TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 05/15 pp269–282
DOI: 10.26599/TST.2021.9010086
Volume 28, Number 2, April 2023

Two-Stage Constructions for the Rate-Compatible Shortened Polar Codes

Chunjie Li, Haiqiang Chen*, Zelin Wang, Youming Sun, Xiangcheng Li, and Tuanfa Qin

Abstract: In this paper, we propose the two-stage constructions for the rate-compatible shortened polar (RCSP) codes. For the Stage-I construction, the shortening pattern and the frozen bit are jointly designed to make the shortened bits be completely known by the decoder. Besides, a distance-greedy algorithm is presented to improve the minimum Hamming distance of the codes. To design the remaining Stage-II frozen bits, three different construction algorithms are further presented, called the Reed-Muller (RM) construction, the Gaussian Approximation (GA) construction, and the RM-GA construction. Then we give the row weight distribution numerical results of the generator matrix after the Stage-I and Stage-II constructions, which shows that the proposed constructions can efficiently increase the minimum Hamming distance. Simulation results show that the proposed RCSP codes have excellent frame error rate (FER) performances at different code lengths and code rates. More specifically, the RM-GA construction performs best and can achieve at most 0.8 dB gain compared to the Wang14 and the quasi-uniform puncturing (QUP) schemes. The RM construction is designed completely by the distance-constraint without channel evaluation thus has the simplest structure. Interestingly, it still has better FER performance than the existing shortening/puncturing schemes, especially at high signal noise ratio (SNR) region.

Key words: polar codes; rate-compatibility; Reed-Muller codes; Hamming distance; shortening

1 Introduction

Polar codes are the first family of codes which have been proven to achieve the capacity of any symmetric binary-input discrete memoryless channel (B-DMC)[1]. However, the error performance of polar codes with short and medium code length under successive cancellation (SC) decoding algorithm do not perform well. An improvement to the SC decoder, called successive cancellation list (SCL) decoder was introduced in Ref. [2], which could achieve the error performance of maximum likelihood decoder. Polar codes have good structural characteristics and low encoding and decoding complexity. At the end of 2016, polar codes were selected as the candidate coding scheme for the fifth-generation (5G) mobile communications, and were finally adopted as the coding standard for the uplink/downlink channel control[3]. Polar codes can also find applications for the secure transmission over parallel relay channels[4] and the 5G-based maritime communications[5].

However, the length of polar codes are limited to powers of 2 due to the original Kronecker power construction, which restricts their flexible applications in practice. Polar codes with arbitrary lengths and rates can be mainly obtained by puncturing, shortening and repetition, resulting the rate-compatible polar codes. These rate-compatible schemes are also recommended

• Chunjie Li, Haiqiang Chen, Zelin Wang, Youming Sun, Xiangcheng Li, and Tuanfa Qin are with the School of Computer, Electronics and Information and Guangxi Colleges and Universities Key Laboratory of Multimedia Communications and Information Processing, Guangxi University, Nanning 530004, China. E-mail: 13077775683@163.com; haiqiang@gxu.edu.cn; 674253256@qq.com; ymsun@gxu.edu.cn; xcli@gxu.edu.cn; tfqin@gxu.edu.cn.

• Haiqiang Chen is also with the Key Laboratory of Disaster Prevention and Structural Safety of Ministry of Education, Guangxi University, Nanning 530004, China.

∗ To whom correspondence should be addressed.

Manuscript received: 2021-09-01; revised: 2021-11-04; accepted: 2021-11-05

in 5G new radio[6]. Besides, the multi-kernel techniques can also construct polar codes with any length[7].

Punctured polar codes are first proposed in Ref. [8], where random puncturing and stopping-tree puncturing were both analyzed and compared. Niu et al.[9] proposed an efficient puncturing scheme, in which the puncturing positions are designed to be quasi-uniform distribution after bit-reversal permutation, thus called the quasi-uniform puncturing (QUP). The QUP has better row weight property than random puncturing and can achieve excellent decoding performances, especially at low code rates. The traditional shortening scheme was discussed in Ref. [10], where a simple shortening method was given, called the Wang14 scheme in this paper. The Wang14 scheme suggests that the last N_p coded bits, whose values are completely determined by the frozen bits, are shortened to form the rate-compatible shortened polar (RCSP) codes. Another similar shortening scheme is proposed in Ref. [11], which performed bit-reversal permutation on the encoded bit and then shortened the last N_p coded bits.

It is shown that the minimum Hamming distance has a significant impact on error performance of the polar codes[12]. Thus, constructing polar codes with large minimum Hamming distance to improve decoding performance becomes possible. Similar work can be seen in Ref. [13], where the minimum Hamming distance is increased by joint optimization of the shortening pattern and the set of frozen symbols. Li et al.[14] proposed the Reed-Muller (RM)-Polar codes, which have better distance property than the conventional polar codes and thus show better performance. Another way to increase the minimum Hamming distance for polar codes is cascading with other codes[15]. Examples can be found in Refs. [16, 17], where the polar codes are concatenated with cyclic redundancy check (CRC) codes. In Ref. [18], Arikan gave a theoretical analysis of such schemes. Moreover, concatenating polar codes as inner codes and parity-check codes as outer codes[19] can also significantly increase the minimum Hamming distance. The concatenated polar codes can outperform other competing codes such as low density parity check codes[20–22] and turbo codes[23].

However, neither the QUP nor the Wang14 shortening scheme considers the distance property in their constructions and this may cause potential performance degradations. Actually, polar codes can be seen as a generalization of RM codes[24, 25] and they share a common generator matrix[26]. This enable us to jointly optimize the distance property of the RCSP codes under the row weight constraint of RM codes.

In this study, we investigated the construction of the RCSP codes with flexible code length and code rate under different constraints. The constructed RCSP codes are able to employ the same polar encoding/decoding structure without any other change. First, we presented a general construction algorithm which is jointly designed by the shortening pattern and the frozen bit positions. Similar to the Wang14 scheme, the shortened bits are designed to be uniquely correlated with the frozen bits such that they can be completely known by the decoder. Such frozen bits form the Stage-I frozen set in this paper, whose cardinality is required to be minimum under the Weight-1 constraint. Second, we presented an algorithm to construct the the Stage-I frozen set with the RM code-aided. The presented algorithm is designed to be distance-greedy, aiming to construct codes that have large minimum Hamming distance. Note that, part of the work for Stage-I construction is discussed in BROADNETS 2021. It is shown that the total frozen bits of the RCSP code are generally greater than the Stage-I frozen bits, which indicates that there still have another part of frozen bits needed to be designed. Such part of frozen bits form the Stage-II frozen set. Then three different construction algorithms are presented to determine the Stage-II frozen bits: (1) The RM construction, which is completely designed by the distance constraint and has the simplest structure; (2) the RM-Gaussian Approximation (GA) construction, which considers both distance and channel reliability and has the best performance; (3) the GA construction, which is a simplified version of the RM-GA construction. The row weight property of the generator matrix after the Stage-I and Stage-II constructions is further analyzed. Numerical results show that the proposed construction schemes can efficiently increase the minimum Hamming distance of the corresponding codes. Finally, we gave the performances of the RCSP codes under the three constructions above with various lengths and rates. Simulation results show that, the RM-GA construction performs best, e.g., it can achieve about 0.8 dB gain compared with the conventional Wang14 and QUP schemes, for the code with $N = 256$ and rate $= 0.25$, at the frame error rate (FER) of 10^{-3}. Moreover, the RM construction has the simplest implementation without the complicated GA channel evaluation. Interestingly, it still has a close performance to the RM-GA construction and can outperform the conventional schemes, e.g., it achieves about 0.30 dB and 0.35 dB gains compared with the Wang14 and the QUP schemes, respectively, for the code with $N = 512$ and rate $= 0.75$, at the FER of 10^{-3}. The GA construction has a comparable performance with the conventional puncturing/shortening schemes.

The rest of the paper is organized as follows. In Section 2, we provide a short background on RM codes and polar codes, and introduce the system model of the proposed RCSP code. The proposed two-stage construction algorithm is demonstrated in details in Section 3, and an outline is given at the end. Section 4 gives the row weight property analysis of generator matrix and computational complexity analysis, then the simulation results compared with the conventional Wang14 and QUP schemes are presented. Section 5 concludes the paper.

2 Background

2.1 RM codes

This subsection uses the Kronecker construction method to describe RM codes. Since RM code is a linear block code, which can be constructed by a generator matrix. Let RM(n, n) denote the n-th order RM code, and let $\boldsymbol{G}_N$ be the N-dimension generator matrix with $N = 2^n$, which can be defined as

$$\boldsymbol{G}_N = \boldsymbol{F}^{\otimes n} \tag{1}$$

where $\boldsymbol{F} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$, $\boldsymbol{F}^{\otimes n}$ is the n-th Kronecker power of $\boldsymbol{F}$. The r-th order RM code RM(n, r) can then be defined as the linear code with a submatrix of $\boldsymbol{G}_N$, which is obtained by selecting rows of $\boldsymbol{G}_N$ with Hamming weights $\geqslant 2^{n-r}$.

The row weight of the generator matrix $\boldsymbol{G}_N$ has the following constraint with the row index. Let i denote an integer, $i \in \{0, 1, \ldots, N-1\}$, and $\boldsymbol{\pi}(i) = [b_{n-1}\ b_{n-2}\ \cdots\ b_1 b_0]$ is the binary representation of i over n bits. Let $w_t(i)$ represent the Hamming weight of $\boldsymbol{\pi}(i)$. The Hamming weight of the i-th row can be calculated by $w_r(i) = 2^{w_t(i)}$.

