Designing Voronoi Constellations to Minimize Bit Error Rate

Citation for the original published paper (version of record):
Li, S., Mirani, A., Karlsson, M. et al (2021). Designing Voronoi Constellations to Minimize Bit Error Rate. IEEE International Symposium on Information Theory - Proceedings, 2021-July: 1017-1022.
http://dx.doi.org/10.1109/ISIT45174.2021.9517815

Designing Voronoi Constellations to Minimize Bit Error Rate

Shen Li*, Ali Mirani†, Magnus Karlsson†, and Erik Agrell*
*Dept. of Electrical Engineering, Chalmers University of Technology, 412 96 Gothenburg, Sweden
†Dept. of Microtechnology and Nanoscience, Chalmers University of Technology, 412 96 Gothenburg, Sweden
Email: {shenl,mirani,magnus.karlsson,agrell}@chalmers.se

***Abstract*—In a classical 1983 paper, Conway and Sloane presented fast encoding and decoding algorithms for a special case of Voronoi constellations (VCs), for which the shaping lattice is a scaled copy of the coding lattice. Feng generalized their encoding and decoding methods to arbitrary VCs. Less general algorithms were also proposed by Kurkoski and Ferdinand, respectively, for VCs with some constraints on their coding and shaping lattices. In this work, we design VCs with a cubic coding lattice based on Kurkoski's encoding and decoding algorithms. The designed VCs achieve up to 1.03 dB shaping gains with a lower complexity than Conway and Sloane's scaled VCs. To minimize the bit error rate (BER), pseudo-Gray labeling of constellation points is applied. In uncoded systems, the designed VCs reduce the required SNR by up to 1.1 dB at the same BER, compared with the same VCs using Feng's and Ferdinand's algorithms. In coded systems, the designed VCs are able to achieve lower BER than the scaled VCs at the same SNR. In addition, a Gray penalty estimation method for such VCs of very large size is introduced.**

I. Introduction

Signal shaping refers to adjusting the distribution of the transmitted signal alphabet to the capacity-achieving distribution, in order to have a more power-efficient modulation format for a given channel. It is well-known that the capacity-achieving distribution for an average power constrained additive white Gaussian noise (AWGN) channel is the Gaussian distribution, and its gain over a uniform distribution (the commonly used rectangular modulation formats such as a quadrature amplitude modulation (QAM)) is 1.53 dB at asymptotically high spectral efficiencies. Much effort was done to design constellations with a near-Gaussian distribution and feasible complexity to reduce this gap, which is typically categorized into geometric shaping [1], [2] and probabilistic shaping [3], [4], from two-dimensional cases [5]–[7], to higher dimensional cases [8]. Among them, Voronoi constellations (VCs), as a structured geometric shaping method, are attractive as modulation alphabets or vector quantizers, due to their properties in terms of minimum distance and average power. Moreover, they do not require look-up tables for searching positions of constellation points, which geometric shaping based on spherical constellations does [9], [10], and unlike probabilistic shaping where a Gaussian-like distribution matcher is needed [11], its complexity does not increase rapidly with the constellation dimension.

Conway and Sloane defined VCs in 1983 by selecting a finite set of points of a lattice (possibly translated) that belong to a scaled-up version of its Voronoi region [12]. They presented simple and elegant algorithms for encoding and decoding VCs, i.e., mapping integers to constellation points and vice versa. Mirani *et al.* designed multidimensional scaled VCs with up to 10^{28} constellation points utilizing these algorithms for the AWGN channel and nonlinear fiber channel, and showed significant bit error rate (BER) gains over QAM in uncoded systems [13].

Forney generalized the concept in 1989 by considering two possibly different lattices, a coding lattice and a shaping lattice. The VCs are constructed by selecting the points of the (translated) coding lattice that belong to a the Voronoi region of the shaping lattice [14]. The only constraint is that the shaping lattice is a sublattice of the coding lattice. A method to enumerate the constellation points in an arbitrary VC was presented by Feng *et al.* [15]. This enumeration admits very fast encoding and decoding algorithms, which are reviewed by Zamir in [16, Ch. 9], [17].

Kurkoski proposed encoding and decoding algorithms called "rectangular encoding" in [18] for VCs whose shaping lattice and coding lattice both have triangular generator matrices. These algorithms are equally simple but less general than Feng's algorithms, and are applicable to a variety of coding lattices and shaping lattices.

Ferdinand *et al.* proposed a two-step "systematic Voronoi shaping" method in [19], based on the concept of "systematic shaping" proposed by Sommer *et al.* in [20]. A coding lattice defined by a lower-triangular parity-check matrix and a shaping lattice satisfying certain constraints related to the coding lattice were combined to achieve high coding and shaping gains, and the symbol error rate (SER) performance was evaluated. For the shaping step, the algorithms to map the integers to points in VCs with a cubic coding lattice and vice versa were explicitly described in [21].

As far as we know, no study apart from the one by Mirani *et al.* [13] has been reported about designing VCs with good BER performance and low complexity. Hence, in this paper, we focus on the VCs with a cubic coding lattice and try to minimize the BER both in uncoded and coded systems. Although such VCs have no coding gain, the high shaping gain is maintained, and their decoding is much simpler than for VCs with rescaled coding and shaping lattices [12], [13]. They also allow more flexibility in spectral efficiencies for mapping integers to bits. The considered VCs in this paper are very large, with up to 9×10^{46} points, but they are nevertheless useful in communications with moderate complexity, since none of the involved algorithms need to store or search all constellation points. We apply pseudo-Gray labeling on top of Kurkoski's encoding and decoding. Our designed VCs show better BER performance than Feng's and Ferdinand's algorithms for the same VCs in uncoded systems. Also, in combination with a low-density parity-check (LDPC) code, the designed VCs can have better BER performance than the scaled VCs, due to the efficient pseudo-Gray labeling for the cubic coding lattice.

