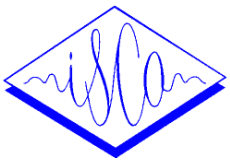


6th International Conference on Spoken Language Processing (ICSLP 2000)
Beijing, China
October 16-20, 2000

AN EFFICIENT CODEBOOK SEARCH ALGORITHMS FOR EVRC

Sung-Kyo Jung, Yong-Soo Choi †, *Young-Cheol Park* ✱, *and Dae-Hee Youn*

ASSP Lab., Dept. of Electrical and Computer Engin., Yonsei University
134 Shinchon-dong Sudaemoon-ku Seoul 120-749, Korea
† Information Systems R&D Lab., LG Information & Communications Ltd.
60-393, Kasan-dong, Kumchun-gu, Seoul, 153-023, Korea
✱ Center for Signal Processing Research, Yonsei University

E-mail : *skjung@lethe.yonsei.ac.kr*

ABSTRACT

In this paper, a fast implementation algorithm for EVRC encoder is presented. Among the routines of CELP-type speech coders, the fixed codebook search is the most computationally demanding. The fast algorithm developed in this study mainly concerns the fixed codebook search procedure. For efficient codebook search, two efficient schemes are developed: limiting the number of possible combination of pulse positions, and using a reduced-tap FIR filter instead of the impulse response of the weighted synthesis filter. The proposed EVRC algorithm was implemented using a fixed-point DSP, and the new EVRC algorithm was subjectively evaluated. It was found that the new FCB search routine could be implemented with only 24.3 % of the original one. Informal subjective tests confirmed that the speech quality of the proposed algorithm was indistinguishable from the original EVRC.

1. INTRODUCTION

IS-127 EVRC [1] has been adopted as a standard speech coder for the CDMA digital cellular system in North America and Korea, and has been in service with 8 kbps Qualcomm CELP (QCELP) [2]. Speech quality of EVRC is comparable to the 13 kbps QCELP [3] serviced for the personal communication service (PCS). Also, EVRC together with the adaptive multirate (AMR) coder [4], has been selected as a candidate for the speech coder standard of 3rd generation mobile communication, known as IMT-2000.

EVRC encoding algorithm is based on the relaxation CELP (RCELP) [5]. It is appropriately modified in order to fit for the variable rate feature and robustness in wireless CDMA environment. While the conventional CELP attempts to match the original speech signal exactly, RCELP tries to match a time-warped version of the original speech signal that conforms to a simplified pitch contour. In such a way, it can be reduced the number of bits needed to encode the pitch information. However, this advantage is obtained with heavy computational operations.

In this paper, we propose an efficient codebook search algorithm for EVRC. Since the fixed codebook (FCB) search is the most computationally demanding among functional routines of EVRC, the proposed algorithm mainly concerns simplifying the FCB search procedure. However, even with the simplification, the quality of the reconstructed speech should not be degraded. For this end, two efficient schemes are developed: limiting the number of possible combination of pulse positions, and using a reduced-tap FIR filter instead of the impulse response of the weighted synthesis filter. Before the FCB search iteration is started, M positions having large amplitudes among the reference samples are selected as possible locations of the excitation pulse. By this pre-selection process, the 4×4×(11×11) FCB search iteration is reduced down to 4×4×(M×11). Based on our experiments, M could be down to 3 with a slight SNR drop of 0.13 dB, which results in (8/11) computational savings.

For the analysis-by-synthesis scheme of FCB search, EVRC uses the impulse response of the weighted synthesis filter with a length of one subframe. Considering the iterative feature of the FCB search, the computational reduction may be obtained by truncating the impulse response. However, simple truncation of the impulse response will result in noticeable quality degradation of the synthesized speech. In order no to have the quality degradation, we modify the weighted synthesis filter to one having a shorter tail. Later, the length of the impulse response is further shortened by introducing the extremely truncated impulse response (ETIR) scheme [9]. The modified synthesis filter is in parallel to the method used in G.729A speech coder.

