Logarithmic Cubic Vector Quantization: Concept and Analysis

Christian Rohlfing, Hauke Krüger and Peter Vary
Institute of Communication Systems and Data Processing (**ind**)
RWTH Aachen University, Germany
Email: {rohlfing,krueger,vary}@ind.rwth-aachen.de

***Abstract*—In this paper, we analyze Logarithmic Cubic Vector Quantization (LCVQ), a novel type of gain-shape vector quantization (GSVQ). In LCVQ, the vector to be quantized is decomposed into a gain factor and a shape vector which is a normalized version of the input vector. Both components are quantized independently and transmitted to the decoder. Compared to other GSVQ approaches, in LCVQ the input vectors are normalized based on the maximum norm (also denoted as L_∞-norm) instead of the typically used Euclidean norm (L_2-norm). Therefore, all shape vectors are located on the surface of the unit hypercube. As a conclusion, the shape vector quantizer can be realized based on uniform scalar quantizers yielding low computational complexity as well as high memory efficiency even in case of very high vector dimensions.**
In this paper, the concept of LCVQ is presented. Also, theoretical quantization performance measures for LCVQ as well as the optimal allocation of bit rate for gain factor and shape vector are derived. In order to assess the proposed LCVQ approach, the quantization performance achieved by LCVQ is compared to results which were recently derived for Logarithmic Spherical Vector Quantization (LSVQ), another highly efficient GSVQ scheme proposed in [1].

I. Introduction

In general, the quantizers are key elements in compression schemes for lossy source coding. In the design of a quantizer in practice, a good trade-off between complexity and quantization performance has to be achieved. On the one hand, scalar quantizers can be realized with low complexity but have only a moderate quantization performance. On the other hand, vector quantization techniques show promising results close to the theoretically achievable performance but have larger implementation costs.
For high-quality speech-audio-coding, vector quantizers with effective bit rates of more than 2 bits per sample are needed. In contrast to this, most of today's speech codecs employ sparse vector codebooks [2] which are only suitable for very low bit rates ($\leq$ 1 bit per vector dimension). Recently, Logarithmic Spherical Vector Quantization (LSVQ), has been theoretically investigated in [1] and [3]. In that context, LSVQ represents a **class** of Gain-Shape Vector Quantizers (GSVQs) which may involve different practical realizations of Spherical Vector Quantizers (SVQ), e.g., described in [4]. Novel realizations of LSVQ which achieve a high quantization performance with low computational cost have been proposed in [5] and [6], respectively. However, the proposed approaches for LSVQ require the storage of precomputed vector codebooks. In practice, this limits the available bit rates to low values ($\leq 2-3$ bits per sample).
Logarithmic Cubic Vector Quantization (LCVQ) is a novel type of gain-shape vector quantization which can be realized in a very efficient way and therefore offers some advantages over LSVQ: It achieves high quantization performance and at the same time operates with low complexity. Furthermore, it can be realized with high memory efficiency even for very high input vector dimensions and allows higher bit rates since no vector codebook is required. It is flexible in terms of the used bit rate which can even be adjusted at runtime.
In this paper, the concept of LCVQ as well as theoretical results are presented: In Section II, the principles of GSVQ in general and LSVQ as well as LCVQ in particular are briefly summarized. In Section III, novel equations for different aspects related to LCVQ are derived, in particular a formula to describe the LCVQ quantization signal-to-noise-ratio (SNR) for high bit rates. The derived equations are verified by Monte Carlo simulations and evaluated in Section IV to compare LCVQ and LSVQ in terms of the achievable quantization performance.

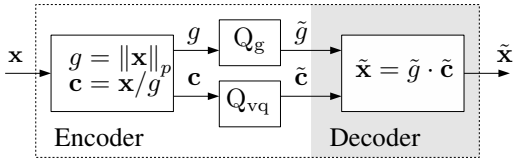


Figure 1. Parallel quantization of the gain and shape component with $p = 2$ and $\mathrm{Q_{vq}} = \mathrm{Q_{svq}}$ for LSVQ and $p = \infty$, $\mathrm{Q_{vq}} = \mathrm{Q_{cvq}}$ for LCVQ.