Since the RM code is a linear code, each row of the generator matrix can be regarded as a legal codeword. Therefore, the minimum row weight of the generator matrix corresponds to the minimum Hamming distance of the RM code. Actually, an RM code is equivalent to a special polar code which has the maximum row weight constraint. For example, an r-th order RM code RM(n, r) is equivalently a polar code with the frozen set $\mathcal{A}^c$ that satisfies the distance constraint $\mathcal{A}^c = \{i \,|\, w_r(i) < 2^{n-r}\}$. With this constraint, the minimum Hamming distance of the polar code is $d_{\min} = \min\{w_r(i) | i \in \mathcal{A}\}$, where $\mathcal{A}$ is the complementary set of $\mathcal{A}^c$, called the information set.

2.2 Polar codes

Given a B-DMC $W : \mathcal{X} \to \mathcal{Y}$, where $\mathcal{X} \in \{0, 1\}$ and $\mathcal{Y}$ denote the input and output alphabet, respectively. The channel transition probabilities can be defined as $W(y|x)$, $y \in \mathcal{Y}$, $x \in \mathcal{X}$. Let a_0^{N-1} denote a row vector $(a_0, a_1, \ldots, a_{N-1})$, and $a_i^j = (a_i, a_{i+1}, \ldots, a_j)$ denote a subvector, $0 \leqslant i \leqslant j \leqslant N-1$. After channel combining and splitting operation on N independent uses of W, we get N successive uses of synthesized binary input channels $W_N^{(i)}$, $i \in \{0, 1, \ldots, N-1\}$, which can be defined by the transition probabilities as follows:

$$W_N^{(i)}\left(y_0^{N-1}, u_0^{i-1} | u_i\right) = \sum_{u_{i+1}^{N-1} \in \mathcal{X}^{N-i-1}} \frac{1}{2^{N-1}} W_N\left(y_0^{N-1} | u_0^{N-1}\right) \tag{2}$$

The N independent subchannels can be divided into two parts. One part of channels with capacity tends to be 1, called “noiseless channel”, and the other part of channels with capacity tends to be 0, called “full noise channel” . The reliability of each subchannel can be computed by using the Bhattacharyya parameter[1], density evolution[27], GA[28] or polarization weight[29]. The K most reliable subchannels with indices in $\mathcal{A}$ carry information bits and the rest subchannels in $\mathcal{A}^c$ are set to be fixed values, such as all zeros. For an (N, K) polar code with K message bits and N coded bits, the encoding process can be defined as

$$\boldsymbol{c}_0^{N-1} = \boldsymbol{u}_0^{N-1} \boldsymbol{G}_N \tag{3}$$

where $\boldsymbol{u}_0^{N-1} = (u_0, u_1, \ldots, u_{N-1})$ is the source information vector and $\boldsymbol{c}_0^{N-1} = (c_0, c_1, \ldots, c_{N-1})$ is the polar codeword. As mentioned above, polar codes can be seen as a generalization of RM codes and both of them are defined by the same generator matrix $\boldsymbol{G}_N$. However, they select the information bits according to different constraints. In particular, the Hamming distance is considered in the RM codes construction, which can be exploited to optimize the proposed RCSP codes in this paper.

2.3 System model

The system model of the proposed RCSP codes construction is depicted in Fig. 1. In the transmitter, a K-bit information block is input into the polar encoder. After polar encoding, we get the N-bit polar codeword.

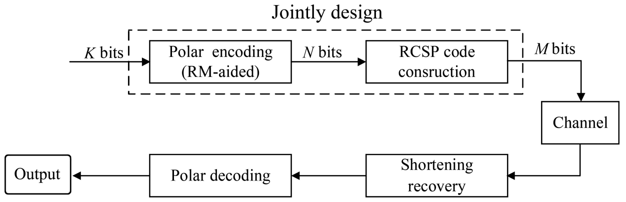


Fig. 1 System model.

To match arbitrary code length, the output polar code needs to be adjusted by shortening some bits from the N-bit encoded block, resulting in the M-bit RCSP codes. Then the RCSP codes with length-M are fed into the channel. In the receiver, we performed the opposite operation to get the corresponding estimated bits. Note that, the proposed RCSP codes are jointly designed with the encoding unit. The row weight constraint of RM codes are employed to maximize the Hamming distance in the code construction. This is quite different from the traditional shortening scheme.

3 Proposed Two-Stage Constructions for Rate Compatible Shortened Polar Codes

3.1 Recursive construction algorithms for the Stage-I frozen bits and shortened bits

Let $g_{i,j}$ denote the element in the i-th row and j-th column of $\boldsymbol{G}_N$. Then we define the index set for each column as follows

$$\mathcal{M}_j = \{i \,|\, 0 \leqslant i \leqslant N-1, g_{i,j} \neq 0\},$$

where $0 \leqslant j \leqslant N-1$. Obviously, $\mathcal{M}_j$ is the non-zero positions for the j-th column. Let $\boldsymbol{p} = (p_0, p_1, \ldots, p_{N-1})$ be the shortening pattern with $p_i \in \{0,1\}$, where the ones imply the shortened positions. For a binary vector $\boldsymbol{a}$, let $Q(\boldsymbol{a})$ denote the index set of the non-zero positions in $\boldsymbol{a}$. Then the shortened positions can be represented as $Q(\boldsymbol{p})$. Let c_j be a code bit of c_0^{N-1}, which can be defined by

$$c_j = \sum_{i \in \mathcal{M}_j} \oplus u_i \tag{4}$$

Assume that c_j is selected as a shortened bit, then all the elements $i \in \mathcal{M}_j$ are designated to be the frozen-bit positions. This is the key step to ensure that c_j is completely determined by the frozen bits, thus is known by the decoder.

Note that these frozen bits are only determined by the shortening pattern, but not the subchannel reliabilities, which is quite different from the conventional polar construction. Such frozen bits can be defined as the Stage-I frozen bits in this paper. Accordingly, we define the Stage-I frozen set $\mathcal{A}_S^c$ as follows:

$$\mathcal{A}_S^c = \bigcup_{j \in Q(\boldsymbol{p})} \mathcal{M}_j \tag{5}$$

In order to minimize the number of Stage-I frozen bits, i.e., the cardinality of $\mathcal{A}_S^c$, the Weight-1 first criterion is introduced in Ref. [10], where the shortened positions are always selected from the index of columns with Weight-1. With this constraint, the number of shortened bits is exactly the number of Stage-I frozen bits, i.e., $N_p = |\mathcal{A}_S^c|$. The Stage-I frozen set $\mathcal{A}_S^c$ with minimum cardinality can be determined by the following recursive construction algorithm with N_p steps. Let $\mathcal{A}_S^{c(k)}$ be the temporary set at the k-th Step, with $\mathcal{A}_S^{c(0)} = \varnothing$. Let $Q(\boldsymbol{p}^{(k)})$ be the corresponding shortening set at the k-th step, with $Q(\boldsymbol{p}^{(0)}) = \varnothing$. Let $f^{(k)}$ be the selected Stage-I frozen-bit position at the k-th Step, then the sets $\mathcal{A}_S^{c(k)}$ and $Q(\boldsymbol{p}^{(k)})$ can be recursively computed by

$$\mathcal{A}_S^{c(k)} = \mathcal{A}_S^{c(k-1)} \bigcup f^{(k)} \tag{6}$$

and

$$Q(\boldsymbol{p}^{(k)}) = Q(\boldsymbol{p}^{(k-1)}) \bigcup f^{(k)} \tag{7}$$

where $1 \leqslant k \leqslant N_p$. The Stage-I frozen bit $f^{(k)}$ can be selected from the temporary Weight-1 set $\mathcal{W}^{(k)}$, which is determined by the Stage-I frozen construction function as follows:

$$\mathcal{W}^{(k)} = \underset{j' \in \mathcal{N}^{(k)}}{\arg} \; |\mathcal{M}_{j'}| = 1 \tag{8}$$

where $\mathcal{N}^{(k)}$ is the index set after shortening, with $\mathcal{N}^{(k)} = \mathcal{N}^{(k-1)} - \mathcal{A}_S^{c(k-1)}$ and $\mathcal{N}^{(0)} = \{0, 1, \ldots, N-1\}$.

The recursive construction algorithm for the Stage-I frozen bits and shortened bits can be described as follows.

Note that, after N_p steps, the Stage-I frozen positions and the shortened positions are jointly determined by the sets $\mathcal{A}_S^{c(N_p)}$ and $Q(\boldsymbol{p}^{(N_p)})$, respectively.

