ISCA Archive
http://www.isca-speech.org/archive

4th European Conference on
Speech Communication and Technology
EUROSPEECH '95
Madrid, Spain, September 18-21, 1995

PREDICTIVE DELTA ADAPTIVE SCALAR QUANTIZATION: AN EFFICIENT METHOD FOR CODING THE SHORT-TERM SPEECH SPECTRUM

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

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

10.21437/Eurospeech.1995-270

ABSTRACT

Line Spectral Frequencies (LSFs) are often used as parameters to represent the vocal tract filter in speech coders using linear prediction. We propose a new method for the quantization of the LSFs, namely Predictive Delta Adaptive Scalar Quantization (PDASQ). This method is a scalar quantization scheme based on a non-linear two-dimensional prediction in the index domain. It is shown that it can be implemented efficiently with negligible computational overhead and memory requirements compared to the simple scalar quantization method. Despite its lower bit rates the quantization distortion resulting from PDASQ is of the same order as with conventional scalar quantization. Satisfactory performance of the new method is verified through experimental tests using computer simulation.

1. INTRODUCTION

Different schemes have been applied by researchers to represent the short-term spectrum in low bit rate speech coders. The importance of this issue derives from the close relationship between the shape of the spectral envelope and the formants of speech signals, which are very significant for speech perception. Traditionally, the short-term spectrum is modelled by an all-pole filter (the "vocal tract filter").

Among different parameters which could be used to represent the short-term spectrum, Line Spectral Frequencies (LSFs), ordered pairs of which are also called line spectral pairs (LSPs), are often used to parameterize the vocal tract filter [1]. The reason for this is their desirable properties for filter characterization and quantization [2], including their monotonic ordering property and strong intra-frame and inter-frame correlation between the LSFs. The LSFs are the arguments of the complex roots of two polynomials, i.e. $P(z)$ (symmetric) and $Q(z)$ (anti-symmetric), which are derived from a decomposition of the polynomial $A(z)$, the inverse of the all-pole filter which models the vocal tract.

A tenth order LP filter is used in the FS1016 Code-Excited Linear Predictive coding (CELP) standard [3], and ten LSFs must therefore be quantized for each frame (30 ms) of the speech signal. This CELP standard uses a fixed scalar quantization scheme with 34 bits per frame, which are allocated to the ten different LSFs according to the pattern (3, 4, 4, 4, 4, 3, 3, 3, 3, 3). Each LSF is independently scalar quantized using different sets of non-uniformly distributed quantization levels.

To take advantage of intra-frame and inter-frame correlations between the LSFs and to reduce the number of bits required, several more efficient quantization methods have been proposed in the past, including various scalar quantization (SQ), vector quantization (VQ) and combined SQ-VQ methods, as well as other techniques such as transform coding. Although VQ methods give better quality (or need fewer bits) than simple SQ, they involve considerably increased computational complexity and memory requirements. Even with very fast VQ methods, such as those introduced in [4], the storage cost of VQ is at least several times of that of SQ methods.

In the past several SQ methods have been reported that take advantage of both inter-frame and intra-frame correlations of the LSFs [5], [6]. Two-dimensional linear prediction of LSFs with fixed prediction coefficients was reported by Kuo et al. [7]. These methods usually suffer from the channel error propagation problem.

In this paper we introduce a new efficient method for scalar quantization of the LSFs with fewer bits. This quantization method, which exploits both the inter-frame and intra-frame correlations of the LSFs, has two-steps. The first step is an ordinary scalar quantization, and the second step is a non-linear mapping in the index domain.

The structure of this paper is as follows. Section 2 describes the principles of the new SQ method. It also describes the training and optimization procedures for designing the quantization tables. Several enhancements of the basic method are introduced in Section 3, including a technique for training a table required for the new SQ 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.

