Variable-Length Coding of ACELP Gain Using Entropy-Constrained VQ

Takayoshi Oshima
Grad. Sch. of Information
Science and Technology
the University of Tokyo
Tokyo, Japan

Yutaka Kamamoto
and Takehiro Moriya
NTT Communication
Science Laboratories
Kanagawa, Japan

Nobutaka Ono
National Institute of Informatics
Tokyo, Japan

Shigeki Sagayama
Grad. Sch. of Information
Science and Technology
the University of Tokyo
Tokyo, Japan

***Abstract*—We designed variable length code for the vector quantization (VQ) gain parameter of the Algebraic Code-Excited Linear Prediction (ACELP) speech coding scheme, aiming at reduction of bit rate and distortion in the environment of IP communication. The code index is selected taking both quantization distortion and average code length into consideration. The VQ tables has been trained by the algorithm based on Entropy-Constrained VQ (ECVQ). It has been shown that this scheme can keep the quality and reduce the average bit rate by 0.2 kbit/s when applied to the state-of-the-art speech coding standard ITU-T G.718.**

I. Introduction

In recent years, the number of worldwide mobile phone connections has exceeded five billion, which indicates that many people use mobile phones as tools for communication. However, there is still a demand for higher speech quality of telephone calls.

For speech coding, Linear Prediction Coding (LPC) is generally used, by which the coefficients of LPC are transmitted efficiently. On the other hand, for the coding of residual signals of LPC, Code-Excited Linear Prediction (CELP) [1], [2] is widely used, by which residual signals are vector-quantized in the time domain. In the encoder using this method, Analysis by Synthesis (AbS) is carried out, which reconstructs speech using the code vectors of a codebook and selects the code vector that minimizes the distortion of the reconstructed speech. Though this method, which has no constraints for construction of a codebook, should reduce distortion, it requires memory capacity for a codebook and computation for a search.

Most speech coding standards, including G.718 [3] for traditional mobile communications, specify Algebraic Code-Excited Linear Prediction (ACELP) [2], [4] for the coding of LPC residual signals. This system uses an adaptive codebook, which was originally used in Self-Excited Linear Prediction (SELP) [5]. This codebook aims for a reduction of memory capacity by using residual signals in a previous frame. Besides the adaptive codebook, an algebraic codebook is used in order to modify the adaptive codebook. This codebook consists of some pulses each of which is placed approximately every 10 sample points, and is generated by a rule. Thus, the memory capacity is almost negligible.

ACELP has been highly tuned to reduce perceptual distortion and to ensure robustness against transmission channel errors. Thus, traditional ACELP schemes intentionally avoid variable length coding of parameters, which is extremely fragile due to bit errors.

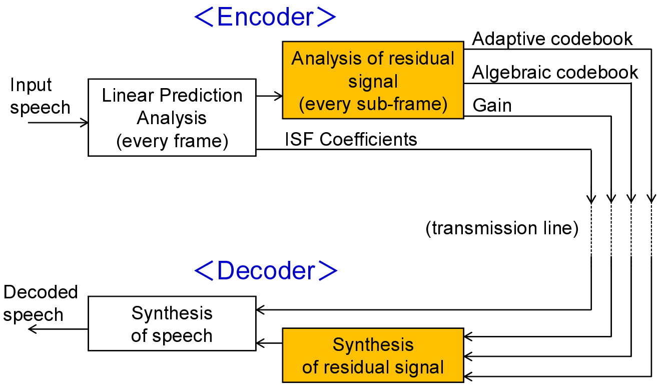


Fig. 1. Outline of the coding system.

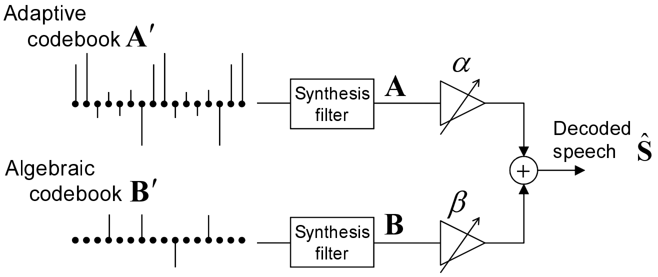

Adaptive codebook $\mathbf{A}'$
Synthesis filter
$\mathbf{A}$
α
Decoded speech $\hat{\mathbf{S}}$
Algebraic codebook $\mathbf{B}'$
Synthesis filter
$\mathbf{B}$
β

Fig. 2. Model of excitation signals.

