

# Sphere Constraint based Enumeration Methods to Analyze the Minimum Weight Distribution of Polar Codes

Jinnan Piao, *Student Member, IEEE*, Kai Niu, *Member, IEEE*, Jincheng Dai, *Member, IEEE*, and Chao Dong, *Member, IEEE*

**Abstract**—In this paper, the minimum weight distributions (MWDs) of polar codes and concatenated polar codes are exactly enumerated according to the distance property of codewords. We first propose a sphere constraint based enumeration method (SCEM) to analyze the MWD of polar codes with moderate complexity. The SCEM exploits the distance property that all the codewords with the identical Hamming weight are distributed on a spherical shell. Then, based on the SCEM and the Plotkin's construction of polar codes, a sphere constraint based recursive enumeration method (SCREM) is proposed to recursively calculate the MWD with a lower complexity. Finally, we propose a parity-check SCEM (PC-SCEM) to analyze the MWD of concatenated polar codes by introducing the parity-check equations of outer codes. Moreover, due to the distance property of codewords, the proposed three methods can exactly enumerate all the codewords belonging to the MWD. The enumeration results show that the SCREM can enumerate the MWD of polar codes with code length up to  $2^{14}$  and the PC-SCEM can be used to optimize CRC-polar concatenated codes.

**Index Terms**—polar codes, concatenated polar codes, sphere constraint based enumeration method, distance spectrum, minimum weight distribution.

## I. INTRODUCTION

**P**OLAR codes have been proved to achieve the capacity by the successive cancellation (SC) decoding as the code length goes to infinity [1]. However, when the code length is small or medium, the performance is unsatisfying. Thus, successive cancellation list (SCL) decoding [2], [3] and successive cancellation stack decoding [4] are introduced to improve the performance of polar codes. Furthermore, the performance is improved by the CRC-aided SCL (CA-SCL) decoding [5] which introduces the CRC detector into the SCL decoding. Thanks to its excellent performance, polar codes have been adopted as the coding scheme for the control channel of the enhanced Mobile Broadband (eMBB) service category in the fifth generation wireless communication systems (5G) [6], [7].

The weight distribution of codewords is the distance spectrum of polar codes, which can be used to evaluate the maximum-likelihood (ML) performance [8]. However, enumerating the distance spectrum has exponential complexity and it is almost impossible for long code length. In the high signal-to-noise ratio (SNR) region, the minimum weight distribution (MWD) is the main factor influencing the ML

performance [9]. Thus, the ML performance can be evaluated by MWD instead of the distance spectrum.

To analyze the MWD of polar codes, Li *et al.* [8] propose an SCL method with excessively large list size to enumerate codewords and analyze MWD. However, due to the large consumption of memory and high complexity, implementing this method on a memory-constrained computer is difficult. Thus, the hard disk is used in [10] to reduce the number of survival paths and decrease the consumption of memory and a multi-level SCL method is proposed to reduce the list size in [11]. Nevertheless, these SCL based methods still have high complexities.

Besides, concatenated polar codes [12], especially CRC-polar concatenated codes [5], have better error performance than polar codes, since the distance spectrum of polar codes is improved by resorting to concatenated schemes. An uniform interleaver approach [13] is proposed to analyze the distance properties of concatenated polar code ensembles, but it cannot obtain the distance spectrum of CRC-polar concatenated codes with definite CRC polynomial. In addition, since this approach enumerates all the codewords, its complexity is extremely high.

In this paper, we exploit the distance property of codewords to exactly evaluate the MWDs of polar codes and concatenated polar codes. The distance property is that the codewords with the identical Hamming weight are distributed on a spherical shell. Hence, a sphere constraint with the minimum Hamming weight can early prune a large amount of unnecessary codewords for analyzing MWD. In addition, the sphere constraint ensures that all the codewords with the minimum weight could be enumerated exactly.

A sphere constraint based enumeration method (SCEM) is first proposed to analyze the MWD of polar codes with moderate complexity. The process of SCEM similar to that of the sphere decoding (SD) algorithm [14]–[17] is regarded as a depth-first tree search, which has negligible memory overhead compared with the SCL based methods. In the SCEM, the sphere constraint with the minimum Hamming weight is used to evaluate the MWD. Thus, the paths satisfying the sphere constraint in the search tree are reserved and the MWD is evaluated exactly. In comparison, the SCL based methods cannot guarantee to enumerate all the codewords with the minimum Hamming weight when the list size is small or medium. Then, although the paths violating the constraint are early pruned to reduce the redundant search, the complexity of SCEM is still too high to evaluate the MWD of long polar codes. Therefore, a sphere constraint based recursive enumeration method (SCREM) is proposed to analyze the MWD with lower complexity on the basis of the SCEM and

This work is supported by National Key R&D Program of China (No. 2018YFE0205501), the National Natural Science Foundation of China (No. 61671080), China Post-Doctoral Science Foundation (No. 2019M660032) and BUPT Excellent Ph.D. Students Foundation (No. CX2019218).

The authors are with the Key Laboratory of Universal Wireless Communications, Ministry of Education, Beijing University of Posts and Telecommunications (BUPT), Beijing 100876, China (email: piaojinnan@bupt.edu.cn, niukai@bupt.edu.cn, daijincheng@bupt.edu.cn, dongchao@bupt.edu.cn).

the Plotkin's construction. Additionally, inspired by the CRC-aided SD (CA-SD) algorithm [17], a parity-check SCEM (PC-SCEM) is proposed to analyze the MWD of concatenated polar codes by introducing the parity-check equations of outer codes.

The main contributions of this paper are summarized as follows:

- 1) We first propose the SCEM to exactly enumerate all the codewords belonging to the MWD by exploits the distance property that all the codewords with the identical Hamming weight are distributed on a spherical shell.
- 2) The SCREM is proposed to recursively analyze the MWD of polar codes with lower complexity compared with the SCEM. In the SCREM, we first prove the property that the MWD of a polar code is related with the MWD of the two component polar codes in terms of the Plotkin's construction. Based on the property, the MWD of the polar code is directly decided without search when the minimum Hamming weight of the two component codes is identical and the complexity of enumerating the MWD of the polar code can be efficiently reduced when the two component codes have different minimum Hamming weight.
- 3) The PC-SCEM is proposed to exactly analyze the MWD of concatenated polar codes by introducing the parity-check equations of outer codes. The parity-check equations are utilized to ensure that all the codewords enumerated are valid codewords. Then, to match the search order of PC-SCEM, Gaussian elimination is used to transform the parity-check equations into new forms. Due to the newly parity-check equations and the sphere constraint, all the codewords belonging to the MWD of concatenated polar codes are exactly enumerated.

The experimental results show that the proposed SCEM and SCREM with code length 128 have up to  $10^4$  and  $10^8$  times lower complexity compared with the SCL methods, respectively. The MWD analysis results show that the SCREM can enumerate the MWD of polar codes with code length up to  $2^{14}$  and the PC-SCEM can analyze the MWD of CRC-polar concatenated codes to optimize the CRC polynomial.

The remainder of the paper is organized as follows. Section II describes the preliminaries of polar codes, SD algorithm and distance spectrum. In Section III, the distance property of codewords and the SCEM are described. The SCREM is provided to recursively analyze the MWD in terms of the Plotkin's construction in Section IV. Section V presents the PC-SCEM to evaluate the MWD of concatenated polar codes. The MWD and the complexity evaluation are provided in Section VI. Section VII concludes this paper.

## II. NOTATIONS AND PRELIMINARIES

### A. Notation Conventions

In this paper, the lowercase letters, e.g.,  $x$ , are used to denote scalars. The bold lowercase letters (e.g.,  $\mathbf{x}$ ) are used to denote vectors. Notation  $\mathbf{x}_i^j$  denotes the subvector  $(x_i, \dots, x_j)$  and  $x_i$  denotes the  $i$ -th element of  $\mathbf{x}$ . The sets are denoted by calligraphic characters, e.g.,  $\mathcal{X}$ , and the notation  $|\mathcal{X}|$  denotes the cardinality of  $\mathcal{X}$ . In addition,  $\mathcal{X} \setminus x$  denotes the set with element  $x$  excluded. The bold capital letters, such as  $\mathbf{X}$ , are

used to denote matrices. The element in the  $i$ -th row and the  $j$ -th column of matrix  $\mathbf{X}$  and the  $i$ -th row of matrix  $\mathbf{X}$  are written as  $x_{i,j}$  and  $\mathbf{X}_i$ , respectively. Furthermore, we write  $\mathbf{F}^{\otimes n}$  to denote the  $n$ -th Kronecker power of  $\mathbf{F}$  and the bit-reversal permutation is denoted by  $\pi(\cdot)$ . Throughout this paper,  $\mathbf{0}$  and  $\mathbf{1}$  mean an all-zero vector and an all-one vector, respectively.

### B. Polar Codes and Concatenated Polar Codes

Polar codes depend on the polarization effect [1] of the matrix  $\mathbf{F} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$ . For an  $(N, K)$  polar code with code length  $N = 2^n$  and code rate  $R = K/N$ , the polarization effect generates  $N$  polarization subchannels. Each subchannel has different reliability and the information bits are transmitted in the  $K$  most reliable subchannels. Therefore, the information set of polar codes defined by  $\mathcal{A}$  with cardinality  $|\mathcal{A}| = K$  is composed of the indices of the  $K$  most reliable subchannels and it is a subset of the index set  $\{1, 2, \dots, N\}$ . Then, the frozen set  $\mathcal{A}^c$  with cardinality  $|\mathcal{A}^c| = N - K$  is a complementary set of  $\mathcal{A}$ . The codeword  $\mathbf{c}$  of polar codes is calculated by  $\mathbf{c} = \mathbf{u}\mathbf{B}\mathbf{G} = \mathbf{v}\mathbf{G}$ , where  $\mathbf{u}$  is an  $N$ -length information sequence,  $\mathbf{B}$  is a bit-reversal permutation matrix,  $\mathbf{G}$  is  $\mathbf{F}^{\otimes n}$  and  $\mathbf{v} = \mathbf{u}\mathbf{B}$ . The information sequence  $\mathbf{u}$  is generated by assigning  $u_i$  to information bit if  $i \in \mathcal{A}$ , and assigning  $u_i$  to 0 if  $i \in \mathcal{A}^c$ . Then, according to  $\mathbf{v} = \mathbf{u}\mathbf{B}$ , an another information set  $\mathcal{B} = \{j | j = \pi(i-1) + 1, i \in \mathcal{A}\}$  is obtained, which means  $v_j$  is an information bit if  $j \in \mathcal{B}$ . Here,  $\pi(\cdot)$  is a bit-reversal permutation.

