A NOVEL VARIABLE RATE LPC QUANTIZER FOR HIGH PERFORMANCE SPEECH CODER

Zijun Yang, Jozsef Vass, Yunxin Zhao[†]*, and Xinhua Zhuang*

Department of Computer Engineering
and Computer Science
University of Missouri-Columbia
Columbia, MO 65211

[†]Beckman Institute and Department of
Electrical and Computer Engineering
University of Illinois
Urbana, IL 61801

ABSTRACT

A highly efficient algorithm termed *adaptive forward-backward vector quantization* (AFBVQ) is developed for variable bit rate quantization of linear predictive coding (LPC) coefficients and integrated with the FS1016 Federal Standard Code Excited Linear Predictive (CELP) coder. This results in a high performance low bit rate speech coder called as AFBVQ-CELP which brings in two-fold bit rate reduction by backward LPC indexing and by forward LPC VQ.

In AFBVQ, a previously decoded and temporally close speech signal is *re-segmented* into *overlapping* blocks. As the LPC coefficients calculated from one of those synthetic blocks are spectrally close to the current *unquantized* LPC coefficients, the backward LPC indexing is used to encode the current speech block; otherwise, the forward linear prediction is practised with the split vector quantization supported by a very efficient codebook initialization termed Mixture Gaussian Clustering (MGC) [1].

When compared to FS1016 CELP coder, AFBVQ-CELP reduces the LPC bit rate by 18 bit-per-frame (bpf) at the same spectral distortion. It means the overall bit rate is reduced from 4.8 kbps (FS1016 CELP) to 4.2 kbps. Furthermore, the proposed AFBVQ consistently outperforms the traditional forward LPC VQ by 3 bpf with the same spectral distortion. Subjective listening tests show that with AFBVQ-CELP the LPC bit rate can be further reduced to 8.4 bpf, resulting in 3.94 kbps overall bit rate without compromising the decoded speech quality.

1. INTRODUCTION

Linear prediction plays a center role in various low and intermediate speech coding algorithms. Usually, linear predictive coding (LPC) coefficients are updated periodically and transmitted to the decoder as side information. In virtually all published speech coding algorithms, predictor coefficients are determined based on the current speech block by using the so-called forward linear prediction. Forward linear prediction brings in exclusive transmission of predictor coefficients and extensive data buffering. As opposed to forward linear prediction, backward linear prediction requires neither transmission of predictor coefficients nor data buffering. But its quality is usually inferior to forward linear prediction.

The linear prediction technique is apparently based on the knowledge that the speech signal is nonstationary but its statistics are slowly time-varying. The forward prediction actually exploits only within-block statistical similarity. Having between-block statistical similarity requires the statistics between the current speech block and some temporally close previous speech blocks be close, signified by close sets of predictor coefficients. A method termed long history quantization (LHQ) [2] was proposed to exploit this between-block similarity. The strategies proposed in the paper are different. By allowing previous decoded speech blocks to be overlapped and using the current *unquantized* set of LPC coefficients for matching, not only the chance for more accurate statistical matching increases, but also the higher order backward linear prediction can be applied, further raising the segmental SNR. By adaptation of quantizer design to the new strategies, the between-block statistical similarity of speech signals will be more thoroughly exploited and a significant bit rate reduction is expected.

Briefly, we may describe our novel *adaptive forward-backward vector quantization* (AFBVQ) of LPC coeffi-

This work was supported in part by NSF Award IRI 95-02074 and by NASA Applied Information Systems Research award NAG-2573 and NASA Innovative Research award NAGW-4698 managed by Glenn Mucklow.