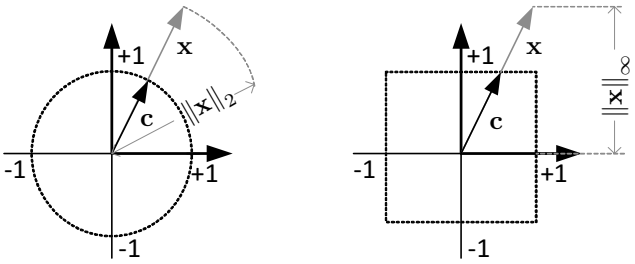
II. Generalized Gain-shape Vector Quantization

The principle of gain-shape vector quantization (GSVQ) [7] is illustrated by Figure 1: The input vector $\mathbf{x} \in \mathbb{R}^L$ with dimension L is decomposed into a gain factor $g \in \mathbb{R}^+$ and a shape vector $\mathbf{c} \in \mathbb{R}^L$ which are quantized independently by means of the scalar quantizer (SQ) Q_g and the vector quantizer (VQ) Q_{vq}, respectively. Gain value and shape vectors are computed as

$$g = \|\mathbf{x}\|_p = \left(\sum_{i=1}^{L} |x_i|^p \right)^{1/p} \quad \text{and} \quad \mathbf{c} = \frac{1}{g} \cdot \mathbf{x} \tag{1}$$

with $\|\mathbf{x}\|_p$ denoting the p-norm of a vector.
Both parts are transmitted to the decoder where the quantized version of the gain factor, $\tilde{g}$, and of the shape-vector, $\tilde{\mathbf{c}}$, are combined to produce the overall reconstruction vector

$$\tilde{\mathbf{x}} = \tilde{g} \cdot \tilde{\mathbf{c}} \ . \tag{2}$$



a) Euclidean norm: $p = 2$ a) Maximum norm: $p = \infty$

Figure 2. Transforming $\mathbf{x}$ into $\mathbf{c}$: Projection onto the different contour shapes, the surface of a unit sphere (a) and a unit cube (b) due to the Euclidean and the maximum norm, respectively, for the example of $L = 2$.

The main difference between LSVQ and LCVQ lies in equation 1 and can be illustrated based on Figure 2 for the example of $L = 2$: In LSVQ, with $p = 2$, the input vectors $\mathbf{x}$ are normalized to their Euclidean norm, $g = \|\mathbf{x}\|_2 = \sqrt{\mathbf{x}^T \cdot \mathbf{x}}$. As a result, all normalized shape vectors $\mathbf{c}$ are located on the surface of the L-dimensional unit **hypersphere**,

$$\mathbf{c} \in \mathcal{S}_L \text{ with } \mathcal{S}_L := \{\mathbf{x} \in \mathbb{R}^L : \|\mathbf{x}\|_2 = 1\}. \tag{3}$$

In contrast to this, LCVQ makes use of the maximum norm with $p = \infty$,

$$g = \|\mathbf{x}\|_\infty = \max(|x_1|, \ldots, |x_L|) \ . \tag{4}$$

As a conclusion, the normalized vectors $\mathbf{c}$ are located on the surface of the L-dimensional unit **hypercube**,

$$\mathbf{c} \in \mathcal{C}_L \text{ with } \mathcal{C}_L := \{\mathbf{x} \in \mathbb{R}^L : \|\mathbf{x}\|_\infty = 1\}. \tag{5}$$

The advantage of LCVQ compared to LSVQ lies in the fact that the design of a high-dimensional vector quantizer can be realized with significantly lower complexity if all vectors are located on the surface of a high dimensional hypercube than if the vectors are located on the surface of a hypersphere.

A. Logarithmic Quantization of the Gain Factor

Both approaches, LCVQ and LSVQ, have in common that the gain factor g is quantized using Logarithmic Scalar Quantization (LSQ) ($\mathrm{Q_g}$). This makes particularily sense for speech and audio signals since a high dynamic range of the input signal should be covered without that quantization overload occurs. Given the A-law companding rule [8] with the number of reconstruction levels N_g, compression factor A and stepsize Δg, it is shown in, e.g., [9], that the effective quantization interval width is non-uniform and a function of the reconstruction value $\tilde{g}$ located in the center of the quantization interval,

$$\Delta g(\tilde{g}) = \frac{C_A}{N_g} \cdot \tilde{g} \qquad \text{with} \qquad C_A := 1 + \ln(A) \tag{6}$$

if the gain factor g to be quantized falls into the logarithmic part of the A-law compression curve. The constant C_A is introduced here for a better readability of the equations derived in the following.

