Matrix Quantizer Design for LPC Speech Using the Generalized Lloyd Algorithm

CHIEH TSAO, STUDENT MEMBER, IEEE, AND ROBERT M. GRAY, FELLOW, IEEE

Abstract—**Rate-distortion theory provides the motivation for using data compression techniques on matrices of N LPC vectors. This leads to a simple extension of speech coding techniques using vector quantization. The effects of using the generalized Lloyd algorithm on such matrices using a summed Itakura–Saito distortion measure are studied, and an extension of the centroid computation used in vector quantization is presented. The matrix quantizers so obtained offer substantial reductions in bit rates relative to full-search vector quantizers. Bit rates as low as 150 bits/s for the LPC matrix information (inclusive of gain, but without pitch and voicing) have been achieved for a single speaker, having average test sequence and codebook distortions comparable to those in the equivalent full-search vector quantizer operating at 350 bits/s. Preliminary results indicate that higher quality or lower bit rates may be achieved with enough computational resources.**

I. Introduction

VECTOR quantization (VQ) [1], [2] was introduced as a method of compressing vocal tract models obtained by linear predictive coding (LPC). It combines the two steps of linear prediction and scalar quantization into one step by using techniques derived from rate-distortion theory based on the Itakura–Saito distance measure. The method used a generalized form of Lloyd's algorithm to cluster LPC speech frames, resulting in a *codebook* of LPC vectors which is locally optimized for a training sequence of finite length [3]. The LPC frames, however, can themselves be considered as scalars in a large alphabet, and hence rate-distortion theory indicates that better performance for a given rate or lower rate for a given performance can be achieved by encoding groups of LPC frames or matrices of parameters. As with scalar compression schemes, an alternative to memoryless vector quantization is the use of feedback quantizers such as finite-state vector quantizers [4], [5] which also exploit the interframe memory in speech. Applications of finite-state quantizers to LPC vectors are described in [5]. The matrix quantizers considered here compare very favorably in terms of complexity and performance. At 150 bits/s, a three-vector matrix quantizer achieves distortion performance comparable to full-search vector quantizers operating at 350 bits/s, while the finite-state vector quantizer described in [5] at the same rate achieves distortion performance comparable to full-search vector quantizers operating at 250 bits/s. This is further reinforced by subjective testing, where there was overwhelming preference for the sound of the matrix quantizer when compared to the finite-state quantizer at 150 bits/s. Furthermore, matrix quantizers are much simpler and quicker to design, and therefore potentially superior if adaptation is desired.

Current research in very low bit rate transmission of speech lies primarily in using techniques derived from speech recognition. For example, Wong and Juang [6], [7] isolate speech segments between onset/offset and steady-state points to obtain a variable-rate matrix code with very low average rate (the term matrix quantization is due to Wong and Juang in this context) under speaker and vocabulary constraints, while Schwartz and Roucos [8]–[11] isolate speech segments between steady-state points (generally similar to diphones) in their segment quantizer with the same objective. Although the experiments in [10] show that variable-rate matrix codes generally perform better than fixed-rate matrix codes, nevertheless, fixed-rate matrix codes continue to be of interest because they are inherently simpler to design and implement, particularly for small matrices. In this paper, we present the design of locally optimal, fixed-rate codes using the generalized Lloyd algorithm on N-matrices of LPC vectors. The method focuses on small values of N, ($N \leq 4$), uses a simple distortion computation, and offers a continuous tradeoff between codebook size, rate, and quality. We first present a simple extension of LPC speech coding using vector quantization, followed by some experimental results.

II. Matrix Quantization

We define some terms for convenience. The number N is the block length in a matrix of N LPC vectors. The *codebook rate* in bits per frame is given by the logarithm to base 2 of the number of codewords, or reproduction matrices in the codebook, divided by N. We shall call a matrix code using a codebook with 2^M codewords and a block length of N an M-bit N-code, with rate $R = M/N$ bits/frame. Intuitively, increasing N will reduce the rate M/N, but at the expense of higher average codebook distortion. To keep the codebook distortion constant, we also need to increase the number of codewords, which increases M. However, because speech is highly structured, M increases less quickly than N in practice for constant

Manuscript received May 16, 1984; revised September 28, 1984. This work was supported in part by the Singapore Government, the Joint Services Electronics Program at Stanford University, the National Science Foundation, and by the Industrial Affiliated Program of the Stanford University Information Systems Laboratory.

The authors are with the Information Systems Laboratory, Department of Electrical Engineering, Stanford University, Stanford, CA 94305.

0096-3518/85/0600-0537$01.00 © 1985 IEEE

codebook distortion. The net result is decreasing transmission bit rate with rising N for comparable vocoding quality. Where N is fixed (as in our study here), we also have the advantage of fixed rate encoding.

The heuristic argument above is expressed more precisely in Shannon's source coding theorem [12], [13], applied to the particular case of LPC speech. Looked at another way, a matrix quantizer for LPC speech is analogous to a waveform vector quantizer where the source symbols are LPC vectors instead of individual speech samples.

The technique of speech coding using vector quantization has been dealt with in detail elsewhere [1], [2]. We now present a straightforward extension to the matrix case. Essentially, this consists of taking a training sequence of K speech frames, with K being some (large) integer, and blocking the sequence into LPC N-matrices. The generalized Lloyd algorithm is then applied to this collection of LPC matrices using a summed Itakura–Saito distortion measure, in exactly the manner described in [14]. A codebook of reproduction matrices is then obtained, and this is the matrix quantizer. In the paragraphs below, and in Appendixes I and II, we develop some notation for matrix quantizer design, restate informally the generalized Lloyd algorithm in the matrix notation so developed, and show how centroids using the summed Itakura–Saito distortion measure may be computed.

