

Quantização vetorial utilizando códigos esféricos em camadas de toros

Fabiano Boaventura de Miranda e Cristiano Torezzan

Resumo—Neste trabalho apresentamos uma proposta de quantizador vetorial de fontes gaussianas que combina códigos esféricos em camadas de toros e a técnica *shape/gain*. Os conceitos básicos de códigos esféricos em camadas de toros são revisados e exemplificados através da construção de um quantizador vetorial para o *shape*, em dimensão 48, que utiliza o reticulado Leech, em dimensão 24, como pré-imagem. O principal custo computacional para a quantização, consiste em decodificar em um reticulado, na metade da dimensão do código. No exemplo proposto, utiliza-se um algoritmo polinomial de decodificação do reticulado Leech. Para a quantização do *gain*, um quantizador escalar é otimizado pelo algoritmo de Lloyd-Max para uma taxa definida a priori. O processo proposto é descrito em detalhes e alguns resultados de simulações numéricas são apresentados, mostrando que o esquema proposto neste trabalho pode superar os melhores quantizadores vetoriais esféricos conhecidos na literatura.

Palavras-Chave—Quantização vetorial, Fontes gaussianas, Códigos esféricos, Compressão de dados.

Abstract—In this paper we present a novel approach to the problem of vector quantization for gaussian sources, combining a spherical code in layers of flat tori and the *shape/gain* technique. The basic concepts of spherical codes in tori layers are reviewed and exemplified by a 48-dimensional vector quantizer for the *shape*, which uses the 24-dimensional Leech lattice as its pre-image. A scalar quantizer is optimized for the *gain* using Lloyd-Max algorithm for a given rate. The main complexity cost refers to the lattice decoding, which is done in the half of the code dimension. In the proposed example, we apply a polynomial algorithm for the Leech lattice decoding. The proposed quantizer is described in details and some numerical results are presented showing that our approach may outperform some competitive state-of-art technique for vector quantization.

Keywords—Vector quantization, Gaussian sources, Spherical codes, Data compression.

I. INTRODUÇÃO

A quantização é um procedimento que consiste em mapear um conjunto de dados de um domínio contínuo (ou muito grande) para um domínio enumerável (em geral finito). O arredondamento e o truncamento de números reais são exemplos simples de quantização. Na área de processamento de sinais, a conversão analógico-digital é um exemplo usual. Dependendo do contexto, a quantização pode ser vista como um processo similar à compressão de dados, onde admite-se alguma perda de informação em troca de simplificar a representação de dados e, assim, obter vantagens em termos de armazenamento. A diferença entre o valor de uma variável e sua versão quantizada é denominada erro de quantização

Instituto de Matemática e Estatística Computacional e Científica - Unicamp, Campinas, SP, E-mail: fabiano.miranda@ueg.br. Faculdade de Ciências Aplicadas - Unicamp, Limeira, SP, E-mail: cristiano.torezzan@fca.unicamp.br. Projeto temático FAPESP 2013/25977-7.

e é uma das principais métricas de desempenho de um quantizador.

Em termos gerais, existem três problemas importantes envolvendo a quantização de fontes: projetar o quantizador, mensurar o erro médio observado (distorção) ao recuperar os dados quantizados e gerenciar o custo computacional (ou tempo gasto) para codificar cada variável proveniente da fonte.

Neste trabalho estamos interessados em um processo específico, denominado quantização vetorial, que consiste em mapear vetores de um espaço k -dimensional em um conjunto finito (ou enumerável) de pontos.

A ênfase será dada à um método de quantização vetorial conhecido pelo termo em inglês *shape/gain* (forma/ganho), que explora uma combinação de quantização escalar e esférica. A ideia básica consiste em representar cada vetor $x \in \mathbb{R}^k$, pelo vetor $\hat{x} = \hat{g}\hat{s} \in \mathbb{R}^k$, onde \hat{g} é um escalar que representa a quantização do valor $g = \|x\|$ (denominado *gain*) e \hat{s} é o ponto de um código esférico que está mais próximo do vetor $s = x/g$ (denominado *shape*). O erro quadrático de quantização é mensurado por $d = \|x - \hat{x}\|^2$ e o sistema de quantização deve ser projetado de forma a minimizar o erro quadrático médio (MSE). Mais detalhes sobre a quantização *shape/gain* podem ser encontrados em [3], [6]. A Figura 1 ilustra a quantização de um vetor $x = gs \in \mathbb{R}^2$.