The signal-to-noise ratio (SNR) of LSQ is independent of the input vector probability density function (PDF),

$$\mathrm{SNR_{lsq}} \approx 6.02 R_{\mathrm{eff}} + \underbrace{10\log_{10}(3) - 20\log_{10}(C_A)}_{=P_{\mathrm{lsq}}}, \tag{7}$$

given on a logarithmic scale **in dB** with $R_{\mathrm{eff}} = \log_2(N_g)$ denoting the *effective bit rate per sample* of the quantizer. However, this independence comes at the price of the penalty-term P_{lsq} compared to the well-known $6\,\mathrm{dB}$-per-bit-rule [9]. In the following, LSQ is considered as a special case of LSVQ and LCVQ for input vector dimension $L = 1$ and will be refered to as the *lower limit* for the SNR plots for both LCVQ and LSVQ in Section IV.

B. Logarithmic Spherical VQ (LSVQ)

In LSVQ, the shape vectors $\mathbf{c}$ are quantized using a spherical vector quantizer (SVQ)

$$\mathrm{Q}_{vq} = \mathrm{Q_{svq}} : \mathbf{c} \to \tilde{\mathbf{c}} \in \mathcal{X}_{\mathrm{svq}} \tag{8}$$

involving a vector codebook $\mathcal{X}_{\mathrm{svq}}$ composed of N_{svq} spherical codevectors $\tilde{\mathbf{c}}$. In order to achieve high quantization performance, the codevectors are distributed over the surface of the L-dimensional unit hypersphere as uniformly as possible.
In [1] and [3], it is shown qualitatively as an intermediate result that LSVQ yields an overall quantization SNR which is independent of the PDF of the input vector. The derived formula, however, is only of qualitative nature due to an unknown quantization cell form-factor. Therefore, assumptions are made in [1] about the shape of the SVQ quantization cells and high bit rates are considered. Under these assumptions, a quantitative formula for an estimate of the maximum achievable LSVQ quantization SNR **in dB** is finally given as

$$\mathrm{SNR_{lsvq}} = 6.02 R_{\mathrm{eff}} - 10\log_{10}\left(\frac{L}{(L+1)^{\frac{L-1}{L}}}\left[2\sqrt{\pi}\frac{\Gamma\left(\frac{L+1}{2}\right)}{\Gamma\left(\frac{L}{2}\right)}\right]^{\frac{2}{L}}\left[\frac{C_A^2}{12}\right]^{\frac{1}{L}}\right). \tag{9}$$

In this formula, in analogy to Section II-A, the quantization SNR is expressed as a function of the effective bit rate per vector dimension (and hence sample) R_{eff} which is defined as

$$2^{R_{\mathrm{eff}} \cdot L} = N_g \cdot N_{\mathrm{svq}} = N_{\mathrm{lsvq}}. \tag{10}$$

Eq. (9) will be used in Section IV to compare the performance of LSVQ with the novel LCVQ scheme. The asymptotic maximum LSVQ quantization SNR **in dB** is

$$\lim_{L\to\infty} \mathrm{SNR_{lsvq}} = 6.02 R_{\mathrm{eff}} \tag{11}$$

and hence follows the 6-dB-per-bit-rule. Eq. (11) will be referred to as the *upper limit* in the SNR plots in Section IV. The design of spherical vector codebooks used for $\mathrm{Q_{svq}}$ is rather complex and shall not be discussed in detail in this paper. Due to practical reasons, in most cases, vector dimensions higher than $L = 16$ can be realized only for low bit rates ($\leq 2 - 3$ bits per vector dimensions).

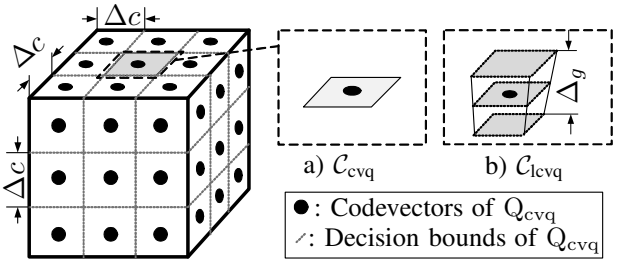


Figure 3. Exemplary hypercube for $L = 3$ with codevectors and quantization decision bounds of Q_{cvq} for $N_c = 3$. An exemplary (flat) quantization cell related to Q_{cvq} ($\mathcal{C}_{\text{cvq}}$) as well as a (non-flat) quantization cell related to the combination of Q_g and Q_{cvq} ($\mathcal{C}_{\text{lcvq}}$) are illustrated on the right side.