Notation: Bold lowercase symbols denote row vectors and bold uppercase symbols denote matrices. The elements of a vector $\boldsymbol{u}$ are denoted by u_i, the rows of a matrix $\boldsymbol{P}$ are denoted by $\boldsymbol{p}_i$, and the elements of a matrix $\boldsymbol{P}$ are denoted by P_{ij}. Uppercase Greek or calligraphic letters denote sets.

II. Preliminaries

Given a set of n linearly independent *basis vectors,* a *lattice* is the set of all linear combinations of these vectors with integer coefficients. If the basis vectors are arranged row-wise into a matrix $\boldsymbol{G}$, then the lattice is

$$\Lambda \triangleq \{\boldsymbol{u}\boldsymbol{G} :\ \boldsymbol{u} \in \mathbb{Z}^n\}. \tag{1}$$

Without loss of generality, we assume that the generator matrix has dimension $n \times n$. From the definition, any lattice includes the all-zero vector $\mathbf{0}$.

The generator matrix of a given lattice is not unique. Two generator matrices $\boldsymbol{G}$ and $\boldsymbol{G}'$ generate the same lattice if and only if $\boldsymbol{G}' = \boldsymbol{U}\boldsymbol{G}$, where $\boldsymbol{U}$ is an integer matrix with determinant ± 1 [22, p. 10].

The *fundamental Voronoi region* of a lattice Λ is the set of vectors in Euclidean space having the all-zero vector as its closest lattice point, i.e.,

$$\Omega(\Lambda) \triangleq \{\boldsymbol{x} \in \mathbb{R}^n :\ \|\boldsymbol{x}\|^2 \leq \|\boldsymbol{x} - \boldsymbol{\lambda}\|^2,\ \forall \boldsymbol{\lambda} \in \Lambda\}. \tag{2}$$

Given an n-dimensional coding lattice Λ, an n-dimensional shaping lattice Λ_s which is a sublattice of Λ, and an offset vector $\boldsymbol{a} \in \mathbb{R}^n$, a *Voronoi constellation* (VC) in its general form defined by Forney [14] is

$$\Gamma \triangleq (\Lambda - \boldsymbol{a}) \cap \Omega(\Lambda_\mathrm{s}). \tag{3}$$

We assume that no points in $\Lambda - \boldsymbol{a}$ fall on the boundary of $\Omega(\Lambda_\mathrm{s})$.[1] The number of points in the VC is

$$M \triangleq |\Gamma| = \frac{|\det \boldsymbol{G}_\mathrm{s}|}{|\det \boldsymbol{G}|}, \tag{4}$$

where $\boldsymbol{G}_\mathrm{s}$ is a generator matrix of Λ_s. This relation can be verified by recognizing $|\det \boldsymbol{G}|$ and $|\det \boldsymbol{G}_\mathrm{s}|$ as the volumes of $\Omega(\Lambda)$ and $\Omega(\Lambda_\mathrm{s})$, respectively [22, p. 4].

Fig. 1 illustrates a two-dimensional VC of lattice partition $\mathbb{Z}^2/2D_2$, where D_2 is the two-dimensional checkerboard lattice. In this simple example, we can choose the generator matrices of Λ and Λ_s to be

$$\boldsymbol{G} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \boldsymbol{G}_\mathrm{s} = \begin{bmatrix} 4 & 0 \\ 2 & 2 \end{bmatrix}, \tag{5}$$

and there are $M = 8$ constellation points, with $\boldsymbol{a} = (-\frac{1}{2}, 0)$. The offset vector $\boldsymbol{a}$ is optimized to minimize the average symbol energy of the constellation, and can be obtained using an iterative algorithm given by [12]. The algorithm may converge to a suboptimal vector for large-size constellations, but as the constellation size increases, the performance difference between VC generated using optimal $\boldsymbol{a}$ and a random $\boldsymbol{a} \in \Omega(\Lambda)$ decreases, and can be neglected for large VCs [13, Fig. 3]. In this paper, for a small or moderate-size VC ($M \leq 2^{17} \approx 1.3 \times 10^5$), $\boldsymbol{a}$ was optimized using the method in [12]; and for a very large VC where we can only approximate the average symbol energy by Monte Carlo simulations, we selected a random $\boldsymbol{a}$ uniformly in $\Omega(\Lambda)$.

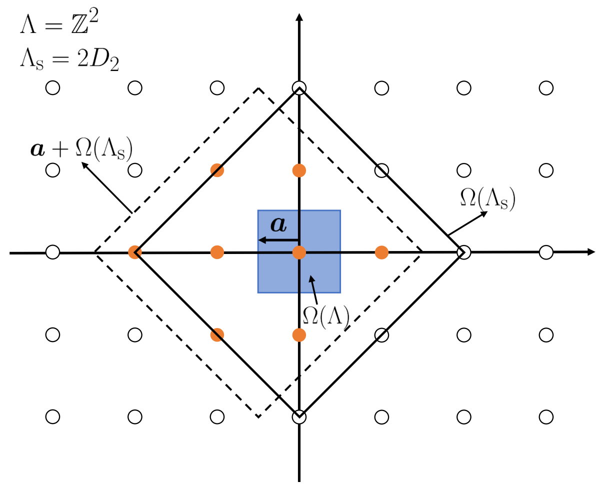

Fig. 1: Example: a two-dimensional VC of the lattice partition $\mathbb{Z}^2/2D_2$, where $\Omega(\Lambda)$ is the Voronoi region of the coding lattice $\mathbb{Z}^2$ and $\Omega(\Lambda_\mathrm{s})$ is the Voronoi region of the shaping lattice $2D_2$.

