

Partition-Based Distribution Matching

Fehenberger, T.; Millar, D.S.; Koike-Akino, T.; Kojima, K.; Parsons, K.

TR2018-023 January 2018

Abstract

Distribution matching is a fixed-length invertible mapping from a uniformly distributed bit sequence to shaped amplitudes and as such plays an important role in the probabilistic amplitude shaping framework. With conventional constant composition distribution matching (CCDM), all output sequences have identical composition. In this paper, we propose partition-based distribution matching (PBDM) where the composition is constant over all output sequences. When considering the desired distribution as a multiset, PBDM corresponds to partitioning this multiset into equal-size subsets. We show that PBDM allows to address more output sequences and thus has lower rate loss than CCDM in all nontrivial cases. By imposing some constraints on the partitioning, a constructive PBDM algorithm is proposed which comprises two parts. A variable-length prefix of the binary data word determines the composition to be used, and the remainder of the input word is mapped with a conventional CCDM algorithm, such as arithmetic coding, according to the chosen composition. For a specific target distribution and a fixed rate loss, we numerically find that PBDM gives a four-fold reduction in block length in comparison to CCDM. Simulations of 64-ary quadrature amplitude modulation over the additive white Gaussian noise channel demonstrate that the block-length saving of PBDM over CCDM for a fixed gap to capacity varies with the signal-to-noise ratio (SNR) and is approximately a factor of 2.5 to 5 at medium and high SNRs, respectively.

arXiv

This work may not be copied or reproduced in whole or in part for any commercial purpose. Permission to copy in whole or in part without payment of fee is granted for nonprofit educational and research purposes provided that all such whole or partial copies include the following: a notice that such copying is by permission of Mitsubishi Electric Research Laboratories, Inc.; an acknowledgment of the authors and individual contributions to the work; and all applicable portions of the copyright notice. Copying, reproduction, or republishing for any other purpose shall require a license with payment of fee to Mitsubishi Electric Research Laboratories, Inc. All rights reserved.

Partition-Based Distribution Matching

Tobias Fehenberger, *Member, IEEE*, David S. Millar, *Member, IEEE*, Toshiaki Koike-Akino, *Senior Member, IEEE*, Keisuke Kojima, *Senior Member, IEEE*, and Kieran Parsons, *Senior Member, IEEE*

Abstract—Distribution matching is a fixed-length invertible mapping from a uniformly distributed bit sequence to shaped amplitudes and as such plays an important role in the probabilistic amplitude shaping framework. With conventional constant-composition distribution matching (CCDM), all output sequences have identical composition. In this paper, we propose partition-based distribution matching (PBDM) where the composition is constant over all output sequences. When considering the desired distribution as a multiset, PBDM corresponds to partitioning this multiset into equal-size subsets. We show that PBDM allows to address more output sequences and thus has lower rate loss than CCDM in all nontrivial cases. By imposing some constraints on the partitioning, a constructive PBDM algorithm is proposed which comprises two parts. A variable-length prefix of the binary data word determines the composition to be used, and the remainder of the input word is mapped with a conventional CCDM algorithm, such as arithmetic coding, according to the chosen composition. For a specific target distribution and a fixed rate loss, we numerically find that PBDM gives a four-fold reduction in block length in comparison to CCDM. Simulations of 64-ary quadrature amplitude modulation over the additive white Gaussian noise channel demonstrate that the block-length saving of PBDM over CCDM for a fixed gap to capacity varies with the signal-to-noise ratio (SNR) and is approximately a factor of 2.5 to 5 at medium and high SNRs, respectively.

Index Terms—Distribution Matching, Probabilistic Amplitude Shaping, Coded Modulation.

I. INTRODUCTION

The combination of high-order modulation, such as quadrature amplitude modulation (QAM), and strong binary codes (such as turbo-codes [1] or low-density parity-check codes [2]) that operate within a fraction of a decibel (dB) of the additive white Gaussian noise (AWGN) channel capacity [3] have become standardized in many digital communication systems. Bit-interleaved coded modulation (BICM) has achieved near universal adoption, due to its low complexity and close-to-optimal performance [4]. Most coded modulation systems employ uniform signaling where each constellation point is sent with equal probability. A method to further increase the information rates is to employ constellation shaping, which gives signal-to-noise-ratio (SNR) improvements of up to 1.53 dB for the AWGN channel [5, Sec. 4.1.3] [6, Sec. IV-B] [7, Sec. IV-B].

Various techniques have been devised to integrate probabilistic shaping into a coded modulation system, see [8, Sec. II] for a review. Recently, probabilistic amplitude shaping (PAS) [8] has been proposed in which the shaping blocks

are placed outside the forward error correction (FEC) encoder and decoder (see Fig. 1). This reverse-concatenation principle allows a low-complexity and powerful integration into existing BICM systems. Since its proposal, PAS has attracted a lot of attention, particularly in fiber-optic communications [9], [10], [11]. We focus on PAS as shaping framework in this paper.

An integral subsystem of a PAS system is the mapping function from the uniformly distributed data bits to shaped amplitudes and its inverse mapping. In [8, Sec. V], constant-composition distribution matching (CCDM) is employed.¹ In simplified terms, CCDM is a fixed-length invertible operation that maps a block of Bernoulli- $\frac{1}{2}$ distributed data bits to a sequence of shaped amplitudes [12]. The constant-composition principle describes that each output sequence must have an identical empirical distribution.

Designing CCDMs suitable for real-time processing is challenging largely for two reasons. Any finite-length DM fundamentally suffers from a rate loss that increases with decreasing length. Hence, it would be beneficial to have CCDM block lengths in the order of 500 shaped output symbols and possibly more, as can be seen from, e.g., [12, Fig. 2]. The most widely used algorithm for implementing CCDM is arithmetic coding [12, Sec. IV], which is an inherently sequential algorithm. The combination of sequential mapping and long block lengths currently makes a real-time implementation of CCDM a highly challenging task, particularly in the context of optical fiber communications where symbol rates may be 30 GBaud or more.