Remarks 1: In each step, only one bit is allowed to be selected from $\mathcal{W}^{(k)}$. However, the cardinality of $\mathcal{W}^{(k)}$ is greater than 1 in most cases, which means there exist more than one possible schemes to pick out $f^{(k)}$. Specifically, if we modify the Stage-I frozen construction function as

$$\mathcal{W}^{*(k)} = \max \underset{j' \in \mathcal{N}^{(k)}}{\arg} \; |\mathcal{M}_{j'}| = 1 \tag{9}$$

then the Algorithm 1 is equivalent to the Wang14 scheme presented in Ref. [10], where the last N_p indices are designated as the shortened positions and thus the frozen-bit positions. An explicit implement can be performed

Algorithm 1 The recursive Stage-I construction algorithm

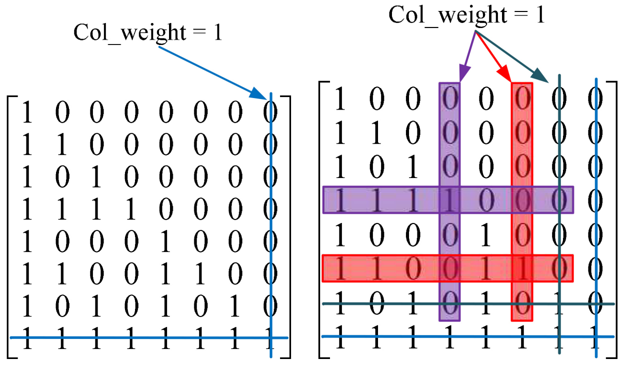
```
Given the required shortened code length $M$, the mother code length $N = 2^{\lceil \log_2 M \rceil}$, the generator matrix $\boldsymbol{G}_N$ and the number of shortened bits $N_p = N - M$;
Initialization: $\mathcal{N}^{(0)} = \{0, 1, \ldots, N-1\}$, $\mathcal{A}_S^{c(0)} = \varnothing$;
for $k = 1 : N_p$ do
    Update $\mathcal{N}^{(k)} = \mathcal{N}^{(k-1)} - \mathcal{A}_S^{c(k-1)}$;
    Compute $\mathcal{W}^{(k)}$ according to Eq. (8);
    Select the $k$-th frozen-bit position $f^{(k)}$ from $\mathcal{W}^{(k)}$;
    Compute $Q(\boldsymbol{p}^{(k)})$ according to Eq. (7);
    Compute $\mathcal{A}_S^{c(k)}$ according to Eq. (6);
end for
```

by successive backward shortening of N_p bits of the encoded codeword.

Example 1: Consider an RCSP code with $M = 6$, then we have $N = 8$ and $N_p = N - M = 2$, and the generator matrix is $\boldsymbol{G}_8 = \boldsymbol{F}^{\otimes 3}$. Figure 2 shows the recursive construction process with the Stage-I frozen construction function defined in Eq. (9). There are 2 steps to perform the construction. At the first step, only the last column $\boldsymbol{g}_7$ satisfies the Weight-1 constraint. Thus we have $\mathcal{W}^{*(1)} = \{7\}$ and $f^{(1)} = 7$. Obviously, $\boldsymbol{p}^{(1)} = (0\ 0\ 0\ 0\ 0\ 0\ 0\ 1)$ and $Q(\boldsymbol{p}^{(1)}) = \{7\}$, $\mathcal{A}_S^{c(1)} = \{7\}$, as shown in Fig. 2a. It can be seen that Column 7 and Row 7 are deleted from $\boldsymbol{G}_8$. At the second step, there exist 3 columns, $\boldsymbol{g}_3, \boldsymbol{g}_5, \boldsymbol{g}_6$ satisfy the Weight-1 constraint. According to Eq. (9), only the maximum index 6 is selected, i.e., $\mathcal{W}^{*(2)} = \{6\}$ and thus $f^{(2)} = 6$. Similarly, $\boldsymbol{p}^{(2)} = (0\ 0\ 0\ 0\ 0\ 0\ 1\ 1)$ and $Q(\boldsymbol{p}^{(2)}) = \{7, 6\}$, $\mathcal{A}_S^{c(2)} = \{7, 6\}$, as shown in Fig. 2b. At this step, Column 6 and Row 6 are deleted from $\boldsymbol{G}_8$.

3.2 The recursive Stage-I construction with RM code-aided

As discussed in the previous subsection, the Stage-I frozen construction function $\mathcal{W}^{(k)}$ is of importance, since the Stage-I frozen bit is determined by this function.



$$\begin{bmatrix} 1&0&0&0&0&0&0&0\\ 1&1&0&0&0&0&0&0\\ 1&0&1&0&0&0&0&0\\ 1&1&1&1&0&0&0&0\\ 1&0&0&0&1&0&0&0\\ 1&1&0&0&1&1&0&0\\ 1&0&1&0&1&0&1&0\\ 1&1&1&1&1&1&1&1 \end{bmatrix} \qquad \begin{bmatrix} 1&0&0&0&0&0&0&0\\ 1&1&0&0&0&0&0&0\\ 1&0&1&0&0&0&0&0\\ 1&1&1&1&0&0&0&0\\ 1&0&0&0&1&0&0&0\\ 1&1&0&0&1&1&0&0\\ 1&0&1&0&1&0&1&0\\ 1&1&1&1&1&1&1&1 \end{bmatrix}$$

(a) First step (b) Second step

Fig. 2 Recursive construction of Example 1.

To guarantee the minimum number of the Stage-I frozen bits, it is required that $|\mathcal{A}_S^{c(N_p)}| = |Q(\boldsymbol{p}^{(N_p)})| = N_p$. Therefore, the Weight-1 constraint is introduced for the Stage-I frozen construction function, as shown in Eq. (8). However, there exist more than one column with Weight-1 at the k step when $k > 1$, implying that the cardinality $|\mathcal{W}^{(k)}| > 1$ thus $f^{(k)}$ may have different construction schemes. Although the scheme according to Eq. (9) has a simple shortening pattern, the construction does not take the Hamming distance into account, which may cause performance degradation.

In this subsection, we propose a distance-greedy shortening scheme with the help of RM codes to jointly construct the Stage-I frozen bits and the corresponding shortened bits. Since polar code is the linear block code obtained by the generator matrix $\boldsymbol{G}_N$, the minimum Hamming distance is then determined by the minimum row weight of $\boldsymbol{G}_N$. Moreover, polar codes are essentially a generalization of RM codes, and they share a common generator matrix $\boldsymbol{G}_N$. We can jointly optimize the distance property of the RCSP codes under the row weight constraint of RM codes.

Let t be an element of $\mathcal{W}^{(k)}$ at the k-th step, which is also a candidate for $f^{(k)}$. According to the row weight property of RM codes, different index t shows different weight for the t-th row of $\boldsymbol{G}_N$. In order to maximize the Hamming distance, the row having the minimum weight at each construction step is deleted first. Thus, the Stage-I frozen bit $f^{(k)}$ at the k-th step can be computed by

$$f^{(k)} = \min \underset{t \in \mathcal{W}^{(k)}}{\arg} \min w_r(t) \tag{10}$$

Equation (10) indicates that the selected index t from the candidates corresponds to the t-th row of $\boldsymbol{G}_N$ with minimum row weight. In other words, the shortened bit is selected to maximize the row weight of generator matrix and thus the Hamming distance of the resulting RCSP codes can be improved.

The RM code-aided recursive construction algorithms for the Stage-I frozen bits and shortened bits can be described as follows.

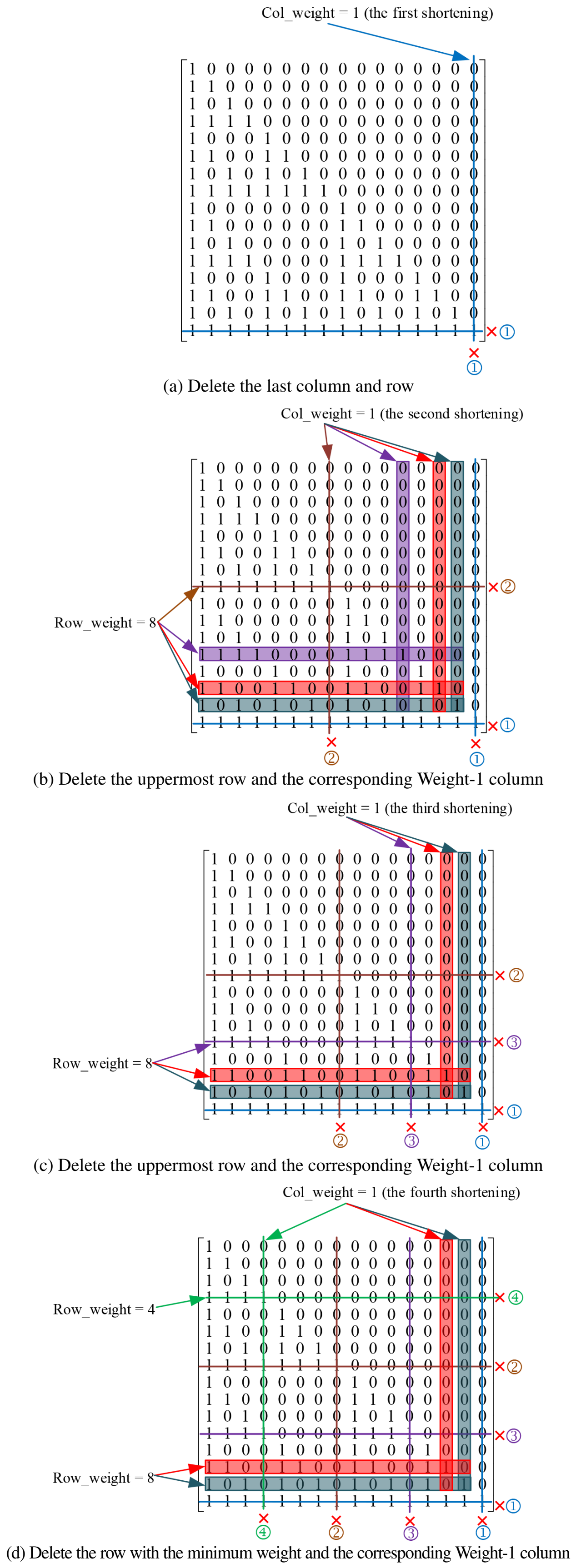
Remarks 2: Different from the Algorithm 1, the proposed recursive construction in Algorithm 2 is designed to be distance-greedy. When the candidates in $\mathcal{W}^{(k)}$ are greater than 1, we choose the one which corresponds to the minimum row weight. Note that, if the index $t \in \mathcal{W}^{(k)}$ produces the same minimum row weight $w_r(t)$, then the minimum index is selected, implying the uppermost row will be deleted.

