IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 29, 2019, accepted October 21, 2019, date of publication October 25, 2019, date of current version November 6, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2949652

CRC-Aided Parity-Check Polar Coding

FENGYI CHENG, AIJUN LIU, (Member, IEEE), JING REN, AND KAI FENG
Army Engineering University of PLA, Nanjing 210007, China

Corresponding author: Aijun Liu (liuaj.cn@163.com)

This work was supported by the National Natural Science Foundation of China, Foundation Item: Physical Layer Secrecy Coding Technology Based on Polar Codes, under Grant 61501508.

ABSTRACT Parity-check (PC) polar codes can yield better error-correcting performance compared with the cyclic redundancy check (CRC) aided polar codes under successive cancellation list decoder. However, PC bits are incapable of detecting error as effective as CRC. To overcome this shortage, this paper proposes a scheme of CRC-aided PC polar codes. The proposed scheme can detect error before decoding is completed and outperform the standard CRC-assisted polar codes with better capability of error detecting.

INDEX TERMS Polar codes, cyclic redundancy check (CRC), successive cancellation list decoder, parity-check coding.

I. INTRODUCTION

POLAR codes, introduced by [1], have been adopted in the 5^{th} generation wireless communication standard as the control channel coding scheme for the enhanced Mobile Broadband (eMBB) service. Currently, the most dominant decoding scheme for polar codes is the cyclic redundancy check (CRC) assisted successive cancellation list (CA-SCL) algorithm [2], [3] which enables polar codes to be competitive with state-of-the-art codes. To further improve the performance of CA-SCL at high signal noise ratio (SNR), some research was conducted based on the CRC code design. Zhang *et al*. searched the optimal CRC polynomials for the standard CA-SCL decoder to eliminate the erroneous polar codewords with minimum Hamming weight (MHW) in decoding [4]. Using the same idea, [5] and [6] respectively designed the protected bits of CRC and the locations of CRC bits.

However, all the above schemes have two drawbacks. First, they cannot detect error in the intermediate decoding process.[1] Second, they cannot correct the decoding error.

To solve the first problem, [7] first proposed a multi-CRC polar codes which can also reduce the decoding delay and memory space. Meanwhile, another partitioning method which can reduce the memory requirements associated with SCL decoding was proposed by [8]. Considering these structures suffer from performance degradation, [9] proposed an optimized scheme. However, its designing complexity is high and it fails to give a universal concatenation scheme.

The second defect can be addressed by parity-check (PC) polar coding which is first proposed by [10]. After that, [11] designed the PC coding based on the polar kernel to improve the performance. Especially, [12] used a single cycle shift register (CSR) to design a PC polar coding scheme, which has been verified to provide significant performance gain.

Indeed, PC bits can be regarded as scattered CRC bits to detect error.[2] However, they cannot improve the performance while at the same time maintaining the similar error-detecting rate of the standard CA-SCL decoder. That's to say, if PC bits were used to detect error, the performance would not be improved. Otherwise, if they were used to correct error, receiver would not be able to confirm whether the decoding is correct or not. Considering the error-detecting capacity of p independent PC bits is inferior to a p length CRC [13], PC bits are mainly used to correct error. This makes some techniques that can be used in CA-SCL decoder (such as adaptive SCL decoder [14], bit-flip algorithm [15], [20], [21] and hybrid-automatic-repeat-request (HARQ) mechanism [22]) cannot be applied to the PC polar coding to repair the failed decoding attempt.

Overall, there does not exist a structure that can provide the similar decoding performance of PC polar codes while detecting error as accurately as the standard CRC-concatenated

The associate editor coordinating the review of this manuscript and approving it for publication was Soon Xin Ng.

[1] Since SC or SCL decoder is serial, the decoding delay is large especially when code length is very high. We should find a method which can detect the decoding error in time, but instead of detecting error at the end of decoding.

[2] PC bits are scattered not just in their locations but in their work mode. PC bits are actually odd even check bits and work independently of each other. On the other hand, CRC, which has a longer code length and each check bits is related, performs better on checking error.

polar codes. To deal with this issue, this paper will propose a CRC-aided PC polar coding scheme that has both the capacity of error correcting and detecting. The main contributions of this paper can be summarized as follows

1) Based on [12], we will propose an enhanced PC coding scheme which can significantly reduce the rate loss.
2) For the enhanced scheme, the suitable number of PC bits is discussed.
3) The locations of CRC and PC bits are designed.

In the proposed scheme, CRC bits are located in the middle of the information sequence and only protect their previous bits. Thus, decoding errors can be detected before the entire decoding is completed. Simulation results will verify that the proposed scheme can provide better performance and error-detecting capacity compared with the standard CRC-concatenated scheme, irrespective of the code length or the code rate.

II. PRELIMINARIES

A. POLAR CODES

Let x_i^j be the vector $[x_i,x_{i+1},\ldots,x_j]$, $i<j$. A polar code with length $N = 2^n$, $n = 1,2,\ldots,$ can be encoded by $c_1^N = u_1^K \mathbf{G}_{\mathcal{A}}$, where u_1^K and c_1^N respectively denote sequence of information bits and codeword, and $\mathbf{G}_{\mathcal{A}}$ is the generator matrix which takes rows with indices $i\in\mathcal{A}$ from matrix $G = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}^{\otimes n}$, with $\mathcal{A}$ the information set and $\otimes$ the Kronecker product. Let $|\mathcal{A}|$ be the cardinality of $\mathcal{A}$ and $\mathcal{A}^c = \{1,2,\ldots,N\}-\mathcal{A}$ be the frozen set. The code rate is $R = \frac{K}{N}$. The encoding process can also be $c_1^N = \bar{u}_1^N G$, where $\bar{u}_1^N$ is the input vector of polar codes. For $\bar{u}_1^N$, the bits with indices in $\mathcal{A}^c$ are all 0 and the bits with indices in $\mathcal{A}$ are information bits.