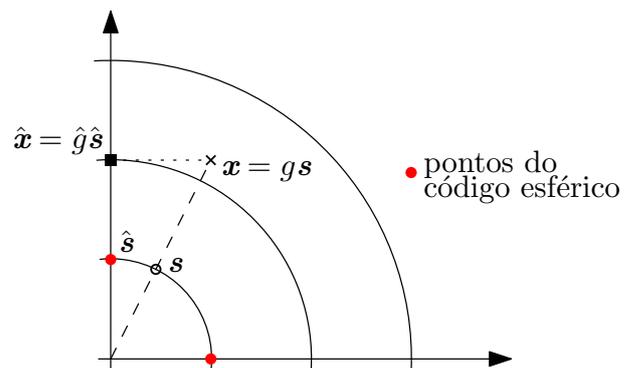


Fig. 1. Ilustração do processo de quantização vetorial *shape/gain* em \mathbb{R}^2 .

Neste artigo apresentamos uma nova proposta de quantização para o *shape*, que utiliza códigos esféricos em camadas de toros [2] para mapear pontos do reticulado Leech da dimensão 24 para a dimensão 48. Para a quantização do *gain* utiliza-se, como usual, um quantizador escalar, otimizado pelo algoritmo de Lloyd-Max.

Na seção II apresenta-se uma breve revisão sobre a construção de códigos esféricos em camadas de toros e descreve-se três famílias de códigos que serão utilizadas nas simulações computacionais. Na seção III o processo

de quantização vetorial é descrito, com ênfase na técnica *shape/gain* e no uso de códigos esféricos em camadas de toros. Na seção IV apresentamos alguns resultados de simulações computacionais utilizando o quantizador proposto neste trabalho e comparamos os resultados com o quantizador descrito em [1], que é considerado uma referência para codificação vetorial de fontes gaussianas. Os resultados numéricos mostram que o quantizador proposto neste trabalho pode superar os resultados de [1], para algumas alocações de taxas.

II. CÓDIGOS ESFÉRICOS EM CAMADAS DE TOROS

Um código esférico é um subconjunto da esfera k -dimensional $S^{k-1} \subset \mathbb{R}^k$ e pode ser visto como a generalização natural das constelações planares M-PSK (*Phase Shift Keying*) para dimensões superiores a 2. Desde o trabalho pioneiro de Shannon [13], tais códigos têm sido extensivamente explorados para a transmissão de sinais através um canal gaussiano [3].

Alternativamente, pontos distribuídos sobre a esfera unitária possuem diversas outras aplicações, incluindo a construção de quantizadores vetoriais, como proposto em [4] e investigado no presente artigo. Dentre as diversas maneiras possíveis para se construir códigos esféricos, neste trabalho focamos em códigos construídos em camadas de toros.

Os códigos esféricos em camadas de toros foram introduzidos em [1] e posteriormente detalhados em [2]. A ideia central da construção desses códigos consiste em explorar a folheação da esfera unitária $2k$ -dimensional, $S^{2k-1} \subset \mathbb{R}^{2k}$, por toros planares e mapear conjuntos regulares de pontos (e.g. reticulados) da dimensão k para a $2k$. Uma das principais vantagens dessa construção é que tanto o processo de construção do código, quanto procedimentos de decodificação podem ser realizados na metade da dimensão do código, reduzindo sua complexidade computacional.

A seguir, apresentamos uma breve revisão, baseada em [1], sobre a construção dos códigos esféricos em camadas de toros.