Example 2: We construct the RCSP code with code

Algorithm 2 Recursive Stage-I construction with RM code-aided

```
Given the required shortened code length $M$, the mother code length $N = 2^{\lceil \log_2 M \rceil}$, the generator matrix $\boldsymbol{G}_N$ and the number of shortened bits $N_p = N - M$;
Initialization: $\mathcal{N}^{(0)} = \{0, 1, \cdots, N-1\}$, $\mathcal{A}_S^{c(0)} = \varnothing$;
for $k = 1 : N_p$ do
    Update $\mathcal{N}^{(k)} = \mathcal{N}^{(k-1)} - \mathcal{A}_S^{c(k-1)}$;
    Compute $\mathcal{W}^{(k)}$ according to Eq. (8);
    Compute the $k$-th pre-frozen bit $f^{(k)}$ according to Eq. (10);
    Compute $Q(\boldsymbol{p}^{(k)})$ according to Eq. (7);
    Compute $\mathcal{A}_S^{c(k)}$ according to Eq. (6);
end for
```

length $M = 12$, then we have $N = 16$ and $N_p = 4$, and the generator matrix is $\boldsymbol{G}_{16} = \boldsymbol{F}^{\otimes 4}$. Figure 3 shows the RM code-aided recursive construction of Algorithm 2, with the Stage-I frozen construction function defined in Eq. (8) and $f^{(k)}$ defined in Eq. (10). We have four steps to recursively construct the shortened code. At the first step, only the last column $\boldsymbol{g}_{15}$ satisfies the Weight-1 constraint. Thus we have $\mathcal{W}^{(1)} = \{15\}$ and $f^{(1)} = 15$. Obviously, $\boldsymbol{p}^{(1)} = (0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1)$ and $Q(\boldsymbol{p}^{(1)}) = \{15\}$, $\mathcal{A}_S^{c(1)} = \{15\}$, as shown in Fig. 3a. It can be seen that Column 15 and Row 15 are deleted from $\boldsymbol{G}_{16}$. At the second step, there exist 4 columns, $\boldsymbol{g}_7, \boldsymbol{g}_{11}, \boldsymbol{g}_{13}$, and $\boldsymbol{g}_{14}$ satisfying the Weight-1 constraint. According to Eq. (8), we have $\mathcal{W}^{(2)} = \{7, 11, 13, 14\}$. It can be seen that, the indices in $\mathcal{W}^{(2)}$ produce the same row-weight 8. According to Eq. (10), the minimum index 7 is selected as the Stage-I frozen position, thus $f^{(2)} = 7$, implying the uppermost Row 7 is deleted. Then we have $\boldsymbol{p}^{(2)} = (0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,1)$ and $Q(\boldsymbol{p}^{(2)}) = \{15, 7\}$, $\mathcal{A}_S^{c(2)} = \{15, 7\}$, as shown in Fig. 3b. At this step, Column 7 and Row 7 are deleted from $\boldsymbol{G}_{16}$. The third step is shown in Fig. 3c. There are 3 columns, $\boldsymbol{g}_{11}, \boldsymbol{g}_{13}$, and $\boldsymbol{g}_{14}$ satisfying the Weight-1 constraint. Accordingly, we had $\mathcal{W}^{(3)} = \{11, 13, 14\}$ and thus $f^{(3)} = 11$. Then, $\boldsymbol{p}^{(3)} = (0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,1\,0\,0\,0\,1)$ and $Q(\boldsymbol{p}^{(3)}) = \{15, 7, 11\}$, $\mathcal{A}_S^{c(3)} = \{15, 7, 11\}$. At this step, Column 11 and Row 11 are deleted from $\boldsymbol{G}_{16}$. At the fourth step, there exist 3 columns, $\boldsymbol{g}_3, \boldsymbol{g}_{13}$, and $\boldsymbol{g}_{14}$ satisfying the Weight-1 constraint. Thus, we had $\mathcal{W}^{(4)} = \{3, 13, 14\}$ and $f^{(4)} = 3$. Similarly, $\boldsymbol{p}^{(4)} = (0\,0\,0\,1\,0\,0\,0\,1\,0\,0\,0\,1\,0\,0\,0\,1)$, $Q(\boldsymbol{p}^{(4)}) = \{15, 7, 11, 3\}$, and $\mathcal{A}_S^{c(4)} = \{15, 7, 11, 3\}$, as shown in Fig. 3d. At this step, Column 3 and Row 3 are deleted from $\boldsymbol{G}_{16}$. Finally, the RCSP code jointly designed by the Stage-I frozen set $\mathcal{A}_S^c = \{15, 7, 11, 3\}$ and the corresponding shortening pattern $\boldsymbol{p} = (0\,0\,0\,1\,0\,0\,0\,1\,0\,0\,0\,1\,0\,0\,0\,1)$ is constructed.



Col_weight = 1 (the first shortening)
×①
(a) Delete the last column and row
Col_weight = 1 (the second shortening)
Row_weight = 8
×②
×①
(b) Delete the uppermost row and the corresponding Weight-1 column
Col_weight = 1 (the third shortening)
Row_weight = 8
×②
×③
×①
(c) Delete the uppermost row and the corresponding Weight-1 column
Col_weight = 1 (the fourth shortening)
Row_weight = 4
Row_weight = 8
×④
×②
×③
×①
(d) Delete the row with the minimum weight and the corresponding Weight-1 column

Fig. 3 RM code-aided recursive construction of Example 2.

3.3 RM code-aided construction algorithm for the Stage-II frozen bits

The shortening pattern $\boldsymbol{p}$ and Stage-I frozen set $\mathcal{A}_S^c$ are jointly designed in Subsection 3.2. With the Weight-1 constraint, the Stage-I frozen bits are computed by the shortening pattern recursively and we have $|Q(\boldsymbol{p})| = |\mathcal{A}_S^c| = N_p$. For the practical code rate requirement, the total frozen bits of the RCSP code are greater than the Stage-I frozen bits, i.e., $|\mathcal{A}^c| > N_p$, which indicates that there still have $|\mathcal{A}^c| - N_p$ frozen bits needed to be designed. Let $\mathcal{A}^{c*}$ denote the set of the remaining frozen-bit positions, called the Stage-II frozen set, with the cardinality of $|\mathcal{A}^{c*}| = |\mathcal{A}^c| - N_p$. The construction of the Stage-II frozen set is of importance, since it has a significant impact on decoding performances. The conventional GA is used in Ref. [10], where the Stage-II frozen bits are selected according to the subchannel reliabilities.

To further improve the distance property of the RCSP code, an RM code-aided construction method for the Stage-II frozen bits is presented in this subsection. Similarly, the construction is distance-greedy. The remaining frozen positions are designed to maximize the minimum Hamming distance thus the rows with the smallest weight are prone to be deleted first.

Let $\boldsymbol{G}_{N,\mathcal{A}_S^c,\boldsymbol{p}}$ be the matrix after deleting N_p columns and N_p rows according to the shortening pattern $\boldsymbol{p}$ and the Stage-I frozen set $\mathcal{A}_S^c$. Obviously, $\boldsymbol{G}_{N,\mathcal{A}_S^c,\boldsymbol{p}}$ is a submatrix of $\boldsymbol{G}_N$. After the recursive construction in Stage-I, N_p subchannels are frozen first. Let $\mathcal{N}^*$ denote the index set of the remaining subchannels, where

$$\mathcal{N}^* = \mathcal{N}^{(0)} - \mathcal{A}_S^c \tag{11}$$

Let $\mathcal{W}_{\mathcal{N}^*}$ denote the set of row weights with respect to the rows of $\boldsymbol{G}_{N,\mathcal{A}_S^c,\boldsymbol{p}}$. Obviously, $|\mathcal{W}_{\mathcal{N}^*}| = N - N_p = M$. According to the RM-rule, the weight of the i-th row can be calculated by $2^{w_t(i)}$. Then the row weight set $\mathcal{W}_{\mathcal{N}^*}$ can be determined by

$$\mathcal{W}_{\mathcal{N}^*} = \{2^{w_t(i)} | i \in \mathcal{N}^*\} \tag{12}$$

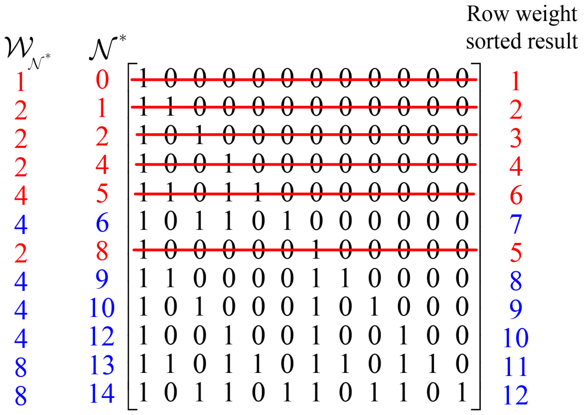
Note that the minimum Hamming distance of the constructed RCSP code is determined by the smallest element in $\mathcal{W}_{\mathcal{N}^*}$. Accordingly, the corresponding row index is then selected as the Stage-II frozen position. Such construction scheme can be performed by sorting the elements of $\mathcal{W}_{\mathcal{N}^*}$, then select the $|\mathcal{A}^c| - N_p$ corresponding row indices with smallest weight to construct the Stage-II frozen set. The proposed scheme can be described as follows.

