# Coded Modulation Schemes for Voronoi Constellations



Citation for the original published paper (version of record):
Li, S., Mirani, A., Karlsson, M. et al (2025). Coded Modulation Schemes for Voronoi Constellations. IEEE Transactions on Communications, In Press. http://dx.doi.org/10.1109/TCOMM.2025.3554118

# Coded Modulation Schemes for Voronoi Constellations

Shen Li, Ali Mirani, Magnus Karlsson, *Fellow, IEEE, Fellow, Optica,* and Erik Agrell, *Fellow, IEEE*

***Abstract*—Multidimensional Voronoi constellations (VCs) have been shown to be more power-efficient than quadrature amplitude modulation (QAM) formats given the same uncoded bit error rate, and also have higher achievable information rates. However, a coded modulation scheme that sustains these gains after forward error correction (FEC) coding is still lacking. This paper designs coded modulation schemes with soft-decision FEC codes for VCs, including bit-interleaved coded modulation (BICM) and multilevel coded modulation (MLCM), together with three bit-to-integer mapping algorithms and log-likelihood ratio calculation algorithms. Simulation results show that VCs can achieve up to 1.84 dB signal-to-noise ratio (SNR) gains over QAM with BICM, and up to 0.99 dB SNR gains over QAM with MLCM for the additive white Gaussian noise channel at the bit error rate of $1.81 \times 10^{-3}$, with a low decoding complexity.**

***Index Terms*—Bit-interleaved coded modulation, constellation labeling, forward error correction coding, geometric shaping, information rates, lattices, multilevel coding, multidimensional modulation formats, Ungerboeck SP, Voronoi constellations.**

## I. Introduction

ADVANCED multidimensional (MD) modulation formats are designed to have larger minimum Euclidean distance at the same average symbol energy than traditional two-dimensional (2D) quadrature amplitude modulation (QAM) formats. MD Voronoi constellations (VCs) are such a structured modulation format, comprising a coding lattice and a shaping lattice, the latter being a sublattice of the coding lattice [1], [2]. The coding lattice determines how constellation points are packed, resulting in a coding gain over the cubic packing. The shaping lattice of VCs determines the boundary shape of the constellation, achieving a shaping gain over a hypercubic boundary. When applying soft-decision (SD) forward error correction (FEC) codes to VCs, the much larger coding gain of FEC coding might fully or partially cover the coding gain of VCs. On the other hand, the shaping gain cannot be realized by FEC coding. It is asymptotically 1.53 dB over QAM for the average-power-constrained additive white Gaussian noise (AWGN) channel and can be achieved using geometric or probabilistic shaping [3]. Geometric shaping (GS) can be achieved through multidimensional constellations, where a multidimensional geometrically shaped constellation with equally likely multidimensional symbols results in unequally likely 2D symbols when projected. In contrast, probabilistic shaping does not require multidimensional structures and achieves unequally likely 2D symbols by using a distribution matcher. While the implementation methods differ, the goal remains the same: to transmit approximately Gaussian distributed 2D symbols. This paper focuses on geometric shaping.

As the spectral efficiency increases (by either increasing the dimension or the constellation size), the shaping gain of unconstrained[1] GS increases until converging to 1.53 dB, whereas the complexity increases exponentially [4]. This makes unconstrained GS intractable at high spectral efficiencies. Those GS methods achieving high shaping gains are usually very complex due to high-cardinality lookup tables for encoding and decoding [4]–[6].

VCs as a structured GS method offer a compromise, as they provide significant shaping gain over QAM, and their complexity does not increase as dramatically as unconstrained GS with dimension. More precisely, VCs can have low-complexity encoding and decoding algorithms, i.e., mapping integers to constellation points and vice versa [1], [7]–[10], which entirely avoids the need to store and process all constellation points individually in the transmitter and receiver. VCs have shown better BER performance than Gray-labeled QAM in uncoded systems [11]–[16]. MI and GMI have also been studied for VCs in [12], [17], showing high gains over QAM.

In modern communication systems, SD FEC codes are usually used to provide significant coding gains over uncoded systems. The joint design of the modulation format, labeling rule, and FEC codes is called a coded modulation (CM) scheme. The most widely used CM scheme is Gray-labeled QAM with bit-interleaved coded modulation (BICM), and serves as a benchmark for other CM schemes. In [18], a multilevel coded modulation (MLCM) scheme with SD FEC codes was proposed for the Hurwitz constellation, in which constellation points are a finite set of lattice points from the 4D checkerboard lattice $D_4$, and the boundary is hypercubic. The performance gains over QAM comes from the coding gain of $D_4$. In [19], [20], CM schemes with non-binary SD codes are designed for the 4D Welti constellation,

This research was funded in part by the Swedish Research Council (VR) under grants no. 2017-03702 and no. 2021-03709 and the Knut and Alice Wallenberg Foundation under grant no. 2018.0090.

S. Li was with the Department of Electrical Engineering, Chalmers University of Technology, 412 96 Gothenburg, Sweden and is now with the Centre of Optics, Photonics and Lasers (COPL), Department of Electrical and Computer Engineering, Université Laval, Québec, Canada. E-mail: shen.li.1@ulaval.ca.

A. Mirani was with the Department of Microtechnology and Nanoscience, Chalmers University of Technology, 412 96 Gothenburg, Sweden and is now with Ericsson AB, 417 56, Gothenburg, Sweden.

M. Karlsson is with the Department of Microtechnology and Nanoscience, Chalmers University of Technology, 412 96 Gothenburg, Sweden.

E. Agrell is with the Department of Electrical Engineering, Chalmers University of Technology, 412 96 Gothenburg, Sweden. E-mail: agrell@chalmers.se.

Manuscript received July xx, 2024; revised xx xx, 2024.

[1]"Unconstrained" refers to that there are no specific structural constraints on the positions of constellation points.

TABLE I: The motivation of this work: CM schemes preserving the high shaping gains of VCs observed in uncoded systems are missing in previous literature.

| | Uncoded | CM |
|---|---|---|
| QAM | trivial | large coding gain<br>no shaping gain<br>low complexity |
| VCs | some coding gain<br>shaping gain<br>low complexity | ? |

which has constellation points from the $D_4$ lattice and uses a hypersphere boundary. The performance gains over QAM with BICM comes from the shaping and coding gains of the Welti constellation itself, and the FEC codes (nonbinary codes or multilevel codes) as well.

As illustrated in Table I, channel coding has been widely applied to QAM constellations, offering substantial coding gains. However, CM schemes with QAM do not provide shaping gains. In contrast, VCs can provide high shaping gains and some coding gains (but relatively little compared to what CM schemes can potentially achieve). However, no CM scheme has yet been proposed for VCs, which motivates this work. Designing CM schemes for MD VCs that outperforms QAM with BICM is challenging, due to that no Gray labeling exists for MD VCs, and the resulting penalty from a non-Gray labeling might cancel out the shaping and coding gains of VCs.

In this paper, we focus on MD VCs with a cubic coding lattice, i.e., VCs having high shaping gains but no coding gain. The absent coding gain is instead achieved by FEC codes. In order to achieve high shaping gains, the considered VCs need to be very large with up to 24 dimensions and up to $5 \times 10^{27}$ constellation points at high spectral efficiencies. Applying FEC codes to such large constellations and designing a CM scheme that outperforms QAM with BICM is challenging due to the following reasons. First, the huge constellation sizes of VCs makes the design of the labeling and log-likelihood ratios (LLRs) hard. Second, Gray labelings do not exist for VCs. An effective labeling rule is needed, otherwise the shaping gains of VCs might be canceled out by the labeling penalty. For instance, traditional set partitioning algorithms with lookup tables storing all constellation coordinates cannot be directly applied to such large VCs. Instead, the lattice should be partitioned, not a finite signal set. The traditional max-log LLR approximation cannot be applied to very large VCs, because all constellation coordinates cannot be enumerated. Third, which CM scheme with what structure is suitable for VCs is unknown.

In this paper, we design several CM schemes with SD FEC codes for VCs for the first time, including BICM and MLCM. Also, not only for VCs, it is the first time, to our best of knowledge, to design CM schemes for such large constellations. However, the proposed labeling rule and LLR calculation algorithm have a very low complexity. Moreover, the FEC overhead of the proposed schemes (11%) is lower than the commonly used overheads (15%–25%) of high-performance SD FEC codes for optical communications. Thus, the application scenario of the proposed CM scheme would be ultra high-rate transmission systems, such as the upcoming 800 Gbps and 1.25 Tbps standards for fiber communications.

The structure of the rest of the paper is as follows. Section II introduces lattices and VCs. Section III first introduces the encoding and decoding algorithms (mapping between integers and constellation coordinates) of VCs, among which the mapping between integers and binary labels is the key design procedure. One mapping method from [11] is first reviewed and then two new mapping methods are proposed for VCs. In Section IV, three CM schemes corresponding to the three mapping methods in Section III are designed for VCs for the first time, including the coding structure and LLR calculation for both BICM and MLCM. Section V analyzes the performance of MD VCs with the proposed CM schemes, in terms of the BER after low-density parity check (LDPC) decoding and achievable information rates (AIRs). Finally, Section VI discusses the complexity of the proposed CM schemes.

*Notation:* Bold lowercase symbols denote row vectors and bold uppercase symbols denote random vectors or matrices. All-zero and all-one vectors are denoted by $\mathbf{0}$ and $\mathbf{1}$, respectively. Vector inequalities are performed element-wise, e.g., for vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$, the inequality $\boldsymbol{x} \leq \boldsymbol{y}$ refers to $x_i \leq y_i$ for $i = 1, \ldots, n$. The sets of integer, positive integer, real, complex, and natural numbers are denoted by $\mathbb{Z}$, $\mathbb{Z}^+$, $\mathbb{R}$, $\mathbb{C}$, and $\mathbb{N}$, respectively. Other sets are denoted by calligraphic symbols. Rounding a vector to its nearest integer vector is denoted by $\lfloor \cdot \rceil$, in which ties are broken arbitrarily. The cardinality of a set or the order of a lattice partition is denoted by $|\cdot|$.

## II. Lattices and VCs

An $n$-dimensional lattice $\Lambda$ is an infinite set of points generated by all integer combinations of the rows of its $n \times n$ generator matrix $\boldsymbol{G}_\Lambda$, i.e.,

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

The *closest lattice point quantizer* of a lattice $\Lambda$, denoted by $\mathcal{Q}_\Lambda(\cdot)$, finds the closest lattice point in $\Lambda$ of an arbitrary point $\boldsymbol{x} \in \mathbb{R}^n$, i.e.,

$$\mathcal{Q}_\Lambda(\boldsymbol{x}) = \arg\min_{\boldsymbol{\lambda} \in \Lambda} \|\boldsymbol{x} - \boldsymbol{\lambda}\|^2. \tag{2}$$

A sublattice $\Lambda'$ of $\Lambda$, denoted by $\Lambda' \subseteq \Lambda$, contains a subset of the lattice points[2] of $\Lambda$, which is spanned by the generator matrix $\boldsymbol{G}_{\Lambda'}$ satisfying

$$\boldsymbol{G}_{\Lambda'} = \boldsymbol{J}\boldsymbol{G}_\Lambda, \tag{3}$$

with an $\boldsymbol{J} \in \mathbb{Z}^{n \times n}$. The *lattice partition* $\Lambda/\Lambda'$ partitions $\Lambda$ into $|\Lambda/\Lambda'| = |\det \boldsymbol{G}_{\Lambda'}|/|\det \boldsymbol{G}_\Lambda| = |\det \boldsymbol{J}|$ disjoint *cosets* of $\Lambda'$ [21], and $|\Lambda/\Lambda'|$ is called the *partition order*. If one arbitrary lattice point is selected from each of these cosets, a

[2]Arbitrary points in $\mathbb{R}^n$ are referred to as "points" in this paper. To avoid ambiguity, "lattice point" is used when a point also belongs to a lattice. Later throughout the paper, "constellation points" refers to the points in VCs.

set of *coset representatives* (not unique) is formed, denoted by $[\Lambda/\Lambda']$. Then every lattice point $\boldsymbol{\lambda} \in \Lambda$ can be written as

$$\boldsymbol{\lambda} = \boldsymbol{c} + \boldsymbol{\lambda}', \tag{4}$$

where $\boldsymbol{\lambda}' \in \Lambda'$ and $\boldsymbol{c} \in [\Lambda/\Lambda']$ can be uniquely labeled by $k = \log_2(|\Lambda/\Lambda'|)$ bits if $|\Lambda/\Lambda'|$ is a power of 2. The whole lattice $\Lambda$ is decomposed as

$$\Lambda = [\Lambda/\Lambda'] + \Lambda'. \tag{5}$$

A *partition chain*, formed by a sequence of lattices $\Lambda^0 \supseteq \Lambda^1 \supseteq \cdots \supseteq \Lambda^q$ with $q \in \mathbb{Z}^+$, is denoted by $\Lambda^0/\Lambda^1/\ldots/\Lambda^q$ [22]. Every lattice point $\boldsymbol{\lambda}_0 \in \Lambda^0$ can be written as

$$\boldsymbol{\lambda}_0 = \sum_{i=1}^{q} \boldsymbol{c}_i + \boldsymbol{\lambda}_q, \tag{6}$$

where $\boldsymbol{c}_i \in [\Lambda^{i-1}/\Lambda^i]$ for $i = 1, \ldots, q$ and $\boldsymbol{\lambda}_q \in \Lambda^q$. If the partition orders $|\Lambda^{i-1}/\Lambda^i|$ for $i = 1, \ldots, q$ are powers of 2, $\boldsymbol{\lambda}_0$ can be uniquely labeled by the binary tuple

$$\boldsymbol{b} = (\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_q), \tag{7}$$

where $\boldsymbol{b}_i$ is the vector of bit labels of $\boldsymbol{c}_i$ with the length of $k_i$ for $i = 1, \ldots, q$ and the total length of $\boldsymbol{b}$ is $\sum_{i=1}^{q} k_i = \log_2(|\Lambda^0/\Lambda^q|)$. The lattice $\Lambda^0$ can be decomposed as

$$\Lambda^0 = [\Lambda^0/\Lambda^1] + \cdots + [\Lambda^{q-1}/\Lambda^q] + \Lambda^q. \tag{8}$$

An $n$-dimensional VC is a set of coset representatives of a lattice partition $\Lambda_{\mathrm{c}}/\Lambda_{\mathrm{s}}$, where $\Lambda_{\mathrm{c}}$ is called the *coding lattice*, $\Lambda_{\mathrm{s}}$ is called the *shaping lattice*, and the partition order is $M = |\Lambda_{\mathrm{c}}/\Lambda_{\mathrm{s}}|$, which is a power of 2 to enable binary labeling with $m = \log_2(M)$ bits. The VC points $\Gamma$ are defined as all lattice points in a translated version of $\Lambda_{\mathrm{c}}$ having the all-zero point $\mathbf{0}$ as their closest lattice point in $\Lambda_{\mathrm{s}}$, i.e. [2],

$$\Gamma \triangleq \{\boldsymbol{x} \in (\Lambda_{\mathrm{c}} - \boldsymbol{a}) : \mathcal{Q}_{\Lambda_{\mathrm{s}}}(\boldsymbol{x}) = \mathbf{0}\}, \tag{9}$$

where the *offset vector* $\boldsymbol{a} \in \mathbb{R}^n$ is usually optimized to minimize the average symbol energy [1]

$$E_{\mathrm{s}} = \frac{1}{M} \sum_{\boldsymbol{x} \in \Gamma} \|\boldsymbol{x}\|^2. \tag{10}$$

The *spectral efficiency* [22]–[24] in bits per 2D symbol for the uncoded system is defined as

$$\beta = \frac{2m}{n} \text{ [bits/2D-symbol]}. \tag{11}$$

## III. Labeling of VCs

### A. Encoding and decoding

A *labeling* function is a map from binary labels of length $m$ to constellation points. For the considered VCs based on the lattice partition $\mathbb{Z}^n/\Lambda_{\mathrm{s}}$, we divide the labeling algorithm into two steps: first from binary labels to integers and then from integers to VC points, i.e., $\{0,1\}^m \to \mathcal{U} \to \Gamma$. The integer set $\mathcal{U}$ is formed by first writing the generator matrix of the shaping lattice $\Lambda_{\mathrm{s}}$ as a lower-triangular form $\boldsymbol{G}_{\mathrm{s}}$ with diagonal elements $\boldsymbol{h} = (h_1, \ldots, h_n)$, and then letting

$$\mathcal{U} = \{\boldsymbol{u} \in \mathbb{Z}^n : \mathbf{0} \leq \boldsymbol{u} \leq \boldsymbol{h} - \mathbf{1}\}. \tag{12}$$

*Encoding:* The function that maps binary labels to integers is denoted by $f : \{0,1\}^m \to \mathcal{U}$, which we call *mapping* through out this paper to distinguish from *labeling* and will be discussed in Sections III-B, III-C, and III-D. The algorithm that maps integers $\mathcal{U}$ to constellation points $\Gamma$ was proposed in [10] and summarized in [17, Alg. 1], which is denoted by $g : \mathcal{U} \to \Gamma$ in this paper.