2. IS-127 EVRC

EVRC speech coder is based on RCELP algorithm which employs the generalized analysis-by-synthesis scheme [6]. Unlike conventional CELP encoders, RCELP matches time-warped version of original speech signal by a simplified pitch contour. This pitch contour is obtained via the linear interpolation of the integer pitch that is estimated in an open-loop fashion for every frame. While a fractional pitch is transmitted in conventional CELP, the integer pitch is transmitted over each frame of speech in RCELP. Consequently, less bits are used for the pitch and more bits are allocated to the FCB excitation encoding and the channel error protection.

Using a rate determination algorithm, EVRC encoder selects one out of three encoding rates: Rate 1 (8.55 kbps), Rate 1/2 (4 kbps), and Rate 1/8 (800 bps). At Rate 1/2 and Rate 1, the encoder uses the RCELP algorithm to match a time-warped version of the original speech. At Rate 1/8, the encoder just characterizes its energy contour.

The FCB in EVRC has an algebraic codebook structure designed using an Interleaved Single-Pulse Permutation (ISPP) scheme. Algebraic codebook structure has advantages in terms of storage, search complexity, and robustness. Each codebook vector of length 55 contains 8 and 3 non-

zero pulses for Rate 1 and Rate 1/2 respectively. For Rate 1, the 55 positions in a single subframe are divided into 5 tracks, where each contains two pulses, as shown in Table 1.

Track	Positions
T_0	0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50
T_1	1, 6, 11, 16, 21, 26, 31, 36, 41, 46, 51
T_2	2, 7, 12, 17, 22, 27, 32, 37, 42, 47, 52
T_3	3, 8, 13, 18, 23, 28, 33, 38, 43, 48, 53
T_4	4, 9, 14, 19, 24, 29, 34, 39, 44, 49, 54

Table 1. Potential positions of individual pulses in the algebraic codebook (Rate 1)

Among the 5 tracks, 3 are allocated with two pulses each and 2 are allocated with one pulse each. This accounts for a total of 8 pulses. The single-pulse tracks can be either T_3-T_4, T_4-T_0, T_0-T_1, or T_1-T_2, as described in Table 2. The choice of single-pulse track is encoded with 2bits. The positions of the pulses in the 2 single-pulse tracks are encoded with 7 bits and their signs are encoded with 2 bits. For each double-pulse track, both positions and signs of the two pulses are encoded with 8 bits. Thus this gives a total of 35 bits.

Double-pulse track order	Single-pulse track order	Codeword
T_0-T_1-T_2	T_3-T_4	00
T_1-T_2-T_3	T_4-T_0	01
T_2-T_3-T_4	T_0-T_1	02
T_3-T_4-T_0	T_1-T_2	11

Table 2. Codeword for the track orders.

The optimal pulse positions are determined using an iterative depth-first tree search strategy. In this approach the 8 pulses are grouped into 4 pairs of pulses. The pulse positions are determined sequentially one pair at a time. In the first iteration, the single-pulse tracks are T_3 and T_4. The search process shall be repeated for other 3 iterations, by assigning the single-pulse tracks to T_4-T_0, T_0-T_1, and T_1-T_2 respectively. The codeword used to represent the various chosen tracks are given in Table 2.

The algebraic codebook is searched on a subframe basis for the best code vector $\mathbf{c}_k$ and gain g to minimize the weighted mean-squared error between the original and synthesis speech given as

$$MSE_{FCB} = \left\| \mathbf{x}_w - g\mathbf{H}\mathbf{c}_k \right\|^2, \qquad (1)$$

where $\mathbf{x}_w$ is the perceptual domain target signal consisting of the weighted speech after subtracting the zero-input response of the weighted synthesis filter and adaptive codebook contribution. The matrix $\mathbf{H}$ is defined as the lower triangular Toeplitz convolutional matrix with diagonal elements $h(0)$, $h(1),...,h(54)$, which denote the impulse response of the weighted synthesis filter.

It can be shown that the optimum codeword is the one maximizing

$$T_k = \frac{C_k}{E_k} = \frac{(\mathbf{d}^T\mathbf{c}_k)^2}{\mathbf{c}_k{}^T\mathbf{\Phi}\mathbf{c}_k}, \qquad (2)$$

where $\mathbf{d} = \mathbf{H}^T\mathbf{x}_w$ is the cross-correlation between the perceptual domain target signal and the impulse response, and $\mathbf{\Phi} = \mathbf{H}^T\mathbf{H}$ is the correlation matrix of the impulse response of perceptually weighted synthesis filter.