C. Logarithmic Cubic VQ (LCVQ)

In LCVQ, the gain factor is computed as

$$g = \max(|x_1|, \ldots, |x_L|) = x_{l_0} \tag{12}$$

with $l_0 = \arg\max_l(|x_l|)$ denoting the index of the maximum vector coordinate of $\mathbf{x}$. Consequently, after normalization with g, all shape vectors $\mathbf{c}$ are lying on the surface of the L-dimensional unit hypercube (5) with edge length equal to 2. Due to this normalization, the quantization of the shape vectors $\mathbf{c}$ has been transformed into an $(L-1)$-dimensional problem. The quantizer $\text{Q}_{\text{vq}} = \text{Q}_{\text{cvq}}$ can therefore be realized in a very efficient way by quantizing the $L-1$ vector coordinates $l \neq l_0$ of $\mathbf{c}$ by means of the **uniform scalar quantizer** Q_{sq} as

$$Q_{\text{cvq}}\colon \mathbf{c} \to \tilde{\mathbf{c}} = \begin{bmatrix}\tilde{c}_1 & \ldots & \tilde{c}_L\end{bmatrix}^T \text{ with } \tilde{c}_l = \begin{cases} Q_{\text{sq}}(c_l) & l \neq l_0 \\ \pm 1 & l = l_0 \end{cases} \tag{13}$$

Each uniform scalar quantizer Q_{sq} has the same number of reconstruction levels N_c yielding a quantization stepsize of $\Delta c = \frac{2}{N_c}$. In a practical application, the quantization indices for all vector coordinates $l \neq l_0$, the position of the maximum vector coordinate l_0 and the sign of vector coordinate $c_{l_0} = \pm 1$ are transferred to the decoder to reconstruct $\tilde{\mathbf{c}}$.
With $2 \cdot L$ surfaces for a hypercube of dimension L, the number of codevectors $\tilde{\mathbf{c}}$ covering the surface of the unit hypercube is $N_{\text{cvq}} = 2 \cdot L \cdot N_c^{L-1}$. With N_g as the number of quantization reconstruction levels of Q_g from Section II-A, the total number of LCVQ codevectors is

$$N_{\text{lcvq}} = 2^{R_{\text{eff}} \cdot L} = N_g \cdot N_{\text{cvq}} = N_g \cdot 2 \cdot L \cdot N_c^{L-1}. \tag{14}$$

Again, R_{eff} denotes the effective bit rate per vector coordinate. An example of the hypercube covered by quantization cells and reconstruction vectors related to Q_{cvq} for $L = 3$, $N_c = 3$ and $N_{\text{cvq}} = 2 \cdot L \cdot N_c^{L-1} = 54$ is given in Figure 3. Using only scalar quantizers, the LCVQ can be realized with very low memory and implementation cost. Also, since no codebook is involved, the bit rate can be easily adapted during runtime. In contrast to LSVQ, LCVQ has no practical limitations regarding the input vector dimension as well as the effective overall bit rate.

III. Theoretical Analysis of LCVQ

In order to derive mathematical equations for the LCVQ quantization SNR, in the first step, the quantization distortion related to Q_{cvq} shall be investigated. In the second step, the combination of Q_{cvq} and Q_g will be taken into account to compute the SNR related to LCVQ. In the last step, based on the derived LCVQ quantization SNR, the optimal allocation of bit rate for Q_g and Q_{cvq} is determined. In all cases, high bit rate assumptions are made such that the probability density function (PDF) of the input vectors $\mathbf{x}$ as well as the normalized input vectors $\mathbf{c}$ is assumed to be constant within each quantization cell. Note that given a vector $\mathbf{x}$ and its quantized version as $\tilde{\mathbf{x}}$, the quantization cost function is the **squared error** in the following,

$$d(\mathbf{x}, \tilde{\mathbf{x}}) = \|\mathbf{x} - \tilde{\mathbf{x}}\|_2^2 . \tag{15}$$

