LOW BIT-RATE SPEECH CODING USING QUANTIZATION OF VARIABLE LENGTH SEGMENTS

R. Mayrench and D. Malah
Department of Electrical Engineering
Technion, Haifa 32000, Israel
ronenm@siglab.technion.ac.il
malah@ee.technion.ac.il

ABSTRACT

This paper describes a new segmentation and quantization technique for low bit-rate speech coders. Bit-rate reduction is achieved by combining segmentation and quantization, where a segment consists of one or more adjacent frames. The algorithm for selecting and quantizing segments from a pre-determined number of frames extends the frame-based trellis techniques. It models the input speech as a sequence of variable length segments with the option to interpolate frames in skipped segments. The new algorithm, denoted as Trellis Segmentation-Quantization (TSQ), reduces the bit-rate needed for spectral envelope representation, with only a small degradation in log-spectral distance values. Experimental results show that TSQ achieves a lower spectral distance than alternate frame transmission, matrix quantization and trellis based frame selection and interpolation.

1. INTRODUCTION

Linear Predictive Coding (LPC) is a well known technique for spectral envelope representation in low and medium bit rate speech coders. Usually, LPC parameters or related Line Spectral Frequencies (LSF) are extracted from the speech signal each frame and quantized. Early speech coders, such as LPC10e, used scalar quantization for quantizing the spectrum envelope. Modern coders reduce the bit rate by using vector quantization techniques [1,2].

Atal *et al.* [1] achieved perceptually transparent quality with split vector quantization using a total of 24 bits per each vector of 10 LSF's. Leblanc *et al.* [2] suggest a multi-stage vector quantizer which achieved good quality using 22 bits and has low complexity. Recently, a coder with improved analysis and quantization was introduced in [3]. This coder uses switched predictive quantization with multi-stage VQ that lowers the number of bits for LSF quantization to 21 bits per vector. In all these coders the main bit budget is still invested in the representation of the spectrum envelope. To further reduce the bit-rate of such coders, different schemes are needed. Such a scheme is provided by representing several LSF vectors, from adjacent frames, by a single codeword, i.e., Matrix quantization (MQ) [4]. However, this method increases the delay and requires very large codebooks to obtain good performance. Usually such voice coders are referred to as fixed length segment vocoders.

Another technique [5,6] is also based on a fixed rate frame analysis but transmits information only for some of the frames in a block of frames. The decoder completes the missing information by interpolation. A simple implementation of such a technique is alternate frame (AF) transmission, where the coder transmits only odd or even frame parameters. Kemp *et al.* [5] suggest an LPC vocoder for 600-1200 bps in which the spectral information of four out of eight frames is transmitted. The last frame of the block is always transmitted to reduce the delay and there are 32 different frame selection patterns. A similar approach, without the boundary frame constrains (hence with a longer delay) is to select the frames to be quantized using a trellis [6]. This is done by organizing N frames into a block and choosing the best M ($M<N$) frames in the block such that the total error is minimized. The selection is done using Dynamic Programming (DP).

A different approach, presented in [7], considers segmenting the input stream of frames into segments of variable length (without frame skipping). The LSF's of each segment are represented by a single codeword obtained by time warping a fixed length codebook to the segment size. The search for the best segmentation-quantization is again done using dynamic programming. This approach results in a variable bit rate. A Similar variable segmentation technique is suggested by Prandoni *et al.* [8]. They used variable length segments with different analysis window lengths to obtain a better estimate of the autocorrelation function used for LPC analysis. The search for the best segmentation is done using dynamic programming as in [7]. In this technique, the segmentation space includes all the possible partitions of the block of frames, but still does not allow skipping of any frame.

In this paper we present an algorithm which adapts to the speech properties using segmentation as in [7], but has a richer partition set since it allows frame skipping. Basically, for a given fixed bit-rate and an allowed delay, the proposed algorithm optimally selects, under the Log Spectral Distance (LSD) measure, the segments to be transmitted. The quantization of a segment is done using a single codebook with a fixed

length and converting it to the segment length. We propose a modified Generalized Lloyd Algorithm (GLA) for designing the needed codebook. It will be shown that MQ and trellis quantization (TQ) are specific cases of such a technique. The algorithm can be implemented in speech coders which use an LPC-based representation of speech frames, such as LSF's.

The paper is organized as follows. In section 2 we introduce the TSQ algorithm. In section 3 the codebook design method will be presented. Results obtained in simulations using the proposed algorithm will be presented in section 4. Finally, we conclude in section 5.