*Decoding:* After receiving a noisy version of a VC point, the algorithm that maps it back to an estimate of the transmitted VC point was proposed in [10] and summarized in [17, Alg. 2], which is denoted by a function $w : \mathbb{R}^n \to \mathcal{U}$ in this paper. Then the binary labels are obtained by the inverse of $f$, i.e., $f^{-1} : \mathcal{U} \to \{0,1\}^m$.

The rest of this section introduces three different mapping functions $f$ for the considered VCs based on the lattice partition $\mathbb{Z}^n/\Lambda_{\mathrm{s}}$. Section III-B reviews the Gray mapping proposed in [11]. Section III-C adapts Ungerboeck's set partitioning (SP) mapping based on lattice partition chains to VCs. Another new hybrid mapping function combining the SP mapping and Gray mapping is then proposed in Section III-D. The design criteria for the two new mapping functions involve addressing the following questions: 1) How many partitioning steps are needed? 2) How should the coset representatives be labeled at each partitioning step? 3) Does the labeling have a sufficiently low penalty to preserve the shaping gains of VCs over QAM with Gray labeling?

### B. Gray mapping

In [11], a mapping method between binary labels and integers is proposed in order to minimize the uncoded BER of VCs, which works in the following way.

First, the binary label $\boldsymbol{b} \in \{0,1\}^m$ is divided into $n$ blocks according to $\boldsymbol{h}$,

$$\boldsymbol{b} = (\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_n),$$

each of which has $\log_2(h_i)$ bits for $i = 1, \ldots, n$, and $\sum_{i=1}^{n} \log_2(h_i) = m$. Then $\boldsymbol{b}_i$ is converted to an integer $u_i$ using the binary reflected Gray code (BRGC) [25] for $i = 1, \ldots, n$, yielding

$$\boldsymbol{u} = (u_1, \ldots, u_n). \tag{13}$$

The above procedures converting $\boldsymbol{b}$ to $\boldsymbol{u}$ according to the BRGC is denoted by $f_{\mathrm{BRGC}}(\boldsymbol{b}, \boldsymbol{h})$, and the inverse process of converting an integer vector $\boldsymbol{u}$ to a binary vector $\boldsymbol{b}$ is denoted by $f^{-1}_{\mathrm{BRGC}}(\boldsymbol{u}, \boldsymbol{h})$ in this paper. After mapping integers to VC points using the function $g$ defined in Section III-A, the labeling is not true Gray, but close to Gray, which is called "pseudo-Gray" labeling[3].

[3]Gray means that all Euclidean neighbors are Hamming neighbors and pseudo-Gray means that almost all Euclidean neighbors are Hamming neighbors. Rectangular QAM constellations have Gray labelings, VCs with cubic coding lattices and Gray mapping have pseudo-Gray labelings [11, Table I], and VCs with advanced coding lattices do not even have pseudo-Gray labelings [13].

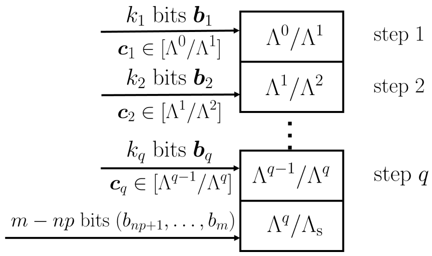



Fig. 1: Illustration of the labeling of a partition chain $\Lambda^0/\Lambda^1/\dots/\Lambda^{q-1}/\Lambda^q/\Lambda_\mathrm{s}$.

TABLE II: Example partition chains in the SP mapping for MD VCs with a cubic coding lattice $\mathbb{Z}^n$.

| $n=2$ | $\mathbb{Z}^2/D_2/2\mathbb{Z}^2/2D_2/4\mathbb{Z}^2/4D_2/\dots$ | | | | |
|---|---|---|---|---|---|
| Step $i$ | 1 | 2 | 3 | 4 | 5 |
| $k_i$ | 1 | 1 | 1 | 1 | 1 |
| $d_i^2$ | 2 | 4 | 8 | 16 | 32 |
| $n=4$ | $\mathbb{Z}^4/D_4/2\mathbb{Z}^4/2D_4/4\mathbb{Z}^4/4D_4/\dots$ | | | | |
| Step $i$ | 1 | 2 | 3 | 4 | 5 |
| $k_i$ | 1 | 1 | 1 | 1 | 1 |
| $d_i^2$ | 2 | 4 | 8 | 16 | 32 |
| $n=8$ | $\mathbb{Z}^8/D_8/E_8\boldsymbol{R}_8/2E_8/2E_8\boldsymbol{R}_8/4E_8/\dots$ | | | | |
| Step $i$ | 1 | 2 | 3 | 4 | 5 |
| $k_i$ | 1 | 3 | 4 | 4 | 4 |
| $d_i^2$ | 2 | 4 | 8 | 16 | 32 |
| $n=16$ | $\mathbb{Z}^{16}/D_{16}/D_{16}\boldsymbol{R}_{16}/\Lambda_{16}/\Lambda_{16}\boldsymbol{R}_{16}/2\Lambda_{16}/\dots$ | | | | |
| Step $i$ | 1 | 2 | 3 | 4 | 5 |
| $k_i$ | 1 | 8 | 3 | 8 | 8 |
| $d_i^2$ | 2 | 4 | 8 | 16 | 32 |

### C. SP mapping

Ungerboeck's SP concept [26] maps binary labels to 1D or 2D constellation points by successively partitioning the constellation into two subsets at each bit level in order to maximize the intra-set minimum squared Euclidean distance (MSED) at each level, so that unequal error protection can be implemented on different bit levels. Since all partition orders are 2, Ungerboeck's SP is also called binary SP. Binary SP has been applied to 1D, 2D [27]–[29], and 4D [18]–[20], [30] signal constellations. When binary SP is applied in larger than 2 dimensions, the MSED might not increase at every bit level. Binary SP requires one encoder and one decoder at each bit level, which has a high complexity in FEC for large constellations. Generalized from the binary SP, signal sets can be partitioned into multiple subsets based on the concept of cosets [2], [21], [31], [32], which enables SP in higher dimensions [2], [31], [32] and increasing MSED at every

TABLE III: An example look-up table for the coset representatives of the lattice partition $D_8/E_8\boldsymbol{R}_8$ and their bit labels.

| $[D_8/E_8\boldsymbol{R}_8]$ | labels |
|---|---|
| (00000000) | (000) |
| (01010000) | (001) |
| (00011000) | (010) |
| (01001000) | (011) |
| (11000000) | (100) |
| (10010000) | (101) |
| (11011000) | (110) |
| (10001000) | (111) |

partition level. However, how the coset representatives are labeled at each partition level is not specified. Since binary SP is not suitable for high-dimensional VCs, to solve the question of how MD VCs should be partitioned and labeled, we propose a systematic algorithm for mapping bits $\boldsymbol{b} \in \{0,1\}^m$ to integers $\boldsymbol{u} \in \mathcal{U}$ based on the concept of SP such that the MSED doubles at every partition level for very large MD VCs based on the lattice partition $\mathbb{Z}^n/\Lambda_\mathrm{s}$.

For large constellations, after getting a sufficiently large intra-set MSED, it is reasonable to not partition the remaining subsets and leave the corresponding bit levels uncoded. One convenient way is to stop partitioning when a scaled integer lattice $2^p\mathbb{Z}^n$ is obtained, where $p \in \mathbb{N}$. Then we map the last $m-np$ bits to integers according to BRGC to minimize the BER for the uncoded bits.

The preprocessing of the proposed SP mapping works as follows. First, $q$ intermediate lattices $\Lambda^1,\dots,\Lambda^q$ are found to form the partition chain

$$\Lambda^0/\Lambda^1/\dots/\Lambda^q/\Lambda_\mathrm{s}, \tag{14}$$

where $\Lambda^0 = \mathbb{Z}^n$ and $\Lambda^q = 2^p\mathbb{Z}^n$ is where to stop the partition. The partition chain should satisfy $\Lambda^0 \supset \Lambda^1 \supset \cdots \supset \Lambda^q \supseteq \Lambda_\mathrm{s}$ and have increasing MSEDs of $d_i^2 = 2^i$ for $i=1,\dots,q$. The order of each partition step is $|\Lambda^{i-1}/\Lambda^i| = 2^{k_i}$ for $i=1,\dots,q$ and $\sum_{i=1}^q k_i = \log_2(|\Lambda^0/\Lambda^q|) = np$. At every partition step $i$, all coset representatives $[\Lambda^{i-1}/\Lambda^i]$ are labeled by $k_i$ bits and the mapping rules are stored in a look-up table $\boldsymbol{C}_i$. Conventionally, the set of coset representatives contains the all-zero lattice point labeled by the all-zero binary tuple. Fig. 1 illustrates the mapping for the partition chain in general. Table II lists some example partition chains and their intra-set MSEDs for MD VCs with a cubic coding lattice. These partition chains contain commonly used lattices as intermediate lattices including the $n$-dimensional checkerboard lattice $D_n$, 8-dimensional (8D) Gosset lattice $E_8$, 16-dimensional (16D) Barnes–Wall lattice $\Lambda_{16}$ [33], and the 24-dimensional (24D) Leech lattice $\Lambda_{24}$ [34, Ch. 4]. The $n \times n$ matrix $\boldsymbol{R}_n$ is an integer orthonormal rotation matrix with a determinant of $\det \boldsymbol{R}_n = 2^{n/2}$ [2], [31]. When multiplied with a lattice generator matrix on the right, it rotates every two dimensions of the lattice by $45^\circ$ and rescales it by $\sqrt{2}$. The size of the look-up table $\boldsymbol{C}_i$ is $2^{k_i}$, which is not more than $2^8$ in Table II and much smaller than a table for the whole VC.

As an example, a set of coset representatives of the partition $D_8/E_8\boldsymbol{R}_8$ and one set of possible bit labels are listed in Table III. Note that neither the choice of the set of coset representatives nor the mapping within the look-up table is unique, and both of them are arbitrarily selected. The effects of different choices on the performance are not studied. However, we conjecture that there would be no big difference since the intra-set MSED cannot be increased by further partitioning the set of coset representatives.

The SP demapping $f_\mathrm{SP}^{-1}$ from an integer vector $\boldsymbol{u}$ to its bit labels $\boldsymbol{b}$ works as follows. Given an integer vector $\boldsymbol{u} \in \mathcal{U}$, starting from the first partition step $\Lambda^0/\Lambda^1$, we know that $\boldsymbol{u}$ belongs to one coset of this partition since $\boldsymbol{u} \in \Lambda^0 = \mathbb{Z}^n$. By full search among a certain set of coset representatives $[\Lambda^0/\Lambda^1]$, there must be only one $\boldsymbol{c}_1 \in [\Lambda^0/\Lambda^1]$ such that $(\boldsymbol{u}-$

$\boldsymbol{c}_1)\cdot\boldsymbol{G}_{\Lambda^1}^{-1}$ yields an integer vector, where $\boldsymbol{G}_{\Lambda^1}$ is the generator matrix of $\Lambda^1$. The bit labels $\boldsymbol{b}_1$ corresponding to $\boldsymbol{c}_1$ is found in the look-up table $\boldsymbol{C}_1$. Then $\boldsymbol{u}-\boldsymbol{c}_1$ is a lattice point of $\Lambda^1$, which must belong to a certain coset of $\Lambda^1/\Lambda^2$. A vector $\boldsymbol{c}_2\in[\Lambda^1/\Lambda^2]$ is found such that $(\boldsymbol{u}-\boldsymbol{c}_1-\boldsymbol{c}_2)\cdot\boldsymbol{G}_{\Lambda^2}^{-1}$ yields an integer vector, where $\boldsymbol{G}_{\Lambda^2}$ is the generator matrix of $\Lambda^2$ and the bit labels $\boldsymbol{b}_2$ corresponding to $\boldsymbol{c}_2$ is found in the look-up table $\boldsymbol{C}_2$. The procedure is repeated until all $\boldsymbol{c}_i$ and $\boldsymbol{b}_i$ are obtained for $i=1,\dots,q$.

Next, the remaining $m-np$ bit labels for the partition $2^p\mathbb{Z}^n/\Lambda_\mathrm{s}$ are found as follows. First, the coset representatives of the partition $\mathbb{Z}^n/2^p\mathbb{Z}^n$ are set as

$$\mathcal{S}=[\mathbb{Z}^n/2^p\mathbb{Z}^n]=\{\boldsymbol{s}\in\mathbb{Z}^n:\boldsymbol{0}\leq\boldsymbol{s}\leq(2^p-1)\cdot\boldsymbol{1}\}. \quad (15)$$

There must be a unique $\boldsymbol{s}\in\mathcal{S}$ such that $\boldsymbol{u}-\boldsymbol{s}\in2^p\mathbb{Z}^n$, and $\boldsymbol{s}$ can be easily found by

$$\boldsymbol{s}=\boldsymbol{u}\bmod 2^p, \quad (16)$$

where mod is the modulo operator that takes the remainder of a vector element-wise and returns a vector. The purpose of choosing such a set of coset representatives is to make sure that $\boldsymbol{u}-\boldsymbol{s}$ still falls within the range of $\mathcal{U}$, i.e., $\boldsymbol{u}-\boldsymbol{s}\in\mathcal{U}\cap2^p\mathbb{Z}^n$. Then we know that

$$\frac{\boldsymbol{u}-\boldsymbol{s}}{2^p}\in\mathbb{Z}^n \quad (17)$$

with the range

$$\boldsymbol{0}\leq\frac{\boldsymbol{u}-\boldsymbol{s}}{2^p}\leq\frac{\boldsymbol{h}}{2^p}-\boldsymbol{1}. \quad (18)$$

The bit labels of $(\boldsymbol{u}-\boldsymbol{s})/2^p$ can be obtained by converting each decimal element to bits according to BRGC, i.e.,

$$(b_{np+1},\dots,b_m)=f_{\mathrm{BRGC}}^{-1}\left(\frac{\boldsymbol{u}-\boldsymbol{s}}{2^p},\frac{\boldsymbol{h}}{2^p}\right). \quad (19)$$

Now we describe the SP mapping $f_{\mathrm{SP}}$ from the bit labels $\boldsymbol{b}$ to the integer vector $\boldsymbol{u}$. Given bit labels $\boldsymbol{b}\in\{0,1\}^m$, the first $np$ bits are divided into $q$ blocks $\boldsymbol{b}_i$ for $i=1,\dots,q$, each of which has $k_i$ bits and indicates a coset representative $\boldsymbol{c}_i$ according to the look-up $\boldsymbol{C}_i$. Then $\boldsymbol{c}=\sum_{i=1}^q\boldsymbol{c}_i$ indicates a coset representative of the lattice partition $\mathbb{Z}^n/2^p\mathbb{Z}^n$, but $\boldsymbol{c}$ might not belong to $\mathcal{S}$, which can be converted to $\boldsymbol{s}\in\mathcal{S}$ by

$$\boldsymbol{s}=\boldsymbol{c}\bmod 2^p. \quad (20)$$

Now we know that $\boldsymbol{u}-\boldsymbol{s}\in2^p\mathbb{Z}^n$. The remaining $m-np$ bits of $\boldsymbol{b}$ indicate an integer vector

$$\boldsymbol{t}=f_{\mathrm{BRGC}}\left((b_{np+1},\dots,b_m),\frac{\boldsymbol{h}}{2^p}\right). \quad (21)$$

Finally, $\boldsymbol{u}$ is obtained by

$$\boldsymbol{u}=\boldsymbol{s}+2^p\boldsymbol{t}. \quad (22)$$

Algorithms 1 and 2 summarize the SP mapping process between $\boldsymbol{u}$ and $\boldsymbol{b}$ for VCs with a cubic coding lattice.

---

**Algorithm 1** SP mapping $f_{\mathrm{SP}}$

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

Preprocessing: The partition chain $\Lambda^0/\Lambda^1/\dots/\Lambda^q/\Lambda_\mathrm{s}$ is given, where all partition orders $|\Lambda^{i-1}/\Lambda^i|$ for $i=1,\dots,q$ are powers of 2. Set up the look-up tables $\boldsymbol{C}_i$ between all coset representatives and their corresponding bit labels for all partition steps. Divide the first $np$ bits of $\boldsymbol{b}$ into $q$ blocks $\boldsymbol{b}_i$ for $i=1,\dots,q$, each of which has $k_i$ bits. Find a lower-triangular generator matrix $\boldsymbol{G}_\mathrm{s}$ of $\Lambda_\mathrm{s}$ and denote the diagonal elements of $\boldsymbol{G}_\mathrm{s}$ as $\boldsymbol{h}$.