For an  $(N, K_I)$  concatenated polar code, the inner code is an  $(N, K)$  polar code and the outer code is a  $(K, K_I)$  binary linear block code. The message sequence  $\mathbf{b}$  is first encoded by the binary linear block code to obtain the encoded sequence  $\mathbf{s}$ . Then,  $\mathbf{s}$  is treated as the information bits of an  $(N, K)$  polar code and it is inserted into the information sequence  $\mathbf{u}$  in terms of the information set  $\mathcal{A}$ . Furthermore, a codeword of the concatenated polar code is calculated as  $\mathbf{c} = \mathbf{u}\mathbf{B}\mathbf{G} = \mathbf{v}\mathbf{G}$ .

Without loss of generality, the binary-input additive white Gaussian noise (BI-AWGN) channel and BPSK modulation are considered in this paper. Thus, each coded bit  $c_i \in \{0, 1\}$  is modulated into the transmitted signal by  $x_i = 1 - 2c_i$ . Then, the received sequence is  $\mathbf{y} = \mathbf{x} + \mathbf{n}$ , where  $n_i$  is i.i.d. AWGN with zero mean and variance  $\sigma^2$ .

### C. Sphere Decoding Algorithm

ML decoding of polar codes is equivalent to the following minimization problem

$$\hat{\mathbf{v}} = \arg \min_{\mathbf{x}} \|\mathbf{y} - \mathbf{x}\|^2 = \arg \min_{\mathbf{v}} \|\mathbf{y} - (\mathbf{1} - 2\mathbf{v}\mathbf{G})\|^2, \quad (1)$$

where  $\mathbf{1}$  is an all-one vector of length  $N$ . SD algorithm can solve the problem by enumerating the possible sequence  $\mathbf{v}$  satisfying the sphere constraint

$$m(\mathbf{v}_1^N) \triangleq \|\mathbf{y} - (\mathbf{1} - 2\mathbf{v}\mathbf{G})\|^2 \leq r^2, \quad (2)$$

where  $r$  denotes the radius for the SD search and  $m(\mathbf{v}_1^N)$  is the squared Euclidean distance along with the sequence  $\mathbf{v}_1^N$ .

Noting that  $\mathbf{G}$  is a lower triangular matrix, we can define the partial squared Euclidean distance as

$$m(\mathbf{v}_i^N) \triangleq \sum_{k=i}^N \left| y_k - \left( 1 - 2 \cdot \bigoplus_{j=k}^N (v_j g_{j,k}) \right) \right|^2, \quad (3)$$

which can be computed recursively as

$$m(\mathbf{v}_i^N) = m(\mathbf{v}_{i+1}^N) + \left| y_i - \left( 1 - 2 \cdot \bigoplus_{j=i}^N (v_j g_{j,i}) \right) \right|^2, \quad (4)$$

where  $\mathbf{v}_i^N$  denotes the bit decisions from the  $i$ -th bit to the  $N$ -th bit, and ' $\bigoplus$ ' denotes summation over  $GF(2)$ . According to (4), the SD algorithm can be regarded as a depth-first search on the tree and the search order is from the  $N$ -th bit  $v_N$  to the first bit  $v_1$ . Once a valid sequence  $\mathbf{v}$  satisfying the sphere constraint is found, the radius is updated by  $\sqrt{m(\mathbf{v}_i^N)}$ . To find the ML decoding sequence, SD adaptively updates the radius  $r$ . In this process,  $r$  decreases rapidly so that the ML solution is efficiently captured.

#### D. Distance Spectrum

The distance spectrum of an  $(N, K)$  binary linear block code, designated by  $A_d$ , is the number of codewords of the code with the Hamming weight  $d$ . The pairwise error probability between two codewords modulated by BPSK differing in  $d$  positions and coherently detected in the AWGN channel is  $Q\left(\sqrt{\frac{2dRE_b}{N_0}}\right)$ , where  $E_b$  is the energy of the transmitted bit,  $N_0$  is the one-sided power spectral density of AWGN and

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{t^2}{2}} dt \quad (5)$$

is the probability that a random Gaussian variable with zero mean and unit variance exceeds the value  $x$ . We assume that an all-zero codeword  $\mathbf{0}$  is transmitted to analyze the ML performance. The union bound of ML decoding performance can be written as

$$P_e \leq \sum_{d=d_{\min}}^N A_d Q\left(\sqrt{\frac{2dRE_b}{N_0}}\right). \quad (6)$$

Then, since the MWD (i.e.,  $d_{\min}$  and  $A_{d_{\min}}$ ) is the main factor influencing the ML performance when the  $E_b/N_0$  is large, (6) can be approximated as

$$P_e \approx A_{d_{\min}} Q\left(\sqrt{\frac{2d_{\min}RE_b}{N_0}}\right), \quad (7)$$

where  $d_{\min}$  is the minimum Hamming weight of the linear block code and  $A_{d_{\min}}$  is the number of the codewords with  $d_{\min}$ . In this paper, the approximate union bound (AUB) calculated by (7) is used to evaluate the performance of polar codes.

### III. SPHERE CONSTRAINT BASED ENUMERATION METHOD

In this section, we first illustrate the codewords distribution of polar codes and the idea of SCEM. Then, the detailed description of SCEM is provided on the basis of the codewords distribution.

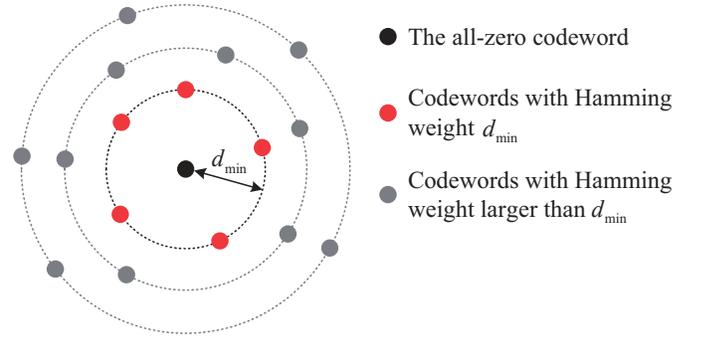


Fig. 1. The description of the codewords distribution with the minimum Hamming weight  $d_{\min}$  in the codeword space.

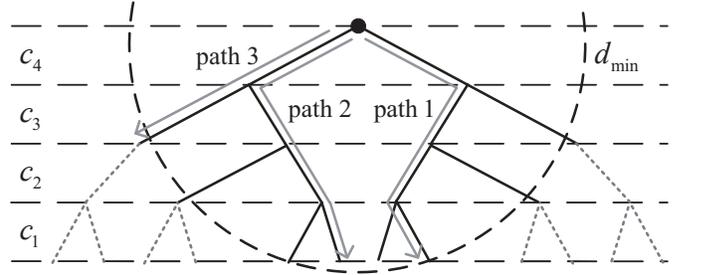


Fig. 2. The illustration of a binary search tree with code length 4 and sphere constraint  $d_{\min}$ .

#### A. An Outline of SCEM

The codewords distribution of polar codes is illustrated in Fig. 1. For a polar code, the codewords with the identical Hamming weight are distributed on a spherical shell. Then, in order to analyze the MWD, the number of all the codewords with the minimum Hamming weight needs to be counted. Based on the codewords distribution, these codewords are covered by a sphere whose radius is the minimum Hamming weight. Thus, a method enumerating all these codewords constrained by the sphere can evaluate the MWD exactly.

SD can find the closest decoded sequence from the received sequence in codeword space under the radius constraint. Inspired by this idea, we propose SCEM to enumerate the codewords under sphere constraint and analyze the MWD. Similar to the SD, SCEM is regarded as a depth-first tree search as well. However, since SCEM is just used to enumerate the codewords, the noise is unnecessary.

Fig. 2 is a toy example to illustrate the process of SCEM. A binary search tree with code length 4 under the sphere constraint  $d_{\min}$  is described in Fig. 2. The branches of the  $i$ -th level in the tree are associated with  $c_{N-i+1}$ . Each path from the root node to a leaf node represents a codeword. In Fig. 2, path 1 and path 2 satisfy the sphere constraint and the two paths are reserved in the search tree. On the contrary, path 3 violates the sphere constraint. Thus, all the paths attached to path 3 are pruned from the search tree, since the corresponding codewords are out of the sphere. Therefore, the proposed SCEM reserves all the codewords satisfying the sphere constraint to analyze the MWD and prunes unnecessary codewords to reduce the redundant search.