cients as follows. A previously decoded and temporally close speech signal is *re-segmented* into *overlapping* blocks. If, and only if, the LPC coefficients calculated from one of those synthetic blocks are spectrally close to the *unquantized* LPC coefficients calculated from the current speech block, the backward LPC scheme shall be applied, i.e., the LPC coefficients based on the previously decoded optimal speech block are used to encode the current block and only the time delay shall be transmitted. In the case that the forward linear prediction is applied, the vector quantization (VQ) is used to encode LPC coefficients. In the paper, we utilize a split vector quantization scheme, in which a vector is split into two separate unequal-length subvectors which are treated independently. The LBG codebook training algorithm represents a local optimization technique, and its performance heavily depends on codebook initialization. A very efficient codebook initialization method termed Mixture Gaussian Clustering (MGC) based upon the work in [1] is also developed. In the paper, we integrate AFBVQ with the FS1016 Federal Standard Code Excited Linear Predictive coder (CELP) [3]. The integration results in a high performance low bit rate speech coder called AFBVQ-CELP which brings in two-fold bit rate reduction by backward LPC indexing and by forward LPC VQ.

The rest of the paper is organized as follows. Section 2 briefly describes *adaptive forward-backward quantization* (AFBQ). Split vector quantization together with MGC codebook initialization is described in Section 3. Performance evaluation of AFBVQ-CELP is illustrated in Section 4. Conclusions are given in the last section.

2. ADAPTIVE FORWARD-BACKWARD LPC QUANTIZATION

As usual, the input speech signal is divided into non-overlapping blocks of M samples and p *forward* LPC coefficients, i.e., $a_1, \ldots, a_p$, are determined based on the current speech block by using, for example, the Levinson-Durbin algorithm.

First, we define the *adaptive forward-backward LPC codebook*, which consists of S code vectors, each having p entries (p is the order of linear prediction). The ith code vector of the adaptive forward-backward codebook is determined by calculating the LPC coefficients, based upon the previously decoded (synthetic) speech block $[y_{n-iK-M},\ y_{n-iK-M+1},\ \cdots,\ y_{n-iK-1}]$ where M is the length of the LPC block and K is the time delay chosen to be $K = M/4$ (see Fig. 1). Then, we use logarithmic spectral distortion (LSD) to evaluate similarity between current and previous sets of LPC coefficients. The one with the smallest distortion, i.e., $\mathrm{LSD}^{(index)}$ with $index = \arg\min_i \mathrm{LSD}^{(i)}$, is selected. If $\mathrm{LSD}^{(index)} > T$ (a predefined threshold), the current LPC coefficients are transmitted to the decoder, otherwise, the corresponding LPC coefficients are used and only the *index* to the codebook needs to be transmitted to the decoder. A classification bit is added to notify the decoder if the backward linear prediction is applied.

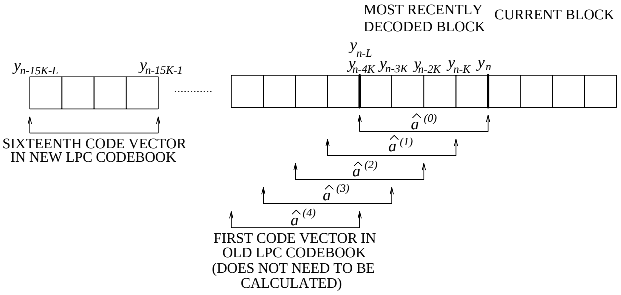

Figure 1: Adaptive forward-backward LPC codebook update scheme.

The AFBQ slightly increases the computational complexity at both the encoder and decoder. At the encoder, for every new block (i) the $M/K = 4$ oldest code vectors of the adaptive forward-backward LPC codebook need to be updated as illustrated in Fig. 1 and (ii) the optimal code vector from the adaptive forward-backward LPC codebook needs to be selected. By using the COSH measure [4] as an upper bound of the LSD measure, the search for optimum can be made computationally efficient. At the decoder, if backward linear prediction is applied, the LPC coefficients are to be determined based on the previously decoded speech samples, i.e., $y_{n-indexK-M}, \ldots, y_{n-indexK-1}$.

3. VECTOR QUANTIZATION

AFBQ can be implemented as either scalar (AFBSQ) or vector quantization (AFBVQ). What follows is the two strategies in the design of the vector quantizer. First, we use split vector quantization (VQ) [5] to achieve a good trade-off between computational complexity and performance. Second, a very efficient codebook initialization method termed *mixture Gaussian clustering* (MGC) originating from [1] is developed to improve the performance of VQ codebook. In MGC, the structure of the training data set containing N training vectors is modeled as a mixture Gaussian density of size $N^{(final)}$. The desired mixture density is progressively estimated through a paired merging process. The MGC algorithm starts with N clusters (each vector in the training set is treated as a separate cluster) and sequentially merges two clusters into one cluster by minimizing a distance measure until the desired number of clusters $N^{(final)}$ is reached.

