효율적인 4.8 KBPS Trellis Exicitation 음성부호화방식

oSangwon Kang* , Sungho Cho* , Insung Lee**
*한양대학교 **한국전자통신연구소

AN EFFICIENT TRELLIS EXCITATION SPEECH CODING AT 4.8 KBPS

oSangwon Kang* , Sungho Cho* , Insung Lee**
*Hanyang University **Electronics and Telecomm. Research Institute

ABSTRACT

In this paper, we present a combination of trellis coded vector quantization and code-excited linear prediction coding, termed trellis excitation coding (TEC), for an efficient 4.8 kbps speech coding system. A training sequence-based algorithm is developed for designing an optimized codebook subject to the TEC structure. Also, we discuss the trellis symbol release rules that avoid excessive encoding delay. Finally, simulation results for the TEC coder are given at bit rate of 4.8 kbps.

I. INTRODUCTION

Good quality speech coding at bit rates under 8 kbps has a growing number of applications for efficient digital transmission and storing of speech data. Traditional waveform coders can provide good speech at bit rates above 16 kbps, but their performance drops rapidly for lower rates. On the other hand, traditional vocoder techniques enable one to encode speech at very low rates, but the perceptual quality is limited, even for relatively large bit rates (4 to 9.6 kbps).

At the lower bit rates, better encoders are obtained by hybrid methods, such as code-excited linear prediction (CELP) [1] and multi-pulse excitation (MPE) [2]. These coders use linear predictive coding (LPC) techniques [3] for removing the short-term correlation of the speech signal. Instead of quantizing the resulting residual signal instantaneously, as in traditional adaptive predictive coding (APC) [4], this signal is quantized on a delayed decision basis using an analysis-by-synthesis approach.

CELP coder is one very promising speech coding system at low bit rates (below about 8 kbps) which we now focus. CELP coding does not require a scalar quantization procedure, but chooses the excitation sequence from a given codebook. Hence, two important research issues in CELP coding are the design and search procedures of a codebook.

Trellis coded quantization (TCQ) [5] is a type of trellis coding that labels the trellis branches with subsets of reproduction symbols. Trellis coded vector quantization (TCVQ) [6] is a generalization of TCQ that allows vector codebook branch labels. Hence the novel feature of TCVQ is the partitioning of an expanded set of vector quantization codewords into subsets and the labeling of the trellis branches with these subsets. In [6], three different structures were given for incorporating vector quantization with TCQ. We consider only the structure 1 formulation in this paper.

Let's consider encoding L-dimensional source vectors at a rate of R_e bits/dimension. If we assume the product R_eL to be an integer, a traditional VQ would have an encoding rate of R_e bits/dimension and 2^{R_eL} output symbols. In the TCVQ structure 1 encoder, we construct a "super" codebook, $\mathcal{S}$, of $2^{(R_e+\tilde{R})L}$ vector codewords, where $\tilde{R}$ is called the "rate expansion factor" (in bits per dimension). Consider an N-state trellis with 2^M branches entering and leaving each trellis state, with M an integer satisfying $M \leq R_eL$. Let $K = \tilde{R}L + M$ and partition the codebook into 2^K subsets, denoted as $S_1, S_2, \ldots, S_{2^K}$. Each subset consists of exactly 2^{R_eL-M} vectors. Each branch of an N-state trellis is labeled with one of the subsets. Given the above structure and an initial state in the trellis, the encoding is performed as follows [6]:

1. For each source vector, $\mathbf{x}$, find the optimal codeword and the corresponding distortion, d_i, in each subset S_i.

2. Consider the branch metric for a branch labeled with subset S_i to be the distortion found in 1, and use the Viterbi algorithm [7] to determine the minimum distortion survivor path through the trellis.

M bits/vector are used to specify the best trellis path. The remaining $(R_eL - M)$ bits/vector are used to specify the codeword in the selected (branch) subset. Thus, R_eL bits/vector are used to specify the sequence of codewords corresponding to the minimum distortion path through the trellis. That is, the actual transmission rate is R_e bits/dimension.