In comparison, since the SCL based methods are breadth-

---

**Algorithm 1:** The SCEM method:  $(\mathcal{T}, A_{d_{\min}}) = \text{SCEM}(N, K, \mathcal{B})$

---

**Input:** The code length  $N$ , the information bit length  $K$  and the information set  $\mathcal{B}$ ;

**Output:**  $\mathcal{T}$  is a set composed of the codewords with  $d_{\min}$  and  $A_{d_{\min}}$  is the number of the codewords with  $d_{\min}$ ;

```

1 Initialize  $d_{\min} \leftarrow \min_{i \in \mathcal{B}} (wt(\mathbf{G}_i))$ ,  $\mathcal{T} \leftarrow \emptyset$  and  $A_{d_{\min}} \leftarrow 0$ ;
2 Initialize the index of searching bit  $i \leftarrow N$ ;
3 Initialize  $\mathbf{v} \leftarrow \mathbf{0}$  and  $\mathbf{c} \leftarrow \mathbf{0}$ ;
4 while  $i \leq N$  do
5   if  $i \in \mathcal{B}$  then // information bit
6      $v_i \leftarrow \arg \min_{v_i \in \{0,1\}} (c_i)$  and  $c_i \leftarrow 0$ ;
7   else // frozen bit
8      $v_i \leftarrow 0$  and  $c_i \leftarrow \left( \bigoplus_{j=i}^N (v_j g_{j,i}) \right)$ ;
9   if  $d(\mathbf{v}_i^N) \leq d_{\min}$  then // satisfy the
     sphere constraint
10    if  $i > 1$  then
11       $i \leftarrow i - 1$ ;
12    else
13       $\mathcal{T} \leftarrow \mathcal{T} \cup \{\mathbf{c}\}$  and  $A_{d_{\min}} \leftarrow A_{d_{\min}} + 1$ ;
14      Go to Step 16;
15    else // prune the search tree
16      while  $i \leq N$  do
17        if  $i \in \mathcal{B}$  and  $c_i = 0$  then
18           $v_i \leftarrow v_i \oplus 1$  and  $c_i \leftarrow 1$ ;
19          Go to Step 9;
20        else
21           $i \leftarrow i + 1$ ;
22  $\mathcal{T} \leftarrow \mathcal{T} - \{\mathbf{0}\}$  and  $A_{d_{\min}} \leftarrow A_{d_{\min}} - 1$ ;

```

---

first tree search, a lot of paths need to be stored in the memory, which results in large memory overhead. Moreover, due to no constraint used in the SCL based methods, the unnecessary search is unavoidable. Furthermore, some codewords with the minimum Hamming weight may be lost in the SCL based methods when the list size is not enough.

### B. Detailed Description of SCEM

For an  $(N, K)$  polar code  $\mathcal{C}$ , all the codewords with the Hamming weight  $d_{\min}$  in the codeword space are on the surface of a sphere with radius  $d_{\min}$ . Based on this, SCEM is proposed to enumerate all these codewords and analyze the MWD. The whole procedure is described in Algorithm 1.

The Hamming weight of a codeword  $\mathbf{c}$  is denoted as  $wt(\mathbf{c}) = \sum_{i=1}^N c_i$ . Then, according to [11], the minimum Hamming weight of polar codes is the minimum row weight of generator matrix, i.e.,

$$d_{\min} = \min_{i \in \mathcal{B}} (wt(\mathbf{G}_i)). \quad (8)$$

According to (8), the sphere constraint used in the SCEM is decided, which is

$$wt(\mathbf{c}) \leq d_{\min}. \quad (9)$$

Thus, the codewords satisfying (9) is on the surface of the sphere constraint except  $\mathbf{c} = \mathbf{0}$ .

Then, since  $\mathbf{G}$  is a lower triangular matrix,  $c_i$  is only related to the subvector  $\mathbf{v}_i^N$ , i.e.,

$$c_i = \bigoplus_{j=i}^N (v_j g_{j,i}). \quad (10)$$

Thus, the partial Hamming weight of  $\mathbf{c}$  is defined as

$$d(\mathbf{v}_i^N) \triangleq wt(\mathbf{c}_i^N) = \sum_{k=i}^N \left( \bigoplus_{j=k}^N (v_j g_{j,k}) \right), \quad (11)$$

which can be calculated recursively as

$$d(\mathbf{v}_i^N) = d(\mathbf{v}_{i+1}^N) + \left( \bigoplus_{j=i}^N (v_j g_{j,i}) \right). \quad (12)$$

According to (12), the process of enumerating all the codewords on the surface of the sphere can be treated as a depth-first tree search and the search order is from  $v_N$  to  $v_1$ . Then, when  $v_i$  is decided,  $d(\mathbf{v}_i^N)$  is decided as well. Therefore, the sphere constraint (9) can be simplified as

$$d(\mathbf{v}_i^N) \leq d_{\min}, \quad (13)$$

which means that the Hamming weight of the codewords attached to  $\mathbf{c}_i^N$  is larger than  $d_{\min}$  when (13) is false. Hence, these codewords need to be pruned from the search tree to avoid the useless search.

Algorithm 1 describes the entire procedure of SCEM, where  $N$  is the code length,  $K$  is the information bit length and  $\mathcal{B}$  is the information set about  $\mathbf{v}$ . Without loss of generality, for describing the SCEM easily, we set the branch with  $c_i = 0$  as the first searching branch during deciding information bit  $v_i$ . Then, when the search of the branch with  $c_i = 0$  is completed, SCEM continues to search the branch with  $c_i = 1$ .

In Algorithm 1, the search order is from  $v_N$  to  $v_1$ . Thus, in terms of (12), when a bit  $v_i$  is decided, whether  $d(\mathbf{v}_i^N)$  satisfies the Hamming weight constraint (13) or not is judged. If satisfying, the search continues to decide next bit  $v_{i-1}$  until that a codeword with Hamming weight  $d_{\min}$  is enumerated. If not, the nodes attached to the path  $\mathbf{c}_i^N$  are pruned from the search tree and the search goes on from a new branch of the tree. By repeating the search process, all the  $A_{d_{\min}}$  codewords with Hamming weight  $d_{\min}$  is enumerated and these codewords are recorded into a codeword set  $\mathcal{T}$ . Thus, the MWD  $\mathcal{T}$  is obtained by Algorithm 1, i.e.,

$$\mathcal{T} = \{\mathbf{c} | wt(\mathbf{c}) = d_{\min}, \mathbf{c} \in \mathcal{C}\}. \quad (14)$$

## IV. SPHERE CONSTRAINT BASED RECURSIVE ENUMERATION METHOD

In this section, we first prove the MWD relationship between a polar code and two component polar codes based on the Plotkin's construction. Then, according to the MWD relationship and the SCEM, we design the SCREM.

### A. MWD Relationship Based on the Plotkin's Construction

According to the Plotkin's construction of polar codes [18], a polar code can be divided into two component polar codes. Then, we find that the MWD of the polar code is related with the MWDs of the two component codes and prove the MWD relationship. Based on this, the MWD can be enumerated recursively.

To prove the MWD relationship, we first describe the Plotkin's construction of polar codes as follows. For an  $(N, K)$  polar code  $\mathcal{C}$  with information set  $\mathcal{B}$ , the encoding process can be expressed as

$$\begin{aligned} \mathbf{c} &= \mathbf{v}\mathbf{G} \\ &= (\mathbf{v}', \mathbf{v}'') \begin{bmatrix} \mathbf{G}' & \mathbf{0} \\ \mathbf{G}' & \mathbf{G}' \end{bmatrix} \\ &= (\mathbf{c}' \oplus \mathbf{c}'', \mathbf{c}''), \end{aligned} \quad (15)$$

where  $\mathbf{G}'$  is  $\mathbf{F}^{\otimes(n-1)}$ ,  $\mathbf{c}'$  is  $\mathbf{v}'\mathbf{G}'$  and  $\mathbf{c}''$  is  $\mathbf{v}''\mathbf{G}'$ . Then,  $\mathbf{c}'$  and  $\mathbf{c}''$  are the codewords of an  $(\frac{N}{2}, K')$  polar code  $\mathcal{C}'$  and an  $(\frac{N}{2}, K'')$  polar code  $\mathcal{C}''$ , respectively. The information set of  $\mathcal{C}'$  is denoted by  $\mathcal{B}'$  and

$$\mathcal{B}' = \left\{ i \mid i \in \mathcal{B}, 1 \leq i \leq \frac{N}{2} \right\}. \quad (16)$$

Similarly,  $\mathcal{B}''$  is the information set of  $\mathcal{C}''$  and

$$\mathcal{B}'' = \left\{ i - \frac{N}{2} \mid i \in \mathcal{B}, \frac{N}{2} + 1 \leq i \leq N \right\}. \quad (17)$$

In addition,  $K' = |\mathcal{B}'|$  and  $K'' = |\mathcal{B}''|$ . The minimum Hamming weight of  $\mathcal{C}'$  and  $\mathcal{C}''$  is denoted by  $d'_{\min}$  and  $d''_{\min}$ , respectively.

Then, in order to prove the MWD relationship among  $\mathcal{C}$ ,  $\mathcal{C}'$ , and  $\mathcal{C}''$ , we first prove Lemma 1 and Lemma 2 as follows.

**Lemma 1.**  $\mathcal{C}'$  is a subcode of  $\mathcal{C}''$ .

*Proof:* According to the partial order [19], if  $v_i$  is an information bit,  $v_{i+\frac{N}{2}}$  is also an information bit. Thus, if  $i \in \mathcal{B}'$ , we have  $i \in \mathcal{B}''$ . Therefore,  $\mathcal{C}'$  is a subcode of  $\mathcal{C}''$ , ■

**Lemma 2.**  $d'_{\min}$  and  $d''_{\min}$  have three combinations:

- 1)  $d'_{\min} = d_{\min}$  and  $d''_{\min} = d_{\min}$ .
- 2)  $d'_{\min} = d_{\min}$  and  $d''_{\min} = \frac{d_{\min}}{2}$ .
- 3)  $d'_{\min} > d_{\min}$  and  $d''_{\min} = \frac{d_{\min}}{2}$ .

*Proof:* Due to Lemma 1, we have

$$d'_{\min} \geq d''_{\min}. \quad (18)$$

According to [16, Sec 4.4], we can obtain

$$d_{\min} = \min(2d''_{\min}, d'_{\min}). \quad (19)$$

Supposing  $2d''_{\min} \geq d'_{\min}$ , according to (18) and (19), we have

$$\begin{cases} 2d''_{\min} \geq d'_{\min} \geq d''_{\min} \\ d'_{\min} = d_{\min} \end{cases} \quad (20)$$

Then, we obtain

$$\frac{d_{\min}}{2} \leq d''_{\min} \leq d_{\min}. \quad (21)$$

Furthermore, since

$$wt(\mathbf{G}_i) = 2^j, \exists j \in \{1, 2, \dots, n\}, \quad (22)$$

we have

$$\begin{cases} d'_{\min} = d_{\min} \\ d''_{\min} = \frac{d_{\min}}{2} \text{ or } d_{\min}. \end{cases} \quad (23)$$

Then, supposing  $2d''_{\min} < d'_{\min}$ , similarly, according to (18) and (19), we can obtain that  $d''_{\min}$  is  $\frac{d_{\min}}{2}$  and  $d'_{\min} > d_{\min}$ .