Seja $\mathbf{c} = (c_1, c_2, \dots, c_k) \in S^{k-1} \subset \mathbb{R}^k$, $c_i > 0$ com $\sum_{i=1}^k c_i^2 = 1$ e $\mathbf{y} = (y_1, y_2, \dots, y_k) \in \mathbb{R}^k$. Seja $\Phi_{\mathbf{c}} : \mathbb{R}^k \rightarrow \mathbb{R}^{2k}$ definida por

$$\Phi_{\mathbf{c}}(\mathbf{y}) = \left(c_1 \left(\cos \frac{y_1}{c_1}, \sin \frac{y_1}{c_1} \right), \dots, c_k \left(\cos \frac{y_k}{c_k}, \sin \frac{y_k}{c_k} \right) \right), \quad (1)$$

a imagem de $\Phi_{\mathbf{c}}$ é o toro $T_{\mathbf{c}}$, um objeto k -dimensional na superfície esférica unitária S^{2k-1} . Como a função $\Phi_{\mathbf{c}}$ é periódica, o toro $T_{\mathbf{c}}$ é também a imagem dos pontos restritos a caixa k -dimensional

$$P_{\mathbf{c}} = \{ \mathbf{y} \in \mathbb{R}^k; 0 \leq y_i < 2\pi c_i, \quad 1 \leq i \leq k \}. \quad (2)$$

Dados $k \geq 2$ e $d \in (0, \sqrt{2})$, seja $C(k, d)$ um código esférico k -dimensional qualquer, com distância mínima maior ou igual a d . O código $C_T(2k, d)$ é construído em duas etapas, como se segue [2]:

- i) Seleccionamos os pontos em $C(k, d)$ que possuem somente coordenadas não negativas. Vamos denotar este sub-código por

$$C(k, d)_+ = \{ \mathbf{c} \in C(k, d) : c_i > 0 \ ; \ 1 \leq i \leq k \}.$$

Cada ponto $\mathbf{c} \in C(k, d)_+$ define um toro planar $T_{\mathbf{c}}$ na esfera unitária S^{2k-1} .

- ii) Para cada toro $T_{\mathbf{c}}$ definido por $C(k, d)_+$, determinamos um conjunto finito de pontos

$$Y_{T_{\mathbf{c}}} \subset P_{\mathbf{c}}, \quad e \quad Y_{T_{\mathbf{c}}} = \{ \mathbf{y}; 0 \leq y_i < 2\pi c_i \}$$

tal que

$$\| \Phi_{\mathbf{c}}(\mathbf{y}) - \Phi_{\mathbf{c}}(\mathbf{x}) \| \geq d, \quad \forall \mathbf{x}, \mathbf{y} \in Y_{T_{\mathbf{c}}}$$

O código esférico em camadas de toros é

$$C_T(2k, d) = \bigcup_{\mathbf{c} \in C(k, d)_+} \Phi(Y_{T_{\mathbf{c}}})$$

Como observado em [2], a estrutura de um $C_T(2k, d)$ está diretamente relacionada ao sub-código $C(k, d)_+$ e ao conjunto de pontos $Y_{T_{\mathbf{c}}}$ escolhidos. Em (i) é desejável que $C(k, d)_+$ tenha boa densidade de empacotamento em S^{k-1} e, se possível, alguma propriedade algébrica ou geométrica. Inclusive um código esférico não estruturado pode ser usado, uma vez que a cardinalidade dos toros é pequena em relação ao número de pontos no código. Para a etapa (ii), consideramos nos exemplos construídos para este trabalho pontos de algum reticulado k -dimensional conhecido. A Figura 2 ilustra o procedimento de construção de uma das camadas de um código esférico $C_T(4, d)$.