2. PREDICTIVE DELTA ADAPTIVE SCALAR QUANTIZATION

The proposed method is called *Predictive Delta Adaptive Scalar Quantization (PDASQ)*. While PDASQ can be used for quantizing any sets of parameters or time-series with intra-parameter and intra-parameter correlations, it is described here in the context of LSF quantization. The first step is to scalar quantize the LSFs (e.g. using the same quantizer as the FS1016 standard). Then the LSFs are divided into two groups, say an *odd group* that will be called *leaders* and an *even group* that will be called *followers*. The indices of the leader LSFs are sent to the decoder without any changes or extra processing, but the indices of the follower LSFs are subjected to further processing.

The basic idea in the PDASQ method is to predict the follower LSFs using the information in the leaders. This prediction is performed in an indirect way; that is, the indices of the follower LSFs are predicted, not the LSF parameters themselves. Then the differences between the true indices of the scalar quantized follower LSFs and their predicted values are encoded and transmitted to the decoder.

For the sake of simplicity, the following steps of PDASQ will be explained through an example. The quantization process is explained for the first LSF pair and it will be generalized to the others later. Assume that the encoder is processing the n-th frame and in this frame the quantization index $I_{1,n}$ of the leader (first LSF) is sent to the decoder. In the second step, the difference $d_{1,n}$ between the index of the quantized leader in the current and the previous frame is calculated. (Remember that both the encoder and decoder can perform this computation since they both have access to this data.) Thus we have

$$I_{1,n} = \mathrm{In}[\mathrm{Q_s}(f_{1,n})] \tag{1}$$

$$d_{1,n} = I_{1,n} - I_{1,n-1} = \mathrm{In}[\mathrm{Q_s}(f_{1,n})] - \mathrm{In}[\mathrm{Q_s}(f_{1,n-1})] \ , \tag{2}$$

where $\mathrm{Q_s}(.)$ represents the scalar quantization of a parameter and $\mathrm{In}[.]$ denotes the index associated with the selected quantization level.

Associated with the PDASQ quantizer there is a table, which is known as the *shift table*. The number of the columns in this table is equal to the dimension of the input vector (e.g. 10 for tenth order linear prediction). If the maximum number of quantization levels for the LSFs is denoted by M (e.g. $M = 16$ for FS1016), the shift table will have rows that are labelled from $-M+1$ to $M-1$. This range covers the whole possible variation range of $d_{p,n}, 1 \le p \le P$, where P is the dimension of the input vector (the LSF vector). Hence, for the quantization table associated with the FS1016 standard, the shift table has a size of 31×10. The preparation of the shift table will be discussed later.

In the next step, the value of $d_{1,n}$ is used to select a row of the table. The column selection depends on which follower parameter is to be quantized. Hence, for this example the second column will be chosen because LSF2 is to be processed for quantization. At the intersection of the selected row and column of the table is a cell that contains a variable, called a *shift variable*. For the current example this is

$$sv_{2,n} = \mathrm{T}(d_{1,n}, 2) \ , \tag{3}$$

where the $sv_{2,n}$ is the shift variable and $\mathrm{T}(d_{1,n},2)$ indicates a transformation that chooses the required element of the look-up table, which is in row $d_{1,n}$ and column 2.

In the third quantization step the value of this variable is added to the quantization index of the LSF2 in the previous frame and the result is called the *offset*. This quantity is available at both the encoder and decoder. To achieve better performance, this offset is constrained to lie between two values $B_{2,\min}$ and $B_{2,\max}$ which depend on the numbers of bits allocated to the quantized LSF parameter in the scalar and PDASQ quantization processes. The selection of these limits will be explained later. Therefore, the bounded offset value is defined by

$$o_{2,n} \triangleq \max\left\{\min\left\{\mathrm{In}[\mathrm{Q}(f_{2,n-1})] + sv_{2,n}, B_{2,\max}\right\}, B_{2,\min}\right\} \ , \tag{4}$$

where $o_{2,n}$ is the offset value corresponding to the second LSF in the n-th frame. The function $\mathrm{Q}(.)$ denotes the whole PDASQ quantization process, and should not be confused with the function $\mathrm{Q_s}(.)$, which represents just the scalar quantization stage. However, in the current example $\mathrm{Q}(.)$ and $\mathrm{Q_s}(.)$ are identical for the odd LSF parameters (leaders), since for these LSFs only scalar quantization is applied, but they are different for the even LSFs. Nevertheless, in the final stage, as will be explained later, $\mathrm{Q}(.)$ uses a subset of the same scalar quantization table, since fewer bits will be used for the quantizing the LSFs with the PDASQ technique than with the independent scalar quantization method. The quantity $\mathrm{In}[.]$ refers to the index of the quantized level in the second row (for LSF2) of the entire scalar quantization table.