Let $\boldsymbol{x}$ be a vector of LPC coefficients $(\log \alpha_M, r_x(0), \cdots, r_x(M))$, where M is the order of the LPC filter, α_M is the gain of the filter, and $r_x(j), j = 0, \cdots, M$ are the $M + 1$ sample autocorrelations for a speech frame [1]. Then, if N is some integer ≥ 1, define the $(M + 2) \times N$ matrix $\boldsymbol{X}$, where $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N]$. Furthermore, define a reproduction matrix $\boldsymbol{Y}$ to be a matrix of LPC vectors $[\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_N]$, where each $\boldsymbol{y} = (\log \sigma^2, r_a(0), \cdots, r_a(M))$, where each $r_a(j), j = 0, \cdots, M$ are the autocorrelations of the inverse filter coefficients and σ is the gain for the LPC model. Consider the distortion measure $D(\boldsymbol{X}, \boldsymbol{Y})$ where

$$D(\boldsymbol{X}, \boldsymbol{Y}) = \frac{1}{N} \sum_{i=1}^{i=N} d_{IS}(\boldsymbol{x}_i, \boldsymbol{y}_i)$$

with

$$d_{IS}(\boldsymbol{x}, \boldsymbol{y}) = \frac{r_a(0)r_x(0) + 2 \sum_{i=1}^{i=M} r_a(i)r_x(i)}{\sigma^2} - \log \alpha_M - 1$$

being the usual modified Itakura–Saito distortion measure [1]. It was shown in [3] that a generalized centroid for $d_{IS}(\cdot, \cdot)$ exists and that the generalized Lloyd algorithm converges to a local minimum for $d_{IS}(\cdot, \cdot)$ in the case of a finite training sequence, since $d_{IS}(\cdot, \cdot)$ is convex and differentiable. It follows that $D(\cdot, \cdot)$, which is a finite sum of $d_{IS}(\cdot, \cdot)$ is also convex, from the properties of convex functions, and differentiable. Hence, by the results in [3], a generalized centroid with respect to $D(\cdot, \cdot)$ exists, and the generalized Lloyd algorithm will converge to a local minimum in the training sequence case. All that needs to be shown now is that the centroid with respect to $D(\cdot, \cdot)$ can be easily computed. This is done in Appendix I. An informal statement of the Lloyd algorithm using the matrix notation above is given in Appendix II.

Some remarks about the use of training sequences with the Lloyd algorithm are appropriate in the matrix case. The training sequence of matrices $\{\boldsymbol{X}\}$ is obtained by sliding the first matrix along the training sequence of speech frames, so that the jth matrix in the sequence is given by $\boldsymbol{X}_j = [\boldsymbol{x}_j, \boldsymbol{x}_{j+1}, \cdots, \boldsymbol{x}_{j+N-1}]$. This sliding block technique ensures that we obtain the greatest number of matrices from a training sequence of any given length, and that all phoneme transitions in speech present in the training sequence are captured.

A test sequence is encoded by blocking the sequence into matrices of N LPC vectors (without sliding) and transmitting the codeword index of the codeword nearest to each block relative to the distortion $D(\cdot, \cdot)$. That is, if the test sequence is given by the sequence of LPC frames $[\boldsymbol{t}_1, \boldsymbol{t}_2, \cdots, \boldsymbol{t}_j \cdots]$, then we block the sequence into matrices $\boldsymbol{T}_j = [\boldsymbol{t}_{N(j-1)+1}, \boldsymbol{t}_{N(j-1)+2}, \cdots, \boldsymbol{t}_{Nj}]$. Then if $\{\boldsymbol{Y}\}$ is the matrix codebook of reproduction vectors obtained above, the transmitted codeword index m for the jth matrix in the test sequence $\boldsymbol{T}_j$ is given by

$$m: D(\boldsymbol{T}_j, \boldsymbol{Y}_m) \leq D(\boldsymbol{T}_j, \boldsymbol{Y}_k) \quad \text{for} \quad \boldsymbol{Y}_k \in \{\boldsymbol{Y}\}$$

where $k = 1, \cdots, B$, B being the number of reproduction matrices in the codebook $\{\boldsymbol{Y}\}$. At the receiver, which will possess a copy of the codebook $\{\boldsymbol{Y}\}$, the matrix $\boldsymbol{Y}_m$ will be synthesized in the normal way. Note that the pitch and voicing parameters are independent of the coding of the LPC information and may be separately transmitted by any previously established method (e.g., using the fake process trellis code as in [6]).

We should add that the whole encode–decode scheme is noted for its simplicity (fixed rate coding, no time-warping required in the distortion computation). Furthermore, no initialization process is required at the start of the transmission, as is required in any comparable finite-state vector quantization scheme.

III. Experimental Results

The experiments were divided into the following two parts.

1) Showing that the generalized Lloyd algorithm converges in practice under the distortion $D(\cdot, \cdot)$. Also, a sequence of codebooks using a relatively short (5000) frame multiple speaker training sequence was generated to show the general trend of bit rate versus distortion obtained via matrix quantization. A short single speaker test sequence (600 frames) was then used on these codebooks to show the effects of a short out-of-train sequence on a codebook generated with a short training sequence. Specific matrix 3-codes and 2-codes were then generated with a longer training sequence (15 000 frames) and these were compared to full-search VQ codebooks of essentially the same average distortion using four test (out-of-train) sequences, each about 2000 frames long. Only a single

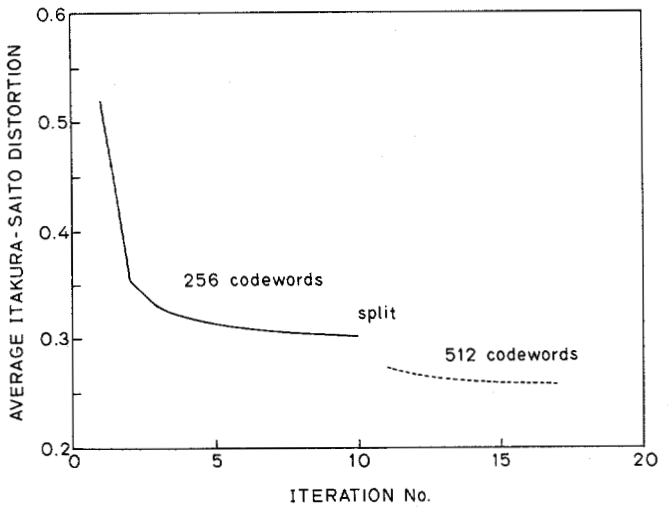


Fig. 1. Convergence of generalized Lloyd algorithm for 9 bit matrix 3-code.