2. TRELLIS SEGMENTAION-QUANTIZATION (TSQ) ALGORITHM

In this section we present the proposed TSQ algorithm for segmentation and quantization of LSF parameters. It extends the TQ approach by modeling the input speech as a sequence of variable length segments (where a segment consists of one or more adjacent frames) with the option to interpolate frames in skipped segments. The purpose of the segmentation-quantization process is to choose a fixed number (M) of *segments* from a block of N ($N>M$) LSF vectors with minimum quantization error. The missing frames are linearly interpolated using the selected segments. Fig. 1 illustrate results of the above process for the case of N=6 and M=2. The algorithm selects a segment containing the first 3 LSF vectors, skips over the 4th frame and chooses as a second segment the last 2 LSF vectors. The skipped frame will be interpolated using the last LSF vector from the previous segment and the first LSF vector from the next segment.

In the following sub-sections a method for variable length segment quantization is presented.

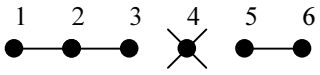

Figure 1. Segmentation results for the case of choosing M=2 segments from N=6 frames.

2.1 Variable Length Segment Quantization

The quantization process represents variable length segments with a fixed length codebook. Let X_{ij} denote a stacked row vector of original LSF vectors from the *i'th* frame until the *j'th* frame, and let $\{Y_L\}_{k=1}^{K}$ denote the codebook vectors, where L is the codeword length (i.e., number of vectors per codeword) and K is the codebook length. The sequence of L LSF vectors in a codeword represents a piece-wise linear trajectory in the LSF's space. The code vectors are linearly interpolated and resampled at $p=j-i+1$ equispaced points, to receive a stacked row LSF code vector with the same length as X_{ij}. The interpolation process can be performed by matrix multiplication:

$$\hat{Y} = Y_L H_p \tag{1}$$

Where, $\hat{Y}$ is the resampled codebook vector, Y_L is a stacked row codebook vector and H_p is a warping matrix:

$$H_p = \begin{bmatrix} h_{11}[I] & h_{12}[I] & \cdots & h_{1p}[I] \\ h_{21}[I] & h_{22}[I] & \cdots & h_{2p}[I] \\ \vdots & \vdots & \ddots & \vdots \\ h_{L1}[I] & h_{L2}[I] & \cdots & h_{Lp}[I] \end{bmatrix} \tag{2}$$

$[I]$ is a $v\times v$ identity matrix and v is the LSF vector dimension. Y_L is a vector with vL elements, H_p has vL rows and vp coulmns, and the resampled codebook has vp elements i.e., p LSF vectors as the original segment. The warping coefficients, h_{ij}, can be calculated as in [7]. Each column has only two nonzero coefficients determined by the original segment length.

The codebook is designed to minimize the distortion between the original segment and the quantized one. The quantized segment distortion- $d(X_{ij}, \hat{Y})$ is determined by the accumulated distortion between original and quantized LSF vectors, as defined in (3).

$$d(X_{ij}, \hat{Y}) = \sum_{k=1}^{p} d_1(X_k, \hat{Y}_k) \tag{3}$$

X_k and $\hat{Y}_k$ are the *k'th* LSF vectors in X_{ij} and $\hat{Y}$ segments, respectively, and $d_1(\cdot)$ is a distortion measure between the two vectors. LSD is often cited as being highly correlated with perceptual performance and it has been shown [9] that a weighted MSE (WMSE) measure can approximate it:

$$d_1(X_k, \hat{Y}_k) = (X_k - \hat{Y}_k) W_{X_k} (X_k - \hat{Y}_k)^T \tag{4}$$

Where, W_{X_k} is a diagonal weighting matrix (properly chosen to approximate the LSD measure [9]). Substituting (4) into (3) results in the matrix representation of (5).

$$d_1(X_{ij}, \hat{Y}) = (X_{ij} - \hat{Y}) W_X (X_{ij} - \hat{Y})^T \tag{5}$$

Where, W_X contains LSF weight vectors (W_{X_k}) on the main diagonal.

2.2 Trellis Segmentation

The process of choosing M segments from a block of N frames is done using a trellis diagram. As demonstrated in Fig. 2, the trellis diagram provides a graphical presentation of candidate segments and transitions between them. The number of stages is equal to M and there are $0.5 \cdot S(S+1)$ nodes per stage, where $S=N-M+1$ is

the maximum segment length. The first and the last nodes in a stage are X_{bb} and X_{ee}, respectively.

