DIXIEME COLLOQUE SUR LE TRAITEMENT DU SIGNAL ET SES APPLICATIONS

NICE du 20 au 24 MAI 1985

SPEECH BIT RATE REDUCTION BY MARKOV-HUFFMAN CODING OF LPC PARAMETERS

Panos E. Papamichalis

Texas Instruments Inc.
P.O. Box 1443, MS 6400
Houston, Texas 77001, USA

RESUME

Il s' agit d'une méthode de traitement "post quantification", qui diminue le débit de la parole codée par LPC sans affecter la qualité de la parole. Les niveaux de quantification des paramètres LPC sont traités comme les états d' un modèle Markov, et les probabilités de transition qui en résultent, sont utilisées pour la génération des tables de codage Huffman. Lors du codage du signal de la parole, la table de codage qui convient, est choisie en fonction du niveau de quantification du paramètre de la trame précédente. Il a été demontré que le codage Markov-Huffman peut permettre d' économiser en moyenne plus de 20% du débit. Un système moins performant est en cours d' études; il permet une mise en oeuvre plus facile de la méthode sur les circuits de traitement du signal, actuellement disponibles.

SUMMARY

A post-quantization processing method is presented which reduces the bit rate of LPC-coded speech without any effect on the speech quality. The quantization levels of the LPC parameters are treated as the states of a Markov model, and the resulting transition probabilities are used to generate Huffman coding tables. During the encoding of the speech signal the appropriate coding table is selected depending on the quantization level of the parameter in the previous frame. It is demonstrated that Markov-Huffman coding can lead to average savings of more than 20% in bit rate. A suboptimal scheme is also investigated, which can facilitate the implementation of the method on currently available signal processing chips.

SPEECH BIT RATE REDUCTION BY MARKOV-HUFFMAN CODING OF LPC PARAMETERS

INTRODUCTION

In the effort to reduce the data rate of the vocoders without a proportional deterioration of the speech quality, the main consideration has been the choice of the appropriate parameters. Usually, a static representation of the speech production model is used which ignores the dynamic evolution of the speech signal. Some attempts have been made to capitalize upon the dynamic behavior of speech by deterministically tracking the time behavior of speech. However this tracking impacts the speech quality because the values of the parameters themselves are affected.

In the present work, a statistical approach is considered, where a Markov model [1]-[2] is used to represent the behavior of the parameters of interest. Each speech parameter is essentially represented by a finite state machine with an associated matrix of transition probabilities. The time dimension is captured in the transition probabilities of the parameters from one frame to the next. This modeling is the basis of the present bit-saving coding scheme. From the following explanation it should be clear that this method does not operate on the unprocessed parameters but only on the parameters after they have been quantized. Hence it does not have any effect on speech quality, since it acts as a post-processing method on already quantized values. The purpose is to efficiently represent these quantized quantities. The particular choice of the speech parameters is not important for the conceptual development of the method (although it may affect the level of performance), as long as the parameters chosen demonstrate a smooth change from one frame to the next. In the present investigation, the speech is represented by an LPC model, plus the energy and the pitch. In other words, if we use an LPC model of order p, there are p+2 parameters per frame. For the purposes of the current research, all p+2 parameters are treated identically.

MARKOV-HUFFMAN CODING

In the Markov modeling of the LPC parameters, each parameter can be represented as a finite state machine, where the states are the different quantization levels as shown in Fig. 1. Because of the smooth variation of the speech parameters we would expect the quantized parameter to jump more often to adjacent states (quantization levels) than to states further away as we proceed from one speech frame to the next. The frequency of transition is represented by the transition probabilities p(k,j/i), of the k-th parameter reaching state j, given that it was in state i in the previous frame. These conditional probabilities define a first order Markov model. (In an n-th order model the transition probability depends on the n previous states.) Figure 1 shows an example of the transition probabilities for a parameter which is quantized to four levels.

To investigate the effects of adjacency, the transition probabilities were computed on a large data base. Figure