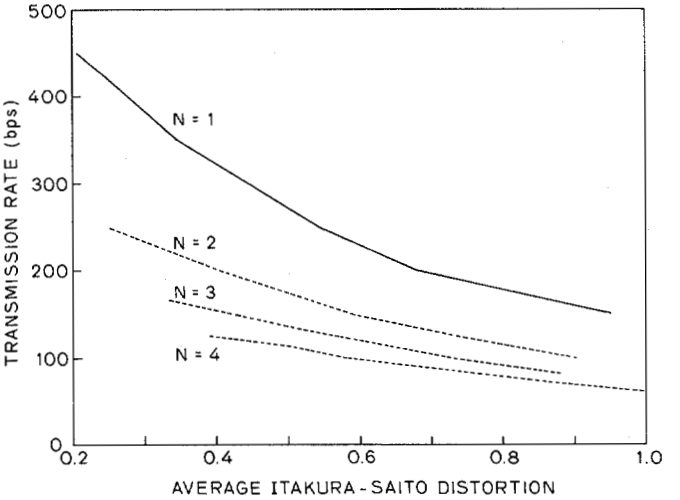


Fig. 2. Transmission rate versus MQ codebook distortion.

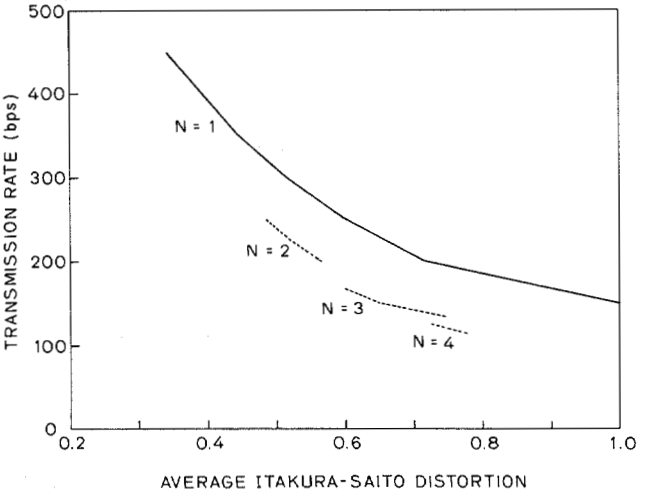


Fig. 3. Transmission rate versus average distortion for test sequence.

speaker was used in both test and training sequences for this latter experiment.

2) Subjective tests comparing the out-of-train single-speaker test sequence vocoded using various matrix and vector quantizers. Intelligibility tests on single-speaker training and test sequences, consisting of reproducing the contents of a 1 min out-of-train test sequence vocoded using a 9-bit 3-code at 150 bits/s for the LPC matrix. These latter tests are not conclusive owing to the fact that the LPC speech used in our experiment was made under difficult conditions, but serve mainly to indicate that matrix quantizers designed under the distortion measure $D(\cdot, \cdot)$ have the potential for producing intelligible speech at lower bit rates.

In all cases, the training sequence had to be kept relatively short (15 000 frames or less) because of the constraints imposed by limited computational resources. This necessarily meant that the codebooks generated would be 'tuned' to the data in the training sequences, making the system speaker and vocabulary dependent. While it is conceivable that true speaker and vocabulary independence may be achieved with a training sequence that is long enough and a codebook that is large enough, such a task is beyond the scope of this study. The details for the test and training sequences are given in Appendix III.

A. Convergence of the Lloyd Algorithm and Distortion Measurements

In Fig. 1, we show the convergence of the generalized Lloyd algorithm for $N = 3$ and a 15 000 frame single speaker training sequence. The initial guess codebook had 256 codewords in it. This converged to a local optimum after 10 iterations (percentage change in distortion ≤ 0.5 percent with the next iteration). The perturbation matrix ϵ was then used to double the number of codewords, and the graph shows convergence to a 512-codeword codebook after the 17th iteration. In Fig. 2, we show plots of transmission bit rate against average codebook distortion, computed as in the previous section, for various block lengths N from 1 (full-search VQ) to 4. We note that for any given value of average distortion, increasing N decreases the bit rate of transmission for the appropriate matrix quantizer. Isolated experiments with longer training sequences indicate that these trends are representative of those obtained with very long training sequences (see the next section).

In Fig. 3 we show plots of transmission bit rate against average distortion obtained with a 600-frame out-of-training test sequence. Because of the shortness of the training and test sequences used, the average distortion with this test sequence is significantly higher than the average codebook distortion. This seems to be particularly true in the matrix case, where the gains obtained with increasing N appear to be substantially less than the codebook distortions would seem to indicate. Nevertheless, we feel that with a substantially longer training sequence, the differences between a test sequence and the training sequence would be considerably smaller. Results from the next section appear to substantiate this claim.

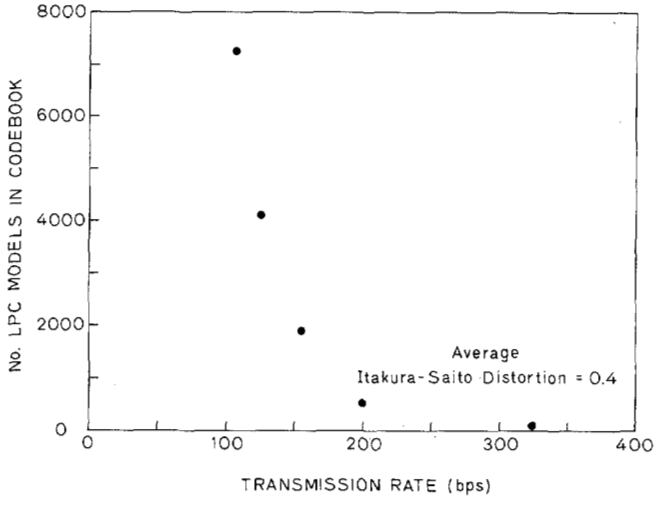


Fig. 4. Codebook size versus transmission rate.