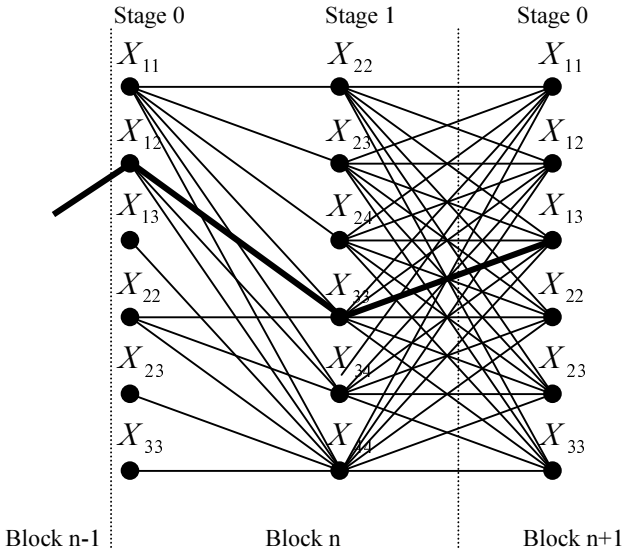

Figure 2. Trellis diagram for choosing 2 segments out of 4 frames

Where, *b*=*stage_number*+1 and *e*=*stage*_number+*S*. The other nodes in the *m'th* stage are X_{ij} ($j \geq i$), where *m*=0,…,*M*-1 and *i,j* takes values in the range [*stage_number*+1,…,*stage_number*+*S*]. The transmitted segment should be selected to minimize the total path cost but under two constraints [6]: Segments should be in ascending order, e.g., the first frame of a segment must have an index bigger then the last frame of the previous segment. The second constraint is the maximum segment length (*S*). The trellis diagram shown in Fig. 2 is for the case of choosing *M*=2 segments out of *N*=4 frames and a maximum segment length of *S*=3 frames. Each node represents a segment (X_{ij}) and the objective is to choose one node for each stage resulting in total path with minimum quantization error. The lines represent all allowed segment selections and the best trajectory is plotted in bold. In this example, for the *n'th* block the algorithm chooses segments X_{12}, X_{33} and interpolates segment X_{44}. Fig. 3 illustrates the segmentation results. The search for the best segmentation-interpolation path is done efficiently with DP using the Viterbi algorithm, similar to [6].

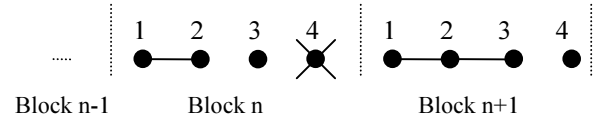

Figure 3. Segmentation Results for the example in Fig. 2

MQ and AF methods are specific cases of the TSQ algorithm and they can be implemented as a fixed path in the above scheme. For example, choosing X_{12} and X_{34} (for each block in Fig. 2) will result in a MQ scheme. A similar statement holds for TQ because it is a simplified version of the TSQ scheme, i.e., trajectories for TQ are only part of the trajectories in the TSQ scheme (TQ include only X_{ii} nodes).

3. CODEBOOK DESIGN

The codebook design algorithm combines segmentation and quantization, iteratively. Given a codebook, the segmentation is done using the trellis scheme, and given a segmentation the codebook can be optimized. In the segmentation process the algorithm defines optimal segments X_{ij}, as discussed above, resulting in a new training set for the codebook design. The vectors in the codebook can be found using the GLA. The centroid, *Y(k)*, of variable-length segments in the *k'th* cluster, S_k, is obtained by minimizing the accumulated distortion D_k :

$$D_k = \sum_{X_{ij} \in S_k} d\left(X_{ij}, Y(k)\right) \tag{6}$$

By differentiating D_k (using (1) and (5)) with respect to *Y(k)* and setting the result equal to zero the centroid is found to be:

$$Y^T(k) = \left[\sum_{X_{ij} \in S_k} H_p W_{X_{ij}} H_p^T \right]^{-1} \cdot \sum_{X_{ij} \in S_k} H_p W_{X_{ij}} X_{ij}^T \tag{7}$$

Where *p* is the length of segment X_{ij}.
It has been found experimentally that the above algorithm convergence.