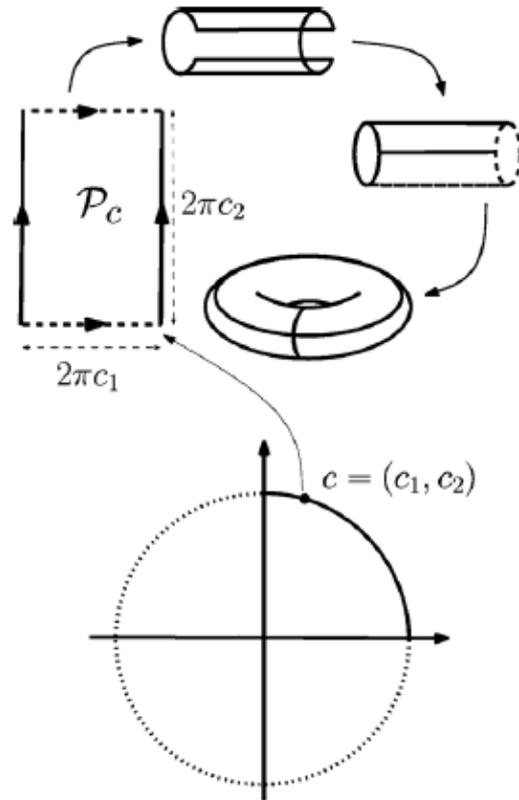


Fig. 2. Construção de uma das camadas de um código esférico $C_T(4, d)$

A. Código esférico em camadas de toros em dimensão 48

Nesta seção descrevemos três famílias de códigos esféricos em dimensão 48, construídos a partir do reticulado Leech (na dimensão metade) que serão utilizados no quantizador proposto neste artigo. Estes códigos esféricos serão denotados por $C_T(48, d)$.

Para etapa (i) da construção devemos escolher um subconjunto de pontos $C(k, d)_+$ que definirão as camadas dos toros. Neste trabalho, consideramos famílias de códigos propostas em [2] e [14], cujos pontos $C(k, d)_+$, que definem os toros, são obtidos pelo conjunto das permutações do vetor

$$c(t) = \frac{(t, 1, \dots, 1)}{\sqrt{23^2 + t^2}}, \quad (3)$$

onde o parâmetro t é ajustado de acordo com a distância mínima pretendida. Esta abordagem produz 24 toros idênticos, com distância mínima dada em função de t por [2]

$$d(t) = \frac{(t-1)\sqrt{2}}{\sqrt{23^2 + t^2}}. \quad (4)$$

Por exemplo, para $t = 1.0705$ temos 24 toros simétricos com distância mínima $d = 0.0203$.

Outro exemplo de código $C(k, d)_+$ pode ser obtido a partir das permutações do vetor

$$g_i(t) = \frac{(e_i + te)}{\|e_i + te\|}, \quad (5)$$

em que $e = (1, \dots, 1) \in \mathbb{R}^{24}$, $t > 0$ e e_i é o i -ésimo vetor canônico de \mathbb{R}^{24} . Esta abordagem também produz 24 toros idênticos, com distância mínima dada em função de t por [14]

$$d(t) = \frac{\sqrt{2}}{\sqrt{24t^2 + 2t + 1}}. \quad (6)$$

Além desses toros, também utilizamos como alternativa para a etapa (i), um conjunto de toros $C(k, d)_+$ obtido pelas permutações do vetor

$$p(t) = \frac{(1, 1+t, \dots, 1+23t)}{\sum_{i=1}^{24} \sqrt{(1+it)^2}}. \quad (7)$$

Esta abordagem produz 24! toros idênticos, com distância mínima dada em função de t por [2]

$$d(t) = \frac{t\sqrt{2}}{\sum_{i=1}^{24} \sqrt{(1+it)^2}}. \quad (8)$$

O uso de diferentes conjuntos para definir os toros ($C(k, d)_+$) tem por finalidade tornar as construções mais versáteis em termos de distância mínima e taxas binárias.

Para a etapa (ii), após definido o código $C(k, d)_+$ utilizando algum dos procedimentos descritos acima, escalonamos os pontos do reticulado Leech para que tenham distância mínima d e alocamos um conjunto finito de pontos desse reticulado dentro de cada toro T_C . Como os toros gerados possuem as mesmas coordenadas, a menos de permutação, a quantidade de pontos em cada toro será a mesma. O código esférico $C_T(48, d)$ será formado pela união das imagens de cada toro pela função (2). Por exemplo, definindo os toros pela equação (3), com distância mínima $d = 0.1$, teremos $t = 1.35234$, taxa

$R = 2.35$ bits por dimensão e o número de pontos no código $C_T(48, 0.1)$ será de aproximadamente 1.07091×10^{34} .