Note that, after performing the Stage-I construction, there still exist rows with small weight in $\boldsymbol{G}_{N,\mathcal{A}_S^c,\boldsymbol{p}}$, which may decrease the average row weight and cause the performance loss. Such rows are deleted in the Stage-II construction and the corresponding indices are picked out to construct the frozen set $\mathcal{A}^{c*}$. Furthermore, the resulting RCSP code is constructed by Algorithm 2 and Algorithm 3 based on the distance-greedy scheme. It is worthwhile to point out that, both of the frozen sets $\mathcal{A}_S^c$ and $\mathcal{A}^{c*}$ are computed by the RM-rule constraint, i.e., the proposed RCSP code can be constructed without the complicated GA method.

Example 3: We constructed the RCSP code with code length $M = 12$ and message length $K = 6$, then we have $N = 16$, $N_p = 4$. After performing the Stage-I construction, the Stage-I frozen set $\mathcal{A}_S^c = \{15, 7, 11, 3\}$ and the shortening pattern $\boldsymbol{p} = (0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1)$. The generator matrix $\boldsymbol{G}_{16,\mathcal{A}_S^c,\boldsymbol{p}}$ is 12×12 after deleting the rows and columns of $\boldsymbol{G}_{16}$ with indices in $\mathcal{A}_S^c$ and $Q(\boldsymbol{p})$, respectively. Figure 4 shows the process of the RM construction for the Stage-II frozen bits of Algorithm 3. According to Eqs. (11)

Algorithm 3 RM construction for Stage-II frozen bits

```
1: Given the required shortened code length $M$, the mother code length $N = 2^{\lceil \log_2 M \rceil}$, the information length $K$, the generator matrix $\boldsymbol{G}_N$ and the number of shortened bits $N_p = N - M$;
2: Perform the Algorithm 2 and get the shortening pattern $\boldsymbol{p}$, the Stage-I frozen set $\mathcal{A}_S^c$ and the submatrix $\boldsymbol{G}_{N,\mathcal{A}_S^c,\boldsymbol{p}}$;
3: Get $\mathcal{N}^*$ according to Eq. (11) and get $\mathcal{W}_{\mathcal{N}^*}$ according to Eq. (12);
4: Sort the elements of $\mathcal{W}_{\mathcal{N}^*}$ in an ascending order;
5: Choose the first $|\mathcal{A}^c| - N_p$ sorted elements and determine the corresponding row indices;
6: Construct the Stage-II frozen set $\mathcal{A}^{c*}$ using the row indices in Step 5;
7: Get $\mathcal{A}^c = \mathcal{A}^{c*} \bigcup \mathcal{A}_S^c$.
```



$\mathcal{W}_{\mathcal{N}^*}$	$\mathcal{N}^*$		Row weight sorted result
1	0	~~1 0 0 0 0 0 0 0 0 0 0 0~~	1
2	1	~~1 1 0 0 0 0 0 0 0 0 0 0~~	2
2	2	~~1 0 1 0 0 0 0 0 0 0 0 0~~	3
2	4	~~1 0 0 1 0 0 0 0 0 0 0 0~~	4
4	5	~~1 1 0 1 1 0 0 0 0 0 0 0~~	6
4	6	1 0 1 1 0 1 0 0 0 0 0 0	7
2	8	~~1 0 0 0 0 0 1 0 0 0 0 0~~	5
4	9	1 1 0 0 0 0 1 1 0 0 0 0	8
4	10	1 0 1 0 0 0 1 0 1 0 0 0	9
4	12	1 0 0 1 0 0 1 0 0 1 0 0	10
8	13	1 1 0 1 1 0 1 1 0 1 1 0	11
8	14	1 0 1 1 0 1 1 0 1 1 0 1	12

Fig. 4 RM code-aided construction of Example 3.

and (12), $\mathcal{N}^* = \{0, 1, 2, 4, 5, 6, 8, 9, 10, 12, 13, 14\}$ and $\mathcal{W}_{\mathcal{N}^*} = \{1, 2, 2, 2, 4, 4, 2, 4, 4, 4, 8, 8\}$. Then, an ascending order sorting of $\mathcal{W}_{\mathcal{N}^*}$ is performed. The sorted result is $(1, 2, 3, 4, 6, 7, 5, 8, 9, 10, 11, 12)$, which represents the row positions after sorting. The first $|\mathcal{A}^c| - N_p = 6$ sorted indices in $\mathcal{N}^*$ are selected to be the Stage-II frozen positions. Thus, the Stage-II frozen set $\mathcal{A}^{c*} = \{0, 1, 2, 4, 8, 5\}$ and the corresponding rows are deleted, as shown in Fig. 4. Consequently, the RCSP code can be constructed with the frozen set $\mathcal{A}^c = \mathcal{A}^{c*} \bigcup \mathcal{A}^c_S = \{15, 7, 11, 3, 0, 1, 2, 4, 8, 5\}$ and the shortening pattern $\boldsymbol{p} = (0\,0\,0\,1\,0\,0\,0\,1\,0\,0\,0\,1\,0\,0\,0\,1)$.

3.4 RM-GA construction algorithms for the Stage-II frozen bits

In Section 3.3, the Stage-II frozen set $\mathcal{A}^{c*}$ is designed to maximize the Hamming distance based on the RM-rule only, which can greatly simplify the code construction. However, without the GA evaluation, some subchannels with small reliabilities are not involved in $\mathcal{A}^{c*}$, but such "bad" subchannels may have impact on the block error probability, as pointed out in Ref. [10]. To solve this problem, we present an algorithm combining both of the RM-rule constraint and the GA tool to construct the RCSP code in this subsection, called the RM-GA construction. The proposed algorithm is first designed by the Hamming distance constraint with the RM-rule, then the remaining subchannels are determined by the GA tool, which can further improve the decoding performance.

For an RM code $\mathrm{RM}(n, r)$ with the minimum Hamming distance $d_{\min} = 2^{(n-r)}$, the number of rows with weights $\geqslant d_{\min}$ can be calculated by

$$\sum_{i=0}^{r} \binom{n}{i} \tag{13}$$

Since the number of the information bits is K, to ensure the Stage-II construction satisfying the RM constraint, we needed to find a parameter $r_{\min}$ that satisfies

$$r_{\min} = \min \left\{ r \,\middle|\, \sum_{i=0}^{r} \left(\binom{n}{i} - d_{n,i} \right) \geqslant K \right\} \tag{14}$$

where $d_{n,i}$ denotes the number of rows with weight 2^{n-i} in $\mathcal{A}^c_S$. Equation (14) indicates that there exist at leat K rows with weight $\geqslant d^*_{\min}$ after the Stage-I construction, where $d^*_{\min} = 2^{n-r_{\min}}$. Let T be the number of rows in $\boldsymbol{G}_{N, \mathcal{A}^c_S, \boldsymbol{p}}$ with weight $\geqslant d^*_{\min}$, which can be calculated as follows:

$$T = \sum_{i=0}^{r_{\min}} \left(\binom{n}{i} - d_{n,i} \right) \tag{15}$$

Accordingly, the Stage-II frozen set $\mathcal{A}^{c*}$ can be constructed by the following two cases: (1) For $T > K$, $\mathcal{A}^{c*}$ can be constructed by selecting the positions with row weight $\leqslant d^*_{\min}$. Moreover, the corresponding frozen subchannels must have the smallest reliabilities under the GA evaluation; (2) for $T = K$, $\mathcal{A}^{c*}$ can be constructed by selecting the positions with row weight $\leqslant d^*_{\min}/2$, since all the K positions with weight $\geqslant d^*_{\min}$ are selected as the information set in this case. Similarly, the selected frozen subchannels in $\mathcal{A}^{c*}$ have the smallest reliabilities under the GA evaluation.

The proposed RM-GA construction algorithm can be described as follows.

Remarks 3: The Stage-II construction can be seen as an RM code with parameters $(n, r_{\min})$ for the generator matrix $\boldsymbol{G}_{N, \mathcal{A}^c_S, \boldsymbol{p}}$. Different from the conventional RM code, the GA is also involved in the construction. For the rows with equal weight, the ones with smaller reliabilities are first selected as the frozen positions.

Note that, if we only considered the GA evaluation in the construction, i.e., the first $|\mathcal{A}^c| - N_p$ positions were selected to form $\mathcal{A}^{c*}$ in Step 6 without RM-rule constraint, then we got a simplified variation of Algorithm 4, called the GA construction.

