STRUCTURED GAUSSIAN MIXTURE MODEL BASED PRODUCT VQ

Saikat Chatterjee and Mikael Skoglund

Communication Theory Laboratory, School of Electrical Engineering
KTH - Royal Institute of Technology, Stockholm - 10044, Sweden
sach@kth.se, skoglund@ee.kth.se

ABSTRACT

In this paper, the Gaussian mixture model (GMM) based parametric framework is used to design a product vector quantization (PVQ) method that provides rate-distortion (R/D) performance optimality and bitrate scalability. We use a GMM consisting of a large number of Gaussian mixtures and invoke a block isotropic structure on the covariance matrices of the Gaussian mixtures. Using such a structured GMM, we design an optimum and bitrate scalable PVQ, namely an split (SVQ), for each Gaussian mixture. The use of an SVQ allows for a trade-off between complexity and R/D performance that spans the two extreme limits provided by an optimum scalar quantizer and an unconstrained vector quantizer. The efficacy of the new GMM based PVQ (GMPVQ) method is demonstrated for the application of speech spectrum quantization.

1. INTRODUCTION

Conventional design of a product vector quantization (PVQ) method is carried out in a non-parametric framework where the quantizer is designed directly from a training database for a specified bitrate. Generally, a PVQ method does not provide R/D optimality and bitrate scalability. To achieve R/D optimality, an optimum bit allocation strategy can be used [1], but with the requirement of better modeling a source pdf in a parametric framework. A simple strategy of modeling a vector source pdf using a unimodal density (such as a Gaussian or Laplacian pdf) may not lead to a better R/D performance. On the other hand, for the issue of bitrate scalability, we desire to design a quantizer that can operate at any specified bitrate (from a lowest bitrate to a specified highest bitrate) without the requirement of retraining the quantizer or storing multiple codebooks designed for continuum bitrates.

To achieve R/D optimality and bitrate scalability, Subramaniam and Rao [2] proposed a pdf optimized parametric framework where a vector source pdf is modeled using a multi-modal Gaussian mixture (GM) density. The use of a Gaussian mixture model (GMM) is well known in the literature for modeling an arbitrary source pdf quite accurately. For designing a GMM based PVQ (GMPVQ) method with R/D optimality, a basic requirement is to design an optimum PVQ for each Gaussian mixture of the GMM. In [2], Subramaniam and Rao had used a GMM of an unconstrained structure, consisting of a lower number of correlated Gaussian mixtures. Hence, for each correlated Gaussian mixture, they used a transform domain scalar quantizer (TrSQ) [1] as a simple PVQ. For the TrSQ of a correlated Gaussian mixture, the relevant mixture specific KLT was used to de-correlate the source vector and then the transform domain uncorrelated components were quantized using an optimum scalar quantizer (SQ) [1]. To design an optimum SQ, a variance based optimum bit allocation strategy was invoked [1]. In the same framework of using an unconstrained GMM, another GMPVQ method was developed in [3] where the use of a lattice VQ was explored as a PVQ. The lattice based GMPVQ method of [3] was shown to provide better R/D performance than the SQ based GMPVQ method of [2] at higher bit rates, but to suffer at lower bitrates. A lattice VQ suffers at lower bitrates due to the problem of lattice scaling in a support region and hence, the use of a lattice VQ does not guarantee better R/D performance for any bitrate.

To quantize a non-stationary source (for example, speech and image sources, and their parameters), the use of a GMM consisting of a large number of Gaussian mixtures leads to better modeling of the source pdf. Because of the choice of a large number of mixtures, it is a standard practice to choose a diagonal covariance matrix for each Gaussian mixture leading to uncorrelated vector components within each mixture [4]. Using such a structured GMM consisting of a large number of uncorrelated Gaussian mixtures, an optimum and scalable GMPVQ method was developed in [5] where an optimum SQ was designed as a PVQ for each uncorrelated Gaussian mixture. For quantization of non-stationary speech parameters at any bitrate, the structured GMM based GMPVQ method of [5] was shown to provide better R/D performance than the unstructured GMM based GMPVQ method of [2] while retaining bitrate scalability.

