LOW COST VECTOR QUANTIZATION METHODS FOR SPECTRAL CODING IN LOW RATE SPEECH CODERS

H.R. Sadegh Mohammadi and W.H. Holmes

School of Electrical Engineering
The University of New South Wales
Sydney 2052, Australia

ABSTRACT

In low rate speech coders based on the linear prediction method, the quality of synthesized speech can be improved by enhancement of the short-term spectrum quantization stage. In this study, we propose two new efficient methods for coding the spectral parameters, namely sorted codebook vector quantization (SCVQ) and fine-coarse vector quantization (FCVQ). The principles of these methods are presented along with the methods of training and optimizing the related codebooks. The performance of the new schemes is compared experimentally with other efficient methods, such as tree-searched vector quantization (TSVQ) and multi-stage vector quantization (MSVQ). We demonstrate that the new methods offer significant cost reduction whilst achieving superior quality.

1. INTRODUCTION

Low rate speech coding area is currently dominated by the linear prediction model, e.g. code-excited linear prediction (CELP). The quality of reproduced speech in these coders depends to a large extent on the performance of the spectral coding stage. Among the various representations of the linear prediction coefficients, line spectrum frequencies (LSFs) [1] are often used for short-term spectrum quantization because of their desirable properties, which are exploited during spectral coding. Various quantization techniques have been developed for LSFs during the last decade, including many scalar quantization (SQ) and vector quantization (VQ) methods.

Although vector quantizers can achieve lower distortion than scalar quantizers at the same bit rates (or alternatively, realize the same distortion at lower bit rates), their application has been restricted by their considerable computational and storage costs. It is well known that the computational complexity and storage costs of unstructured VQ grow linearly with the dimension of target vector and the size of codebook. In selecting the best vector quantization scheme, there is a compromise between high quality and low cost, where cost is measured in terms of computational and storage requirements.

To reduce the costs of VQ, two general approaches have been used in the past. Firstly, storage can be reduced by combining several smaller codebooks. Split vector quantization and multi-stage vector quantization (MSVQ) are examples of this approach [2]. Secondly, to reduce computational complexity, various structured codebook VQ methods have been developed, such as tree-searched vector quantization (TSVQ).

Cost reduction has its price; namely an increase in quantization distortion. Fortunately this quality degradation is small, and there are now VQ methods requiring only medium amounts memory and moderate computation.

In this study, we concentrate on a tenth order LP model. Split VQ is considered as a good basis for development of a low cost method for spectrum quantization. The entire LSF vector in each frame of speech (30 ms) is divided into three sub-vectors of dimensions 3, 3 and 4. The LP parameters are extracted in a similar way to the FS1016 CELP standard [3]. This standard also is used for the simulation with quantized LSF values.

Section 2 deals with the principles of a new VQ method, called sorted codebook vector quantization (SCVQ). It also describes the training and optimization procedures for designing the related codebook. In Section 3 we propose another spectral coding method based on fine-coarse vector quantization (FCVQ) [4], and also briefly discuss the corresponding codebook training and optimization method.

Simulation experiments are presented in Section 4, where the results of quantization with different methods are evaluated in terms of the spectral distortion (SD) measure. Section 5 concludes the paper.

0-7803-2431-5/95 $4.00

2. SORTED CODEBOOK VECTOR QUANTIZATION

Sorted codebook vector quantization (SCVQ) is a new type of low cost VQ that is outlined here briefly. Given an L-dimensional target vector $\mathbf{F}_t = [f_{1t}, f_{2t}, \ldots, f_{Lt}]^T$ and a codebook C of size N, we define a sorting parameter $s_t = g(f_{1t}, f_{2t}, \ldots, f_{Lt})$, which is a scalar by definition, where $g(\cdot)$ is a suitable function, chosen in such a way that neighbouring target vectors give neighbouring values of s_t. Then the indices of the codebook are sorted in ascending order of the sorting parameter for each codevector, according to the vector $\mathbf{S} = [s_1, s_2, \ldots, s_N]^T$ with $s_1 \le s_2 \le \ldots \le s_N$.

To code the target vector, in the first step the quantity s_t is scalar quantized by $\mathbf{S}$. Suppose s_i is the result of scalar quantization, with $1 \le i \le N$. The index of s_i (i.e. i) is called the central index. In the next step, the target vector is vector quantized using an extensive local search in the neighbourhood of the central index. For example, only the codevectors with indices within the range of $i-k+1$ to $i+k$ may be searched, where k is an offset value.