Example 4: For convenience of comparisons, we considered the construction of the RCSP code with the same parameters as Example 3. After performing the Stage-I construction, we had the Stage-I set $\mathcal{A}^c_S = \{15, 7, 11, 3\}$ and the shortening pattern $\boldsymbol{p} = (0\,0\,0\,1\,0\,0\,0\,1\,0\,0\,0\,1\,0\,0\,0\,1)$. The generator matrix $\boldsymbol{G}_{16, \mathcal{A}^c_S, \boldsymbol{p}}$ is obtained by deleting the rows and columns of $\boldsymbol{G}_{16}$ with indices in $\mathcal{A}^c_S$ and $Q(\boldsymbol{p})$, respectively. According to Eqs. (11) and (12),

Algorithm 4 RM-GA construction for Stage-II frozen bits

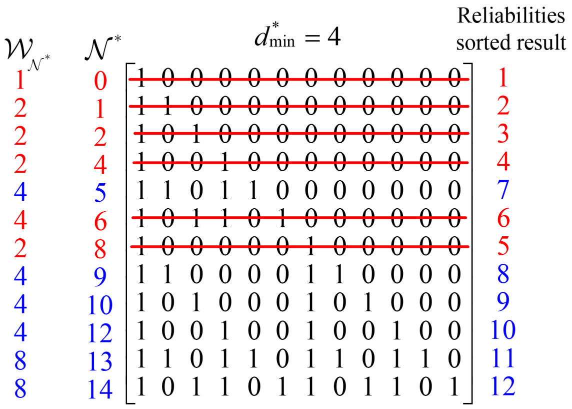
```
1: Given the required shortened code length $M$, the mother code length $N = 2^{\lceil \log_2 M \rceil}$, the generator matrix $\boldsymbol{G}_N$ and the number of shortened bits $N_p = N - M$;
2: Perform the Algorithm 2 and get the shortening pattern $\boldsymbol{p}$, the Stage-I frozen set $\mathcal{A}^c_S$ and the corresponding generator matrix $\boldsymbol{G}_{N, \mathcal{A}^c_S, \boldsymbol{p}}$;
3: Get $\mathcal{N}^*$ according to Eq. (11) and get $\mathcal{W}_{\mathcal{N}^*}$ according to Eq. (12);
4: Determine the $r_{\min}$ according to Eq. (14) and $T$ according to Eq. (15);
5: Perform the GA and sort the reliabilities in an ascending order;
6: Select the first $|\mathcal{A}^c| - N_p$ positions of the sorted reliabilities with row weight $\leqslant d^*_{\min}$ (for $T > K$ ) or row weight $\leqslant d^*_{\min}/2$ (for $T = K$ ), add the corresponding indices to $\mathcal{A}^{c*}$;
7: Get $\mathcal{A}^c = \mathcal{A}^{c*} \bigcup \mathcal{A}^c_S$.
```

we have $\mathcal{N}^* = \{0, 1, 2, 4, 5, 6, 8, 9, 10, 12, 13, 14\}$ and $\mathcal{W}_{\mathcal{N}^*} = \{1, 2, 2, 2, 4, 4, 2, 4, 4, 4, 8, 8\}$. Figure 5 shows the process of the RM-GA construction for stage-II frozen bits of Algorithm 4. The parameter $r_{\min} = 2$ according to Eq. (14) and $T = 7$ according to Eq. (15), which indicates that the minimum row weight $d^*_{\min} = 2^{n-r_{\min}} = 4$. Perform the GA evaluation then sort the reliabilities in an ascending order. The sorted result is $(1, 2, 3, 4, 7, 6, 5, 8, 9, 10, 11, 12)$, which represents the positions of the subchannel after sorting. Thus, the first $|\mathcal{A}^c| - N_p = 6$ positions with row weight $\leqslant 4$ are selected, then add the corresponding indices to $\mathcal{A}^{c*}$ and we have $\mathcal{A}^{c*} = \{0, 1, 2, 4, 8, 6\}$. The corresponding rows with indices in $\mathcal{A}^{c*}$ are deleted, as shown in Fig. 5. Finally, the RCSP code is constructed by the frozen set $\mathcal{A}^c = \mathcal{A}^{c*} \bigcup \mathcal{A}^c_S = \{15, 7, 11, 3, 0, 1, 2, 4, 8, 6\}$ and the shortening pattern $\boldsymbol{p} = (0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1)$.

3.5 An outline of the two-stage constructions

In this subsection, an outline of the two-stage constructions to illustrate the main work of this paper is given.

As shown in Fig. 6, there are two stages to construct the RCSP codes. The basic constraint of the RCSP code is given in Algorithm 1, where the shortening pattern and the frozen bit are jointly designed to make the shortened bits be completely known by the decoder. Thus the decoder can set the LLRs of the shortened bits to be infinity (or minus infinity), which can improve the performance, as pointed out in Ref. [10]. The Stage-I frozen set $\mathcal{A}^c_S$ and the shortening pattern $\boldsymbol{p}$ can be determined by the RM-aided (Algorithm 2) proposed in this paper or the successive backward shortening scheme in Ref. [10]. Then three methods are presented to construct the Stage-II frozen set $\mathcal{A}^{c*}$ based on the Algorithm 2, which are the RM construction (Algorithm 3), the GA construction (simplified Algorithm 4), and the RM-GA construction (Algorithm 4). If $\mathcal{A}^{c*}$ is constructed by the GA evaluation based on the conventional successive shortening, the resulting algorithm is equivalent to the Wang14 scheme. Note that, the Stage-I and Stage-II frozen sets form the set of total frozen bits, i.e., $\mathcal{A}^c = \mathcal{A}^{c*} \bigcup \mathcal{A}^c_S$.



$\mathcal{W}_{\mathcal{N}^*}$	$\mathcal{N}^*$	$d^*_{\min} = 4$	Reliabilities sorted result
1	0	~~1 0 0 0 0 0 0 0 0 0 0 0~~	1
2	1	~~1 1 0 0 0 0 0 0 0 0 0 0~~	2
2	2	~~1 0 1 0 0 0 0 0 0 0 0 0~~	3
2	4	~~1 0 0 1 0 0 0 0 0 0 0 0~~	4
4	5	1 1 0 1 1 0 0 0 0 0 0 0	7
4	6	~~1 0 1 1 0 1 0 0 0 0 0 0~~	6
2	8	~~1 0 0 0 0 0 1 0 0 0 0 0~~	5
4	9	1 1 0 0 0 0 1 1 0 0 0 0	8
4	10	1 0 1 0 0 0 1 0 1 0 0 0	9
4	12	1 0 0 1 0 0 1 0 0 1 0 0	10
8	13	1 1 0 1 1 0 1 1 0 1 1 0	11
8	14	1 0 1 1 0 1 1 0 1 1 0 1	12

Fig. 5 RM-GA construction of Example 4.

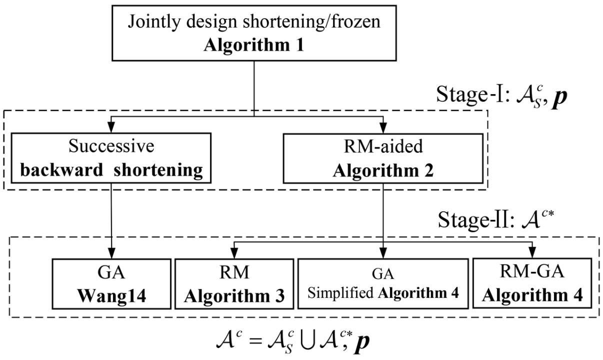

Jointly design shortening/frozen
Algorithm 1
Stage-I: $\mathcal{A}^c_S, \boldsymbol{p}$
Successive backward shortening
RM-aided Algorithm 2
Stage-II: $\mathcal{A}^{c*}$
GA Wang14
RM Algorithm 3
GA Simplified Algorithm 4
RM-GA Algorithm 4
$\mathcal{A}^c = \mathcal{A}^c_S \bigcup \mathcal{A}^{c*}; \boldsymbol{p}$

Fig. 6 An outline of the two-stage constructions.

4 Simulation Results

4.1 Row weight property

The minimum Hamming distance of the resulted RCSP code can be analyzed by the row weight property of $\boldsymbol{G}_{N,\mathcal{A}^c,\boldsymbol{p}}$ after the Stage-I and Stage-II constructions. In this subsection, we compared the row weight distributions with different code lengths and code rates for three different Stage-II construction methods, which are the RM construction in Algorithm 3 (called RM), the RM-GA construction in Algorithm 4 (called RM-GA), and the GA construction in the simplified variation of Algorithm 4 (called GA). The comparisons are performed based on the same Stage-I frozen set $\mathcal{A}^c_S$.

Consider the RCSP code with $N = 256$, $M = 192$, $N_p = 64$, $R = 0.25, 0.50$, and 0.75, respectively. After constructing the Stage-I frozen set $\mathcal{A}^c_S$, there exist $M = 192$ rows. Then after performing the Stage-II frozen set $\mathcal{A}^{c*}$, there exist 48, 96, and 144 remaining rows of the generator matrix $\boldsymbol{G}_{256,\mathcal{A}^c,\boldsymbol{p}}$ for the above three code rates. Table 1 shows the distributions of the row number with different weights of $\boldsymbol{G}_{256,\mathcal{A}^c,\boldsymbol{p}}$. From Table 1, we have the following observations.

- For code rate $R = 0.25$, there exist 5 rows with minimum Weight-16 for the GA; while there are 33 rows with minimum Weight-32 both for the RM and RM-GA;
- For code rate $R = 0.50$, there exist 5 rows with minimum Weight-8 for the GA; while there are 45 rows with minimum Weight-16 both for the RM and RM-GA;

Table 1 Row weight distributions under different construction schemes ($N = 256, N_p = 64$, SNR = 3 dB).

Scheme	R	Row weight					
		4	8	16	32	64	128
RM	0.25	0	0	0	33	13	2
	0.50	0	0	45	36	13	2
	0.75	0	38	55	36	13	2
GA	0.25	0	0	5	28	13	2
	0.50	0	5	40	36	13	2
	0.75	2	36	55	36	13	2
RM-GA	0.25	0	0	0	33	13	2
	0.50	0	0	45	36	13	2
	0.75	0	38	55	36	13	2

- For code rate $R = 0.75$, there exist 2 rows with minimum Weight-4 for the GA; while there are 38 rows with minimum Weight-8 both for the RM and RM-GA;
- The RM and RM-GA have the same row weight distributions and they can efficiently increase the minimum row weight for different code rates.

We also consider another RCSP code with a longer block length $N = 512$, as shown in Table 2. The row weight distributions are analyzed under the following parameters, $M = 384$, $N_p = 128$, $R = 0.25$, 0.50, and 0.75, respectively. We have a similar observation as follows.