Several GMPVQ methods have been developed in recent literature [6]-[9], notably the switched quantization methods. Most of these methods address the issue of R/D optimality, but not the bitrate scalability. Developing a bitrate scalable GMPVQ method is hamstrung due to the prohibitive memory requirement of storing PVQ codebooks designed for all Gaussian mixtures at continuum bitrates. In case of the GMPVQ methods of [2] and [5], the use of SQs (as PVQs) allows for invoking a simple statistical normalization (mean removal and variance normalization) technique without compromising the optimality of R/D performance and inherently helps to solve the memory complexity problem. For these SQ based GMPVQ methods, storing a set of optimum codebooks designed for a zero mean and unit variance (ZMUV) Gaussian scalar source at continuum bits/scalar suffices to address the R/D optimality as well as bitrate scalability; note that the codebook storage requirement of a ZMUV Gaussian scalar source at continuum bits/scalar is minimally intensive.

For the scalable GMPVQ methods of [2] and [5], the use of SQ results in a loss of higher dimensional coding advantage, mainly the space-filling loss [10]. To recover the space-filling loss at any bit rate, we propose to use an optimum product VQ, namely an split VQ (SVQ) [11]. In the new GMPVQ method, an optimum SVQ is designed for each Gaussian mixture. For an optimal SVQ to quantize an un-

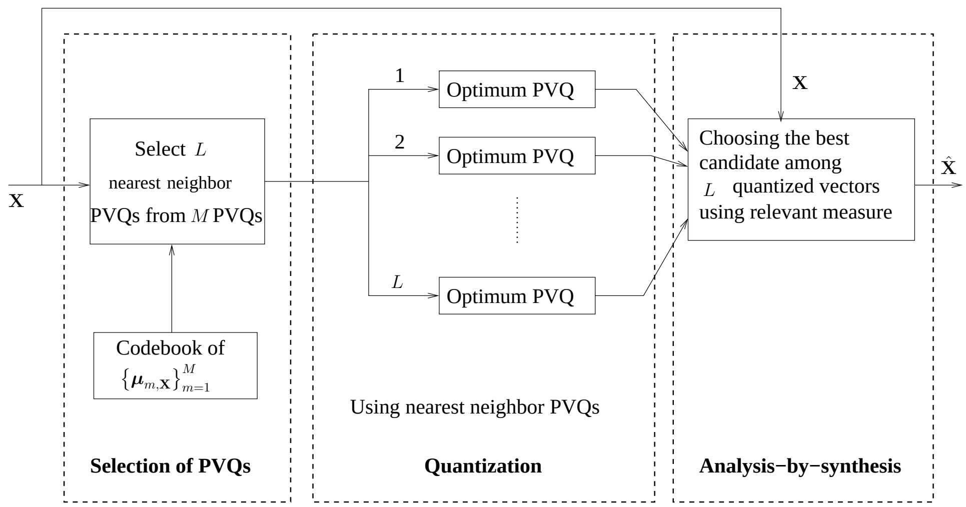

Figure 1: Gaussian mixture model based product vector quantization (GMPVQ) method.

correlated Gaussian vector source, a source vector is split into subvectors and then quantized independently following a subvector-based optimum bit allocation strategy [11]. The R/D performance and complexity trade-off of an optimum SVQ spans the two extreme limits provided by an optimum SQ and an unconstrained VQ. In general, for quantizing a vector source with arbitrary covariance structure, an optimum SVQ does not provide bitrate scalability through the use of a statistical normalization method. To provide bitrate scalability, we use a GMM consisting of a large number of uncorrelated Gaussian mixtures and further impose a structure on the covariance matrices of the Gaussian mixtures. We propose to use a *block isotropic* covariance matrix for each uncorrelated Gaussian mixture. Based on such a structured GMM, we show that it is possible to design and use a bitrate scalable SVQ through the application of a statistical normalization method. For speech spectrum quantization, we show that the new SVQ based GMPVQ method provides better R/D performance than the SQ based GMPVQ methods of [2] and [5] while retaining bitrate scalability.