Nas simulações computacionais realizadas, o parâmetro t foi ajustado em função da taxa desejada do quantizador, quanto maior a taxa desejada, menor deve ser a distância mínima para que o código comporte mais pontos.

Na próxima seção descrevemos como os códigos esféricos podem ser utilizados para projetar um quantizador vetorial.

III. QUANTIZAÇÃO VETORIAL UTILIZANDO CÓDIGOS ESFÉRICOS EM CAMADAS DE TOROS

A. Quantização vetorial shape/gain

Seja $\mathbf{x} = (x_1, x_2, \dots, x_k) \in \mathbb{R}^k$ uma sequência de variáveis aleatórias independentes e identicamente distribuídas (iid), com função de densidade de probabilidade (fdp) dada por $p(x)$. O processo de quantização da fonte envolve duas etapas: codificação e decodificação. O codificador associa a sequência \mathbf{x} a um único índice $Q(\mathbf{x}) \in \{1, 2, \dots, 2^{kR}\}$. O decodificador representa \mathbf{x} por um estimador $\hat{\mathbf{x}}$ associado ao índice $Q(\mathbf{x})$. O valor R é denominado taxa de quantização, em bits. Se definido o número de bits que irá representar a informação, o número de palavras código no dicionário será dado por

$$N = 2^{k \cdot R}. \quad (9)$$

O erro no processo de quantização vetorial (QV) é denominado distorção e pode ser mensurado pelo quadrado da distância euclidiana entre o vetor quantizado e o vetor de entrada. Se \mathbf{x} é o vetor de entrada e $\hat{\mathbf{x}}$ é o vetor quantizador então a distorção é dada por:

$$d(\mathbf{x}, \hat{\mathbf{x}}) = \sum_{i=1}^k (x_i - \hat{x}_i)^2. \quad (10)$$

O erro quadrático médio (MSE) por dimensão é definido como sendo a esperança do quadrado da distância euclidiana por dimensão, ou seja,

$$D = \frac{1}{k} E[d(\mathbf{x}, \hat{\mathbf{x}})]. \quad (11)$$

Uma medida alternativa para mensurar o desempenho do quantizador é a SNR, do inglês *signal-to-noise ratio*, definida como

$$\text{SNR} = 10 \log_{10} \frac{E[\|\mathbf{x}\|^2]}{D}, \quad (12)$$

onde a unidade de medida é a decibels (dB) e $E[\|\mathbf{x}\|^2]$ é a potência (energia) do sinal de entrada com relação a norma euclidiana. Pode-se dizer que o SNR compara o nível do sinal desejado pela distorção gerada pelo quantizador. Assim, minimizar o MSE equivale a maximizar o SNR.

Existe um limitante inferior teórico para fontes gaussianas, proposto por Shannon [6], que expressa a distorção D em função da taxa R e da variância σ^2 através da equação

$$D(R) = \sigma^2 2^{-2R}. \quad (13)$$

Pode-se mostrar que esse limitante é alcançado assintoticamente fazendo ($k \rightarrow \infty$), obtendo um limitante superior para quantização de fontes gaussianas dado por

$$\text{SNR}^* = 6.02R \text{ dB}. \quad (14)$$

Como o propósito básico de qualquer quantizador é minimizar o MSE, as técnicas se diferenciam principalmente na maneira de distribuir os pontos do quantizador no espaço. Neste trabalho propomos a utilização do esquema de quantização esférica *shape/gain*, descrito como segue.

Seja $\mathbf{x} \in \mathbb{R}^k$ um vetor de variáveis aleatórias gaussianas iid, o *gain* de \mathbf{x} é representado por $g = \|\mathbf{x}\|$ e seu *shape* por $\mathbf{s} = \frac{\mathbf{x}}{\|\mathbf{x}\|}$. Assim, $\mathbf{x} = g\mathbf{s}$. Como o *gain* de \mathbf{x} é um escalar, ou seja, $g \in \mathbb{R}$ e o *shape* é um vetor, a quantização *shape/gain* é a composição de uma quantização escalar com uma quantização vetorial. Chamaremos de \hat{g} o valor que quantiza g e $\hat{\mathbf{s}}$ a quantização do vetor \mathbf{s} . Assim, o vetor $\hat{\mathbf{x}}$ que quantiza \mathbf{x} é dado por $\hat{\mathbf{x}} = \hat{g}\hat{\mathbf{s}}$, conforme ilustrado na Figura 1.