In this work, we examine distribution matching techniques for which the constant-composition property of conventional CCDM is lifted. Non-constant-composition DMs are based on the principle that the *ensemble average* over all output sequences must have the desired composition, as opposed to the CCDM principle that every output has identical composition. While the removal of this constraint enables large gains over CCDM in the range of block lengths where brute-force computation or numerical optimization are feasible, these techniques are impossible in the block-length regime where low absolute rate loss is achievable. For example, a DM with a sequence length of merely 10 symbols and 1.5 bits of entropy for 4 shaped amplitudes will select 2^{15} sequences from a possible 2^{20} . Due to the scaling of this combinatorial problem, it is clear that we need to have a constructive algorithm for generating non-constant-composition distribution matchers.

We propose partition-based distribution matching (PBDM), which forms output sequences that are equal-length parti-

T. Fehenberger was with Mitsubishi Electric Research Laboratories. He is now with Technical University of Munich, Germany. E-mail: tobias.fehenberger@tum.de.

D. S. Millar, T. Koike-Akino, K. Kojima and K. Parsons are with Mitsubishi Electric Research Laboratories. E-mails: millar@merl.com; koike@merl.com; kojima@merl.com; parsons@merl.com.

¹Note that PAS is not restricted to the use of algebraic distribution matchers (DMs) such as CCDM. Potential alternatives are, for example, lookup tables (whose size could become prohibitively large) or shell mapping (which has limited granularity) [5, Sec. 4.3].

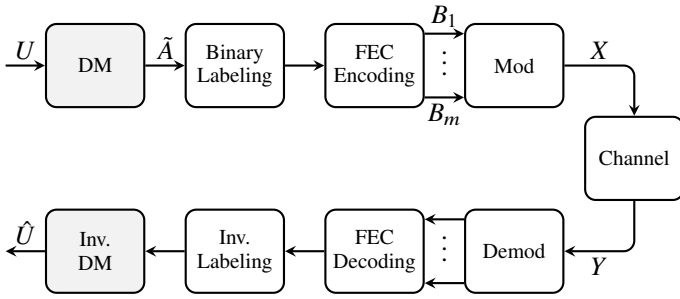


Fig. 1. Diagram of the probabilistic amplitude shaping (PAS) building blocks. This paper studies finite-length distribution matchers (DMs) (gray boxes) that implement an invertible mapping function from the uniform data bits U to the shaped amplitudes \tilde{A} . On the receiver side, an inverse DM undoes this operation such that $U = \hat{U}$ in the case of error-free FEC output. The PAS logic of combining shaped amplitudes with uniform sign bits is explained in detail in [8, Sec. IV].

tionings of a multiset with the desired distribution. In the binary partitioning case, for example, we consider sequences which pairwise follow the target distribution. This set of sequences will have the set of CCDM sequences as a subset, and is therefore a generalization of CCDM. We demonstrate numerically that PBDM requires significantly shorter block lengths than CCDM at a particular rate loss, which is conjectured to ease realization in hardware. PBDM is fundamentally different from product distribution matching [13] where the target distribution is factorized such that parallel CCDMs can be used for constituent distributions of smaller alphabet size. Our approach, in contrast, is more general in that the alphabet size remains unchanged.

For an implementation of PBDM, two constraints on the choice of sequences are imposed. Firstly, pairwise partitioning is deployed where each sequence has a complement such that their average has the desired composition. By further requiring the number of sequences of a particular composition to be a power of 2, PBDM with a binary-tree structure is proposed. This enables implementing PBDM by splitting the binary data word into a prefix that chooses the composition and the payload that is mapped in the conventional CCDM fashion. In this paper, we focus on implementation aspects and performance comparisons of distribution matchers. A numerical analysis finds that pairwise tree-based PBDM has significantly lower rate loss and thus better AWGN performance than conventional CCDM. To the best of our knowledge, the proposed PBDM is the first distribution matcher that lifts the constant-composition principle for a fixed alphabet size.

II. FUNDAMENTALS OF DISTRIBUTION MATCHING

A DM is an injective mapping from a sequence of length k of uniformly distributed data bits U to n shaped amplitudes. Its integration in the PAS framework is shown in Fig. 1. We consider fixed-length block-wise distribution matching only since variable-length DMs have practical disadvantages such as varying buffer sizes. For finite-length DM, the target distribution P_A must be quantized to $P_{\tilde{A}}$ such that the number of occurrences of each amplitude is integer. Following [12], [14], this quantization is carried out to minimize informational

divergence between P_A and $P_{\tilde{A}}$. The resulting amplitudes \tilde{A} have the probability mass function (PMF) $P_{\tilde{A}}$, also referred to as type [15, Sec. 11.1], [16, Sec. II], on the alphabet $\mathcal{A} = \{a_1, \dots, a_{|\mathcal{A}|}\}$.

Consider a DM output sequence $\underline{x}^n = \{x_1, x_2, \dots, x_n\}$ of length n where each element x_j with $j \in 1, \dots, n$ is chosen from \mathcal{A} according to $P_{\tilde{A}}$. The number of occurrences $n(a_i)$ of an amplitude a_i in the sequence \underline{x}^n is

$$n(a_i) = |\{j : x_j = a_i\}|, \quad j \in 1, \dots, n, \quad i \in 1, \dots, |\mathcal{A}|, \quad (1)$$

and we have $\sum_{i=1}^{|\mathcal{A}|} n(a_i) = n$. In the following, we write n_i instead of $n(a_i)$ to indicate the number of occurrences of the i^{th} amplitude a_i . We call the ordered set of occurrences $C = \{n_1, \dots, n_{|\mathcal{A}|}\}$ a *composition*, which has the type $P_{\tilde{A}}$. We say that a sequence has composition C if (1) corresponds to C . The set of unique permutations of \underline{x}^n for a given C is referred to as type class [15, Sec. 11.1], and its size is the multinomial coefficient [15, Eq. (11.17)]

$$M(C) = \binom{n}{n_1, n_2, \dots, n_{|\mathcal{A}|}} = \frac{n!}{n_1! n_2! \dots n_{|\mathcal{A}|}!}. \quad (2)$$

