Parallel Bisection-based Distribution Matching for Probabilistic Shaping

Mengfan Fu, Qiaoya Liu, Xiaobo Zeng, Yiwen Wu, Lilin Yi, Weisheng Hu, and Qunbi Zhuge*
State Key Laboratory of Advanced Optical Communication Systems and Networks, Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, 200240, China
**Corresponding author: qunbi.zhuge@sjtu.edu.cn*

Abstract: We propose a parallel bisection-based distribution matching for constant composition probabilistic shaping. The number of serial operations can be significantly reduced without performance loss, making it a suitable architecture for large block lengths. © 2020 The Author(s)
OCIS codes: (060.2330) Fiber optics communications, (060.4080) Modulation

1. Introduction

Probabilistic shaping (PS) has been widely investigated in coherent optical systems for its shaping gain and the ability to achieve adaptive rate [1, 2]. Distribution matching (DM), which transforms a uniform input bit sequence into a target-distributed output symbol sequence, is a key building block for PS implementation. A popular DM algorithm is constant composition distribution matching (CCDM) [3], whose probability distribution of each amplitude is fixed for all output sequences. The shaping gain of CCDM depends mainly on the block length. A longer block length leads to a larger shaping gain. Arithmetic coding is widely used as a method to implement CCDM. The arithmetic coding has a serial implementation structure which can cause a large latency for a long block length. Besides, the implementation complexity also tends to rapidly increase as the block length increases. Subset ranking CCDM (SR-CCDM) [4] is proposed to reduce the number of serial operations, but the number is still too large for a long block length. Hierarchical distribution matching (HiDM) [5] is proposed to reduce both the complexity and latency by using hierarchical look-up tables (LUTs) with small sizes but at the expense of performance loss.

In this paper, we propose a novel parallel bisection-based scheme to realize both binary and non-binary constant composition distribution matching. The bisection-based CCDM (BS-CCDM) is realized by repetitively dividing a longer composition into two shorter compositions with half the length. And the input bit sequence is correspondingly divided into two shorter input bit sequences. With the parallel architecture, we can reduce the number of serial operations to $\lceil \log_2 n \rceil$, where n is the length of output sequence, leading to a more efficient implementation structure for long block lengths. In addition, it achieves the same performance as other CCDM schemes, which is validated in both simulations and experiments.

2. Bisection-based Constant Composition Distribution Matching (BS-CCDM)

DM as an essential module to implement PS has been studied extensively. However, most of the DM schemes are based on a serial architecture which might cause a large latency and a high complexity especially for long block lengths. We propose the BS-CCDM scheme based on a parallel architecture to realize the CCDM as shown in Fig. 1 (a), and the number of serial operations can be reduced to $\lceil \log_2 n \rceil$.

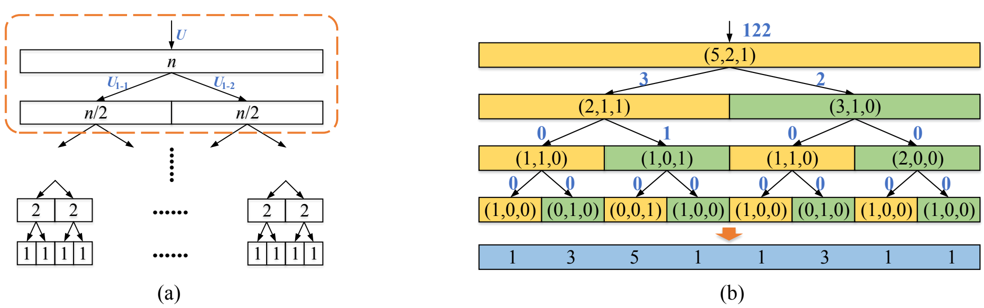

Fig. 1. (a) The parallel architecture of BS-CCDM; (b) An illustrative example of BS-CCDM.