2. GMM BASED PVQ

In this section, we discuss the general structure of a GMPVQ method. Let $\mathbf{X}$ be the p-dimensional source vector; the pdf of $\mathbf{X}$ is modeled using a GMM of M Gaussian components as

$$f_{\mathbf{X}}(\mathbf{x}) \approx \sum_{m=1}^{M} \alpha_m \, \mathcal{N}(\mathbf{x}; \mu_{m,\mathbf{X}}, \mathbf{C}_{m,\mathbf{X}}), \tag{1}$$

where α_m, $\mu_{m,\mathbf{X}}$ and $\mathbf{C}_{m,\mathbf{X}}$ are the prior probability, mean vector and covariance matrix of the m'th Gaussian mixture $\mathcal{N}(\mathbf{x}; \mu_{m,\mathbf{X}}, \mathbf{C}_{m,\mathbf{X}})$. The approximate equality, used in eq. (1), is because of modeling a source pdf using the GMM with finite number of M mixtures. For the proposed GMPVQ method, we choose a large number of mixtures (i.e. a large M) and invoke a structure on the covariance matrices of the Gaussian mixtures. A GMM consisting of a large number of mixtures has been used in the literature [12] to predict the theoretical high rate performance of a full search VQ. Following a standard practice, we use an expectation-maximization (EM) algorithm for evaluating the GMM parameters.

In the case of an optimum GMPVQ, an optimum PVQ is designed for each Gaussian mixture of the GMM. Therefore, a GMPVQ method comprises a set of M PVQs. A general block diagram of a GMPVQ is shown in Fig. 1. To quantize an input vector, the algorithmic steps are as follows:

1. *Selection of PVQs*: The input vector is compared with the M mean vectors of all Gaussian mixtures using the square Euclidean distance (SED) measure and the SED distance values are rank ordered (sorted); according to the rank ordering, the L number of nearest neighbor (NN) optimum PVQs are chosen from the set of M optimum PVQs for quantization.
2. *Quantization*: Quantize the input vector using the selected L NN optimum PVQs.
3. *Analysis-by-synthesis*: Reconstruct the L quantized vectors. Choose the best quantized vector using an application specific relevant distance measure.

For the GMPVQ method of Subramaniam and Rao [2] and its later extension [3], a lower number of Gaussian mixtures is used to model a GMM. Therefore, the mixtures are highly overlapping in nature and thus, it is required to use all the mixture specific PVQs to choose the best quantized vector through the analysis-by-synthesis (AbS) technique. This approach of using all the quantizers from a set of available quantizers follows a framework of universal quantization (UQ) [13]. As we use a GMM consisting of a large number of Gaussian mixtures, strictly following the UQ framework of employing all the PVQs leads to high computational complexity. Following the work of [5], [9], we use a subset of quantizers from a set of all quantizers. We use a selection stage where an input vector is compared with M mean vectors of Gaussian mixtures using the SED measure (i.e. using nearest neighbor criteria) and a set of L optimum PVQs are chosen from the set of M optimum PVQs using rank ordering (sorting). For a fixed M, an increase of L results in a decrease of quantization distortion. Theoretically, least quantization distortion is achieved if we truly follow the UQ framework, i.e., if we use $L = M$. In practical cases, a decreasing trend of quantization distortion saturates with an increase of L. Therefore, a suitable L can be chosen on the basis of a trade-off between computational complexity and quantization distortion. For an input vector, the use of L

PVQs (practically $L \ll M$) keeps the complexity under check and permits us to choose a much higher value of M for better source density modeling. Also, the increase of M allows for using Gaussian mixtures with diagonal covariance matrices. The use of uncorrelated Gaussian mixtures helps to invoke further structure on the covariance matrices.

3. STRUCTURED GMM BASED PVQ

We design a scalable and optimum GMPVQ method where a scalable and optimum PVQ is designed for each Gaussian mixture of the GMM. Invoking a structure on the covariance matrices of the Gaussian mixtures in the GMM, we design a set of scalable and optimum PVQs which are not storage intensive. To model the source pdf using a GMM, we use a large number of Gaussian mixtures and assume that the mixtures are uncorrelated.