B. PARITY-CHECK POLAR CODING

Of all the existing PC polar coding schemes, we only consider the one in [12] for its two advantages. First,its construction is so simple and universal that it does not need to redesign and store different PC functions for different polar coding schemes. Second, it can achieve satisfactory performance gain compared with all the other existing schemes. The process of this PC coding can be summarized as follows:

1) Construct the polar code to achieve $\mathcal{A}$, where $|\mathcal{A}| = p+K$, with $p = 1,2,\ldots,N-K$, and α a parameter determined by list size of SCL decoder.

2) Determine the locations of PC bits. Let $\mathcal{A}_m$ and $\mathcal{A}_s$ be the set that satisfy

$$\mathcal{A}_m = \{i\in\mathcal{A}|w(g_n^{(i)}) = d_m\},\ \mathcal{A}_s = \{i\in\mathcal{A}|w(g_n^{(i)}) = d_s\} \quad (1)$$

where d_m and d_s are the MHW and sub-minimum Hamming weight of polar code, respectively, and $w(g_n^{(i)})$ is the weight of the i-th row in G. The positions of PC bits are selected in these two sets according to the descending order of their own reliability which is obtained by coding construction. Concretely, if $p\leq|\mathcal{A}_m|$, p indices in $\mathcal{A}_m$ are selected. If $p>|\mathcal{A}_m|$, all the indices in $\mathcal{A}_m$ should be selected and the extra $p-|\mathcal{A}_m|$ indices are chosen from set $\mathcal{A}_s$. We use $\mathcal{A}_{in}$ to denote the different set between $\mathcal{A}$ and $\mathcal{P}$, where $\mathcal{P}$ is the set of the selected locations for PC bits.

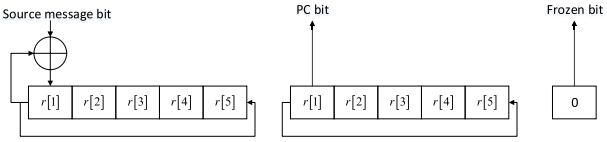


FIGURE 1. The PC Pre-coding process of the scheme in [12].

3) Implement the PC pre-coding by a 5-length CSR as shown in Fig.1. Let $r[k]_i$ be the value of the k-th register after i times of cyclic shift. We use $\bar{u}_1^N$ to denote the PC encoded vector, or equivalently the input vector of polar code. The value for each register is initialized by $r[1]_0 = r[2]_0 = \ldots = r[5]_0 = 0$, and a count parameter k is set 1. Then, one considers all the indices from 1 to N one by one. When considering any $i\in\{1,2,\ldots,N\}$, the register is first left cyclic shifted, i.e.,

$$r[1]_i = r[2]_{i-1}, r[2]_i = r[3]_{i-1}, \ldots, r[5]_i = r[1]_{i-1}. \quad (2)$$

If $i\in\mathcal{A}_{in}$, then successively set

$$\bar{u}_i = u_k,\quad r[1]_i = u_k\oplus r[1]_i,\quad k = k+1. \quad (3)$$

If $i\in\mathcal{P}$, then set $\bar{u}_i = r[1]_i$. If $i\in\mathcal{A}^c$, then set $\bar{u}_i = 0$.

Example: In the modified PC coding, when $N = 32$, $K = 16$, $\mathcal{A}_{in} = \{8, 12, 14, 15, 16, 20, 22, 23, 24, 26, 27, 28, 29, 30, 31, 32\}$, $\mathcal{P} = \{13, 18, 19, 21, 25\}$. Then there are $|\mathcal{P}| = 5$ PC functions. Concretely,

$$u_8 \oplus u_{13} = 0$$
$$u_8 \oplus u_{18} = 0$$
$$u_{14} \oplus u_{19} = 0$$
$$u_{16} \oplus u_{21} = 0$$
$$u_{15} \oplus u_{20} \oplus u_{25} = 0$$

III. CRC-AIDED PC POLAR CODING

A. MODIFIED PC CHECK

Actually, the scheme in [12] can be sightly modified by using all the frozen bits as PC bits. As shown Algorithm 1, in the modified pre-coding process, when considering the index $i\in\mathcal{A}^c$, it still has $\bar{u}_i = r[1]_i$. Considering SCL decoder can approach the ML performance with practical list size, e.g., $L = 8$, the number of MHW codewords has a great impact on the decoding results [16]. Therefore, even if all the frozen bits are used as PC bits, the index set $\mathcal{P}$ that are selected from $\mathcal{A}_m$ is still necessary for parity checking. This is because only using frozen bits to check cannot essentially reduce the number of the MHW codewords.

In this modified PC coding scheme, there are two kinds of PC bits. One is the bits at frozen indices, which are called as *frozen parity-check bits* (FPCB), and another one is the bits whose positions are selected from $\mathcal{A}_m$, i.e., the

Algorithm 1 Modified PC Pre-Coding

Input: $\mathcal{A}_{in}, u_1^K$
Output: $\bar{u}_1^N$
Initialization: $r[1]_0 = r[2]_0 = \ldots = r[5]_0$, $\bar{u}_1^N = \mathbf{0}$, $k = 0$
for $i = 1, 2, \cdots, N$ **do**
  $r[1]_i = r[2]_{i-1}, r[2]_i = r[3]_{i-1}, \ldots, r[5]_i = r[1]_{i-1}$
  if $i \in \mathcal{A}_{in}$ **then**
    $\bar{u}_i = u_k, \quad r[1]_i = u_k \oplus r[1]_i, \quad k = k+1$
  if $i \notin \mathcal{A}_{in}$ **then**
    $\bar{u}_i = r[1]_i$
Return $\bar{u}_1^N$;

bits with indices in $\mathcal{P}$, which are referred to as *information parity-check bits* (IPCB). Obviously, under the same condition, the modified PC coding should outperform the original one [12].

B. SUITABLE LENGTH OF PC CHECK

One of the main drawbacks of the PC coding is that it cannot detect error. Intuitively, we can use a CRC to solve this problem. However, too many check bits will cause rate loss to offset the benefits. Thus, the total number of both the two kinds of check bits should be limited. Under this condition, we will first tradeoff between the suitable length of these two kinds of check bit.