Find the corresponding $\boldsymbol{c}_i$ of $\boldsymbol{b}_i$ according to $\boldsymbol{C}_i$ for $i=1,\dots,q$.
Let $\boldsymbol{c}\leftarrow\sum_{i=1}^q\boldsymbol{c}_i$
Let $\boldsymbol{s}\leftarrow\boldsymbol{c}-\lfloor\boldsymbol{c}/2^p\rfloor\cdot2^p$
Let $\boldsymbol{t}\leftarrow f_{\mathrm{BRGC}}((b_{np+1},\dots,b_m),\boldsymbol{h}/2^p)$
Let $\boldsymbol{u}\leftarrow\boldsymbol{s}+2^p\boldsymbol{t}$

---

**Algorithm 2** SP demapping $f_{\mathrm{SP}}^{-1}$

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

The partition chain $\Lambda^0/\Lambda^1/\dots/\Lambda^q/\Lambda_\mathrm{s}$ is given, where all partition orders $|\Lambda^{i-1}/\Lambda^i|$ are powers of 2. Set the look-up tables $\boldsymbol{C}_i$ between all coset representatives and their corresponding bit labels for all steps $i=1,\dots,q$. Set the set of coset representatives of the partition $\Lambda^0/\Lambda^q$ as $\mathcal{S}$ defined in (15). Find a lower-triangular generator matrix $\boldsymbol{G}_\mathrm{s}$ of $\Lambda_\mathrm{s}$ and denote the diagonal elements of $\boldsymbol{G}_\mathrm{s}$ as $\boldsymbol{h}$.

Let $\boldsymbol{v}\leftarrow\boldsymbol{u}$
**for** $i=1,\dots,q$ **do**
  Find the only $\boldsymbol{c}_i\in[\Lambda^{i-1}/\Lambda^i]$ such that $\boldsymbol{u}-\boldsymbol{c}_i\in\Lambda^i$
  Let $\boldsymbol{b}_i$ be the bit labels of $\boldsymbol{c}_i$ according to $\boldsymbol{C}_i$
  Let $\boldsymbol{u}\leftarrow\boldsymbol{u}-\boldsymbol{c}_i$
**end for**
Let $\boldsymbol{s}\leftarrow\boldsymbol{v}\bmod 2^p$
Let $(b_{np+1},\dots,b_m)=f_{\mathrm{BRGC}}^{-1}\left((\boldsymbol{v}-\boldsymbol{s})/2^p,\boldsymbol{h}/2^p\right)$
Let $\boldsymbol{b}\leftarrow(\boldsymbol{b}_1,\dots,\boldsymbol{b}_q,b_{np+1},\dots,b_m)$

---

### *D. Hybrid mapping*

This mapping is a special case of the SP mapping, which is carefully designed for the considered VCs based on the lattice partition $\mathbb{Z}^n/\Lambda_\mathrm{s}$ to tackle the problem of high Gray penalty in the SP mapping, which will be discussed in Section V. The idea is to only consider $q$ intermediate lattices which are a multiple of the cubic lattice, i.e., $\Lambda^i=2^{p_i}\mathbb{Z}^n$ for $i=0,\dots,q$ with positive integers $p_1<p_2<\ldots<p_q=p$ and $p_0=0$. This yields a partition chain $2^{p_0}\mathbb{Z}^n/2^{p_1}\mathbb{Z}^n/2^{p_2}\mathbb{Z}^n/\dots/2^{p_q}\mathbb{Z}^n/\Lambda_\mathrm{s}$. Thus, the order of each partition step is $|\Lambda^{i-1}/\Lambda^i|=2^{k_i}=2^{n(p_i-p_{i-1})}$ for $i=1,\dots,q$ and $\sum_{i=1}^q k_i=\log_2(|\Lambda^0/\Lambda^q|)=np$. The intra-set MSED is $d_i^2=2^{p_i}$ at the $i$th partition step. The coset representatives in each partition step is simply set as

$$\boldsymbol{C}_i=[2^{p_{i-1}}\mathbb{Z}^n/2^{p_i}\mathbb{Z}^n]=\{\boldsymbol{c}:\boldsymbol{0}\leq\boldsymbol{c}\leq(2^{p_i-p_{i-1}}-1)\cdot\boldsymbol{1}\}, \quad (23)$$

for $i=1,\dots,q$, which is labeled by $k_i=n(p_i-p_{i-1})$ bits for $i=1,\dots,q$. Thanks to that these intermediate lattices are a multiple of $\mathbb{Z}^n$, no full search from $\boldsymbol{C}_i$ is needed to find

**Algorithm 3** Hybrid mapping $f_{\mathrm{H}}$

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

Preprocessing: Given the partition chain $2^{p_0}\mathbb{Z}^n/2^{p_1}\mathbb{Z}^n/2^{p_2}\mathbb{Z}^n/\ldots/2^{p_q}\mathbb{Z}^n/\Lambda_{\mathrm{s}}$ with positive integers $p_1 < p_2 < \ldots < p_q = p$ and $p_0 = 0$, set $q$ sets of coset representatives $\boldsymbol{C}_i$ as in (23) for $i = 1,\ldots,q$. Divide the first $np$ bits of $\boldsymbol{b}$ into $q$ blocks $\boldsymbol{b}_i$ for $i = 1,\ldots,q$, each of which has $n(p_i - p_{i-1})$ bits. Find a lower-triangular generator matrix $\boldsymbol{G}_{\mathrm{s}}$ and denote the diagonal elements of $\boldsymbol{G}_{\mathrm{s}}$ as $\boldsymbol{h}$.

1: Let $\boldsymbol{c}_i \leftarrow f_{\mathrm{BRGC}}(\boldsymbol{b}_i, 2^{p_i - p_{i-1}} \cdot \mathbf{1})$ for $i = 1,\ldots,q$
2: Let $\boldsymbol{s} \leftarrow \sum_{i=1}^{q} \boldsymbol{c}_i$
3: Let $\boldsymbol{t} \leftarrow f_{\mathrm{BRGC}}\left((b_{np+1},\ldots,b_m), \boldsymbol{h}/2^p\right)$
4: Let $\boldsymbol{u} \leftarrow \boldsymbol{s} + 2^p \boldsymbol{t}$

the unique coset representative $\boldsymbol{c}_i \in \boldsymbol{C}_i$ as in the SP mapping. The coset representative $\boldsymbol{c}_i$ can be mapped to $n(p_i - p_{i-1})$ bits by directly converting each element of $\boldsymbol{c}_i$ to $p_i - p_{i-1}$ bits according to BRGC, i.e.,

$$\boldsymbol{b}_i = f_{\mathrm{BRGC}}^{-1}(\boldsymbol{c}_i, 2^{p_i - p_{i-1}} \cdot \mathbf{1}), \tag{24}$$

for $i = 1,\ldots,q$.

The hybrid demapping $f_{\mathrm{H}}^{-1}$ from an integer vector $\boldsymbol{u}$ to its bit labels $\boldsymbol{b}$ works as follows. Given an integer vector $\boldsymbol{u} \in \mathcal{U}$, starting from the first partition step $\mathbb{Z}^n/2^{p_1}\mathbb{Z}^n$, the unique $\boldsymbol{c}_1 \in \boldsymbol{C}_1$ can be easily found by

$$\boldsymbol{c}_1 = \boldsymbol{u} \bmod 2^{p_1}. \tag{25}$$

Then $\boldsymbol{u} - \boldsymbol{c}_1$ is a lattice point of $2^{p_1}\mathbb{Z}^n$, which must belong to a certain coset of $2^{p_1}\mathbb{Z}^n/2^{p_2}\mathbb{Z}^n$. Then the unique $\boldsymbol{c}_2$ is obtained by

$$\boldsymbol{c}_2 = (\boldsymbol{u} - \boldsymbol{c}_1) \bmod 2^{p_2}. \tag{26}$$

Repeating this procedure, $\boldsymbol{c}_i$ is found successively by

$$\boldsymbol{c}_i = \left(\boldsymbol{u} - \sum_{j=1}^{i-1} \boldsymbol{c}_j\right) \bmod 2^{p_i} \tag{27}$$

for $i = 1,\ldots,q$. Then the first $np$ bit labels of $\boldsymbol{u}$ can be obtained by (24). Similar to the SP mapping, the remaining $m - np$ bits are obtained by (19), where $\boldsymbol{s} = \sum_{i=1}^{p} \boldsymbol{c}_i$ in this case.

The hybrid mapping $f_{\mathrm{H}}$ finding the corresponding integer vector $\boldsymbol{u}$ of bit labels $\boldsymbol{b}$ works as follows. Given bit labels $\boldsymbol{b} \in \{0,1\}^m$, $\boldsymbol{c}_i \in \boldsymbol{C}_i$ can be directly obtained by

$$\boldsymbol{c}_i = f_{\mathrm{BRGC}}(\boldsymbol{b}_i, 2^{p_i - p_{i-1}} \cdot \mathbf{1}). \tag{28}$$

The integer vector $\boldsymbol{s} = \sum_{i=1}^{p} \boldsymbol{c}_i$ must belong to the set $\mathcal{S}$ defined in (15) due to the definition of $\boldsymbol{C}_i$ in (23). Then $\boldsymbol{u}$ is obtained combining (22) and (21).

Algorithms 3 and 4 summarize the hybrid mapping process between $\boldsymbol{u}$ and $\boldsymbol{b}$ for VCs with a cubic coding lattice.

*Example 1:* A simple example is a 2D VC based on the lattice partition $\mathbb{Z}^n/4D_2$, where $4D_2$ is the scaled 2D checkerboard lattice with the generator matrix

$$\boldsymbol{G}_{\mathrm{s}} = \begin{pmatrix} 8 & 0 \\ 4 & 4 \end{pmatrix}.$$

**Algorithm 4** Hybrid demapping $f_{\mathrm{H}}^{-1}$

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

Preprocessing: Given the partition chain $2^{p_0}\mathbb{Z}^n/2^{p_1}\mathbb{Z}^n/2^{p_2}\mathbb{Z}^n/\ldots/2^{p_q}\mathbb{Z}^n/\Lambda_{\mathrm{s}}$ with positive integers $p_1 < p_2 < \ldots < p_q = p$ and $p_0 = 0$, set $q$ sets of coset representatives $\boldsymbol{C}_i$ as in (23) for $i = 1,\ldots,q$. Find a lower-triangular generator matrix $\boldsymbol{G}_{\mathrm{s}}$ and denote the diagonal elements of $\boldsymbol{G}_{\mathrm{s}}$ as $\boldsymbol{h}$.

1: **for** $i = 1,\ldots,q$ **do**
2: Let $\boldsymbol{c}_i = \boldsymbol{u} \bmod 2^{p_i}$
3: Let $\boldsymbol{b}_i \leftarrow f_{\mathrm{BRGC}}^{-1}(\boldsymbol{c}_i, 2^{p_i - p_{i-1}} \cdot \mathbf{1})$
4: Let $\boldsymbol{u} \leftarrow \boldsymbol{u} - \boldsymbol{c}_i$
5: **end for**
6: Let $(b_{np+1},\ldots,b_m) \leftarrow f_{\mathrm{BRGC}}^{-1}(\boldsymbol{u}/2^p, \boldsymbol{h}/2^p)$
7: Let $\boldsymbol{b} = (\boldsymbol{b}_1,\ldots,\boldsymbol{b}_q, b_{np+1},\ldots,b_m)$

This VC does not provide any shaping gain, which is just for simplicity of illustration. Fig. 2 illustrates the three different mapping rules $f$ for this example VC. It can be observed that all integer points have only 1-bit difference from their nearest neighbors in the Gray mapping. The SP mapping is based on the lattice partition chain $\mathbb{Z}^2/D_2/2\mathbb{Z}^2/4D_2$ ($q = 2,\ p = 1$). The hybrid mapping is based on the lattice partition chain $\mathbb{Z}^2/2\mathbb{Z}^2/4D_2$ ($q = p = 1$), and $[\mathbb{Z}^n/2\mathbb{Z}^n] = \{(0,0),(1,0),(0,1),(1,1)\}$. For both SP and hybrid mapping, within each coset of $2\mathbb{Z}^2/4D_2$, the points have only 1-bit difference among the last three bits from their closest neighbors.

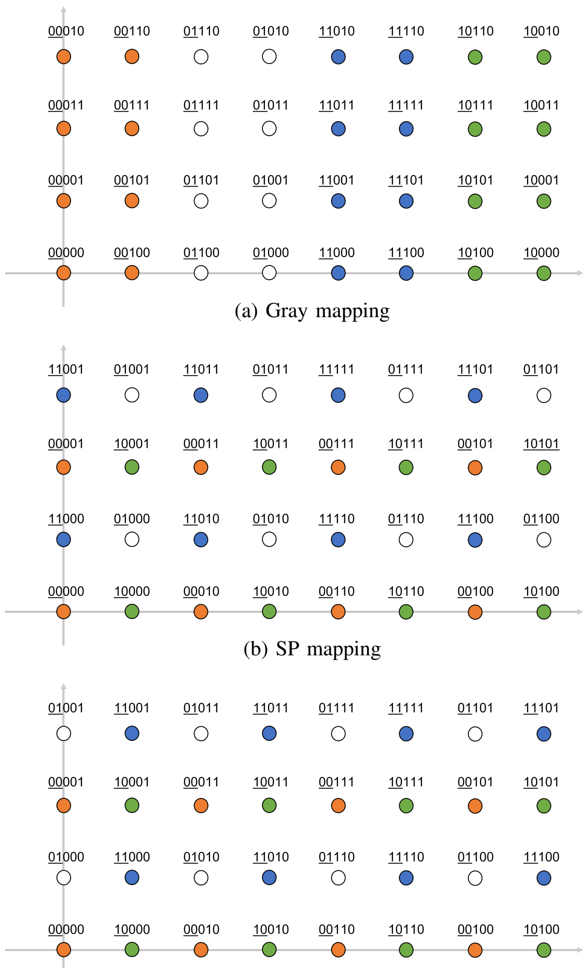
## IV. CM SCHEMES

The joint design of FEC coding and modulation formats is called coded modulation, which plays a vital part in modern communication systems. Designing a CM scheme involves a trade-off among the spectral efficiency, power, and complexity. In this section, we propose three SD CM schemes for VCs based on the lattice partition $\mathbb{Z}^n/\Lambda_{\mathrm{s}}$, adopting the three labeling schemes introduced in Section III, and the computation of the LLRs for SD decoding is discussed.

The proposed CM schemes are designed to be combined with an outer hard-decision (HD) code, known as concatenated coding [35], [36]. Concatenated codes are widely used in many communication standards nowadays, such as the DVB-S2 standards [37] for satellite communications and the 400ZR [38] and upcoming 800G standards for fiber-optical communications [20]. The inner CM scheme brings the uncoded BER down to a certain target BER (e.g., around $10^{-3}$ for fiber-optic communications). Then the outer code can further eliminate the error floor and achieve a very low BER as needed. Commonly used outer codes include Reed–Solomon codes [39], turbo product codes [40], Bose–Chaudhuri–Hocquenghem codes [41], staircase codes [42], and zipper codes [43].

### A. BICM for VCs with Gray mapping

For a memoryless discrete channel with the conditional probability $f_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{y}|\boldsymbol{x})$, the MI between the $n$-dimensional




(a) Gray mapping

(b) SP mapping

(c) Hybrid mapping

Fig. 2: Example 1: Different mapping rules $f$ between integer vectors and bit labels for the VC based on the lattice partition $\mathbb{Z}^2/4D_2$. Integer points having the same first two bits are filled with the same color for better visualization and comparison.

equally likely random transmitted symbols $\boldsymbol{X}$ and received symbols $\boldsymbol{Y}$ is defined as

$$I(\boldsymbol{X};\boldsymbol{Y}) \triangleq \frac{1}{M}\sum_{\boldsymbol{x}\in\Gamma}\int_{\mathbb{R}^n} f_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{y}|\boldsymbol{x})\log\frac{f_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{y}|\boldsymbol{x})}{f_{\boldsymbol{Y}}(\boldsymbol{y})}d\boldsymbol{y}. \tag{29}$$

If the transmitted symbols $\boldsymbol{X}$ are labeled by $m$ bits $(\boldsymbol{B}_1,\ldots,\boldsymbol{B}_m)$, (29) can be written as

$$\begin{aligned} I(\boldsymbol{X};\boldsymbol{Y}) &= I(\boldsymbol{Y};\boldsymbol{B}_1,\ldots,\boldsymbol{B}_m) \\ &= I(\boldsymbol{Y};\boldsymbol{B}_1) + I(\boldsymbol{Y};\boldsymbol{B}_2|\boldsymbol{B}_1) + \ldots \\ &\quad + I(\boldsymbol{Y};\boldsymbol{B}_m|\boldsymbol{B}_1,\ldots,\boldsymbol{B}_{m-1}) \end{aligned} \tag{30}$$

according to the chain rule. If all bit levels $\boldsymbol{B}_1,\ldots,\boldsymbol{B}_m$ are mutually independent of each other and the conditions in all conditional MIs of (30) are neglected, the GMI is obtained [45, Eq. (14)]

$$I_{\text{GMI}} \triangleq \sum_{k=1}^{m} I(\boldsymbol{Y};\boldsymbol{B}_k) \leq I(\boldsymbol{Y};\boldsymbol{X}). \tag{31}$$

The channel is regarded as $m$ independent bit subchannels, which can be encoded and decoded independently. BICM utilizes this concept and contains only one binary component code to protect all bit subchannels. An interleaver is added between the encoder and symbol mapper to distribute the coded bits evenly to all bit subchannels.