A. Analysis of Q_{cvq}

With the specification of Q_{cvq} in (13), quantization decision bounds as well as codevectors are defined which are located on the surface of the unit hypercube. Therefore, given a codevector as $\tilde{\mathbf{c}}$, a quantization cell is defined as all vectors in the area surrounding the codevector, constrained to be located on the hypercube surface,

$$\mathcal{C}_{\text{cvq}} := \{\mathbf{c} \in \mathcal{C}_L : Q_{\text{cvq}}(\mathbf{c}) = \tilde{\mathbf{c}}\} \tag{16}$$

with $\mathcal{C}_L$ from (5). An example quantization cell $\mathcal{C}_{\text{cvq}}$ is highlighted on the surface of the cube in Figure 3 and shown separately in Figure 3 a) for the example of $L = 3$. Due to the use of uniform scalar quantizers, the overall surface is covered with equally shaped quantization cells.
In order to derive the per-vector-coordinate quantization distortion of Q_{cvq}, contributions originating from all N_{cvq} quantization cells must be taken into account. Due to the high bit rate assumptions, the distribution of normalized vectors $\mathbf{c}$ within each quantization cell can be expressed

$$p_{\mathcal{C}_{\text{cvq}}}(\mathbf{c}) = 1/V_{\mathcal{C}_{\text{cvq}}} \quad \text{with} \quad V_{\mathcal{C}_{\text{cvq}}} = \int_{\mathcal{C}_{\text{cvq}}} \mathrm{d}\mathbf{c} \tag{17}$$

as the content (volume) of the quantization cell.
The quantization distortion for each single cell is

$$D_{\tilde{\mathbf{c}}} = \text{E}\left\{\|\mathbf{c} - \tilde{\mathbf{c}}\|_2^2\right\}/L = \int_{\mathcal{C}_{\text{cvq}}} p_{\mathcal{C}_{\text{cvq}}}(\mathbf{c}) \cdot d(\mathbf{c}, \tilde{\mathbf{c}}) \cdot \mathrm{d}\mathbf{c}. \tag{18}$$

The contributions from all cells are summed afterwards to yield

$$D_{\text{cvq}} = \frac{L-1}{L} D_{\text{sq}} = \frac{L-1}{L} \frac{1}{3 \cdot N_c^2}. \tag{19}$$

In this formula, the distortion related to Q_{cvq} is given as a function of the distortion related to a single scalar quantizer, $D_{\text{sq}} = \frac{\Delta c^2}{12}$, with N_c as defined in Section II-C.
The per-vector-coordinate signal power related to the normalized vectors $\mathbf{c}$ is defined as

$$S_{\text{cvq}} = \text{E}\left\{\|\mathbf{c}\|_2^2\right\}/L \quad \forall \quad \mathbf{c} \in \mathcal{C}_L. \tag{20}$$

S_{cvq} depends on the multivariate distribution of the vectors to be quantized, $p(\mathbf{x})$. For a uniform distribution of $\mathbf{x}$, it can be expressed as

$$S_{\text{cvq}} = \frac{1}{3} \cdot \frac{L-1}{L} + \frac{1}{L} \tag{21}$$

In all other cases, S_{cvq} can be determined numerically.

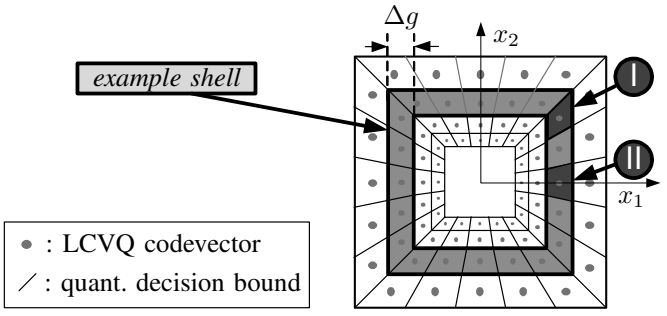


Figure 4. Coverage of vector space by *shells* of quantization cells for $L = 2$. Also, two exemplary cell shape types (labels I and II) are highlighted.

B. Analysis of Q_{lcvq}

In LCVQ, the combination of Q_{cvq} with Q_g leads to an an aggregation of scaled versions of the hypercube covered by codevectors as described in Section III-A. This aggregation is illustrated in Figure 4 for the example of $L = 2$ and leads to hypercube *shells* filling the L-dimensional vector space. One *example shell* is highlighted by darker color in Figure 4. The "thickness" of each shell is given by the quantization interval width related to Q_g, Δg. As a conclusion, the flat quantization cells $\mathcal{C}_{\text{cvq}}$ are transformed into non-flat LCVQ quantization cells $\mathcal{C}_{\text{lcvq}}$ by "adding" the height Δg. This transformation is also illustrated in Figure 3 b) where an exemplary LCVQ cell $\mathcal{C}_{\text{lcvq}}$ with height Δg is shown for the example of $L = 3$.
For the computation of the SNR, at first all LCVQ quantization cells within one shell are considered. Again, the distribution of signal vectors $\mathbf{x}$ within each LCVQ quantization cell is assumed to be uniform,

