Proc. Biennial Symp. Commun. (Kingston, ON), pp. B5.1–B5.4, June 1984

Vector Quantization in Adaptive Predictive Coding of Speech

MARK ABRAMSON and PETER KABAL

The design and implementation of vector quantizers have recently attracted considerable attention in the speech coding field. In this paper, vector quantization is applied in an adaptive predictive coder, both to code the parameters of the linear predictor and to code the residual signal. Traditional speech coders have applied scalar quantization, i.e. coefficient by coefficient quantization, to these quantities.

1. Adaptive Predictive Coders

Fig. 1 shows a block diagram of an adaptive predictive coder (APC) used for speech. An estimate of the input signal, $S(z)$, based on the past reconstructed signal is formed by the predictor filter, $A(z)-1$. The purpose of this (formant) predictor is to remove sample-to-sample correlations. This filter forms the weighted linear combination of past samples, typically using 8–16 past values. It is the weights or coefficients of this predictor filter which must be transmitted to the decoder. The figure also shows a second predictor filter, $B(z)-1$, which operates over longer time lags. After whitening by the formant predictor, the residual speech signal still contains periodic components due to the pitch excited nature of speech. The pitch predictor filter removes correlations at lags corresponding to the pitch period of the speech. This filter typically has from 1–3 coefficients which must also be coded for transmission.

The residual signal after prediction, $E(z)$, is scaled and coded for transmission. The scaling parameter, σ, is coded separately for transmission.

2. Design of Vector Quantizers

A vector quantizer takes a vector of input data, $\mathbf{x} = \{x_n\}$, and finds an output vector, $\hat{\mathbf{x}}$ drawn from a reproduction alphabet $\mathbf{y}_i$ of N vectors. The output vector is chosen to minimize some distortion criterion so that $d(\mathbf{x},\mathbf{y}_i) \leq d(\mathbf{x},\mathbf{y}_j)$ for all j. Both the design (choice of the $\mathbf{y}_i$) and the coding of input data (finding the appropriate index i so that $\mathbf{y}_i$ is the representative for $\mathbf{x}$) is impractically complex for arbitrary configurations when the number of representative vectors becomes large. Instead, tree-searched codebooks are used.

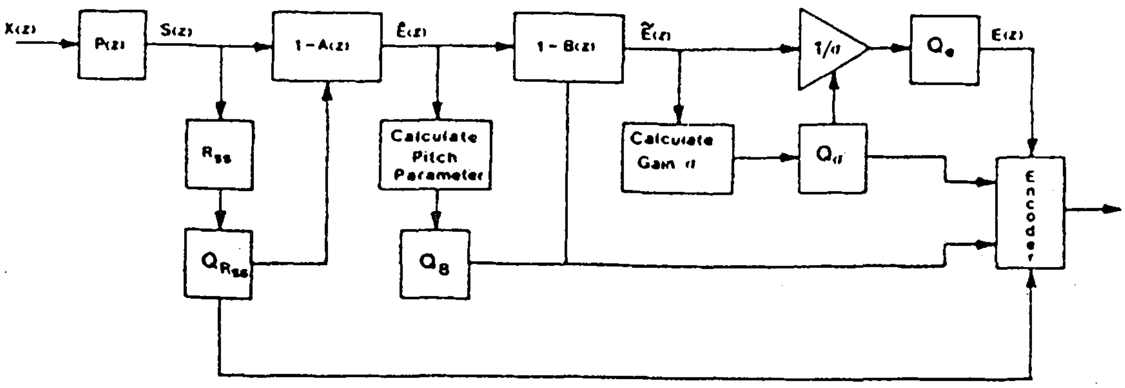
2.1 Tree-Searched Codebooks

In the design stage of a tree-searched codebook, a set of training vectors is used. These vectors represent points in an N-space. The first step is to partition this space into a small number of regions, each containing a representative vector which the the centroid (with respect to the weighting implied by the distortion measure) of the region. Iterative clustering techniques are applied to find a good partitioning of the space. The next stage in the design of the tree-searched codebook is to further subdivide each region (which is now considered fixed) into a small number of partitions to be further optimized iteratively. This process of subdivision is carried out until the desired number of representative vectors is reached. Using this tree structured approach, a codebook can be designed in reasonable times.