Each input bit sequence with a length of k can be uniquely represented by an integer U ($0 \leq U \leq 2^k - 1$). With the bisection-based architecture, the composition with a sequence length of n can be evenly divided into two parts with a length of $n/2$. The input integer U can also be divided into two numbers and each is used as the input of the corresponding composition in the next layer. Therefore, a mapping with a block length of n turns to two new mappings with a block length of $n/2$. To reduce the latency and complexity, it is necessary to implement the two new mappings in an independent manner. Through the same implementation of each new mapping, we can finally arrive at the last layer which contains n sequences with a length of 1 for each. A critical step of BS-CCDM is to divide the composition into two parts with the same block length and below we will describe this step in detail.

To realize constant composition, the sum of the two sub-compositions with a length of $n/2$ should be equal to the composition with a length of n. For the composition with a length of n, there exists multiple sub-composition pairs $(\{C_1,\overline{C}_1\}, \{C_2,\overline{C}_2\}, \ldots, \{C_m,\overline{C}_m\})$. We need to select one according to the input integer and the permutation numbers of each sub-composition pair. The sub-composition pairs of each composition and their permutation numbers can be stored in a look-up table (LUT). The permutation number of each sub-composition pair is the product of the permutation numbers of two sub-compositions, i.e.,

$$M\left(\{C_i,\overline{C}_i\}\right) = M(C_i)\times M\left(\overline{C}_i\right) \tag{1}$$

where $M(\cdot)$ indicates the permutation number of a composition pair or a composition. The permutation number of a composition can be calculated according to [6]. Then a sub-composition pair is selected by comparing the input integer U and the sum of permutation numbers of sub-composition pairs. When

$$\textstyle\sum_{i=1}^{l-1} M\left(\{C_i,\overline{C}_i\}\right) \le U < \sum_{i=1}^{l} M\left(\{C_i,\overline{C}_i\}\right) \qquad 1 \le l \le m \tag{2}$$

is satisfied, the l-th sub-composition pair $\{C_i,\overline{C}_i\}$ is selected. Then we renew the value of the input integer U by

$$U = U - \textstyle\sum_{i=1}^{l-1} M\left(\{C_i,\overline{C}_i\}\right) \tag{3}$$

The two new input integers are obtained by calculating $U \div M\left(\overline{C}_l\right)$. The quotient and the remainder are the input integers for sub-composition C_l and $\overline{C}_l$, respectively. By the division, the next two mappings are decoupled and can be implemented independently. Through an inverse implementation, we can recover the input bit sequence from the output amplitude sequence at the receiver side.

To better illustrate the above process, an illustrative example is shown in Fig. 1 (b). The output amplitude sequence length is 8 and the composition is (5, 2, 1). The input bit sequence is '1111010', which corresponds to the input integer $U = 122$. The sub-composition pairs and corresponding permutation numbers are listed in Table 1. According to Eq. (2), we can select the 5th sub-composition pair. And according to Eq. (3), the value of U is updated to 14 by subtracting the sum of permutation numbers of the first four sub-composition pairs. By calculating $14 \div 4$, the quotient is 3 and the remainder is 2, which are the input integers of composition (2, 1, 1) and (3, 1, 0), respectively. Finally, the output sequence is obtained as '13511311'.

Table 1. Sub-composition Pairs of Each Composition and the Corresponding Number of Permutations

C	$\{C_i,\overline{C}_i\}$	$M(C_i)$	$M(\overline{C}_i)$	$M(\{C_i,\overline{C}_i\})$	$M(C)$	C	$\{C_i,\overline{C}_i\}$	$M(C_i)$	$M(\overline{C}_i)$	$M(\{C_i,\overline{C}_i\})$	$M(C)$
(5,2,1)	{(4,0,0); (1,2,1)}	1	12	12	160	(2,1,1)	{(2,0,0); (0,1,1)}	1	2	2	12
	{(3,1,0); (2,1,1)}	4	12	48			{(1,1,0); (1,0,1)}	2	2	4	
	{(3,0,1); (2,2,0)}	4	6	24			{(1,0,1); (1,1,0)}	2	2	4	
	{(2,2,0); (3,0,1)}	6	4	24			{(0,1,1); (2,0,0)}	2	1	2	
	{(2,1,1); (3,1,0)}	12	4	48		(3,1,0)	{(2,0,0); (1,1,0)}	1	2	2	4
	{(1,2,1); (4,0,0)}	12	1	12			{(1,1,0); (2,0,0)}	2	1	2	
(1,1,0)	{(1,0,0); (0,1,0)}	1	1	1	2	(1,0,1)	{(1,0,0); (0,0,1)}	1	1	1	2
	{(0,1,0); (1,0,0)}	1	1	1			{(0,0,1); (1,0,0)}	1	1	1	