For memoryless sources, the TCVQ scheme offers significant improvement over vector quantization. In this paper, we develop an effective low-rate speech coder that incorporates TCVQ in the CELP structure. The resulting system is referred to as a trellis excitation coding (TEC) system. The encoding rate under consideration is 4.8 kbps.

II. TRELLIS EXCITATION CODING

The trellis excitation coder is a hybrid speech coder which uses the analysis-by-synthesis approach with the trellis coding. The basic structure of the TEC coder is shown in Fig. 1. The short-term predictor $A(z)$ is described as

$$A(z) = \sum_{k=1}^{p} a_k z^{-k}, \qquad (1)$$

where $a_1, a_2, \ldots, a_p$ are the p th order LPC parameters. The long-term predictor $P(z)$ is represented by

$$P(z) = b_1 z^{-(M_p-1)} + b_2 z^{-M_p} + b_3 z^{-(M_p+1)}, \quad (2)$$

where b_1, b_2, and b_3 are three predictor coefficients and M_p describes a pitch delay in the range 16-143 samples. The speech production model includes two adaptive cascaded LPC synthesis filters $1/A(z)$ and $1/P(z)$, a TCVQ "super" codebook of excitation vectors $\mathbf{c}_j$, and a gain term G. A transfer function appropriate for the weighting filter is $W(z) = A(z)/A(z/\gamma)$ [8], where $A(z)$ includes the quantized predictor parameters and $0 \leq \gamma \leq 1$. The purpose of the perceptual weighting filter $W(z)$ is to shape the spectra of the noise signal so that it is similar to the spectra of the speech signal, thus using the masking effect of the human ear [9]. Once the prediction coefficients and pitch period are found for each frame and encoded, the original speech vectors $\mathbf{s}(n)$ within that frame are encoded. Suppose an N-state trellis has been searched using the Viterbi algorithm to a time index of $n-1$. The memory hangover vector of the cascaded filters $1/P(z)$ and $1/A(z)$

$$\hat{\mathbf{s}}^i(n) = [\hat{s}^i_1(n), \hat{s}^i_2(n), \ldots, \hat{s}^i_L(n)]^T$$

is computed based on the survivor path ending at node i at time $(n-1)$. There are N different vectors, one for each trellis state. Assuming the pitch delay M_p is such that $M_p \geq L+1$, the memory hangover vector of the pitch predictor is described as

$$\hat{\mathbf{v}}^i(n) = [\hat{v}^i_1(n), \hat{v}^i_2(n), \ldots, \hat{v}^i_L(n)]^T \quad (3)$$

with

$$\hat{v}^i_l(n) = \sum_{j=-J}^{J} b_j v^*_j, \quad (4)$$

where

$$v^*_j = \begin{cases} \hat{v}^i_{l-l^*}(n-k^*) & \text{for } l > l^* \\ \hat{v}^i_{L-l^*+l}(n-k^*-1) & \text{for } l \leq l^* \end{cases}$$

$k^* = \lceil \frac{M_p+j}{L} \rceil$, $l^* = M_p + j - k^* L$, and the vector sequence $\{\hat{\mathbf{v}}^i(n-k)\}$ is the encoded version of the pitch predictor output, related to the survivor path ending at node i at time $n-1$. The memory hangover vector $\hat{\mathbf{s}}^i(n)$ of the cascaded filters $1/P(z)$ and $1/A(z)$ is then written as

$$\hat{s}^i_l(n) = \begin{cases} \sum_{j=1}^{l-1} a_j \hat{s}^i_{l-j}(n) + \sum_{j=0}^{L-1} a_{j+l} \hat{s}^i_{L-j}(n-1) \\ \quad +\hat{v}^i_l(n) \qquad \text{for } L \geq p \\ \sum_{j=1}^{l-1} a_j \hat{s}^i_{l-j}(n) + \sum_{j=0}^{L-1} a_{j+l} \hat{s}^i_{L-j}(n-1) \\ \quad + \cdots + \sum_{j=0}^{L-1} a_{j+l+mL} \hat{s}^i_{L-j}(n-1-m) \\ \quad +\hat{v}^i_l(n) \qquad \text{for } L < p, \end{cases} \quad (5)$$