In the original PC coding scheme in [12], if the rate matching technique is not used, the number of PC bits is set to

$$p = \lceil \log_2 N(\alpha - |\alpha(K/N - 1/2)|^2) \rceil. \tag{4}$$

Especially, when code rate equals to 0.5 and α is fixed, the number of IPCB will achieve the maximum value $\alpha \log_2 N$. Since α is a variable parameter, [12] did not provide the closed-form solution of the most suitable number of IPCB. Note that if α is set 1, p will fluctuate around $\log_2 N$ according to the code rates. Obviously, FPCB will not cause rate loss. Thus, we only discuss the number of IPCB in the following experiment.

As shown in Fig.2-3, we give the frame error rate (FER) of the PC coding scheme by arranging p from 1 to $\log_2 N$ (with step 1) under SCL decoder. We consider the polar codes with length $N \in \{1024, 512, 256\}$ and rate $R \in \{1/2, 2/3\}$. Thus, $|\mathcal{A}|$ is set to $K+p$. When $R = 1/2$, the transmitting SNR is 2.4 dB and the list size of SCL decoder is 8. When $R = 2/3$, the transmitting SNR is 2.6 dB and the list size of SCL decoder is 16. Fig.2 and Fig.3 respectively consider the original PC polar coding scheme in [12] and the modified PC coding scheme given in Algorithm 1. In Fig.2, we can find that as p gets to smaller, the performance is getting worse. However, the opposite situation occurs in Fig.3. When p gets to smaller, the performance is getting better. This is mainly because in the modified scheme, the FPCB can also

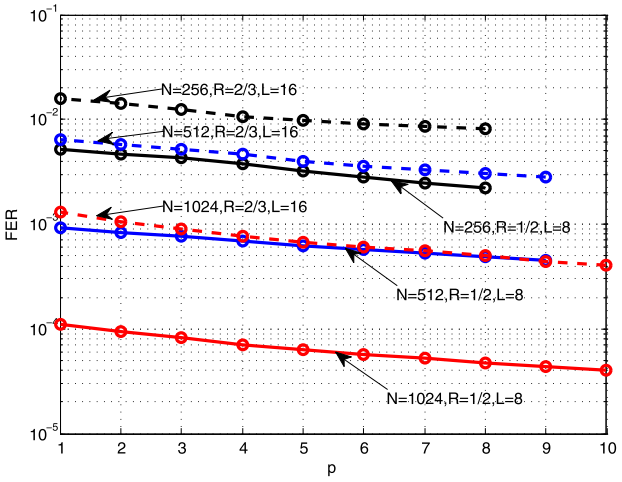


FIGURE 2. FER of the original PC scheme in [12] with p arranging from 1 to $\log_2 N$.

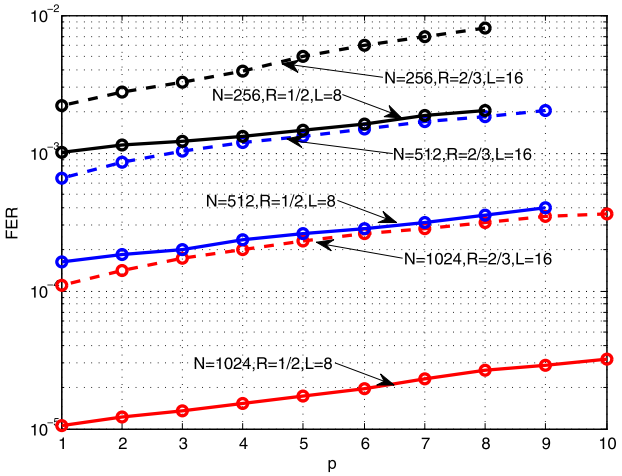


FIGURE 3. FER of the modified PC scheme in algorithm 1 with p arranging from 1 to $\log_2 N$.

benefit the decoding performance. On the other hand, in the original PC coding scheme, the error correcting only relies on IPCB, and thus the number of IPCB is more important for performance compared with the modified scheme. That's to say, using the original PC coding, the number of IPCB cannot be further reduced.

To make a more fair comparison, in Fig.4-5, the number of information bits is changed with p and we only adopt the modified PC coding. The code lengths of the considered polar codes will be $N \in \{256, 512, 1024\}$. In Fig.4, the number of information bits is set as

$$K = \frac{N}{2} + \log_2 N - p, \quad p = 1, 2, \ldots, \log_2 N. \tag{5}$$

That's to say, $|\mathcal{A}|$ is fixed to $\frac{N}{2} + \log_2 N$. The transmitting SNR is 2.4 dB. We can observe that the performance of the PC polar code with $p = 1$ is very close to the one with $p = \log_2 N$.

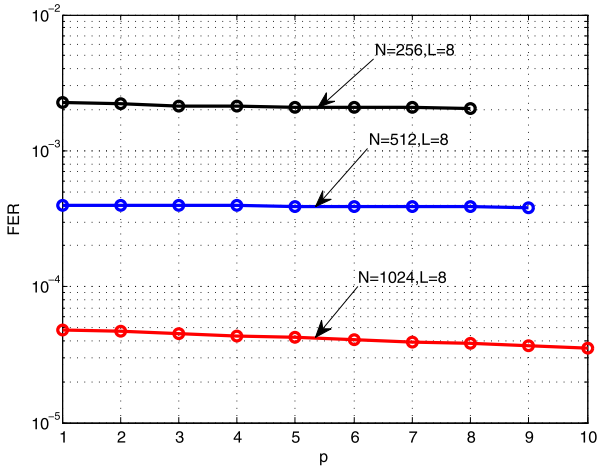


FIGURE 4. FER of the modified PC scheme in algorithm 1 with p arranging from 1 to $\log_2 N$, where $|\mathcal{A}|$ is fixed to $\frac{N}{2}+\log_2 N$.