$$p_{\mathcal{C}_{\text{lcvq}}}(\mathbf{x}) = 1/V_{\mathcal{C}_{\text{lcvq}}} \quad \text{with} \quad V_{\mathcal{C}_{\text{lcvq}}} = \int_{\mathcal{C}_{\text{lcvq}}} \mathrm{d}\mathbf{x} \tag{22}$$

as the volume of the LCVQ quantization cell. Given the considered shell with the corresponding quantized gain factor $\tilde{g}$, the partial quantization distortion related to each quantization cell is computed in analogy to (18). In analogy to Equation (19), the contributions from all N_{cvq} cells within the shell are summed yielding

$$D_{\text{lcvq}}(\tilde{g}) \approx \frac{\tilde{g}^{L+2}}{V_{\mathcal{C}_{\text{lcvq}}}} \cdot \Delta c^{L-1} \cdot ([f_1(N_g) - f_2(N_g)] \cdot S_{\text{cvq}} + f_2(N_g) \cdot D_{\text{cvq}}(N_c)) \tag{23}$$

with S_{cvq} as the signal power (21) depending on $p(\mathbf{x})$, $D_{\text{cvq}}(N_c)$ as the quantization distortion (19) related to Q_{cvq} and $f_1(N_g)$ and $f_2(N_g)$ as constants defined as

$$f_1(N_g) = \left[\left(1 + \frac{C_A}{2N_g}\right)^{L+2} - \left(1 - \frac{C_A}{2N_g}\right)^{L+2} \right] \frac{1}{L+2} \tag{24}$$

and

$$f_2(N_g) = \left[(1 + \frac{C_A}{2N_g})^L - (1 - \frac{C_A}{2N_g})^L \right] \cdot \frac{L-1}{L \cdot (L+1)} + \left[(1 + \frac{C_A}{2N_g})^L + (1 - \frac{C_A}{2N_g})^L \right] \cdot \frac{C_A}{N_g \cdot (L+1)} \; . \tag{25}$$

The derivation of (23) involves more mathematical effort since the LCVQ quantization cells have different shapes depending on the position. In order to demonstrate this, two exemplary cells of different shape are highlighted by the labels I and II in Figure 4.
In analogy to the derivation of (23), the signal power related to all vectors $\mathbf{x}$ located within the considered shell can be derived as

$$S_{\text{lcvq}}(\tilde{g}) \approx \frac{\tilde{g}^{L+2}}{V_{\mathcal{C}_{\text{lcvq}}}} \cdot \Delta c^{L-1} \cdot f_1(N_g) \cdot S_{\text{cvq}}. \tag{26}$$

With (23) and (26), the overall quantization SNR related to the specified shell is

$$\frac{S_{\text{lcvq}}(\tilde{g})}{D_{\text{lcvq}}(\tilde{g})} \approx \frac{f_1(N_g) \cdot S_{\text{cvq}}}{[f_1(N_g) - f_2(N_g)] \cdot S_{\text{cvq}} + f_2(N_g) \cdot D_{\text{cvq}}}. \tag{27}$$

Obviously, the term $\frac{\tilde{g}^{L+2}}{V_{\mathcal{C}_{\text{lcvq}}}}$ in (23) and (26) cancels out for (27), and also the other variables in (27) do not depend on $\tilde{g}$. Therefore, the SNR computed for the specified shell is equal for all shells which makes (27) the formula for the LCVQ quantization SNR in general.