Codebook size, however, also increases very rapidly and at very low bit rates, codebooks with several thousand matrices are required, particularly for $N \geq 4$, if the distortion is to be acceptable. This greater than exponential growth in codebook size is a limitation of the method (vector quantization based methods suffer from exponential codebook growth in general) at present. However, large codebook size seems inherent in current very low-bit rate systems. For example, [6] reports codebook sizes of 3478 matrices comprising matrices with an average size of 13 10-ms frames, giving 45 214 LPC models in the codebook for an overall LPC matrix bit rate of 150 bits/s without gain, pitch, and voicing. Fig. 4 shows the increase in the number of LPC models in the codebook with decreasing transmission bit rate, for a fixed distortion level. Naturally, the number of LPC models for a given rate would increase substantially as the distortion decreases, so the figures implied in the curve cannot be directly compared to those reported in [6]. The trend appears to be approximately proportional to $(1/R)2^{1/R}$ where R is the transmission bit rate. Extensions of shape-gain vector quantizers [1] are possible, however, and the design of shape-gain matrix quantizers is the subject of current research. These could alleviate the problem of very large codebook size considerably.

We now present results obtained from a 15 260-frame single speaker training sequence and four single-speaker test sequences for matrix 3-codes and matrix 2-codes, and compare these against figures obtained for 7 bit and 8 bit VQ codes obtained from the same training and test sequences.

Tables I–IV show the codebook distortions and test sequence distortions obtained for 9 bit and 10 bit matrix 2-codes and 3-codes, respectively. For the matrix code, the average test-sequence distortion is given by

$$\frac{1}{L}\sum_{j=1}^{j=L} \min_{Y\in\{Y\}} D(\boldsymbol{T}_j, \boldsymbol{Y})$$

where $\boldsymbol{T}_j$ is the jth (nonsliding) block in the test sequence, as defined previously, and L is the number of matrices in the test sequence. The test-sequence distortion computation for the VQ case is exactly the same with $N = 1$. We note that in the case of the 9 bit matrix codes, where we have a compression ratio of about 1:30 (for our purposes, the compression ratio is defined by the number of matrices in the training sequence divided by the number of matrices in the codebook) the increase in distortion for the four test sequences was no worse than 1.14 dB, and was typically significantly less than that (test sequence number 3 seemed to give higher distortion). No part of the

TABLE I
512 CODEWORDS, MATRIX 3-CODE, CODEBOOK DISTORTION = 0.257

test seq	distortion	dB increase	no. frames
1	0.287	0.488	1900
2	0.289	0.516	1900
3	0.325	1.02	2420
4	0.271	0.225	2500

mean = 0.293, std dev = 0.020, av dB increase = 0.569

TABLE II
1024 CODEWORDS, MATRIX 3-CODE, CODEBOOK DISTORTION = 0.220

test seq	distortion	dB increase	no. frames
1	0.266	0.832	1900
2	0.267	0.844	1900
3	0.296	1.298	2420
4	0.246	0.493	2500

mean = 0.269, std dev = 0.018, av dB increase = 0.873

TABLE III
512 CODEWORDS, MATRIX 2-CODE, CODEBOOK DISTORTION = 0.213

test seq	distortion	dB increase	no. frames
1	0.247	0.628	1900
2	0.243	0.573	1900
3	0.277	1.140	2420
4	0.225	0.229	2500

mean = 0.248, std dev = 0.019, av dB increase = 0.661

TABLE IV
1024 CODEWORDS, MATRIX 2-CODE, CODEBOOK DISTORTION = 0.179

test seq	distortion	dB increase	no. frames
1	0.221	0.932	1900
2	0.221	0.936	1900
3	0.252	1.407	2420
4	0.206	0.617	2500

mean = 0.225, std dev = 0.017, av dB increase = 0.993

TABLE V
128 CODEWORDS, VQ CODE, CODEBOOK DISTORTION = 0.240

test seq	distortion	dB increase	no. frames
1	0.250	0.176	1900
2	0.244	0.068	1900
3	0.290	0.817	2420
4	0.252	0.216	2500

mean = 0.259, std dev = 0.018, av dB increase = 0.331

TABLE VI
256 CODEWORDS, VQ CODE, CODEBOOK DISTORTION = 0.193

test seq	distortion	dB increase	no. frames
1	0.212	0.420	1900
2	0.205	0.275	1900
3	0.244	1.023	2420
4	0.203	0.237	2500

mean = 0.216, std dev = 0.017, av dB increase = 0.489

mq m - N = m-bit matrix N-code

vq m = m-bit full-search code

test sequences was contained in the training sequence. Furthermore, while some of the vocabulary in the test sequences does inevitably overlap with that in the training sequence, the training sequence was made from random newspaper clippings about medicare and Queen Elizabeth II, while the test sequences were about a day at the zoo and a passage from a fairy tale. In the case of the 10 bit code, where we have a compression ratio of about 1:15, there was a greater increase in the test sequence distortion figures (no worse than 1.41 dB). We should remark that the training and test sequences used are still too short for us to draw precise conclusions about the limiting behavior of matrix quantizers as the training sequence becomes very long, but the distortion figures obtained above do suggest that the matrix codebooks obtained are not very sensitive to out-of-train test sequences which are made under recording conditions identical to the training sequence, for a single speaker.