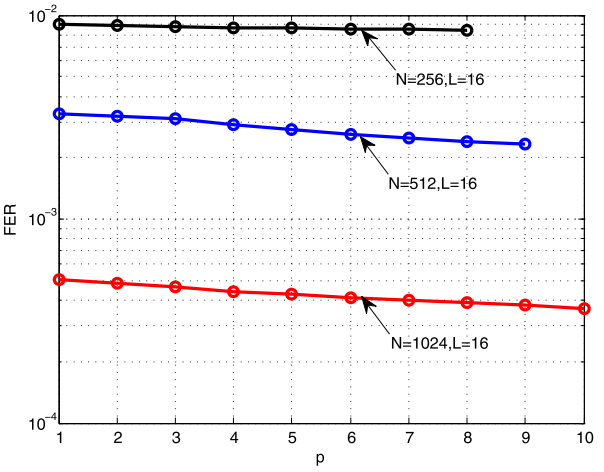


FIGURE 5. FER of the modified PC scheme in algorithm 1 with p arranging from 1 to $\log_2 N$, where $|\mathcal{A}|$ is fixed to $\frac{2N}{3}+\lceil\frac{35\log_2 N}{36}\rceil$.

In Fig.5, the code rate is higher and set as

$$K=\frac{2N}{3}+\lceil\frac{35\log_2 N}{36}\rceil-p,\ p=1,2,\ldots,\lceil\frac{35\log_2 N}{36}\rceil. \quad (6)$$

The transmitting SNR is 2.6 dB. We can observe that the performance gap between the scheme with $p=\lceil\frac{35\log_2 N}{36}\rceil$ and the one with $p=1$ is still narrow.

In Fig.6, we consider the modified PC scheme with

$$K=\frac{3N}{4}+\left\lceil\frac{15\log_2 N}{16}\right\rceil-p,\ p=1,2,\ldots,\left\lceil\frac{15\log_2 N}{16}\right\rceil. \quad (7)$$

The transmitting SNR is 3.0 dB. Although the curves are still fairly flat, they are a little steeper than those in Fig.4 and 5.

From Fig.4-6, we can observe that when the code rate is getting higher, the number of IPCB has a greater impact on the error correcting performance.

Thus, for the modified PC coding scheme, we can draw a conclusion that just one IPCB is enough to obtain satisfactory performance gain if the code rate is not very high

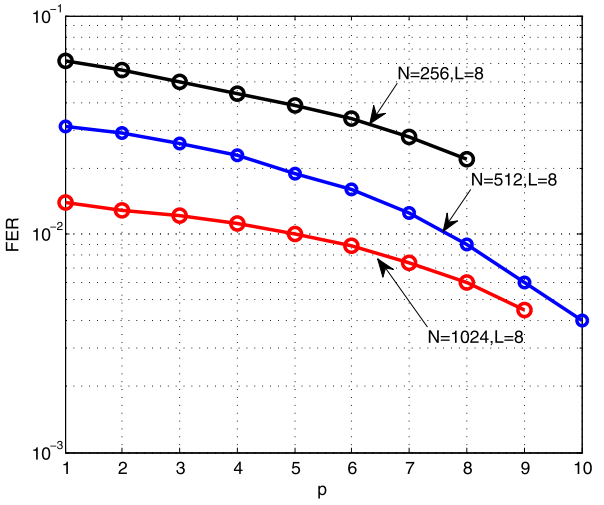


FIGURE 6. FER of the modified PC scheme in algorithm 1 with p arranging from 1 to $\log_2 N$, where $|\mathcal{A}|$ is fixed to $|\mathcal{A}|=\frac{3N}{4}+\left\lceil\frac{15\log_2 N}{16}\right\rceil$.

(i.e., $\leq\frac{3}{4}$). Under this condition, if we use a CRC-aided PC code, the number of CRC bits can be selected as

$$p_{crc}=\lceil\log_2 N(\alpha-|\alpha(K/N-1/2)|^2)\rceil-1 \quad (8)$$

C. THE LOCATION OF CHECK BITS

In this part, we will design the locations of the two kinds of check bit. Based on the discussion in the previous part, the number of IPCB can be fixed to 1. We want to maximize the path-metric penalty for the error path. Thus, the location of IPCB should be the maximum index in $\mathcal{A}_m$.

As for CRC bits, they are expected to detect the decoding error as soon as possible. Thus, we need to focus on the most likely location for the first error bit. In other words, the positions where the first error bit rarely appears does not need to be protected by CRC. Recalling that at high SNR the performance of SCL decoder approaches the ML bound, [6] verified that the main error patterns of the SCL decoder are the input vector $\bar{u}_1^N$ whose corresponding codeword has the MHW. Meanwhile, [4] also proved that such the input vector $\bar{u}_1^N$ should satisfy

$$\bar{u}_1^{i-1}=\mathbf{0},\ \bar{u}_i=1,\ \bar{u}_{i+1}^N=\{0,1\}^{N-i} \quad (9)$$

with $i\in\mathcal{A}_m$. This implies that at high SNR if a decoding attempt fails, the first error of the entire frame rarely occurs after the maximum index in $\mathcal{A}_m$.

On the other hand, at low SNR, most of the first error bits also tend to appear in the first few positions. This is because at low SNR the performance mainly depends on the reliability of the bit-channel and the first several ones are less reliable. That's to say, for any SNR, CRC can obtain a satisfactory detecting effect by only protecting the bits with indices before the maximum index in $\mathcal{A}_m$.

According to the above analysis, the locations of CRC bits can be determined. If $\mathcal{A}$ is arranged in ascending order,

we have

$$\mathcal{A} = \{\mathcal{A}(1), \mathcal{A}(2), \ldots, \mathcal{A}(|\mathcal{A}|)\}. \quad (10)$$

We denote the maximum index of $\mathcal{A}_m$ as $\mathcal{A}(M)$, with $M = 1, 2, \ldots, |\mathcal{A}|$. Then the set of the positions of CRC is

$$\mathcal{A}_{crc} = \{\mathcal{A}(M - p_{crc}), \mathcal{A}(M - p_{crc} + 1), \ldots, \mathcal{A}(M-1)\}. \quad (11)$$

Note that $\mathcal{A}(M)$ is also the location of the PC bit. We have

$$\mathcal{A}_{in} = \mathcal{A} - \{\mathcal{A}(M)\}. \quad (12)$$

D. CRC-AIDED PC CODING SCHEME

Then, the proposed CRC-aided PC coding scheme can be summarized in Algorithm 2. We can find that the source information vector u_1^K is first encoded by CRC code. Then the obtained vector $\widetilde{u}_1^{K+p_{crc}}$ is PC encoded into $\bar{u}_1^N$.

