S4.6

Low-Complexity Multi-Mode VXC Using Multi-Stage Optimization and Mode Selection

Mark Johnson* and Tomohiko Taniguchi

Speech Signal Processing, Fujitsu Laboratories Ltd.
1015 Kamikodanaka, Nakahara-ku, Kawasaki 211, Japan

ABSTRACT

Multi-mode coding, proposed in an earlier paper, has been shown to produce good quality speech at low bit rates, at the cost of a computational complexity which is more than double that of normal VXC. This paper will propose an algorithm in which multi-mode coding is a transparent adaptation of multi-stage VXC, requiring almost zero additional computation. In the process, we will show how the cost of jointly optimized multi-stage VXC can be minimized using a backward-transform implementation of the Perceptually Orthogonal VXC algorithm previously proposed by the authors.

1. INTRODUCTION

In the previous work of one of the authors [1], it was shown that a multi-mode bit-assignment algorithm could be used to dramatically improve the quality of standard VXC coding, but at a rather significant increase in computational complexity. This paper will describe several stages in the development of multi-stage multi-mode perceptually-orthogonal VXC, a low-complexity multi-mode coding algorithm.

Section 2 will describe the perceptually-orthogonal VXC algorithm, in which the codebooks in each stage of a multi-stage VXC coder are adaptively transformed in such a way that the perceptually weighted excitation vectors are orthogonal to each other. This orthogonalization allows the joint optimization of M different vector gains without the solution of an $O[M^2]$ matrix equation. Section 3 shows how POVXC can be efficiently implemented for some codebooks using backward-transformation of the input and filter matrix.

Section 4 will show how a multi-mode coder can be designed to trade transmitted bits between the LPC filter and a third or fourth vector quantization stage in a multi-stage POVXC coder. This multi-stage multi-mode coder will be shown to have a complexity similar to that of a normal multi-stage coder, and much less than that of a normal multi-mode coder. Section 5 will show that the performance of such a coder, while not as good as that of a normal multi-mode coder, is noticeably better than that of normal POVXC.

2. PERCEPTUALLY-ORTHOGONAL VXC

Vector Excitation Coding (VXC) is the class of analysis-by-synthesis vocoders which are excited by a sum of gain-shape quantization vectors, chosen, using a perceptually weighted MSE criterion, from one or more adaptive or pre-determined vector codebooks [2]. All VXC coders pick their excitation vectors one at a time, since searching for a globally optimum vector set would be computationally prohibitive. A typical coder will choose each vector so that the weighted sum up to and including that vector is an MSE-optimal estimation of the input speech. This is usually done either by re-optimizing all of the vector gains as each new vector is chosen, or by orthogonalizing the weighted vectors, as shown in Fig. 1. Except in special circumstances [3], the computational complexity of both of these approaches grows with the square of the number of excitation vectors.

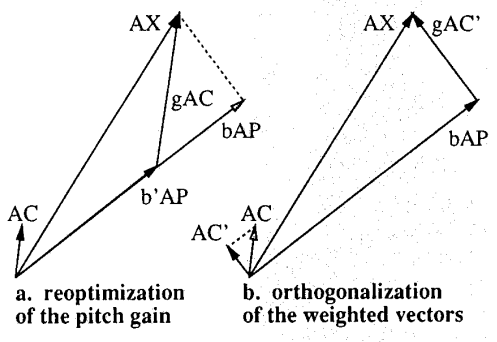

Fig. 1: Multi-vector joint-MSE minimization

We showed in a previous article [4] that, if the perceptual weighting matrix is known, it is possible to adaptively transform each unweighted VXC codebook in such a way that, without any further modification, the weighted codebook will be orthogonal to the weighted excitation vectors chosen at each previous stage (Fig. 2). As applied to standard 2-vector CELP, this technique was called Pitch-Orthogonal LPC. In the current article, we will refer to the more general multi-stage technique as Perceptually-Orthogonal VXC.