From the above, Lemma 2 has been proved. ■

According to Lemma 1 and Lemma 2, the MWD relationship among  $\mathcal{C}$ ,  $\mathcal{C}'$ , and  $\mathcal{C}''$  is provided as Lemma 3.

**Lemma 3.** Given  $\mathcal{T}$ ,  $\mathcal{T}'$  and  $\mathcal{T}''$  are the codeword sets of  $\mathcal{C}$ ,  $\mathcal{C}'$  and  $\mathcal{C}''$  with Hamming weights  $d_{\min}$ ,  $d'_{\min}$  and  $d''_{\min}$ , respectively.

- 1) When  $d'_{\min} = d_{\min}$  and  $d''_{\min} = d_{\min}$ , we have

$$\mathcal{T} = \mathcal{T}_1 \cup \mathcal{T}_2. \quad (24)$$

- 2) When  $d'_{\min} = d_{\min}$  and  $d''_{\min} = \frac{d_{\min}}{2}$ , we have

$$\mathcal{T} = \mathcal{T}_1 \cup \mathcal{T}_2 \cup \mathcal{T}_3 \cup \mathcal{T}_4. \quad (25)$$

- 3) When  $d'_{\min} > d_{\min}$  and  $d''_{\min} = \frac{d_{\min}}{2}$ , we have

$$\mathcal{T} = \mathcal{T}_3. \quad (26)$$

In (24), (25) and (26),

$$\mathcal{T}_1 = \{(\mathbf{c}', \mathbf{0}) \mid \mathbf{c}' \in \mathcal{T}'\}, \quad (27)$$

$$\mathcal{T}_2 = \{(\mathbf{0}, \mathbf{c}') \mid \mathbf{c}' \in \mathcal{T}'\}, \quad (28)$$

$$\mathcal{T}_3 = \{(\mathbf{c}'', \mathbf{c}'') \mid \mathbf{c}'' \in \mathcal{T}''\}, \quad (29)$$

$$\mathcal{T}_4 = \{(\mathbf{c}' \oplus \mathbf{c}'', \mathbf{c}'') \mid \mathbf{c}' \in \mathcal{T}', \mathbf{c}'' \in \mathcal{T}'', wt(\mathbf{c}' \oplus \mathbf{c}'') = \frac{d_{\min}}{2}\}. \quad (30)$$

*Proof:* See the Appendix. ■

Lemma 3 describes the relationship among  $\mathcal{T}$ ,  $\mathcal{T}'$ , and  $\mathcal{T}''$ . Based on this,  $\mathcal{T}$  can be directly decided by  $\mathcal{T}'$  and  $\mathcal{T}''$ .

### B. Detailed Description of SCREM

In order to exploit Lemma 3 to analyze the MWD of  $\mathcal{C}$ , the MWD of  $\mathcal{C}'$  and  $\mathcal{C}''$  need to be evaluated first. Fortunately, the SCREM can be used to exactly enumerate the MWD of  $\mathcal{C}'$  and  $\mathcal{C}''$ . Then, due to the recursive structure of the generator matrix of polar codes, the MWD of  $\mathcal{C}$  can be enumerated recursively by using the SCREM and Lemma 3.

The SCREM is proposed to enumerate the MWD of polar codes recursively and the method is described as Algorithm 2. In Algorithm 2, polar code  $\mathcal{C}$  is first divided into two component polar codes  $\mathcal{C}'$  and  $\mathcal{C}''$  in terms of the Plotkin's construction (step 1 to 4). Then,  $\mathcal{T}$  can be obtained by  $\mathcal{T}'$  and  $\mathcal{T}''$  on the basis of Lemma 3 (step 10 to 16). Also, SCREM is used to enumerate  $\mathcal{T}'$  and  $\mathcal{T}''$  (step 8 and 9). Hence, the MWD of  $\mathcal{C}$  can be analyzed recursively. In addition, when  $\mathcal{C}$  cannot be divided into two component codes, i.e.,  $N = 2$  or  $K' = 0$ , or the division cannot reduce the search complexity, i.e.,  $K'' = \frac{N}{2}$ , the recursion is stop and the SCREM is used to enumerate the MWD of  $\mathcal{C}$  (step 5 and 6). Thus, according to Lemma 3 and the SCREM, we can obtain  $\mathcal{T}$  recursively.

---

**Algorithm 2:** The SCREM:  $(\mathcal{T}, A_{d_{\min}}) = \text{SCREM}(N, K, \mathcal{B}, d_{\min})$

---

**Input:** The code length  $N$ , the information bit length  $K$ , the information set  $\mathcal{B}$  and the minimum Hamming weight  $d_{\min}$ ;

**Output:**  $\mathcal{T}$  is the codeword set with  $d_{\min}$  and  $A_{d_{\min}}$  is the number of the codewords with  $d_{\min}$ ;

- 1 Initialize  $\mathcal{T} \leftarrow \emptyset$  and  $A_{d_{\min}} \leftarrow 0$ ;
  - 2 Initialize  $\mathcal{B}'$  and  $\mathcal{B}''$  by (16) and (17), respectively;
  - 3 Initialize  $K'$  and  $K''$  by  $|\mathcal{B}'|$  and  $|\mathcal{B}''|$ , respectively;
  - 4  $d'_{\min}$  and  $d''_{\min}$  are the minimum Hamming weight of  $\mathcal{C}'$  and  $\mathcal{C}''$ , respectively;
  - 5 **if**  $N = 2$  **or**  $K' = 0$  **or**  $K'' = \frac{N}{2}$  **then**
  - 6      $(\mathcal{T}, A_{d_{\min}}) \leftarrow \text{SCREM}(N, K, \mathcal{B}, d_{\min})$ ;
  - 7 **else**
  - 8      $(\mathcal{T}', A_{d'_{\min}}) \leftarrow \text{SCREM}(\frac{N}{2}, K', \mathcal{B}', d'_{\min})$ ;
  - 9      $(\mathcal{T}'', A_{d''_{\min}}) \leftarrow \text{SCREM}(\frac{N}{2}, K'', \mathcal{B}'', d''_{\min})$ ;
  - 10    **if**  $d'_{\min} = d_{\min}$  **and**  $d''_{\min} = d_{\min}$  **then**
  - 11        //case 1
  - 11         $\mathcal{T} \leftarrow \mathcal{T}'_1 \cup \mathcal{T}'_2$  and  $A_{d_{\min}} \leftarrow 2A_{d'_{\min}}$ ;
  - 12    **else if**  $d'_{\min} = d_{\min}$  **and**  $d''_{\min} = \frac{d_{\min}}{2}$  **then**
  - 12        //case 2
  - 13        Obtain  $\mathcal{T}_4$  by enumerating all the combinations of  $\mathbf{c}'$  and  $\mathbf{c}''$  which satisfy  $\mathbf{c}' \in \mathcal{T}'$ ,  $\mathbf{c}'' \in \mathcal{T}''$  and  $wt(\mathbf{c}' \oplus \mathbf{c}'') = \frac{d_{\min}}{2}$ ;
  - 14         $\mathcal{T} \leftarrow \mathcal{T}'_1 \cup \mathcal{T}'_2 \cup \mathcal{T}_3 \cup \mathcal{T}_4$  and
  - 14         $A_{d_{\min}} \leftarrow 2A_{d'_{\min}} + A_{d''_{\min}} + |\mathcal{T}_4|$ ;
  - 15    **else if**  $d'_{\min} > d_{\min}$  **and**  $d''_{\min} = \frac{d_{\min}}{2}$  **then**
  - 15        //case 3
  - 16         $\mathcal{T} \leftarrow \mathcal{T}_3$  and  $A_{d_{\min}} \leftarrow A_{d''_{\min}}$ ;
- 

## V. PARITY-CHECK SCEM

In this section, we first describe the parity-check equations and transform them into new forms to match the search order of the PC-SCEM. Then, the detailed description of PC-SCEM is provided.

### A. Parity-Check Equations

For an  $(N, K_I)$  concatenated polar code, the inner code is an  $(N, K)$  polar code and the outer code is a  $(K, K_I)$  binary linear block code. The parity-check matrix and the codeword of the binary linear block code is denoted by  $\mathbf{H}$  and  $\mathbf{s}$ , respectively. Each row of  $\mathbf{H}$  represents a parity-check equation, i.e.,

$$\bigoplus_{j=1}^K h_{i,j} s_j = 0, \quad i = 1, 2, \dots, K_P, \quad (31)$$

where  $K_P = K - K_I$  is the number of the parity-check equations.

Then, parity-check sets are used in this paper to represent the parity-check equations.