Algorithm 2 CRC-Aided PC Pre-Coding Scheme

Input: $\mathcal{A}_{in}, M, u_1^K$
Output: $\bar{u}_1^N$
CRC encoding:
Use a p_{crc}-length CRC code to encode the source vector u_1^K and the CRC bits are written as $u_1^{crc}, u_2^{crc}, \ldots, u_{p_{crc}}^{crc}$.
The encoded vector $\widetilde{u}_1^{K+p_{crc}}$ is arrayed as:
$\widetilde{u}_1^{K+p_{crc}} = [u_1, u_2, \ldots, u_{M-1-p_{crc}}, u_1^{crc}, u_2^{crc}, \ldots, u_{p_{crc}}^{crc}, u_{M-p_{crc}}, u_{M+1-p_{crc}}, \ldots, u_K]$
Initialization: $r[1]_0 = r[2]_0 = \ldots = r[5]_0$, $\bar{u}_1^N = \mathbf{0}$, $k = 0$
for $i = 1, 2, \cdots, N$ **do**
  $r[1]_i = r[2]_{i-1}, r[2]_i = r[3]_{i-1}, \ldots, r[5]_i = r[1]_{i-1}$
  if $i \in \mathcal{A}_{in}$ **then**
    $\bar{u}_i = \widetilde{u}_k, \quad r[1]_i = \widetilde{u}_k \oplus r[1]_i, \quad k = k+1$
  if $i \notin \mathcal{A}_{in}$ **then**
    $\bar{u}_i = r[1]_i$
Return $\bar{u}_1^N$;

In Fig.7, we compare the different CRC-concatenated polar coding schemes and the modified PC coding scheme. Fig.7(a) shows the standard scheme where the CRC bits are appended at the tail of the source information vector and protect all the message bits. Fig.7(b) depicts the scheme in [5] where the CRC bits are appended at the tail of the source vector while protect the bits with indices in $\mathcal{A}_m$. Then, Fig.7(c) presents the scheme in [6] where the CRC bits are located at the front of the source vector and protect the bits with indices in $\mathcal{A}_m$. Besides, Fig.7(d) depicts the structure in [7] where CRC bits are partitioned into serval parts and each part only protects its previous information bits. Fig.7(e) depicts the modified PC coding scheme given in Algorithm 1 where only the PC bits are adopted. Finally, we give the pstructure of the proposed CRC-aided PC polar coding in Fig.7(f) and we can observe that the CRC bits are located in the middle of the source vector and only protect their previous bits. From Fig.7, compared

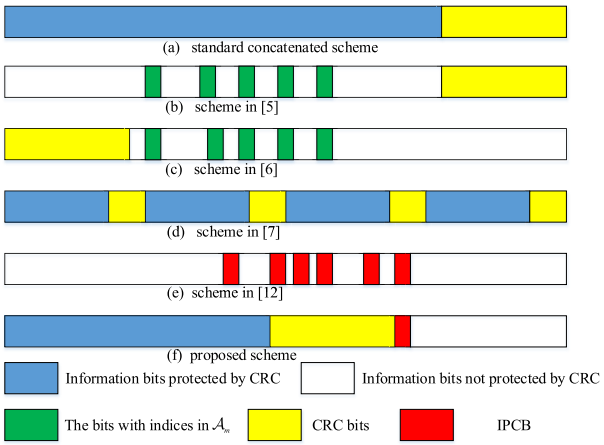


FIGURE 7. CRC encoded vectors of different structures.

with all the existing CRC-concatenated polar codes, the proposed scheme has the following advantages:

1) The CRC bits are located in the middle of the information vector so that the failed decoding attempt can be detected before the completion of entire decoding process.

2) The detecting efficiency is better since the protected bits are only part of the entire information vector and would contain the first erroneous decoding bit with extremely high probability.

3) The PC bit can improve the performance.

4) For the case that $|\mathcal{A}_m|$ is very small compared with $|\mathcal{A}|$, the schemes that CRC only protects the bits with indices in $\mathcal{A}_m$ would be incapable and suffer from extremely high undetecting rate. However, the proposed scheme is suitable for the polar codes with all the possible $|\mathcal{A}_m|$.

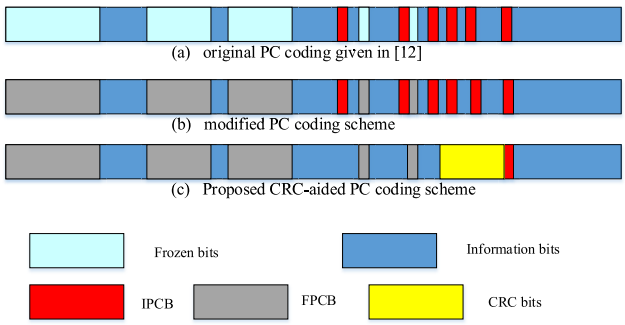


FIGURE 8. Different PC coding schemes.

In Fig.8, we also give different PC coding schemes. Fig.8(a) depicts the original PC coding scheme given in [12] where the IPCBs are located at the indices in $\mathcal{P}$ and the information bits are located at the indices in $\mathcal{A}_{in}$. Meanwhile, Fig.8(b) shows the modified PC coding scheme which is similar with the original PC coding scheme. The only difference is that the modified PC coding scheme takes the frozen bits as the FPCB. In Fig.8(c) we give the proposed CRC-aided PC coding scheme where only one IPCB is adopted and the CRC

bits are placed in front of the IPCB. Note that in Fig.7 we did not consider the frozen bits in each schemes.

Compared with the PC coding, the superiority of the proposed scheme is also notable. The proposed scheme can detect the failed decoding attempt so that the re-decoding mechanisms can be applied.