All future telephone communications, however, (including mobile phones) will make use of IP networks, where compressed speech codes are carried in a payload. Future speech coders for IP communications will therefore not have to be concerned about bit errors in payloads. It is important to try to use any tools available to eliminate redundancy, even if they are not robust against bit errors. To optimize the gain table with both speech distortion and code length as criteria, we propose a design method that considers the relation between distortion and code length by means of an algorithm based on

978-1-4673-1157-1/12/$31.00 © 2012 IEEE

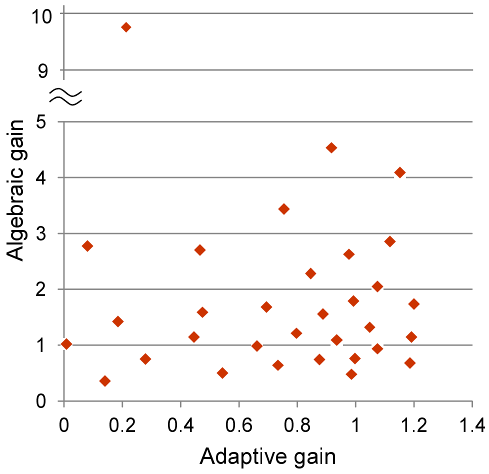


Fig. 3. Gain table in G.718.

entropy-constrained vector quantization (ECVQ) [6] instead of the commonly used VQ [7].

In section II, we explain the outline of the ACELP system used in G.718. In section III, we describe the algorithms for VQ and ECVQ. In section IV, the coefficients for transformation from distortion to code length are derived theoretically. Finally, in sections V and VI, we show how we design ECVQ tables and present an evaluation of speech quality.

II. ACELP System in G.718

In the G.718 system, there are four encoding modes for 12-kbps encoding. Two of them are the Voiced-Coding (VC) mode and the Generic-Coding (GC) mode. The VC mode encodes voiced sounds that have much stationarity or continuity, and the GC mode encodes other various voiced sounds. As shown in Fig. 1, Linear Prediction Analysis is carried out for each frame (20-ms long), and encoding modes are switched on each frame. On the other hand, predictive residual signals are encoded for each sub-frame (5-ms long) since the residual signals vary faster.

In ACELP, residual signals are expressed in the form of a linear combination of an adaptive codebook and an algebraic codebook (Fig. 2). The gain of these codebooks is calculated so that the mean-squared weighted error between a decoded speech signal and a target speech signal is minimized. The error is given by

$$\begin{aligned} d(\mathbf{S}, \hat{\mathbf{S}}) &= \|\hat{\mathbf{S}} - \mathbf{S}\|^2 \\ &= \|\alpha\mathbf{A} + \beta\mathbf{B} - \mathbf{S}\|^2 \,, \end{aligned} \tag{1}$$

where $\mathbf{S} = (s_1, s_2, \cdots, s_{64})$ is the target vector, $\hat{\mathbf{S}}$ is the reconstructed vector, $\mathbf{A}$ is the filtered adaptive code vector, $\mathbf{B}$ is the filtered algebraic code vector, and α and β are the gains of each code vector.

The adaptive and algebraic gains are jointly vector-quantized using a 5-bit codebook table (Fig. 3). In G.718, because 5-bit fixed-length coding is assumed, the gain codebook table is designed to minimize only speech distortion. However, in VoIP, taking into consideration not only distortion but also average code length is expected to be effective.

III. Vector Quantization

In this section, we first describe the Linde-Buzo-Gray (LBG) algorithm [7], which is generally used for VQ design. Then, we present ECVQ algorithm, which this paper use to design codebooks.

A. LBG algorithm

VQ is a quantization method that replaces each sample of multidimensional continuous data with one of the finite sets of code vectors.