III. Encoding and Decoding

We adopt Kurkoski's algorithms for encoding and decoding, and also review Feng's algorithms in this section.

Feng's algorithms: Given Λ and its sublattice Λ_s with their generator matrices $\boldsymbol{G}$ and $\boldsymbol{G}_\mathrm{s}$, respectively, since all basis vectors of Λ_s belong to Λ, there exists by definition (1) an integer matrix $\boldsymbol{U}$ such that $\boldsymbol{G}_\mathrm{s} = \boldsymbol{U}\boldsymbol{G}$, for any choice of Λ and Λ_s. Then $\boldsymbol{U}$ has a Smith normal form [23, Ch. 15]

$$\boldsymbol{J} \triangleq \mathrm{diag}(J_{11}, \ldots, J_{nn}) = \boldsymbol{S}\boldsymbol{U}\boldsymbol{T}, \tag{6}$$

where $\boldsymbol{S}$ and $\boldsymbol{T}$ are integer matrices with determinant ± 1 and $J_{ii} \in \mathbb{Z}^+$ for all $i = 1, \ldots, n$. Then alternative generator matrices of Λ and Λ_s can be constructed as $\boldsymbol{G}' = T^{-1}\boldsymbol{G}$ and $\boldsymbol{G}'_\mathrm{s} = S\boldsymbol{G}_\mathrm{s}$, with the relation [15, Th. 6]

$$\boldsymbol{G}'_\mathrm{s} = \boldsymbol{J}\boldsymbol{G}'. \tag{7}$$

[1] An arbitrarily small perturbation of the offset $\boldsymbol{a}$ in a random direction prevents points from falling on the boundary with probability one.

Algorithm 1 Feng's encoding [15]

Input: $\boldsymbol{u}$. Output: $\boldsymbol{c}$.

Preprocessing: Given generator matrices $\boldsymbol{G}$ and $\boldsymbol{G}_{\mathrm{s}}$, find two integer matrices $\boldsymbol{S}$ and $\boldsymbol{T}$ with determinant ± 1 such that $\boldsymbol{J} = \boldsymbol{S}\boldsymbol{G}_{\mathrm{s}}\boldsymbol{G}^{-1}\boldsymbol{T}$ is the Smith normal form of $\boldsymbol{G}_{\mathrm{s}}\boldsymbol{G}^{-1}$. Then let $\boldsymbol{G}' = \boldsymbol{T}^{-1}\boldsymbol{G}$.

Let $\boldsymbol{x} \leftarrow \boldsymbol{u}\boldsymbol{G}' - \boldsymbol{a}$
Let $\boldsymbol{z} \leftarrow \arg\min_{\boldsymbol{\lambda}\in\Lambda_{\mathrm{s}}} \|\boldsymbol{x} - \boldsymbol{\lambda}\|^2$
Let $\boldsymbol{c} \leftarrow \boldsymbol{x} - \boldsymbol{z}$

Algorithm 2 Feng's decoding [15]

Input: $\boldsymbol{y} \in \mathbb{R}^n$, which is a noisy version of $\boldsymbol{c}$. Output: $\boldsymbol{u}$.

Preprocessing: as in Algorithm 1.

Let $\boldsymbol{x} \leftarrow \arg\min_{\boldsymbol{\lambda}\in\Lambda} \|\boldsymbol{y} + \boldsymbol{a} - \boldsymbol{\lambda}\|^2$
Let $\boldsymbol{u} \leftarrow \boldsymbol{x}\boldsymbol{G}'^{-1}$
Let $u_i \leftarrow u_i \mod J_{ii}$, $\forall i = 1, \dots, n$

Algorithm 3 Kurkoski's encoding [18]

Input: $\boldsymbol{u}$. Output: $\boldsymbol{c}$.

Let $\boldsymbol{x} \leftarrow \boldsymbol{u}\boldsymbol{G} - \boldsymbol{a}$
Let $\boldsymbol{z} \leftarrow \arg\min_{\boldsymbol{\lambda}\in\Lambda_{\mathrm{s}}} \|\boldsymbol{x} - \boldsymbol{\lambda}\|^2$
Let $\boldsymbol{c} \leftarrow \boldsymbol{x} - \boldsymbol{z}$

Algorithm 4 Kurkoski's decoding [18]

Input: $\boldsymbol{y} \in \mathbb{R}^n$, which is a noisy version of $\boldsymbol{c}$. Output: $\boldsymbol{u}$.

Preprocessing: Given lower-triangular $\boldsymbol{G}$ and $\boldsymbol{G}_{\mathrm{s}}$, let $\boldsymbol{L} = \boldsymbol{G}_{\mathrm{s}}\boldsymbol{G}^{-1}$.