The distance measure is derived based on the following reasoning: Since the merging process always increases the within-cluster dispersion, two clusters shall be merged only if the increase of within-cluster dispersion is kept at a minimum. Since the total dispersion of training data is equal to the sum of within-cluster dispersion and between-cluster distance, the above merging rule maximizes the decrease of between-cluster distance too. Suppose that a pair of clusters j and k containing L_j and L_k training vectors, respectively, are merged. Then the increase of within-cluster dispersion can be derived [1] as

$$\Delta T(j,k) = \frac{L_j L_k}{(L_j + L_k)L} ||\mu_j - \mu_k||^2, \qquad (1)$$

where μ_j and μ_k represent the mean vector of clusters j and k, respectively, and L denotes the size of the training set. The above equation defines the distance measure D_{MGC} which is used for selecting the pairs in the merging process.

To make the merging algorithm computationally more efficient, we apply the pruning algorithm [6] first to reduce the initial number of clusters from N to $N^{(1)}$ prior to MGC.

4. PERFORMANCE EVALUATION

In our research, we integrate AFBVQ with the FS1016 Federal Standard CELP coder [3] resulting in a high performance variable bit rate speech coder called AFBVQ-CELP, which brings in two-fold bit rate reduction by backward LPC indexing and by forward LPC VQ. Both segmental signal-to-noise ratio (segSNR) and logarithmical spectral distortion (LSD) are used to evaluate the performance of the proposed coder. The test data contains 600 seconds of speech spoken by both male and female speakers. For VQ codebook design, the training data set is different from the above testing data set and contains 1440 seconds of speech signal spoken by a total of 96 male and female speakers. The speech database used in the experiments was obtained from CSLU [7].

Fig. 2 shows the LSD as the function of LPC bit rate with different adaptive forward-backward codebook size S when scalar quantization is used (AFBSQ). The size of the adaptive forward-backward LPC codebook may vary from $S = 1$ to $S = 128$ requiring 0 bit or 7 bits to specify the time delay or, just the same, for indexing. As a trade-off between computational complexity and LPC bit rate we chose $S = 16$ in the following experiments.

Fig. 3 further compares the performance of AFBSQ, AFBVQ, and split VQ (using exclusively forward linear prediction) [5]. As seen, AFBVQ reduces the bit rate spent on transmission of LPC coefficients by 5–18 bit-per-frame (bpf) compared to AFBVQ. In other words, at a given bit rate AFBVQ decreases LSD by 0.72–0.85 dB. When compared to split VQ, AFBVQ reduces the LSD by 0.2–0.3 dB at the same bit rate, or equivalently decreases the LPC bit rate by 3–4 bpf having the same spectral distortion.

Now, we compare the proposed pruning+MGC codebook initialization with random initialization. In random initialization [6], code vectors from the training set are randomly selected to populate the initial codebook. In Fig. 4, LSD is plotted as the function of the average LPC bit rate when pruning+MGC and random initialization are used. Pruning+MGC initialization outperforms random initialization by an average LPC bit rate reduction of 2 bpf. Furthermore pruning+MGC initialization provides a smaller initial distortion than random initialization, resulting in a faster convergence of the LBG training algorithm. There were 22 or 45 iterations needed for the pruning+MGC or random initialization, respectively, when the same convergence criterion was adopted and the yielded final distortion was even smaller with the pruning+MGC initialization.