For training a vector-quantized codebook, the LBG algorithm is used generally. This algorithm executes alternate repetitions of the k-means algorithm and division of code vectors. First, the code vector that minimizes a sum total of the distortion of all samples is found. Then this code vector is divided into two slightly separated code vectors. Subsequently, following the k-means algorithm, a code vector that minimizes distortion is assigned to each sample, then the code vectors are updated so that a sum of the distortion in each cluster is minimized, and these two processes are run iteratively. When the sum total of distortion of all samples converges, each code vector is divided again into two. After that, the k-means algorithm and division of code vectors are repeated in the same way.

B. Entropy-Constrained Vector Quantization

While the quantizer in the LBG algorithm is designed to minimize only distortion, average code length also plays an important role as mentioned above. ECVQ offers a quantizer based on the criteria of both distortion and average code length. Though the framework of ECVQ follows the k-means algorithm, ECVQ differs from LBG algorithm in some aspects.

First, the VQ table of intended size by the LBG algorithm is set as an initial table, and then iterative computation is carried out by an algorithm similar to the k-means algorithm. In the process that selects the best code vector for each sample, the distance function consists of not only distortion but also of the code length of each code vector in the case of variable coding, as folllows:

$$d^*(i) = d(i) + \lambda \cdot l(i) \,, \tag{2}$$

where i is an index of the codebook and $l(i)$ is code length of the index. Then, the code vector that makes less distortion and has shorter code length is selected.

Another difference from the LBG algorithm is that code lengths have to be updated in each iterative process. Each time the best code vector for every sample is assigned, the frequencies of code vectors are calculated. On the basis of the information about the frequencies, the code length of each code vector is calculated.

However, the LBG algorithm needs to calculate code length in each iterative process (e.g., using Huffman algorithm), so the computational complexity is high in an actual training process. Then, for simplicity, code vectors are allowed to have noninteger code lengths, and the amount of information about each code vector is used as an approximate value of code length:

$$\begin{aligned} d^*(i) &= d(i) + \lambda \cdot l(i) \\ &\simeq d(i) + \lambda\{-\log_2 p(i)\}\,. \end{aligned} \tag{3}$$

It has been shown that the system with this approximation produces a system whose performance is nearly identical to a system that includes the Huffman algorithm within the loop [6].

In the process that updates a centroid of each cluster, the vector that minimizes the total distortion in the cluster is calculated as a centroid in the same way as the LBG algorithm.

Defining the distance function in the form of the sum of distortion and code length decreases the performance function monotonically in each process. Thus, using the ECVQ algorithm, a quantizer can be designed by using both distortion and code length as criteria.

However, in this distance function two elements that have different dimensions are added, and the relation between distortion and code length is not considered. It therefore can not be necessarily said that an optimal quantizer is obtained from the view point of distortion and code length.

IV. Design of Quantizer Considering the Relation Between Distortion and Code Length

In order to optimize distortion and code length simultaneously, the relation between distortion and code length should be considered.

First, let us suppose a case of scalar quantization to simplify the problem. When the quantization level is Δ, and data are uniformly distributed, the mean square error is expressed as

$$\bar{d} = \int_{-\frac{\Delta}{2}}^{\frac{\Delta}{2}} q^2 \frac{dq}{\Delta} = \frac{1}{12}\Delta^2\,. \tag{4}$$

Therefore, when b bits are added, the quantization level is $2^{-b}\Delta$, and the mean square error decreases by a factor of 2^{-2b}. Then, the distortion for b bits, D_b, is as follows:

$$D_b = 2^{-2b} D_0\,. \tag{5}$$

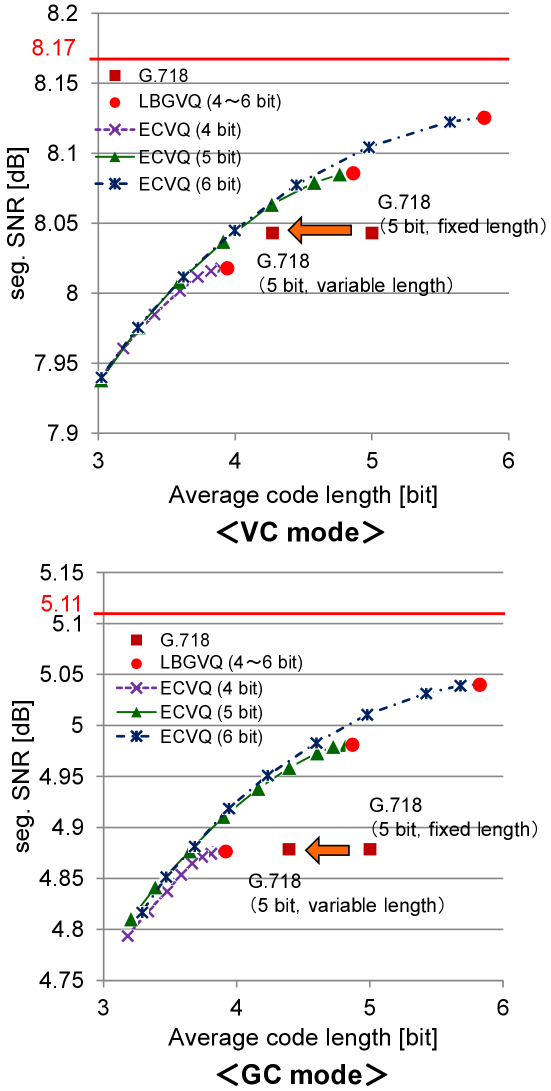
where D_0 is the distortion for 0 bit.

The residual signals, which this research actually deals with, are vectors. When there is no correlation between the dimensions of each vector, it would be appropriate to assume that b bits are distributed equally among N dimensions. (It can be found that there is not much difference even if it is assumed that b bits are given to only one dimension.) Then, the distortion for b bits is as follows:

$$D_b = 2^{-2\frac{b}{N}} D_0\,. \tag{6}$$

Then, the logarithm of ratio of the distortion is

$$\begin{aligned} \log_{10}\frac{D_b}{D_0} &= \log_{10} 2^{\frac{-2b}{N}} \\ &= -\frac{2}{N}\log_{10} 2 \times b \\ &= -\lambda b\,. \end{aligned} \tag{7}$$



※Each bold line (8.17 / 5.11) denotes seg. SNR without quantization

Fig. 4. seg. SNR and average code length using ECVQ table.

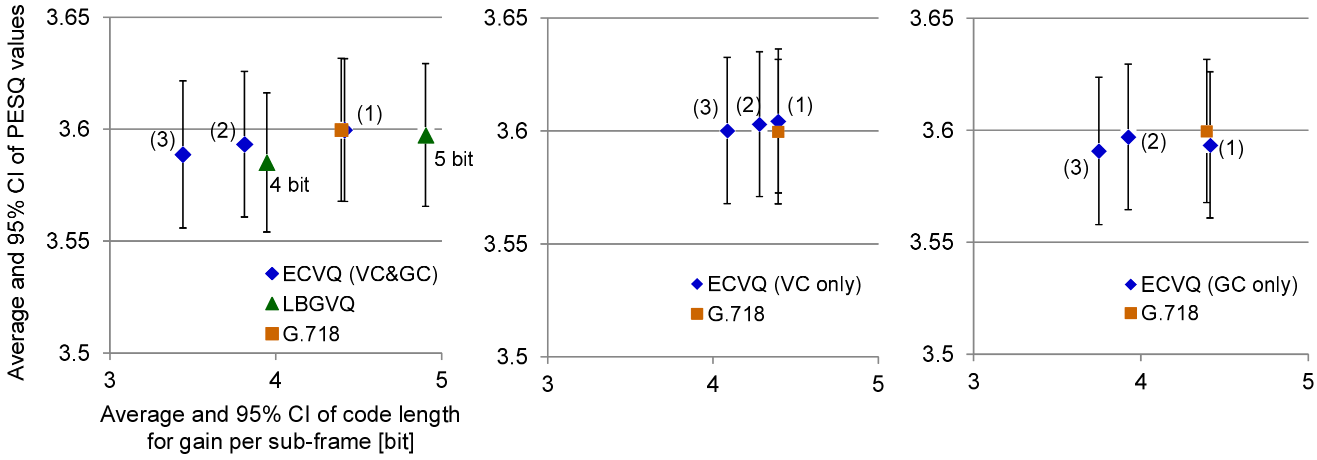
Code length can be transformed into distortion as a linear expression in this equation. It is therefore thought that optimization in terms of both distortion and code length can be achieved by adding this λb as a term of a penalty for code length in a distance function. Therefore, the distance function should be described newly as