Let $\boldsymbol{x} \leftarrow \arg\min_{\boldsymbol{\lambda}\in\Lambda} \|\boldsymbol{y} + \boldsymbol{a} - \boldsymbol{\lambda}\|^2$
Let $\boldsymbol{u} \leftarrow \boldsymbol{x}\boldsymbol{G}^{-1}$
for $i = n, n-1, \dots, 1$ **do**
    $v_i \leftarrow \lfloor u_i / L_{ii} \rfloor$
    $\boldsymbol{u} \leftarrow \boldsymbol{u} - v_i \boldsymbol{l}_i$
end for

Every point $\boldsymbol{c} \in \Gamma$ can be uniquely enumerated by vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ such that

$$\boldsymbol{c} = \boldsymbol{u}\boldsymbol{G}' + \boldsymbol{v}\boldsymbol{G}'_{\mathrm{s}} - \boldsymbol{a}, \tag{8}$$

where $u_i \in \{0, \dots, J_{ii} - 1\}$ for $i = 1, \dots, n$ and $\boldsymbol{v} \in \mathbb{Z}^n$. There are $M = \det \boldsymbol{J} = \prod_i J_{ii}$ possible values of $\boldsymbol{u}$, and each of them occurs exactly once among all points $\boldsymbol{c} \in \Gamma$. Hence, $\boldsymbol{u}$ is used to label $\boldsymbol{c}$, regardless of $\boldsymbol{v}$.

In encoding, $\boldsymbol{u}$ and $\boldsymbol{a}$ in (8) are known, and the unique value of $\boldsymbol{z} = -\boldsymbol{v}\boldsymbol{G}'_{\mathrm{s}}$ that fulfills $\boldsymbol{c} \in \Gamma$ is found. In decoding, given $\boldsymbol{c}$ and $\boldsymbol{a}$,

$$\boldsymbol{u} = (\boldsymbol{c} + \boldsymbol{a})\boldsymbol{G}'^{-1} - \boldsymbol{v}\boldsymbol{J}, \tag{9}$$

which can be solved by setting u_i to the i^{th} element of $(\boldsymbol{c} + \boldsymbol{a})\boldsymbol{G}'^{-1}$ modulo J_{ii}. If the decoder input vector is noisy, then it is first rounded to the nearest point in the translated lattice $\Lambda - \boldsymbol{a}$. The encoding and decoding are summarized in Algorithm 1 and 2.

The $\arg\min$ operations are carried out by an algorithm to find the closest point in a given lattice, which is a well-studied problem. Specific algorithms are available for many common lattices [22, Ch. 20], [24], [25], and other lattices can be handled by general algorithms [26].

Kurkoski's algorithms: Given Λ and its sublattice Λ_{s}, Kurkoski's algorithms are applicable only when Λ and Λ_{s} both have triangular generator matrices $\boldsymbol{G}$ and $\boldsymbol{G}_{\mathrm{s}}$, respectively. Conventionally, lower-triangular $\boldsymbol{G}$ and $\boldsymbol{G}_{\mathrm{s}}$ are used. Then the diagonal elements of the lower-triangular integer matrix $\boldsymbol{L} = \boldsymbol{G}_{\mathrm{s}}\boldsymbol{G}^{-1}$ can be used to enumerate the $M = \prod_i L_{ii}$ constellation points as

$$\boldsymbol{c} = \boldsymbol{u}\boldsymbol{G} + \boldsymbol{v}\boldsymbol{G}_{\mathrm{s}} - \boldsymbol{a}, \tag{10}$$

where $u_i \in \{0, \dots, L_{ii} - 1\}$ for $i = 1, \dots, n$ and $\boldsymbol{v} \in \mathbb{Z}^n$.

The encoding is similar to Algorithm 1, but there is no need to calculate $\boldsymbol{G}'$. In decoding, given $\boldsymbol{c}$ and $\boldsymbol{a}$, (10) is solved for $\boldsymbol{u}$. This can be done sequentially, beginning from u_n, thanks to the triangular structure of $\boldsymbol{L}$. Specifically, the algorithms operate as in Algorithm 3 and 4. If $\boldsymbol{L} = r\boldsymbol{I}$ for an identity matrix $\boldsymbol{I}$ and a positive integer r, then the algorithms specialize into the classical encoding and decoding algorithms [12].

Specifically for the application of Kurkoski's algorithms to VCs with $\Lambda = \mathbb{Z}^n$, different triangular generator matrices $\boldsymbol{G}$ result in different encoding. It is important that $\boldsymbol{G}$ is set to $\boldsymbol{I}_n$ for pseudo-Gray labeling, where $\boldsymbol{I}_n$ is an n-dimensional identity matrix, which is discussed in section V.

Ferdinand *et al.* proposed equally simple but less general encoding and decoding algorithms than Feng's and Kurkoski's specifically for VCs with a cubic coding lattice, which is applicable for most commonly used shaping lattices, e.g., the 4-dimensional checkerboard lattice D_4, 8-dimensional lattice E_8, 16-dimensional Barnes–Wall lattice Λ_{16}, and the 24-dimensional Leech lattice Λ_{24} [22, Ch. 4]. Ferdinand's algorithms can have the same mapping rule as Feng's, when the generator matrices of shaping lattices are written in nice lower-triangular matrices as in [22, Ch. 4]. For the explicit algorithms, see [21].

IV. Voronoi Constellations with Cubic Coding Lattice

In this section, we study a specific kind of VC, whose coding lattice $\Lambda = \mathbb{Z}^n$. One reason why this kind of VC is interesting is that the search for the closest lattice point in decoding (the $\arg\min$ operation in Algorithm 2 and 4), which usually dominates the complexity for the encoding and decoding, especially for high-dimensional lattices, is just to round the received noisy vector to an integer vector. With this trivial search algorithm, the decoding is a set of very low-complexity linear operations.

