Multi Switched Split Vector Quantization of Narrowband Speech Signals

M. Satya Sai Ram, P. Siddaiah, and M. Madhavi Latha

***Abstract*—**Vector quantization is a powerful tool for speech coding applications. This paper deals with LPC Coding of speech signals which uses a new technique called Multi Switched Split Vector Quantization (MSSVQ), which is a hybrid of Multi, switched, split vector quantization techniques. The spectral distortion performance, computational complexity, and memory requirements of MSSVQ are compared to split vector quantization (SVQ), multi stage vector quantization(MSVQ) and switched split vector quantization (SSVQ) techniques. It has been proved from results that MSSVQ has better spectral distortion performance, lower computational complexity and lower memory requirements when compared to all the above mentioned product code vector quantization techniques. Computational complexity is measured in floating point operations (flops), and memory requirements is measured in (floats).

***Keywords*—**Linear predictive Coding, Multi stage vector quantization, Switched Split vector quantization, Split vector quantization, Line Spectral Frequencies (LSF).

I. Introduction

CODING of speech signals is a major impairment in today's low bit rate tele communication systems. The aim of the quantizers used at the transmission end is to compress the speech signal by reducing the bit rate. Coding of speech signals is a challenging task and has been the focus of intense research effort. This paper deals with Linear predictive coding [1]-[2] (LPC) of narrow band speech signals which uses a novel vector quantization [3] scheme called Multi Switched split vector quantization, that is a hybrid of Multi stage vector quantization [4], Switched Split vector quantization [4], split vector quantization [4]-[5]. In MSSVQ, vector quantizers are formed as a cascade, where the difference between the input vector and quantized vector of one stage is fed as an input to the next successive stages. At each stage the quantized vector is obtained by switching from one codebook to the other connected in parellely.

MSSVQ algorithm mainly consists of the following steps a) Selection of a switch b) Extracting the codebook from the trained vectors c) obtaining the quantized vector from a set of codewords d) Extracting the new trained sequence from the 'old and quantized' training sequence e) Repeat steps b to d for the required number of stages.

The aim of this article is to provide a general review of MSSVQ, and to compare its performance with other product code vector quantization schemes. The practical limitations, regarding computational complexity and memory requirements as a function of bit rate are discussed. The spectral distortion performance[6] of MSSVQ is evaluated in LSF parameter quantization [7]-[9] for narrow band speech coding. The performance is evaluated by using the spectral distortion method.

II. Multi Switched Split Vector Quantization

The basic idea of MSSVQ is to use n stages, m switches and s splits, so as to improve the performance of quantization by decreasing the computational complexity and memory requirements when compared to SVQ, MSVQ, and SSVQ. The use of switch vector quantizer exploits the correlation that exists across all dimensions of the vector space. In each SVQ the 10-dimmensional LSF vector is split into 3 parts of 3, 3, 4 divisions respectively. During codebook generation bits are allocated depending on the frequency of the LSFs. Preference is given to high frequency LSFs, when the number of bits is not divisible by 3. For a particular switch the generation of codebooks at different stages is shown in Fig. 1.