Fig. 3 shows the structure of a perceptually-orthogonal VXC coder. As shown, the coder first prepares the input, using LPC analysis (A(z)), perceptual weighting (1/A'(z)), and finally, subtraction of the ringing left over from perceptual weighting of the previous frame. The weighted input is then used as the target vector for a number of vector quantizers. Each of these

*currently studying in the Department of Electrical Engineering and Computer Science, MIT, Cambridge, Massachusetts, USA

CH2977-7/91/0000-0221 $1.00 © 1991 IEEE

quantizers first transforms its codebook with a cascade of perceptual orthogonalizations, H, and then searches the weighted, transformed codebook for an MSE-optimal quantization of the input.

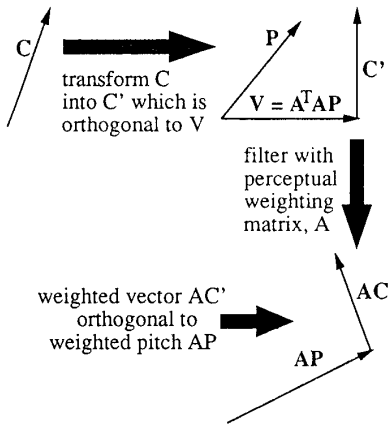

Fig. 2: Perceptually Orthogonal VXC
$(AC')^T(AP) = 0$ if and only if $C'^TV = 0$

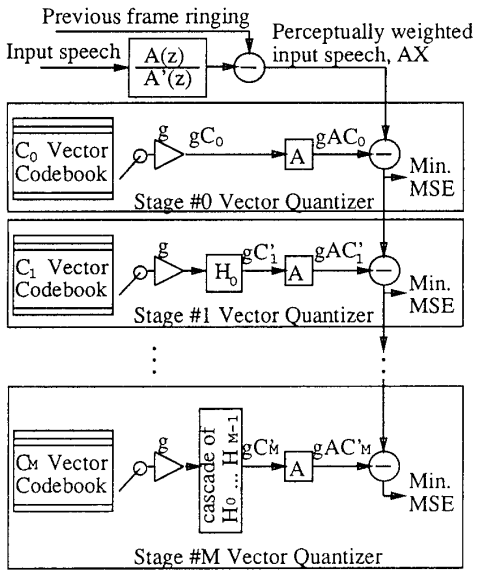

Fig. 3: Perceptually Orthogonal VXC
A = perceptual weighting matrix
Hn orthogonalizes Cm to Vn, n<m

As discussed in [4], the perceptual orthogonalization transform, H, can be one of several possibilities. For example, a POVXC coder using the weighting-filter-space Gram-Schmidt transform, introduced in [4], will produce synthesized speech identical to that produced by the standard re-optimization algorithm shown in Fig. 1. POVXC using the householder transform, on the other hand, will transform the unweighted codevectors without changing their magnitudes, so that, for example, the unity variance search criterion proposed in [5] can be applied to POVXC as easily as to standard CELP.

The vector codebook used in each quantizer is also variable. In a CELP-type coder, the first of these quantizers uses an adaptive codebook made to represent the pitch, and the remaining quantizers usually use fixed stochastic codebooks made to model the noise component of the input. Of course, depending on the type of codebook used at each stage, there are far more efficient ways to implement POVXC than the brute force approach depicted in Fig. 3. Section 3 will describe one of these techniques.

3. BACKWARD TRANSFORMATION

The pitch residual in a normal CELP-type coder has often been noticed to resemble an impulsive noise vector, rather than a dense, Gaussian noise vector, and Kleijn, Krasinski, and Ketchum [6] reported getting their best coding performance using a stochastic codebook with only ten percent non-zero samples. Because of this, one of the most popular computational reduction techniques in VXC is the use of sparse [7], algebraic [8], or sparse-delta [5] codebooks, with a codebook search algorithm which has been modified to take advantage of the codebook structure.