In the final quantization step, the follower parameter, the difference between the $\mathrm{In}[\mathrm{Q_s}(f_{2,n})]$ and the offset value, is calculated. Since the number of bits allocated by PDASQ for the quantization of LSF2 is lower than for the scalar quantization step, sometimes it is not possible to cover the resulting difference with the number of bits in hand. In this situation, the difference is brought lower/higher to match the maximum/minimum addressable index. This approximation is represented by the $\mathrm{A}\{.\}$ process; therefore, the index of the final quantized LSF2 is calculated as

$$I_{2,n} = \mathrm{In}[\mathrm{Q}(f_{2,n})] = \mathrm{A}\{\mathrm{In}[\mathrm{Q_s}(f_{2,n})] - o_{2,n}\} \ . \tag{5}$$

The entire procedure is performed for all leader and follower parameters (odd and even LSFs) in the current frame, and the algorithm will be repeated every frame.

The limits are calculated as follows. Suppose that the m-th LSF is a follower and that it is quantized with $b_{m,s}$ bits in the scalar quantization stage and with $b_{m,p}$ bits in the PDASQ stage. Then the limits are calculated by

$$B_{m,\max} = 2^{(b_{m,p}-1)} \tag{6}$$

$$B_{m,\min} = 2^{b_{m,s}} - 2^{(b_{m,p}-1)} \tag{7}$$

The index of the final quantized LSF2 is always in the range $(o_{2,n} - 2^{(b_{2,p}-1)} + 1, o_{2,n} + 2^{(b_{2,p}-1)})$.

An implementation that has been tested is to retain the same bit allocation as FS1016 for the leader LSFs, and to have one bit less than the FS1016 allocation for the follower LSFs. This saves one bit for each pair, or 5 bits overall, which is a reduction of approximately 15% compared to the 34 bits in the FS1016 standard. This implementation of PDASQ will be used in this study (after several enhancements that will be discussed shortly).

The proposed method can also be used in other ways. For example, one way is to allocate more quantization levels to each follower LSF in the scalar quantization step (i.e. one bit more) and then to reduce the number of possible selections as before, ending up with the same number of bits as in the FS1016 standard, but with less distortion.

3. ENHANCEMENTS AND TRAINING

To improve the performance of PDASQ for LSF quantization, several enhancements have been incorporated into the basic algorithm. Although the basic PDASQ method outlined above works very well in principle, it suffers from one major problem in the real world. Channel errors will affect the received bit stream to the decoder and if the data are not protected with the proper forward error correction codes, this will result in misinterpretation of the real data.

This problem is not very important by itself with simple SQ methods, because the well-known ordering property of the LSF parameters can be invoked to force the filter to be stable. Hence, for simple SQ methods channel errors just change the spectrum of the decoded filter to some extent, with relatively little effect on the quality of the synthesized speech signal (at least for channels with low error probabilities).

By contrast, the effects of channel errors with PDASQ (and other differential or predictive inter-frame based quantizers) are greatly magnified because of error propagation. Fortunately, with PDASQ error propagation can be avoided by changing the quantization procedure every other frame. Thus, in one frame PDASQ will send the scalar quantized indices of the odd LSFs directly and perform the differential procedure for the even LSFs, whereas in the next frame it will reverse the process. That is, the leaders and followers are swapped in alternate frames.