- For code rate $R = 0.25$, there exists one row with minimum Weight-16 for the GA; while there are 30 and 34 rows with minimum Weight-32 for the RM and RM-GA, respectively;
- For code rate $R = 0.50$, there exists one row with minimum Weight-8 for the GA; while there are 35 and 42 rows with minimum Weight-16 for the RM and RM-GA, respectively;
- For code rate $R = 0.75$, there exists one row with minimum Weight-4 for the GA; while there are 26 and 34 rows with minimum Weight-8 for the RM and RM-GA, respectively.

Table 2 Row weight distributions under different construction schemes ($N = 512, N_p = 128$, SNR = 3 dB).

Scheme	R	Row weight						
		4	8	16	32	64	128	256
RM	0.25	0	0	0	30	49	15	2
	0.50	0	0	35	91	49	15	2
	0.75	0	26	105	91	49	15	2
GA	0.25	0	0	1	33	45	15	2
	0.50	0	1	42	83	49	15	2
	0.75	1	33	97	91	49	15	2
RM-GA	0.25	0	0	0	34	45	15	2
	0.50	0	0	42	84	49	15	2
	0.75	0	34	97	91	49	15	2

In summary, the proposed RM and RM-GA schemes can efficiently increase the minimum row weight for different code lengths and code rates, which show better distance property than that of the GA-only construction and may have a positive impact on the decoding performance, as shown in the performance analysis in the next subsection.

4.2 Complexity analysis

The total complexity can be divided into encoding complexity and decoding complexity. Since decoding an RCSP code by different constructions has the same complexity, we only discuss the encoding complexity here.

The encoding process consists of two steps, the Stage-I construction and the Stage-II construction in this paper.

(1) For the Stage-I, the target is to construct the Stage-I frozen set $\mathcal{A}_S^c$, which requires N_p steps. In each step, the complexity is caused by two operations. One is to find the Weight-1 columns and the other is to find the row with minimum weight. We denote such complexity in each step as $C_{\text{Stage-I}}$. Therefore, there requires $N_p C_{\text{Stage-I}}$ complexity in Stage-I for the presented GA construction, the RM construction and the RM-GA construction. By contrast, the Stage-I frozen set $\mathcal{A}_S^c$ for the Wang14 is simply constructed by selecting the last N_p positions successively. Hence, the complexity can be ignored and we denote 0 complexity in this case.

(2) For the Stage-II, the target is to construct the Stage-II frozen set $\mathcal{A}^{c*}$. For the Wang14, it performs the Gaussian approximation tool to select $|\mathcal{A}^{c*}|$ positions, which is the same as the presented GA construction. We denote such complexity as C_{GA}. For the presented RM construction, it requires two steps to construct $\mathcal{A}^{c*}$. The first step is to compute the weights of $N - N_p$ rows of the corresponding generator matrix. The second step is to find the $|\mathcal{A}^{c*}|$ rows with minimum weight. The complexity caused by these two steps are denoted by $C_{\text{Stage-II}}$. Obviously, for the presented RM-GA constructions, the complexity is $C_{\text{GA}} + C_{\text{Stage-II}}$.

The computational complexity for different constructions are shown in Table 3.

Table 3 Computational complexity with different constructions.

Scheme	Encoding complexity	
	Stage-I	Stage-II
Wang14	0	C_{GA}
GA	$N_p C_{\text{Stage-I}}$	C_{GA}
RM	$N_p C_{\text{Stage-I}}$	$C_{\text{Stage-II}}$
RM-GA	$N_p C_{\text{Stage-I}}$	$C_{\text{GA}} + C_{\text{Stage-II}}$

4.3 Decoding performance

In this subsection, we give the FER performances of the proposed RCSP codes. The codes are constructed with the same Stage-I frozen set in Algorithm 2. Three different Stage-II frozen sets are considered, which are the RM-GA in Algorithm 4, the RM in Algorithm 3, and the GA construction in simplified Algorithm 4. For comparisons, we also consider the Wang14 scheme in Ref. [10] and the QUP scheme in Ref. [9]. In the simulations, the GA method is used for the channel reliabilities estimation both of the Wang14 and the QUP schemes. All the codes are decoded by the SCL algorithm with list size $L = 4$ under the binary input additive white Gaussian noise (AWGN) channel. The total number of simulation frames is 10^7, and the maximum number of error frames is 300. When the simulation reaches the total number of frames or reaches the maximum number of error frames, the simulation is stopped.

Figure 7 shows the performances of the polar codes with $M = 192$, $N_p = 64$, and mother code length $N = 256$. The code rates are set to be $R = 0.25$, 0.50, and 0.75, respectively. We had the following observations.

- For code rate $R = 0.25$, the RM-GA construction, the RM construction and the QUP scheme have similar performances and they outperform the other two schemes. For example, at the FER $= 10^{-3}$, there is a performance gain of about 0.70 dB compared with the Wang14 scheme and the GA construction.
- For code rate $R = 0.50$, the RM-GA construction performs slightly better than the RM construction and both of them perform better than other three algorithms. For example, the RM-GA construction outperforms the Wang14 scheme, the GA construction, the QUP scheme about 0.80 dB, 0.80 dB, and 0.50 dB, respectively, at the FER $= 10^{-3}$.
- For code rate $R = 0.75$, the RM-GA construction and the RM construction almost have the same performances and both of them perform better than other three schemes. For example, at the FER $= 10^{-3}$, there is a gain of about 0.70 dB compared with the QUP scheme and the Wang14 scheme, and 0.50 dB compared with the GA construction.

Figure 8 shows the performances of the polar codes with $M = 384$, $N_p = 128$, and a longer mother code length $N = 512$. Similarly, the code rates are set to be $R = 0.25$, 0.50, and 0.75, respectively.

- For code rate $R = 0.25$, the RM-GA construction has the best performance than other schemes. For example, at the FER $= 10^{-3}$, there is a gain of about 0.20 dB, 0.45 dB, 0.50 dB, and 0.70 dB compared with the RM construction, the QUP scheme, the GA construction, and the Wang14 scheme, respectively. The RM construction performs as well as the QUP scheme, the GA construction and the Wang14 scheme at low signal noise ratio (SNR) region, but outperforms them at high SNR region about 0.25 dB, 0.30 dB, and 0.50 dB, respectively, at the FER $= 10^{-3}$.
- For code rate $R = 0.50$, the RM-GA construction still performs the best than other schemes. For example, it outperforms the RM construction, the GA construction,

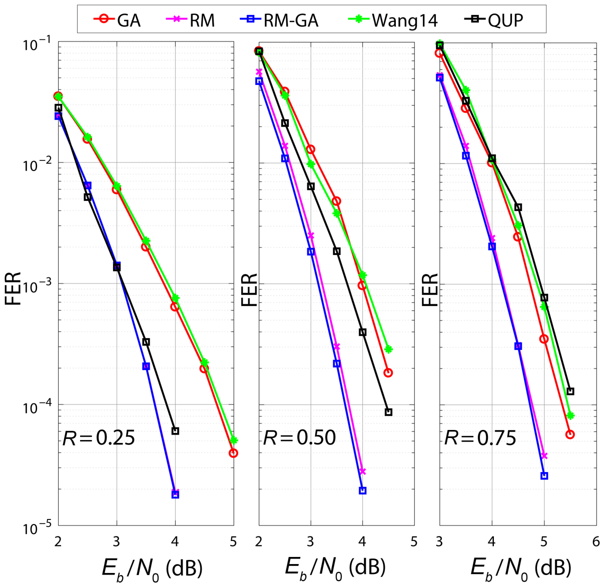


Fig. 7 Performance comparisons under different rate matching algorithms with $M = 192$.

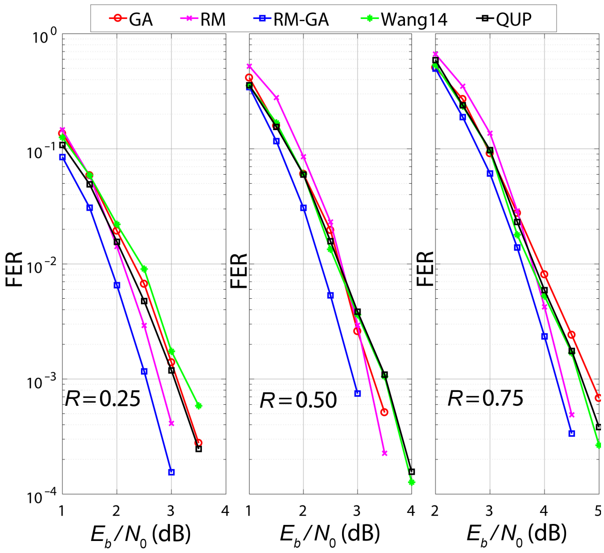


Fig. 8 Performance comparisons under different rate matching algorithms with $M = 384$.

the Wang14 scheme, and the QUP scheme about 0.25 dB, 0.35 dB, 0.55 dB, and 0.55 dB, respectively, at the FER $= 10^{-3}$. The RM construction performs slightly worse than the GA construction at low SNR region. However it has a steeper FER curve and outperforms the GA construction about 0.10 dB at FER $= 10^{-3}$. Both of the RM and GA constructions have better performances than the QUP and Wang14 schemes.