3. Simulation and Experiment Results

We validated the performance of BS-CCDM and compared it to CCDM with arithmetic coding (AC-CCDM) in a linear channel by Monte Carlo simulations. Only additive white Gaussian noise (AWGN) was added and no digital signal process (DSP) algorithms were employed. We used the generalized mutual information (GMI) as the metric [7]. Various modulation formats, spectral efficiencies (SE) and block lengths were considered.

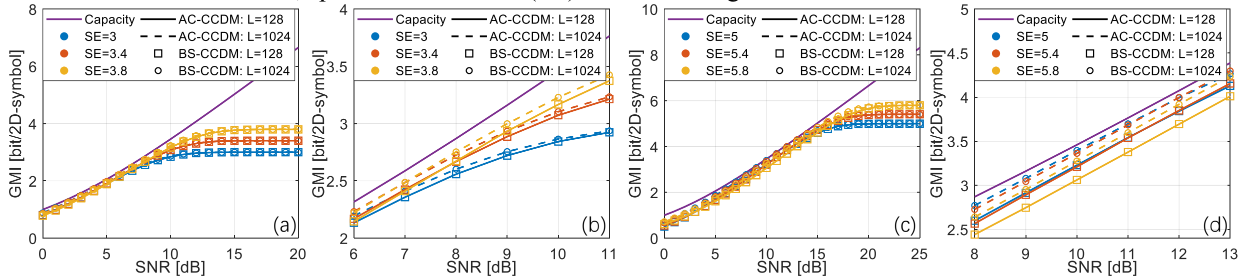

Fig. 2. Performance comparison between BS-CCDM and AC-CCDM: the GMI of (a) 16QAM with SNR from 0 to 20 dB; (b) 16QAM with SNR from 6 to 11 dB; (c) 64QAM with SNR from 0 to 25 dB; (d) 16QAM with SNR from 8 to 13 dB.

Fig. 2 (a) and (c) show the performance of BS-CCDM and AC-CCDM with the same block length. Their performances are exactly the same across the whole signal-to-noise ratio (SNR) range and the curves completely overlap with each other. This is consistent with the theory since they are both based on constant composition and have the same probability distribution. The influence of the block length is shown in Fig. 2 (b) and (d). A larger block length is shown to achieve a better performance. We also find that the performance decreases drastically with a short block length for higher order modulation formats. In particular, the gap between the block length of 128 and 1024 is less than 0.3 dB for 16QAM, but it increases to beyond 0.6 dB for 64QAM. It is because the rate loss due to quantization with a limited block length can be larger when there are more amplitudes. Therefore, a larger block length is required for higher order modulation formats, which have become more essential for higher capacity optical transmissions. In this scenario, the parallel BS-CCDM provides a more suitable architecture for the design of long block length shaping compared to AC-CCDM.

The performance of BS-CCDM was also evaluated by experiments compared to AC-CCDM. The experiment setup is depicted in Fig. 3. At the transmitter, the shaped symbols were generated via BS-CCDM or AC-CCDM and then passed through a root raised cosine (RRC) filter in the offline DSP. Subsequently, the samples were uploaded to an arbitrary waveform generator (AWG) with a sampling rate of 80 GSa/s. Each output signal was driven by a radio frequency (RF) driver followed by a dual-polarization I/Q modulator (DP-I/Q Mod). The adopted tunable laser had a linewidth of 25 kHz and a central frequency of 193.41 THz. The output optical signal was amplified by an Erbium-doped fiber amplifier (EDFA). After 10 km transmission over standard single mode fiber (SSMF), the received optical power (ROP) of the signal was changed by a variable optical attenuator (VOA). After an integrated coherent receiver, the electrical signals were sampled by a 100 GSa/s digital storage oscilloscope (DSO). The digital signals were processed offline and the DSP algorithms were shown in Fig. 3.