Apart from its simplicity in decoding, this VC maintains the asymptotic shaping gain. To see this, two figures of merit that are useful for comparing different modulation formats are considered as in [8], [27]: the *spectral efficiency* $SE = 2\log_2(M)/n$ bits/symbol/dimension-pair and the *power efficiency* $PE = d_{\text{min}}^2 \log_2(M)/(4E_{\mathrm{s}})$, where d_{min} is the minimum Euclidean distance of the constellation, and

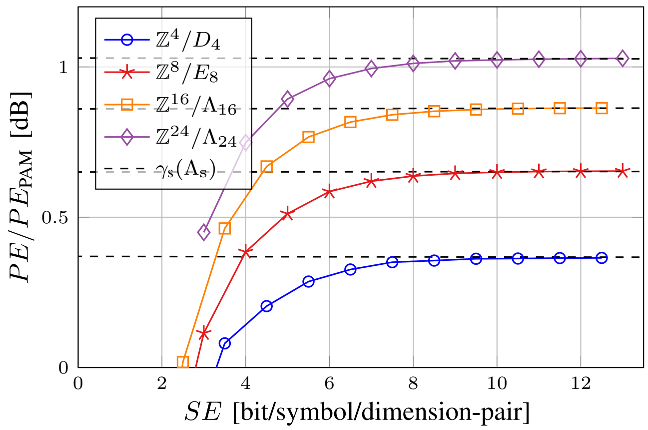

Fig. 2: The PE/PE_{PAM} as a function of SE for VCs with cubic coding lattice. The black dashed lines are the asymptotic shaping gains $\gamma_s(\Lambda_s)$ for these shaping lattices stated in [14, Table I].

$E_s = (1/M)\sum_{\boldsymbol{c}\in\Gamma}\|\boldsymbol{c}\|^2$ is the average symbol energy. Then the power efficiency of the one-dimensional pulse amplitude modulation (PAM) is $PE_{\text{PAM}} = 3SE/(2(2^{SE}-1))$. The ratio PE/PE_{PAM} in dB is the gain that a constellation can obtain over cubic constellations. At high spectral efficiencies, the PE/PE_{PAM} ratio of VCs of the lattice partition $\mathbb{Z}^n/\Lambda_s$ should converge to the asymptotic shaping gain $\gamma_s(\Lambda_s) = 1/(12G(\Omega(\Lambda_s)))$ defined in [8], where $G(\Omega(\Lambda_s))$ is the normalized second moment of the Voronoi region $\Omega(\Lambda_s)$ [28, Eq. (9)]. Forney listed $G(\Omega(\Lambda_s))$ and $\gamma_s(\Lambda_s)$ of Voronoi regions of some classical multidimensional shaping lattices in [14, Table I]. In Fig. 2, we present PE/PE_{PAM} as a function of SE for VCs with a cubic coding lattice and D_4, E_8, Λ_{16}, and Λ_{24} as shaping lattices.

In addition, the separate shaping lattice and coding lattice allow for improved granularity in spectral efficiencies than the scaled VC, which gives us more flexibility in choosing different data rates as needed.

V. Labeling of Constellation Points

The Gray penalty G_p of a constellation was defined as the average number of different bits per pair of adjacent symbols [30], [31]. It has been extensively used in the literature [32]–[34], since it predicts the asymptotic ratio of the BER P_b and SER P_s for the AWGN channel:

$$P_b = \frac{G_p}{\log_2(M)} P_s. \tag{11}$$

Clearly $G_p \geq 1$ and we want G_p to be close to 1. Gray penalty is a good measure for selecting labeling schemes for VCs. However, its calculation requires enumeration of all constellation points, which is infeasible for very large VCs.

We propose a new method to accurately estimate G_p for constellations too large to enumerate. With the cubic coding lattice, each symbol $\boldsymbol{c} \in \Gamma$ (except those on the boundary of the constellation) has $2n$ neighbors, forming a set $nbr(\boldsymbol{c}) \triangleq \{\boldsymbol{h} : \|\boldsymbol{h}-\boldsymbol{c}\|^2 = 1,\ \boldsymbol{h}\in\mathbb{Z}^n\}$, which can be easily enumerated by adding all permutations and sign changes of the n-tuple $(\pm 1, 0, \ldots, 0)$ to $\boldsymbol{c}$. Thus G_p can be estimated by Monte Carlo simulations for very large M as follows. We define $enc(\cdot)$ as the encoding function performing Algorithm 1 or 3, $dec(\cdot)$ as the decoding function performing Algorithm 2 or 4, and $map(\cdot)$ as the mapping function converting each element u_i of an integer input vector $\boldsymbol{u} = (u_1, u_2, \ldots, u_n)$ to a binary vector of length $\log_2(J_{ii})$ (or $\log_2(L_{ii})$) for all $i = 1, 2, \ldots, n$ using a one-dimensional natural binary code (NBC) or binary reflected Gray code (BRGC), and concatenating them together to a binary vector of length $\log_2(M)$ in total. The explicit estimating process of G_p is described in Algorithm 5, where d_H denotes the Hamming distance between two binary vectors.