To achieve better performance, always $2k$ codevectors are always searched. For example, this means that for the case of $i \le k$, the first $2k$ codewords in the codebook are considered for local search, and for $i \ge N-k$ the last $2k$ codevectors are searched. Generally, the computational complexity of this method grows linearly with k, which normally is set to be much less than N.

This SCVQ method can be applied to any proper codebook, such as an optimal unstructured codebook, without any further training process. However, the total distortion can be reduced if the following optimization algorithm is used to enhance the codebook:

Step 1. *Initialization*: Set $n = 0$, $C_n = C$, the initial codebook. Calculate the sorting parameter related to each codevector and sort the codebook in ascending order of the sorting parameter.

Step 2. *Classification*: Code the entire training database with the C_n codebook using the SCVQ method.

Step 3. *Centroid Calculation*: Compute the C_{n+1} codebook. This is done by substituting for each codeword $\mathbf{F}_m$, $1 \le m \le N$ in C_n by the centroid of the training vectors allocated to $\mathbf{F}_m$ during the training.

Step 4. *Sorting*: Recalculate the sorting parameters related to the codewords of the new codebook C_{n+1} and sort the codebook in ascending order of the sorting parameter. Also, set $n = n+1$.

Step 5. *Termination*: If the total distortion is less than a threshold value (or any other reasonable condition), then stop the algorithm; otherwise go to Step 2.

The memory storage required for SCVQ is similar to that needed for unstructured vector quantization, which is about half the memory needed for a uniform binary tree-searched vector quantization of the same bit rate. Although there are some cases in which SCVQ may be outperformed by tree-searched VQ (with similar complexity), it will be shown later that this method works very well for the split-VQ of LSFs.

3. FINE-COARSE VECTOR QUANTIZATION

Fine-coarse vector quantization (FCVQ) is a two-step process which has been used in other applications [4]. In this method, first the input vector is quantized using a fine codebook. This stage normally uses a fast quantization technique, such as scalar quantization. Then the selected codevector is vector quantized by the coarse codebook, generally using a look-up table transformation. The main advantage of this scheme over other fast methods such as tree-searched VQ is its higher speed.

In this paper, we propose the use of a FCVQ method for the quantization of LSFs. The LSF vector is divided into sub-vectors as before. For fine quantization of each sub-vector, the first element is simply scalar quantized using a non-uniform quantization table. Other elements in each sub-vector are scalar quantized by a differential scheme, also using non-uniform quantization tables. The use of a differential scalar quantization method in the first step of the process helps to cover the most heavily populated regions of the LSF space better than simple independent scalar quantization.

In the second stage of the process, the vectors of the indices of the quantized values are simply mapped to a coarse codebook by a look-up table method. The coarse codebook can be for example an optimal unstructured codebook trained by any proper method. From the storage point of view, the scalar quantization levels, the coarse codebook and the look-up table values must be kept in memory.

The performance of this FCVQ method can be improved by optimization in the training process. Firstly, in training the scalar quantization stage the quantizers can be jointly optimized. Secondly, the coarse codebook and the look-up table can also be optimized simultaneously using the technique presented by Moayeri et al. [4]. Furthermore, in opimization of the coarse codebook the codevector perturbation is also employed [5]. These optimizations enhance the performance of the FCVQ scheme.

4. SIMULATION EXPERIMENTS

Various codebooks have been trained on a database of 10,240 LSF vectors extracted from almost 5 minutes of speech signal. Two thirds of the training set was taken from the TIMIT database and the rest from other speech sample sources. The same number of sentences was taken from male and female speakers. Each speaker pronounced just one sentence.

The LSF extraction method is similar to the FS1016 standard. That is the LSFs are computed from the tenth order LP coefficients, which are extracted from non-overlapped frames of 240 samples of speech signals. The autocorrelation method is employed for linear prediction after passing the speech through the hamming window, and no pre-emphasis filter is used.

The LSF vectors are divided into three sub-vectors of lengths 3, 3 and 4, and separate codebooks have been generated for each sub-vector. We have trained several codebooks for unstructured vector quantization (UVQ), tree-searched VQ (TSVQ), sorted codebook VQ, and fine-coarse VQ. All of these codebooks use 8 bits per sub-vector for quantization.

In the scalar quantization stage of the FCVQ, the bit allocation scheme is assumed to be similar to that of the FS1016 standard. In the SCVQ method, the chosen sorting parameter s_t was simply the sum of the elements in each sub-vector, and the offset value for the final codebook search was $k = 8$.