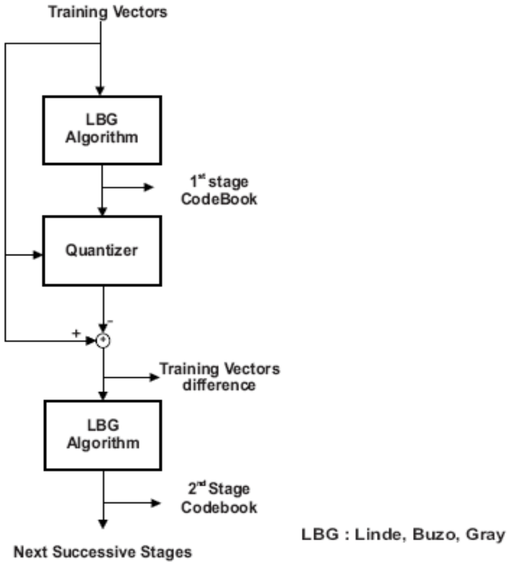


Fig. 1 Codebook Generation at different stages

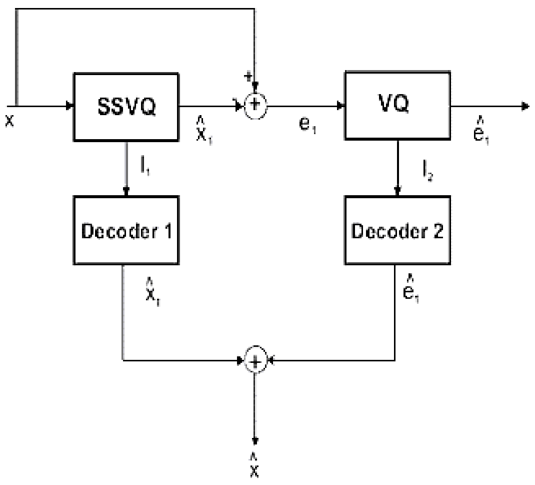
M. Satya Sai Ram is with Department of ECE, R.V.R& J.C College of Engineering, Guntur-522019, A.P, India (e-mail: m_satyasairam@yahoo.co.in).

P. Siddaiah is with Department of ECE, K. L. College of Engineering, Guntur-522502, A.P, India (e-mail: siddaiah_p@yahoo.com).

M. Madhavi Latha is with Department of ECE, J.N.T.U College of Engineering, Hyderabad-500072, A.P, India (e-mail: mlmakkena@yahoo.com).

- Initially the codebook at the first stage is generated by using the Linde, Buzo and Gray (LBG) [10] algorithm with the training set as an input.
- Secondly the training difference vectors are extracted by applying the training set and the codebook of the first stage to the quantizer.
- Finally the training difference vectors are used to generate the codebook of the second stage.

The above procedure is used for the required number of stages.



($I_i$denotes the Index of I^th quantizer)

SSVQ : Switched Split Vector Quantization

Fig. 2 Block Diagram of MSSVQ

An n x m x s MSSVQ is shown in Fig. 2, where n corresponds to the number of stages, m corresponds to the number of switches, and s corresponds to the number of splits. Each input vector x that is to be quantized is applied to SSVQ of the first stage so as to obtain an approximate vector $\hat{x}_1$=Q[x]. The error vector, $e_1 = x_1 - \hat{x}_1$, is calculated and is quantized by using SSVQ at the second stage so as to obtain the quantized version of the error vector $\hat{e}_1 = Q[e_1]$. This process can be continued for the required number of stages. Finally the decoder takes the indices, I_i, from each stage and adds the quantized vectors at each stage so as to obtain the reconstructed vector $\hat{x}$.

III. Complexity and Memory Requirements

The computational complexity and memory requirements of MSSVQ are given by:

$$\text{Complexity}_{MSSVQ} = n(4d2^{b_m}-1) + \sum_{j=1}^{n}[\sum_{i=1}^{s}(4d_i 2^{b_{ij}}-1)]$$

$$\text{Memory}_{MSSVQ} = nd2^{b_m} + 2^{b_m}\sum_{j=1}^{n}[\sum_{i=1}^{s} d_i 2^{b_{ij}}]$$

d is the dimension of the vector, b_m is the number of bits allocated to the switch vector quantizer, m = 2^{b_m} is the number of switching directions, n is the number of stages, s is the number of splits

IV. Results

Tables I, II, III and IV shows the spectral distortion (dB), computational complexity (kflops/frame), and memory requirements (ROM) at various bit rates for a three part split vector quantizer, three stage split multistage vector quantizer, two part switched split vector quantizer, three stage multi switched split vector quantizer. From Tables I to IV and from Figs. 3&4 it is observed that MSSVQ has better spectral distortion performance, less computational complexity, and memory requirements when compared to SVQ, MSVQ, and SSVQ. From Fig's 5&6 it is observed that for MSSVQ as the number of stages increases the computational complexity, and memory requirements decreases as the number of bits/frame increases. For SVQ transparency in quantization is achieved at 24 bits/frame, for MSVQ transparency is achieved at 22 bits/frame, for SSVQ transparency is achieved at 22 bits/frame and for MSSVQ transparency is achieved at 21 bits/frame. From the results it is proved that MSSVQ has better spectral distortion performance, less computational complexity and memory requirements when compared to all the above mentioned product code vector quantization schemes.