This scheme will limit the error propagation problem to just one frame. The only difficulty with this enhancement is the possible need to send an extra bit to the decoder in each frame to identify the group of LSFs (odds or evens) which uses the differential mechanism. Fortunately, this can be done in some low rate speech coding methods, such as FS1016, without spending any extra bits, since the synchronization scheme of this standard changes the state of the *sync* bit every other frame - hence the same bit can be used as a signal by the decoder to detect the frame classification state for the PDASQ method.

Another improvement has been added to the generic PDASQ algorithm. Although for the middle LSFs the delta adaptation could be done using either the higher or lower LSF parameter, the selection of a fixed direction for each group of LSFs (e.g. forward adaptation for the even group and backward for odd one) is not the best choice. A superior selection method, based on both intra-frame and inter-frame correlations between the LSF frequencies, has been adopted for the PDASQ algorithm, as follows.

A statistical correlation computation has been done over 3000 frames of speech signals. According to the results of this test, it has been concluded that the optimum adaptation direction choice is as shown in Table 1. In this table the forward/backward adaptation means that for the current LSF, the leader parameter is the previous/next LSF.

As has been explained earlier, the shift table plays a key role in the PDASQ method, so that its design and training is quite an important issue. A speech file with 800 frames has been used as the training database. A similar procedure to that in the FS1016 standard has been used to calculate the LSF parameters. Then the following method has been implemented for the training of the quantization levels for each follower LSF parameter.

LSF Index	1	2	3	4	5	6	7	8	9	10
Adaptation Direction	B	B	B	F	B	F	F	F	B	F

Table 1: Adaptation direction for different LSF parameters (B = Backward, F = Forward)

First, the leader parameter is selected individually for each follower parameter according to Table 1. In the second step, both leaders and followers are quantized using the FS1016 quantization table. Then for each possible shift value (difference between the indices of the quantized leader LSF parameter in two successive frames, which is in the ranges (-15,15) for LSF2-5 and (-7,7) for LSF1 and LSF6-10, respectively), possible ranges of offset are examined. The best offset values for each shift quantity are recorded.

In the final step, the best offset values were selected manually from the candidates resulting from these possible ranges. Also, some extrapolations were performed to select the shift values associated with larger offset values that did not occur during the training.

A default set of quantized LSFs is defined so that the very first frame can be encoded, since there is no previous frame to use for quantizing that frame. These preset quantized LSFs are selected to be the closest quantization levels that approximate a neutral flat amplitude spectrum. It is known that, for a frame of signal whose amplitude spectrum is quite flat, the LSFs are distributed equidistantly around the unit circle. Therefore, the indices for the default quantized LSFs are known in both encoder and decoder.

These preset quantized LSFs may also be used in the decoder if a frame is declared invalid by the decoding algorithm, e.g. if burst errors totally destroy it.

4. SIMULATION EXPERIMENTS

A computer simulation has been done to evaluate the proposed method. The standard CELP algorithm was implemented with different LSF quantization schemes, including SQ, unstructured split VQ (using three partitions according to a 3, 3, 4 pattern, and with 8 bits assigned to each subvector) and no quantization, as well as the proposed method. The training of the shift table for PDASQ and of the VQ codebooks was done using 800 and 10240 frames, respectively.

The tests were done with 2290 frames of speech signals outside the training database. Different male and female speakers were included. In the preparation of the training and test vectors, linear prediction is done as in the FS1016 standard, and then the prediction coefficients were converted to the LSFs. Table 2 shows the results of this experimental test, using different objective measures described in [8].

Quantization Method	Bits per Frame	SD (dB)	SSD (dB)	Seg-SNR (dB)
Unquantized	Infinite	----	2.12	9.17
SQ (FS1016)	34	1.49	2.31	8.73
PDASQ	29	1.65	2.41	8.71
Split VQ	24	1.46	2.35	8.93

Table 2: Results of the quality comparisons

It is noteworthy that the computational and storage costs of PDASQ are far less than those of the fastest vector quantization methods such as tree-searched VQ. Table 3 shows the computational complexity and storage costs of different quantization methods. The same assumptions as in [4] have been used for calculating the implementation costs.