A quantização do *gain* pode ser feita utilizando qualquer quantizador escalar otimizado. Neste trabalho utilizamos um quantizador escalar definido através do algoritmo de Lloyd-Max, com uma taxa fixada a priori.

Para a quantização do *shape* utilizaremos pontos de um código esférico em camadas de toros construídos especificamente para esta finalidade.

Assim, se $C_g = 1, \dots, N_g$ e $C_s = 1, \dots, N_s$ representam, respectivamente, os dicionários dos quantizadores do *gain* e do *shape*, o dicionário do quantizador será dado por $C = C_g \times C_s$ e o número de palavras em C é dado por $N_c = N_g N_s$.

A codificação pode ser realizada em duas etapas: calcula-se g e encontra-se \hat{g} no dicionário (*codebook*) C_g , que pode ser encontrado através do algoritmo de Lloyd-Max e então, separadamente, calcula-se \mathbf{s} e decodifica-se $\hat{\mathbf{s}}$ no dicionário C_s do *shape*, obtendo por fim, o vetor $\hat{\mathbf{x}} = \hat{g}\hat{\mathbf{s}}$. A cardinalidade do dicionário do quantizador *shape/gain* é dada por

$$\|C\| = \|C_g\| \cdot \|C_s\|. \quad (15)$$

A taxa R deste quantizador é medida somando-se a taxa da quantização do *gain* R_g com a do *shape* R_s , $R = R_g + R_s$, logo

$$R = R_g + R_s = \frac{\log_2 N_g}{k} + \frac{\log_2 N_s}{k} = \frac{\log_2 N_g \cdot N_s}{k}. \quad (16)$$

O processo *shape/gain* pode ser representado pelo esquema da Figura 3.

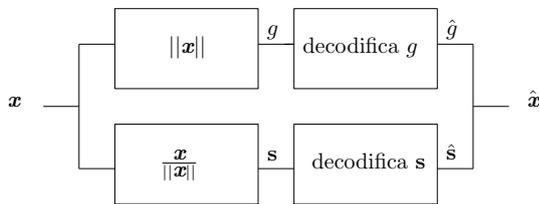


Fig. 3. Esquema do processo *shape/gain*

B. Quantizador vetorial *shape/gain* para códigos esféricos em camadas de toros

O processo de quantização vetorial *shape/gain* para códigos esféricos em camadas de toros pode ser descrito como segue.

Dada uma taxa $R = R_s + R_g$ e uma distância mínima d , teremos $N = 2^{2kR}$ pontos para distribuímos entre os dois dicionários C_g e C_s .

Seja $\mathbf{x} \in \mathbb{R}^{2k}$, onde x_i é a realização de uma variável aleatória gaussiana iid. Decomponha $\mathbf{x} = g\mathbf{s}$ conforme descrito na Seção anterior. Utilizando o algoritmo de Lloyd-Max o dicionário C_g é construído com $|C_g| = 2^{2kR_g}$ e é possível encontrar $\hat{g} \in C_g$ tal que $\hat{g} = \arg \min_i |g - \hat{g}_i|$ para $i = 1, 2, \dots, |C_g|$.

De forma independente, o ponto $\hat{\mathbf{s}} \in C_s$ é encontrado tal que $\hat{\mathbf{s}} = \arg \min_{\mathbf{y} \in C_T(2k, d)} \|\mathbf{s} - \mathbf{y}\|$, isto é, $\hat{\mathbf{s}}$ é o ponto do código esférico que está mais próximo de \mathbf{s} . Para esse propósito pode-se utilizar um algoritmo de decodificação, como o proposto em [2] para códigos em camadas de toros.

A decodificação de \mathbf{x} consiste simplesmente em computar $\hat{\mathbf{x}} = \hat{g}\hat{\mathbf{s}}$ e o erro do processo é medido em termos de MSE.