Algorithm 5 Gray penalty estimation

Input: Generate N_s random integer vectors $\boldsymbol{u}_r$, $r = 1, 2, \ldots, N_s$, of which where each element $u_{r,i}$ is uniformly distributed in $\{0, 1, \ldots, J_{ii}-1\}$ for all $i = 1, 2, \ldots n$ (replace J_{ii} with L_{ii} if Algorithm 4 is used). Output: Estimated Gray penalty G_p.

```
counter_1 = 0
counter_2 = 0
for r = 1, 2, ..., N_s do
    c ← enc(u_r)
    H ← nbr(c)
    for h ∈ H do
        x ← argmin_{λ∈Λ_s} ||h − λ||^2
        if x ≠ 0 then  remove h from H
    end for
    counter_1 ← counter_1 + |H|
    m ← map(u_r)
    K ← {dec(c) : c ∈ H}
    B ← {map(u) : u ∈ K}
    counter_2 ← counter_2 + Σ_{b∈B} d_H(m, b)
end for
G_p = counter_1/counter_2
```

In Table I, we list the estimated G_p values for very large multidimensional VCs, when Feng's and Kurkoski's encoding and decoding algorithms and both NBC and BRGC are used. For the constellations with $M \leq 2^{17}$, we validated that using $N_s = 10^4$ symbols, Algorithm 5 estimates the Gray penalties accurately with an error less than 0.5% compared with the exact G_p values calculated by enumerating all constellation points. The reason why Kurkoski's algorithms always yield a smaller G_p is that the generator matrix $\boldsymbol{G}$ of the coding lattice $\mathbb{Z}^n$ is $\boldsymbol{I}_n$, whereas when Feng's algorithms are used, in order to fullfil (7) when D_4, E_8, Λ_{16}, and Λ_{24} are used as shaping lattices, $\boldsymbol{G}'$ has to be a non-identity matrix, which will change the natural or Gray ordering of $\boldsymbol{u}$ after multiplication. Specifically in this paper, we write $\boldsymbol{G}_s = \boldsymbol{G}'_s$ as lower-triangular matrices [22, Ch. 4], then $\boldsymbol{G}'$ must also be lower-triangular. For the example VC in Fig. 1, in Feng's encoding,

$$\boldsymbol{G}' = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}. \tag{12}$$

TABLE I: Estimated Gray penalties of VCs of lattice partitions $\mathbb{Z}^n/\Lambda_\mathrm{s}$ at high spectral efficiencies.

$\mathbb{Z}^n/\Lambda_\mathrm{s}$	$\mathbb{Z}^4/D_4$	$\mathbb{Z}^8/E_8$	$\mathbb{Z}^{16}/\Lambda_{16}$	$\mathbb{Z}^{24}/\Lambda_{24}$
SE [bit/sym/pol]	12.5	12	11.5	13
Feng's NBC	3.46	5.36	8.88	13.13
Kurkoski's NBC	1.98	2.01	2.04	2.07
Feng's BRGC	1.75	2.81	4.86	7.40
Kurkoski's BRGC	1.02	1.08	1.16	1.17

Similarly, in Ferdinand's encoding, which is equivalent to Feng's encoding for the considered VCs in Table I, $\boldsymbol{u}$ also needs to multiply with a non-identity matrix, see the example in [21, Eq. (14)]. Ferdinand's decoding is equivalent to Kurkoski's decoding, and yields the same G_p as Feng's algorithms. As the constellation dimension increases, their G_p grows rapidly with the number of dimensions. However, with Kurkoski's algorithms, G_p stays steady at different dimensions both for NBC and BRGC.

According to Table I, Kurkoski's algorithms are expected to reduce the BER by a factor up to 6.3 compared with Feng's and Ferdinand's algorithms. Again, we implemented Ferdinand's algorithms, and find the same BER performance as Feng's. In Fig. 3, we illustrate the required E_b/N_0 gain (in dB) of Kurkoski's algorithms over Feng's as a function of SE, at a BER of 10^{-4}, where $E_\mathrm{b} = E_\mathrm{s}/\log_2(M)$ is the energy per bit and N_0 is the noise variance of the Gaussian noise per two dimensions. As n and SE increase, higher gains are obtained both using NBC and BRGC. The curves with BRGC have a larger slope than NBC, and reach up to 1.1 dB for $\mathbb{Z}^{24}/\Lambda_{24}$ at 13 bits per symbols per dimension-pair. Higher gains are expected if we further increase n and SE.

We also investigate the BER performance of the designed VCs based on Kurkoski's algorithms, with BRGC applied to integer vectors in coded systems for 4-dimensional case. LDPC codes from the digital video broadcasting (DVB-S2) standard [35] with a codeword length of 64800, 50 decoding iterations, and the max-log approximation of the log-likelihood ratio [36, Eq. (6)] is applied to 1) the designed VCs, (2) the same VCs based on Feng's encoding and decoding, and (3) the scaled VCs. Different code rates R_c are used to compare these VCs under almost the same information rate, defined as $R = SE \cdot R_\mathrm{c}$ bits/symbol/dimension-pair. The labeling of the scaled VCs follows the "quasi-Gray labeling" in [13].

Fig. 4 shows 0.8–1 dB E_b/N_0 gains of Kurkoski's algorithms over Feng's at a BER of 10^{-4} after LDPC decoding. The designed VCs also outperform the scaled VCs, which means that the loss of coding gain due to cubic coding lattice is more than compensated by the LDPC code, implying no need for including more complex coding lattices. The better BER performance might come from the pseudo-Gray labeling for the cubic coding lattice, as our VCs have lower Gray penalties than the scaled VCs. Also, the higher spectral efficiencies allow us to use a higher overhead error-correction code.

