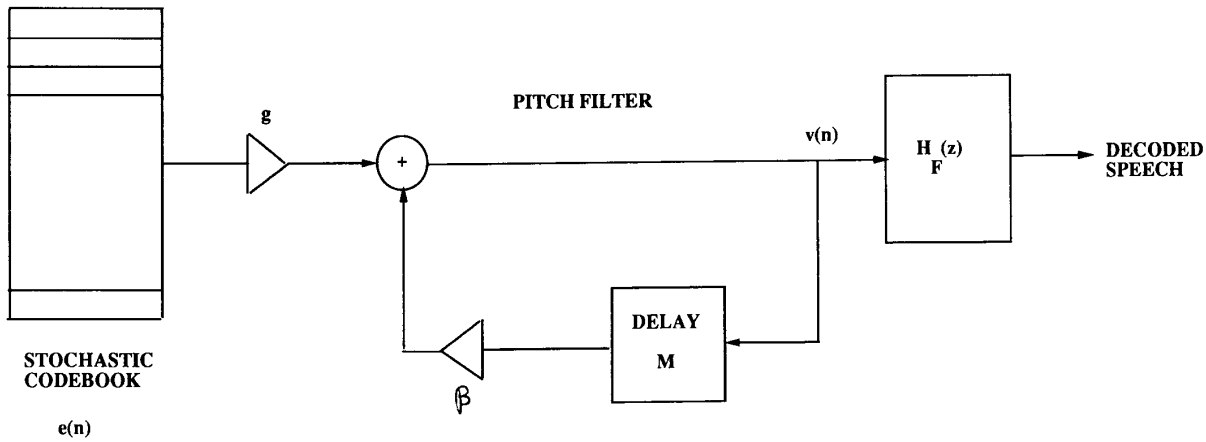

FIGURE 1 - CELP DECODER WITH 1 TAP PITCH PREDICTOR

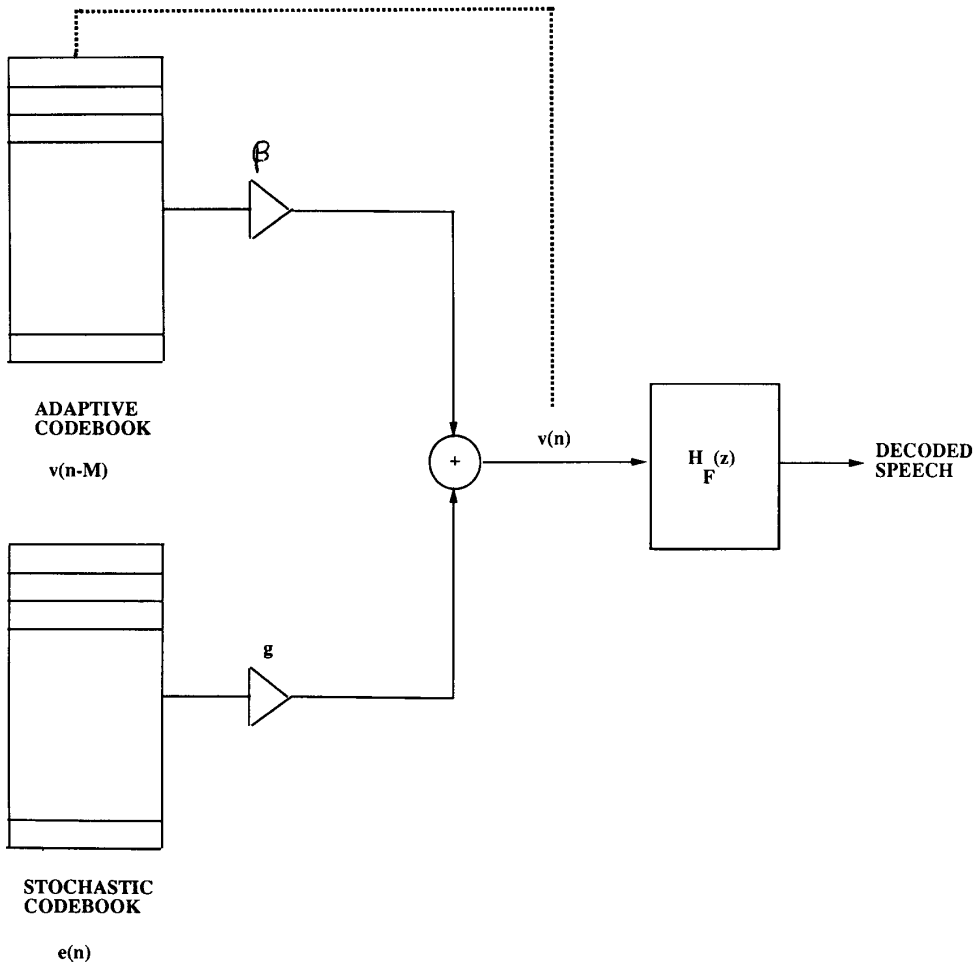

FIGURE 2 - CELP DECODER WITH ADAPTIVE CODEBOOK