where $m = \lceil \frac{p-l}{L} \rceil$, and the vector sequence $\{\hat{\mathbf{s}}^i(n-k)\}$ is the encoded version of the input speech vector sequence $\{\mathbf{s}(n-k)\}$, related to the survivor path ending at node i at time $(n-1)$. The vector $\mathbf{z}^i(n)$ is determined by subtracting the memory hangover output $\hat{\mathbf{s}}^i(n)$ from $\mathbf{s}(n)$. $\hat{\mathbf{z}}^j(n)$ is the reconstructed vector generated by each TCVQ subset codeword $\mathbf{c}_j$ scaled by the gain G. The weighted prediction error is then given by

$$\begin{aligned} \mathbf{e}^i_j(n) &= \mathbf{W}(\mathbf{z}^i(n) - \hat{\mathbf{z}}^j(n)) \\ &= \mathbf{z}^i_w(n) - \hat{\mathbf{z}}^j_w(n), \end{aligned} \quad (6)$$

where $\mathbf{W}$ is a L by L lower triangular matrix described in terms of the impulse response $w(n)$ of the weighting filter, and $\mathbf{z}^i_w(n)$ and $\hat{\mathbf{z}}^i_w(n)$ are the frequency weighted versions of the vectors $\mathbf{z}^i(n)$ and $\hat{\mathbf{z}}^j(n)$, respectively. The weighted synthesis filter output is

$$\hat{\mathbf{z}}^j_w(n) = G\mathbf{H}\mathbf{c}_j, \quad (7)$$

where $\mathbf{H}$ is a L by L lower triangular matrix with terms determined by the impulse response, say $h(n)$, of the combined pitch, formant, and weighting filters. For a given transition branch in the trellis, the optimal TCVQ subset codeword is determined to minimize the squared Euclidean distance $\|\mathbf{e}^i_j(n)\|^2_2$. Since the vectors $\mathbf{z}^i_w(n)$ only need be computed once for each trellis transition, their computation is only a small part of the encoder's computational complexity. The gain parameter G is obtained by computing the root mean square value of the forward prediction error for each group of L_g consecutive speech samples.

Assume an N-state trellis is used to encode to a vector index of $n-1$. Given the survivor paths ending at time $n-1$, the N survivor paths at time n can be determined as follows. Let $d^i_{n-1}(\mathbf{z}_w, \hat{\mathbf{z}}_w)$ be the overall distortion related to the survivor path ending at node i at time $n-1$. Assume there are 2^M branches labeled with subsets entering and leaving each node. We denote the subset associated with the branch leaving node i and entering node k as S^k_i. Let the 2^M nodes at time $n-1$ with branches entering node k be $i_1, i_2, \ldots, i_{2^M}$. The updated survivor path ending at node k at time n is determined by finding for each branch entering state k at time n, the best subset codeword that minimizes the distortion between the weighted input vector $\mathbf{z}^i_w(n)$ and the weighted synthesis vector $\hat{\mathbf{z}}^i_w(n)$. Let this best codeword from subset S^k_i be $\mathbf{c}^k_i$. Thus, $\mathbf{c}^k_i$ is the codeword $\mathbf{c}$ from S^k_i that minimizes

$$\|\mathbf{z}^i_w(n) - \hat{G}\mathbf{H}\mathbf{c}\|^2_2, \quad i = i_1, i_2, \ldots, i_{2^M}.$$

Then, we compute the overall distortion associated with each of the 2^M possible paths to node k at time n and select the path with the minimum distortion as the updated survivor path ending at state k. Thus, the TCVQ minimization procedure computes

$$d^k_n(\mathbf{z}_w, \hat{\mathbf{z}}_w) = \min_{i \in \{i_1, i_2, \ldots, i_{2^M}\}} (d^i_{n-1} + \|\mathbf{z}^i_w(n) - \hat{G}\mathbf{H}\mathbf{c}^k_i\|^2_2). \quad (8)$$

After all N survivor paths at time n are determined, the time index is incremented and the process is repeated until a certain depth corresponding to positive integer multiples of the vector dimension, L. Then, M bits per vector to specify the best trellis path and $R_i L - M$ bits per vector to indicate the subset codeword are transmitted. In the receiver, the transmitted data produce a sequence of codewords. Each codeword is scaled by $\hat{G}$, the corresponding quantized gain. The resulting signal is passed through the pitch and formant predictors to produce the reconstructed version of the input speech vector $\mathbf{s}(n)$.