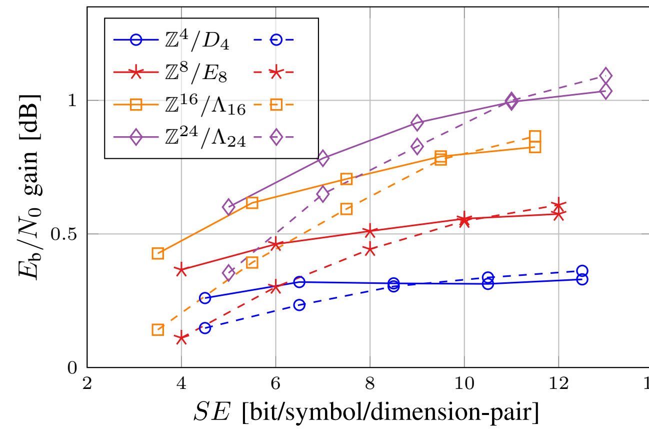

Fig. 3: Required E_b/N_0 gain of Kurkoski's algorithms over Feng's algorithms at a BER of 10^{-4} for VCs with cubic coding lattice and some classical shaping lattices. Solid curves refer to cases where NBC are used, and dashed curves mean BRGC are used.

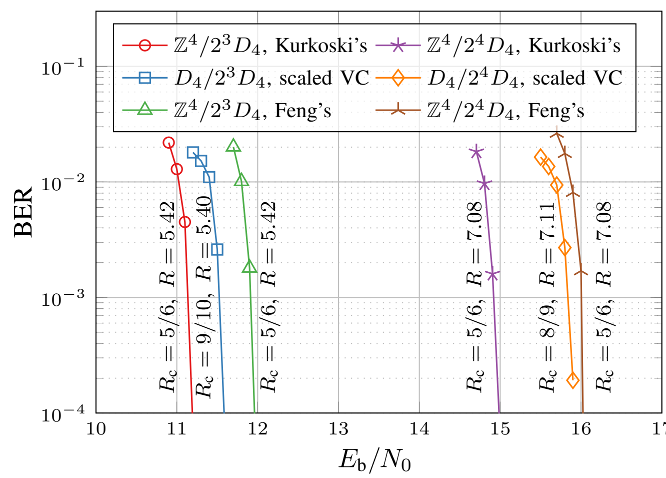

Fig. 4: The BER as a function of E_b/N_0 after LDPC decoding for 4-dimensional VCs.

VI. Conclusion

We designed low-complexity VCs with a cubic coding lattice based on Kurkoski's encoding and decoding algorithms. With the pseudo-Gray labeling, the designed VCs can reduce the BER up to 6.3 times at the same SNR or reduce the required SNR by up to 1.1 dB at the same BER, compared with the two benchmark algorithms in the literature in uncoded systems. The Gray penalty estimation algorithm for large VCs with cubic coding lattice can be used to choose good labeling schemes. In combination with a LDPC code, the designed VCs can have better BER performance with lower decoding complexity compared with the scaled VCs, due to the efficient pseudo-Gray labeling for the cubic coding lattice. In the future, the mutual information of such VCs is a worthy topic to study.

Acknowledgment

This work was supported by the Swedish Research Council (VR) under grant no. 2017–03702.

References

[1] F.-W. Sun and H. C. A. van Tilborg, "Approaching capacity by equiprobable signaling on the Gaussian channel," *IEEE Trans. Inf. Theory,* vol. 39, no. 5, pp. 1714–1716, Sept. 1993.

[2] G. D. Forney, Jr., R. G. Gallager, G. R. Lang, F. M. Longstaff, and S. U. Qureshi, "Efficient modulation for band-limited channels," *IEEE J. Sel. Areas Commun.,* vol. 2, no. 5, pp. 632–647, Sept. 1984.

[3] A. R. Calderbank and L. H. Ozarow, "Nonequiprobable signaling on the Gaussian channel," *IEEE Trans. Inf. Theory,* vol. 36, no. 4, pp. 726–740, July 1990.

[4] F. R. Kschischang and S. Pasupathy, "Optimal nonuniform signaling for Gaussian channels," *IEEE Trans. Inf. Theory,* vol. 39, no. 3, pp. 913–929, May 1993.

[5] E. N. Gilbert, "A comparison of signalling alphabets," *Bell Syst. Tech. J.,* vol. 31, no. 3, pp. 504–522, May 1952.

[6] C. N. Campopiano and B. G. Glazer, "A coherent digital amplitude and phase modulation scheme," *IRE Trans. Commun. Sys.,* vol. 10, no. 1, pp. 90–95, Mar. 1962.

[7] G. J. Foschini, R. D. Gitlin, and S. Weinstein, "Optimization of two-dimensional signal constellations in the presence of Gaussian noise," *IEEE Trans. Commun.,* vol. 22, no. 1, pp. 28–38, Jan. 1974.

[8] G. D. Forney, Jr. and L.-F. Wei, "Multidimensional constellations—part I: Introduction, figures of merit, and generalized cross constellations," *IEEE J. Sel. Areas Commun.,* vol. 7, no. 6, pp. 877–892, Aug. 1989.

[9] Y. C. Gültekin, F. M. J. Willems, W. J. van Houtum, and S. Şerbetli, "Approximate enumerative sphere shaping," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT),* Vail, CO, 2018.

[10] Y. C. Gültekin, T. Fehenberger, A. Alvarado, and F. M. J. Willems, "Probabilistic shaping for finite blocklengths: distribution matching and sphere shaping," *Entropy,* vol. 22. no. 5, paper 581, May 2020.