AFBVQ-CELP can be viewed as an effort to enhance VQ-CELP by including AFBVQ. In terms of LSD, AFBVQ-CELP consistently outperforms VQ-CELP by 1–2 bpf (see Fig. 5). As shown in Fig. 6, in terms of segSNR AFBVQ-CELP is superior to VQ-CELP by 0.3 dB at low bit rates. The LPC bit rate of AFBVQ-CELP is reduced by up to 6 bpf while having the same segSNR as VQ-CELP. Similar to AFBSQ-CELP, AFBVQ-CELP represents a flexible variable-bit-rate coder. As opposed to VQ-CELP where different bit rates can only be derived from VQ codebooks of different sizes, AFBVQ-CELP facilitates the bit rate control simply by using the threshold T.

Tables 1 and 2 summarize the objective performance of AFBSQ-CELP and AFBVQ-CELP, respectively, using the same adaptive forward-backward LPC codebook size $S = 16$. By varying threshold $T = 3.0$ to $T = 6.0$ dB, the performances of the two variable rate coders are evaluated by segSNR and LSD.

Subjective performance evaluation is based on eight sentences chosen from the TIMIT database. The results show that in terms of decoded speech quality AFBVQ-CELP is statistically indistinguishable from VQ-CELP. So the bit rate spent on transmission of

LPC coefficients is reduced by a factor of 4 (from 34 bpf of FS1016 CELP to 8.4 bpf of AFBVQ-CELP) without compromising the decoded speech quality. This means that the overall bit rate of the coder is reduced from 4.8 kbps to 3.94 kbps. So in AFBVQ-CELP only 6% of the overall bit budget is spent on transmission of predictor coefficients compared 23% of the FS1016 CELP coder.

We finish the section with a few more words about long history quantization (LHQ) [2]. As far as we know, LHQ has never been integrated with CELP by the authors. We did it earlier and found AFBQ outperformed LHQ by 1 bpf at the same spectral distortion [8]. By AFBQ, the order of linear prediction can be chosen even higher when backward linear prediction is applied. In doing so, we found segSNR could be further increased. For instance, when the 12th order backward LPC was used, segSNR was raised by 0.2 dB.

T [dB]	LPC Rate [bpf]	Overall Rate [bps]	segSNR [dB]	LSD [dB]
0	34.0	4800	10.85	1.53
3.0	24.8	4493	10.28	1.81
3.5	21.7	4391	10.16	2.01
4.0	18.8	4290	10.12	2.25
4.5	16.3	4210	10.05	2.52
5.0	14.1	4137	10.01	2.80
5.5	12.3	4076	9.94	3.07
6.0	10.8	4027	9.86	3.33

Table 1: Objective performance of AFBSQ-CELP with $M = 240$, $S = 16$.

T [dB]	LPC Rate [bpf]	Overall Rate [bps]	segSNR [dB]	LSD [dB]
0	24.0	4467	10.59	1.03
3.0	17.7	4257	10.31	1.45
3.5	15.6	4187	10.27	1.69
4.0	13.6	4120	10.15	1.98
4.5	12.0	4067	10.12	2.25
5.0	10.6	4020	10.04	2.57
5.5	9.4	3980	10.02	2.86
6.0	8.4	3947	9.95	3.14

Table 2: Objective performance of AFBVQ-CELP with $M = 240$, $S = 16$, $N^{(final)} = 2^{12}$, using pruning+MGC initialization.

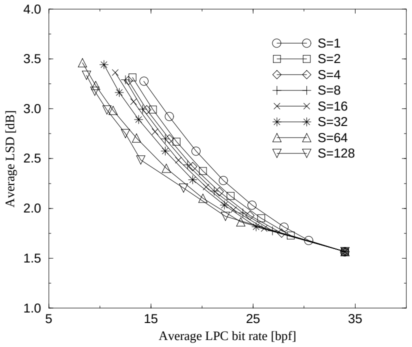

Figure 2: Comparison of AFBSQ-CELP with different S in terms of LSD.

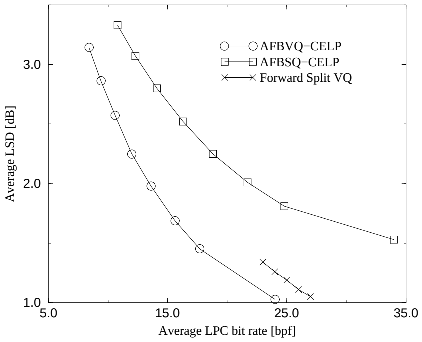

Figure 3: Comparison of AFBSQ, AFBVQ, and split VQ [5] in terms of LSD.