Tables V and VI show the training and test sequence distortions for 7 bit and 8 bit full-search VQ ($N = 1$, $R = 7$, and $R = 8$). We note that the average codebook distortion is only slightly lower than that for a 9 bit matrix 3-code ($R = 3$). While the difference between the test sequence distortions is a little higher, we attribute this to the higher compression ratios used for the full-search codes (1:119 and 1:60 for 7 bit and 8 bit codes, respectively). Tables VII and VIII compare the distortion figures for matrix and VQ codes. Notice that the differences in codebook and test sequence distortions between the matrix codes and the VQ codes is at most 1 dB, and is typically significantly less. For all practical purposes, this difference is negligible. However, the 9 bit matrix 3-code ($R = 3$) transmits at 150 bits/s, while the 7 bit VQ code ($R = 7$) transmits at 350 bits/s. Similarly, the 9 bit matrix 2-code ($R = 4.5$) transmits at 225 bits/s while the 8 bit VQ code ($R = 8$) transmits at 400 bits/s. This implies that, at least from the point of view of average IS distortion, there is a reduction in bit rate by greater than 50 percent for the 9 bit matrix 3-code, while the 9 bit matrix 2-code gave a 43.8 percent reduction. It is not inconceivable that with 11 or 12 bit 4-codes and 5-codes, we can achieve even lower bit rates (and this seems further strengthened by the trends implicit in Fig. 2). The production of reasonable codes of this size, however, requires training sequences of about 150 000 frames in length, together with very large amounts of CPU time, and we shall leave this exercise to organizations which are better endowed to cope with such burdens. In any case, as the matrices grow larger, variable-rate codes become more efficient, as shown in [10].

TABLE VII

mq9-3 vs vq7			
	average distortion		
	mq9-3	vq7	dB difference
codebook	0.257	0.240	0.294
testseq1	0.287	0.250	0.605
testseq2	0.289	0.244	0.741
testseq3	0.325	0.290	0.496
testseq4	0.271	0.252	0.302

TABLE VIII

mq9-2 vs vq8			
	average distortion		
	mq9-2	vq8	dB difference
codebook	0.213	0.193	0.445
testseq1	0.247	0.212	0.653
testseq2	0.243	0.205	1.074
testseq3	0.277	0.244	0.560
testseq4	0.225	0.203	0.437

B. Subjective Tests

A subjective comparison between vocoded speech using matrix quantizaton and that using full-search VQ and finite-state VQ (designed with stochastic iteration) [5] with varying rates was made with the out-of-train test sequence

TABLE IX

Scores for MQ/VQ/FSVQ Comparison				
Pair		Score		
1.	2.	1 better	2 better	Undecided
mq9-2	vq4	12	0	0
(225 bps)	fsvq3	10	0	2
	vq5	8	0	4
	vq6	7	0	5
	vq7	6	3	3
	vq9	4	0	8
mq9-3	vq4	12	0	0
(150 bps)	fsvq3	12	0	0
	vq5	3	1	8
	vq6	7	1	4
	vq7	5	4	3
	vq9	4	2	6
mq10-4	vq4	4	4	4
(125 bps)	fsvq3	5	4	3
	vq5	0	7	5
	vq6	1	8	3
	vq7	1	10	1
	vq9	0	12	0

Legend:
mqM-N = M-bit matrix codebook with N LPC vectors per matrix
vqM/fsvqM = M-bit full-search/finite state codebook
No. of training matrices used: approx 5000 (sliding blocks on train)

and the multiple-speaker training sequence in the previous part (those used to obtain Figs. 2 and 3). Pitch and voicing information was obtained from the ILS software made by Signal Technology Inc., Santa Barbara, CA. They were not encoded in any way, and were directly used to process the vocoded sequences. Since the issue here is the compression of the LPC information, only this was subjected to compression. Note that pitch and voicing can be compressed using existing methods (e.g., the trellis coder for pitch reported under development in [6] which adds another 25 bits/s to the rate). Results were gathered from 12 listeners, each of which was asked to put down their personal preference for one of a pair of sequences on an *A–B* comparison in terms of intelligibility and general speech quality. The sequences were recorded on cassette tape and played back on loudspeakers from Radio Shack. The results are shown in Table IX. The matrix coded sequences at 225 bits/s clearly out-performed full-search VQ at higher bit rates, and was even judged competitive with full-search VQ at 350 bits/s. At 150 bits/s, the matrix code clearly out-performed both full-search VQ at 200 bits/s and finite state VQ at 150 bits/s. There was some indecision with full-search VQ at 250 bits/s owing to the fact that the order of presentation was reversed for this particular comparison, but those that did decide still preferred the matrix code. The matrix code still seemed competitive at 300 and 350 bits/s. For the matrix code at 125 bits/s, there was overwhelming preference for the VQ codes at the higher bit rates, although it still appeared competitive with finite state at 150 bits/s and full-search at 200 bits/s. The reason for this marked degradation in quality was that this code had 1024 codewords instead of the 512 in the previous matrix codes. Since only 5000 training matrices were used, there was clearly an insufficient number of training matrices to produce a good 10 bit 4-code, and this resulted in a marked degradation in quality. Some remarks should be made regarding the results obtained with the matrix codes at 150 and 225 bits/s and full-search VQ at 450 bits/s. At 450 bits/s, the full-search VQ code began to show more warble than the matrix codes, and this generally pushed some listeners to favor the matrix codes. Listening with headphones, however, indicated that there was degradation in the clarity of the articulation of words, particularly consonants, between 450 and 150 bits/s, but only slight degradation between 450 and 225 bits/s.

The sound of the matrix codes is generally quite different from the kind of speech produced by full-search or finite-state codes. The matrix codes, while retaining intelligibility to a surprising degree at such low bit rates, manifested distortion in the character of the speech reproduced, and this became more marked with lower bit rates. The matrix codes seemed to impart a foreign accent to the speaker (in particular, the "bad" code at 125 bits/s had a very "gargly" quality and gave the speaker a cockney accent). In contrast, the full-search VQ and finite-state VQ codes sounded more muffled at the lower rates, but without much distortion of the character of the speech. It would appear that the matrix codes could somehow approximate the right phonemic units in speech, but not exactly the ones uttered. This seems to be the main effect of the higher average distortions noted in Fig. 3 with the matrix codes.