[11] P. Schulte and G. Böcherer, "Constant composition distribution matching," *IEEE Trans. Inf. Theory,* vol. 62, no. 1, pp. 430–434, Jan. 2016.

[12] J. H. Conway and N. J. A. Sloane, "A fast encoding method for lattice codes and quantizers," *IEEE Trans. Inf. Theory,* vol. IT-29, no. 6, pp. 820–824, Nov. 1983.

[13] A. Mirani, E. Agrell, and M. Karlsson, "Low-complexity geometric shaping," *J. Lightw. Technol.,* vol. 39, no. 2, pp. 363–371, Jan. 2021.

[14] G. D. Forney, Jr., "Multidimensional constellations—part II: Voronoi constellations," *IEEE J. Sel. Areas Commun.,* vol. 7, no. 6, pp. 941–958, Aug. 1989.

[15] C. Feng, D. Silva, and F. R. Kschischang, "An algebraic approach to physical-layer network coding," *IEEE Trans. Inf. Theory,* vol. 59, no. 11, pp. 7576–7596, Nov. 2013.

[16] R. Zamir, *Lattice Coding for Signals and Networks,* Cambridge, UK: Cambridge University Press, 2014.

[17] ——, "How to design an efficient lattice coding scheme," in *Proc. IEEE Inf. Theory Workshop (ITW),* Hobart, Australia, Nov. 2014.

[18] B. M. Kurkoski, "Encoding and indexing of lattice codes," *IEEE Trans. Inf. Theory,* vol. 64, no. 9, pp. 6320–6332, 2018.

[19] N. S. Ferdinand, B. M. Kurkoski, M. Nokleby, and B. Aazhang, "Low-dimensional shaping for high-dimensional lattice codes," *IEEE Trans. Wireless Commun.,* vol. 15, no. 11, pp. 7405–7418, Nov. 2016.

[20] N. Sommer, M. Feder, and O. Shalvi, "Shaping methods for low-density lattice codes," in *Proc. IEEE Inf. Theory Workshop (ITW),* Taormina, Italy, Oct. 2009.

[21] N. S. Ferdinand, B. M. Kurkoski, B. Aazhang, and M. Latva-aho, "Shaping low-density lattice codes using Voronoi integers," in *Proc. IEEE Inf. Theory Workshop (ITW),* Hobart, Australia, 2014.

[22] J. H. Conway and N. J. A. Sloane, *Sphere Packings, Lattices and Groups,* 3rd ed. New York, NY: Springer, 1999.

[23] W. C. Brown, *Matrices over Commutative Rings,* New York: Marcel Dekker, 1993.

[24] J. H. Conway and N. J. A. Sloane, "Fast quantizing and decoding algorithms for lattice quantizers and codes," *IEEE Trans. Inf. Theory,* vol IT-28, no. 2, pp. 227–232, Mar. 1982.

[25] ——, "On the Voronoi regions of certain lattices," *SIAM J. Alg. Disc. Meth.,* vol. 5, no. 3, pp. 294–305, Sept. 1984.

[26] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Trans. Inf. Theory,* vol. 48, no. 8, pp. 2201–2214, Aug. 2002.

[27] M. Karlsson and E. Agrell, "Spectrally efficient four-dimensional modulation," in *Proc. Optical Fiber Communications Conference (OFC),* Los Angeles, CA, 2012.

[28] J. H. Conway and N. J. A. Sloane, "Voronoi regions of lattices, second moments of polytopes, and quantization," *IEEE Trans. Inf. Theory,* vol. 28, no. 2, pp. 211–226, Mar. 1982.

[29] A. Alvarado, T. Fehenberger, B. Chen, and F. M. J. Willems, "Achievable information rates for fiber optics: Applications and computations," *J. Lightw. Technol.,* vol. 36, no. 2, pp. 424—439, 2018.

[30] K. Simon and J. G. Smith, "Hexagonal multiple phase-and-amplitude-shift-keyed signal sets," *IEEE Trans. Commun.,* vol. 21, no. 10, pp. 1108—1115, 1973.

[31] J. G. Smith, "Odd-bit quadrature amplitude-shift keying," *IEEE Trans. Commun,* vol. 23, no. 3, pp. 385–389, Mar. 1975.

[32] P. K. Vitthaladevuni, M.-S. Alouini, and J. C. Kieffer, "Exact BER computation for cross QAM constellations," *IEEE Trans. Wireless Commun.*, vol. 4, no. 6, pp. 3039–3050, Nov. 2005.

[33] J. Leibrich and W. Rosenkranz, "Multidimensional constellations for power-efficient and flexible optical networks," *IEEE Photon. Technol. Lett.,* vol. 26, no. 8, pp. 753–756, Apr. 2014.

[34] K. N. Pappi, A. S. Lioumpas, and G. K. Karagiannidis, "θ-QAM: A parametric quadrature amplitude modulation family and its performance in AWGN and fading channels," *IEEE Trans. Commun,* vol. 58, no. 4, pp. 1014–1019, Apr. 2010.

[35] "Digital video broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for broadcasting, interactive services, news gathering and other broadband satellite applications (DVBS2)," ETSI, Sophia Antipolis, France, Tech. Rep. ETSIEN 302307 V1.2.1 (2009-08), Aug. 2009.

[36] A. J. Viterbi, "An intuitive justification and a simplified implementation of the MAP decoder for convolutional codes," *IEEE J. Sel. Areas Commun.,* vol. 16, no. 2, pp. 260–264, Feb. 1998.