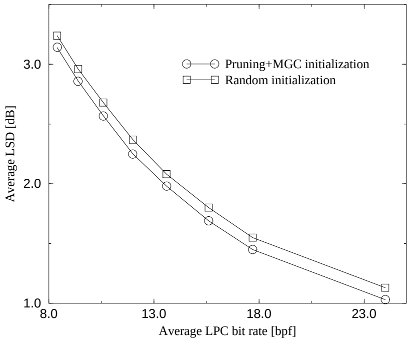

Figure 4: Comparison of random and pruning+MGC initialization in terms of LSD.

5. CONCLUSIONS

In this paper, AFBVQ was integrated into the FS1016 CELP coder. Naturally, AFBQ scheme can also be applied to other speech coding algorithms for which exclusively forward linear prediction is used. As mentioned above, the bit rate in AFBQ can be easily controlled by deciding the between-block similarity in terms of the threshold T. This would further provide a valuable feature with most cellular mobile applications.

6. ACKNOWLEDGMENTS

The authors would like to thank the Center for Spoken Language Understanding at Oregon Graduate Institute of Science and Technology for releasing the speech database which was used in computer experiments.

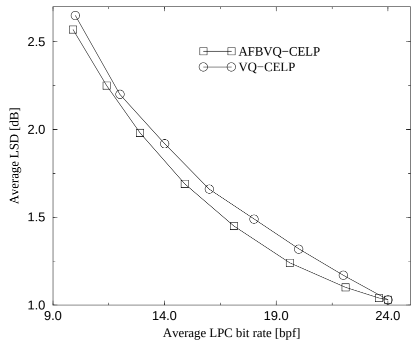

Figure 5: Comparison of AFBVQ-CELP and VQ-CELP in terms of LSD.

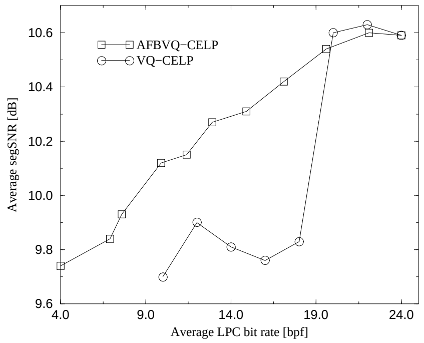

Figure 6: Comparison of AFBVQ-CELP and VQ-CELP in terms of segSNR.

7. REFERENCES

[1] Y. Zhao. A speaker-independent continuous speech recognition system using continuous mixture density HMM of phoneme-sized units. *IEEE Transactions on Speech and Audio Processing*, 1(3):345–361, 1993.

[2] C.S. Xydeas and K.K.M. So. A long history quantization approach to scalar and vector quantization of LSP coefficients. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 1–4, 1993.

[3] J.P. Campbell, V.C. Welch and T.E. Tremain. An expandable error-protected 4800 BPS CELP coder (U.S. Federal Standard 4800 BPS voice coder). In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 735–738, 1989.

[4] A.H. Gray and J.D. Markel. Distance measures for speech processing. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 24:380–391, October 1976.

[5] K.K. Paliwal and B.S. Atal. Efficient vector quantization of LPC parameters at 24 bits/frame. *IEEE Transactions on Speech and Audio Processing*, 1(1):3–14, January 1993.

[6] A. Gersho and R.M. Gray. *Vector Quantization and Signal Compression*. Kluwer Academic Publishers, Norwell, MA, 1992.

[7] R.C. Cole, M. Fanty, M. Noel and T. Lander. Telephone speech corpus development at CSLU. In *Proceedings of IEEE International Conference on Spoken Language Processing*, pages 1–3, September 1994.

[8] J. Vass, Y. Zhao, and X. Zhuang. Adaptive forward-backward quantizer for low bit rate high quality speech coding. *IEEE Transactions on Speech and Audio Processing, to appear*, 1997.