The Gray mapping in Section III-B maps each bit level independently, and is suitable to be combined with BICM. Fig. 3a illustrates the BICM scheme for VCs. The total rate of BICM for VCs is $\beta R_{\text{c}}$ [bits/2D-symbol], where $R_{\text{c}}$ is the code rate of the inner code. Decoding is based on bit LLRs, which is described below.

Consider an $n$-dimensional AWGN channel $\boldsymbol{Y}=\boldsymbol{X}+\boldsymbol{Z}$, where $\boldsymbol{X}$ is the transmitted symbol from $\Gamma$, $\boldsymbol{Y}$ is the channel output, and $\boldsymbol{Z}\in\mathbb{R}^n$ is a zero-mean Gaussian random variable with variance $n\sigma^2/2$. The signal-to-noise ratio (SNR) is defined as $E_{\text{s}}/(n\sigma^2/2)$. The max-log approximation [46] of the $k$th bit after receiving a $\boldsymbol{y}^j\in\mathbb{R}^n$ for $j=1,\ldots,N/m$ is defined as

$$\begin{aligned} &\text{LLR}_{\text{max-log}}(b_k|\boldsymbol{y}^j) = \\ &-\frac{1}{\sigma^2}\left(\min_{\boldsymbol{x}\in\Gamma^{(k,0)}}\|\boldsymbol{y}^j-\boldsymbol{x}\|^2 - \min_{\boldsymbol{x}\in\Gamma^{(k,1)}}\|\boldsymbol{y}^j-\boldsymbol{x}\|^2\right), \end{aligned} \tag{32}$$

where $\Gamma^{(k,0)}$ and $\Gamma^{(k,1)}$ are the sets of constellation points with 0 and 1 at bit position $k$, respectively, and $\Gamma^{(k,0)}\cup\Gamma^{(k,1)}=\Gamma$. Computing (32) needs a full search in $\Gamma$, which is infeasible for very large constellations. In [17], an LLR approximation method based on importance sampling is proposed and exemplified for very large VCs based on the lattice partition $\mathbb{Z}^n/\Lambda_{\text{s}}$ for the AWGN channel. The idea is to only search from a small portion of the whole constellation, which is called "importance set". Following this idea, we further approximate the max-log LLR by replacing the $\Gamma$ in (32) by a "Euclidean ball" centered at $\lfloor \boldsymbol{y}^j+\boldsymbol{a}\rceil$

$$\mathcal{B}(\boldsymbol{y}^j,R^2) \triangleq \{\boldsymbol{e}: \|\boldsymbol{e}+\boldsymbol{a}-\lfloor \boldsymbol{y}^j+\boldsymbol{a}\rceil\|^2 \leq R^2, \boldsymbol{e}+\boldsymbol{a}\in\mathbb{Z}^n\}, \tag{33}$$

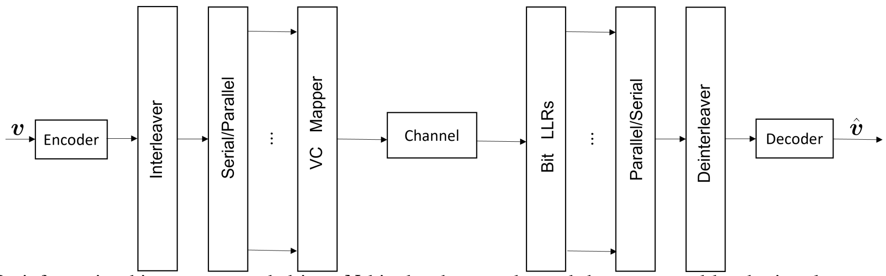
where $\boldsymbol{a}$ is the same offset vector when generating the VC as in (9), $\lfloor\cdot\rceil$ represents rounding a vector to its nearest integer vector, and $R^2\geq 0$ is the squared radius of the Euclidean ball. The Euclidean ball $\mathcal{B}(\boldsymbol{y}^j,R^2)$ is much smaller than the whole VC $\Gamma$. Then the LLR approximation for VCs with BICM is

$$\begin{aligned} &\text{LLR}_{\text{BICM}}(b_k|\boldsymbol{y}^j) = -\frac{1}{\sigma^2} \\ &\left(\min_{\boldsymbol{x}\in\mathcal{B}(\boldsymbol{y}^j,R^2)^{(k,0)}}\|\boldsymbol{y}^j-\boldsymbol{x}\|^2 - \min_{\boldsymbol{x}\in\mathcal{B}(\boldsymbol{y}^j,R^2)^{(k,1)}}\|\boldsymbol{y}^j-\boldsymbol{x}\|^2\right), \end{aligned} \tag{34}$$

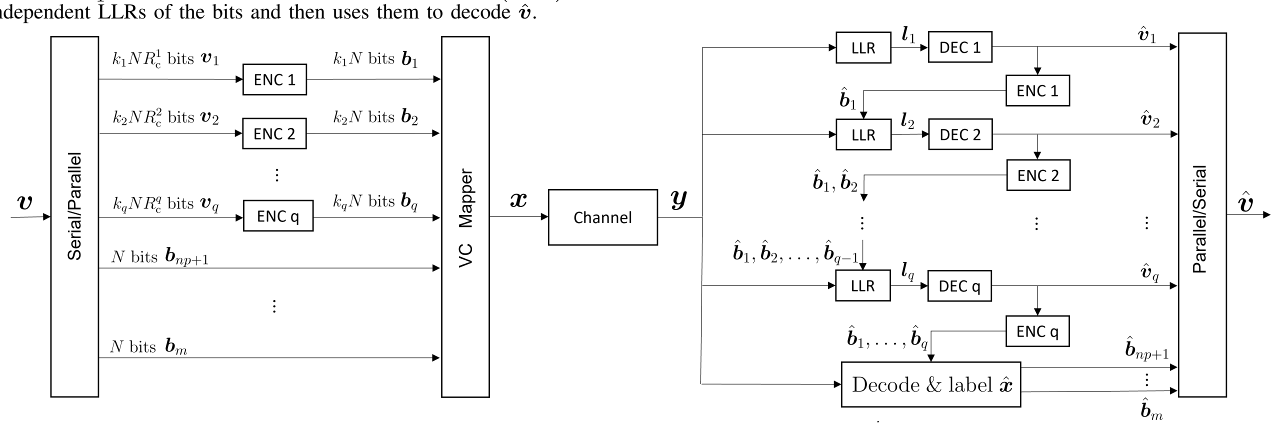
where $\mathcal{B}(\boldsymbol{y}^j,R^2)^{(k,0)}$ and $\mathcal{B}(\boldsymbol{y}^j,R^2)^{(k,1)}$ are the sets of points in $\mathcal{B}(\boldsymbol{y}^j,R^2)$ with 0 and 1 at bit position $k$, respectively, and $\mathcal{B}(\boldsymbol{y}^j,R^2)^{(k,0)}\cup\mathcal{B}(\boldsymbol{y}^j,R^2)^{(k,1)}=\mathcal{B}(\boldsymbol{y}^j,R^2)$. Optionally, $\mathcal{B}(\boldsymbol{y}^j,R^2)$ in (34) can be replaced by $\mathcal{B}(\boldsymbol{y}^j,R^2)\cap\Gamma$, as in [17, Eq. (33)], which might improve the decoding performance at the cost of a much higher complexity because the closest lattice point quantizer needs to be applied to all points in $\mathcal{B}(\boldsymbol{y}^j,R^2)$ in order to determine the intersection with $\Gamma$.

This article has been accepted for publication in IEEE Transactions on Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCOMM.2025.3554118






(a) BICM: A block of $NR_{\mathrm{c}}$ information bits $\boldsymbol{v}$ are encoded into $N$ bits by the encoder and then permuted by the interleaver to avoid burst errors [44]. The serial bits after interleaving are converted to $m$ parallel bit streams $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_m$ of length $N/m$. At the time slot $j = 1, \ldots, N/m$, a VC mapper first maps $m$ bits $\boldsymbol{b}^j = (b_1^j, \ldots, b_m^j)$ to an integer $\boldsymbol{u}^j$ by $\boldsymbol{u}^j = f_{\mathrm{BRGC}}\left(\boldsymbol{b}^j, \boldsymbol{h}\right)$, and then maps $\boldsymbol{u}^j$ to a VC point $\boldsymbol{x}^j \in \Gamma$ by $\boldsymbol{x}^j = g(\boldsymbol{u}^j)$. The receiver deinterleaves $N$ independent LLRs of the bits and then uses them to decode $\hat{\boldsymbol{v}}$.




(b) MLCM: A block of serial information bits $\boldsymbol{v}$ are partitioned into $q$ parallel bit streams $\boldsymbol{v}_i$ with $k_iNR_{\mathrm{c}}^i$ bits for $i = 1, \ldots, q$ and $m - np$ parallel uncoded bit streams $\boldsymbol{b}_{np+1}, \ldots, \boldsymbol{b}_m$ with length $N$. Then $\boldsymbol{v}_i$ is encoded into $k_iN$ bits $\boldsymbol{b}_i$ by encoder (ENC) $i$ for $i = 1, \ldots, q$. The VC mapper first maps bits to $N$ integer vectors by the SP or hybrid mapping, and then encode these integer vectors into $N$ VC points $\boldsymbol{x}$. Multistage decoding is performed at the receiver after receiving $N$ noisy symbols $\boldsymbol{y}$. Decoder (DEC) 1 first decodes $k_1NR_{\mathrm{c}}^1$ bits $\hat{\boldsymbol{v}}_1$ back based on $\boldsymbol{y}$ and $k_1N$ LLRs $\boldsymbol{l}_1$. Then $\hat{\boldsymbol{v}}_1$ is encoded into $\hat{\boldsymbol{b}}_1$ by encoder 1. Decoder $i = 2, \ldots, q$ successively decodes $\hat{\boldsymbol{v}}_i$ and reencodes it into $\hat{\boldsymbol{b}}_i$ based on $\boldsymbol{y}$ and LLRs $\boldsymbol{l}_i$, given all previous bits $\hat{\boldsymbol{b}}_1, \ldots, \hat{\boldsymbol{b}}_{i-1}$. The estimation of the uncoded bits $\hat{\boldsymbol{b}}_{np+1}, \ldots, \hat{\boldsymbol{b}}_m$ is obtained after getting $\hat{\boldsymbol{b}}_1, \ldots, \hat{\boldsymbol{b}}_q$.

Fig. 3: Block diagrams of BICM and MLCM for VCs.

The bit labels of the points in the Euclidean ball $\boldsymbol{e} \in \mathcal{B}(\boldsymbol{y}^j, R^2)$ can be obtained by

$$f_{\mathrm{BRGC}}^{-1}\left(w(\boldsymbol{e}), \boldsymbol{h}\right). \tag{35}$$

Then for each bit position $k$, $\mathcal{B}(\boldsymbol{y}^j, R^2)$ can be divided into two subsets $\mathcal{B}(\boldsymbol{y}^j, R^2) = \mathcal{B}^{(k,0)}(\boldsymbol{y}^j, R^2) \cup \mathcal{B}^{(k,1)}(\boldsymbol{y}^j, R^2)$, containing points within $\mathcal{B}(\boldsymbol{y}^j, R^2)$ with 0 and 1 at bit position $k$, respectively. Thus, the approximated max-log LLRs $\boldsymbol{l}^j$ of the $j$th channel realization $\boldsymbol{y}^j$ contain $m$ independent values $l_k^j$ for $k = 1, \ldots, m$. The LLR of the $k$th bit is computed as

$$\begin{aligned} l_k^j = &-\frac{1}{\sigma^2}\Bigg(\min_{\boldsymbol{e}\in\mathcal{B}^{(k,0)}(\boldsymbol{y}^j, R^2)} \|\boldsymbol{y}^j + \boldsymbol{a} - \boldsymbol{e}\|^2 \\ &- \min_{\boldsymbol{e}\in\mathcal{B}^{(k,1)}(\boldsymbol{y}^j, R^2)} \|\boldsymbol{y}^j + \boldsymbol{a} - \boldsymbol{e}\|^2\Bigg). \end{aligned} \tag{36}$$

If either $\mathcal{B}^{(k,0)}(\boldsymbol{y}^j, R^2)$ or $\mathcal{B}^{(k,1)}(\boldsymbol{y}^j, R^2)$ is empty, then the corresponding minimum in (36) is set to a large default value $r > R^2$. Here only integer values of $R^2$ are considered, because the $\|\boldsymbol{e} + \boldsymbol{a} - \lfloor \boldsymbol{y}^j + \boldsymbol{a} \rceil\|^2$ in (33) is always an integer. The choice of $R^2$ involves a trade-off between computation complexity and decoding performance. For high-dimensional VCs, the LLR approximation can have high complexity when the Euclidean ball contains a larger number of points. Given $R^2$, $r$ can be roughly optimized by testing which value gives the best decoding performance.

### B. MLCM for VCs with SP mapping

Denoting the terms of (30) as $I_1, \ldots, I_m$, a channel can be regarded as $m$ virtual independent "equivalent subchannels" with MIs

$$I_k = I(\boldsymbol{Y}; \boldsymbol{B}_k | \boldsymbol{B}_1, \ldots, \boldsymbol{B}_{k-1}) \tag{37}$$

for $k = 1, \ldots, m$. This concept directly implies an MLCM scheme proposed by Imai and Hirakawa in [47], where the bit subchannels are protected unequally with different component channel codes and a multistage decoder decodes the bits successively from $\boldsymbol{B}_1$ to $\boldsymbol{B}_m$ provided that the previous bits are given. Practical design rules of the code rates can be found in [27]. The suitable labeling for MLCM is Ungerboeck's SP labeling.

Fig. 3b shows an MLCM scheme for VCs, which contains $q$ component codes with code rates $R_{\mathrm{c}}^i$ for $i = 1, \ldots, q$ and the same codeword length $N$ to protect the first $np$ bit levels of the VC symbols, and the last $m - np$ bit levels remain uncoded. Thus, the MLCM for VCs has a total rate of

$$R_{\mathrm{tot}} = \frac{\sum_{i=1}^{q} k_i R_{\mathrm{c}}^i + (m - np)}{n/2} [\text{bits/2D-symbol}]. \tag{38}$$

The transmitter forms $m$ bits $\boldsymbol{b}^j = (\boldsymbol{b}_1^j, \ldots, \boldsymbol{b}_q^j, b_{np+1}^j, \ldots, b_m^j)$ for $j = 1, \ldots, N$, where $\boldsymbol{b}_i^j = (b_i^{k_i(j-1)+1}, \ldots, b_i^{k_ij})$ are the $(k_i(j-1)+1)$th to $k_ij$th bits of $\boldsymbol{b}_i$ for $i = 1, \ldots, q$. The $\boldsymbol{b}_i$

is illustrated in Fig. 3b with length $k_i N$ for $i = 1, \dots, q$. For the SP mapping, the VC mapper first maps $\boldsymbol{b}^j$ to an integer by $\boldsymbol{u}^j = f_{\text{SP}}(\boldsymbol{b}^j)$ and then encodes the integer into a VC point by $\boldsymbol{x}^j = g(\boldsymbol{u}^j)$.

At the receiver side, after getting $(\hat{\boldsymbol{b}}_1^j, \dots, \hat{\boldsymbol{b}}_q^j)$, the coset representative $\hat{\boldsymbol{c}}_i^j$ is obtained according to $\boldsymbol{C}_i$ for $i = 1, \dots, q$, and the $\hat{\boldsymbol{c}}^j \in [\mathbb{Z}^n / 2^p \mathbb{Z}^n]$ is found by $\hat{\boldsymbol{c}}^j = \sum_{i=1}^q \hat{\boldsymbol{c}}_i^j$. Then the estimation of the transmitted point should be found by searching a point within the subset $2^p \mathbb{Z}^n + \hat{\boldsymbol{c}}^j$ that is closest to $\boldsymbol{y}^j + \boldsymbol{a}$. This is equivalent to

$$\begin{aligned} \hat{\boldsymbol{x}}^j &= \mathcal{Q}_{2^p \mathbb{Z}^n + \hat{\boldsymbol{c}}^j}(\boldsymbol{y}^j + \boldsymbol{a}) \\ &= 2^p \left\lfloor \frac{\boldsymbol{y}^j + \boldsymbol{a} - \hat{\boldsymbol{c}}^j}{2^p} \right\rceil + \hat{\boldsymbol{c}}^j - \boldsymbol{a}. \end{aligned} \tag{39}$$

Finally, the bit labels of $\hat{\boldsymbol{x}}^j$ are obtained by

$$\hat{\boldsymbol{b}}^j = (\hat{\boldsymbol{b}}_1^j, \dots, \hat{\boldsymbol{b}}_q^j, \hat{b}_{np+1}^j, \dots, \hat{b}_m^j) = f_{\text{SP}}^{-1}(w(\hat{\boldsymbol{x}}^j)), \tag{40}$$

where the last $m - np$ bits $(\hat{b}_{np+1}^j, \dots, \hat{b}_m^j)$ are the estimation of the uncoded bits mapped to $\hat{\boldsymbol{x}}^j$.