In order to give some indication of how intelligible speech at 150 bits/s using matrix coding is, we decided to conduct an informal intelligibility test. It should be mentioned at the outset that the test was conducted using LPC speech made under difficult conditions. The training and test sequences were made with an 8-bit D/A converter in a room with air-conditioning noise. To minimize this, the speaker had to move very close to the microphone (about 3 or 4 in), so that there were substantial variations in average gain owing to movements of the speaker's head in some places. Limitations in disk storage and CPU cycles owing to a very heavily used VAX limited the length of the training sequence to about 5 800 20 ms frames, which is just under 2 min of speech. The length of the test sequence was about 1 min and 16 sec. All this resulted in training and test sequences with a low (under 30 dB) signal to noise, and considerable average gain variation in some places, resulting in LPC speech containing some of the problems one might expect in a practical environment. With a 9 bit 3-code (512 codewords, rate 150 bits/s), there were only 11 matrices per codeword in the training sequence. This compression ratio is only just adequate for obtaining a good average to every codeword in the codebook. As usual, the uncoded voicing and pitch parameters

from the original test sequence obtained via the ILS package, was used in vocoding the sequence.

The training sequence was made with a single speaker reading a newspaper clipping about Queen Elizabeth II and the salary of the Royal Family. The test sequence was made from a child's essay about a day at the zoo. Care was taken that none of the listeners had significant prior knowledge of the content of the test sequence. The vocoded test sequence was taped and played back to each of the listeners. The listeners were required to write down the contents of the test sequence, dictation-style. They were also allowed to back up to any portion of the sequence they wanted to listen to again as often as they wanted (in practice, the limit seemed to be about four times). Eight listeners correctly identified on average about 81 percent of the 90-word text, with a standard deviation of 3.6 percent.

The ability of the listeners to make sense of the text varied somewhat. The three highest scores were obtained from people who have heard vocoded or synthetic speech before. The lower scores were from those who have not heard such speech. Two of the people approached could not understand the sequence at all. They were discounted from the test. The nature of much of the vocoded speech was akin to heavily accented English, and some people have difficulty with that, especially people whose native language is not English. Most of the errors for all the listeners occurred in two badly garbled places in the text. All seemed to agree that the beginning and ending sentences were clear and easy to comprehend. We attribute most of this to the codebook size, which was dictated by the shortness of the training sequence. There were not enough codewords to adequately capture all transitions present in the test sequence. This is not really surprising, since the quality is comparable to that obtained with 7 bit VQ, which is less than fully intelligible. Some of the problems were also caused by the gain variation mentioned and one or two incorrect voicing decisions.

The point of all this, however, is not to implement a 150 bits/s speech vocoder, but to demonstrate that matrix quantizers designed using the generalized Lloyd algorithm have the potential for synthesizing intelligible speech using fixed-rate codes at very low bit rates.

IV. General Remarks

We should make a few general remarks about the matrix quantizer design technique outlined above. The first thing one observes is that N times as many adds/multiples are required in computing $D(\boldsymbol{X}, \boldsymbol{Y})$ when compared to standard VQ ($N = 1$). This means (assuming that the total time for all distortion computations is predominant) that a matrix quantizer takes N times as long to train as a standard full-search vector quantizer for the same length of training sequence using the sliding block technique. The other observation is that for a given sequence of speech we want to encode, the number of adds/multiplies required is similar to that for a VQ codebook containing the same number of codewords as the matrix codebook. However, a larger codebook size is typically required to obtain equivalent speech quality for matrix quantization, so this means an increase in the number of adds/multiples in practice. How much this will increase typically depends on N. For a matrix 2-code, only a slight increase is needed (probably at most a factor of 2) while this increases significantly for larger N. From a data-storage viewpoint, a matrix codebook would require N times as much storage as a full-search codebook of the same size. In practice, several times N would be appropriate to obtain equivalent quality.

Up to this point, we have only dealt with the full-search matrix quantizer, in the sense of [2]. Obviously, suboptimal variants are possible, such as tree-search matrix quantizers and shape-gain matrix quantizers [1]. The properties of these variants are subjects for future research. An interesting possibility would be the finite-state matrix quantizer, which might be a good alternative to dealing with huge codebooks with large N.

Within the last one or two years, standard vector quantization techniques have seen growing acceptance as a data reduction method for various speech recognition systems. We listened to individual matrix quantizer codewords with $N = 4$, by breaking down an encoded speech sequence into its constituent matrix codewords, and these codewords sound remarkably similar to groups of phonemes or allophones. Vowel-like sounds seem to be captured especially well by the matrix quantizer, whereas consonants are captured less well. This more general version of the Lloyd vector quantizer for LPC speech might well find some application in the recognition field.

V. Summary

We have presented an extension to the technique of designing vector quantizers for LPC speech motivated by an application of rate-distortion theory. Matrix quantizers designed by the generalized Lloyd algorithm are locally optimal quantizers which generally have substantially lower rates for a given quality of reproduction than standard VQ. They also have the advantage of fixed-rate and use a simple distortion computation. The design technique is a general one which allows a continuous tradeoff between computation time, data storage, and transmission bit-rate for a fixed average distortion by varying the block length. A matrix quantizer operating at 150 bits/s (512 codewords, three-vector matrix) has codebook and test sequence distortions comparable to a full-search vector quantizer operating at 350 bits/s. In contrast, a finite-state vector quantizer at the same rate designed with stochastic iteration has distortion performance comparable to a full-search vector quantizer operating at 250 bits/s. Matrix quantizers are also simpler and quicker to design than finite-state vector quantizers. At 150 bits/s for the LPC matrix on a short training sequence, an average of 81 percent of a 90-word out-of-training text for a single speaker was intelligible based on eight listeners. Subjective MQ/VQ/FSVQ comparison tests conducted among 12 listeners indicated a definite preference for matrix quantizers at com-

parable rates. Codebook distortion figures tend to confirm this trend.