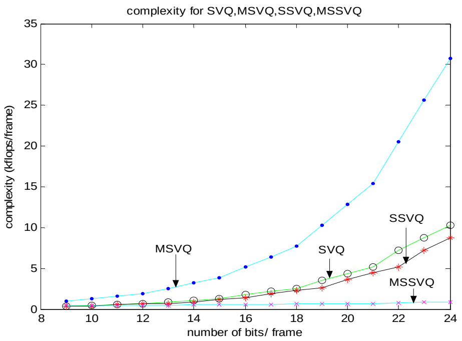


Fig. 3 Complexity for SVQ, MSVQ, SSVQ, and MSSVQ

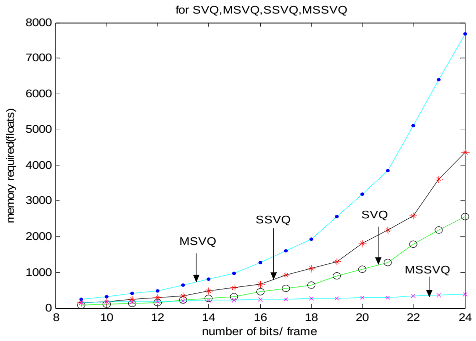


Fig. 4 Memory requirements for SVQ, MSVQ, SSVQ, and MSSVQ

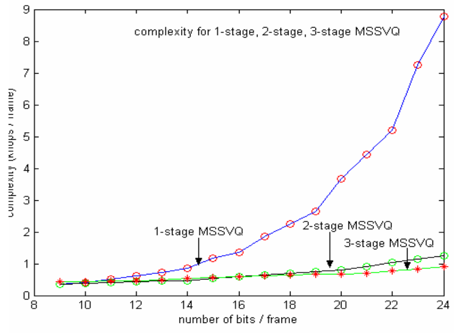


Fig. 5 Complexity for 1-stage, 2 -stage, and 3 -stage MSSVQ

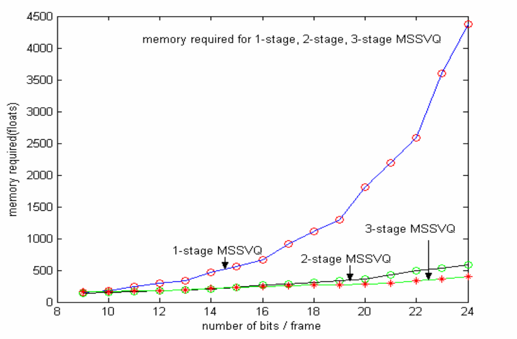


Fig. 6 Memory requirements for 1 -stage, 2 -stage, and 3 -stage MSSVQ

TABLE I
SPECTRAL DISTORTION, COMPLEXITY, AND MEMORY REQUIREMENTS FOR SPLIT VECTOR QUANTIZATION

Bits / frame	SD(dB)	2-4 dB	>4dB	Complexity (kflops/frame)	ROM (floats)
24(8+8+8)	1.45	0.43	0	10.237	2560
23(7+8+8)	1.67	0.94	0	8.701	2176
22(7+7+8)	1.701	0.78	0.1	7.165	1792
21(7+7+7)	1.831	2.46	0.2	5.117	1280