In order to determine the optimal algebraic codebook vector, T_k should be computed for all possible combinations of pulse positions and signs. In EVRC, this search procedure is simplified by using two schemes for searching the pulse signs and positions. One is to preset the pulse signs before searching pulse positions and the other is to use efficient pulse position search method using the preset pulse signs. Firstly, the pulse signs are preset by considering the sign of an appropriate reference signal. The reference signal, $e(i)$, is computed using an weighted sum of the residual domain target vector $x(i)$ and the backward filtered signal $d(i)$:

$$e(i) = \frac{\sqrt{\sum_{n=0}^{L-1} d^2(n)}}{\sqrt{\sum_{n=0}^{L-1} x^2(n)}} x(i) + 2d(i); \quad 0 \le i \le L-1, \qquad (3)$$

Also, the optimal pulse positions are determined using an iterative depth-first tree search strategy. In this approach, the 8 pulses are grouped into 4 pairs of pulses. The pulse positions are determined sequentially one pair at a time.

3. FAST IMPLEMENTATION ALGORITHMS IN THE FCB SEARCH

In CELP type speech coder, the heaviest computational load is generated by the FCB search routine. Thus, fast algorithms should concern the FCB search routine. In this section, we develop fast algorithms by simplify the FCB search. Another goal of the fast algorithms is that it should maintain the quality of the reconstructed speech. For these goals, two efficient schemes are suggested: (1) limiting the number of possible combination of pulse positions, and (2) using a reduced-tap FIR filter instead of the impulse response of the weighted synthesis filter.

3.1 Reduction of Pulse Position Combination

In EVRC encoder, an iterative depth-first tree search is used. In order to determine 8 non-zero pulses in the Rate 1 codebook, the search process is repeated for 4 iteration by assigning the two single-pulse tracks to corresponding pairs among 4 combinations, shown as Table 2. Thus, only 1056(=4×4×(6×11)) possible pulse position combinations are tested in the worst case out of 1936(=4×4×(11×11)) position combinations, using the magnitude information of the reference signal which is used to preset the pulse signs in Eq. (3).

By reducing the number of pulse position combination, the FCB search can be greatly simplified. In this study, we reduced 4 combinations of single-pulse track and double-pulse track or restricting the pulse positions. Table 3 shows the comparison of possible pulse position and SNRs of the original method and proposed methods. In the table, 'M 1-1'

gets rid of 25 % possible pulse positions by using 3 combinations of single-pulse tracks and double-pulse tracks out of 4 combinations. 'M 1-2' and 'M 1-3' reduce the number of the possible pulse position combination by strictly restricting the pulse position using the magnitude information of reference signal. Instead of the reference signal $e(i)$, we used the backward filtered signal $d(i)$ used in G.729 FCB search [7]. 'M 1-3' saves about 50 % of complexity of the original method. This method, however, brings a slight degradation of performance. The SNR was degraded by 0.13dB.

FCB Search Method 1	Possible Position Combination		SNR (dB)
Original	4×4×(6×11)	1056	17.268
M 1-1	3×4×(6×11)	792	17.125
M 1-2	4×2×{(5×11)+(5×5)}	640	17.130
M 1-3	4×4×(3×11)	528	17.140

Table 3: Comparison of pulse position and performance for Method 1.

3.2 Extremely Truncated FIR Synthesis Filter

The FCB search target vector, $x_w(n)$, is generalized in the perceptual domain as

$$x_w(n) = \hat{s}_w(n) - g_p \lambda(n), \quad 0 \le i \le L-1, \tag{4}$$

where $\hat{s}_w(n)$ and $\lambda(n)$ are weighted modified speech signal and the filtered adaptive codebook (ACB) excitation respectively.

