3rd International Conference on
Spoken Language Processing (ICSLP 94)
Yokohama, Japan
September 18-22, 1994

Complexity Reduction Methods for Vector Sum Excited Linear Prediction Coding

Sung Joo Kim, Seung Jong Park, Yung Hwan Oh

Department of Computer Science
Korea Advanced Institute of Science and Technology
373-1, Kusong-dong, Yusong-gu, Taejon 305-701 Korea

ABSTRACT

The vector sum excited linear prediction (VSELP) coding gives high quality of synthetic speech at bit rates as low as 4.8kbps, but its computational complexity is prohibitive for real-time applications. In this paper, we propose three methods to reduce the computations in VSELP coding. First, we use the overlapped sparse codebook for the basis vectors. Second, we introduce the preprocessing step to the stochastic codebook search procedure. It decides some combination coefficients of basis vectors using heuristics so that the search space decreases. Third, some candidates are preselected before the adaptive codebook search procedure by comparing them with the ideal excitation sequence. We develop a 4.8kbps coder using all the proposed methods and perform the quality test. It has been shown that the proposed coder retains good quality of synthetic speech and it is more than twice as fast as the original coder.

I. INTRODUCTION

The vector sum excited linear prediction (VSELP) coding is one of the most efficient speech coding methods. It falls into the class of the vector excitation coding the same as the code excited linear prediction (CELP) coding[1]. The major feature of VSELP coding is its excitation codevector that is constructed as a linear combination of basis vectors. Due to this feature, the codebook search of VSELP coding is very efficient than that of the original CELP coding. Moreover, the excitation codewords for VSELP coding are more robust to bit errors caused by channel errors.

We can code the speech signal at bit rate as low as 4.8kbps using VSELP coding and the quality is good enough to adopt this method for telephony. In fact VSELP coding was selected as the standard for use in North American digital cellular telephone systems. Nevertheless, it still involves the huge complexity. So we need a certain 40mips DSP chip, which can do floating-point multiply-add operations, to implement the full 8kbps VSELP coder[2]. Therefore, it is worth while reducing the complexity of VSELP coding for the low cost implementation.

In this paper, we present three methods to reduce the computational complexity of VSELP coding so that the coder with reduced computations can be implemented with less cost. The key ideas of the proposed methods are as follows.

(1) use of overlapped sparse basis vectors
(2) heuristic decision over the combination coefficients of basis vectors
(3) preselection of the candidate adaptive codevectors

It should be noticed that the overlapped sparse codebook has been used as the stochastic codebook of CELP coding[3]. It reduces the computations for filtering every codevector. In the case of the preselection method using the ideal excitation sequence, it has also been applied to the stochastic codebook search of CELP coding[4]. However, in this study, we apply these methods to VSELP coder with the other complexity reduction method.

At the end of this paper, we describe the implementation of a fast VSELP coder and the results of its performance evaluation.

II. VSELP CODING

VSELP coder uses the analysis-by-synthesis method to select the excitation codevectors from given codebooks. In this work, we use one stochastic codebook for the randomness and one adaptive codebook for the periodicity of the excitation signal. To shape the spectral envelope of the excitation signal, VSELP coder uses a LPC all-pole filter as the synthesis filter. The filter coefficients are estimated in every frame and updated once a subframe through interpolation, where a frame consists of four subframes. The codebook gains and indices are selected in every subframe to minimize the total weighted error.

Because of the computational load, the VSELP coder searches the codebooks sequentially. It searches the adaptive codebook and the stochastic codebook in that order. To achieve the joint optimization during the sequential search, the weighted stochastic codevectors are orthogonalized to the previously selected weighted excitation. After all the codevectors are selected, the gains are evaluated simultaneously. Fig. 1 displays the block diagram of the codebook search procedure.