E. DECODING OF CRC-AIDED PC POLAR CODES

Let $\mathsf{L}_1^N = [\mathsf{L}_1, \mathsf{L}_2, \ldots, \mathsf{L}_N]$ be the log-likelihood ratio (LLR) vector received from channel. Besides, we use $\hat{u}_i[l]$ to denote the i-th decoding bit at the l-th path when just decoding u_i, with $i = 1, 2, \ldots, N$ and $l = 1, 2, \ldots, L$. Finally, $\mathsf{PM}_i[l]$ denotes the path metric (PM) of the l-th path when just decoding u_i. From [18], $\mathsf{PM}_i[l]$ can be expressed as

$$\mathsf{PM}_i[l] = \begin{cases} \mathsf{PM}_{i-1}[l], & \text{if } \hat{u}_i[l] = \frac{1}{2}[1 - \text{sign}(\mathsf{L}_i[l])] \\ \mathsf{PM}_{i-1}[l] + \left|\mathsf{L}_i[l]\right|, & \text{otherwise} \end{cases} \tag{13}$$

where

$$\mathsf{L}_i[l] = \ln\left(\frac{\Pr\left(\mathsf{L}_i^N, \hat{u}_1^{i-1}[l] \,|\hat{u}_i[l] = 0\right)}{\Pr\left(\mathsf{L}_i^N, \hat{u}_1^{i-1}[l] \,|\hat{u}_i[l] = 1\right)}\right) \tag{14}$$

Obviously, the most reliable path is the one with minimum PM. The decoding process of the proposed scheme is summarized in Algorithm 3.

Algorithm 3 Decoding of CRC-Aided PC Polar Codes

Input: L_1^N
Output: decoded vector $\hat{u}_1^N$
for $i = 1, 2, \cdots, N$ **do**
  if $i \in \mathcal{A}_{in}$ **then**
    $\hat{u}_i[l] \leftarrow \text{SCL}(\mathsf{L}_1^N, \hat{u}_1^{i-1}[l])$
    $\mathsf{PM}_i[l] \leftarrow$ Update PM by Eq.(12)
    if $i = \mathcal{A}(M-1)$ **then**
      for $l = 1, 2, \ldots, L$ **do**
        if $CRC(\hat{u}_1^{\mathcal{A}(M-1)}[l])$ *fails* **then**
          $\mathsf{PM}_i[l] \leftarrow +\infty$
    if *All the paths cannot pass CRC* **then**
      Terminate decoding and declare failure.
  if $i \notin \mathcal{A}_{in}$ **then**
    $\hat{u}_i[l] \leftarrow \text{PCFunc}(\hat{u}_1^{i-1}[l])$
    $\mathsf{PM}_i[l] \leftarrow$ Update PM by Eq.(12)
  if $i = N$ **then**
    The path with the minimum PM is output as $\hat{u}_1^N$
Return $\hat{u}_1^N$;

In Algorithm 3, we can observe that for any $i \in \mathcal{A}_{in}$, SCL decoder successively decodes u_i. Especially, if $i = \mathcal{A}(M-1)$, then all the current L decoding trajectories $\hat{u}_1^{\mathcal{A}(M-1)}$ should be testified by CRC. For the trajectories that can not pass CRC, their PM is set positive infinity. This operation equivalently means removing these trajectories from the candidate list. For any $i \notin \mathcal{A}_{in}$, $\hat{u}_1^i[l]$ is obtained by the PC function established by Algorithm 2. If $i = N$, then one should select the path with minimum PM as the output of decoder.

One of the main advantages of such decoder is that CRC can be used not only to detect error but also to correct error. All the trajectories that can be seen as the inference of the correct path, i.e., those trajectories that cannot pass the CRC, are eliminated during intermediate decoding.

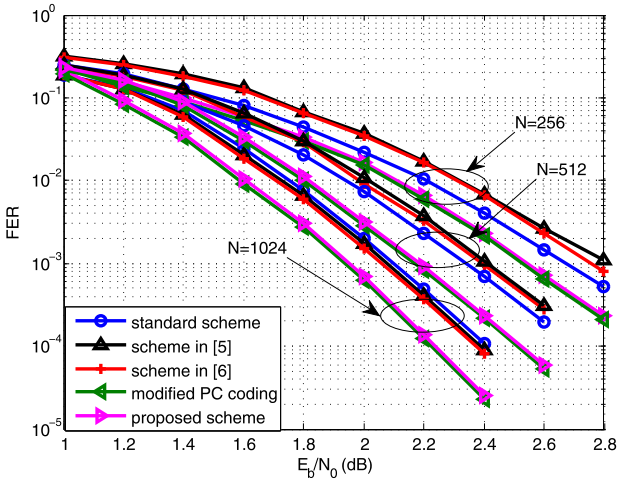


FIGURE 9. Performance comparison of the proposed structure with the scheme in [5], the scheme in [6], the standard CRC-concatenated polar codes and the modified PC polar coding in algorithm 1.

IV. SIMULATION RESULTS

A. PERFORMANCE COMPARISON

Fig.9 compares the FER of the proposed structure with the scheme in [5], the scheme in [6], the standard CRC-concatenated polar code and the modified PC polar coding in Algorithm 1. For all the schemes, the sum of the number of check bits is determined by Eq.(4). The code lengths of polar code are set to 256, 512 and 1024. The code rate is fixed to 0.5 and the CRC polynomials are selected from [17]. Note that under polarization weight (PW) [19] coding construction, when code lengths are 256, 512 and 1024, the corresponding $|\mathcal{A}_m|$ are 4, 1, 25, respectively. The list size of SCL decoder is 8 and $\alpha = 1$. We can find that when $N = 1024$, the scheme of [5] and [6] provide similar FER performance compared with the standard CRC-concatenated polar codes. When $N = 256$ or $N = 512$, the performance of these two schemes is inferior to the standard one. This means when $|\mathcal{A}_m|$ is too small compared with $|\mathcal{A}|$, these two schemes are incapable. Meanwhile, the performance of the proposed scheme is very close to the that of the modified PC polar coding scheme, irrespective of the code length.