---

**Algorithm 3:**  $\{\mathcal{Q}_i(\mathbf{v})\} = \text{Transform}(\{\mathcal{R}_i(\mathbf{v})\})$

---

**Input:** The parity-check sets  $\mathcal{R}_i(\mathbf{v})$ ,  $l = 1, 2, \dots, K_P$ ;

**Output:** Total  $K_P$  transformed parity-check sets  $\mathcal{Q}_i(\mathbf{v})$ ;

- 1 Initialize  $\mathbf{D}$  as a  $K_P \times N$  matrix and the  $i$ -th row of  $\mathbf{D}$  represents the parity-check equation obtained by  $\mathcal{R}_i(\mathbf{v})$ ;
  - 2 Employ GE on the rows of  $\mathbf{D}$  and obtain a row echelon form matrix  $\mathbf{E}$ ;
  - 3 Each row of  $\mathbf{E}$  represents the newly parity-check equation and the corresponding parity-check set is  $\mathcal{Q}_i(\mathbf{v})$ ;
- 

**Definition 1.** The parity-check sets corresponding to the parity-check equations of  $\mathbf{s}$  are given as

$$\mathcal{R}_i(\mathbf{s}) \triangleq \{j | h_{i,j} = 1\}, \quad i = 1, 2, \dots, K_P. \quad (32)$$

Since  $\mathbf{s}$  is inserted into  $\mathbf{u}$  in terms of the information set  $\mathcal{A}$ , the parity-check sets corresponding to  $\mathbf{u}$  are defined as

$$\mathcal{R}_i(\mathbf{u}) = \{t | t = f(j), j \in \mathcal{R}_i(\mathbf{s})\}, \quad i = 1, 2, \dots, K_P \quad (33)$$

where the function  $f(t)$  is the index mapping from  $\mathbf{s}$  to  $\mathbf{u}$  and  $f(t)$  is different for various concatenated polar code schemes. Then, the parity-check sets  $\mathcal{R}_i(\mathbf{v})$  corresponding to  $\mathbf{v}$  are derived by performing the bit-reversal permutation to all the elements in  $\mathcal{R}_i(\mathbf{u})$ , i.e.,

$$\mathcal{R}_i(\mathbf{v}) = \{k | k = \pi(t - 1) + 1, t \in \mathcal{R}_i(\mathbf{u})\}, \quad i = 1, 2, \dots, K_P \quad (34)$$

In terms of the definition of parity-check sets, for any  $i = 1, 2, \dots, K_P$ , we have

$$\bigoplus_{j \in \mathcal{R}_i(\mathbf{s})} s_j = \bigoplus_{t \in \mathcal{R}_i(\mathbf{u})} u_t = \bigoplus_{k \in \mathcal{R}_i(\mathbf{v})} v_k = 0. \quad (35)$$

Since the search order of the PC-SCEM method is from  $v_N$  to  $v_1$ , the bit with the least index in each parity-check set  $\mathcal{R}_i(\mathbf{v})$  can be directly judged by the previous searched bits. Followed by this, the definition of parity-check bit index of the parity-check set is given.

**Definition 2.** The index  $k_i$  of parity-check bit corresponding to  $\mathcal{R}_i(\mathbf{v})$  is defined as  $k_i = \min(\mathcal{R}_i(\mathbf{v}))$ .

Similar to the CA-SD [17], if two more parity-check sets have the same index of parity-check bit, the *colliding decision* phenomenon will happen where this parity-check bit cannot be uniquely judged. This severe problem leads to the search error. However, the above colliding decision problem can be solved by the linear combination of multiple parity-check equations. Thus, to avoid the colliding decision, Gaussian elimination (GE) is used to transform the parity-check equations into new forms to ensure the indices of parity-check bits are different with each other. The process is described in Algorithm 3.

In Algorithm 3, the  $i$ -th row of a  $K_P \times N$  matrix  $\mathbf{D}$  is first initialized in terms of the parity-check equation obtained by  $\mathcal{R}_i(\mathbf{v})$ . Then, GE is used on the rows  $\mathbf{D}$  to obtain a row echelon form matrix  $\mathbf{E}$ . Finally, the newly parity-check set  $\mathcal{Q}_i(\mathbf{v})$  is obtained by the  $i$ -th row of  $\mathbf{E}$ ,  $i = 1, 2, \dots, K_P$  and

**Algorithm 4:** The PC-SCEM

---

**Input:**  $N$ ,  $\mathcal{B}$  and  $\{\mathcal{Q}_i(\mathbf{v})\}$ ;  
**Output:**  $\mathcal{T}$ ,  $d_{\min}$  and  $A_{d_{\min}}$ ;

- 1 Initialize  $\mathcal{T} \leftarrow \emptyset$ ,  $A_{d_{\min}} \leftarrow 0$  and  $r \leftarrow \min_{i \in \mathcal{B}}(wt(\mathbf{G}_i))$ ;
- 2 Initialize  $k \leftarrow N$ ,  $\mathbf{v} \leftarrow \mathbf{0}$  and  $\mathbf{c} \leftarrow \mathbf{0}$ ;
- 3 Initialize  $\mathcal{P} = \{k_i | k_i = \min(\mathcal{Q}_i(\mathbf{v})), i = 1, 2, \dots, K_P\}$ ;
- 4 **while**  $A_{d_{\min}} = 0$  **do**
- 5     **while**  $k \leq N$  **do**
- 6         **if**  $k \in \mathcal{B} - \mathcal{P}$  **then**     // information bit
- 7              $v_k \leftarrow \arg \min_{v_k \in \{0,1\}}(c_k)$  and  $c_k \leftarrow 0$ ;
- 8         **else if**  $k \in \mathcal{P}$  **then**     // parity-check bit
- 9             Find  $i$  making  $k_i = k$ ;
- 10              $v_k \leftarrow \bigoplus_{t \in (\mathcal{Q}_i(\mathbf{v}) \setminus k_i)} v_t$ ;
- 11              $c_k \leftarrow \left( \bigoplus_{j=k}^N (v_j g_{j,k}) \right)$ ;
- 12         **else**     // frozen bit
- 13              $v_k \leftarrow 0$  and  $c_k \leftarrow \left( \bigoplus_{j=k}^N (v_j g_{j,k}) \right)$ ;
- 14         **if**  $d(\mathbf{v}_k^N) \leq r$  **then**     // satisfy the sphere constraint
- 15             **if**  $k > 1$  **then**
- 16                  $k \leftarrow k - 1$ ;
- 17             **else**
- 18                  $\mathcal{T} \leftarrow \mathcal{T} \cup \{\mathbf{c}\}$  and  $A_{d_{\min}} \leftarrow A_{d_{\min}} + 1$ ;
- 19                 Go to Step 21;
- 20         **else**     // prune the search tree
- 21             **while**  $k \leq N$  **do**
- 22                 **if**  $k \in \mathcal{B} - \mathcal{P}$  and  $c_k = 0$  **then**
- 23                      $v_k \leftarrow v_k \oplus 1$  and  $c_k \leftarrow 1$ ;
- 24                     Go to Step 14;
- 25                 **else**
- 26                      $k \leftarrow k + 1$ ;
- 27          $\mathcal{T} \leftarrow \mathcal{T} - \{\mathbf{0}\}$  and  $A_{d_{\min}} \leftarrow A_{d_{\min}} - 1$ ;
- 28         **if**  $A_{d_{\min}} = 0$  **then**
- 29              $r \leftarrow r + 2$ ;
- 30         **else**
- 31              $d_{\min} \leftarrow r$ ;

---

the parity-check bits of these sets are different due to the row echelon form. Thus, the parity-check bit indices are

$$k_i = \min(\mathcal{Q}_i(\mathbf{v})), \quad i = 1, 2, \dots, K_P. \quad (36)$$

and these bits can be decided by

$$v_{k_i} = \bigoplus_{k \in (\mathcal{Q}_i(\mathbf{v}) \setminus k_i)} v_k, \quad i = 1, 2, \dots, K_P. \quad (37)$$

### B. Detailed Description of PC-SCEM

Since all the codewords of concatenated polar codes with the Hamming weight  $d_{\min}$  in the codeword space is on the

surface of a sphere, a sphere constraint can also be used to enumerate these codewords with  $d_{\min}$ . Based on this, an PC-SCEM is proposed to analyze the MWD of concatenated polar codes.

Algorithm 4 describes the entire procedure of the PC-SCEM. In the method, since  $d_{\min}$  determines the sphere constraint, deciding  $d_{\min}$  is the first thing to analyze the MWD. However, there are no simple methods to calculate the  $d_{\min}$  of concatenated polar codes. Therefore, a greedy method is used to determine  $d_{\min}$ . We first set the radius of the sphere constraint with a lower bound of  $d_{\min}$ . Since concatenated polar codes are the subcode of the corresponding polar codes, the minimum Hamming weight of polar codes is the lower bound of the minimum Hamming weight of concatenated polar code. Thus, the radius is first set as

$$r = \min_{i \in \mathcal{B}}(wt(\mathbf{G}_i)). \quad (38)$$

Then, considering that polar code is the subcode of RM code and the Hamming weight of the codewords of RM code is even [16, Sec. 4.3.], the Hamming weight of the codewords of concatenated polar codes is even as well. Thus, if no codewords can be enumerated in the sphere constraint (38),  $r$  is added 2 until finding codewords in the sphere constraint and  $r$  is the  $d_{\min}$  of the concatenated polar code.

To enumerate the codewords in the sphere constraint, all the bits are divided into three types: information bits, frozen bits and parity-check bits. For the information bits and frozen bits, the search process is same as that in the SCCEM. For the parity-check bits, they are directly calculated by the previous searched bits according to the corresponding  $\mathcal{Q}_i(\mathbf{v})$  such that the codewords searched by the PC-SCEM belong to the MWD of the concatenated polar code.

## VI. MWD AND COMPLEXITY EVALUATION

In this section, we first provide the MWD of polar codes. Then, the optimal CRC polynomial of the CRC-polar concatenated codes and the corresponding MWD are provided. Finally, the complexity comparison between the three proposed methods and the SCL based methods is provided. The improved GA [20] and the polarization weight (PW) [21] are applied to construct polar codes.

### A. MWD of Polar Codes

In this subsection, the MWD of polar codes with different code rates is first provided. Then, we provide the MWD of polar codes with different SNR. Finally, the BLER performance of polar codes decoded by SCL with list size 32 and the corresponding AUB are provided.

Table I provides the MWD of polar codes constructed by GA and PW with different code lengths and code rates. Since the MWD of the polar codes constructed by GA changes along with SNR, the polar codes are constructed at  $E_b/N_0 = 3\text{dB}$ . The polar codes marked with “\*” are constructed at  $E_b/N_0 = 2.5\text{dB}$ , since  $A_{d_{\min}}$  of these polar codes at  $E_b/N_0 = 3\text{dB}$  is so large that the codewords are difficult to be enumerated. In Table I, we can observe that the MWD of polar codes constructed by GA and PW with  $N = 256$  is almost the same.