III. TEC OPTIMAL CODEBOOK DESIGN

In this section, we will introduce a procedure for designing the optimal "super" codebook, subject to the TEC structure, by applying the generalized Lloyd algorithm [10] to a training sequence of input vectors $\mathbf{z}(n)$. For each optimization iteration, the updated codebook is optimal for the current input sequence in the sense that the perceptual weighted distortion between the input vectors $\mathbf{z}(n)$ and the reconstructed vectors $\hat{\mathbf{z}}(n)$ is minimized. However, since any change of the codebook alters the input sequence, convergence of the design algorithm can not be assured.

As an initial "super" codebook, we use a random codebook in which each codeword is constructed of samples of an unit-variance white Gaussian process. We chose the Gaussian distribution since the probability density function of the prediction error sequence (after both formant and pitch predictions) is reasonably modeled as white and Gaussian [11]. Given the initial codebook, the initial subsets are formed based on Ungerboeck's signal set partitioning method [12].

The codebook optimization algorithm of the TEC encoder is described as follows and is similar to the algorithm for structure 1 optimization in [6]. Given a training input sequence, X and an initial codebook, $\mathcal{S}^1$, set $k = 1$, and at iteration k:

1. Encode the training sequence using the Viterbi algorithm and the TCVQ structure with codebook $\mathcal{S}^k$. Denote the resulting distortion as

$$E(k) = \frac{1}{\|\mathcal{X}\|} \sum_{\mathbf{z}(n)\in\mathcal{X}} \|\mathbf{z}_w(n) - \hat{G}\mathbf{H}\mathbf{c}_j^k\|_2^2.$$

2. Partition the input vectors $\mathbf{z}(n)$ associated with the training sequence into sets Q_j^k, so that $\mathbf{z}(n) \in Q_j^k$, if and only if its weighted version $\mathbf{z}_w(n)$ was encoded as $\hat{G}\mathbf{H}\mathbf{c}_j^k$.

3. Update the TCVQ codebook as $\mathcal{S}^{k+1}$ by

$$\mathbf{c}_j^k = \Big(\sum_{\mathbf{z}(n)\in Q_j^k} \hat{G}^2\mathbf{H}^T\mathbf{H}\Big)^{-1} \sum_{\mathbf{z}(n)\in Q_j^k} \hat{G}\mathbf{H}^T\mathbf{z}(n).$$

Set $k = k + 1$ and go to step 1.

Since the input vectors $\mathbf{z}(n)$ computed from the speech signal are dependent on the current excitation codebook, $E(k)$ is not guaranteed to decrease monotonically with k. Typically, a large decrease of E is obtained in the first few iterations. The optimization process can be halted by a suitable termination criteria. In this paper, the process is stopped after 12 iterations. The codebook $\mathcal{S}^k$ generating the minimum distortion is selected as the "optimal" TCVQ codebook.

IV. SYMBOL RELEASE RULE

If we search the entire trellis before releasing any symbols, the best performance can be achieved. However, this search introduces excessive encoding delay, and a trellis symbol release rule corresponding to a suboptimum strategy is required in a practical speech coding system.