SPEECH BIT RATE REDUCTION BY
MARKOV-HUFFMAN CODING OF LPC PARAMETERS

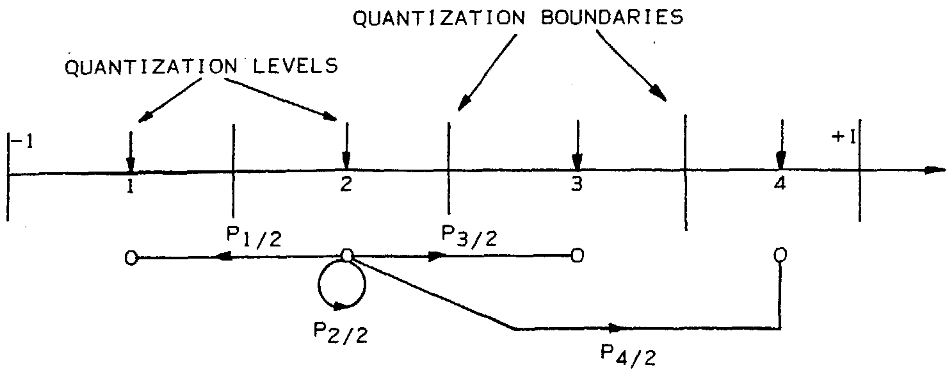

Figure 1. Example of the transition probabilities from the second quantization level of a parameter K_i. K_i is assumed to be quantized to 4 levels.

i \ j	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	35	14	9	6	5	5	5	5	4	3	2	2	2	1	1	1
2	15	23	16	10	8	6	5	4	3	2	2	1	1	1	1	1
3	7	14	22	17	11	7	5	4	3	2	2	2	1	1	1	0
4	4	8	15	21	16	11	7	5	3	2	2	1	1	1	1	0
5	3	5	8	15	21	16	11	7	5	3	2	1	1	1	1	0
6	2	3	5	9	16	20	17	11	7	4	3	2	1	1	1	0
7	1	2	3	5	10	16	19	16	10	6	4	2	2	1	1	0
8	1	2	3	4	6	11	16	20	15	10	5	3	2	1	1	0
9	1	1	2	3	4	7	12	16	19	15	9	5	3	2	1	1
10	1	1	1	2	3	5	7	12	17	19	16	9	5	2	1	1
11	1	1	1	2	2	3	4	7	11	17	21	16	9	4	2	1
12	1	1	1	1	2	2	3	4	7	11	17	23	16	8	3	1
13	1	1	1	1	1	1	2	3	4	6	11	19	25	18	6	2
14	0	1	1	1	1	1	1	2	3	3	6	11	20	29	16	5
15	0	0	1	1	1	1	1	1	2	2	4	5	10	22	34	16
16	0	0	0	1	1	1	1	1	1	2	2	3	4	10	21	53

Figure 2. Transition probability matrix for K_2. In a constant bit rate system, K_2 is assigned 4 bits for this example.

shows the transition probability matrix which corresponds to the second reflection coefficient, for the first order Markov model, when K_2 is quantized into 16 levels. The i-th row gives the conditional probabilities (in %) that K_2 will be quantized to the j-th level, $1 \leq j \leq 16$, given that it was quantized at the i-th level in the previous frame. It is clear from Figure 2 that the adjacent states are preferred, and each conditional probability has a distribution centered around the diagonal element. The probability distribution has a rather small standard deviation, and this was typical for all the parameters considered. This probabilistic behavior suggests that Huffman coding [2]-[3] could be used for data reduction.

In Huffman coding, codewords of variable length are assigned to represent messages of different probability: More probable messages are assigned shorter

QUANTIZATION LEVEL	FIXED WORD	PROBABILITY	HUFFMAN WORD
1	0000	.01	1111011
2	0001	.02	100011
3	0010	.03	10000
4	0011	.04	11101
5	0100	.06	1001
6	0101	.11	011
7	0110	.16	110
8	0111	.20	00
9	1000	.15	101
10	1001	.10	010
11	1010	.05	11111
12	1011	.03	11100
13	1100	.02	111100
14	1101	.01	100010
15	1110	.01	11110101
16	1111	.00	11110100

Figure 3. Example of Huffman coding. The codewords correspond to the eighth row of Figure 2. The expected value of the codeword length is now 3.4 bits instead of 4.

codewords, while less probable messages are assigned longer codewords. The selection of the codewords is done so that when a sequence of bits arrives at the decoder the transmitted codewords are immediately recognized. Huffman coding can be applied to every row of the transition probability matrix, where the probabilities used for coding are the conditional probabilities mentioned earlier. Figure 3 shows an example of the Huffman coding table corresponding to the eighth row of the matrix of Figure 2. The second column gives the codewords which are typically used to represent the different quantization levels. The third column shows the probability of occurrence of each quantization level. Note that using fixed length words we must use 4 bits/parameter, while using Huffman codewords we only use 3.4 bits/parameter on the average.

For the application of the scheme, when a parameter is coded at the i-th level, in the next frame the Huffman codewords corresponding to the i-th row of the transition matrix are used. This method, which will be called Markov-Huffman coding, leads to a variable bit rate system.