TABLE II
SPECTRAL DISTORTION, COMPLEXITY, AND MEMORY REQUIREMENTS FOR MULTI STAGE VECTOR QUANTIZATION

Bits / frame	SD(dB)	2-4 dB	>4dB	Complexity (kflops/frame)	ROM (floats)
24(8+8+8)	0.984	1.38	0	30.717	7680
23(7+8+8)	1.238	1.2	0.1	25.597	6400
22(7+7+8)	1.345	0.85	0.13	20.477	5120
21(7+7+7)	1.4	1.08	0.3	15.357	3840

TABLE III
SPECTRAL DISTORTION, COMPLEXITY, AND MEMORY REQUIREMENTS FOR SWITCHED SPLIT VECTOR QUANTIZATION

Bits / frame	SD(dB)	2-4 dB	>4dB	Complexity (kflops/frame)	ROM (floats)
24(8+8+8)	0.957	1.06	0	8.78	4372
23(7+8+8)	1.113	1.29	0.14	7.244	3604
22(7+7+8)	1.119	0.52	1.3	5.196	2580
21(7+7+7)	1.127	1.3	0.56	4.428	2196

TABLE IV
SPECTRAL DISTORTION, COMPLEXITY, AND MEMORY REQUIREMENTS FOR MULTI SWITCHED SPLIT VECTOR QUANTIZATION

Bits / frame	SD(dB)	2-4 dB	>4dB	Complexity (kflops/frame)	ROM (floats)
24(8+8+8)	0.0322	0	0	0.9	396
23(7+8+8)	0.0381	0	0	0.836	364
22(7+7+8)	0.0373	0	0	0.772	332
21(7+7+7)	0.0377	0	0	0.708	300

V. CONCLUSION

MSSVQ is having better trade-off between bit rate and spectral distortion performance, computational complexity, and memory requirements, when compared to other product code vector quantization schemes like SVQ, MSVQ, and SSVQ. So MSSVQ is proved to be better. The decrease in the computational complexity is due to the less availability of bits at each stage of quantization as the number of stages increases. From Fig. 4 it is observed that for SSVQ memory required is high when compared to SVQ. But for MSSVQ memory required is less when compared to SVQ, MSVQ, and SSVQ. So MSSVQ is proved to be better when compared to the above product code vector quantization techniques.

REFERENCES

[1] Atal, B.S. The history of linear prediction. IEEE Signal Processing Magazine, Vol 23, pp.154-161, March 2006.

[2] Harma, A. Linear predictive coding with modified filter structures. IEEE Trans. Speech Audio Process, Vol 9, pp.769-777, Nov 2001.

[3] Gray, R.M., Neuhoff, D.L.. Quantization. IEEE Trans. Inform. Theory, pp.2325-2383, 1998.

[4] Stephen, So., & Paliwal, K. K. Efficient product code vector quantization using switched split vector quantiser. Digital Signal Processing journal, Elsevier, Vol 17, pp.138-171, Jan 2007.

[5] Paliwal., K.K, Atal, B.S. Efficient vector quantization of LPC Parameters at 24 bits/frame. IEEE Trans. Speech Audio Process, pp.3-14,1993.

[6] Sara Grassi., “Optimized Implementation of Speech Processing Algorithms,” Electronics and Signal Processing Laboratory, Institute of Micro Technology, University of Neuchatel, Breguet 2, CH-2000 Neuchatel, Switzerland, 1988.

[7] Bastiaan Kleijn., W. Fellow, IEEE, Tom Backstrom., & Paavo Alku. On Line Spectral Frequencies. IEEE Signal Processing Letters, Vol.10, no.3, 2003.

[8] Soong, F., & Juang, B. Line spectrum pair (LSP) and speech data compression. IEEE International Conference on ICASSP, 9, pp 37-40,1984.

[9] P. Kabal and P. Rama Chandran. “The Computation of Line Spectral Frequencies Using Chebyshev polynomials” IEEE Trans. On Acoustics, Speech Signal Processing, vol 34, No.6, pp. 1419-1426, 1986.

[10] Linde, Y., Buzo, A., & Gray, R. M. An Algorithm for Vector Quantizer Design. IEEE Trans.Commun, 28, pp. 84-95, Jan. 1980.