TABLE I  
THE MWD OF POLAR CODES CONSTRUCTED BY GA AND PW WITH DIFFERENT CODE LENGTHS AND CODE RATES.

			N													
			256		512		1024		2048		4096		8192		16384	
			$d_{\min}$	$A_{d_{\min}}$												
R	1/9	GA	32	88	64	4376	64	2608	64	224	128	394848	128	47296	128*	384*
		PW	32	88	32	16	64	3120	64	1632	64	704	64	384	64	256
	1/8	GA	32	152	32	16	64	8752	64	1376	128	1036896	128	292032	128	128
		PW	32	152	32	48	64	6960	64	5216	64	2752	64	1408	64	768
	1/7	GA	32	344	32	48	64	19760	64	8288	128	3039840	128	1850560	128	11648
		PW	32	280	32	112	32	32	64	14944	64	14528	64	5504	64	4864
	1/6	GA	32	920	32	432	64	65328	64	39008	64	1216	128	10958016	128	786816
		PW	32	920	32	432	32	96	32	64	64	54464	64	57728	64	45824
	1/5	GA	32	2840	32	1840	32	224	64	255584	64	47296	64*	2432*	128	29096320
		PW	32	2840	32	2096	32	1376	32	448	32	384	32	256	64	381696
	1/4	GA	16	48	32	12592	32	4704	32	64	64	1408192	64	55680	64*	256*
		PW	16	48	16	32	32	9312	32	7360	32	3456	32	2816	32	1536
	1/3	GA	16	944	16	96	32	161376	32	47296	32	128	64	30026112	64	606976
		PW	16	1072	16	608	16	192	16	128	32	158080	32	189184	32	181760
	1/2	GA	8	32	16	52832	16	20672	16	896	32	15280512	32	3298048	32	1536
		PW	8	96	8	64	16	54464	16	57728	16	45824	16	22016	16	19456
	2/3	GA	8	11360	8	5824	8	896	16	3520896	16	2061056	16	230912	16	1024
		PW	8	11360	8	11456	8	5504	8	2816	8	3584	8	3072	8	2048
	3/4	GA	4	64	8	65728	8	57728	8	23296	8	3584	16	63694336	16	24431616
		PW	4	64	8	65728	8	78208	8	90880	8	50688	8	44032	8	38912
4/5	GA	4	448	4	128	8	344448	8	262912	8	108032	8	44032	8	6144	
	PW	4	448	4	384	4	256	8	508672	8	706048	8	658432	8	366592	
5/6	GA	4	1216	4	384	4	256	8	1065728	8	1017344	8	461824	8	186368	
	PW	4	1216	4	896	4	768	4	512	4	1024	8	2952192	8	9246720	
6/7	GA	4	2752	4	1408	4	768	4	512	8	3442176	8	4197376	8	2037760	
	PW	4	2752	4	2432	4	1792	4	1536	4	1024	4	2048	8	13899776	
7/8	GA	4	6848	4	5504	4	2816	4	1536	4	1024	8	11340800	8	15210496	
	PW	4	6848	4	5504	4	2816	4	1536	4	3072	4	2048	4	4096	
8/9	GA	4	12992	4	7552	4	4864	4	3584	4	3072	4	2048	8	36313088	
	PW	4	12992	4	9600	4	4864	4	5632	4	3072	4	6144	4	4096	

Polar codes marked with "\*" are constructed at  $E_b/N_0 = 2.5\text{dB}$ .

TABLE II  
THE MWD OF POLAR CODES CONSTRUCTED BY GA IN DIFFERENT SNR WITH CODE RATE 1/2 AND DIFFERENT CODE LENGTHS.

$\frac{E_b}{N_0}$ (dB)	(256, 128)		(512, 256)		(1024, 512)		(2048, 1024)	
	$d_{\min}$	$A_{d_{\min}}$	$d_{\min}$	$A_{d_{\min}}$	$d_{\min}$	$A_{d_{\min}}$	$d_{\min}$	$A_{d_{\min}}$
0.0	8	224	8	64	16	66752	16	86400
0.5	8	224	8	64	16	66752	16	61824
1.0	8	224	8	64	16	54464	16	57728
1.5	8	96	8	64	16	54464	16	33152
2.0	8	96	16	61024	16	45248	16	27008
2.5	8	96	16	58976	16	35008	16	5504
3.0	8	32	16	52832	16	20672	16	896
3.5	8	32	16	44640	16	12992	32	17822912
4.0	16	60720	16	39520	16	5824	32	13382848
4.5	16	60720	16	30816	16	704	32	10843328

Then, the difference of the MWD between GA and PW occurs and becomes larger as the code length increases. Specifically, the MWD of polar codes constructed by GA has larger  $d_{\min}$  or less  $A_{d_{\min}}$ . Based on this, we can explain why GA and PW have almost the same performance for short polar codes, but the performance of GA is better for long polar codes generally.

Table II provides the MWD of polar codes with code rate 1/2 and different code lengths at various SNR. Polar codes are constructed by GA. In Table II, we can observe that the

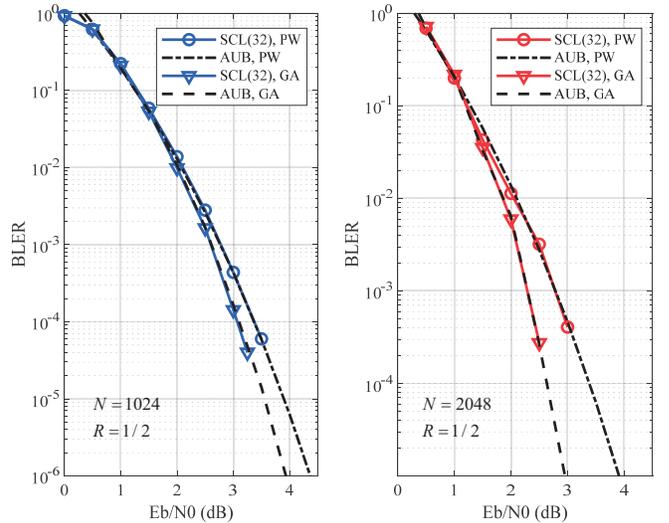


Fig. 3. The BLER performance of polar codes decoded by SCL with list size 32.

MWD of polar codes constructed by GA is variable with the change of SNR. The reason is that the information set obtained in terms of GA is distinct in different SNR regions. In addition, the MWD with fixed code length and code rate is

also improved as the SNR increases, i.e., larger  $d_{\min}$  or less  $A_{d_{\min}}$ . Thus, no error floor occurs in the performance curve of GA due to the improved MWD. Moreover, compared with the MWD of polar codes constructed by PW (shown in Table I), the polar codes constructed by GA has better AUB in the high SNR region, which leads the performance is better in Fig. 3.

Fig. 3 shows the BLER performance of polar codes with code rate 1/2 and code length 1024 and 2048. According to Table I and Table II, the AUB calculated by (7) is provided. In Fig. 3, we can observe that the BLER performance coincides with the corresponding AUB. Thus, the AUB calculated by the MWD can be used to evaluate the BLER performance of polar codes. Then, in the low SNR region, the performance of polar codes constructed by GA is almost the same as that constructed by PW. As the SNR increases, the performance gap between them occurs and becomes larger. The reason is that the MWD of polar codes constructed by GA is improved with the increase of SNR and better than that constructed by PW. Therefore, from the viewpoint of MWD, GA is appropriate for constructing long polar codes rather than PW.

### B. MWD of CRC-Polar Concatenated Codes

In this subsection, we first provide the MWD of CRC-polar concatenated codes with optimal CRC polynomial. Then, the corresponding BLER performance are provided.

Table III provides the MWD of CRC-polar concatenated codes for different code lengths and code rates with optimal CRC and standard CRC. PW is used to construct CRC-polar concatenated codes. The standard CRC polynomials are provided in [22]. By exhausting all the CRC polynomial and analyzing the corresponding MWD of CRC-polar concatenated codes, the optimal CRC is obtained. The optimization principles are 1) maximizing  $d_{\min}$  and 2) minimizing  $A_{d_{\min}}$  when  $d_{\min}$  is identical.

Fig. 4 shows the BLER performance of CRC-polar concatenated codes with the optimal CRC and the standard CRC, where code length is 128 and CRC length is 6. As shown in Table III, the optimal CRC polynomials for code rates 1/4, 1/2 and 3/4 are 0x5B, 0x73 and 0x73, respectively, and the standard CRC polynomial is 0x59. In Fig. 4, the BLER performance is close to the AUB in the high SNR region. Then, since the CRC-polar concatenated codes with the optimal CRC has better MWD, the performance is better in the medium to high SNR regions.

Fig. 5 illustrates the BLER performance of CRC-polar concatenated codes with code length 512 and 11-bit CRC. As shown in Table III, the optimal CRC polynomials for code rates 1/4, 1/2 and 3/4 are 0x9A7, 0xC23 and 0xE81, respectively, and the standard CRC polynomial is 0xCBB. Similarly to Fig. 4, the BLER performance is also close to the AUB and the performance of the optimal CRC is better in the high SNR region.

### C. Complexity Evaluation

In this subsection, we provide the complexity comparison between the proposed methods and the SCL based methods.

TABLE III  
THE MWD OF CRC-POLAR CONCATENATED CODES CONSTRUCTED BY PW WITH DIFFERENT CODE LENGTHS AND CODE RATES.