In the CCDM approach [12], each of the 2^k output sequences \underline{x}^n is of type $P_{\tilde{A}}$, and we denote the single typical CCDM output composition as $C_{\text{typ}} = \{nP_{\tilde{A}}(a_1), \dots, nP_{\tilde{A}}(a_{|\mathcal{A}|})\}$. The constant-composition mapping of a uniform input sequence to a sequence that has C_{typ} is denoted as $f_{\text{ccdm}}(C_{\text{typ}})$ and can, for example, be carried out via arithmetic coding [12, Sec. IV]. The number of input bits for CCDM of a particular C_{typ} is given by

$$k = \lfloor \log_2 M(C_{\text{typ}}) \rfloor, \quad (3)$$

where $\lfloor \cdot \rfloor$ denotes rounding down to the closest integer. From (3), we can compute the finite-length rate loss [13, Eq. (1)]

$$R_{\text{loss}} = \mathbb{H}(\tilde{A}) - \frac{k}{n}, \quad (4)$$

where $\mathbb{H}(\cdot)$ denotes entropy in bits. The rate loss vanishes for large n (see [12, Eq. (23)]), which means that an infinite-length CCDM can achieve the target rate $\mathbb{H}(\tilde{A})$. For distribution matching with fixed n and C , it is desirable to make k as large as possible in order to minimize the rate loss. In the following, we introduce a new class of distribution matcher that has a significantly smaller rate loss than a conventional CCDM.

III. PARTITION-BASED DISTRIBUTION MATCHING

A. Principle

PBDM is based on the observation that not every possible DM output sequence \underline{x}^n must be of type $P_{\tilde{A}}$ (or equivalently have the composition C) in order to achieve *on average* the target distribution. As the input bits U are uniformly distributed and a DM establishes an injective mapping, it is by the law of large numbers sufficient if the ensemble average of all output sequences has the target composition. Thus, we choose those output sequences whose compositions C_l satisfy

$$\frac{\sum_l^{N_{\text{comp}}} c_l \cdot C_l}{\sum_l^{N_{\text{comp}}} c_l} \stackrel{!}{=} C_{\text{typ}}, \quad (5)$$

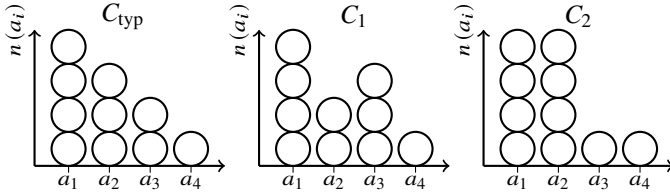


Fig. 2. Illustration of non-constant composition for $|\mathcal{A}| = 4$. Combining one sequence that has C_1 and one that has C_2 gives the target composition C_{typ} .

where l indices the N_{comp} possible compositions of the PBDM output sequences and c_l is the number of occurrences of C_l at the PBDM output, with $0 \leq c_l \leq M(C_l)$. Hence, (5) states that the *average* type of all sequences that are the output of a DM must be $P_{\bar{A}}$. The number of distinct compositions is given by

$$N_{\text{comp}} = \binom{n + |\mathcal{A}| - 1}{n}, \quad (6)$$

which can be proven, for example, with the stars-and-bars technique [17, Sec. II-5].

Example 1 (Non-Constant Composition): Figure 2 shows a set of figures demonstrating the concept of non-constant composition. We have the typical composition $C_{\text{typ}} = \{4, 3, 2, 1\}$ for CCDM with $n = 10$, which gives the entropy $\mathbb{H}(\bar{A}) = 1.85$ bits. The total number of distinct sequences which have this composition is $M(C_{\text{typ}}) = 12600$. This determines the rate of the binary distribution matcher to be $\log_2(\lfloor 12600 \rfloor_2)/10 = 1.3$ bits per symbol. By (4), we have a rate loss R_{loss} of $1.85 - 1.3 = 0.55$ bits for CCDM. By combining one occurrence of a sequence that has $C_1 = \{4, 2, 3, 1\}$ and one of $C_2 = \{4, 4, 1, 1\}$, the average behavior is that of C_{typ} . The number of distinct sequences with C_1 and C_2 are $M(C_1) = 12600$ and $M(C_2) = 6300$, respectively. If, in addition to C_{typ} , these two compositions are used, 6300 additional sequences can each be generated such that (5) is fulfilled. Hence, by considering all three compositions in Fig. 2, we may now use 12600 sequences from C_{typ} ; 6300 sequences from C_1 ; and 6300 from C_2 — 25200 in total. This increases the rate of the non-constant composition distribution matcher by 0.1 bit/symbol and reduces the rate loss from 0.55 bits to 0.45 bits.

Suppose the number of input bits k and the DM output length n is fixed. Then, the accumulated composition of all used output sequences is $C_{\text{acc}} = 2^k \cdot C_{\text{typ}}$. The non-trivial problem is now to find the partitioning of C_{acc} into 2^k integer subsets while fulfilling (5) and obeying two constraints in the subset selection: the sum of the integer elements in each subset must be equal to n in order to have a fixed-length DM, and each subset cannot occur more often than their multinomial coefficient $M(C)$ (see (2)) such that an injective mapping function is established.

Example 2 (General PBDM): Consider a DM with $k = 17$, $n = 10$, and $P_{\bar{A}} = [0.4, 0.3, 0.2, 0.1]$. We have $C_{\text{typ}} = \{4, 3, 2, 1\}$, and the accumulated composition is $C_{\text{acc}} = \{524288, 393216, 262144, 131072\}$. A general PBDM seeks to find those 2^{17} out of the 2^{20} possible DM output sequences whose number of occurrences of each amplitude gives C_{acc} , i.e., that fulfills (5). By (6), there are $N_{\text{comp}} = 286$ composi-

tions for these sequences. The problem is equivalent to finding the non-unique integer sets (each of which corresponds to a particular C) that sum up to C_{acc} , given the constraints that the sum over each set must be $n = 10$ and that each set occurs at most $M(C)$ times.