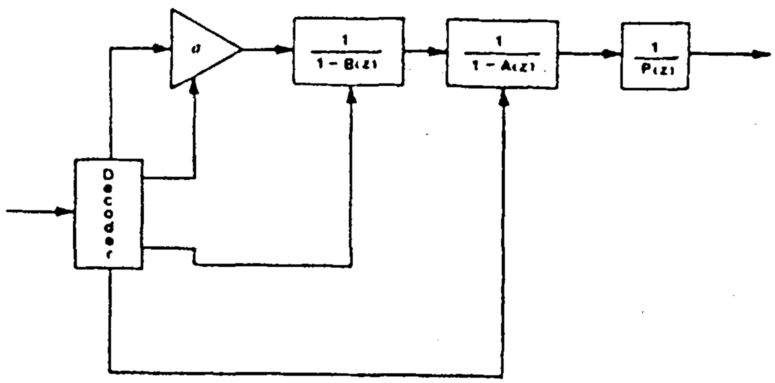
The coding of a new input vector proceeds by following the tree of partitions defined in the design stage. At the first level in the tree, a decision is made as to which region is to be used. At subsequent levels in the tree, the subpartitions of this region are identified. The decoder uses the index of the subpartition to determine a representative vector to use as its output. The penalty

P. Kabal is with the Department of Electrical Engineering, McGill University, Montreal, Quebec, H3A 3A4 and INRS-Télécommunications, Université du Québec, Verdun, Quebec, H3E 1H6.

M. Abramson was with the Department of Electrical Engineering, McGill University, Montreal, Quebec, H3A 3A4. He is now with the Communications Research Center, Shirley Bay, Ontario.



(a) Coder



(b) Decoder

Fig. 1 Adaptive Predictive Coding

to be paid for a codebook arranged in the form of a tree is a suboptimal performance arising from the constraint on the form of the partitioning of the space. In addition, searching the codebook by following the tree, while reducing the computational complexity, requires added storage in the form of the representative vectors at the intermediate nodes of the tree.

2.2 Parameter Separation

The complexity of vector quantizers grows with the dimensionality of the space of vectors. Simplifications at the expense of non-optimality are possible by applying separate quantizers to different parameters. In this way, the dimensionality of the vector quantizers can be reduced. In the specific case under investigation, a fully optimal system would apply vector quantization to the combination of predictors and residuals—effectively vector quantizing a block of speech samples. We choose instead to quantize parameters which are only slightly coupled, separately. The quantizers which are separately formulated are those for the formant predictor (10 coefficients), the formant predictor gain (1 parameter), the pitch value (1 parameter), the pitch predictor coefficients (3 coefficients) and the residual signal itself. For the latter, a different block sizes (vector dimensions) and different numbers of representative vectors were used.

3. Vector Quantization in APC

3.1 Formant Predictor

In this study, the formant predictor coefficients were quantized using a vector quantizer. Following Gray and Linde [1], the vector quantizer was designed using a tree-searched codebook. The optimality criterion is the Itakura-Saito distortion measure,

$$d(|X|^2|;1/|A|^2) \triangleq \frac{1}{2\pi}\int_{-\pi}^{\pi}\Big[|XA|^2 - \ln|XA|^2 - 1\Big]\,d\omega\ , \tag{1}$$

where $|X|^2 \triangleq |X(e^{j\omega})|^2$ is the energy density of the input signal and $|A|^2 \triangleq |A(e^{j\omega})|^2$ is the response of the predictor filter. For the purposes of calculation, this distortion measure can be expressed as

$$d(|X|^2;1/|A|^2) = \frac{\alpha}{\sigma^2} + \ln\sigma^2 - \ln\alpha_\infty - 1\ , \tag{2}$$

where α is the residual energy, α_∞ is the residual energy in the limit as the number of predictor coefficients grows, and σ is the filter gain. We choose to use a gain-separated quantizer; the coefficients and the gain are quantized separately. The coefficient quantizer is designed to minimize α and then the gain is quantized using a scalar Lloyd-Max quantizer designed to minimize the mean-square error based on a tabulated distribution obtained from the training sequence. The filter coefficients were quantized using 10 bits and the filter gain using 6 or 7 bits. The coefficients are updated every 240 samples (30 ms for 8 kHz sampling rate).

3.2 Pitch Predictor

The pitch predictor has three coefficients. The vector quantizer for these coefficients was designed using a mean-square error distortion measure. The quantizer uses 4 bits at the first and second levels of the tree, and two bits at the third level, for a total of 10 bits. The pitch value itself is coded with 7 bits. The pitch predictor is updated every 240 samples.