For the PVQs, we propose to use scalable and optimum SVQs. Let us consider to design a scalable and optimum SVQ for the m'th Gaussian mixture. For the m'th Gaussian mixture specific SVQ, an input vector $\mathbf{X}$ is split into S_m number of subvectors ($1 \leq S_m \leq p$) such that $\mathbf{X} = \left[\mathbf{X}_{m,1}^T\ \mathbf{X}_{m,2}^T\ \ldots \mathbf{X}_{m,S_m}^T\right]^T$ and each subvector $\mathbf{X}_{m,i}$ is vector quantized independently. Let, the dimension of $\mathbf{X}_{m,i}$ be $q_{m,i}$ such that $\sum_{i=1}^{S_m} q_{m,i} = p$. It is possible to design an optimum SVQ for each Gaussian mixture through optimum bit allocation among subvectors [11], but difficult to address the issue of bitrate scalability. For a GMM consisting of a large number of uncorrelated Gaussian mixtures, we invoke further constraint on the structure of the GMM to address the issue of designing bitrate scalable SVQs. Using the split vector notation, let us express the diagonal covariance matrix of the m'th Gaussian mixture $\mathbf{C}_{m,\mathbf{X}}$ as:

$$\mathbf{C}_{m,\mathbf{X}} = \begin{bmatrix} \mathbf{C}_{\mathbf{X}_{m,1}\mathbf{X}_{m,1}} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_{\mathbf{X}_{m,2}\mathbf{X}_{m,2}} & \ldots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \ldots & \mathbf{C}_{\mathbf{X}_{m,S_m}\mathbf{X}_{m,S_m}} \end{bmatrix}, \quad (2)$$

where $\mathbf{C}_{\mathbf{X}_{m,i}\mathbf{X}_{m,i}}$ is the covariance matrix of the i'th subvector $\mathbf{X}_{m,i}$; each $\mathbf{C}_{\mathbf{X}_{m,i}\mathbf{X}_{m,i}}$ is a diagonal matrix of dimension $q_{m,i} \times q_{m,i}$. We invoke a constraint such that a sub-vector covariance matrix is isotropic, i.e.

$$\mathbf{C}_{\mathbf{X}_{m,i}\mathbf{X}_{m,i}} = \sigma_{m,i}^2 \mathbf{I}. \quad (3)$$

Using such a block isotropic covariance matrix, we note that the $q_{m,i}$-dimensional subvector $\mathbf{X}_{m,i}$ can be quantized using a codebook designed for a $q_{m,i}$-dimensional subvector consisting of ZMUV Gaussian scalar components. Let us refer to a subvector consisting of ZMUV Gaussian scalar components as a ZMUV subvector. An input subvector $\mathbf{X}_{m,i}$ is mean removed and covariance normalized to produce a ZMUV subvector

$$\mathbf{Y}_{m,i} = \left\{\mathbf{C}_{\mathbf{X}_{m,i}\mathbf{X}_{m,i}}^{-1}\right\}^{1/2} \left[\mathbf{X}_{m,i} - \mu_{\mathbf{X}_{m,i}}\right], \quad (4)$$

where $\mu_{\mathbf{X}_{m,i}}$ is the corresponding subvector part of the mean vector of m'th Gaussian mixture $\mu_{m,\mathbf{X}} = \left[\mu_{\mathbf{X}_{m,1}}^T\ \mu_{\mathbf{X}_{m,2}}^T\ \ldots \mu_{\mathbf{X}_{m,S_m}}^T\right]^T$. The ZMUV subvector $\mathbf{Y}_{m,i}$ is vector quantized as $\hat{\mathbf{Y}}_{m,i}$ and then the quantized subvector $\hat{\mathbf{X}}_{m,i}$ is computed through statistical inverse normalization as

$$\hat{\mathbf{X}}_{m,i} = \left\{\mathbf{C}_{\mathbf{X}_{m,i}\mathbf{X}_{m,i}}\right\}^{1/2} \hat{\mathbf{Y}}_{m,i} + \mu_{\mathbf{X}_{m,i}}. \quad (5)$$

For vector quantizing $\mathbf{X}_{m,i}$, we note that the use of an isotropic sub-covariance matrix allows to use a codebook designed for a ZMUV subvector $\mathbf{Y}_{m,i}$ without any loss of quantization performance. That is, we achieve the same quantization performance as that of the case where a codebook is directly designed and used for vector quantizing $\mathbf{X}_{m,i}$.