Many partitioning problems are known to be NP-complete [18, Sec. 3.1.5], yet algorithms giving approximate solutions with reasonable complexity are known for special cases [18, Sec. 4.2]. However, even if a general solution to the partitioning problem could be found, it would remain a challenging task to establish an implementable mapping function between DM input and output sequences, in particular if the DM dimensions prohibit the use of a lookup table. By imposing some structure onto the partitioning, a construction of a PBDM device is made feasible at the expense of a slightly increased rate loss, as we will show next.

B. Pairwise PBDM

To facilitate the implementation of PBDM, we simplify the general partitioning problem (5) by considering pairwise typical sequences only. Note that other structured partitioning schemes, for instance into triples or quadruples, are also possible.² In this pairwise case, we require that for every composition C_l , a complementary composition \bar{C}_l must exist such that

$$C_l + \bar{C}_l = 2 \cdot C_{\text{typ}}. \quad (7)$$

The number of valid pairs that fulfill this relation can be computed by modifying (6) with the inclusion-exclusion method [19, Theorem 4.2], taking into account that certain compositions can never occur in a constrained setting such as the considered pairwise PBDM. The PBDM output sequences that have C_l or \bar{C}_l must have the same probability of occurrence, which implies that the total number of permutations for a pair is governed by the constituent composition that has fewer permutations. We denote the permutation count of a pair $\{C_l, \bar{C}_l\}$ as

$$M(\{C_l, \bar{C}_l\}) = \begin{cases} 2 \cdot \min(M(C_l), M(\bar{C}_l)), & C_l \neq \bar{C}_l, \\ M(C_{\text{typ}}), & C_l = \bar{C}_l, \end{cases} \quad (8)$$

where the first case corresponds to non-degenerate pairs and the latter case is the degenerate CCDM “pair”. For a pairwise PBDM with N_{pairs} distinguishable pairs $\{C_l, \bar{C}_l\}$ that satisfy (7), the total number of permutations is

$$N_{\text{perms}} = \sum_{l=1}^{N_{\text{pairs}}} M(\{C_l, \bar{C}_l\}). \quad (9)$$

Note that $\{C_l, \bar{C}_l\}$ is invariant to permutations of the compositions and switching the two compositions (i.e., $\{\bar{C}_l, C_l\}$ instead of $\{C_l, \bar{C}_l\}$) does not give a new unique pair. The rate loss improvement of PBDM over CCDM (see Sec. IV) is the result of including non-degenerate pairs in (9) in addition to

²With PBDM, $P_{\bar{A}}$ can also be requantized such that the divergence between the target PMF and $P_{\bar{A}}$ is reduced. Potential benefits of this approach remain for future work.

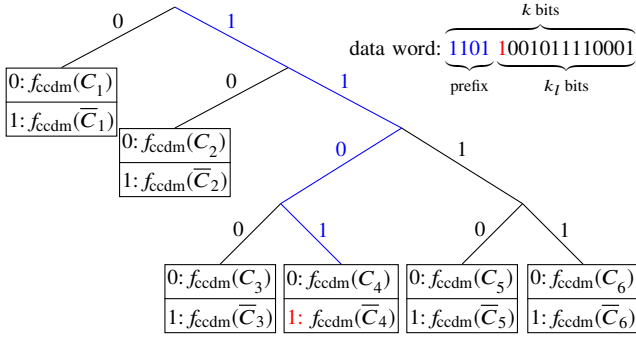


Fig. 3. Illustration of a tree-structured pairwise PBDM with six pairs. The mapping operation for a 17-bit input sequence (top right) to the composition \bar{C}_4 is exemplified. A 4-bit prefix (blue) chooses the pair. The next bit (red) selects the composition within the pair. Mapping the 12-bit payload (black) onto a sequence of shaped amplitudes that has \bar{C}_4 can be carried out with a conventional CCDM algorithm, such as arithmetic coding.

the typical CCDM composition.³ For any DM with binary input, (9) is rounded down to the nearest power of 2, i.e., we have

$$2^k = \lfloor N_{\text{perms}} \rfloor_2. \quad (10)$$

The same requirement is made for binary CCDM in (3).

Example 3 (Pairwise PBDM): Consider $P_A = [0.4415, 0.3209, 0.1654, 0.0722]$ (taken from [12, Example A]) and $n = 10$. We use [14, Algorithm 2] to quantize P_A to $P_{\bar{A}} = [0.4, 0.3, 0.2, 0.1]$, which has $\mathbb{H}(\bar{A}) = 1.85$ bits and $C_{\text{typ}} = \{4, 3, 2, 1\}$, see Example 1. For pairwise PBDM, there are $N_{\text{pairs}} = 49$ pairs (including the degenerate one) that fulfill (7). The new total permutation count is 164214, which increases the number of input bits to $k = 17$ and thus reduces the rate loss to 0.15 bits.

Although considering pairwise-typical compositions greatly simplifies the search for valid partitionings, the implementation of such a pairwise PBDM is not straightforward if a lookup table is not to be used. In the following, we impose another constraint, again at the expense of transmission rate, that enables implementation of PBDM with reasonable complexity.

C. Implementation with a Binary Tree Structure

Any DM with binary input uses $\lfloor N_{\text{perms}} \rfloor_2$ out of N_{perms} output sequences. In order to design an implementable pairwise PBDM, we require, in addition to (7), that the number of permutations $M(\{C_l, \bar{C}_l\})$ of each pair $\{C_l, \bar{C}_l\}$ must be a power of 2. A pair $\{C_l, \bar{C}_l\}$ thus represents an integer k_l bits, and (8) becomes

$$2^{k_l} = \begin{cases} 2 \cdot \min \left(\lfloor M(C_l) \rfloor_2, \lfloor M(\bar{C}_l) \rfloor_2 \right), & C_l \neq \bar{C}_l, \\ \lfloor M(C_{\text{typ}}) \rfloor_2, & C_l = \bar{C}_l. \end{cases} \quad (11)$$

³It is only in the trivial case where C_{typ} has only one non-zero element that the number of permutations of CCDM and PBDM is identical.