$$\log_{10} D^* = \log_{10} D + \lambda l(i)\,, \tag{8}$$

where $l(i)$ is code length of a code vector whose index is i.

Besides, the standardized squared error between the target speech and speech reproduced by a code vector should be used as distortion in the distance function instead of a simple square distance between code vectors, because the geometric distance between code vectors does not necessarily correspond to speech quality:

$$D(\mathbf{S}, \hat{\mathbf{S}}) = \frac{\|\hat{\mathbf{S}} - \mathbf{S}\|^2}{\|\mathbf{S}\|^2} = \frac{\|\alpha\mathbf{A} + \beta\mathbf{B} - \mathbf{S}\|^{\mathbf{2}}}{\|\mathbf{S}\|^{\mathbf{2}}} \tag{9}$$



<(i) VC&GC modes>　<(ii) VC mode only>　<(iii) GC mode only>

Fig. 5. PESQ evaluation results and average code length with 95% confidence interval.

where $\mathbf{S}$ is the target speech vector, $\hat{\mathbf{S}}$ is the decoded speech vector, $\mathbf{A}$ is the decoded signal of the adaptive codebook, $\mathbf{B}$ is the decoded signal of the algebraic codebook, and α and β are the gains of the two codebooks respectively.

Accordingly, the dimension of a vector in the transformation coefficient λ should correspond to the distortion. As the length of a speech vector per sub-frame is 64, 64 is substituted for N in λ so the λ is calculated to be 0.009.

V. Training of ECVQ table

In this section, we present the experiments for training the gain parameters of ACELP in G.718 using the design method described in the previous section. The purpose is to verify that code vectors can be optimized in terms of both distortion and code length and that less distortion and shorter code length than G.718 can be achieved.

In G.718, VC and GC modes are constrained to use common code vectors in case of code errors. However, in VoIP that constraint is unnecessary, and the histograms of gain parameters in VC and GC modes are actually very different. We therefore trained code vectors for VC and GC modes individually. And for a wide range of comparison between the ECVQ code vectors and G.718 code vectors, we trained 4, 5 and 6 bit size codebooks for VC and GC modes respectively.

We trained the code vectors using 10 steps of λs for each mode, the values of which were around the theoretical value of λ:

$$\lambda_m = 2 \times 10^{-3} m \;\; (m = 0, 1, 2, \cdots, 9) \tag{10}$$

The training data were about 3 hours of clean/noisy speech data consisting of several languages and a cappella song about 3 minutes long. The test data were about 40 minutes of clean/noisy speech data.

VI. Evaluation of speech quality

A. Segmental SNR evaluation

The evaluation results are shown in Fig. 4. The horizontal axis shows Huffman code length, and the vertical axis shows the values of the Segmental Signal to Noise Ratio (seg. SNR). The seg. SNR is the average of the SNR values in all samples;

$$\begin{aligned} \text{seg. SNR} &= 10 \log_{10} \frac{\|\mathbf{S}\|^{\mathbf{2}}}{\|\hat{\mathbf{S}} - \mathbf{S}\|^2} \\ &= 10 \log_{10} \frac{\|\mathbf{S}\|^{\mathbf{2}}}{\|\alpha\mathbf{A} + \beta\mathbf{B} - \mathbf{S}\|^{\mathbf{2}}} . \end{aligned} \tag{11}$$

It is desirable that the ECVQ plots have a higher seg. SNR and shorter code length than G.718. Fig. 4 shows that, using ECVQ code vectors, higher seg. SNR values and shorter code lengths than those in G.718 can be obtained.