Now, for the m'th Gaussian mixture specific SVQ, let us choose the dimensions of subvectors as equal as possible. Suppose $r_m = \left\lfloor \frac{p}{S_m} \right\rfloor$ and we choose the dimensions of subvectors as

$$q_{m,i} = \begin{cases} r_m & \text{for } 1 \leq i \leq S_m - 1, \\ p - r_m & \text{for } i = S_m. \end{cases} \quad (6)$$

Using such dimensions, we note that the m'th Gaussian mixture specific SVQ becomes scalable through the use of a set of codebooks designed for a r_m-dimensional ZMUV subvector and a $(p - r_m)$-dimensional ZMUV subvector.

Further invoking a constraint that the number of splits is equal for all the SVQs, i.e. $\forall m, S_m = S$, we note that all the M SVQs become scalable through the use of a set of codebooks designed for a r-dimensional ZMUV subvector and a $(p - r)$-dimensional ZMUV subvector, where $r = \left\lfloor \frac{p}{S} \right\rfloor$. The codebooks are designed at varying bits/subvector and used as reference codebooks. Using such a design choice of equal number of splits, subvector dimensions and structured GMM, we design bitrate scalable SVQs and hence, design a scalable GMPVQ. Note that if we use $S = p$, the optimum SVQs are nothing but optimum SQs providing poorest R/D performance with minimum complexity and the SVQ based GMPVQ method becomes the SQ based GMPVQ method of [5]. On the other hand, for $S = 1$, the SVQs are nothing but unconstrained VQs with best R/D performance at the expense of highest complexity. Therefore, the use of an optimum SVQ provides a trade-off between R/D performance and complexity which spans the two extreme limits of using an optimum SQ and an unconstrained VQ. For a practical SVQ method, a choice of splits decides the trade-off.

3.1 Optimum R/D Performance

In this subsection, we address the R/D performance optimality of the SVQ based GMPVQ method. For a b bits/vector GMPVQ coder, $b_c = \log_2 M$ bits are used for transmitting the winning quantizer (or mixture) identity and thus, the remaining $(b - b_c)$ bits are used for the corresponding PVQ, here the corresponding SVQ. In case of the m'th Gaussian mixture specific optimum SVQ, $(b - b_c)$ bits are allocated optimally to the S subvectors as shown in eq. (7) [11], where $b_{m,i}$ denotes the allocated bits to the i'th subvector. In eq. (7), K_r is a dimensionality dependent constant as $K_r = 2\left(\frac{r}{2}\Gamma\left(\frac{r}{2}\right)\right)^{\frac{2}{r}} \left(\frac{r+2}{r}\right)^{\frac{r}{2}}$. For each Gaussian mixture, the optimum bit allocation is carried out to minimize the mean square error (MSE) based on the high rate quantization theory as shown in [11]. To find the integer bit allocation from the real valued bit allocation of eq. (7), the well-known water-filling algorithm or the recently proposed lattice based algorithm by Farber and Zeger [15] may be applied. We use a

$$
b_{m,i} = \begin{cases} r\frac{b-b_c}{p} + \frac{r}{2}\log_2\left[\frac{\frac{1}{r}K_r\left|\mathbf{C}_{\mathbf{x}_{m,i}\mathbf{x}_{m,i}}\right|^{\frac{1}{r}}}{\left[\left(\frac{K_r}{r}\right)^{\frac{r}{S-1}}\frac{K_{p-r}}{p-r}\right]^{\frac{1}{p}}\left|\mathbf{C}_{m,\mathbf{x}}\right|^{\frac{1}{p}}}\right], & \text{for } 1 \le i \le S-1, \\ (p-r)\frac{b-b_c}{p} + \frac{(p-r)}{2}\log_2\left[\frac{\frac{1}{(p-r)}K_{(p-r)}\left|\mathbf{C}_{\mathbf{x}_{m,i}\mathbf{x}_{m,i}}\right|^{\frac{1}{(p-r)}}}{\left[\left(\frac{K_r}{r}\right)^{\frac{r}{S-1}}\frac{K_{p-r}}{p-r}\right]^{\frac{1}{p}}\left|\mathbf{C}_{m,\mathbf{x}}\right|^{\frac{1}{p}}}\right], & \text{for } i = S, \end{cases} \tag{7}
$$

simple heuristic algorithm [1] to get the integer bit allocation for practical implementation. The quantized informations are the winning quantizer identity $m^\star$ using $b_c = \log_2 M$ bits and the indices of SVQ codebooks using $\{b_{m^\star,i}\}_{i=1}^{S}$ bits. At the decoder, the reconstructed vector is obtained after covariance scaling and mean compensation.