The total number of permutations with this power-of-2 constraint is thus

$$2^k = \sum_{l=1}^{N_{\text{pairs}}^\dagger} 2^{k_l}, \quad (12)$$

where N_{pairs}^\dagger out of the initially available N_{pairs} pairs are selected as to maximize N_{perms} (while keeping it a power of 2 to have a binary DM) and thus, to maximize k . The selection of pairs can be done by sorting $\{C_l, \bar{C}_l\}$ according to k_l in descending order and including only the first N_{pairs}^\dagger pairs in that ranked list until k is integer. We note that the constraint (11) can lead to fewer permutations than for unconstrained pairwise PBDM (see (9)) and thus an increased rate loss.

Example 4 (Pairwise PBDM with Binary Tree Structure): Consider the case of Example 3. With (11) and (12), the total number of permutations is computed to be 122688, which gives $k = 16$ input bits and $R_{\text{loss}} = 0.25$ bits, see the marker in Fig. 4. The number of pairs that is necessary to address the 16 bits is $N_{\text{pairs}}^\dagger = 9$ out of the initial $N_{\text{pairs}} = 49$. Note that C_{typ} is not included in these 9 pairs as they already maximize the integer-valued k .

The power-of-2 constraint of (11) allows to implement PBDM in a binary-tree structure as follows. The N_{pairs}^\dagger different pairs are sorted by their k_l in ascending order, and the compositions within a pair are labeled 0 and 1, respectively. Note that this single-bit label is omitted in the special case of the CCDM composition C_{typ} . The two pairs with the smallest k_l (i.e., the least permutations) form a branch, with one element labeled 1 and the other 0. If more than one branch remains, i.e., if there are pairs that have not been used in the tree yet, the merging and labeling process is repeated. When only a single branch remains, the prefix tree is completed. This standard source-coding technique gives an optimal labeling for the prefix.

Once this binary tree is generated, the mapping from k -bit uniform data word to shaped amplitude sequence is done by splitting the PBDM input sequence into three parts. The first $k - k_l$ bits are the prefix that identifies the pair. The next bit chooses the composition within that pair. For the mapping of the final $k_l - 1$ bits onto the shaped symbols according to the selected composition, conventional CCDM based on arithmetic coding can be employed [12, Sec. IV]. This tree-structured design is illustrated in Fig. 3.

At the receiver, inverse PBDM of a shaped sequence x^n must be performed in order to recover the initially transmitted data word. Note that PBDM is designed as an invertible function and hence will not introduce any errors if the input, i.e., the FEC decoder output, is error-free. First, the composition of x^n is determined by, e.g., a simple histogram operation, from which the binary prefix of length $k - k_l + 1$ can be looked up. In order to obtain the remaining $k_l - 1$ payload bits, an inverse CCDM algorithm based on arithmetic coding can be employed. This recovers the transmitted bit sequence.

In the next section, we compare pairwise PBDM with a binary-tree-structured implementation (which we simply refer to as PBDM) to conventional CCDM.

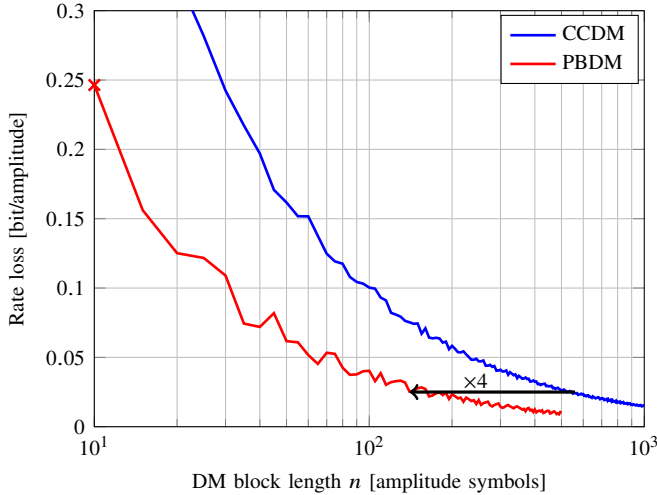


Fig. 4. Rate loss over block length for conventional CCDM and pairwise PBDM implemented with a tree structure. The target PMF is $P_A = [0.0722, 0.1654, 0.3209, 0.4415]$ from [12, Example A]. The marker for PBDM at $n = 10$ refers to Example 4.

IV. NUMERICAL RESULTS

This section numerically studies the rate loss and AWGN performance of pairwise PBDM with the tree-structure design outlined in Sec. III. Rate loss is computed with (4), where the input length k is computed from (3) for CCDM and from (12) for PBDM. For the AWGN channel results, we consider quadrature amplitude modulation (QAM) channel input as the concatenation of two one-dimensional amplitude-shift keying (ASK) symbols. The figure of merit is the achievable information rate (AIR) for bit-metric decoding reduced by the DM rate loss,

$$\text{AIR}_{\text{DM}} = \left[\mathbb{H}(\mathbf{B}) - \sum_{i=1}^m \mathbb{H}(B_i|Y) \right] - R_{\text{loss}}. \quad (13)$$

The derivation of (13) is given in the Appendix. A quantized version of the optimal Maxwell-Boltzmann distribution [20, Sec. IV] for each SNR is used as $P_{\tilde{A}}$. We focus on 64QAM for the AWGN rate analysis, emphasizing that PBDM is feasible with any modulation format that is compatible with PAS.

Figure 4 shows rate loss over block length for the target PMF of [12, Example A]. We observe that the pairwise PBDM achieves a smaller rate loss compared to CCDM for all block lengths. For a rate loss of 0.025 bits per amplitude symbol, PBDM can operate with approximately $n = 140$ symbols, whereas a conventional CCDM requires a fourfold increase in length. Note that the jagged shape of CCDM and PBDM rate loss is due to flooring operations in (3) and (10), respectively.