In Fig.10, we give the performance curves of the proposed scheme, the original PC coding [12], the modified PC coding and the standard CRC-concatenated scheme. The code

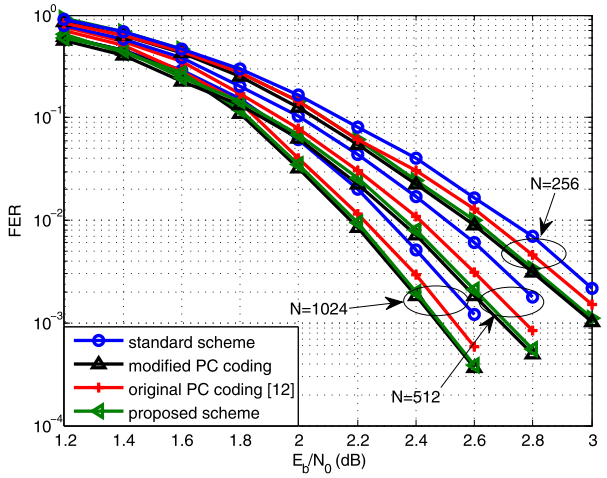


FIGURE 10. Performance comparison of the proposed structure with the original PC coding [12], the modified PC coding in algorithm 1 and the standard CRC-concatenated polar codes.

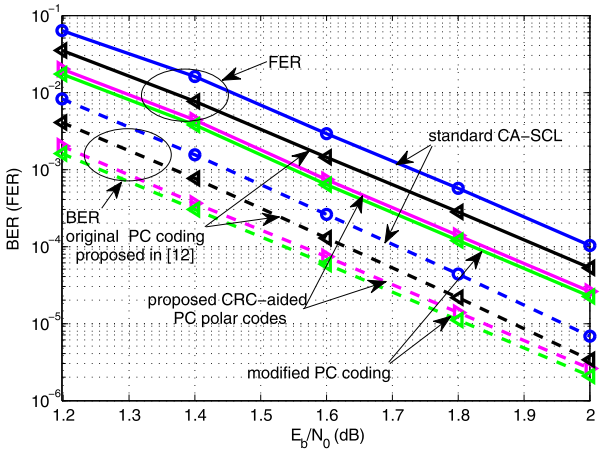


FIGURE 12. Performance comparison of the proposed structure with the original PC coding [12], the modified PC coding in algorithm 1 and the standard CRC-concatenated polar codes. The length of polar codes is 4096.

lengths of the considered polar codes are $N \in \{256, 512, 1024\}$ and the code rate is fixed to $2/3$. The total number of the check bits is fixed to $\log_2 N$. The list size is 16. We can find that the modified PC coding can provide the best performance among all the schemes, irrespective of code length. The performance of the proposed scheme is very close to that of the modified PC coding. Compared with the standard CRC-concatenated scheme, all the PC coding schemes can give better performance.

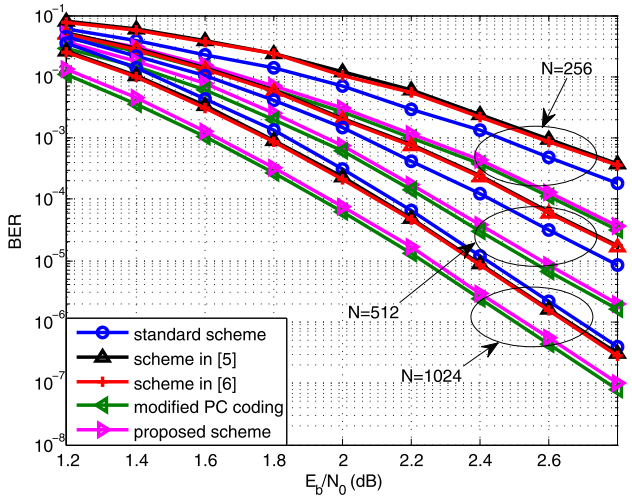


FIGURE 11. Performance comparison of the proposed structure with the scheme in [5], the scheme in [6], the standard CRC-concatenated polar codes and the modified PC polar coding in algorithm 1.

In Fig.11, we give the bit error rate (BER) of the same schemes considered in Fig.9. The parameters of simulations are also identical with those used in Fig.9. We can find that the BER of the modified PC coding scheme is also the most outstanding. The BER of the proposed CRC-aided PC polar coding is very close to that of the modified PC coding.

In Fig.12, the length of the adopted polar codes is changed to 4096. The code rate is 0.5 and the list size of SCL decoder is 8. For each schemes, the total number of the check bits is based on Eq.(4). We can find that the proposed scheme also outperforms the standard CA-SCL decoder, irrespective of the FER or BER. This further shows the universality of the proposed scheme.

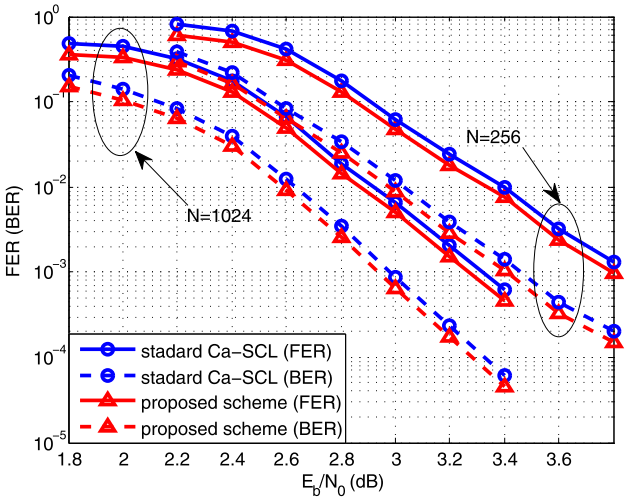


FIGURE 13. Performance comparison of the proposed structure with the standard CRC-concatenated polar codes. The code rate is $\frac{3}{4}$.

To determine the border-line of the practicality of the proposed method, we give Fig.13 where the code rate of polar codes is $\frac{3}{4}$. The code length is $N = \{256, 1024\}$. The list size of SCL decoder is 8. We can find that the proposed scheme can not provide that impressive performance gains as it does at low code rates. This is because at high code rate the number of FPCB, which can correct the decoding error, is reduced.