N	$K_I$	$K_P$	Optimal CRC			Standard CRC		
			$g(x)$	$d_{\min}$	$A_{d_{\min}}$	$g(x)$	$d_{\min}$	$A_{d_{\min}}$
128	32	6	0x5B	24	270	0x59	16	12
		8	0x1E7	24	128	0x1D5	16	5
		11	0xD11	24	34	0xCBB	16	3
	64	6	0x73	12	300	0x59	8	56
		8	0x14D	12	99	0x1D5	8	14
		11	0xD63	12	15	0xCBB	12	147
	96	6	0x73	6	16	0x59	6	53
		8	0x18D	6	6	0x1D5	4	8
		11	0xEFC	8	2453	0xCBB	4	12
256	64	6	0x79	32	1640	0x59	16	8
		8	0x1F9	32	362	0x1D5	32	758
		11	0x895	32	41	0xCBB	32	136
	128	6	0x57	16	5853	0x59	12	23
		8	0x1D7	16	1397	0x1D5	12	16
		11	0xC31	16	200	0xCBB	16	553
	192	6	0x57	8	4647	0x59	8	9494
		8	0x14D	8	1621	0x1D5	8	3521
		11	0xCB9	8	155	0xCBB	8	606
512	128	6	0x43	32	498	0x59	32	1036
		8	0x1F3	32	95	0x1D5	32	256
		11	0x9A7	32	3	0xCBB	32	32
	256	6	0x57	16	1912	0x59	16	4344
		8	0x14D	16	362	0x1D5	16	918
		11	0xC23	16	28	0xCBB	16	213
	384	6	0x43	8	2563	0x59	8	5220
		8	0x187	8	368	0x1D5	8	1193
		11	0xE81	8	6	0xCBB	8	708

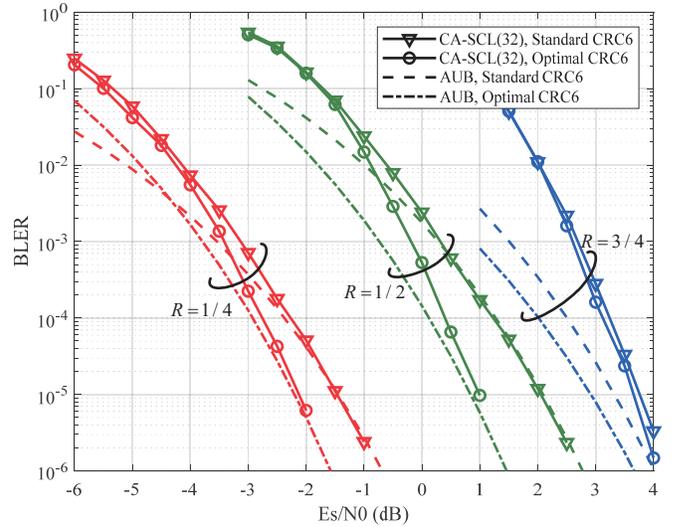


Fig. 4. The BLER performance of CRC-polar concatenated codes with code length 128 and CRC length 6.

Fig. 6 illustrates the complexities of the SCEM, the SCREM and the SCL based methods with code length 128 and different code rates. Considering the MWD of polar codes constructed by GA changes along with SNR, PW is used in Fig. 6. The complexity of SCEM and SCREM is counted by the average visited nodes (AVN). The AVN of enumerating all the codewords is  $2^K N \log N$ , which is the upper bound of the complexity of enumerating MWD. The AVN of SCL method [10] with list size  $L_1$  are  $\min(2^K N \log N, L_1 N \log N)$ .

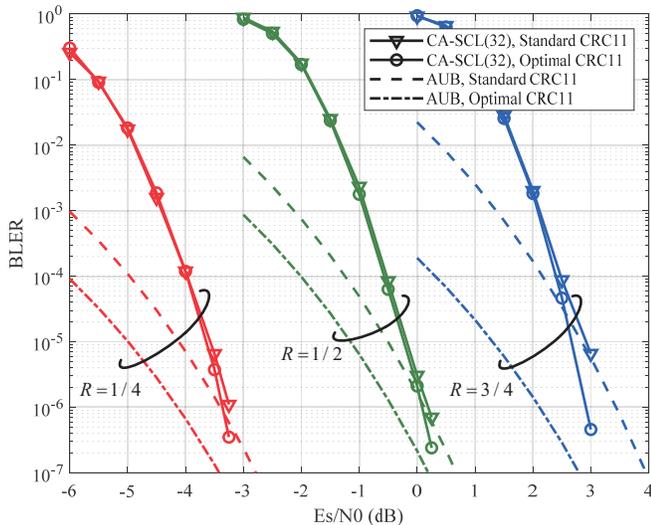


Fig. 5. The BLER performance of CRC-polar concatenated codes with code length 512 and CRC length 11.

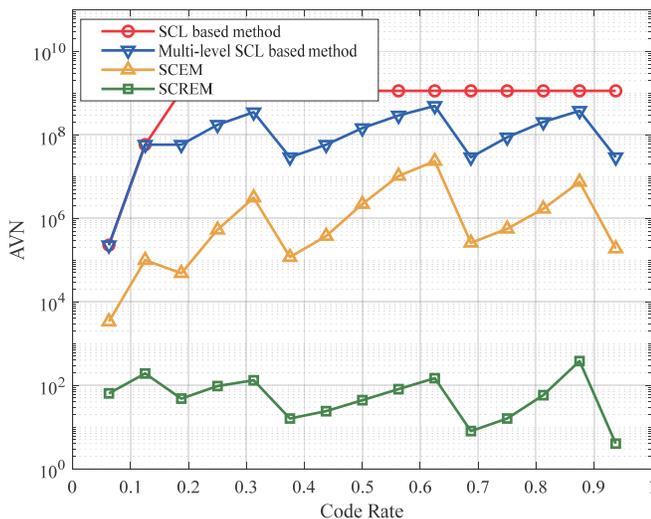


Fig. 6. The complexity of the SCЕМ, the SCREM and the SCL based methods with  $N = 128$ .

The AVN of multi-level SCL method [11] with list size  $L_2$  and level number  $M$  are  $\min(2^K N \log N, ML_2 N \log N)$ .  $L_1$  and  $L_2$  used in [10] and [11] are 1280000 and 32768, respectively, and  $M$  is the number of row with the row weight  $d_{\min}$  in the generator matrix of polar codes.

In Fig. 6, the complexity of SCЕМ is lower than those of the SCL based methods, since the sphere constraint can prune the search tree to reduce the redundant search. Specifically, the AVN of the proposed SCЕМ achieves 3 to 4 magnitude reduction compared with both the SCL method and the multi-level SCL method. Furthermore, due to the recursive structure of SCREM, its complexity is lower than the SCЕМ. In comparison to the SCЕМ, the SCL method and the multi-level SCL method, the SCREM can achieve up to  $10^5$ ,  $10^8$ , and  $10^7$  times complexity reduction.

## VII. CONCLUSION

In this paper, we propose three methods to analyze the MWD of polar codes. The SCЕМ is first proposed to exactly enumerate all the codewords belonging to the MWD, which exploits the distance property that all the codewords with the identical Hamming weight are distributed on a spherical shell. Then, based on the SCЕМ and the Plotkin's construction, we propose SCREM to recursively analyze the MWD with lower complexity. Finally, the PC-SCЕМ is proposed by introducing the parity-check equations and the sphere constraint to analyze the MWD of concatenated polar codes. The experimental results illustrate that the complexities of the proposed SCЕМ and SCREM with code length 128 are lower than those of the SCL based methods.

## APPENDIX

According to (15), a codeword  $\mathbf{c}$  with minimum Hamming weight  $d_{\min}$  can be expressed as

$$wt(\mathbf{c}) = wt(\mathbf{c}' + \mathbf{c}'') + wt(\mathbf{c}'') = d_{\min}. \quad (39)$$

Then,  $d'_{\min}$  and  $d''_{\min}$  are divided into three cases by Lemma 2. In the proof of each case, classified discussion is used.

1) When  $d'_{\min} = d_{\min}$  and  $d''_{\min} = d_{\min}$ ,  $\mathcal{T}$  is obtained as follows.

a) Supposing  $wt(\mathbf{c}'') = 0$ ,  $wt(\mathbf{c})$  is simplified as

$$wt(\mathbf{c}) = wt(\mathbf{c}') = d_{\min}. \quad (40)$$

Thus,  $\forall \mathbf{c}' \in \mathcal{T}'$  makes  $wt(\mathbf{c})$  is  $d_{\min}$ .

b) Supposing  $wt(\mathbf{c}'') = d_{\min}$ , similarly,  $wt(\mathbf{c})$  is simplified as

$$wt(\mathbf{c}' + \mathbf{c}'') = 0. \quad (41)$$

Hence, we have  $\mathbf{c}'' = \mathbf{c}'$ . Furthermore, according to Lemma 1, we have  $\mathcal{T}' \subset \mathcal{T}''$ . Therefore, for  $\forall \mathbf{c}' \in \mathcal{T}'$ ,  $\exists \mathbf{c}'' \in \mathcal{T}''$  makes  $wt(\mathbf{c}' + \mathbf{c}'') = 0$ , i.e.,  $\mathbf{c}'' = \mathbf{c}'$ . Thus,  $(\mathbf{0}, \mathbf{c}')$ ,  $\mathbf{c}' \in \mathcal{T}'$ , is the codeword of  $\mathcal{C}$  and its Hamming weight is  $d_{\min}$ .

c) Supposing  $wt(\mathbf{c}'') > d_{\min}$ , it is clear that  $wt(\mathbf{c}) > d_{\min}$ .

In conclusion,  $\mathcal{T} = \mathcal{T}_1 \cup \mathcal{T}_2$ , where  $\mathcal{T}_1 = \{(\mathbf{c}', \mathbf{0}) | \mathbf{c}' \in \mathcal{T}'\}$  and  $\mathcal{T}_2 = \{(\mathbf{0}, \mathbf{c}') | \mathbf{c}' \in \mathcal{T}'\}$ .