O esquema descrito acima pode ser resumido no Algoritmo 1.

Algoritmo 1: Quantizador *shape/gain* para $C_T(2k, d)$

Entrada: $\mathbf{x} \in \mathbb{R}^k$
Saída: $\hat{\mathbf{x}}$

- 1 **início**
- 2 Compute $g = \|\mathbf{x}\|$ e $\mathbf{s} = \frac{\mathbf{x}}{g}$.
- 3 Use o dicionário do *gain* para quantizar o escalar g como \hat{g} .
- 4 Encontre o índice ξ que identifica o toro $T_{C_s^\xi}$ mais próximo de \mathbf{s} .
- 5 Gere as coordenadas de \mathbf{s} dentro da caixa $P_{C_s^\xi}$ e encontre o ponto \mathbf{a} mais próximo no reticulado escolhido.
- 6 Calcule $\hat{\mathbf{s}} = \Phi_{C_s^\xi}(\mathbf{a})$.
- 7 Obtenha o vetor quantizado: $\hat{\mathbf{x}} = \hat{g}\hat{\mathbf{s}}$.
- 8 **fim**

IV. RESULTADOS COMPUTACIONAIS

Nesta seção apresentamos resultados computacionais obtidos com a proposta do quantizador vetorial esférico apresentado neste artigo.

O dicionário do quantizador vetorial proposto foi otimizado experimentalmente e sua performance foi calculada com base em 504.000 variáveis gaussianas aleatórias independentes e identicamente distribuídas (iid) divididas em blocos de 10.500 vetores de tamanho 48. Para decodificação do *shape*, implementamos o decodificador de distância-limitada para o reticulado Leech apresentado em [9], [11].

A Tabela I apresenta uma comparação entre o SNR médio do quantizador proposto neste trabalho com resultados apresentados na Tabela III do artigo [3] e o limitante de Shannon.

A Figura 4 apresenta um gráfico SNR (em dB) em função da taxa (em bits/dimensão) para quantizadores C_T construídos

TABELA I

TABELA DE COMPARAÇÃO DO QUANTIZADOR $C_T(48, d)$ COM OS RESULTADOS DESCRITOS NA TABELA III DO ARTIGO [3]

Método	bits/dimensão				
	2.5	3	4	5	5.5
Limitante de Shannon	15.05	18.06	24.08	30.10	33.11
QV usando $C_T(48, d)$	14.61	17.70	23.43	29.74*	32.90
QV usando $W_{\Lambda_{24}}$	14.19*	17.36	23.33	29.29	32.28*
QE Lloyd-Max	11.96*	14.62	20.22	26.02	28.95*
QE uniforme	11.76*	14.27	19.38	24.57	27.20*

* Valores estimados.

com base na família de códigos descrita em (5). A medida que o valor do parâmetro t aumenta, a taxa do *shape* também aumenta e o valor do SNR cresce, acompanhando o limitante de Shannon até atingir um platô assintótico em torno de 21dB.

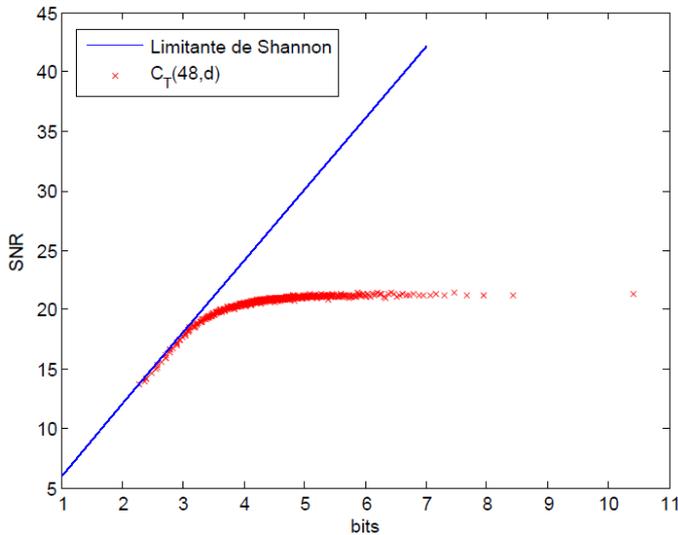


Fig. 4. Distorção em SNR para quantizadores C_T construídos com base na família de códigos descrita em (5) para $0.3536 < t < 0.7071$.