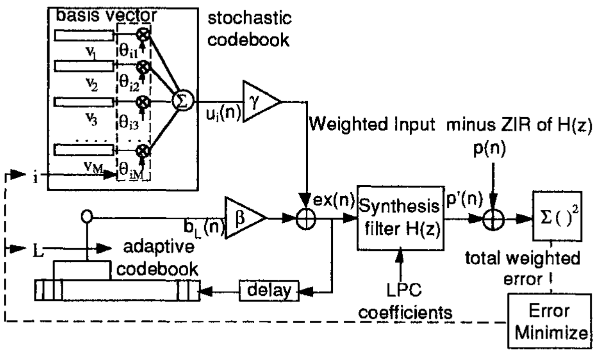

Fig. 1. VSELP excitation search procedure

As you can see in Fig. 1, M basis vectors, v_m, construct the stochastic codebook and a stochastic codevector u_i can be defined as (2.1).

$$u_i(n) = \sum_{m=1}^{M} \theta_{im} v_m(n)\,, \tag{2.1}$$

for $0 \le i \le 2^M\text{-}1$ and $0 \le n < N$.

$$\theta_{im} = \begin{cases} +1 & \text{if (bit m of codeword i) = 1} \\ -1 & \text{if (bit m of codeword i) = 0} \end{cases}$$

It is useful to let p(n) be the target signal of excitation search procedure that is produced by subtracting zero input response of H(z) from perceptually weighted input speech. Then, the whole search procedure selects the optimal codwords L and i, subject to the minimum mean squared error criteria (2.2).

$$E_{L,i} = \sum_{n=0}^{N-1} \left(p(n) - \beta b'_L(n) - \gamma f_i(n)\right)^2, \tag{2.2}$$

where b'_L and f_i are the zero state responses of H(z) to an adaptive codevector b_L and a stochastic codevector u_i, respectively. (2.2) implies that we must search all the combinations of the adaptive and the stochastic codevectors to find out the optimal excitation signal, but this strategy involves too many computations. Therefore, the search procedures are executed separately and sequentially in real implementation. As a result, we must be satisfied with the suboptimal excitation signal. However, the selected codevectors are jointly optimized through orthogonalization.

First, the adaptive codebook search procedure selects the codeword L which minimizes E'_L of (2.3), where β' is optimal gain for each weighted codevector b'_L.

$$E'_L = \sum_{n=0}^{N-1} \left(p(n) - \beta' b'_L(n)\right)^2. \tag{2.3}$$

Because of the free gain term β', the adaptive codebook search procedure can be regarded as selecting the codevector b_L which minimizes $\cos^2 \omega_{pb'_L}$, where $\omega_{pb'_L}$ is the angle between the two vectors p and b'_L.

Every weighted stochastic codevector must be orthogonalized to b'_L after finding the optimal adaptive codevector b_L. Fortunately, for the VSELP codebook, this task is completed by orthogonalizing only M weighted basis vectors q_m to b'_L. Let q'_m be q_m after orthogonalization to b'_L. Then f'_i, that is f_i after orthogonalization to b'_L, can be expressed as:

$$f'_i(n) = \sum_{m=1}^{M} \theta_{im} q'_m(n)\,, \tag{2.4}$$

for $0 \le i \le 2^M\text{-}1$ and $0 \le n < N$

Now we can find the optimal excitation codeword i which minimizes:

$$E'_i = \sum_{n=0}^{N-1} \left(p(n) - \gamma f'_i(n)\right)^2 \tag{2.5}$$

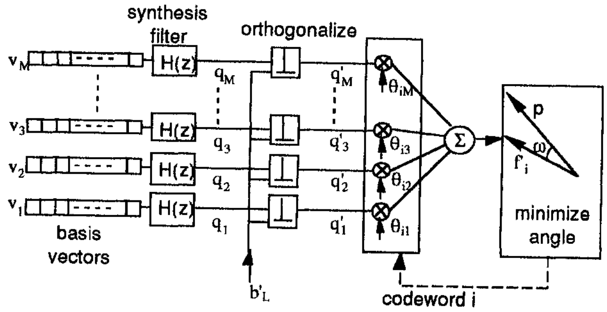

Fig. 2. Stochastic codebook search

where γ is the optimal gain for f'_i. As the case of the adaptive codebook search, the optimal codeword i minimizes $\cos^2 \omega_{pf'_i}$, where $\omega_{pf'_i}$ is the angle between the two vectors p and f'_i. Fig. 2 depicts the stochastic codebook search procedure conceptually.