The symbol release rule considered herein is similar to that in [6] and is described as follows. Let $K_r \geq 1$ and $K_d \geq 0$ define, respectively, the number of branch symbols released and the depth of the trellis search at which a hard decision is made. Suppose the trellis encoding has proceeded to a sample $n = jK_r$, j an integer. The survivor path with the minimum distortion is traced back $K_r + K_d$ branches, and the K_r branch symbols (codewords) corresponding to samples $jK_r - K_d - K_r, \ldots, jK_r - K_d - 1$ are released. Define the node that the best survivor path at sample jK_r passes through at sample $jK_r - K_d$ as $z^*(jK_r - K_d)$. Each survivor path at sample jK_r is traced back to sample $jK_r - K_d$. If the resulting node is not $z^*(jK_r - K_d)$, then the associated survivor metric at sample jK_r is set to ∞. If the resulting node is $z^*(jK_r - K_d)$, then no change is made. This effectively "prunes" all survivor paths that would lead to an inconsistent trellis path. The performance of the coder is expected to increase at the expense of longer encoding delay as K_r or/and K_d increase. Hence, K_r and K_d are generally selected as large as permissible in each application.

V. EXPERIMENTAL RESULTS

In this section, we evaluate the effectiveness of TEC coders at low bit rates. Both Korean and English sentences are used to evaluate encoding performance. In each language, Sentences 1, 3, 6, and 7 were used to design the coder, and sentences 1-5 were used to evaluate the coder performance. The TEC coding system encodes three different parameters (gain, formant, and pitch parameters) as the side information. The quantization levels for the scalar quantization of gain parameters were designed by applying the generalized Lloyd algorithm to a training sequence. The formant coefficients are first transformed to LSP parameters [13], and then predictive trellis coded quantization (TCQ) scheme [16] is applied to quantize the LSP parameters. The pitch coefficients are treated as vectors and encoded by vector quantization techniques.

Table 3 presents the simulation results for the unweighted and weighted 4.8 kbps TEC coders. The simulations using Korean sentences produced the average SNR and SEGSNR of 12.48 and 12.24 dB, respectively. The simulations using English sentences produced the average SNR and SEGSNR of 12.2 and 11.23 dB, respectively. To assess the subjective quality of the TEC encoder, a TEC reconstructed sentence was compared with that of a μ-law PCM system [15] ($\mu = 255$) operating at bit rates of 3 through 8 bits per sample. Informal listening tests indicate the 4.8 kbps TEC system performs roughly between the 5-bit and 6-bit μ-law PCM with $\mu = 255$. Informal listening tests revealed that the advantage of error weighting is small, but can be heard. For example a "warble" noise in the word "rob" of the English sentence 4 was reduced with the weighting filter. An empirically "optimal" value for γ was found to be 0.8.

VI. CONCLUSIONS

An effective 4.8 kbps speech coding system, called trellis excitation coding (TEC), was introduced, which incorporates TCVQ [6] in the CELP structure [1]. We formulated such a combination of TCVQ and CELP. A training sequence-based algorithm was then introduced for iteratively designing the optimal codebooks subject to the TEC structure. Also, we described the trellis symbol release rules. Then, simulation results for the efficient 4.8 kbps TEC coder was presented in terms of SNR and SEGSNR, and the informal listening tests.

REFERENCES

[1] M. R. Schroeder and B. S. Atal, "Code-Excited Linear Prediction (CELP): High-quality speech at very low bit rates," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 1985, pp. 252-255.

[2] B. S. Atal and J. R. Remde, "A new model of LPC excitation for producing natural-sounding speech at low bit rates," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Apr. 1982, pp. 614-617.

[3] J. Makhoul, "Linear prediction: A tutorial review," *Proc. IEEE*, vol. 63, pp. 561-580, April 1975.

[4] B. S. Atal and M. R. Schroeder, "Adaptive predictive coding of speech signals," *Bell Syst. Tech. J.*, vol. 49, pp. 1973-1986, Oct. 1970.

[5] S. Kang, I. Lee, and K. Han, "An efficient vocoder for digital cellular system," *Korean Institute of*

Comm. Sciences, vol. 18, pp. 1348-1357

[6] M. W. Marcellin and T. R. Fischer, "Trellis coded quantization of memoryless and Gauss-Markov sources," *IEEE Trans. Commun.*, vol. 38, pp. 82-93, Jan. 1990.

[7] T. R. Fischer, M. W. Marcellin, and Min Wang, "Trellis coded vector quantization," submitted to *IEEE Trans. Inform. Theory*.