3.3 Residual Signal

The main focus of this work is the vector quantization of the residual signal. The complexity of the quantizer was limited to 8 bits or 256 representative vectors. These bits were allocated to code various block lengths of the residual signal. For instance, 8 bits over a block length of 4 samples corresponds to an average of 2 bits/sample. The quantizers use 4 bits at the first and second levels of the tree, and the remaining bits (if any) at the third level. Mean-square error was used as the optimality criterion.

4. Experimental Results

Training sequences containing over 25 000 vectors were used. For 8-bit quantizers, this is roughly 100 vectors per quantizer region. The vector quantizers were designed using the procedures described earlier. Experience showed that around 10 iterations were needed to perform the clustering used in partitioning a region, to obtain near asymptotic results.

An important case for establishing the performance of the vector quantizer for the residual signal is that for which the number of bits per residual sample is fixed. The performance as a function of block length can then be studied. This result is shown in Fig. 2 for two bits/sample. Significant improvements in signal-to-noise ratio (SNR) are obtained as the block length is increased. As a check, the vector quantizer with 1 sample per block has the same performance as the corresponding Lloyd-Max quantizer designed using a completely different procedure. This performance is in turn better than that obtained for a quantizer with uniformly spaced steps.

Several comments can be made on the results shown in the figure. The formant predictor parameter quantization is relatively effective in that with the small number of bits assigned, little performance loss is seen in quantizing the filter coefficients. The performance in terms of SNR decreases slightly with pitch prediction. The SNR increases with increasing block size for a constant bit rate. Coding blocks of 5 samples increases the SNR about 2.5 dB over that for scalar quantization at the same bit rate.

Subjective evaluation tests were carried out. Based on these tests, the pitch predictor did offer an improved subjective quality, but probably not enough to warrant the computational effort required

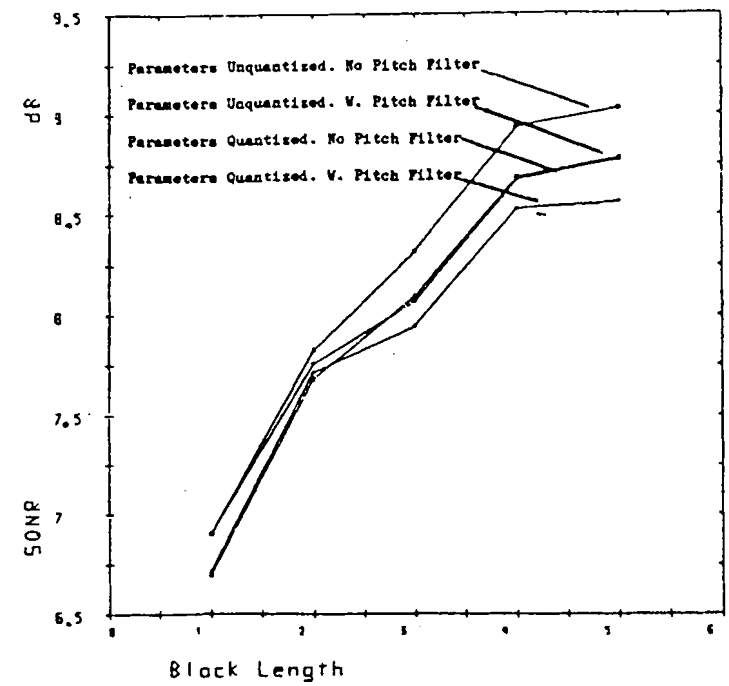


Fig. 2 Two-Bit/Sample Vector Quantizers

to implement it. In subjective quality, a two-bit per sample vector quantizer with 5 samples/block was rated to be around the same quality as a three-bit per sample Lloyd-Max quantizer. In terms of absolute numbers, this quality is about the same as 5 bit/sample log companded PCM.

Vector quantization of the residual at 1 bit/sample was also evaluated. Similar gains with respect to scalar quantization were noted.

The present study has investigated vector quantizers of limited complexity. Even so, the design time for some of the configurations severely taxes computer resources. However, with the rapid progress in the capabilities of digital signal processing hardware, vector quantization may soon become an important practical technique for speech coding at low bit rates.

Reference

1. R. M. Gray and Y. Linde, "Vector quantizers and predictive quantizers for Gauss-Markov sources", *IEEE Trans. Commun.*, vol. COM-30, pp. 381–389, February 1982.