III. FAST EXCITATION SEARCH METHODS

3.1 Overlapped sparse basis vectors

After analyzing the amount of computations for stochastic codebook search, we can find out that the filtering operations, i.e., convolution operations, for all basis vectors consume almost 40% of the total computations.

It is well known that the overlapped sparse codebook reduces the computations for filtering every codevector greatly, and the coding result is very similar to that of conventional codebook if all codevectors contain random sequences.

As described in [2], the basis vectors of VSELP coder may be the random sequences. So, if the basis vectors are N dimensional, we can generate M basis vectors, v_m, using an overlapped codebook C as (3.1).

$$v_m(n) = C[M\alpha - m\alpha + n], \tag{3.1}$$

where the codebook C is realized by a 1 dimensional array containing $N+\alpha(M-1)$ samples of random values and α is the number of shift between two adjacent basis vectors[3].

Now, we can evaluate the filter response q_m to every basis vector v_m, recursively like (3.2).

$$q_1(n) = \sum_{k=0}^{n} v_1(k)\ h(n-k)\ ,\quad \text{if } 0 \le n < N \tag{3.2}$$

$$q_m(n) = \begin{cases} \sum_{k=0}^{n} v_m(k)\ h(n-k), & \text{if } 0 \le n < \alpha \\ q_{m-1}(n-\alpha) + \sum_{k=0}^{\alpha-1} v_m(k) h(n-k), & \text{if } \alpha \le n < N \end{cases}$$

for $1 < m \le M$

Assuming C contains zeros with probability σ, the total number of multiply-add operations for evaluating every q_m becomes $(1-\sigma)[(N+1)N/2+(M-1)\{\alpha(\alpha+1)/2+(N-\alpha)\alpha\}]$.

We have tested various sets of basis vectors concerning the degree of overlapping and sparsity. The results show that the subjective and objective quality of synthesized speech is retained if α is larger than 5 and σ is smaller than 0.9.

3.2 Heuristic decision over the combination coefficients

The stochastic codebook search procedure calculates all the 2^{M-1} angles between the target signal p and each weighted codevector f'_i to decide the combination coefficients of all basis vectors. Although we can compute those values recursively one by one, it takes about 44.2% of the total computations of the stochastic codebook search procedure.

In this section, we propose the heuristic preprocessing that finds some basis vectors whose combination coefficients can be fixed heuristically. Thus, the search procedure decides just the remaining coefficients after the preprocessing step.

The error criteria of stochastic codebook search can be rewritten as (3.3) using vector notation.

$$E'_i = \| p - \gamma' \sum_{m=1}^{M} \theta_{im} q'_m \|^2 . \tag{3.3}$$

Because the free gain term γ' is multiplied, minimizing (3.3) is equivalent to maximizing the match score of (3.4):

$$\text{match}_i = \frac{<p, f'_i>^2}{\| f'_i \|^2} = \|p\|^2 \cos^2 \omega_{pf'_i} . \tag{3.4}$$

The angle between p and f'_i should be minimized to maximize (3.4) since $\|p\|^2$ is a constant during the codebook search procedure. Reminding that f'_i is a linear combination of q'_m, we can interpret this task as making a vector whose direction is as near as possible to that of target vector by adding or subtracting M given vectors. To do this work, one heuristic method that sets the combination coefficient θ_{im} to the sign of $\cos\omega_{pq'_m}$, provides a reasonable result.

But, if we set every θ_{im} to the sign of $\cos\omega_{pq'_m}$, for $1 \le m \le M$, the quality of synthetic speech degrades too much and the weighted segmental SNR decreases by 1.2 dB. Therefore, there must be some restriction to apply this heuristic decision so that this method determines θ_{im} only if q'_m satisfies a certain condition.

While we compare the sign of $\cos\omega_{pq'_m}$ with θ_{im}, we notice that θ_{im} always equals the sign of $\cos\omega_{pq'_m}$ if the resemblance is big enough. Besides, the resemblance can be measured by SP_m as follows:

$$SP_m = \|q'_m\|^2 \cos^2 \omega_{pq'_m} . \tag{3.5}$$

Using this fact, we can restrict the heuristic decision over the combination coefficient θ_{im}, only if SP_m is bigger than a threshold, or has one of the biggest K values. If the heuristic decision sets K combination coefficients in advance, the other M-K coefficients can be decided using original search procedure with reduced search space of 2^{M-K} combinations instead of 2^{M-1} combinations.