B. Objective evaluation

The above results show that the ECVQ algorithm achieves higher seg. SNR values of speech, Next, to evaluate the quality as speech more precisely, we conducted objective evaluation experiments of speech quality. We used the Perceptual Evaluation of Speech Quality (PESQ) [8] as a measure, which is widely used for evaluations of speech quality. The range of the PESQ value is -0.5 to 4.5. The higher the PESQ value is, the better the speech quality.

We compared the PESQ values and average code lengths in three conditions with the size of gain tables of 5 bits: (1) The transformation coefficient λ_m is set to achieve higher seg. SNR and the same average code length as G.718 (Huffman coding) (Higher seg. SNR); (2) λ_m is set to achieve smaller bit-rates and the same seg. SNR as G.718 (Same seg. SNR); (3) λ_m is set to achieve lower seg. SNR and smaller bit-rates than G.718 (Lower seg. SNR). Furthermore, these three conditions are divided into three sub-conditions respectively; (i) ECVQ is applied to both VC and GC modes; (ii) ECVQ is applied to only VC mode, and G.718 codebook is applied to GC mode; (iii) ECVQ is applied to only GC mode, and G.718 codebook is applied to VC mode. For 300 input speech files, we calculated the values of PESQ and average code lengths with 95% confidence interval through both VC and GC modes in Huffman coding for each condition. The results are shown in Fig. 5. In condition (i), the result for a 4-bit table with λ_0 in each mode is also shown.

It was found that in condition (ii) higher values of PESQ and shorter average code length than the condition in G.718 were simultaneously achieved. In condition (i), the reduction in average code length of about 1 bit per sample was achieved with much the same value of PESQ as the condition in G.718. On the other hand, for the 4-bit table with λ_0, which corresponds to the LBG algorithm, there is a larger decrease in the value of PESQ. Therefore, it is thought that the offered design algorithm based on ECVQ is effective. One sample (sub-frame) has 5-ms length. So a reduction of 1 bit per sample corresponds to information compression of 0.2 kbps. If the extra bits can be given to other parameters of the system, speech quality should be improved.

VII. CONCLUSION

We improved ITU-T G.718 assuming that variable-length coding is used in VoIP. We focused on the gain table of residual signal in ACELP for 12 kbps. To optimize the gain table in terms of both distortion of speech and code length, we introduced a design method that considers the relation between distortion and code length by means of the algorithm based on ECVQ instead of usual VQ. By the objective evaluation of PESQ, we showed that much the same speech quality as G.718 and information compression of about 0.2 kbps at most can be simultaneously achieved.

In the future, we intend to consider evaluation criteria for distortion and carry out subjective evaluation experiments.

REFERENCES

[1] M. R. Schroeder and B. S. Atal, “Code-excited linear prediction (CELP): High-quality speech at very low bit rates,” Proc. ICASSP 85, pp.937–940, 1985.

[2] Takehiro Moriya, “Speech Coding,” IEICE, 1998.

[3] ITU-T, June 2008, ITU-T Recommendation G.718 Frame error robust narrow-band and wideband embedded variable bit-rate coding of speech and audio from 8-32 kbit/s.

[4] R. Salami, C. Laflamme, and J. P. Adoul, “8 kbit/s ACELP coding of speech with 10 ms speech-frame: A candidate for CCITT standardization,” Proc. ICASSP 94, pp.II-97–II-100, 1994.

[5] W. B. Kleijn, D. J. Krasinski, and R. H. Ketchum, “Improved speech quality and efficient vector quantization in SELP,” Proc. ICASSP 88, pp.155–158, 1988.

[6] P. A. Chou, T. Lookabaugh, and R. M. Gray “Entropy-Constrained Vector Quantization,” IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 37, No.1, pp.31–42, 1989.

[7] Y. Linde, A. Buzo, and R. M. Gray, “An Algorithm for Vector Quantizer Design,” IEEE Transactions on Communications, Vol. 28, No.1, pp.84–95, 1980.

[8] ITU-T, November 2007, ITU-T Recommendation P.862.2 Wideband extension to Recommendation P.862 for the assessment of wideband telephone networks and speech codecs.