Appendix I

We show how centroids with respect to the distortion measure $D(\cdot, \cdot)$ may be computed. The centroid is defined as that reproduction matrix $\boldsymbol{Y}$ which minimizes the expected distortion of a cell of matrices $\boldsymbol{X}$, where $\boldsymbol{X}$ is taken to be random. Since

$$E(D(\boldsymbol{X}, \boldsymbol{Y})) = \sum_{i=1}^{i=N} E(d_{IS}(\boldsymbol{x}_i, \boldsymbol{y}_i))$$

we may write

$$\min_{\boldsymbol{Y}} E(D(\boldsymbol{X}, \boldsymbol{Y})) \leq \sum_{i=1}^{i=N} E(d_{IS}(\boldsymbol{x}_i, \boldsymbol{y}_i))$$

for any given $\boldsymbol{y}_i$, $i = 1, \cdots, N$. This implies that

$$\min_{\boldsymbol{Y}} E(D(\boldsymbol{X}, \boldsymbol{Y})) \leq \sum_{i=1}^{i=N} \min_{\boldsymbol{y}_i} E(d_{IS}(\boldsymbol{x}_i, \boldsymbol{y}_i)).$$

Conversely, we have

$$\sum_{i=1}^{i=N} \min_{\boldsymbol{y}_i} E(d_{IS}(\boldsymbol{x}_i, \boldsymbol{y}_i)) \leq E(D(\boldsymbol{X}, \boldsymbol{Y}))$$

for any given $\boldsymbol{Y}$, and this implies that

$$\sum_{i=1}^{i=N} \min_{\boldsymbol{y}_i} E(d_{IS}(\boldsymbol{x}_i, \boldsymbol{y}_i)) \leq \min_{\boldsymbol{Y}} E(D(\boldsymbol{X}, \boldsymbol{Y})).$$

Hence, we have

$$\min_{\boldsymbol{Y}} E(D(\boldsymbol{X}, \boldsymbol{Y})) = \sum_{i=1}^{i=N} \min_{\boldsymbol{y}_i} E(d_{IS}(\boldsymbol{x}_i, \boldsymbol{y}_i)).$$

That is, the centroid of a cell of matrices $\boldsymbol{X}$ is merely the matrix which comprises the centroids of the individual component vectors $\boldsymbol{x}_i$ of $\boldsymbol{X}$. In practice, we replace the expectation E with the arithmetic average over the cell. From [15] and [1] we know that the centroid of a component vector $\boldsymbol{x}_i$ is merely the arithmetic average of the sample autocorrelations $r_x(j)$, $j = 0, \cdots, M$ for that component.

Appendix II

An informal statement of the generalized Lloyd algorithm using the matrix notation developed in the text is given.

Suppose an initial guess of B reproduction matrices (codewords) $\{\boldsymbol{Y}_j, j = 1, \cdots, B\}$ exists, and we have a training sequence of LPC matrices $\{\boldsymbol{X}_j, j = 1, \cdots, K\}$, where K is some large integer (these will be known as $\{\boldsymbol{Y}\}$ and $\{\boldsymbol{X}\}$ for short). Assign each matrix $\boldsymbol{X}$ in the training sequence to some codeword $\boldsymbol{Y}$ by minimizing $D(\boldsymbol{X}, \boldsymbol{Y})$ over all the codewords $\{\boldsymbol{Y}\}$, for each $\boldsymbol{X}$ in the training sequence. The training sequence $\{\boldsymbol{X}\}$ is now partitioned into B or fewer cells. Find the centroid $\bar{\boldsymbol{X}}$ of a given cell by averaging each sample autocorrelation coefficient, as follows:

$$\bar{\boldsymbol{X}} = [\bar{\boldsymbol{x}}_1, \cdots, \bar{\boldsymbol{x}}_N]$$

where

$$\bar{\boldsymbol{x}}_j = \frac{1}{L} \sum_{i=1}^{i=L} r_{\boldsymbol{x}_{ji}}(k), \qquad k = 0, \cdots, M \qquad j = 1, \cdots, N$$

with L being the number of matrices in any given cell. The proof that this is the centroid of the matrices for that cell is given in Appendix I. The inverse filter model for each $\bar{\boldsymbol{x}}_j$ is then found by conventional means (e.g., Levinson's algorithm) and the reproduction matrices $\{\boldsymbol{Y}\}$ so obtained from $\{\bar{\boldsymbol{X}}\}$ will constitute the new codebook. This procedure is continued until the percentage change in average distortion given by

$$\frac{1}{K} \sum_{j=1}^{j=K} \min_{\boldsymbol{Y} \in \{\boldsymbol{Y}\}} D(\boldsymbol{X}_j, \boldsymbol{Y})$$

is smaller than some threshold Δ. At this point, a local minimum is approached. If B is equal to the number of codewords we want, then the algorithm ends. Otherwise, we perturb each reproduction matrix by some perturbation matrix $\boldsymbol{\epsilon}$ and repeat the process with an initial codebook of $2B$ reproduction matrices, $\{\boldsymbol{Y}\} \cup \{\boldsymbol{Y} + \boldsymbol{\epsilon}\}$. We perturb the gain term σ^2 by some quantity ϵ for each reproduction vector $\boldsymbol{y}$ in $\boldsymbol{Y}$ in our experiments. Empty cells, that is, codewords around which no matrices cluster, are treated by retaining that codeword for the next iteration. Since the algorithm is only locally optimal, many bad codebooks can exist owing to shallow local optima. However, a good codebook is usually found after one or two different guesses at the initial codebook, or by starting with the centroid of the training sequence and splitting to obtain the desired rate.

Appendix III

The specifications for the training and test sequences used to obtain the experimental results are as follows:

1—Fig. 1: The training sequence was made at Stanford University. It was sampled at 8 kHz, using an 8 bit analog-to-digital converter. Tenth-order LPC analysis using the autocorrelation method was done using a preemphasis of 0.9 and a Hamming window at 160 samples/frame giving standard 20 ms frames. SNR was marginally adequate at about 30 dB. The sequence was 15 200 frames long and contained random extracts from various newspapers read by a single speaker.