The approximated max-log LLRs of $\boldsymbol{b}_i^j$ contains $k_i$ LLR values independent of each other, denoted by $\boldsymbol{l}_i^j = (l_{i,1}^j, \dots, l_{i,k_i}^j)$, which is computed by the following procedure. Given $\hat{\boldsymbol{b}}_1^j, \dots, \hat{\boldsymbol{b}}_{i-1}^j$, the coset representatives $\hat{\boldsymbol{c}}_1^j, \dots, \hat{\boldsymbol{c}}_{i-1}^j$ are directly obtained according to look-up tables $\boldsymbol{C}_1, \dots, \boldsymbol{C}_{i-1}$. Then we know that the corresponding integer belongs to the lattice $\Lambda^{i-1} + \sum_{t=1}^{i-1} \hat{\boldsymbol{c}}_t^j$. In the $i$th partition step, the coset representatives $[\Lambda^{i-1}/\Lambda^i]$ have been labeled by the look-up table $\boldsymbol{C}_i$. Then we can divide $\boldsymbol{C}_i$ into two subsets $\boldsymbol{C}_i = \boldsymbol{C}_i^{(e,0)} \cup \boldsymbol{C}_i^{(e,1)}$, representing coset representatives having a bit 0 and 1 at the $e$th bit of $\boldsymbol{b}_i$, respectively. For all $\hat{\boldsymbol{c}}_i^j \in \boldsymbol{C}_i^{(e,0)}$, we find the closest point to $\boldsymbol{y}^j + \boldsymbol{a}$ from the lattice $\Lambda^i + \sum_{t=1}^{i-1} \hat{\boldsymbol{c}}_t^j + \hat{\boldsymbol{c}}_i^j$, and denote all such closest points as the set

$$\mathcal{Z}_i^{(e,0)} = \{\boldsymbol{z} = \mathcal{Q}_{\Lambda^i + \sum_{t=1}^i \hat{\boldsymbol{c}}_t^j}(\boldsymbol{y}^j + \boldsymbol{a}) : \hat{\boldsymbol{c}}_i^j \in \boldsymbol{C}_i^{(e,0)}\}. \tag{41}$$

The set $\mathcal{Z}_i^{(k,1)}$ is defined analogously. Then the max-log LLR of the $e$th bit of $\boldsymbol{b}_i^j$ can be approximated as

$$\begin{aligned} &\text{LLR}_{\text{MLCM}}\left(b_i^j | \boldsymbol{y}^j, \hat{\boldsymbol{b}}_1^j, \dots, \hat{\boldsymbol{b}}_{i-1}^j\right) = l_{i,e}^j \\ &= -\frac{1}{\sigma^2}\left(\min_{\boldsymbol{z} \in \mathcal{Z}_i^{(e,0)}} \|\boldsymbol{y}^j + \boldsymbol{a} - \boldsymbol{z}\|^2 - \min_{\boldsymbol{z} \in \mathcal{Z}_i^{(e,1)}} \|\boldsymbol{y}^j + \boldsymbol{a} - \boldsymbol{z}\|^2\right). \end{aligned} \tag{42}$$

This expression is not the exact max-log LLR but an approximation, because the point $\boldsymbol{z} \in \mathcal{Z}_i^{(e,0)}$ or $\boldsymbol{z} \in \mathcal{Z}_i^{(e,1)}$ that is closest to $\boldsymbol{y}^j + \boldsymbol{a}$ might fall outside of the VC. Replacing $\mathcal{Z}_i^{(e,0)}$ and $\mathcal{Z}_i^{(e,1)}$ with $\Gamma \cap \mathcal{Z}_i^{(e,0)}$ and $\Gamma \cap \mathcal{Z}_i^{(e,1)}$, respectively, gives the expression of the exact max-log LLR with additional complexity coming from applying the closest lattice point quantizer to the sets $\mathcal{Z}_i^{(e,0)}$ and $\mathcal{Z}_i^{(e,1)}$. The computation complexity (42) depends on the partition orders $k_i$, which is much lower than the complexity of (36) in BICM. However, MLCM uses $q$ component codes, which adds complexity and delay compared with BICM.

TABLE IV: The considered VCs and TDHQ formats in the simulation.

| Name | $n$ | $\Lambda/\Lambda_{\text{s}}$ | $M$ | $m$ | $\beta$ |
|---|---|---|---|---|---|
| $E_8^{24}$ | 8 | $\mathbb{Z}^8/8E_8$ | $16,777,216$ | 24 | 6 |
| $\Lambda_{24}^{72}$ | 24 | $\mathbb{Z}^{24}/2\Lambda_{24}\boldsymbol{R}_{24}$ | $\approx 4.7 \times 10^{21}$ | 72 | 6 |
| $E_8^{32}$ | 8 | $\mathbb{Z}^8/16E_8$ | $\approx 4.3 \times 10^9$ | 32 | 8 |
| $\Lambda_{24}^{96}$ | 24 | $\mathbb{Z}^{24}/4\Lambda_{24}\boldsymbol{R}_{24}$ | $\approx 7.9 \times 10^{28}$ | 96 | 8 |
| $E_8^{40}$ | 8 | $\mathbb{Z}^8/32E_8$ | $\approx 1.1 \times 10^{12}$ | 40 | 10 |
| $\Lambda_{24}^{120}$ | 24 | $\mathbb{Z}^{24}/8\Lambda_{24}\boldsymbol{R}_{24}$ | $\approx 1.3 \times 10^{36}$ | 120 | 10 |
| $E_8^{48}$ | 8 | $\mathbb{Z}^8/64E_8$ | $\approx 2.8 \times 10^{14}$ | 48 | 12 |
| $\Lambda_{24}^{144}$ | 24 | $16\mathbb{Z}^{24}/4\Lambda_{24}\boldsymbol{R}_{24}$ | $\approx 2.2 \times 10^{43}$ | 144 | 12 |
| $\Lambda_{16}^{76}$ | 16 | $\mathbb{Z}^{16}/16\Lambda_{16}$ | $\approx 7.9 \times 10^{22}$ | 76 | 9.5 |
| $\Lambda_{16}^{92}$ | 16 | $\mathbb{Z}^{16}/32\Lambda_{16}$ | $\approx 5.0 \times 10^{27}$ | 92 | 11.5 |
| Name | $t_1, t_2$ | $M_1, M_2$ | $m_{\text{QAM}}$ | $\beta_{\text{QAM}}$ | |
| TDHQ1 | 4, 4 | 512,1024 | 76 | 9.5 | |
| TDHQ1 | 4, 4 | 2048,4096 | 92 | 11.5 | |

### C. MLCM for VCs with hybrid mapping

The MLCM scheme for VCs with the hybrid mapping in Section III-D is a special case of Fig. 3b with $p = p_q$ and $k_i = n(p_i - p_{i-1})$. At time step $j = 1, \dots, N$, the VC mapper maps $m$ bits $\boldsymbol{b}^j = (\boldsymbol{b}_1^j, \dots, \boldsymbol{b}_q^j, b_{np+1}^j, \dots, b_m^j)$ to an integer $\boldsymbol{u}^j = f_{\text{H}}(\boldsymbol{b}^j)$ and then maps $\boldsymbol{u}^j$ to a VC point $\boldsymbol{x}^j = g(\boldsymbol{u}^j)$. At the receiver side, successive decoding is performed based on $\boldsymbol{y}^j$ and all previous bits $(\hat{\boldsymbol{b}}_1^j, \dots, \hat{\boldsymbol{b}}_{i-1}^j)$ for decoder $i$. After decoding $(\hat{\boldsymbol{b}}_1^j, \dots, \hat{\boldsymbol{b}}_q^j)$, the coset representative $\hat{\boldsymbol{c}}_i^j$ is obtained by (28) for $i = 1, \dots, q$. The estimation of the coset representative of the partition $\mathbb{Z}^n / 2^p \mathbb{Z}^n$ is calculated as $\hat{\boldsymbol{s}}^j = \sum_{i=1}^q \hat{\boldsymbol{c}}_i^j$. The estimation of the transmitted VC point $\hat{\boldsymbol{x}}^j$ is decoded by (39), where $\hat{\boldsymbol{c}}^j$ is replaced by $\hat{\boldsymbol{s}}^j$. Finally, the estimation of bit labels of $\hat{\boldsymbol{x}}^j$ is obtained by

$$\hat{\boldsymbol{b}}^j = (\hat{\boldsymbol{b}}_1^j, \dots, \hat{\boldsymbol{b}}_q^j, \hat{b}_{np+1}^j, \dots, \hat{b}_m^j) = f_{\text{H}}^{-1}(w(\hat{\boldsymbol{x}}^j)), \tag{43}$$

where the last $m - np$ bits $(\hat{b}_{np+1}^j, \dots, \hat{b}_m^j)$ are the estimation of the uncoded bits mapped to $\boldsymbol{x}^j$. The hybrid CM scheme for VCs has a total rate of

$$R_{\text{tot}} = \frac{\sum_{i=1}^q n(p_i - p_{i-1}) R_{\text{c}}^i + (m - np)}{n/2} \text{[bits/2D-symbol]}. \tag{44}$$

The approximation of the max-log LLRs of $\boldsymbol{b}_i^j = (b_{i,1}^j, \dots, b_{i,n(p_i - p_{i-1})}^j)$ contain $n(p_i - p_{i-1})$ independent LLR values for $i = 1, \dots, q$, denoted by $\boldsymbol{l}_i^j = (l_{i,1}^j, \dots, l_{i,n(p_i - p_{i-1})}^j)$ and calculated as follows. Given the previous estimated bits $\hat{\boldsymbol{b}}_1, \dots, \hat{\boldsymbol{b}}_{i-1}$, the coset representatives $\boldsymbol{c}_t$ for $t = 1, \dots, i-1$ are obtained by (28). If $|2^{p_{i-1}} \mathbb{Z}^n / 2^{p_i} \mathbb{Z}^n|$ is not a very large number, the approximated max-log LLR of the $e$th bit of $\boldsymbol{b}_i^j$, denoted by $l_{i,e}^j$, can be calculated using (42) with $\Lambda^{i-1} = 2^{p_{i-1}} \mathbb{Z}^n$ and $\Lambda^i = 2^{p_i} \mathbb{Z}^n$. If $|2^{p_{i-1}} \mathbb{Z}^n / 2^{p_i} \mathbb{Z}^n|$ is large, then $l_{i,e}^j$ can be calculated by enumerating a scaled Euclidean ball centered at the closest lattice point of $2^{p_{i-1}} \mathbb{Z}^n$ to $\boldsymbol{y}^j$, i.e.,

$$\begin{aligned} \mathcal{D}(\boldsymbol{y}^j, R^2) \triangleq \Bigg\{ &\boldsymbol{e} : \left\|\boldsymbol{e} + \boldsymbol{a} - \left\lfloor \frac{\boldsymbol{y}^j + \boldsymbol{a}}{2^{p_{i-1}}} \right\rceil \cdot 2^{p_{i-1}} \right\|^2 \leq 2^{2p_{i-1}} R^2, \\ &\boldsymbol{e} + \boldsymbol{a} \in 2^{p_{i-1}} \mathbb{Z}^n \Bigg\}, \end{aligned} \tag{45}$$

TABLE V: The parameters of the considered CM schemes in simulation.

| Constellation | Mapping | $\beta$ | CM | Partition chain | #LDPC codes | Code rates | #Coded bit levels/$m$ | $R_{\text{tot}}$ [2D-symbol] |
|---|---|---|---|---|---|---|---|---|
| $E_8^{24}$ | Gray | 6 | BICM | - | 1 | $R_{\text{c}}=8/9$ | 24/24 | 5.33 |
| 64-QAM | Gray | 6 | BICM | - | 1 | $R_{\text{c}}=8/9$ | 6/6 | 5.33 |
| $E_8^{24}$ | hybrid | 6 | MLCM | $\mathbb{Z}^8/2\mathbb{Z}^8/E_8^{24}$ | 1 | $R_{\text{c}}^1=2/3$ | 8/24 | 5.33 |
| $\Lambda_{24}^{72}$ | hybrid | 6 | MLCM | $\mathbb{Z}^{24}/2\mathbb{Z}^{24}/\Lambda_{24}^{72}$ | 1 | $R_{\text{c}}^1=2/3$ | 24/72 | 5.33 |
| 64-QAM | hybrid | 6 | MLCM | - | 1 | $R_{\text{c}}^1=2/3$ | 2/6 | 5.33 |
| $E_8^{32}$ | Gray | 8 | BICM | - | 1 | $R_{\text{c}}=9/10$ | 32/32 | 7.2 |
| 256-QAM | Gray | 8 | BICM | - | 1 | $R_{\text{c}}=9/10$ | 8/8 | 7.2 |
| 512-QAM | quasi-Gray | 9 | BICM | - | 1 | $R_{\text{c}}=4/5$ | 9/9 | 7.2 |
| $E_8^{32}$ | hybrid | 8 | MLCM | $\mathbb{Z}^8/2\mathbb{Z}^8/E_8^{32}$ | 1 | $R_{\text{c}}^1=3/5$ | 8/32 | 7.2 |
| $\Lambda_{24}^{96}$ | hybrid | 8 | MLCM | $\mathbb{Z}^{24}/2\mathbb{Z}^{24}/\Lambda_{24}^{96}$ | 1 | $R_{\text{c}}^1=3/5$ | 24/96 | 7.2 |
| 256-QAM | hybrid | 8 | MLCM | - | 1 | $R_{\text{c}}^1=3/5$ | 2/8 | 7.2 |
| 256-QAM | SP | 8 | MLCM | - | 2 | $R_{\text{c}}^1=1/3, R_{\text{c}}^2=8/9$ | 2/8 | 7.22 |
| $E_8^{40}$ | Gray | 10 | BICM | - | 1 | $R_{\text{c}}=9/10$ | 40/40 | 9 |
| 1024-QAM | Gray | 10 | BICM | - | 1 | $R_{\text{c}}=9/10$ | 10/10 | 9 |
| $E_8^{40}$ | hybrid | 10 | MLCM | $\mathbb{Z}^8/2\mathbb{Z}^8/E_8^{40}$ | 1 | $R_{\text{c}}^1=1/2$ | 8/40 | 9 |
| $\Lambda_{24}^{120}$ | hybrid | 10 | MLCM | $\mathbb{Z}^{24}/2\mathbb{Z}^{24}/\Lambda_{24}^{120}$ | 1 | $R_{\text{c}}^1=1/2$ | 24/120 | 9 |
| 1024-QAM | hybrid | 10 | MLCM | - | 1 | $R_{\text{c}}^1=1/2$ | 2/10 | 9 |
| $E_8^{48}$ | Gray | 12 | BICM | - | 1 | $R_{\text{c}}=9/10$ | 48/48 | 10.8 |
| 4096-QAM | Gray | 12 | BICM | - | 1 | $R_{\text{c}}=9/10$ | 12/12 | 10.8 |
| $E_8^{48}$ | hybrid | 12 | MLCM | $\mathbb{Z}^8/2\mathbb{Z}^8/E_8^{48}$ | 1 | $R_{\text{c}}^1=2/5$ | 8/48 | 10.8 |
| $\Lambda_{24}^{144}$ | hybrid | 12 | MLCM | $\mathbb{Z}^{24}/2\mathbb{Z}^{24}/\Lambda_{24}^{144}$ | 1 | $R_{\text{c}}^1=2/5$ | 24/144 | 10.8 |
| 4096-QAM | hybrid | 12 | MLCM | - | 1 | $R_{\text{c}}^1=2/5$ | 2/12 | 10.8 |
| $E_8^{48}$ | SP | 12 | MLCM | $\mathbb{Z}^8/D_8/E_8\boldsymbol{R}_8/2\mathbb{Z}^8/E_8^{48}$ | 1 | $R_{\text{c}}^1=0, R_{\text{c}}^2=0, R_{\text{c}}^3=4/5$ | 8/48 | 10.8 |
| $\Lambda_{16}^{92}$ | Gray | 11.5 | BICM | - | 1 | $R_{\text{c}}=9/10$ | 92/92 | 10.35 |
| TDHQ2 | Gray | 11.5 | BICM | - | 1 | $R_{\text{c}}=9/10$ | 92/92 | 10.35 |
| $\Lambda_{16}^{92}$ | hybrid | 11.5 | MLCM | $\mathbb{Z}^{16}/2\mathbb{Z}^{16}/\Lambda_{16}^{92}$ | 1 | $R_{\text{c}}^1=2/5$ | 16/92 | 10.3 |
| TDHQ2 | hybrid | 11.5 | MLCM | - | 1 | $R_{\text{c}}^1=2/5$ | 16/92 | 10.3 |

which consists of two subsets $\mathcal{D}(\boldsymbol{y}^j,R^2)=\mathcal{D}(\boldsymbol{y}^j,R^2)^{(e,0)}\cup\mathcal{D}(\boldsymbol{y}^j,R^2)^{(e,1)}$, representing points with 0 and 1 at the $e$th bit of $\boldsymbol{b}_i$, respectively. Then $l_{i,e}^j$ is computed as

$$l_{i,e}^j=-\frac{1}{\sigma^2}\Bigg(\min_{\boldsymbol{e}\in\mathcal{D}^{(e,0)}(\boldsymbol{y}^j,R^2)}\|\boldsymbol{y}^j+\boldsymbol{a}-\boldsymbol{e}\|^2 - \min_{\boldsymbol{e}\in\mathcal{D}^{(e,1)}(\boldsymbol{y}^j,R^2)}\|\boldsymbol{y}^j+\boldsymbol{a}-\boldsymbol{e}\|^2\Bigg). \tag{46}$$

Again, the set $\mathcal{D}(\boldsymbol{y}^j,R^2)$ might not fully fall into the VC region, making (46) an approximation of the exact max-log LLRs.