In Fig. 5, AIR_{DM} in bits per 2D-symbol (bit/2D-sym) is shown over SNR in dB for 64QAM. In addition to the AIRs for PBDM of short ($n = 30$), medium ($n = 100$), and large size ($n = 250$), the AWGN capacity (solid black), an infinite-length DM without rate loss (dotted) and uniform 64QAM (dashed) are included as references. We observe that PBDM with length as small as $n = 30$ has larger AIR than uniform 64QAM over the relevant SNR range. For significantly shorter PBDMs

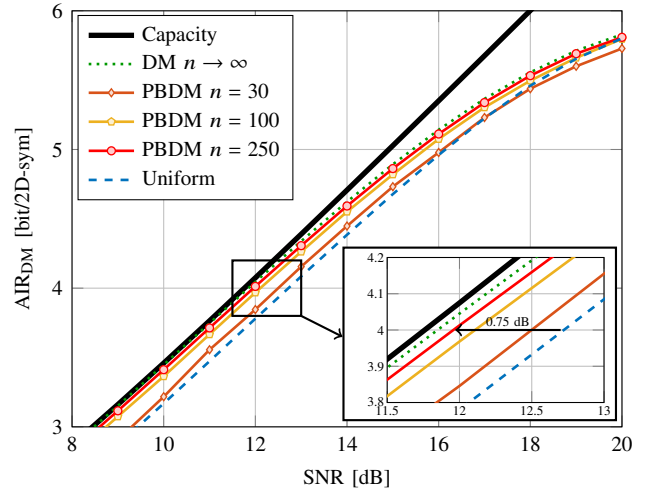


Fig. 5. AIR in bit/2D-sym over SNR in dB for bit-metric decoding 64QAM. The AWGN capacity $\log_2(1 + \text{SNR})$ is shown as reference. The inset zooms into the region around $\text{AIR}_{\text{DM}} = 4$ bit/2D-sym where PBDM of length $n = 250$ is 0.75 dB more power-efficient than uniform 64QAM.

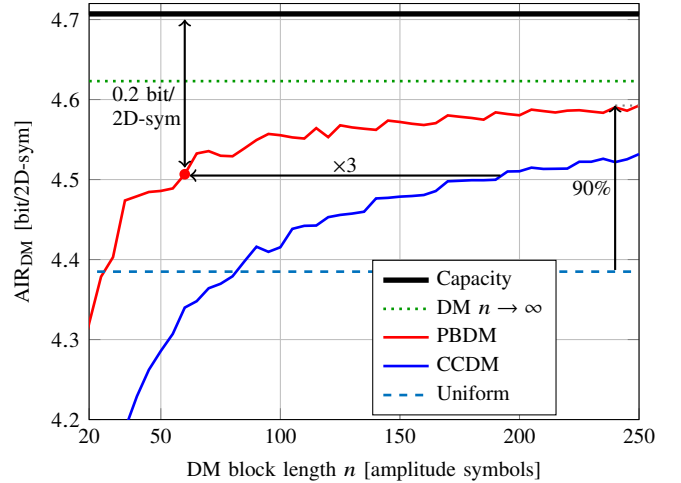


Fig. 6. AIR in bit/2D-sym over the block length n for 64QAM at 14 dB SNR. PBDM with $n = 60$ (marker) operates within 0.2 bit/2D-sym of capacity, whereas a conventional CCDM requires three times the length. At $n = 250$, the PBDM achieves 90% of the maximum available shaping gain at this SNR, which is given by an infinite-length DM.

(not shown), the rate loss is larger than the shaping gain. By increasing n to 250 symbols, the PBDM closely approaches the asymptotic limit.

In Fig. 6, a comparison of PBDM and CCDM as a function of the block length n in 1D amplitude symbols is shown for a fixed SNR of 14 dB. At this SNR level and for 64QAM, an infinite-length CCDM without any rate loss operates approximately 0.1 bit/2D-sym of the AWGN capacity. PBDM with $n = 60$ is able to operate within 0.1 bit/2D-sym of the infinite-length limit and thus within 0.2 bit/2D-sym of capacity. We further note that for $n = 60$, PBDM achieves half of the available shaping gain of 0.24 bit/2D-sym. By increasing the PBDM length to $n = 250$, 90% of the shaping gain are attainable.

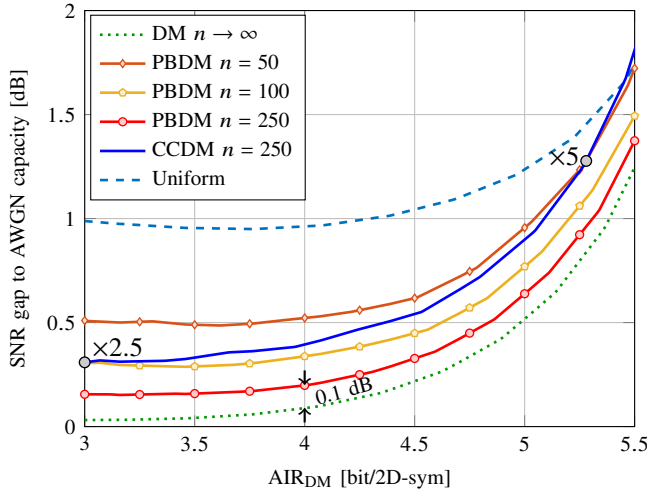


Fig. 7. SNR gap to AWGN capacity in dB over AIR. The gray markers show that the block length reduction of PBDM compared to CCDM is between a factor of 2.5 and 5.