3.2 Design of codebooks and memory complexity

For the SVQs of the GMPVQ method, we need to store a set of codebooks designed for a r-dimensional ZMUV subvector and a $(p-r)$-dimensional ZMUV subvector for varying bits/subvector. Let us consider that the highest bits/vector allocated to the GMPVQ method is $b^{(h)}$. We assume that the coder always uses a bitrate of b bits/vector where $b \le b^{(h)}$. For the highest $b^{(h)}$ bits/vector allocation, we can find the highest bit allocations to a subvector among the r-dimensional subvectors and to a $(p-r)$-dimensional subvector using eq. (7). For $b^{(h)}$ bits/vector allocation, let us denote the highest bit allocation to a subvector among r-dimensional subvectors as $b_r^{(h)}$ bits/subvector and to a $(p-r)$-dimensional subvector as $b_{p-r}^{(h)}$ bits/subvector. Therefore, $b_r^{(h)} = \max\left\{\left\{\{b_{m,i}\}_{i=1}^{S-1}\right\}_{\forall m}\right\}\Big|_{b=b^{(h)}}$ and $b_{p-r}^{(h)} = \max\left\{\left\{\{b_{m,i}\}_{i=S}\right\}_{\forall m}\right\}\big|_{b=b^{(h)}}$. To design scalable SVQs for all the Gaussian mixtures, we need to store the codebooks designed for a r-dimensional ZMUV subvector and a $(p-r)$-dimensional ZMUV subvector at continuum bits/subvector upto respectively $b_r^{(h)}$ bits/subvector and $b_{p-r}^{(h)}$ bits/subvector. For the SVQ based GMPVQ method, the total memory complexity ($\mathscr{M}$) to store the codebooks is (in floats)

$$
\mathscr{M} = \sum_{i=1}^{b_r^{(h)}} r \times 2^i + \sum_{i=1}^{b_{p-r}^{(h)}} (p-r) \times 2^i. \tag{8}
$$

Synthetic training datasets of ZMUV subvectors are used to design the codebooks through employing the LBG algorithm [14]. To keep the complexity under check, we assume that an allocation of highest bits/subvector does not exceed a practical limit of 10 bits/subvector[1]. Such a choice of highest bits/subvector can be invoked through a design choice of number of splits S.

4. SPEECH SPECTRUM QUANTIZATION

The application of the new GMPVQ method is demonstrated in the context of speech spectrum quantization. Even though the new quantization method can be used for several other applications, such as in image coding, speech waveform quantization etc., we consider the wide-band speech spectrum quantization problem as the real world application since the performance can be compared against the benchmark results available in the literature.

In linear prediction (LP) based speech coding, the LP spectrum is generally coded through the quantization of line spectrum frequency (LSF) parameters. The performance of LSF quantization is measured using a perceptual distance measure called spectral distortion (SD). It is prescribed that an average SD of 1 dB is required to achieve a transparent quality quantization performance (i.e. to achieve inaudible spectrum quantization distortion) [16], [17]. We assume that a coder need not perform better than achieving the quality of 1 dB average SD. As the direct use of SD is computationally intensive, an auxiliary distortion measure of vector dependent weighted square Euclidean distance (WSED) measure can be used for VQ of LSF parameters at high rate [16], [18]. The WSED approximates the SD at high rate [18].