It is worth noting that, when $p_i=i$ for $i=1,\dots,q$ (i.e., the partition chain $\mathbb{Z}^n/2\mathbb{Z}^n/\dots/2^q\mathbb{Z}^n/\Lambda_{\text{s}}$ is considered), setting $R^2=1$ in (46) is sufficient, thanks to (23) and (24) in the hybrid labeling. The computational complexity of the approximated LLR in (46) will be very low since $\mathcal{D}(\boldsymbol{y}^j,1)$ contains only $2n+1$ points. Also, $\mathcal{D}^{(e,0)}(\boldsymbol{y}^j,1)$ or $\mathcal{D}^{(e,1)}(\boldsymbol{y}^j,1)$ can never be an empty set for $i=1,\dots,n$, due to (23) and (24) again.

## V. Performance analysis

In this section, we present the coded BER performance in the AWGN channel for VCs with the three proposed CM schemes introduced in Section IV, and compare them with the most commonly used benchmark at the same rate: Gray-labeled QAM with BICM [19], [20], [44], [48]. In order to see how much gain is actually from shaping, we also apply the proposed hybrid mapping in Section III-D to QAM and combine it with MLCM. This scheme is new, but other types of hybrid mapping for QAM with MLCM exist in the literature [28], [29]. However, optimizing the design of MLCM for QAM in concatenated CM schemes is not the focus of this paper.

For fairness of comparison, $n/2$ 2D QAM formats are multiplexed in the time domain to fill the same number of dimensions and to achieve the same uncoded spectral efficiencies as VCs. The traditional way of realizing non-integer spectral efficiencies for QAM formats is through time-domain hybrid QAM (TDHQ) [49]–[52]. To form an $n$-dimensional TDHQ format, $t_1$ and $t_2$ 2D QAM formats with cardinalities $M_1$ and $M_2$, respectively, are used, satisfying

$$t_1+t_2=\frac{n}{2}$$
$$\beta_{\text{QAM}}=\beta=\frac{t_1\log_2(M_1)+t_2\log_2(M_2)}{t_1+t_2}.$$

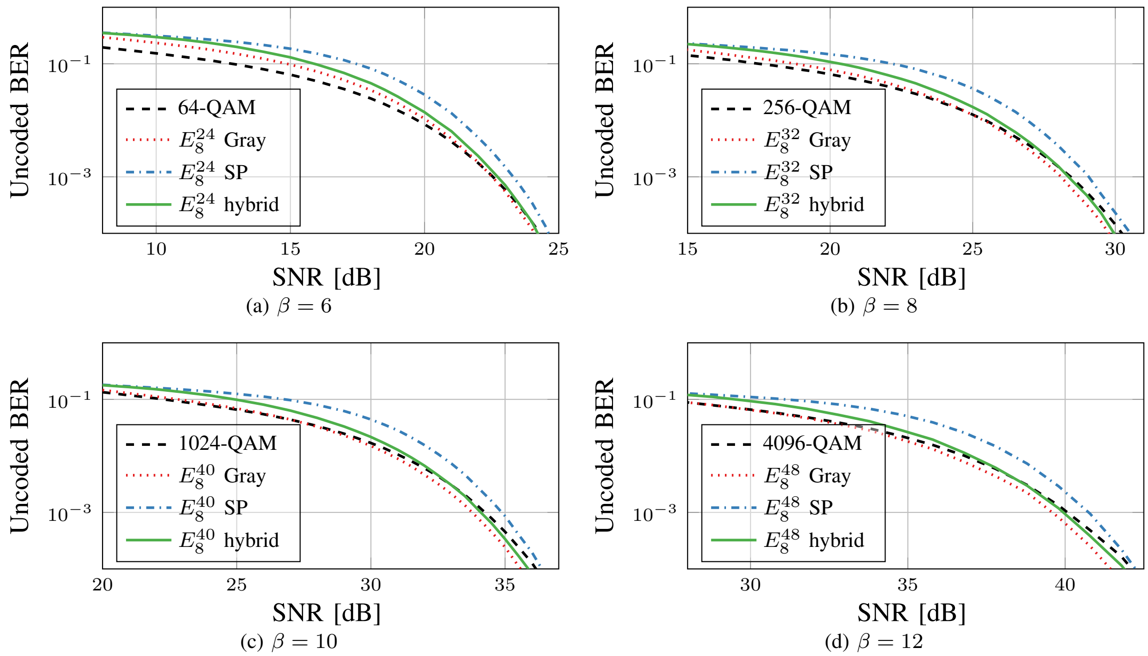
For example, one TDHQ symbol $\boldsymbol{x}$ having the same spectral efficiency as $\Lambda_{16}^{92}$ ($\beta=11.5$ [bits/2D-symbol]) consists of $t_1=4$ 4096-QAM and $t_2=4$ 2048-QAM symbols:

$$\boldsymbol{x}=(\underbrace{\boldsymbol{x}_1,\boldsymbol{x}_2,\boldsymbol{x}_3,\boldsymbol{x}_4}_{\in\text{4096-QAM}},\underbrace{\boldsymbol{x}_5,\boldsymbol{x}_6,\boldsymbol{x}_7,\boldsymbol{x}_8}_{\in\text{2048-QAM}}).$$

The two constituent QAM constellations are scaled to the same minimum distance, which maximizes the minimum distance

This article has been accepted for publication in IEEE Transactions on Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCOMM.2025.3554118





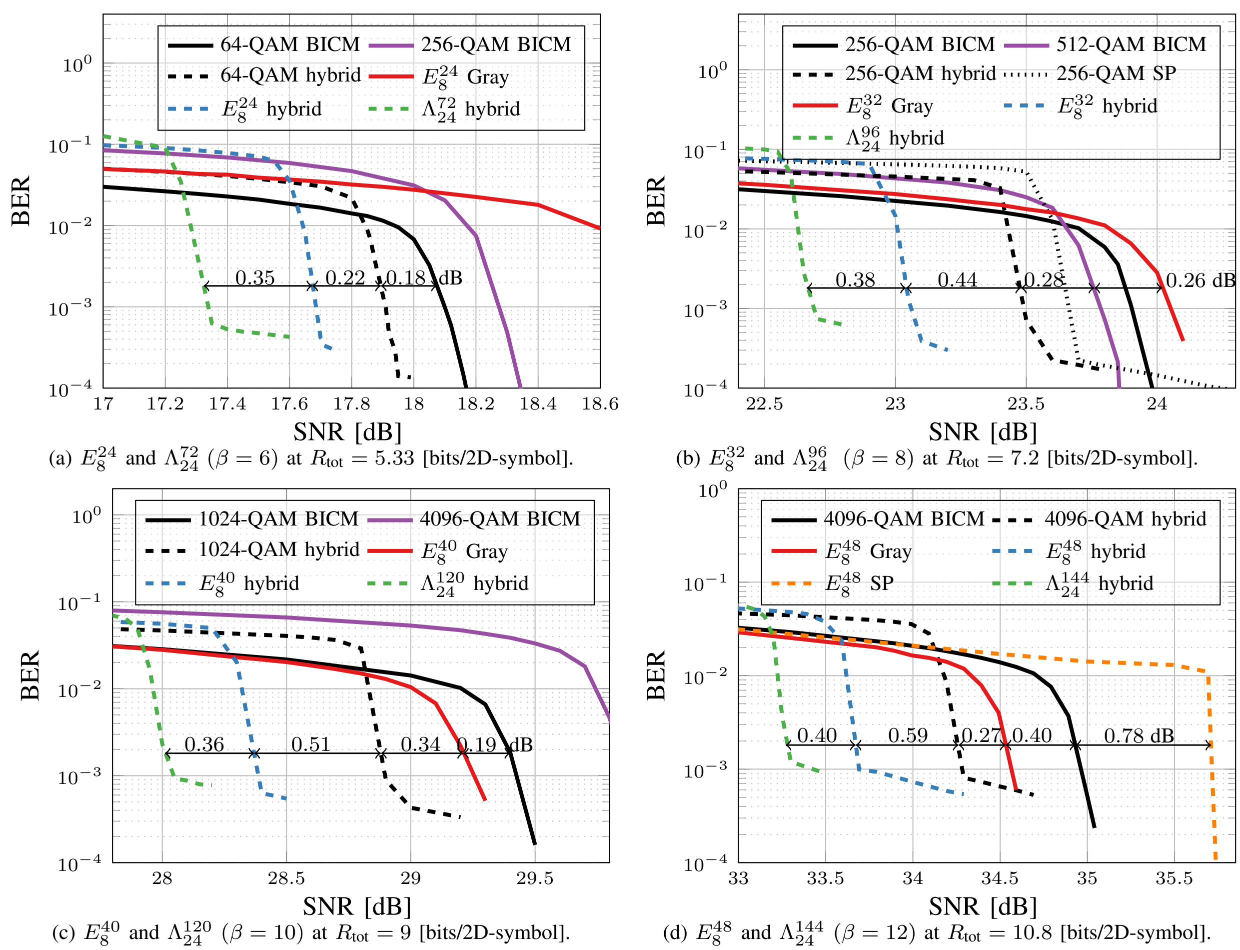

(a) $\beta = 6$

(b) $\beta = 8$

(c) $\beta = 10$

(d) $\beta = 12$

Fig. 4: Uncoded BER performance of 8D VCs compared with QAM at the same spectral efficiency.




(a) $E_8^{24}$ and $\Lambda_{24}^{72}$ ($\beta = 6$) at $R_{\text{tot}} = 5.33$ [bits/2D-symbol].

(b) $E_8^{32}$ and $\Lambda_{24}^{96}$ ($\beta = 8$) at $R_{\text{tot}} = 7.2$ [bits/2D-symbol].

(c) $E_8^{40}$ and $\Lambda_{24}^{120}$ ($\beta = 10$) at $R_{\text{tot}} = 9$ [bits/2D-symbol].

(d) $E_8^{48}$ and $\Lambda_{24}^{144}$ ($\beta = 12$) at $R_{\text{tot}} = 10.8$ [bits/2D-symbol].

Fig. 5: Coded BER performance of 8D and 24D VCs compared with QAM at the same total rate. Solid lines represent BICM performance and dashed lines represent MLCM performance with hybrid mapping. The LLRs of VCs with BICM are calculated using (36) with $R^2 = 6$ and $r = 20$; the LLRs of VCs with MLCM and hybrid mapping are calculated using (46) with $R^2 = 1$; the LLRs of VCs with MLCM and SP mapping are calculated using (42).

TABLE VI: Estimated Gray penalties of 8D VCs with different labeling schemes.

| VC | Gray | SP | Hybrid |
|---|---|---|---|
| $E_8^{24}$ | 1.63 | 4.57 | 2.12 |
| $E_8^{32}$ | 1.31 | 4.24 | 1.81 |
| $E_8^{40}$ | 1.56 | 4.10 | 1.66 |
| $E_8^{48}$ | 1.08 | 4.01 | 1.58 |

of the resulting hybrid QAM constellation for a given $n$-dimensional symbol energy $E_{\mathrm{s}}$ [53, Ch. 4.3].

VCs show high uncoded BER gains at high dimensions and spectral efficiencies [12, Fig. 5]. Thus, we investigate the performance of 8D, 16D, and 24D VCs with high spectral efficiencies of up to 12 bits/2D-symbol. The parameters of the considered VCs and the benchmark QAM formats are listed in Table IV.

### A. Uncoded BER

Fig. 4 shows the uncoded BER for 8D VCs with three different mapping rules, compared with Gray-labeled QAM. For VCs in uncoded systems, the Gray mapping has the lowest uncoded BER among the three mappings and achieves an increasing SNR gain over QAM as $\beta$ increases, which implies that VCs with Gray mapping can outperform QAM in systems with a single HD FEC code [12, Fig. 5]. The hybrid mapping has marginal SNR gains over QAM at high SNRs, since the penalty of a non-Gray mapping for the VC almost counteracts its shaping gains. The SP mapping yields the worst performance and shows no gain over Gray-labeled QAM due to not efficient labeling.

To evaluate the efficiency of different labeling schemes for VCs, one could estimate the Gray penalty defined as the average number of different bits per pair of adjacent symbols [54], [55] by [11, Alg. 5]. Table VI lists the estimated Gray penalties $G_{\mathrm{p}}$ of the considered 8D VCs. By definition, the Gray penalty cannot be smaller than 1, and higher values indicate greater penalties, corresponding to worse uncoded BER performance at asymptotically high SNRs. The hybrid mapping reduces the high $G_{\mathrm{p}}$ of the SP mapping. The Gray penalties of the Gray and hybrid mapping decreases as the spectral efficiency increases, which explains why the SNR gains observed in Fig. 4 increase with the spectral efficiency.

### B. BER after SD FEC decoding

Apart from the uncoded BER discussed in Section V-A, the MI and GMI defined in Section IV-A serve as fundamental limits of a CM scheme, thus are the other two commonly used metrics to evaluate different GS methods. The performance gain of a geometrically shaped constellation over QAM is usually described by the reduced required SNR to achieve a certain uncoded BER, MI, or GMI. For example, in [56, Table I], some published GS methods in optical fiber communication literature are listed and their SNR gains over QAM are compared under these performance metrics. Among these metrics, the MI does not consider any labeling, thus not predicting the SNR gain after SD FEC decoding by itself. The uncoded BER, known as the hard-decision FEC threshold, is shown not accurate enough to predict the BER after SD FEC decoding, which should be replaced by GMI [57]. However, the GMI assumes independent labeling and its predicting accuracy of SNR gains after SD FEC decoding is not guaranteed when Gray labeling is absent or, to the best of our knowledge, has not been investigated. Less work has shown the BER after SD FEC decoding of various proposed GS schemes. In [58], a 0.51 dB SNR gain achieved by GS over 256-QAM has been reported at 6.22 bits/2D-symbol for the AWGN channel. Below, we present the BER after SD FEC decoding of 8D, 16D, and 24D VCs at rates between 5.33 and 10.80 bits/2D-symbol.

The performance of 8D and 24D VCs compared with QAM constellations with both BICM and MLCM in coded systems is shown in Fig. 5. A set of LDPC codes from the digital video broadcasting (DVB-S2) standard [37] with multiple code rates[4] is considered as the inner code. The codeword length is $N = 64800$ and 50 decoding iterations are used. Table V lists the parameters of the considered CM schemes in this paper. For all the VCs with hybrid mapping listed in Table V, $p = q = 1$ and $k_1 = n$. If we target a BER of $1.81 \times 10^{-3}$ when a zipper code [43] is used as the outer code, 8D VCs with MLCM and hybrid mapping yield an increasing SNR gain over QAM with MLCM and hybrid mapping from 0.22 to 0.59 dB as $\beta$ increases. These gains mainly come from shaping. For Gray-labeled QAM with BICM, different combinations of QAM orders and code rates are explored to achieve optimal performance at a specific $R_{\mathrm{tot}}$. For instance, in Fig. 5b, 512-QAM[5] with a $4/5$ code rate outperforms 256-QAM with a $9/10$ code rate. Comparing to QAM with BICM, the most commonly used benchmark, $0.22+0.18 = 0.4$ to $0.59+0.27+0.40 = 1.26$ dB SNR gains are achieved by 8D VCs with MLCM and hybrid mapping. The MLCM achieves SNR gains over BICM due to its effective utilization of FEC overheads to protect the most significant bit levels. However, MLCM has a high error floor at a BER around $10^{-3}$ due to the uncoded bit levels, whereas the BICM scheme does not, as all bit levels are protected by FEC codes.