C. Optimal Bit Allocation

In order to optimally distribute the overall bit budget per vector, $R = L \cdot R_{\text{eff}}$, to Q_{g} and Q_{cvq}, at first a general citerion is investigated motivated by the well-known *normalized inertia* measure defined to find the optimal shape of a quantization cell in [10]. Given a quantizer with all quantization cells of identical shape and each described by means of a cubic base area with edge length Δc, a specific height Δg and a specific overall cell area content (volume), the cell shape is optimal in terms of the maximum achievable quantization performance if the overall cell is a cube, hence

$$\Delta g \stackrel{!}{=} \Delta c \; . \tag{28}$$

This constraint shall be adopted for LCVQ. However, in LCVQ, the resulting quantization cells are not perfectly cubic and not of the same shape (demostrated by Figure 4). In order to consider this, the diversity of cell shapes is taken into account by a modification of constraint (28) as

$$\alpha \cdot \Delta g(\tilde{g}) \stackrel{!}{=} \tilde{g} \cdot \Delta c \tag{29}$$

with the still unknown *average cell shape correction factor* α. Note here that in comparison to the general consideration to define (28), the height of the quantization cell depends on $\tilde{g}$ in case of LCVQ. Combining (29) and (14) yields

$$N_g = \sqrt[L]{N}\frac{1}{2}\sqrt[L]{\frac{(\alpha C_A)^{L-1}}{L}} \quad \text{and} \quad N_c = \sqrt[L]{N}\sqrt[L]{\frac{1}{\alpha C_A L}} \tag{30}$$

to compute N_g and N_c as a function of $N = 2^R$ and the constant C_A. In order to compute the optimal value for α, (30) is substituted in (27). The resulting equation yields the LCVQ quantization SNR given the effective overall bit rate R_{eff} as a function of the unknown coefficient α. Therefore, it must be optimized to find the optimal value α which, however, can only be done numerically due to the complexity of that equation. Exemplary values for α are given in Figure 5 for $L = \{2, 8, 48\}$ and $R_{\text{eff}} = 1 \ldots 10$.

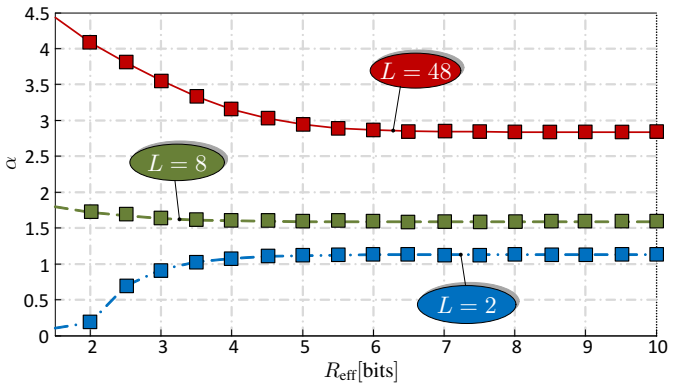


Figure 5. Exemplary values for α for the optimal allocation of bits for Q_g and Q_{cvq} for $L = \{2, 8, 48\}$ and $R_{\text{eff}} = 1 \ldots 10$.

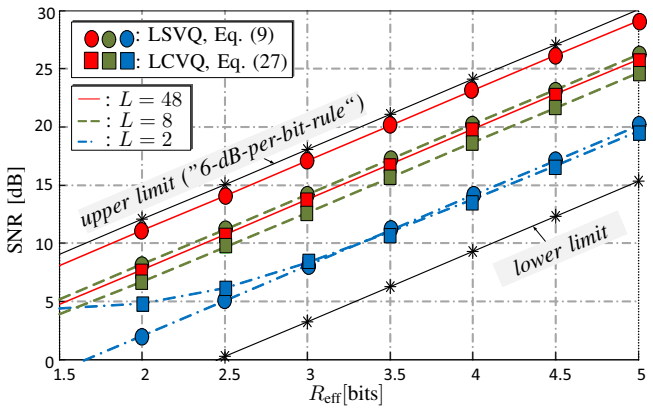


Figure 6. LCVQ quantization SNR (27) compared to LSVQ quantization SNR (9) for **normally** distributed input vectors.

IV. Evaluation

For all evaluations, the companding factor of all logarithmic scalar quantizers is set to $A = 5000$ which is shown as a reasonable trade-off between dynamic range and performance for the quantization of audio signals in [1].
In order to verify the correctness of the derived equations for LCVQ, Monte Carlo simulations were conducted for uniformly and normally distributed input vectors: The theoretical solution leads to asymptotically identical SNR values as the simulation for high bit rates. For lower bit rates and lower vector dimensions a deviation of the computed SNR values from the simulated values was observed. This can be explained with the assumptions of high bit rates in Section III.