2) When  $d'_{\min} = d_{\min}$  and  $d''_{\min} = \frac{d_{\min}}{2}$ ,  $\mathcal{T}$  is obtained as follows.

a) Supposing  $wt(\mathbf{c}'') = 0$ ,  $\forall \mathbf{c}' \in \mathcal{T}'$  makes  $wt(\mathbf{c})$  is  $d_{\min}$ .

b) Supposing  $wt(\mathbf{c}'') = \frac{d_{\min}}{2}$  and  $wt(\mathbf{c}') = 0$ , obviously,  $(\mathbf{c}'', \mathbf{c}'')$  is the codeword of  $\mathcal{C}$  and its Hamming weight is  $d_{\min}$ .

c) Supposing  $wt(\mathbf{c}'') = \frac{d_{\min}}{2}$  and  $wt(\mathbf{c}') = d_{\min}$ , to obtain the codeword  $\mathbf{c}$  with  $d_{\min}$ , all the  $\mathbf{c}' \in \mathcal{T}'$  and  $\mathbf{c}'' \in \mathcal{T}''$  are enumerated to satisfy

$$wt(\mathbf{c}' + \mathbf{c}'') = \frac{d_{\min}}{2}. \quad (42)$$

d) Supposing  $wt(\mathbf{c}'') = \frac{d_{\min}}{2}$  and  $wt(\mathbf{c}') > d_{\min}$ , it is clear that  $wt(\mathbf{c}' + \mathbf{c}'') > \frac{d_{\min}}{2}$ . Thus,  $wt(\mathbf{c}) > d_{\min}$ .

- e) Supposing  $\frac{d_{\min}}{2} < wt(\mathbf{c}'') < d_{\min}$ , in order to make  $wt(\mathbf{c})$  is  $d_{\min}$ , we have

$$0 < wt(\mathbf{c}' + \mathbf{c}'') < \frac{d_{\min}}{2}. \quad (43)$$

Then, according to Lemma 1,  $\mathbf{c}'$  is the codeword of  $\mathcal{C}''$ . Thus,  $\mathbf{c}' + \mathbf{c}''$  is also the codeword of  $\mathcal{C}''$ . However, since  $d_{\min}''$  is  $\frac{d_{\min}}{2}$ , no codeword of  $\mathcal{C}''$  can satisfy (43). Therefore, in this case, no codeword of  $\mathcal{C}$  with  $d_{\min}$  can be found.

- f) Supposing  $wt(\mathbf{c}'') = d_{\min}$ ,  $wt(\mathbf{c})$  is simplified as

$$wt(\mathbf{c}' + \mathbf{c}'') = 0. \quad (44)$$

According to the 1)-b) of the proof of Lemma 3,  $(\mathbf{0}, \mathbf{c}')$ ,  $\mathbf{c}' \in \mathcal{T}'$ , is the codeword of  $\mathcal{C}$  and its Hamming weight is  $d_{\min}$ .

- g) Supposing  $wt(\mathbf{c}'') > d_{\min}$ , obviously,  $wt(\mathbf{c}) > d_{\min}$ .

In conclusion,  $\mathcal{T} = \mathcal{T}_1 \cup \mathcal{T}_2 \cup \mathcal{T}_3 \cup \mathcal{T}_4$ , where  $\mathcal{T}_3 = \{(\mathbf{c}'', \mathbf{c}'') | \mathbf{c}'' \in \mathcal{T}''\}$  and  $\mathcal{T}_4 = \{(\mathbf{c}' \oplus \mathbf{c}'', \mathbf{c}'') | \mathbf{c}' \in \mathcal{T}', \mathbf{c}'' \in \mathcal{T}'', wt(\mathbf{c}' \oplus \mathbf{c}'') = \frac{d_{\min}}{2}\}$ .

- 3) When  $d_{\min}' > d_{\min}$  and  $d_{\min}'' = \frac{d_{\min}}{2}$ ,  $\mathcal{T}$  is obtained as follows.

- a) Supposing  $wt(\mathbf{c}'') = 0$ , it is clear that  $wt(\mathbf{c}) > d_{\min}$ .  
 b) Supposing  $wt(\mathbf{c}'') = \frac{d_{\min}}{2}$  and  $wt(\mathbf{c}') = 0$ , obviously,  $(\mathbf{c}'', \mathbf{c}'')$  is the codeword of  $\mathcal{C}$  and its Hamming weight is  $d_{\min}$ .  
 c) Supposing  $wt(\mathbf{c}'') = \frac{d_{\min}}{2}$  and  $wt(\mathbf{c}') > d_{\min}$ , we have

$$wt(\mathbf{c}' + \mathbf{c}'') > \frac{d_{\min}}{2}. \quad (45)$$

Thus,  $wt(\mathbf{c}) > d_{\min}$ .

- d) Supposing  $\frac{d_{\min}}{2} < wt(\mathbf{c}'') < d_{\min}$ , in order to make  $wt(\mathbf{c})$  is  $d_{\min}$ , we have

$$0 < wt(\mathbf{c}' + \mathbf{c}'') < \frac{d_{\min}}{2}. \quad (46)$$

Then, according to the 2)-e) of the proof of Lemma 3, no codeword of  $\mathcal{C}$  with  $d_{\min}$  can be found in this case.

- e) Supposing  $wt(\mathbf{c}'') = d_{\min}$ , we have  $wt(\mathbf{c}' + \mathbf{c}'') > 0$ . Thus,  $wt(\mathbf{c}) > d_{\min}$ .  
 f) Supposing  $wt(\mathbf{c}'') > d_{\min}$ , obviously,  $wt(\mathbf{c}) > d_{\min}$ .

In conclusion,  $\mathcal{T}_3 = \{(\mathbf{c}'', \mathbf{c}'') | \mathbf{c}'' \in \mathcal{T}''\}$ .

From the above, Lemma 3 has been proved.

## REFERENCES

- [1] E. Arkan, "Channel polarization: A method for constructing capacity achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3051–3073, Jul. 2009.
- [2] I. Tal and A. Vardy, "List decoding of polar codes," *IEEE Trans. Inf. Theory*, vol. 61, no. 5, pp. 2213–2226, May. 2015.
- [3] K. Chen, K. Niu, and J. R. Lin, "List successive cancellation decoding of polar codes," *Electron. Lett.*, vol. 48, no. 9, pp. 500–501, 2012.
- [4] K. Chen, K. Niu and J. Lin, "Improved Successive Cancellation Decoding of Polar Codes," *IEEE Trans. Commun.*, vol. 61, no. 8, pp. 3100–3107, Aug. 2013.
- [5] K. Niu and K. Chen, "CRC-aided decoding of polar codes," *IEEE Commun. Lett.*, vol. 16, no. 10, pp. 1668–1671, Oct. 2012.
- [6] 3<sup>rd</sup> Generation Partnership Project (3GPP), "Multiplexing and channel coding," *3GPP TS 38.212 V15.0.0*, 2017.
- [7] J. Dai, J. Gao and K. Niu, "Learning to Mitigate the FAR in Polar Code Blind Detection," *IEEE Wireless Commun. Lett.*, early access.
- [8] B. Li, H. Shen and D. Tse, "An Adaptive Successive Cancellation List Decoder for Polar Codes with Cyclic Redundancy Check," *IEEE Commun. Lett.*, vol. 16, no. 12, pp. 2044–2047, December 2012.
- [9] S. Lin and D. J. Costello, "Error Control Coding (2nd ed.)," PrenticeHall, Inc., 2004.
- [10] Z. Liu, K. Chen, K. Niu and Z. He, "Distance spectrum analysis of polar codes," in *Proc. 2014 IEEE WCNC*, Istanbul, 2014, pp. 490–495.
- [11] Q. Zhang, A. Liu, X. Pan and K. Pan, "CRC Code Design for List Decoding of Polar Codes," *IEEE Commun. Lett.*, vol. 21, no. 6, pp. 1229–1232, June 2017.
- [12] E. Arkan, "Serially Concatenated Polar Codes," *IEEE Access*, vol. 6, pp. 64549–64555, 2018.
- [13] G. Ricciutelli, T. Jerkovits, M. Baldi, F. Chiaraluce and G. Liva, "Analysis of the Block Error Probability of Concatenated Polar Code Ensembles," *IEEE Trans. Commun.*, vol. 67, no. 9, pp. 5953–5962, Sep. 2019.
- [14] S. Kahraman and M. E. Celebi, "Code based efficient maximum-likelihood decoding of short polar codes," in *Proc. 2012 IEEE Int. Symp. Inf. Theory*, Cambridge, MA, pp. 1967–1971, 2012.
- [15] K. Niu, K. Chen and J. Lin, "Low-Complexity Sphere Decoding of Polar Codes Based on Optimum Path Metric," *IEEE Commun. Lett.*, vol. 18, no. 2, pp. 332–335, Feb. 2014.
- [16] J. Guo and A. Guillen i Fbregas, "Efficient sphere decoding of polar codes," in *Proc. 2015 IEEE Int. Symp. Inf. Theory*, Hong Kong, pp. 236–240, 2015.
- [17] J. Piao, J. Dai and K. Niu, "CRC-Aided Sphere Decoding for Short Polar Codes," *IEEE Commun. Lett.*, vol. 23, no. 2, pp. 210–213, Feb. 2019.
- [18] S. Ejaz, F. Yang, T. H. Soliman, "Network polar coded cooperation with joint sc decoding", *Electron. Lett.*, vol. 51, no. 9, pp. 695–697, 2015.
- [19] C. Schrch, "A partial order for the synthesized channels of a polar code," in *Proc. 2016 IEEE Int. Symp. Inf. Theory*, Barcelona, pp. 220–224, 2016.
- [20] J. Dai, K. Niu, Z. Si, C. Dong and J. Lin, "Does Gaussian Approximation Work Well for the Long-Length Polar Code Construction?," *IEEE Access*, vol. 5, pp. 7950–7963, 2017.
- [21] G. He et al., "Beta-Expansion: A Theoretical Framework for Fast and Recursive Construction of Polar Codes," in *Proc. 2017 IEEE Global Communications Conference*, Singapore, pp. 1–6, 2017.
- [22] P. Koopman and T. Chakravarty, "Cyclic redundancy code (CRC) polynomial selection for embedded networks," in *International Conference on Dependable Systems and Networks*, Florence, Italy, 2004, pp. 145–154, 2004.