We compared between three GMPVQ methods which are developed based on unstructured and structured GMMs: (1) the GMPVQ method of [2] (unstructured GMM based), (2) the GMPVQ method of [5] (structured GMM based), and (3) the proposed GMPVQ method (structured GMM based). The GMPVQ method of Subramaniam and Rao [2] uses an optimal TrSQ as a PVQ and hence, we refer this method as GMTrSQ. The GMPVQ method of Chatterjee and Sreenivas [5] is referred to as GMSQ where an optimal SQ is used as a PVQ. An SVQ is used as a PVQ in the new GMPVQ method and hence, it is referred to as GMSVQ. All these three GMPVQ methods are bitrate scalable and use AbS technique to choose the best codevector. For AbS, we use the WSED measure where the spectral sensitivity coefficients are used as the weighting coefficients [18]. We note that the GMTrSQ and GMSQ methods were earlier used for wideband LSF quantization [17], [5]. For GMTrSQ method, we used eight full covariance Gaussian mixtures to model a GMM (i.e. $M = 8$). To achieve best R/D performance, we designed a variable rate GMTrSQ method as implemented in [2]. On the other hand, for GMSQ and GMSVQ, we used a GMM consisting of 256 uncorrelated Gaussian mixtures (i.e. $M = 256$). To keep the computational complexity under check, $L = 10$ and $L = 5$ were respectively chosen for GMSQ and GMSVQ.

The speech data used in the experiments is from the TIMIT database. The specification of AMR-WB speech codec [19] is used to compute the 16-th order LP parameter vectors which are then converted to 16-dimensional LSF vectors. We used 361,046 LSF vectors as training data and 87,961 LSF vectors as test data (distinct from training data). Using the training data, the GMMs were trained using EM algorithm. For the SVQs of the GMSVQ method, we used a split arrangement of (3,3,3,3,4)-dimensional subvectors to code a 16-dimensional LSF vector. Therefore, in case of the GMSVQ method, we store the codebooks designed for

[1]For a practical application, a search complexity more than $O(2^{10})$ is computationally intensive.

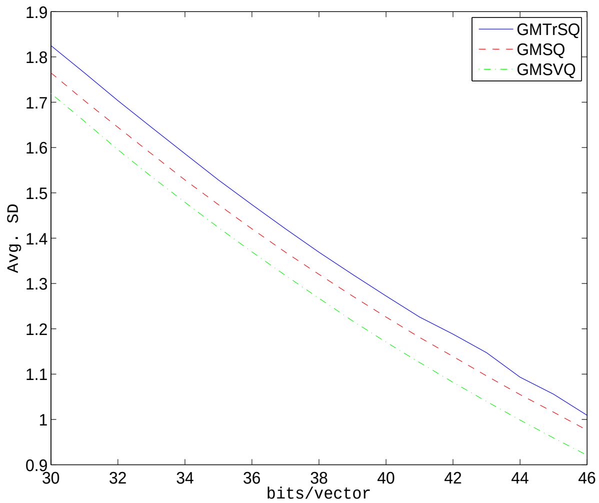

Figure 2: R/D performance comparison between three GMPVQ methods: GMTrSQ, GMSQ and GMSVQ.

three and four dimensional ZMUV subvectors from zero bits/subvector to the highest 10 bits/subvector.

Fig. 2 shows the R/D performance of the three methods. We note that the use of a GMM consisting of a large number of Gaussian mixtures in the GMSQ method leads to better performance than the GMTrSQ method where a GMM consisting of a few Gaussian mixtures is used. The use of a large number of mixtures helps to model a pdf better. Next, we note that the GMSVQ method provides better performance than the GMSQ method. The GMSVQ method provides an improvement of 1 bits/vector than the GMSQ method at any chosen bitrate. The GMSVQ method provides the transparent quality quantization performance of achieving 1 dB average SD at 44 bits/vector. For the GMSVQ method, we assume that more than 44 bits/vector is not required to use at any point of time. At 44 bits/vector, the coder uses a rate of nearly 3 bits/scalar which is in the region of high rate. On the other hand, at 30 bits/vector, the coder uses slightly less than 2 bits/scalar which is in the region of lower rate. Unlike the case of a lattice quantizer as shown in [3], we note that the use of SVQ does not lead to a poorer quantization performance compared to the use of SQ at the lower rate region.

5. CONCLUSIONS

We show that the use of a structured GMM along-with an optimum PVQ leads to a design of a bitrate scalable and optimum quantizer. For the application of wideband speech spectrum quantization, the new GMPVQ method is shown to perform the best compared to the recent GMPVQ methods.

We mention that there is still scope for further improvement in R/D performance considering the lower bound of LSF quantization [20]. Therefore, further research should consider to develop improved scalable PVQs along-with better source density modeling.