Quantization Method	Storage Cost (kbyte)	Computation Cost (No. of Instructions)
SQ (FS1016)	0.44	34
PDASQ	0.74	84
Split VQ	5.00	7,680

Table 3: Implementation costs

The storage cost of PDASQ is almost 70 percent greater than for the simple scalar quantization used in the FS1016 standard. This additional memory is needed for storage of the shift table. Also, the complexity of PDASQ is higher than that of FS1016, but nevertheless the implementation costs of PDASQ are only marginally higher than those of SQ and are far less than for split VQ.

The PDASQ method is an attempt to achieve the same quality as independent SQ with fewer bits. Independent SQ gives the lower distortion bound for the PDASQ quantization method. It is therefore expected that the performance of PDASQ will be inferior to the other methods as a result of the five bits saving in the current realization.

The SD and SSD simulation results show that PDASQ is in fact worse than the corresponding SQ method (FS1016), as was expected. However, the Seg-SNR does not show any significant degradation. There is a simple reason for this difference in assessments. As the principles of PDASQ indicate, this method generally provides the same quantization quality as SQ whenever the variation of each leader and follower pair are either in the same direction or they have slight variations in opposite directions. Obviously, the variations of spectral envelope within voiced frames of speech show this type of behaviour, so that the spectrum quantization in the voiced frames that are quantized by the PDASQ scheme is almost the same as with independent SQ. Since Seg-SNR depends mainly on the high and medium energy frames, approximately the same amount of Seg-SNR should be expected.

However, for the SD and SSD measures, all frames, including silence and very low energy frames, are included in the assessment. In such frames the leader and follower pair of LSFs might have large variations in opposite directions, so that PDASQ may not be able to mimic the SQ method completely, and consequently will have higher distortion. It is predicted that perceptually transparent quality for these frames can be achieved with spectral distortion values greater than 1. Informal listening tests show no degradation with PDASQ compared to the scalar quantization method.

Moreover, in transitional frames (from voiced to unvoiced or vice versa) related LSF pairs might also have large variations in different directions. In such frames preservation of the temporal variation of the signal is perceptually more important than having a low spectral distortion. Again, the short-coming of PDASQ in comparison with SQ in these frames should not cause any additional perceptual distortion.

5. CONCLUSION

A new scalar quantization method (PDASQ) for the spectral coding of speech was introduced in this paper and the training of a corresponding shift table was described. Several techniques were proposed for enhancing its performance, and the performance of the new method was compared with other scalar and vector quantization techniques through experimental tests. Compared to scalar quantization, the new method offers significant bit savings at small computational and storage cost and with little effect on quantization quality.

ACKNOWLEDGMENTS

The first author wishes to thank Jahad Daneshgahi, Islamic Republic of Iran for supporting him during this research.. This research was also assisted by GIRD Grant No. 17016.

REFERENCES

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

[2] F.K. Soong and B.H. Juang, "Line spectrum pair (LSP) and speech data compression", *Proc. ICASSP*, vol. 1, pp. 1.10.1-4, 1984.

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

[4] H.R. Sadegh Mohammadi and W.H. Holmes, "Low cost vector quantization methods for spectral coding in low rate speech coders ", *Proc. ICASSP*, vol. 1, pp. 720-723, 1995.

[5] S. Arai and S. Ozawa, "An efficient coding using the correlation between time series of differential LSP parameters", *Electronics and Communications in Japan*, Part 3, vol. 72, no. 3, pp. 13-22, 1989.

[6] E. Erzin and A.E. Cetin, "Interframe differential vector coding of line spectrum pairs", Presented at 26-th Conf. on Information Sciences and Systems, Mar. 1992.

[7] C.C. Kuo, F.R. Jean and H.C. Wang, "New LSP encoding method based on two-dimensional linear prediction", *IEE Proceedings-I*, vol. 140, no. 6, pp. 415-419, Dec. 1993.

[8] 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 Techn.*, pp. 118-123, Dec. 1994.