The target signal for codebook search is generated by filtering the modified residual signal through the perceptual weighted IIR synthesis filter in Eq. (5).

$$H_w(z) = \frac{W(z)}{\hat{A}_q(z)} = \frac{1}{\hat{A}_q(z)}\left(\frac{\hat{A}(z/\gamma_1)}{\hat{A}(z/\gamma_2)}\right), \tag{5}$$

where $1/\hat{A}_q(z)$ is the synthesis filter based on the quantized, interpolated LP filter coefficients and $W(z)$ is the weighting filter based on the interpolated LP filter coefficients. 0.9 and 0.5, respectively, were used for γ_1 and γ_2. Note that the value of γ_1 and γ_2 determine the frequency response of the filter $W(z)$.

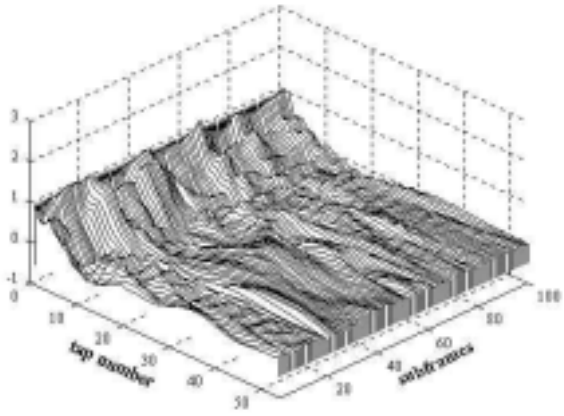
The weighted synthesized signal is reconstructed by filtering the excitation obtained from codebook through the perceptual weighted FIR synthesis filter. This FIR filter is given as the impulse response of the perceptual weighted synthesis filter and has the same number of taps as the subframe length. The best way to lessen the computational load for the codebook search is the truncation of the FIR filter taps. However, the truncation of the filter taps can cause significant distortion of the synthesized speech due to the analysis-by-synthesis process.

The weighting synthesis filter is simplified by using the perceptual weighting filter based on the quantized LP filter coefficients. This weighting filter is similar to one used in G.729A [8] and represented as

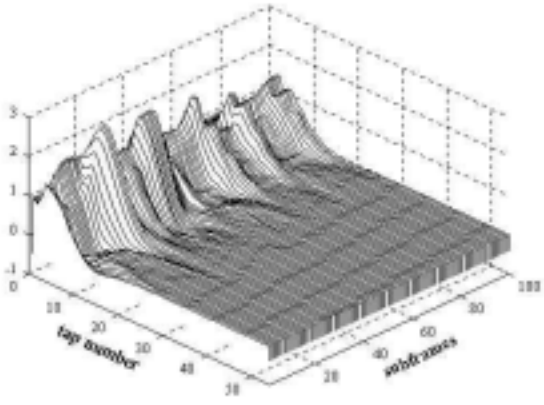
$$\hat{H}_w(z) = \frac{\hat{W}(z)}{\hat{A}_q(z)} = \frac{1}{\hat{A}_q(z)}\left(\frac{\hat{A}_q(z)}{\hat{A}_q(z/\gamma_3)}\right) = \frac{1}{\hat{A}_q(z/\gamma_3)}, \tag{6}$$

where the value of γ_3 is fixed to 0.85. This weighted synthesis filter can reduce the length of the impulse response, and, thus, can reduce the operations for computing the target signal and for updating the filter states. However, it has been reported that the simplification of the weighting filter may result in quality degradation in cases of input signals with flat response [8].

Figure 1 shows the impulse responses of two perceptual synthesis filters given in Eqs. (5) and (6). Since the impulse response shown in Figure 1 (a) has a long tail, severe distortion may result from the tap truncation. Note that the impulse response of Figure 1 (b) has a shorter tail in comparison with that of Figure 1 (a). Though the 20-tap impulse response is used in codebook search, subjective quality of reconstructed speech is not degraded. However, more tap truncation results in quality degradation in this case.



(a)



(b)

Figure 1 : Impulse response of weighted synthesis filters. (a) $H_w(z)$, (b) $\hat{H}_w(z)$.

In order to use shorter tap without any degradation of quality, the FIR synthesis filter shall be used to generate the tar-

get signal for FCB search instead of IIR synthesis filter [9]. In this case a few tap of FIR synthesis filter in both codebook search and target signal generation may minimize the quality degradation. When the number of taps is larger than zero and less than the subframe length, the shorter filter taps cause a stronger perceptual weighting [9]. When taps of impulse response is reduced, objective performance in SNR is degraded but subject quality is not degraded. The objective performance degradation of proposed methods results from strong weighting in quantization of excitation signal.