REFERENCES

[1] A. Gersho and R.M. Gray, “Vector quantization and signal compression,” Kluwer Academic, 1992.

[2] A.D. Subramaniam and B.D. Rao, “PDF optimized parametric vector quantization of speech line spectral frequencies,” *IEEE Trans. Speech, Audio Proc.*, vol. 11, No.2, pp. 130-142, March 2003.

[3] A.D. Subramaniam, W.R. Gardner and B.D. Rao, “Low-complexity source coding using Gaussian mixture models, lattice vector quantization, and recursive coding with application to speech spectrum quantization,” *IEEE Trans. Audio, Speech, Language Proc.*, vol. 14, Issue 2, pp. 524-532, March 2006.

[4] D.A. Reynolds and R.C. Rose, “Robust text-independent speaker identification using Gaussian mixture speaker models,” *IEEE Trans. Speech, Audio Proc.*, vol. 3, No. 1, pp. 72-83, Jan 1995.

[5] S. Chatterjee and T.V. Sreenivas, “Low complexity wideband LSF quanization using GMM of uncorrelated Gaussian mixtures,” *Proc. EUSIPCO*, Lausanne, Switzerland, August 2008.

[6] T.Z. Shabestary and P. Hedelin, “LSP quantization by a union of locally trained codebooks,” *IEEE Trans. Audio, Speech, Language Proc.*, vol. 13, Issue 5, pp. 811-820, Sept. 2005.

[7] S. Chatterjee and T.V. Sreenivas, “Switched Conditional PDF-based split VQ using Gaussian mixture model,” *IEEE Signal Proc. Letters*, vol. 15, pp. 91-94, 2008.

[8] S. Chatterjee and T.V. Sreenivas, “Optimum switched split vector quantization of LSF parameters,” *Signal Proc.*, vol. 88, Issue 6, pp. 1528-1538, June 2008.

[9] S. Chatterjee and T.V. Sreenivas, “Analysis-by-synthesis based switched transform domain split VQ using Gaussian mixture model,” *Proc. ICASSP*, Taipei, Taiwan, pp. 4117-4120, April 2009.

[10] J. Makhoul, S. Roucos and H. Gish, “Vector quantization in speech coding,” *Proc. IEEE*, vol. 73, No. 11, pp. 1551-1588, November 1985.

[11] S. Chatterjee and T.V. Sreenivas, “Optimum transform domain split VQ,” *IEEE Signal Proc. Letters*, vol. 15, pp. 285-288, 2008.

[12] P. Hedelin and J. Skoglund, “Vector quantization based on Gaussian mixture models,” *IEEE Trans. Speech, Audio Proc.*, vol. 8, No.4, pp. 385-401, July 2000.

[13] P.A. Chou, M. Effros and R.M. Gray, “A vector quantization approach to universal noiseless coding and quantization,” *IEEE Trans. Info. Theory*, vol. 42, no. 4, pp. 1109-1138, July 1996.

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

[15] B. Farber and K. Zeger, “Quantization of multiple sources using nonnegative integer bit allocation,” *IEEE Trans. Info. Theory*, vol. 52, Issue 11, pp. 4945-4964, June 2006.

[16] K.K. Paliwal and B.S. Atal, “Efficient vector quantization of LPC parameters at 24 bits/frame,” *IEEE Trans. Acoust., Speech, Signal Proc.*, vol. 1, pp. 3-14, Jan 1993.

[17] S. So and K.K. Paliwal, “A comparative study of LPC parameter representations and quantisation schemes for wide-band speech coding,” *Digital Signal Proc.*, Vol. 17, Issue 1, pp. 114-137, Jan 2007.

[18] W.R. Gardner and B.D. Rao, “Theoretical analysis of the high-rate vector quantization of LPC parameters,” *IEEE Trans. Speech, Audio Proc.*, vol. 3, No.5, pp. 367-381, Sept 1995.

[19] “AMR wide-band speech codec, transcoding functions (Release 5):” 3GPP TS 26.190 V 5.1.0.

[20] S. Chatterjee and T.V. Sreenivas, “Predicting VQ performance bound for LSF coding,” *IEEE Signal Proc. Letters*, vol. 15, pp. 166-169, 2008.