Fig. 5 also shows that 8D VCs with BICM do not outperform QAM with BICM at $\beta = 6$ and $\beta = 8$, and start to achieve 0.19 and 0.40 dB SNR gains at $\beta = 10$ and $\beta = 12$, respectively. This observation is consistent with [12, Fig. 6, Fig. 9] that the GMI performance of VCs outperform QAM only at high spectral efficiencies.

For 256-QAM with MLCM and SP mapping, the rate allocation affects the performance significantly. As the first 2 bit levels are protected and the other 6 bit levels are uncoded,

[4] The set of all possible code rates in the standard [37] is $\{1/4, 1/3, 2/5, 1/2, 3/5, 2/3, 4/5, 5/6, 8/9, 9/10\}$.

[5] No Gray labeling exists for cross-QAM constellations such as 512-QAM. Here we adopt the "impure" Gray labeling in [54], which is implemented in the MATLAB *qammod* function and has a Gray penalty of $63/61$. The best known cross-QAM labeling is the one proposed for 32-QAM in [59, Fig. 7.53], which generalizes to a 512-QAM labeling with a Gray penalty of $62/61$ using labeling expansion [25, Sec. IV-B].

the MIs of the 8 equivalent subchannels are lower-bounded as

$$\begin{aligned} I(\boldsymbol{Y};\boldsymbol{X}) &= I(\boldsymbol{Y};\boldsymbol{B}_1,\dots,\boldsymbol{B}_8) \\ &\geq I(\boldsymbol{Y};\boldsymbol{B}_1)+I(\boldsymbol{Y};\boldsymbol{B}_2|\boldsymbol{B}_1)+\sum_{k=3}^{8} I(\boldsymbol{Y};\boldsymbol{B}_k|\boldsymbol{B}_1,\boldsymbol{B}_2). \end{aligned} \tag{47}$$

We define the eight conditional MI values of each bit level $I_k$ for $k=1,\dots,8$ as the eight terms in (47) and Fig. 6 shows $I_k$ for $k=1,\dots,8$. From Fig. 6, $I_3$ to $I_8$ have a conditional MI of close to 1 bit/2D-symbol, which means it is reasonable to let these bit levels stay uncoded. Targeting a total rate of 7.2 bits/2D-symbol and given the limited availability of code rates in the DVB-S2 standard, Fig. 7 shows the BER performance of 256-QAM with MLCM and SP mapping under all possible rate allocations for the first two bit levels. Among them, choosing $R_c^1=2/5$ and $R_c^2=4/5$ yields the best performance. This allocation is closest to the "capacity rule" from [27, Sec. IV-A], i.e., choosing the code rates as the MIs of the equivalent subchannels $I_i$ defined in (37). If we look back from the MIs in Fig. 6 at SNR $=$ 24.3 dB, it suggests $R_c^1=2/5$ and $R_c^2=4/5$ approximately according to the capacity rule. The BER of 256-QAM with MLCM and SP mapping and this rate allocation is also presented in Fig. 5b as the black dotted curve, which does not outperform 256-QAM with MLCM and hybrid mapping, although two component codes are used. This is due to the worse uncoded BER performance than the hybrid mapping and possibly not optimal rate allocation under the limited availability of code rates in the DVB-S2 standard.

In Fig. 5d, we show the BER performance of $E_8^{48}$ with MLCM and SP mapping. The partition chain is $\mathbb{Z}^8/D_8/E_8\boldsymbol{R}_8/2\mathbb{Z}^8/E_8^{48}$ with parameters $p=1, q=3, k_1=1, k_2=3$ and $k_3=4$. A bit different from (30) and [27], since we can have multiple bits per partition level, the bits at the same partition level are considered independent of each other, and protected by the same code. Thus, the MIs of the first 8 equivalent subchannels are lower-bounded as

$$I(\boldsymbol{Y};\boldsymbol{B}_1,\dots,\boldsymbol{B}_8) \geq I(\boldsymbol{Y};\boldsymbol{B}_1)+I(\boldsymbol{Y};\boldsymbol{B}_2,\boldsymbol{B}_3,\boldsymbol{B}_4|\boldsymbol{B}_1) + I(\boldsymbol{Y};\boldsymbol{B}_5,\boldsymbol{B}_6,\boldsymbol{B}_7,\boldsymbol{B}_8|\boldsymbol{B}_1,\boldsymbol{B}_2,\boldsymbol{B}_3,\boldsymbol{B}_4) \tag{48}$$

$$\geq I(\boldsymbol{Y};\boldsymbol{B}_1)+\sum_{k=2}^{4} I(\boldsymbol{Y};\boldsymbol{B}_k|\boldsymbol{B}_1)+\sum_{k=5}^{8} I(\boldsymbol{Y};\boldsymbol{B}_k|\boldsymbol{B}_1,\dots,\boldsymbol{B}_4) \tag{49}$$

The eight conditional MI values of each bit level $I_k$ for $k=1,\dots,8$ are defined as the eight terms in (49). Fig. 8 shows the estimated $I_i$ for $i=1,\dots,8$ using the method based on importance sampling proposed in [12], [17]. Bits at the same partition level are protected by the same code. Thus, three different code rates should be used for the lattice partition $\mathbb{Z}^8/D_8/E_8\boldsymbol{R}_8/2\mathbb{Z}^8$. For a bit level with an estimated conditional MI lower than 0.2 bits/2D-symbol, we do not use that subchannel to carry information, and set the code rate to 0. This is equivalent to mapping bits to a subset of constellation points, inherently enabling a larger MSED than using all bit levels at the cost of a lower total rate. If we look at the MIs at SNR =34.8 dB in Fig. 8, the first four bit channels have an MI lower than $1/4$ (the lowest LDPC code rate in the standard [37]). Thus, the first four bits are not used to transmit information with $R_c^1=R_c^2=0$, and $R_c^3=4/5$ according to the capacity rule[6], yielding a total rate of $R_{\text{tot}}=10.8$ bits/2D-symbol. From Fig. 5d, a 0.78 dB SNR loss is observed for $E_8^{48}$ with MLCM and SP mapping compared with 4096-QAM with BICM. This is due to the bad uncoded BER performance for VCs with SP mapping resulting from the high penalty of non-Gray labeling, and the FEC code cannot sufficiently reduce such a high uncoded BER. In addition, the rate allocation might not be globally optimal with the limited choices of code rates. Thus, we do not consider SP mapping for 16D VCs in the following results.

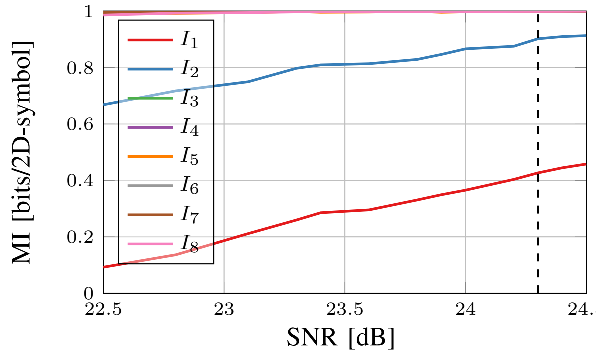



Fig. 6: The MIs of all 8 bit levels for 256-QAM with the SP mapping. The $I_k$ for $k=3,\dots,8$ overlap at approximately 1 bit/2D-symbol for all SNRs.

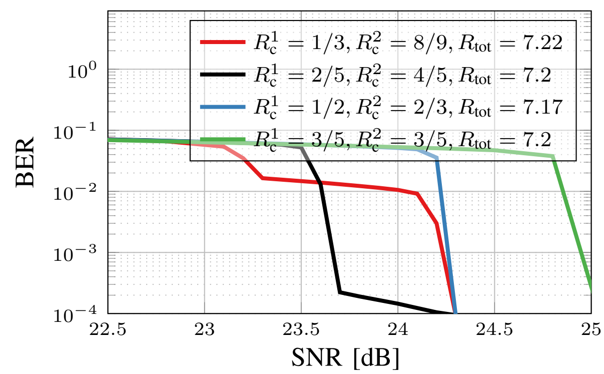



Fig. 7: The BER of 256-QAM with MLCM and SP mapping at different rate allocations for the first two bit levels.

Among the 8D results, VCs with MLCM and hybrid mapping always yield the best performance. In Fig. 5, we also illustrate the performance of 24D VCs with MLCM and hybrid mapping. It shows that 24D VCs can achieve 0.57 to 0.99 dB gains over QAM with MLCM and hybrid mapping at different $\beta$. When compared with QAM with BICM, up to 0.75–1.66 dB gains are achieved by 24D VCs. Larger SNR gains over QAM formats are observed than in the 8D case, since 24D VCs inherently have a higher asymptotic shaping gain than 8D VCs [17, Table I].

For 16D VCs, which have noninteger spectral efficiencies, Fig. 9 presents the uncoded and coded BER performance compared with TDHQ formats. In Fig. 9b, $\Lambda_{16}^{92}$ with MLCM

[6] In order to make sure this is the optimal rate allocation under the limited choices of DVB-S2 code rates, the BER performance with $R_c^1=0$, $R_c^2=1/4$, and $R_c^3=3/5$ yielding $R_{\text{tot}}=10.79$ bits/2D-symbol was also checked and found to be worse than the considered rate allocation.

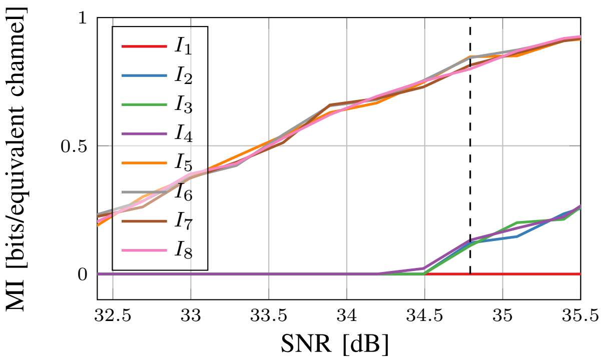



Fig. 8: The MIs of the first 8 bit levels for $E_8^{48}$ with the SP mapping.

and hybrid mapping achieves $0.99$ dB SNR gain over TDHQ2 with MLCM and hybrid mapping. In total, it achieves up to $0.99 + 0.19 + 0.66 = 1.84$ dB SNR gain over TDHQ2 with BICM.

The SNR gains of the proposed CM schemes over QAM with BICM come at the cost of unprotected bit levels, resulting in high error floors at around $10^{-3}$. However, with the outer code, these high error floors have no impact on system performance. This comparison is fair when QAM with BICM also requires an outer code to achieve low BER targets. For instance, an outer code is indispensable in high-speed fiber-optic communications to meet the stringent BER target of $10^{-15}$. On the other hand, in application scenarios where an outer code is not used and higher error floors are acceptable, BICM could be adapted to compare more favorably with the proposed coding schemes.

### C. Achievable information rates (AIRs)

The amount of information per symbol that a given channel can transmit reliably, using a specific modulation format and an encoder/decoder pair, is referred to as an AIR [45]. In [17], the MI of VCs is well investigated, showing large shaping gains at finite SNRs. Fig. 10 illustrates the AIRs of VCs with hybrid mapping calculated as

$$\tilde{I}(\boldsymbol{Y};\boldsymbol{X}) = \sum_{i=1}^{n} I(\boldsymbol{Y};\boldsymbol{B}_i) + \sum_{i=n+1}^{m} I(\boldsymbol{Y};\boldsymbol{B}_i|\boldsymbol{B}_1,\ldots,\boldsymbol{B}_n). \tag{50}$$

Compared with the MIs, the AIRs of the VCs with hybrid mapping almost have no rate loss when they are higher than $0.85\beta$ and start to deviate from the MIs when they are lower than $0.85\beta$. The rates of the VCs and QAM constellations achieved after LDPC decoding at the BER of $1.81 \times 10^{-3}$ from Fig. 5 are shown close to the AIRs.

## VI. Complexity analysis

The complexity of a CM scheme depends on almost every aspect: the number of channel codes used, the FEC overhead, the number of encoded bit levels for each symbol, the LLR calculation complexity, and the complexity of the labeling algorithm. The total complexity of a CM scheme is cumbersome to measure. However, the complexity of each above-mentioned factor can be discussed and the complexity of the dominating factor, the LLR computation, can be quantified. Among the three proposed CM schemes, VCs with MLCM and hybrid mapping have the lowest complexity, as the scheme needs only one FEC code and only some of the bit levels are encoded. Most importantly, the computation of LLRs, which dominates the decoding complexity, is the fastest. In general, VCs with MLCM and SP mapping use more than one component code, which increases the complexity, and estimating the MIs of equivalent subchannels in order to allocate the rates to different bit levels has a high complexity. VCs with BICM also use just one component code and have good performance gains at high $\beta$ thanks to the small loss in the LLR approximation. However, the complexity of the LLR approximation in (34) is higher for VCs with BICM than (42) for the two MLCM schemes for VCs.

The computational complexity of LDPC decoding is dominated by the max-log LLR calculation. Within the max-log LLR calculation, the most computationally demanding aspect is the evaluation of the numerous squared terms. Now we make a comparison in terms of the number of squares required to calculate the max-log LLR between 1) VCs with MLCM and hybrid mapping and 2) QAM constellations with BICM, whose performances were compared in Fig. 5. For an $n$-dimensional VC symbol, the number of required squares per 2D symbol is

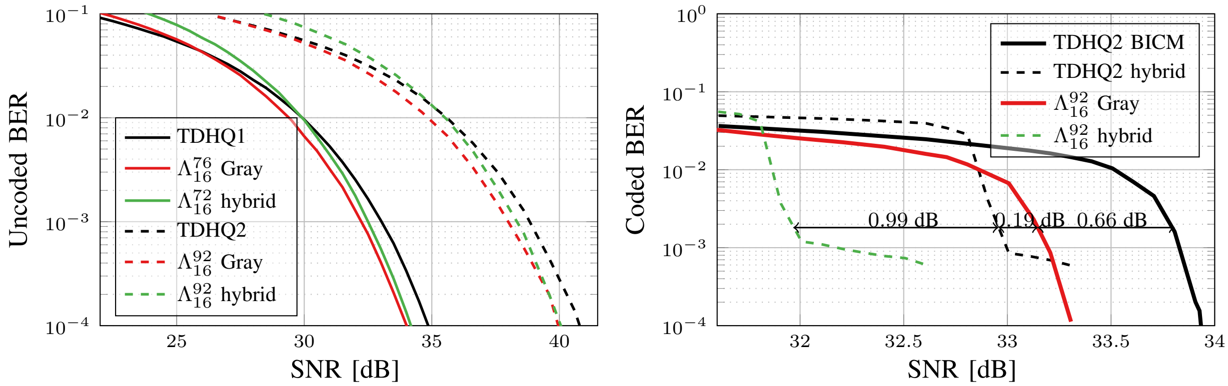
$$N_{\text{squares}} = N_{\text{CB}} \times \text{lookup table size} \times n \times \frac{2}{n}, \tag{51}$$

where $N_{\text{CB}}$ is the number of coded bit levels, the lookup table size for finding the minimum Euclidean distance is $|\mathcal{D}(\boldsymbol{y}^j, 1)| = n + 1$ by (46), the third factor $n$ represents the number of scalar squares for calculating an $n$-dimensional Euclidean distance, and $2/n$ is to normalize $N_{\text{squares}}$ to two dimensions. For QAM, it is well-known that any Gray-labeled square QAM constellation is a Cartesian product of two Gray-labeled PAM constellations. The max-log LLR implementation for square $M$-QAM can be simplified leveraging this property, allowing the minimizations over 2D points in (32) to be replaced with two separate minimizations over one-dimensional points, resulting $\sqrt{M}$ squares. Thus, the total number of squares required for a QAM symbol is $N_{\text{CB}}\sqrt{M}$. Table VII lists the $N_{\text{squares}}$ values for comparison. The decoding complexity of VCs does not increase with higher spectral efficiency, and 8D VCs with MLCM and hybrid mapping offer lower decoding complexity compared to 1024-QAM and 4096-QAM with BICM.

Theoretically, high-cardinality geometrically shaped constellations can provide more shaping gains than VCs [4]. However, usually large-size look-up tables introduce a significant complexity increase in decoding. The $N_{\text{squares}}$ for high-cardinality GS methods can be calculated as in (51). VCs can provide high shaping gains with much lower implementation complexity than unconstrained GS methods.