For a comparison of LCVQ with LSVQ, SNR values for LCVQ as well as LSVQ were computed based on Equation (27) and Equation (9), respectively, for different effective bit rates $1.5 < R_{\text{eff}} < 5$ bit and vector dimensions $L = \{2, 8, 48\}$ for uniformly and normally distributed random vectors.
In case of **uniformly** distributed vectors, LCVQ achieves approximately the same quantization SNR as LSVQ for all input vector dimensions L. Due to the limited space in this paper and the fact that the SNR curves for LCVQ and LSVQ are almost identical, however, the corresponding SNR plots shall not be shown here. Instead, the SNR curves computed for a **normal** distribution are shown in Figure 6: The shown SNR plots for LCVQ and LSVQ are bounded by the *lower limit* which is the performance of LCVQ and LSVQ for a vector dimension $L = 1$ (LSQ, refer to Section II-A) and the *upper limit* which is the asymptotic performance of the LSVQ for infinite vector dimension, $L \to \infty$, Equation (11). In the figure, it is shown that LSVQ has a performance benefit compared to LCVQ, approximately 4 dB benefit for $L = 48$, 2 dB benefit for $L = 8$ and almost no benefit for $L = 2$. The reason for the lower performance of the LCVQ is its suboptimal cubic arrangement of codevectors given a spherically distributed random variable (refer to, e.g., [11]). However, the LCVQ was introduced as an alternative to LSVQ with lower complexity and higher flexibility. This partly compensates this drawback in practice as it allows the use of LCVQ at significantly higher vector dimensions and bit rates than the maximum supported bit rates of the LSVQ.

V. Conclusions

In this paper, the concept of Logarithmic Cubic Vector Quantization (LCVQ) was proposed. A theoretical analysis was presented based on the assumption of a high overall bit rate available for quantization. Equations describing the quantization signal-to-noise ratio were derived, and, finally, the optimal allocation of bit rate for the quantization of the gain factor and the normalized shape vectors was computed.
The derived theoretical results were verified by Monte Carlo simulations. Also, the LCVQ was compared to the rececently proposed Logarithmic Spherical Quantization (LSVQ): LCVQ offers the advantages of low complexity, high memory efficiency and a higher flexibility. Nevertheless, based on the derived equations it was shown that the quantization performance is very similar to that of the LSVQ for uniformly distributed signals. Only for signals following a normal distribution, the LSVQ has performance benefits compared to LCVQ.

References

[1] H. Krüger, *Low Delay Audio Coding Based on Logarithmic Spherical Vector Quantization*, ser. Aachener Beiträge zu digitalen Nachrichtensystemen (Dissertation). Wissenschaftsverlag Mainz, 2010, vol. 25.

[2] C. Laflamme et al., "On reducing computational complexity of codebook search in CELP coder through the use of algebraic codes," in *Proceedings of ICASSP*, 1990, pp. 177–180.

[3] H. Krüger, R. Schreiber, B. Geiser, and P. Vary, "On Logarithmic Spherical Vector Quantization," in *Proceedings of ISITA*, 2008.

[4] J. Hamkins, "Design and Analysis of Spherical Codes," Ph.D. dissertation, University of Illinois, 1996.

[5] H. Krüger, B. Geiser, P. Vary, H. T. Li, and D. Zhang, "Gosset lattice spherical vector quantization with low complexity," in *Proceedings of ICASSP*, May 2011, pp. 485 –488.

[6] H. Krüger and P. Vary, "SCELP: Low Delay Audio Coding with Noise Shaping Based on Spherical Vector Quantization," in *Proceedings of EUSIPCO, EURASIP 2006*, 2006.

[7] M. Sabin and R. Gray, "Product code vector quantizers for waveform and voice coding," *IEEE Trans. on Acoust., Speech and Signal Proc.*, vol. 32, no. 3, pp. 474 – 488, Jun. 1984.

[8] N. S. Jayant and P. Noll, *Digital Coding of Waveforms, Principles and Applications to Speech and Video*. Englewood Cliffs NJ, USA: Prentice-Hall, 1984.

[9] P. Vary and R. Martin, *Digital Speech Transmission: Enhancement, Coding and Error Concealment*. John Wiley & Sons, 2006.

[10] A. Gersho, "Asymptotically Optimal Block Quantization," *Information Theory, IEEE Transactions on*, vol. 25, no. 4, pp. 373 – 380, Jul. 1979.

[11] T. D. Lookabough and R. Gray, "High-Resolution Quantization Theory and the Vector Quantizer Advantage," *IEEE Transactions on Information Theory*, vol. 35, no. 5, pp. 1020–1083, 1989.