- For code rate $R = 0.75$, we had a similar observation. The RM-GA construction has the best performance than other schemes. For example, at the FER $= 10^{-3}$, there is a gain of about 0.1 dB compared with the RM construction, 0.4 dB compared with the Wang14 scheme, 0.45 dB compared with the QUP scheme, and 0.6 dB compared with the GA construction. The RM construction has better performances than other three algorithms at the high SNR region. For example, at the FER $= 10^{-3}$, the RM construction outperforms the Wang14 scheme, the QUP scheme, and the GA construction about 0.30 dB, 0.35 dB, and 0.50 dB, respectively.

The experimental simulation results above show that the proposed three construction methods achieve excellent FER performances with different code lengths and code rates. The RM-GA construction combines both of the subchannel reliabilities and the distance property to construct the Stage-II frozen set, thus it has the best performance. In contrast, the RM construction determines the Stage-II frozen set only by the distance property with the help of RM code, thus it has the simplest structure since it can avoid the complicated GA evaluation. Nevertheless, the RM construction still performs better than the Wang14 and QUP schemes, especially in high SNR region. Moreover, it is shown that for the rate compatible polar codes, the shortening works better for high code rates while the puncturing works better for low code rates in general[30]. However, the proposed RM-GA construction always shows the best performances at different code rates.

5 Conclusion

In this study, we had proposed the two-stage constructions for the RCSP codes with flexible code length and code rate. For the Stage-I construction, the frozen set and the shortening pattern are jointly designed to enhance the minimum Hamming distance with the RM code-aided. Then we had further presented three construction schemes, called the RM construction, the RM-GA construction, and the GA construction to design the Stage-II frozen set. Simulation results show that the proposed RCSP codes provide good FER performances at different code lengths and code rates. More specifically, the RM-GA construction performs best and can achieve at most 0.80 dB gain compared to the Wang14 and the QUP schemes. The RM construction has the simplest structure while it still perform better than the existing shortening/puncturing schemes, especially at high SNR region.

Acknowledgment

The authors are very grateful to the editor and reviewers for their comments and constructive suggestions, which help to enrich the content and improve the presentation of this paper. This work was supported by the Interdisciplinary Scientific Research Foundation of GuangXi University (No. 2022JCC015), the National Natural Science Foundation of China (Nos. 61761006, 61961004, and 61762011), and the Natural Science Foundation of Guangxi of China (Nos. 2017GXNSFAA198263 and 2018GXNSFAA294059).

References

[1] E. Arikan, 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, 2009.

[2] 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.

[3] TS38.212 V.15.1.0, NR; Multiplexing and channel coding, https://www.3gpp.org/ DynaReport/38-series.htm, 2018.

[4] C. Sun, Z. S. Fei, D. Jia, C. Z. Cao, and X. Y. Wang, Secure transmission scheme for parallel relay channels based on polar coding, *Tsinghua Science and Technology*, vol. 23, no. 3, pp. 357–365, 2018.

[5] J. Du, J. Song, Y. Ren, and J. T. Wang, Convergence of broadband and broadcast/multicast in maritime information networks, *Tsinghua Science and Technology*, vol. 26, no. 5, pp. 592–607, 2021.

[6] TS38.212, V15.0.0, NR; Multiplexing and channel coding, https://www.3gpp.org/ DynaReport/38-series.htm, 2017.

[7] F. Gabry, V. Bioglio, I. Land, and J. C. Belfiore, Multi-kernel construction of polar codes, in *Proc. 2017 IEEE Int. Conf. on Communications Workshops* (*ICC Workshops*), Paris, France, 2017, pp. 761–765.

[8] A. Eslami and H. Pishro-Nik, A practical approach to polar codes, in *Proc. 2011 IEEE Int. Symp. on Information Theory Proc.*, St. Petersburg, Russia, 2011, pp. 16–20.

[9] K. Niu, K. Chen, and J. R. Lin, Beyond turbo codes: Rate-compatible punctured polar codes, in *Proc. 2013 IEEE Int. Conf. on Communications* (*ICC*), Budapest, Hungary, 2013, pp. 3423–3427.

[10] R. X. Wang and R. K. Liu, A novel puncturing scheme for

polar codes, *IEEE Commun. Lett.*, vol. 18, no. 12, pp. 2081–2084, 2014.

[11] K. Niu, J. C. Dai, K. Chen, J. R. Lin, Q. T. Zhang, and A. V. Vasilakos, Rate-compatible punctured polar codes: Optimal construction based on polar spectra, arXiv preprint arXiv: 1612.01352, 2016.

[12] 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, 2012.

[13] V. Miloslavskaya, Shortened polar codes, *IEEE Trans. Inf. Theory*, vol. 61, no. 9, pp. 4852–4865, 2015.

[14] B. Li, H. Shen, and D. Tse, A RM-polar codes, arXiv preprint arXiv: 1407.5483, 2014.

[15] M. Bakshi, S. Jaggi, and M. Effros, Concatenated polar codes, in *Proc. 2010 IEEE Int. Symp. on Information Theory*, Austin, TX, USA, 2010, pp. 918–922.

[16] I. Tal and A. Vardy, List decoding of polar codes, in *Proc. 2011 IEEE Int. Symp. on Information Theory Proc.*, St. Petersburg, Russia, 2011, pp. 1–5.

[17] K. Niu and K. Chen, CRC-aided decoding of polar codes, *IEEE Commun. Lett.*, vol. 16, no. 10, pp. 1668–1671, 2012.

[18] E. Arikan, Serially concatenated polar codes, *IEEE Access*, vol. 6, pp. 64549–64555, 2018.

[19] T. Wang, D. M. Qu, and T. Jiang, Parity-check-concatenated polar codes, *IEEE Commun. Lett.*, vol. 20, no. 12, pp. 2342–2345, 2016.

[20] R. Gallage, Low-density parity-check codes, *IRE Trans. Inf. Theory*, vol. 8, no. 1, pp. 21–28, 1962.

[21] P. Wang, L. G. Yin, and J. H. Lu, Efficient helicopter-satellite communication scheme based on check-hybrid LDPC coding, *Tsinghua Science and Technology*, vol. 23, no. 3, pp. 323–332, 2018.

[22] B. H. Lin, Y. K. Pei, L. G. Yin, and J. H. Lu, Design and efficient hardware implementation schemes for non-Quasi-Cyclic LDPC codes, *Tsinghua Science and Technology*, vol. 22, no. 1, pp. 92–103, 2017.

[23] C. Berrou, A. Glavieux, and P. Thitimajshima, Near Shannon limit error-correcting coding and decoding: Turbo-codes. 1, in *Proc. ICC'93–IEEE Int. Conf. on Communications*, Geneva, Switzerland, 1993, pp. 1064–1070.

[24] D. E. Muller, Application of Boolean algebra to switching circuit design and to error detection, *Trans.I.R.E. Prof. Group Electron. Comput.*, vol. EC-3, no. 3, pp. 6–12, 1954.

[25] I. Reed, A class of multiple-error-correcting codes and the decoding scheme, *Trans. IRE Prof. Group on Inf. Theory*, vol. 4, no. 4, pp. 38–49, 1954.

[26] E. Arikan, A survey of reed-muller codes from polar coding perspective, in *Proc. 2010 IEEE Information Theory Workshop on Information Theory (ITW)*, Cairo, Egypt, 2010, pp. 1–5.

[27] R. Mori and T. Tanaka, Performance of polar codes with the construction using density evolution, *IEEE Commun. Lett.*, vol. 13, no. 7, pp. 519–521, 2009.

[28] P. Trifonov, Efficient design and decoding of polar codes, *IEEE Trans. Commun.*, vol. 60, no. 11, pp. 3221–3227, 2012.

[29] G. N. He, J. C. Belfiore, I. Land, G. H. Yang, X. C. Liu, Y. Chen, R. Li, J. Wang, Y. Q. Ge, R. Zhang, et al., Beta-expansion: A theoretical framework for fast and recursive construction of polar codes, in *Proc. 2017 IEEE Global Communications Conf.*, Singapore, 2017, pp. 1–6.

[30] V. Bioglio, F. Gabry, and I. Land, Low-complexity puncturing and shortening of polar codes, in *Proc. 2017 IEEE Wireless Communications and Networking Conf. Workshops (WCNCW)*, San Francisco, CA, USA, 2017, pp. 1–6.

Chunjie Li received the BS degree from Guangxi University, Nanning, China, in 2018. He is currently pursuing the MS degree at Guangxi University, Nanning, China. His research interests focus on channel coding, especially the polar code.

Haiqiang Chen received the MS degree from Guangxi University, Nanning, China, in 2002 and the PhD degree from Sun Yat-sen University, Guangzhou, China, in 2011. He is currently a professor and also the instructor of graduate student in Guangxi University, Nanning, China. His research interests include coding theory and relay system.

Zelin Wang received the BS degree from Bengbu University, Bengbu, China, in 2019. He is currently pursuing the MS degree at Guangxi University, Nanning, China. His research interests mainly focus on channel coding, especially the polar code.

Youming Sun received MS degree from Guangxi Normal University, Guilin, China, in 2006 and the PhD degree from South China University of Technology, Guangzhou, China, in 2017. His research interests include classical and modern coding theory, LDPC codes, and their applications on communications.

Tuanfa Qin received the PhD degree from Nanjing University, Nanjing, China, in 1997. Now he is a professor at Guangxi University, Nanning, China. He is also the senior member of China Institute of Electronics, senior member of China Communications Institute. His research interests include network coding, multimedia communications, video encoding, and image retrieval.

Xiangcheng Li received the BS and MS degrees in electrical information engineering from Guangxi University, Nanning, China, in 2002 and 2008, respectively, and the PhD degree from the South China University of Technology, Guangzhou, China, in 2017. His research interests include channel coding, relay system, and signal processing for digital communications.