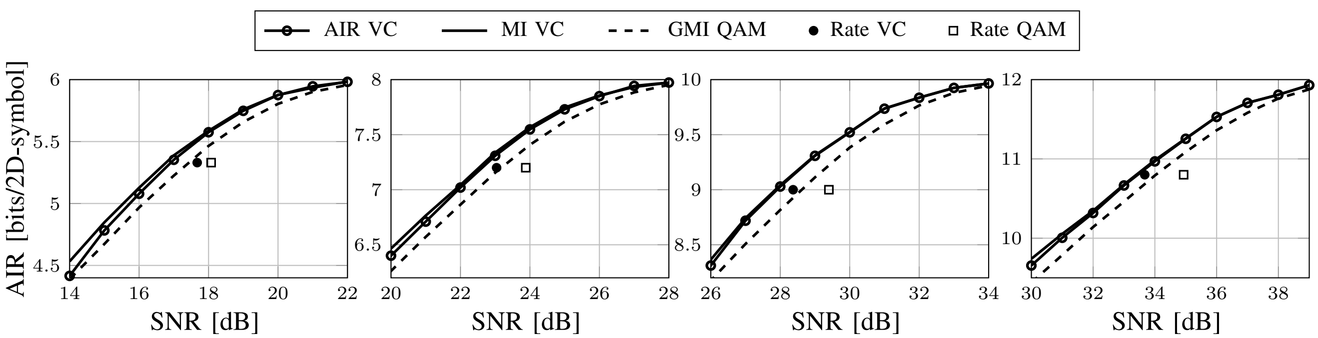
## VII. Conclusion

In this paper, we propose three CM schemes for very large MD VCs, including bit-to-integer mapping algorithms and LLR computation algorithms. This makes very large VCs




(a) $\Lambda_{16}^{76}$ ($\beta = 9.5$) and $\Lambda_{16}^{92}$ ($\beta = 11.5$). (b) $\Lambda_{16}^{92}$ ($\beta = 11.5$) at $R_{\text{tot}} = 10.3$ [bits/2D-symbol].

Fig. 9: Uncoded and coded BER performance of 16D VCs compared with TDHQ formats. Solid lines represent BICM performance and dashed lines represent MLCM performance. The LLRs of VCs with BICM are calculated using (36) with $R^2 = 2$ and $r = 20$; the LLRs of VCs with MLCM and hybrid mapping are calculated using (46) with $R^2 = 1$.




(a) $E_8^{24}$ and 64-QAM (b) $E_8^{32}$ and 256-QAM (c) $E_8^{40}$ and 1024-QAM (d) $E_8^{48}$ and 4096-QAM

Fig. 10: The AIRs and MIs of the considered 8D VCs with hybrid mapping compared with the GMIs of QAM constellations at the same spectral efficiencies.

TABLE VII: The number of required squares per 2D symbol in calculating/approximating the max-log LLR between VCs with MLCM and hybrid mapping and QAM with BICM presented in Fig. 5.

| Constellation | $R_{\text{total}}$ | $N_{\text{CB}}$ | $(n+1)$ or $\sqrt{M}$ | $N_{\text{squares}}$ |
|---|---|---|---|---|
| $E_8^{24}$ | 5.33 | 8 | 72 | 144 |
| $\Lambda_{24}^{72}$ | 5.33 | 24 | 25 | 1200 |
| 64-QAM | 5.33 | 6 | 8 | 48 |
| $E_8^{32}$ | 7.2 | 8 | 9 | 144 |
| $\Lambda_{24}^{96}$ | 7.2 | 24 | 25 | 1200 |
| 256-QAM | 7.2 | 8 | 16 | 128 |
| $E_8^{40}$ | 9 | 8 | 9 | 144 |
| $\Lambda_{24}^{120}$ | 9 | 24 | 25 | 1200 |
| 1024-QAM | 9 | 10 | 32 | 320 |
| $E_8^{48}$ | 10.8 | 8 | 9 | 144 |
| $\Lambda_{24}^{120}$ | 10.8 | 24 | 25 | 1200 |
| 4096-QAM | 10.8 | 12 | 64 | 768 |

adoptable in practical communication systems with SD FEC codes. Among them, one MLCM scheme for VCs with hybrid mapping has even lower decoding complexity than very high-order QAM with BICM. The simulation results for the AWGN channel show that even with some penalty from the non-Gray labeling, VCs achieve high shaping gains over QAM with both BICM and MLCM, especially at high spectral efficiencies. Very recently, the proposed MLCM scheme with hybrid mapping for VCs in this paper has been applied to multi-core fiber transmission to increase the maximum reach [60]. An experimental demonstration in [61] has shown an up to 6 dB launch power gain for the nonlinear four-core fiber channel. In [62], the present shaping gains in this paper have been validated in wideband wavelength-division multiplexing transmission experiments. In [63], a comparison is presented for the first time between the proposed scheme and probabilistic shaping with varying distribution matcher block lengths. The results demonstrate that VCs exhibit notable gains over probabilistic shaping, particularly for short block lengths.

For future works, more applications of the proposed CM schemes can be investigated, e.g., applying other existing channel codes or designing a tailored code. It might be worthwhile to explore code rate adaptation techniques, such as puncturing and shortening, to better demonstrate the adaptability of the proposed schemes.

## References

[1] 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, 1983.

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

[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, 1990.

[4] E. Sillekens, G. Liga, D. Lavery, P. Bayvel, and R. I. Killey, "High-cardinality geometrical constellation shaping for the nonlinear fibre channel," *J. Lightw. Technol.*, vol. 40, no. 19, pp. 6374–6387, 2022.

[5] M. Schaedler, S. Calabrò, F. Pittalà, G. Böcherer, M. Kuschnerov, C. Bluemm, and S. Pachnicke, "Neural network assisted geometric shaping for 800Gbit/s and 1Tbit/s optical transmission," in *Proc. Opt. Fiber Commun. Conf. (OFC)*, San Diego, CA, Mar. 2020.

[6] K. Gümüş, A. Alvarado, B. Chen, C. Häger, and E. Agrell, "End-to-end learning of geometrical shaping maximizing generalized mutual information," in *Proc. Opt. Fiber Commun. Conf. (OFC)*, San Diego, CA, Mar. 2020.

[7] 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, 1982.

[8] 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, 2013.

[9] 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, 2016.

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

[11] S. Li, A. Mirani, M. Karlsson, and E. Agrell, "Designing Voronoi constellations to minimize bit error rate," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Melbourne, Australia, July 2021.

[12] ——, "Power-efficient Voronoi constellations for fiber-optic communication systems," *J. Lightw. Technol.*, vol. 41, no. 5, pp. 1298–1308, 2023.

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

[14] A. Mirani, K. Vijayan, Z. He, S. Li, E. Agrell, J. Schröder, P. Andrekson, and M. Karlsson, "Experimental demonstration of 8-dimensional Voronoi constellations with 65,536 and 16,777,216 symbols," in *Proc. Eur. Conf. Opt. Commun. (ECOC)*, Bordeaux, France, Sep. 2021.

[15] A. Mirani, K. Vijayan, S. Li, Z. He, J. Schröder, P. Andrekson, E. Agrell, and M. Karlsson, "Comparison of physical realizations of multidimensional Voronoi constellations in single mode fibers," in *Proc. Eur. Conf. Opt. Commun. (ECOC)*, Basel, Switzerland, Sept. 2022.

[16] A. Mirani, K. Vijayan, S. Li, Z. He, E. Agrell, J. Schröder, P. Andrekson, and M. Karlsson, "Physical realizations of multidimensional Voronoi constellations in optical communication systems," *J. Lightw. Technol.*, vol. 41, no. 17, pp. 5557–5563, 2023.

[17] S. Li, A. Mirani, M. Karlsson, and E. Agrell, "Low-complexity Voronoi shaping for the Gaussian channel," *IEEE Trans. Commun.*, vol. 70, no. 2, pp. 865–873, 2022.

[18] F. Frey, S. Stern, J. K. Fischer, and R. F. H. Fischer, "Two-stage coded modulation for Hurwitz constellations in fiber-optical communications," *J. Lightw. Technol.*, vol. 38, no. 12, pp. 3135–3146, 2020.

[19] S. Stern, M. Barakatain, F. Frey, J. Pfeiffer, J. K. Fischer, and R. F. H. Fischer, "Coded modulation for four-dimensional signal constellations with concatenated non-binary forward error correction," in *Proc. Eur. Conf. Opt. Commun. (ECOC)*, Brussels, Belgium, Dec. 2020.

[20] S. Stern, M. Barakatain, F. Frey, J. K. Fischer, and R. F. H. Fischer, "Concatenated non-binary coding with 4D constellation shaping for high-throughput fiber-optic communication," in *Signal Process. in Photon. Commun. (SPPCom)*, Washington, D.C., US, July 2021.

[21] A. R. Calderbank and N. J. A. Sloane, "New trellis codes based on lattices and cosets," *IEEE Trans. Inf. Theory*, vol. 33, no. 2, pp. 177–195, 1987.

[22] 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, 1989.

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

[24] E. Agrell and M. Karlsson, "Power-efficient modulation formats in coherent transmission systems," *J. Lightw. Technol.*, vol. 27, no. 22, pp. 5115–5126, 2009.

[25] E. Agrell, J. Lassing, E. G. Ström, and T. Ottosson, "On the optimality of the binary reflected Gray code," *IEEE Trans. Inf. Theory*, vol. 50, no. 12, pp. 3170–3182, 2004.

[26] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inf. Theory*, vol. 28, no. 1, pp. 55–67, 1982.

[27] U. Wachsmann, R. F. H. Fischer, and J. B. Huber, "Multilevel codes: theoretical concepts and practical design rules," *IEEE Trans. Inf. Theory*, vol. 45, no. 5, pp. 1361–1391, 1999.

[28] M. Isaka, R. H. Morelos-Zaragoza, M. P. C. Fossorier, S. Lin, and H. Imai, "Multilevel coded 16-QAM modulation with multistage decoding and unequal error protection," in *Proc. IEEE Glob. Commun. Conf. (GLOBECOM)*, Sydney, NSW, Australia, 1998, pp. 3548–3553.

[29] R. Yuan, J. Fang, R. Xu, B. Bai, and J. Wang, "A hybrid MLC and BICM coded-modulation framework for 6G," in *IEEE Globecom Workshops*, Madrid, Spain, 2021.

[30] L. Beygi, E. Agrell, J. M. Kahn, and M. Karlsson, "Rate-adaptive coded modulation for fiber-optic communications," *J. Lightw. Technol.*, vol. 32, no. 2, pp. 333–343, 2014.

[31] G. D. Forney, Jr., "Coset codes—part I: Introduction and geometrical classification," *IEEE Trans. Inf. Theory*, vol. 34, no. 5, pp. 1123–1151, 1988.

[32] L.-F. Wei, "Trellis-coded modulation with multidimensional constellations," *IEEE Trans. Inf. Theory*, vol. 33, no. 4, pp. 483–501, 1987.

[33] D. Pook-Kolb, E. Agrell, and B. Allen, "The Voronoi region of the Barnes–Wall Lattice $\Lambda_{16}$," *IEEE J. Sel. Areas Commun.*, vol. 4, pp. 16–23, 2023.

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

[35] G. D. Forney, Jr., *Concatenated codes.* Cambridge, MA: MIT Press, 1965.

[36] M. Barakatain, D. Lentner, G. Böecherer, and F. R. Kschischang, "Performance-complexity tradeoffs of concatenated FEC for higher-order modulation," *J. Lightw. Technol.*, vol. 38, no. 11, pp. 2944–2953, 2020.

[37] "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 302 307 V1.2.1 (2009-08), Aug. 2009.

[38] Optical Internetworking Forum, "OIF-400ZR-01.0 – implementation agreement 400ZR," https://www.oiforum.com/wp-content/uploads/OIF-400ZR-01.0_reduced2.pdf, March 2020.

[39] S. B. Wicker and V. K. Bhargava, *Reed-Solomon codes and their applications*. John Wiley & Sons, 1999.

[40] W. Wang, Z. Long, W. Qian, K. Tao, Z. Wei, S. Zhang, Z. Feng, Y. Xia, and Y. Chen, "Real-time FPGA investigation of potential FEC schemes for 800G-ZR/ZR+ forward error correction," *J. Lightw. Technol.*, vol. 41, no. 3, pp. 926–933, 2023.

[41] G. D. Forney, Jr., "On decoding BCH codes," *IEEE Trans. Inf. Theory*, vol. 11, no. 4, pp. 549–557, 1965.

[42] B. P. Smith, A. Farhood, A. Hunt, F. R. Kschischang, and J. Lodge, "Staircase codes: FEC for 100 Gb/s OTN," *J. Lightw. Technol.*, vol. 30, no. 1, pp. 110–117, 2011.

[43] A. Y. Sukmadji, U. Martínez-Peñas, and F. R. Kschischang, "Zipper codes: Spatially-coupled product-like codes with iterative algebraic decoding," in *Can. Workshop on Inf. Theory (CWIT)*, 2019, pp. 1–6.

[44] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Trans. Inf. Theory*, vol. 44, no. 3, pp. 927–946, 1998.

[45] 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.

[46] 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, 1998.

[47] H. Imai and S. Hirakawa, "A new multilevel coding method using error-correcting codes," *IEEE Trans. Inf. Theory*, vol. 23, no. 3, pp. 371–377, 1977.

[48] A. Guillén i Fàbregas, A. Martinez, and G. Caire, "Bit-interleaved coded modulation," *Foundations and Trends in Communications and Information Theory*, vol. 5, no. 1–2, pp. 1–153, 2008.

[49] J. Cho and P. J. Winzer, "Probabilistic constellation shaping for optical fiber communications," *J. Lightw. Technol.*, vol. 37, no. 6, pp. 1590–1607, 2019.

[50] Q. Zhuge, X. Xu, M. Morsy-Osman, M. Chagnon, M. Qiu, and D. V. Plant, "Time domain hybrid QAM based rate-adaptive optical transmissions using high speed DACs," in *Proc. Opt. Fiber Commun. Conf. (OFC)*, Anaheim, CA, USA., March 2013.

[51] X. Zhou, L. E. Nelson, P. Magill, R. Isaac, B. Zhu, D. W. Peckham, P. I. Borel, and K. Carlson, "High spectral efficiency 400 Gb/s transmission using PDM time-domain hybrid 32–64 QAM and training-assisted carrier recovery," *J. Lightw. Technol.*, vol. 31, no. 7, pp. 999–1005, 2013.

[52] X. Zhou, L. E. Nelson, and P. Magill, "Rate-adaptable optics for next generation long-haul transport networks," *IEEE Commun. Mag.*, vol. 51, no. 3, pp. 41–49, 2013.

[53] G. L. Stüber, *Principles of Mobile Communication*, 4th ed. Springer, 2017.

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

[55] M. 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.

[56] B. Chen, Y. Lei, G. Liga, Z. Liang, W. Ling, X. Xue, and A. Alvarado, "Geometrically-shaped multi-dimensional modulation formats in coherent optical transmission systems," *J. Lightw. Technol.*, vol. 41, no. 3, pp. 897–910, 2023.

[57] A. Alvarado, E. Agrell, D. Lavery, R. Maher, and P. Bayvel, "Replacing the soft-decision FEC limit paradigm in the design of optical communication systems," *J. Lightw. Technol.*, vol. 33, no. 20, pp. 4338–4352, 2015.

[58] T. H. Lotz, X. Liu, S. Chandrasekhar, P. J. Winzer, H. Haunstein, S. Randel, S. Corteselli, B. Zhu, and D. W. Peckham, "Coded PDM-OFDM transmission with shaped 256-iterative-polar-modulation achieving 11.15-b/s/Hz intrachannel spectral efficiency and 800-km reach," *J. Lightw. Technol.*, vol. 31, no. 4, pp. 538–545, 2012.

[59] A. Graell i Amat and L. Schmalen, "Forward error correction for optical transponders," in *Springer Handbook of Optical Networks*, B. Mukherjee *et al.*, Ed. Cham, Switzerland: Springer Nature, 2020, ch. 7.

[60] C. Zhao, B. Chen, Y. Lei, S. Li, J. Cai, D. Hu, W. Fang, and L. Sun, "Extending the reach of multi-core fiber via Voronoi constellations with concatenated multi-level coding," *Opt. Lett.*, vol. 49, no. 8, pp. 1997–2000, 2024.

[61] C. Zhao, B. Chen, J. Cai, Z. Liang, Y. Lei, J. Xiong, L. Ma, D. Hu, L. Sun, and G. Shen, "Experimental demonstration of 16D Voronoi constellation with two-level coding over 50km four-core fiber," *arXiv preprint arXiv:2407.06662*, 2024.

[62] Z. He, S. Li, P. Andrekson, E. Agrell, M. Karlsson, and J. Schröder, "12.2 bit/s/Hz C-band transmission with high-gain low-complexity 24-dimensional geometric shaping," *J. Lightw. Technol.*, vol. 42, no. 14, pp. 4829–4836, 2024.

[63] Y. Sheng, B. Chen, Y. Lei, J. Deng, J. Xu, M. Fu, Q. Zhuge, and S. Li, "Multidimensional Voronoi constellations vs. short blocklength probabilistic shaping: A comparison for multilevel coding approach," in *Asia Commun. and Photon. Conf. (ACP)*, Beijing, China, Nov. 2024.