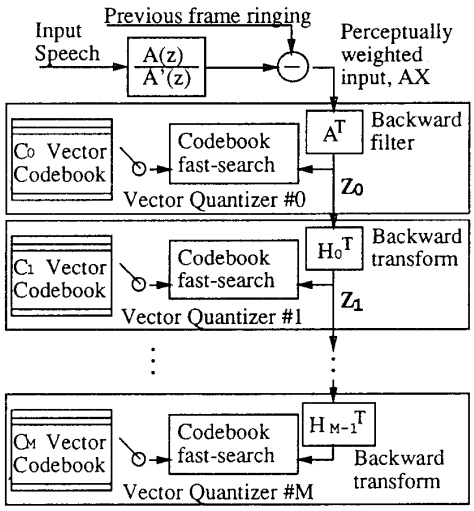

Fig. 4: POVXC with Backward Transformation

If we write each orthogonalization transform as some matrix H_m, as shown in Fig. 3, then POVXC can be similarly simplified. First, according to a standard simplification, we can minimize the weighted quantization error at each stage of the coder by maximizing the function $F(\mathbf{C}_m)$, where

$$F(\mathbf{C}_m) = \frac{(AH_{m-1}...H_0\mathbf{C}_m)^T(A\mathbf{X})}{|AH_{m-1}...H_0\mathbf{C}_m|^2} \quad (1)$$

$$= \mathbf{C}_m^T\mathbf{Z}_m / \mathbf{C}_m^T G_m \mathbf{C}_m \quad (2)$$

For commutative H_m, we can simplify the computation of equation (2) by recursively calculating $\mathbf{Z}_m$ and G_m using

"backward transformation," that is, by recursively multiplying some $\mathbf{Z}_0$ and G_0 by the transpose of each H_{m-1} matrix:

$$\mathbf{Z}_m = H_{m-1}^T \mathbf{Z}_{m-1}, \quad \mathbf{Z}_0 = A^T A\mathbf{X}, \tag{3}$$

$$G_m = H_{m-1}^T G_{m-1} H_{m-1}, \quad G_0 = A^T A. \tag{4}$$

Fig. 4 shows an implementation of POVXC using backward transformation. Since backward transformation can be done recursively, instead of building up into a series like the forward transforms in equation 1, the computational complexity of the algorithm increases only linearly with the number of excitation vectors, instead of increasing as the square of the number of vectors, as most algorithms do. Furthermore, if the untransformed codebook $\mathbf{C}_m$ is sparse, ternary-valued, or sparse-delta in construction, then equation 2 can be implemented using the appropriate codebook fast-search algorithm, often greatly reducing the complexity of each codebook search.

4. MULTI-STAGE MULTI-MODE POVXC

If multi-stage POVXC can be efficiently implemented, then we can find a variety of contexts in which it can be applied. One particularly promising context is that of multi-mode VXC.

Multi-mode VXC, as previously proposed by one of the authors [1], is an adaptive bit allocation scheme in which several VXC coders with different bit allocations are run in parallel, and the coder with the lowest quantization error for each frame is chosen. The speech model parameters are then quantized according to the bit allocation scheme used by the chosen coder, and transmitted to the decoder, along with a coder index, or mode number, telling the decoder how to interpret the quantized parameters. In the simulation results presented in section 5, we will demonstrate a simple two-mode coder operating at 4 Kb/s which provides 1.4dB of segmental SNR improvement over conventional single-mode VXC.

Multi-stage multi-mode POVXC will be an attempt to achieve similar quality improvements without paying the computational cost of running more than one VXC coder in-parallel. We can do this by building up from each coder to the next, so that each coder is a specific extension of the lower-order coders, rather than an independent VXC coder in its own right. In Fig. 5, for example, the lowest-order coder only uses one excitation codebook, but has extremely accurate quantization of the LPC coefficients. The higher-order coders, then, each re-quantize the LPC coefficients with fewer bits, and use the extra bits as the index into an additional excitation codebook. The last coder uses a maximum number of excitation codebooks, and transmits no information about the LPC coefficients, which are assigned to their values from the previous frame.

Fig. 6 represents a slightly more pragmatic implementation of a two-mode coder. In this coder, the gradual trade-off of LPC information for excitation information is replaced by an all-or-nothing choice between an A-mode coder, which transmits the new LPC coefficients, and a B-mode coder, which uses a third excitation codebook. The first two codebooks, one a pitch codebook, and one a pseudo-random noise codebook, are first searched by the A-mode coder to find the minimum-MSE choice of excitation vectors from each of these codebooks. The selected codebook indices are then passed to the B-mode coder, which weights each vector using the previous-frame perceptual weighting matrix, re-calculates each vector gain, and finally searches the third codebook for an additional excitation vector. The total error of both coders is then calculated and compared, and the coder with the minimum error is selected for transmission.