Figure 7 shows the SNR gap to capacity in dB over AIR_{DM} . For all considered rates, a length-250 PBDM operates within approximately 0.1 dB of its asymptotic limit. When comparing CCDM with $n = 250$ to PBDM of various lengths, we observe that the PBDM length reduction is approximately a factor of 2.5 at low AIRs (left gray marker). At $\text{AIR}_{\text{DM}} = 3$ bit/2D-sym, PBDM of length 100 has an SNR gap to capacity of 0.3 dB. When increasing AIR_{DM} , CCDM approaches (and eventually crosses) the PBDM curve of $n = 50$, which corresponds to a fivefold length reduction (right gray marker). For large AIRs beyond 5.5 bit/2D-sym (not shown) where the QAM PMF is close to uniform, the length reduction is up to a factor of 10, indicating that the PBDM benefit depends on how strongly the quantized PMF is shaped. For a heavily shaped distribution, pairwise PBDM only gives few additional permutations over a conventional CCDM. As we have seen from the results in this section, this number and thus the potential input sequence k increases drastically when the PMF is closer to a uniform one, leading to superior performance of PBDM over CCDM.

V. CONCLUSION

We have proposed partition-based distribution matching (PBDM) which generalizes conventional CCDM by lifting the constant-composition property of the distribution matcher output sequences. As a result, PBDM has smaller rate loss for a fixed block length in all relevant cases of distribution matching. By imposing constraints on the choice of partitionings and their number of occurrences, a constructive algorithm for distribution matching and dematching is devised. PBDM is numerically found to allow the block length to be reduced by a factor of 4 for the same rate loss as CCDM. AWGN simulations with 64QAM demonstrate that this reduction depends on the SNR (i.e., the target distribution) and amounts to a factor of 2.5 to 5 for a fixed gap to AWGN capacity.

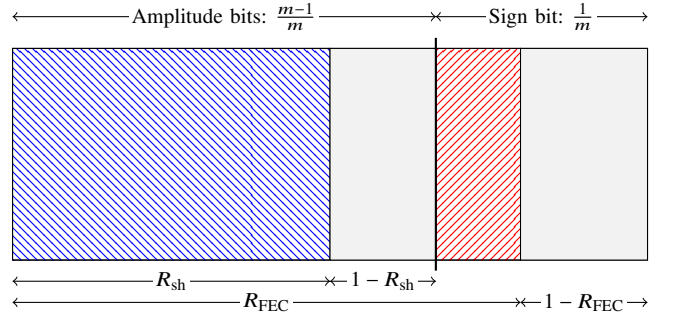


Fig. 8. Composition of an ASK symbol with shaping according to the PAS scheme. The fixed PAS boundary (thick vertical line) between $m - 1$ shaped amplitude and 1 uniform sign bit is shown for 8ASK ($m = 3$). The striped areas show the information in the amplitude bits (blue) and in the sign bit (red). The gray areas represent the redundancy of the shaping code and the FEC code, respectively.

APPENDIX

AIR EVALUATION FOR FINITE-LENGTH DM

The following derivation shows how an achievable information rate (AIR) for a finite-length DM can be computed from the conventionally estimated AIR (assuming infinite-length DM) and the DM rate loss. Both DM and FEC are considered as codes with rates R_{sh} and R_{FEC} , respectively. We then evaluate the information content per ASK symbol for an infinite-length and finite-length DM.

In the PAS scheme, $m - 1$ shaped amplitude bits of each ASK symbol are combined with 1 uniform sign bit. The source of these sign bits can be the uniform data that is to be transmitted or the parity bits of the FEC code. A schematic illustration of such an ASK-symbol composition is given in Fig. (8). The combined striped areas represent the overall amount of information contained in each symbol. This information content I_{PAS} in bits per ASK symbol is

$$I_{\text{PAS}} = \left[\frac{m-1}{m} \cdot R_{\text{sh}} + \left(R_{\text{FEC}} - \frac{m-1}{m} \right) \right] \cdot m, \quad (14)$$

where the first term inside the brackets corresponds to the information contained in the shaped amplitudes (blue striped area in Fig. (8)) and the second term is the information within the sign bits (red striped area), sometimes denoted γ [8, Sec. IV-D]. For PAS, R_{FEC} must be at least $\frac{m-1}{m}$ [8, Sec. IV-B]. The shaping rate R_{sh} in (14) is defined as

$$R_{\text{sh}} = \begin{cases} \frac{\frac{k}{n}}{m-1} & \text{for finite-length DM,} \\ \frac{\mathbb{H}(\hat{A})}{m-1} & \text{for infinite-length DM,} \end{cases} \quad (15)$$

respectively, and thus describes the ratio of information contained in the shaped amplitude bits. With the definitions (14) and (15), we can state a performance measure of a finite-length DM. We define the efficiency η of a finite-length DM as the ratio of I_{PAS} for finite-length and infinite-length DM, i.e.,

$$\eta = \frac{\frac{m-1}{m} \cdot \frac{\frac{k}{n}}{m-1} + \left(R_{\text{FEC}} - \frac{m-1}{m} \right)}{\frac{m-1}{m} \cdot \frac{\mathbb{H}(\hat{A})}{m-1} + \left(R_{\text{FEC}} - \frac{m-1}{m} \right)} \quad (16)$$

$$= \frac{\frac{k}{n} + 1 + m \cdot (R_{\text{FEC}} - 1)}{\mathbb{H}(\tilde{A}) + 1 + m \cdot (R_{\text{FEC}} - 1)}. \quad (17)$$

In the following, we use this shaping efficiency to compute AIRs for finite-length DM.