4. SIMULATIONS

Several experiments were conducted to examine the performance of the proposed TSQ algorithm. LSF vectors of dimension *v*=10 were calculated using the split Levinson algorithm [10]. The speech was taken from the TIMIT database, down-sampled to 8KHz, and analyzed using a 22.5ms Hamming window. The training set included 300,000 vectors of different male and female speakers. We used a 22 bit split-VQ codebook, split into two vectors, with the first four elements in the first vector and last six elements in the second vector (similar to [1]). The TSQ was tested for the case of choosing *M*=3 segments out of *N*=6 frames and a maximum segment length of *S*=4 frames, using codebook segment length of *L*=2. The testing material was outside the training set and included 3000 vectors. The quantization error was calculated using the LSD between the original spectrum and the reconstructed spectrum represented by the original and quantized LSF vectors, respectively. The codebook design was based on minimizing the WMSE error with a weighting matrix which optimized the LSD function (in the $[-\pi,\pi]$ range), according to [9].

We observed that the codebook design algorithm converged after 3-4 iterations. The performance of TSQ, TQ, MQ, and AF, obtained in our simulations, are presented in Table 1. The first row (Split-VQ) is the LSD value obtained for coding all LSF vectors with a 22 bit per frame codebook. All other results are for half-rate coding with a 22 bit Split-VQ codebook, i.e., effectively 11 bit per frame. The TSQ algorithm has been implemented as part of a LPC-based coder operating at 1200bps, and in informal listening tests we found that it sounds better than any of the other 3 algorithms, at the same rate.

Scheme	**LSD [dB]**
Split-VQ (22 bit)	1.92
AF (11 bit)	2.38
MQ (11 bit)	2.58
TQ (11 bit)	2.21
TSQ (11 bit)	2.11

Table 1. LSD values obtained in simulation for half-rate schemes.

5. CONCLUSION

We have presented a new algorithm for low bit-rate speech codings (TSQ) based on variable length segment quantization. The coding problem can be solved using a trellis diagram and efficient search via dynamic programming. The proposed TSQ algorithm can be viewed as a generalization of the AF, MQ and TQ algorithms and has a better performance than these algorithms, in the LSD sense. The proposed algorithm achieves reduction in the bit rate needed for spectral representation (via LSF's) by a factor of 2 with only a small degradation (about 0.2 dB) in the LSD measure. We have proposed an iterative algorithm for codebook design, and it has been found, experimentally, that it converges. The TSQ algorithm has been implemented in a speech coder and achieved better results in informal listening tests than the other examined half-rate algorithms.

6. REFERENCES

[1] K. K. Paliwal, B. S. Atal, "Efficient Vector Quantization of LPC Parameters at 24 Bits/Frame", *IEEE Trans. On Speech and Audio Processing*, Vol. 1, No. 1, Jan. 1993, pp. 3-14.

[2] W. P. LeBlanc, B. Bhattacharya, S. A. Mahmoud, V. Cuperman, "Efficient Search and Design Procedure for Robust Multi-Stage VQ of LPC Parameters for 4 kb/s Speech Coding", *IEEE Trans. On Speech and Audio Processing*, Vol. 1, No. 4, Oct. 1993, pp. 373-385.

[3] A. McCree, J. C. De Martin, "A 1.7KB/S Melp Coder With Improved Analysis and Quantization", *ICASSP 1998*, pp. 593-596.

[4] C. Tsao, R. M. Gray, "Matrix Quantizer Design for LPC Speech using the Generalized Lloyd algorithm", *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSP-33, pp. 537-545, 1985.

[5] D. P. Kemp, J. S. Collura, T. E. Tremain, "LPC Parameter Quantization at 600, 800 and 1200 Bits Per Second", *Proceedings of the Tactical Communications Conference*, 1992, pp. 71-75.

[6] E. Bryan George, A. V. McCree, V. R. Viswanathan, "Variable Frame Rate Parameter Encoding Via Adaptive Frame Selection Using Dynamic Programming", *ICASSP 1996*, pp. 271-274.

[7] Y. Shiraki, M. Honda, "LPC Speech Coding Based on Variable-Length Segment Quantization", *IEEE Trans. On Acoustic Speech and Signal Processing*, Vol. 36, 1988, pp. 1437-1444.

[8] P. Prandoni, M. Goodwin, M. Vetterli, "Optimal Time Segmentation For Signal Modelling and Compression", *ICASSP 1997*, pp. 2029-2032.

[9] W. Gardner, B. Rao, "Theoretical Analysis of The High-Rate Vector Quantization of LPC Parameters", *IEEE Trans. On Speech and Audio Processing*, VOL. 3, pp. 367-381. Sept. 1995.

[10] S. Saudi, J.M. Boucher, A.Le Guyader, "A New Efficient Algorithm to Compute LSP Parameters for Speech Coding", *Signal Processing (Elsevier)*, pp. 201-212, 1992.