2—Fig. 2: The training sequence was obtained from Signal Technology Inc., Santa Barbara, CA. It was sampled at 6.5 kHz, using a 12 bit analog-to-digital converter. Tenth-order LPC analysis using the autocorrelation method was done using a preemphasis of 0.9 and a hamming window at 128 samples/frame giving standard 20 ms frames. SNR was very good at about 50 dB. The sequence was 5000 frames long and contained conversational speech from five speakers.

3—Fig. 3: The test sequence used was also obtained from Signal Technology, Inc., Santa Barbara, CA. Specifications are the same as for (2) except that the sequence was 600 frames long and contained speech from a single speaker. The codebooks were obtained from the training sequence in (2).

4—Tables I–VIII: The four test sequences used have the same specifications as in (1) except that the sequences contained speech based on a child's essay on a day at the zoo and a passage from a fairy tale. The training sequence used for these tests is that in (1).

5—Table IX: The training sequence used was that in (2) and the test sequence that in (3).

References

[1] A. Buzo, A. H. Gray, Jr., R. M. Gray, and J. D. Markel, "Speech coding based upon vector quantization," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-28, pp. 562–574, Oct. 1980.

[2] R. M. Gray, A. H. Gray, Jr., G. Rebolledo, and J. E. Shore, "Rate distortion speech coding with a minimum discrimination information distortion measure," *IEEE Trans. Inform. Theory,* vol. IT-27, pp. 708–721, Nov. 1981.

[3] R. M. Gray, J. C. Kieffer, and Y. Linde, "Locally optimal block quantizer design," *Inform. Contr.,* vol. 45, pp. 178–198, May 1980.

[4] J. Foster, R. M. Gray, and M. Dunham, "Finite-state vector quantization for waveform coding," *IEEE Trans. Inform. Theory,* to be published.

[5] M. Dunham and R. M. Gray, "An algorithm for the design of labeled-transition finite-state vector quantizers," *IEEE Trans. Commun.,* vol. COM-33, pp. 83–89, Jan. 1985.

[6] D. Y. Wong and B. H. Juang, "Vector/matrix quantization for narrow-bandwidth digital speech compression," Final Rep. Signal Technol., Inc., Contract F30602-81-C-0054, July 1982.

[7] D. Y. Wong, B. H. Juang, and D. Y. Cheng, "Very low data rate speech compression with LPC vector and matrix quantization," in *Proc. ICASSP,* Boston, MA, Apr. 1983, pp. 65–68.

[8] S. Roucos, R. Schwartz, and J. Markhoul, "Vector quantization for very-low-rate coding of speech," *Globecom '82 Conf. Record,* Dec. 1982, pp. 1074–1078.

[9] S. Roucos, J. Makhoul, and R. Schwartz, "A variable-order Markov chain for coding of speech spectra," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing,* Paris, Apr. 1982, vol. 1, pp. 582–585.

[10] S. Roucos, R. Schwartz, and J. Makhoul, "Segment quantization for very-low rate speech coding," in *Proc. IEEE Int. Conf. Acoust. Speech, Signal Processing,* Paris, Apr. 1982, vol. 3, pp. 1565–1568.

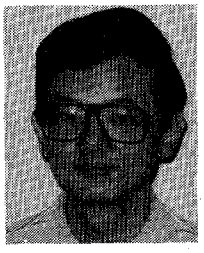
[11] ——, "A segment vocoder at 150 B/S," in *Proc. ICASSP,* Boston, MA, Apr. 1983, pp. 61–64.

[12] R. J. McEliece, *The Theory of Information and Coding.* Reading, MA: Addison-Wesley, 1977.

[13] R. G. Gallager, *Information Theory and Reliable Communication.* New York: Wiley, 1968.

[14] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun.,* vol. COM-28, Jan. 1980, pp. 84–95.

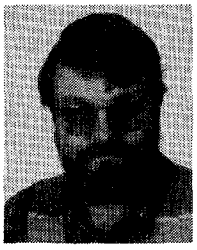
[15] R. M. Gray, A. Buzo, A. H. Gray, Jr., and Y. Matsuyama, "Distortion Measures for speech processing," *IEEE Trans. Acoust., Speech,* Signal Processing, vol. ASSP-28, pp. 367–376, Aug. 1980.



Chieh Tsao (S'84) was born in Singapore on December 27, 1953. He received the B.Sc. degree from the Institute of Science and Technology, University of Manchester, Manchester, England, in 1975 and the M.S. degree from Stanford University, Stanford, CA, in 1981, both in electrical engineering.

Currently, he is a Ph.D. candidate at Stanford University under sponsorshop from the government of Singapore. His research interests are speech coding and recognition.

Mr. Tsao is a member of Sigma Xi. He was awarded the IEE prize for academic excellence at the University of Manchester in 1975.



Robert M. Gray (S'68–M'69–SM'77–F'80) was born in San Diego, CA, on November 1, 1943. He received the B.S. and M.S. degrees from the Massachusetts Institute of Technology, Cambridge, in 1966 and the Ph.D. degree from the University of Southern California, in 1969, all in electrical engineering.

Since 1969 he has been with Stanford University, Stanford, CA, where he is currently a Professor of Electrical Engineering and Director of the Information Systems Laboratory. His research interests are the theory and design of data compression and classification systems, speech and image coding, and ergodic and information theory.

Dr. Gray is a member of the Board of Governors of the IEEE Information Theory Group and served on that Board from 1974 to 1980. He has been on the Program Committee of several IEEE International Symposia on Information Theory and was an IEEE Delegate to the Joint IEEE/USSR Workshop on Information Theory in Moscow in 1975. He was corecipient, with Lee D. Davisson, of the 1976 IEEE Information Theory Group Paper Award and corecipient, with Andrez Buzo, A. H. Gray, Jr., and J. D. Markel of the 1983 IEEE ASSP Senior Award. In 1981 he was a Fellow of the Japan Society for the Promotion of Science and of the John Simon Guggenheim Memorial Foundation from 1981 to 1982. In 1984 he was awarded an IEEE Centennial Medal. He is a member of Sigma Xi, Eta Kappa Nu, SIAM, IMS, AAAS, and the Societé des Ingenieurs et Scientifiques de France.