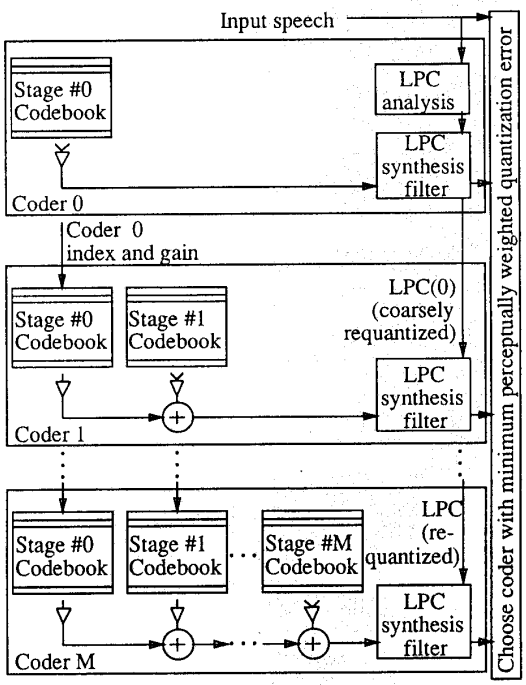

Fig. 5: Multi-Stage Multi-Mode Speech Coding

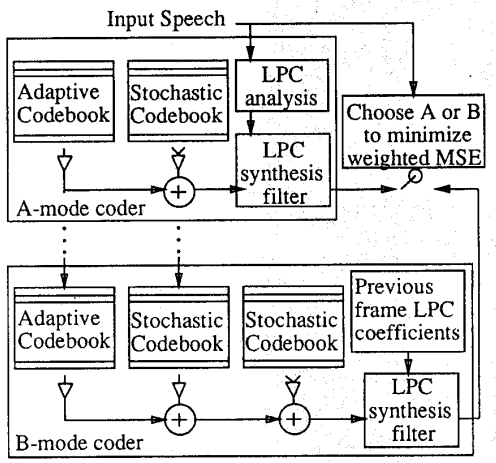

Fig. 6: Two-Mode VXC Speech Coder

5. SIMULATION RESULTS

The coder in Fig. 6 was simulated in software at 4 Kb/s, using the "A coder" and "B coder" bit allocations listed in Table 1, and using **10% sparse stochastic codebooks** [7] for the second and third stage of residual quantization. Table 2 shows the results of the simulation, and of *a priori* estimation of the algorithm's computational complexity. For comparison, the same results are given for single-mode 2- and 3-vector POVXC, and for a standard two-stage multi-mode coder using bit allocations A and C from Table 1.

	Coder A		Coder B		Coder C		3-Stage	
	bit/ms	kb/s	bit/ms	kb/s	bit/ms	kb/s	bit/ms	kb/s
LPC taps	36/ 27	1.3					36/ 27	1.3
Pitch delay	7/ 9	0.8	7/ 9	0.8	7/6.75	1.0	7/ 9	0.8
Pitch gain	4/ 9	0.4	4/ 9	0.4	5/6.75	0.7	4/ 9	0.4
VQ index #1	9/ 9	1.0	9/ 9	1.0	10/6.75	1.5	9/ 9	1.0
VQ gain #1	4/ 9	0.4	4/ 9	0.4	5/6.75	0.7	4/ 9	0.4
VQ index #2			8/ 9	0.9			8/ 9	0.9
VQ gain #2			4/ 9	0.4			4/ 9	0.4
Total	4.0 Kb/s		4.0 Kb/s		4.0 Kb/s		5.3 Kb/s	

Table 1: Bit allocations of simulated coders (2-vector 1-mode coder uses A-mode allocation)

	multistage multimode	2-vector multimode	2-vector 1-mode	3-vector 1-mode
Segmental SNR	12.3 dB	12.7 dB	11.3 dB	13.6 dB
Computation (est. Mflops)	11.5 Mflps	18.3 Mflps	7.9 Mflps	10.5 Mflps
Time spent in B-mode coder	60 %	65 %		