According to the comparison test, the weighted segmental SNR of synthesized speech decreased by only 0.14dB even if we fix six combination coefficients heuristically with K-selection method at the preprocessing step.

3.3 Preselection of the candidate adaptive codevectors

M. Ahmed proposed to preselect the candidates using the ideal excitation sequence before stochastic codebook search[4]. We apply this method to the adaptive codebook search.

Like the stochastic codebook search, the adaptive codebook search chooses the optimal lag L which minimizes:

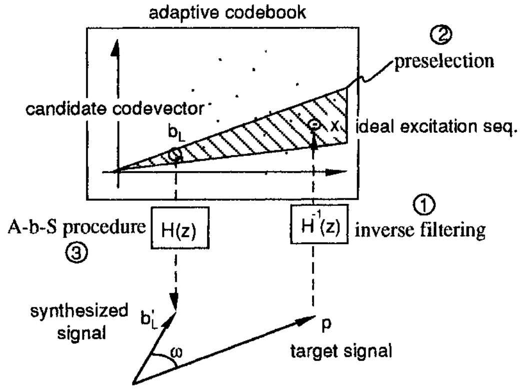

Fig. 3. Preselection of the candidate adaptive codevectors

$$E'_L = \| p - \beta' b'_L \|^2 = \| p - H(\beta' b_L) \|^2, \tag{3.6}$$

where H is a lower triangular matrix, and its i^{th} column has the zero state response of H(z) to $\delta[i]$.

Let x be the ideal excitation that will produce p through the synthesis filter H(z). x can be constructed by solving (3.7) using the back substitution method, while it represents the result of inverse filtering the target signal p.

$$p = Hx \tag{3.7}$$

Then, (3.6) can be rewritten as:

$$E'_L = \| H(x - \beta' b_L) \|^2 = \| H d_L \|^2 . \tag{3.8}$$

So the total weighted error for an adaptive codevector b_L is the power of the filter response to the difference d_L. In most cases, the power of the filter output is proportional to the power of the input. Hence, we can preselect some codevectors bearing the smallest power of the difference d_L, as candidates of the optimal adaptive codevector.

If K candidates are preselected, the whole procedure takes only K+1 filtering operations for the inverse filtering and the A-b-S procedure. We have tested the performance of preselection. The degradation of quality is negligible when it chooses more than 16 candidates out of 128 adaptive codevectors. In Fig. 3, we represent the preselection procedure conceptually.

IV. Design of 4.8kbps fast VSELP coder

4.1 Specifications

Using all the proposed speed-up methods, we implemented a 4.8kbps VSELP coder. Table 1 shows the bit allocations and update timings for our VSELP coder at 8kHz sampling rate. It should be noticed that two bits are reserved in a frame for future expansion. LPC coefficients are converted to LSPs and then quantized in every frame. The adaptive codebook search procedure follows the method for DoD 4.8kbps CELP coder: using noninteger pitch lags, delta searching, hierarchical searching and modified minimization of the weighted error[3]. As to the hierarchical searching, the optimal integer pitch is selected first and then nearby noninteger pitches are tested. Before searching the integer pitches, 16 candidates are preselected as we described in 3.3. In the stochastic codebook search, the proposed coder determines 6 combination coefficients heuristically as mentioned in 3.2.

By the way, it uses an overlapped sparse codebook of $\alpha = 6$ and $\sigma = 0.9$ to generate 12 basis vectors. The basis vectors for the stochastic codebook can be optimized over a training database[2]. In the case of proposed basis vectors, we cannot train them by the algebraic methods as described in [2]. However, we can train the overlapped sparse basis vectors using the steepest descending method and improve the speech quality in the weighted segmental SNR. The adaptive codebook gain and the stochastic codebook gain are quantized independently using nonuniform quantization levels made by LBG algorithm.

4.2 Performance evaluation

At first, accumulating the amount of multiply-add operations for selecting the excitation sequence, we compared the complexity of the proposed coder with the conventional VSELP coder. Fig. 4 displays the ratio of reduced computations caused by three fast searching methods. Additionally, the results of software simulation certified that the coding speed of fast VSELP coder is twice as fast as the origin.