The main problem of the above described scheme is the huge amount of storage required for the matrices containing the Huffman codewords. To reduce the storage requirements, a suboptimal coding technique was studied in which the matrix of codewords of each parameter is collapsed to a single row. This is motivated from the observation that each transition probability matrix has the form of Figure 2: The conditional probability distribution of each row looks very much like the previous row, but it is shifted by one position. To implement this idea, a "super-row" is created for each matrix. The elements of the super-row are the average probabilities of jumping j levels away. Then, an approximation of the transition probability matrix is constructed by circularly shifting the super-row. In other words, we store only one row of

Table 1. Average number of bits per parameter for each speech frame for a first order Markov model. Results are presented for both statistical averages and for the 9 test files.

		OPTIMAL		SUBOPTIMAL	
PARAMETER	CONSTANT BIT RATE	MRKV-HUFF TRAINING	AVERAGE OVER TEST FILES	MRKV-HUFF TRAINING	AVERAGE OVER TEST FILES
ENERGY	5	3.5	3.8	3.8	4.0
PITCH	5	1.8	2.0	2.2	2.4
K1	5	3.9	3.7	4.3	4.0
K2	4	3.3	3.0	3.4	3.2
K3	4	3.3	3.2	3.4	3.3
K4	4	3.4	3.2	3.5	3.4
K5	4	3.4	3.3	3.6	3.5
K6	4	3.4	3.3	3.5	3.4
K7	4	3.4	3.3	3.5	3.5
K8	4	3.3	3.3	3.5	3.4
K9	3	2.6	2.6	2.9	2.9
K10	2	1.8	1.7	1.8	1.8
TOTAL	48	37.1	36.6	39.5	38.7

Huffman codewords for each probability transition matrix. Then, to generate the appropriate row, we shift the stored row circularly until its shorter codeword is on the diagonal of the matrix.

RESULTS AND DISCUSSION

In order to have a high level of certainty in the statistical results, a large data base was used for the generation of the general transition probability matrices. The total duration of the training material was about 1 hour of speech and included men and women whose speech was collected both in a sound booth and over local telephone lines. In addition to the training speech, 9 sentences, which were independent of the training set, were used to evaluate the actual performance of the scheme. These test sentences were collected either under high quality conditions or over the local telephone lines and included men, women, and children.

The speech material was sampled at 8 kHz and then LPC-analyzed using a 10-th order model. The analysis conditions were 20 msec frame period and 30 msec analysis window (Hamming window). The LPC parameters selected were the log area ratios. Forty eight bits per frame were distributed among the different parameters as shown in the constant bit rate columns of Table 1.

Table 1 summarizes the results for the optimal and the suboptimal (one row per matrix) coding of the first order Markov model. The average bits on the training data represent the long term statistics, while the 9 test files give an actual example. As can be seen from this table, Markov-Huffman coding can reduce the bits per frame from 48 in the case of constant bit rate to an average of 37.1, a savings of almost 23%. The test data average over the 9 test files is 36.6, in line with the training data average. The results for the individual test files showed a reasonable spread as well. Pitch is most effectively represented through such a model, while on the other hand the gains from K9 and K10 are miniscule.

If Markov-Huffman coding is to be used with voice response systems, additional savings may be gained by using transition probability matrices generated from the particular material to be processed. When speech material from a single speaker was processed, it gave 36.4 bpf for optimal and 40.5 bpf for suboptimal coding. As we see, the differences are not dramatic. Most probably, the transition matrices tend to their long term values quite rapidly.

To investigate the effect of the different frame periods, the Markov-Huffman coding method was applied for three different frame periods: 10, 15, and 20 msec. As expected, the bits/frame decrease as the frame period decreases. The bits per second increase, of course, as the frame period decreases, but the rate of increase is smaller than in the constant bit rate case. Experimentation showed that the average bit rate for the 15 msec frame period is about 2400 bits/sec. This leads to the interesting possibility of improving the speech quality of a 2400 bits/sec system by using a lower frame period (15 instead of 20 msec) combined with Markov-Huffman coding.

CONCLUSION

The present research work has demonstrated that by judiciously choosing the codeword representation of the quantized speech parameters we can have considerable savings in bit rate. This savings is achieved by combining a first order Markov model of the LPC parameters with Huffman coding. It should be emphasized again that this reduction in bit rate does not have any impact on the speech quality, since it operates only on the representation of the quantization levels. The price paid for that gain is increased storage for the codewords and the burden of a variable bit rate system.

REFERENCES

1. A. Papoulis, Probability, Random Variables, and Stochastic Processes, McGraw-Hill, New York, 1965.

2. R.G. Gallager, Information Theory and Reliable Communication, Wiley, New York, 1968.

3. D. A. Huffman, "A Method for the Construction of Minimum Redundancy Codes," Proceed. of IRE, pp. 1098-1101, September 1952.