V. CONCLUSÕES

Neste trabalho descrevemos um quantizador vetorial, utilizando a técnica de quantização *shape/gain*, para códigos esféricos em camadas de toros. Este quantizador obteve desempenho superior ao quantizador $W_{\Lambda_{24}}$ proposto em [3], que é considerado uma referência em quantização vetorial de fontes gaussianas através de códigos esféricos.

Além do ganho em termos de distorção, o quantizador apresentado neste trabalho apresenta vantagens em termos de complexidade computacional, pois permite decodificar vetores do \mathbb{R}^{48} utilizando o mesmo decodificador para o reticulado Leech usado em [3], sendo essa a etapa mais cara do processo de decodificação. No entanto, o código esférico $W_{\Lambda_{24}}$, utilizado em [3], possui dimensão igual a 25, ou seja, é possível atingirmos complexidade computacional similar ao $W_{\Lambda_{24}}$ para quantizar em \mathbb{R}^{48} o que pode significar uma importante vantagem em muitas aplicações.

REFERÊNCIAS

- [1] Torezzan, C.; Costa, S.I.R.; Vaishampayan, V.A., *Spherical codes on torus layers*, 2009 IEEE International Symposium on Information Theory, Seoul, 2009, pp. 2033-2037.
- [2] Torezzan, C.; Costa, S.I.R.; Vaishampayan, V.A., *Constructive Spherical Codes on Layers of Flat Tori*. Information Theory, IEEE Transactions on , vol.59, no.10, pp.6655,6663, Oct. 2013.
- [3] Hankins, J. *Design and Analysis of Spherical Codes*. PhD thesis, University of Illinois, 1996.
- [4] Hamkins, J. Zeger, K.. *Gaussian Source Coding With Spherical Codes*. IEEE Trans. Inf. Theory, vol. 48, pp. 2980-2989, Nov. 2002.
- [5] J. H. Conway, N. J. A. Sloane, and E. Bannai. *Sphere-Packings, Lattices, and Groups*. Springer-Verlag New York, 1987. Inc., New York, NY, USA.
- [6] Gersho, A. Gray, R. M., *Vector quantization and signal compression*. Boston: Kluwer Academic Publishers, 2001.
- [7] Utkovski, Z. Utkovski, A *Master Thesis: A Spherical Quantizer for Gaussian Sources*. Chalmers University of Technology, 2004.
- [8] Boaventura, F. M. *Dissertação de mestrado: Quantização Vetorial utilizando Códigos Esféricos*. Universidade Estadual de Campinas, 2015.
- [9] J. H. Conway and N. J. A. Sloane. *Soft decoding techniques for codes and lattices, including the Golay code and the Leech lattice*. IEEE Trans. on Information Theory, IT-32:41-50, 1986.
- [10] G. D. Forney Jr. *Coset codes-part II: binary lattices and related codes*. IEEE Trans. on Information Theory, IT-34:1152-1187, September 1988.
- [11] G. D. Forney Jr. *A bounded-distance decoding algorithm for the Leech lattice, with generalizations*. IEEE Trans. on Information Theory, IT-35:906-909, July 1989.
- [12] Y. Be'er, B. Shahar, e J. Snyders. *Fast decoding of the Leech lattice*. *IEEE Selected Areas on Communications*, 7:959-967, August 1989.
- [13] Shannon, C. E. *Probability of error for optimal codes in a Gaussian channel*, in *The Bell System Technical Journal*, vol. 38, no. 3, pp. 611-656, May 1959.
- [14] Naves, L.R.B., Torezzan, C., e Costa, S.I.R.. (2013). Construção e análise de códigos esféricos com boas taxas binárias. *TEMA (São Carlos)*, 14(1), 13-22. <https://dx.doi.org/10.5540/tema.2013.014.01.0013>