[8] G. D. Forney, Jr., "The Viterbi algorithm," *Proc. IEEE (Invited Paper)*, vol. 61, pp. 268-278, Mar. 1973.

[9] 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, June 1979.

[10] M. R. Schroeder, B. S. Atal, and J. L. Hall, "Optimizing digital speech coders by exploiting masking properties of the human ear," *J. Acoust. Soc. Amer.*, pp. 1647-1652, Dec. 1979.

[11] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun.*, vol. COM-28, pp. 84-95, Jan. 1980.

[12] B. S. Atal, "Predictive coding of speech at low bit rates," *IEEE Trans. Commun.*, vol. Com-30, pp. 600-614, April 1982.

[13] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 55-67, Jan. 1982

[14] F. Itakura, "Line spectrum representation of linear predictive coefficients of speech signals," *J. Acoust. Soc. Amer.*, vol. 57, supplement no. 1, S35(A). 1975.

[15] N. Sugamura and N. Farvardin, "Quantizer design in LSP speech analysis-synthesis," *IEEE Journ. Selected Areas in Commun.*, vol. 6, pp. 432-440, Feb. 1988.

[16] N. S. Jayant and P. Noll, *Digital Coding of Waveforms*, Englewood Cliffs, NJ: Prentice-hall, Inc., 1984.

[17] K. T. Malone, "Adaptive predictive coding of speech using trellis coded vector quantization" Ph.D. dissertation, Texas A&M University, December 1989.

[18] J. D. Gibson and W. W. Chang, "Fractional rate multi-tree speech coding," *IEEE Global Telecomm. Conf.*, Vol. 2, pp. 1906-1910, Dec. 1989.

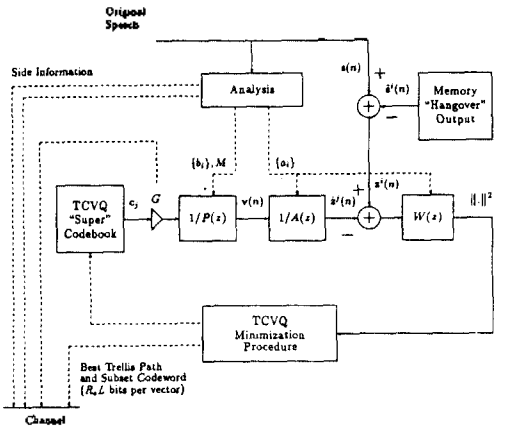

Figure 1. Block diagram of the TEC coder.

Table 1. Korean sentences Used to Evaluate Encoding Performance.

1. 미는 피부 한 껍질 차이입니다.	(Female)
2. 지나친 흡연은 건강을 해칩니다.	(Female)
3. 이번겨울은 예년과 달리 포근합니다.	(Male)
4. 과학기술은 경제발전의 원동력이다.	(Male)
5. 일에서 십까지의 합은 오십오 입니다.	(Male)
6. 어제 산 물건이 벌써 고장이 났다.	(Male)
7. 올림픽은 전 인류의 축제입니다.	(Female)

Table 2. English sentences Used to Evaluate Encoding Performance.

1. The pipe began to rust while new.	(Female)
2. Add the sum to the product of these three.	(Female)
3. Oak is strong and also gives shade.	(Male)
4. Thieves who rob friends deserve jail.	(Male)
5. Cats and dogs each hate the other.	(Male)
6. Almost everything involved making the child mind.	(Male)
7. The trouble with swimming is that you can drown.	(Female)

Table 3. The Performance of the 4.8 kbps TEC Coder.

		SNR/SEGSNR (dB)				
		Sentence Number				
	γ	1	2	3	4	5
Korean	1	13.28/12.85	13.31/13.23	12.11/11.86	10.91/10.87	12.78/12.39
	0.8	12.52/12.32	12.91/12.34	11.72/11.03	10.71/10.10	12.18/11.92
English	1	13.78/12.46	13.76/12.44	10.93/9.85	10.43/10.27	12.09/11.12
	0.8	13.18/11.50	13.32/11.58	10.47/8.99	9.91/9.86	11.67/10.58