As to the quality test, we compared three types of 4.8kbps speech coder. They were conventional VSELP coder, fast VSELP coder and DoD CELP coder. We adopted the weighted segmental SNR to measure the quality of the synthetic speech and informal listening test was performed.

As you can see in Fig. 5, the fast VSELP coder outperformed the DoD CELP coder and had only a small drop as compared to the conventional VSELP coder. Moreover, there was no significant evidence of the quality degradation during the informal listening test. It was hard to discriminate the synthesized speech of proposed coder from that of original coder.

V. CONCLUSION

We have described three complexity reduction methods for the VSELP coding. Firstly, the overlapped sparse basis vectors reduce the computational load to calculate the filter response to every basis vector. Secondly, the proposed preprocessing step reduces the search space efficiently before applying every linear combination of the basis vectors to the stochastic codebook search. It determines whether the combination coefficient of each basis vector can be fixed using heuristics so that the number of combinations decreases. Finally, to determine the pitch period, only the preselected candidates are applied to the pitch search procedure. The candidates are selected by comparing them with the ideal excitation sequence.

We have implemented the fast VSELP coder with reduced complexity. Also, we have shown the comparison of the fast VSELP coder with the conventional VSELP coder and DoD CELP coder. The fast VSELP coder reduces the computations for searching the adaptive and stochastic codebook by 48.2% and 79.2% respectively, compared with those of the original method. A slight degradation of speech quality, compared with the conventional VSELP coder, is scarcely discriminatory. Instead, it shows a superior quality of speech to that of DoD CELP coder.

REFERENCES

[1] G. Davidson, M. Yong, A. Gersho, "Real-time vector excitation coding of speech at 4800 bps", Int. Conf. Acoust. Speech. Signal Processing, 1987, pp.2189~2192

[2] I. A. Gerson and M. A. Jasiuk, "Vector Sum Excited Linear Prediction (VSELP) Speech Coding at 8kbps", Int. Conf. Acoust. Speech. Signal Processing, 1990, pp.461~464

[3] Federal Standard 1016, "Analog to Digital Conversion of Radio Voice by 4.8kbps CELP", Feb. 14, 1991

[4] M.E. Ahmed, "Fast Methods for Code Search in CELP", IEEE trans. speech and audio processing, Vol. 1, No. 3, 1993, pp.315~325

Table 1. Bit allocation for 4.8kbps fast VSELP coder

	linear predictor	adaptive codebook	stochastic codebook
update	30 ms	7.5 ms	7.5 ms
parameter	10 LSPs	256 codeword	2048 codeword
bits per frame	34 (3,4,4,4,4, 3,3,3,3,3)	index: 8,6,8,6 gain: 5x4	index: 11x4 gain: 4x4

(Note : two bits per frame are unused.)

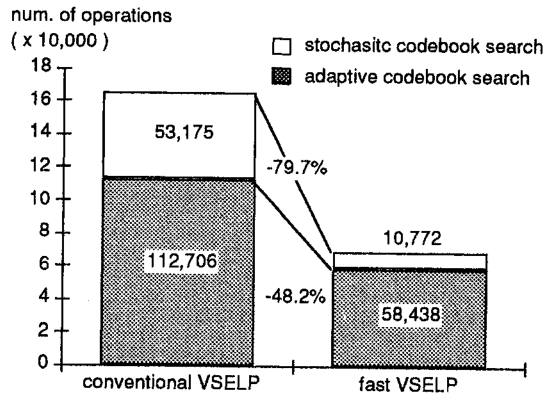

Fig. 4. Comparison of the computational complexity

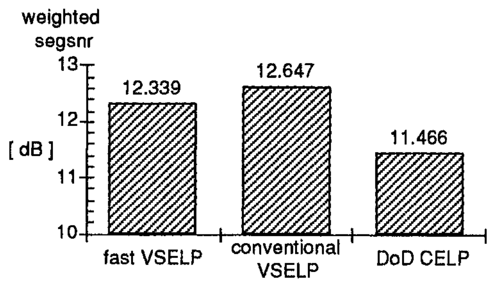

Fig. 5. Comparison of the speech quality