Table 2: Complexity and SNR of simulated coders

As shown, multi-stage multi-mode POVXC gives only about a 1dB improvement over standard POVXC, which is not as high as the 1.4dB obtained for standard multi-mode. This drop in quality is most likely a result of the strong inter-dependence of the two coding modes in a multi-stage multi-mode coder. Specifically, if the first two excitation vectors are chosen by one of the coding modes, then, because of the differing LPC coefficients, they will often not be minimum-MSE excitation vectors for the other coder, and the signal to noise ratio will drop.

Because of this inter-dependence, however, the computational complexity of 3-vector multi-mode POVXC is little more than that of a 3-vector single-mode coder, and is significantly less than that of standard 2-vector multi-mode.

6. CONCLUSION

We have presented the multi-stage multi-mode POVXC algorithm, a computationally tractable method for dynamic bit allocation in low-bit-rate VXC. Multi-mode coding involves a choice between several coding modes, each of which represents a particular trade-off between accurate LPC quantization and accurate quantization of the excitation waveform. In order to make this choice computationally tractable, we have made each mode into a multi-stage coder, so that most of the excitation vectors used by each coder can be copied directly from the choice made by a lower-order coder. This multi-stage vector selection is further simplified by the use of backward transformation POVXC, in which the excitation vectors are guaranteed to be jointly optimum by appropriately transforming the input and the weighting matrix before searching each new excitation codebook.

One of our ambitions for future research is the development of a multi-mode bit allocation scheme which has a closer match to our phonetic understanding of speech. The LPC coefficients, for example, seem to be most predictable during steady-state vowels, and least predictable during transitions. The adaptive codebook, on the other hand, seems to be most necessary during vowels, while the LPC excitation during transitions is often an unpredictable impulse stream. It seems that we could get better speech quality by considering these observations in our multi-mode design, perhaps by trading between LPC coefficient information and an accurate adaptive codebook.

ACKNOWLEDGMENTS

The authors would like to thank Prof. Allen Gersho of the University of California, and his students, for their personal kindness, and for several years of professional inspiration.

We would also like to thank F. Amano and K. Murano of Fujitsu Labs Ltd., for their consistent encouragement and helpful comments.

REFERENCES

[1] T. Taniguchi, Y. Tanaka, and R. M. Gray, "Speech Coding with Dynamic Bit Allocation (Multi-Mode Coding)," *Advances in Speech Coding*; Kluwer Academic Publishers, Norwell, MA, 1990, ed. Atal, Cuperman, and Gersho.

[2] G. Davidson and A. Gersho, "Multiple-Stage Vector Excitation Coding of Waveforms," *Pro. ICASSP*, pp. 1040-1046: April 1988.

[3] P. Dymarski, N. Moreau, and A. Vigier, "Optimal and Sub-Optimal Algorithms for Selecting the Excitation in Linear Predictive Coders," *Proc. ICASSP*, pp. 485-488: April 1990.

[4] M. Johnson and T. Taniguchi, "Pitch-Orthogonal Code-Excited LPC," *Proc. GLOBECOM*, pp. 542-546: Dec. 1990.

[5] T. Taniguchi, M. Johnson, and Y. Ohta, "Pitch Sharpening for Perceptually Improved CELP, and the Sparse-Delta Codebook for Reduced Computation," *Proc. ICASSP*, to appear: May 1991.

[6] W. Kleijn, D. Krasinski, and R. Ketchum, "Improved Speech Quality and Efficient Vector Quantization in SELP," *Proc. ICASSP*, pp. 155-158: April 1988.

[7] G. Davidson and A. Gersho, "Real-Time Vector Excitation Coding of Speech at 4800 bps," *Pro. ICASSP*, pp. 2189-2192: April 1987.

[8] J-P. Adoul, P. Mabilleau, M. Delprat, and S. Morisette, "Fast CELP Coding Based on Algebraic Codes," *Proc. ICASSP*, pp. 1957-1960: April 1987.