Consider an AIR in bits per symbol that is computed without including any DM rate loss. A highly relevant AIR for PAS and binary FEC is the bit-metric decoding (BMD) rate R_{BMD} defined as [8, Eq. (63)]

$$R_{\text{BMD}} = \left[\mathbb{H}(\mathbf{B}) - \sum_{i=1}^m \mathbb{H}(B_i|Y) \right], \quad (18)$$

where $\mathbf{B} = [B_1, B_2, \dots, B_m]$ is the binary input vector and Y the channel output (see Fig. 1). The computation of R_{BMD} can, for example, be carried out via numerical integration if the channel law is known, or in Monte Carlo simulations and by mismatched decoding for an unknown channel [21, Sec. VI]. We use numerical integration for the AWGN results in Sec. IV. Note that R_{BMD} according to (18) is achievable only for a DM without rate loss. The BMD rate for PAS and a finite-length DM, referred to as AIR_{DM} , is given by

$$\text{AIR}_{\text{DM}} = \eta \cdot R_{\text{BMD}}. \quad (19)$$

This means that by using a finite-length DM, the information content of each symbol is reduced by η . Note that (19) also holds for AIRs other than R_{BMD} , such as mutual information. To simplify (19), we consider PAS with capacity-achieving codes that operate at their thresholds, in which case we have

$$R_{\text{BMD}} \equiv I_{\text{PAS}}. \quad (20)$$

Solving (14) for the FEC rate then gives

$$R_{\text{FEC}} = \frac{R_{\text{BMD}}}{m} + \left(1 - \frac{\mathbb{H}(\tilde{A})}{m-1} \right) \cdot \frac{m-1}{m}. \quad (21)$$

By inserting (21) into (17) we get

$$\eta = 1 + \frac{\frac{k}{n} - \mathbb{H}(\tilde{A})}{R_{\text{BMD}}}, \quad (22)$$

which, inserted into (19), finally gives a simplified expression for the BMD rate of a finite-length DM,

$$\text{AIR}_{\text{DM}} = R_{\text{BMD}} + \frac{k}{n} - \mathbb{H}(\tilde{A}) \quad (23)$$

$$= R_{\text{BMD}} - R_{\text{loss}}, \quad (24)$$

where R_{loss} was introduced in (4). We use (24) in Sec. IV to compare the AIRs of DMs that have different rate losses, in particular of a conventional CCDM and the proposed PBDM.

REFERENCES

- [1] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. International Conference on Communications (ICC)*. Geneva, Switzerland: vol. 2, pp. 1064–1070, May 1993.
- [2] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low density parity check codes," *IEEE Electronics Letters*, vol. 33, no. 6, pp. 457–458, Mar. 1997.
- [3] S.-Y. Chung, G. D. Forney, T. J. Richardson, and R. Urbanke, "On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit," *IEEE Communications Letters*, vol. 5, no. 2, pp. 58–60, Feb. 2001.
- [4] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Transactions on Information Theory*, vol. 44, no. 3, pp. 927–946, May 1998.
- [5] R. F. H. Fischer, *Precoding and signal shaping for digital transmission*. New York, NY, USA: John Wiley & Sons, 2005.
- [6] G. D. Forney, Jr., R. Gallager, G. R. Lang, F. M. Longstaff, and S. U. Qureshi, "Efficient modulation for band-limited channels," *IEEE Journal on Selected Areas in Communications*, vol. 2, no. 5, pp. 632–647, Sep. 1984.
- [7] G. D. Forney and G. Ungerboeck, "Modulation and coding for linear Gaussian channels," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2384–2415, Oct. 1998.
- [8] G. Böcherer, P. Schulte, and F. Steiner, "Bandwidth efficient and rate-matched low-density parity-check coded modulation," *IEEE Transactions on Communications*, vol. 63, no. 12, pp. 4651–4665, Dec. 2015.
- [9] F. Buchali, F. Steiner, G. Böcherer, L. Schmalen, P. Schulte, and W. Idler, "Rate adaptation and reach increase by probabilistically shaped 64-QAM: An experimental demonstration," *Journal of Lightwave Technology*, vol. 34, no. 7, pp. 1599–1609, Apr. 2016.
- [10] T. Fehenberger, A. Alvarado, G. Böcherer, and N. Hanik, "On probabilistic shaping of quadrature amplitude modulation for the nonlinear fiber channel," *Journal of Lightwave Technology*, vol. 34, no. 22, pp. 5063–5073, Nov. 2016.
- [11] A. Ghazisaeidi, I. F. de Jauregui Ruiz, R. Rios-Muller, L. Schmalen, P. Tran, P. Brindel, A. C. Meseguer, Q. Hu, F. Buchali, G. Charlet *et al.*, "65Tb/s transoceanic transmission using probabilistically-shaped PDM-64QAM," in *Proc. European Conference on Optical Communications (ECOC)*. Düsseldorf, Germany: Paper Th.3.C.4, Sep. 2016.
- [12] P. Schulte and G. Böcherer, "Constant composition distribution matching," *IEEE Transactions on Information Theory*, vol. 62, no. 1, pp. 430–434, Jan. 2016.
- [13] G. Böcherer, P. Schulte, and F. Steiner, "High throughput probabilistic shaping with product distribution matching," *arXiv preprint arXiv:1702.07510*, 2017.
- [14] G. Böcherer and B. C. Geiger, "Optimal quantization for distribution synthesis," *IEEE Transactions on Information Theory*, vol. 62, no. 11, pp. 6162–6172, Sep. 2016.
- [15] T. M. Cover and J. A. Thomas, *Elements of information theory*, 2nd ed. New York, NY, USA: John Wiley & Sons, 2006.
- [16] I. Csiszár, "The method of types," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2505–2523, Oct. 1998.
- [17] W. Feller, *An introduction to probability theory and its applications. Volume 1*, 3rd ed. John Wiley & Sons, 1968.
- [18] M. Garey and D. Johnson, *Computers and intractability: a guide to the theory of NP-completeness*. W. H. Freeman, 1979.
- [19] R.B.J.T. Allenby and Alan Slomson, *How to count: an introduction to combinatorics*, 2nd ed., ser. Discrete mathematics and its applications. CRC Press, 2011.
- [20] F. R. Kschischang and S. Pasupathy, "Optimal nonuniform signaling for Gaussian channels," *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 913–929, May 1993.
- [21] D. Arnold, H.-A. Loeliger, P. Vontobel, A. Kavcic, and W. Zeng, "Simulation-based computation of information rates for channels with memory," *IEEE Transactions on Information Theory*, vol. 52, no. 8, pp. 3498–3508, Aug. 2006.