In addition, two codebooks for multi-stage VQ of the entire LSF vector (without splitting) have been trained, i.e. three stages of 256-entry codebooks (MSVQ3-8) and four stages of 64-entry codebooks (MSVQ4-6). The multi-stage codebooks were later enhanced by a joint-optimization technique [6]. The search method used for multi-stage VQ was sequential full search.

The test database includes 25 utterances (69 seconds of speech) with the same ratios of male/female speakers and TIMIT/non-TIMIT data as the training database. Neither the speakers nor the sentences are common to the two databases. The LSFs of the test database were quantized by the various methods, and then used in a simulated speech coder which is similar to FS1016 (apart from the LSF quantization method).

Table 1 and Table 2 show the objective measures obtained with the various quantization schemes. For comparison, results of simulations with unquantized LSFs and with the FS1016 quantization method (using 34 bits/frame) are also shown. In Table 1, SD denotes the log spectral distortion between the original and quantized LPC spectra.

Quantization Method	SD [dB]	Percent. of Frame with SD>2 dB	Percent. of Frame with SD>4 dB
Unquantized	---	---	---
FS1016 Std.	1.49	10.6	0.4
UVQ	1.46	12.9	0.0
MSVQ3-8	1.40	11.1	0.1
MSVQ4-6	1.47	13.1	0.2
TSVQ	1.65	21.5	0.3
SCVQ	1.60	20.2	0.4
FCVQ	1.60	19.0	0.2

Table 1. Results of spectral distortion comparison

In Table 2, the SSD is the *synthesized spectral distortion* that measures the spectral distortion between the LPC spectra of the original signal and the one extracted from the synthesized speech [5]. The third objective measure is segmental signal to noise ratio (Seg-SNR), which evaluates the errors of the synthesized speech produced by decoder.

Quantization Method	No. of Bits per Frame	SSD [dB]	Seg-SNR [dB]
Unquantized	--	2.12	9.17
FS1016 Std.	34	2.31	8.73
UVQ	24	2.35	8.93
MSVQ3-8	24	2.30	8.49
MSVQ4-6	24	2.32	8.46
TSVQ	24	2.43	8.82
SCVQ	24	2.40	8.85
FCVQ	24	2.41	8.83

Table 2. Results of SSD and Seg-SNR comparison

Table 3 depicts the computational complexity and storage costs of different vector quantization methods. The storage costs are calculated assuming double integer precision for the VQ elements and floating point for the scalar quantization elements. One byte is considered for each look-up table cell in the fine-coarse vector quantization.

In the computational cost estimation, one instruction represents multiply-add, comparison or data format conversion. It is seen that sorted codebook vector quantization results in the best combination of complexity and storage costs, while fine-coarse vector quantization presents the minimum computational cost. Both quantization methods are superior to binary uniform TSVQ.

Quantization Method	Storage Cost [kbyte]	Computation Cost [No. of Instr.]
Unstructured VQ	5	7,680
MSVQ3-8	15	23,040
MSVQ4-6	5	7,680
TSVQ	9.96	480
SCVQ	5	562
FCVQ	13.44	48

Table 3. Cost comparison

5. CONCLUSION

Two new vector quantization methods are proposed for spectral coding in low rate speech coders. These methods have low costs (even less than tree-searched VQ). The concepts of the new methods are explained; moreover, the training and optimization of the associated codebooks are described. Simulation experiments show that new vector quantization methods achieve a superior performance to binary uniform tree-searched VQ and multi-stage VQ.

REFERENCES

[1] F. Itakura, "Line spectrum representation of linear predictive coefficients of speech signals", J. Acoust. Soc. Am., vol. 57, p. S35(A), 1975.

[2] A. Gersho and R.M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Pub., Boston, 1991.

[3] R. Finiceh, *Federal Standard 1016, Telecommunications: Analog to Digital Conversion of Radio Voice by 4,800 bit/second Code Excited Linear Prediction (CELP).* National Communications Systems, Office of Technology and Standards, Washington, DC20305-2010, 14 Feb. 1991.

[4] N. Moayeri, D.L. Neuhoff and W.E. Stark, "Fine-coarse vector quantization", IEEE Trans. on Signal Proc., vol. 39, no. 7, pp. 1503-1515, July 1991.

[5] H.R. Sadegh Mohammadi and W.H. Holmes, "Fine-coarse split vector quantization: an efficient method for spectral coding", Proc. of Fifth Australian Intern. Conf. on Speech Sci. and Tech., pp. 118-123, Dec. 1994.

[6] W.Y. Chan, S. Gupta and A. Gersho, "Enhanced multi-stage vector quantization by joint codebook design", IEEE Trans. on Commun., vol. 40, pp. 1693-1697, Nov. 1992.