4. FAST EVRC ENCODER AND SUBJECTIVE EVALUATION

We applied the fast algorithms described in Section 3 to EVRC encoder and evaluated the subjective performance of the developed methods. Evaluation was performed with fast algorithms searching pulse positions for half of the pulse position combination ('M 1-3') was used and a 7-tap truncated FIR synthesis filter was applied to the codebook search. We implemented fast algorithms using a fixed-point Oak DSP. Table 3 compares the computational complexity of original method and the developed method. The computational saving with the fast FCB search method was significant. In comparison to the original routine, the fast FCB search method was implemented with only 24.3% of DSP cycles. In consequence, it was possible to save 4.43 DSP MIPS in total.

Encoding algorithm	Computational complexity (cycles/MIPS)	Percentage
EVRC	364,936 / 18.25	100.0 %
Fast algorithms	276,300 / 13.82	75.7 %

Table 4: Comparison of computational complexity in FCB search.

The subjective listening tests were performed via a Preference test. Four Korean sentences of two female and two male were used for the evaluation. 20 subjects were participated in the tests. Table 5 summarizes the results. Results in Table 5 confirm that the difference of subjective quality between the original EVRC and the fast EVRC was not noticeable in spite of small degradation of SNR.

Preferences	Percentages
preference A	25 %
preference B	20 %
No preference	55 %

Table 5: Subjective test results. (A: original EVRC, B: Fast EVRC)

5. CONCLUSION

In this paper, we presented a fast implementation algorithm for EVRC. In order to simplify the fixed codebook search which is one of the most computationally intensive routines in EVRC, efficient search schemes were developed. The developed method searches codebook pulses for half of the pulse position combinations of the original EVRC, and uses 7-tap truncated FIR synthesis filter for the codebook search. Using the developed methods, computational load for the fixed codebook search was reduced to 24.3 % of the original method in EVRC. Informal subjective tests confirmed that the difference of the synthesized speech quality between the original EVRC and the proposed method was not noticeable.

6. REFERENCES

[1] QUALCOMM Inc., Proposed TIA/EIA/PN-3292 Standard - Enhanced Variable Rate Codec, Speech Service Option 3 for Wideband Spread Spectrum Digital Systems, Official Ballot Version, April 19, 1996.

[2] TIA/EIA/IS-96-A Interim Standard - Speech Service Option Standard for Wideband Spread Spectrum Digital Cellular System, May 1995.

[3] QUALCOMM Inc., Proposed Standard - High Rate Speech Service Option for Wideband Spread Spectrum Communication Systems, May 1995.

[4] GSM 06.74, Digital cellular telecommunication system (Phase 2+); Adaptive Multi-Rate speech transcoding, ETSI EN 301 704, 1998

[5] W. B. Kleijn, P. Kroon, and D. Nahumi, "The RCELP Speech-Coding Algorithm," *European Transactions on Telecommunications*, Vol 5, Number 5. Sept/Oct 1994, pp. 573-582.

[6] W. B. Kleijn, *Analysis-by-Synthesis Speech Coding Based on Relaxed Waveform-Matching Constraints*, Ph. D. dissertation, Delft University of Technology, 1991.

[7] ITU-T Rec. G.729, Coding of Speech at 8 kbit/s Using Conjugate-structure Algebraic-code-excited Linear-prediction (CS-ACELP), March 1996.

[8] Redwan Salami, Claude Laflamme, Bruno Bessette, and Jean-Pierre Adoul, "ITU-T G.729 Annex A : Reduced Complexity 8 kbit/s CS-ACELP Codec for Digital Simultaneous Voice and Data," *IEEE Communication Magazine*, Sept. 1997, pp. 56-63.

[9] H. Ohmura, and K. Mano, "A Low-Complexity 4.6 kbit/s Speech Coder Based on Extremely Truncated Impulse Response," *Proc. of IEEE Workshop on Speech Coding*, 1997, pp. 63-64.