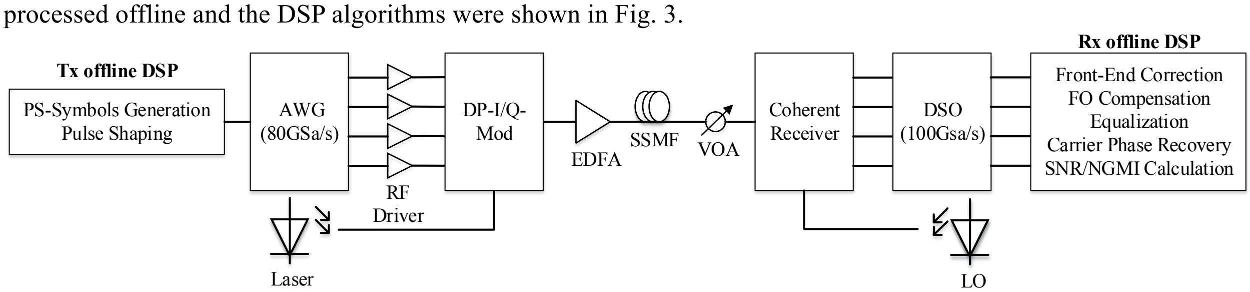

Fig. 3. Experiment setup (FO: frequency offset; LO: local oscillator).

Figure 4 (a) and (b) show the measured GMI of probabilistically shaped 64QAM with a SE of 5.4 bit/symbol and 5.8 bit/symbol, respectively. They both show the curves of AC-CCDM and BS-CCDM with the same block length almost overlap with each other, which validates that BS-CCDM and AC-CCDM have the same performance. And the results are consistent with both the theoretical and numerical results. In addition, the GMI gaps of the two block lengths are also consistent with the simulation results.

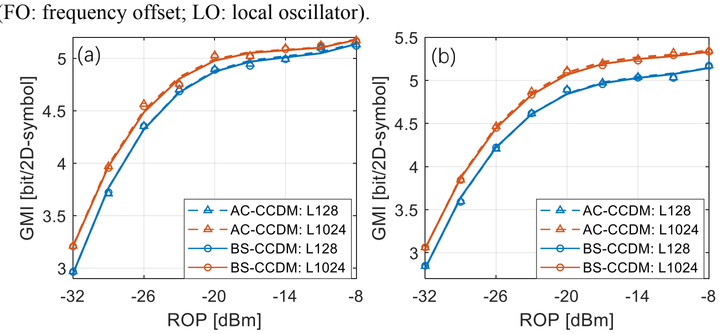

Fig. 4. Experiment results: the GMI of 64QAM with SE of (a) 5.4 bit/symbol; (b) 5.8 bit/symbol

4. Conclusion

In this paper, a parallel scheme based on bisection is proposed to realize CCDM and it can reduce the number of serial operations to $\lceil \log_2 n \rceil$ without sacrificing performance. This parallel architecture is more suitable for implementing CCDM with long block lengths. The performance is validated via both simulations and experiments.

This work was supported by NSFC (61801291), Shanghai Rising-Star Program (19QA1404600) and National Key R&D Program of China (2018YFB1801203).

5. References

[1] G. Böcherer et al., "Bandwidth Efficient and Rate-Matched …," *IEEE Trans. Commun.*, **63**(12), 4651-4665, 2015.
[2] T. Fehenberger et al., "On probabilistic shaping of quadrature …," *J. Lightw. Technol.*, **34**(22), 5063–5073, 2016.
[3] P. Schulte et al., "Constant composition distribution matching," *IEEE Trans. Inf. Theory*, **62**(1), 430–434, 2016.
[4] T. Fehenberger, et al., "Parallel-amplitude architecture …," arXiv, 2019. [Online]. Available: https://arxiv.org/abs/1902.08556.
[5] T. Yoshida et al., "Hierarchical Distribution Matching for …," *J. Lightw. Technol.*, **37**(6), 1579-1589, 2019.
[6] T. Fehenberger, et al., "Multiset-Partition Distribution Matching," *IEEE Trans. Commun.*, **67**(3), 1885-1893, 2019.
[7] A. Alvarado et al., "Replacing the Soft-Decision FEC …," *J. Lightw. Technol.*, **33**(20), 4338-4352, 2015.