Finally, we also consider the effect of the decoding scheme in the presence of the coded-modulation. In Fig.14-15, we

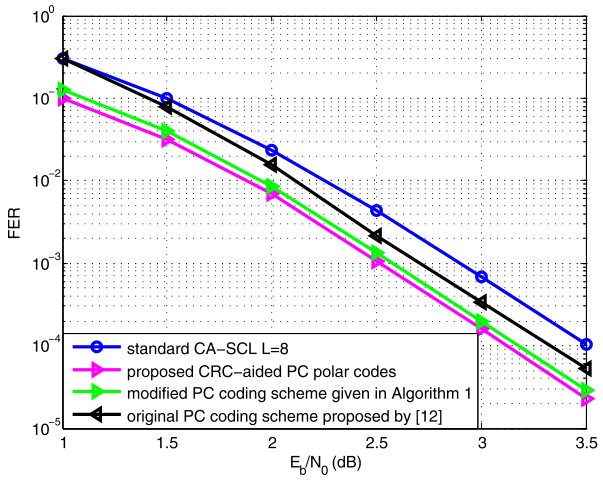


FIGURE 14. Performance comparison of the proposed structure with the original PC coding [12], the modified PC coding in Algorithm 1 and the standard CRC-concatenated polar codes. The codeword is 16-QAM modulated.

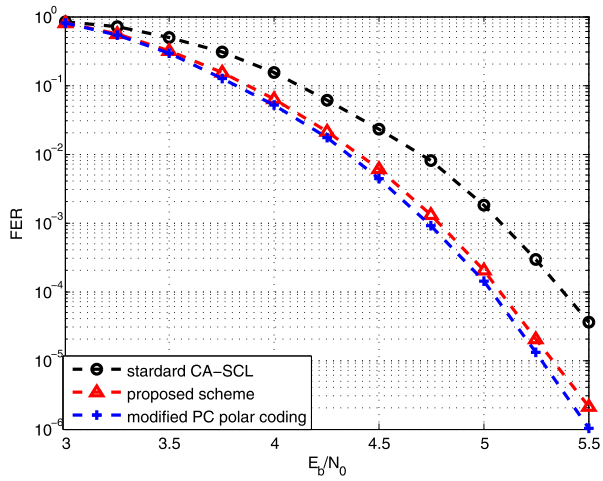


FIGURE 15. Performance comparison of the proposed structure with the modified PC coding in algorithm 1 and the standard CRC-concatenated polar codes. The codeword is 64-QAM modulated.

give the performance curves associated with high order modulation, where gray mapping is adopted. In Fig.14 we adopt 16-QAM and in Fig.15 we use 64-QAM. In Fig.14, the list size of SCL decoder is fixed to 8. The code rate is 0.5 and the code length is 1024. In Fig.15, the list size of SCL decoder is 32. The code rate is $\frac{1}{3}$ and the code length is 1536. The puncturing method proposed in [24] is adopted. For each scheme, the number of check bits is fixed to 8. This means that the information set of all the three schemes are identical. Thus, their puncturing patterns are also the same. This ensures the fairness of the comparison. We can find that when high order modulated, the proposed scheme can also provide impressive performance gain compared with the standard CA-SCL decoder. Similar to the case of BPSK, the modified PC coding provides the best performance among all the considered schemes.

TABLE 1. The undetecting rate of five concatenation schemes with standard CRC polynomial.

$(N,R,E_b/N_0)$	MHW	$\lvert\mathcal{A}_m\rvert$	Scheme	P_{ud} (%)
(1024, 1/2, 1 dB)	16	25	Standard	14.20
			[5]	46.93
			[6]	41.27
			[7]	21.12
			Proposed	10.03
(1024, 1/2, 2 dB)	16	25	Standard	6.30
			[5]	4.70
			[6]	5.00
			[7]	10.07
			Proposed	5.30
(1024, 2/3, 1 dB)	8	7	Standard	14.27
			[5]	56.43
			[6]	52.43
			[7]	24.23
			Proposed	12.90
(1024, 2/3, 2 dB)	8	7	Standard	9.50
			[5]	13.17
			[6]	11.93
			[7]	12.50
			Proposed	6.10

B. DETECTING EFFICIENCY COMPARISON

In Table.1, we give the undetecting rate (P_{ud}) of all the CRC-concatenated polar codes depicted in Fig.7. For the PC polar codes, we can regard its undetecting rate as 100% since all the check bits are used to correct error. The length of the considered polar code is 1024. The number of the CRC bits is determined by Eq.(4). The polynomial of CRC is selected based on [17]. The multi-CRC scheme [7] partitions the information vector into 2 parts for protecting. The list size of SCL decoder is 8 and α is set 1. The results in Table.1 are derived based on a statistics for 3000 error decoding frames.

Using these parameters, we construct polar codes by PW algorithm [19]. Note that when the code rate is 1/2, $|\mathcal{A}_m|$ is 25. Meanwhile, when the code rate is changed to 2/3, $|\mathcal{A}_m|$ will be 7. From Table.1 one can find that at low SNR (1 dB), the undetecting rate of the schemes in [5] and [6] is higher than 40%. That of the multi-CRC scheme is higher than 20%. However, for both standard and the proposed scheme, their undetecting rates are lower than 15%. At high SNR (2 dB), there is a remarkable improvement for the schemes in [5] and [6]. However, the proposed scheme is still the most outstanding one. This is because compared with the standard CRC-concatenated scheme, the protected bits of the proposed scheme is less and the detecting efficiency is higher. The schemes in [5] and [6] only consider the polar codewords with MHW, and thus they will be incapable when SNR is low or the $|\mathcal{A}_m|$ is small. For the multi-CRC scheme, when the sum number of the CRC bits is fixed, the detecting effect of the partitioned CRC should be inferior to a single entire one.

We can also observe that when $|\mathcal{A}_m|$ is small compared with $|\mathcal{A}|$, the undetecting rate of schemes in [6] and [5] will be high. However, the proposed scheme can provide satisfactory undetecting rate regardless of $|\mathcal{A}_m|$.

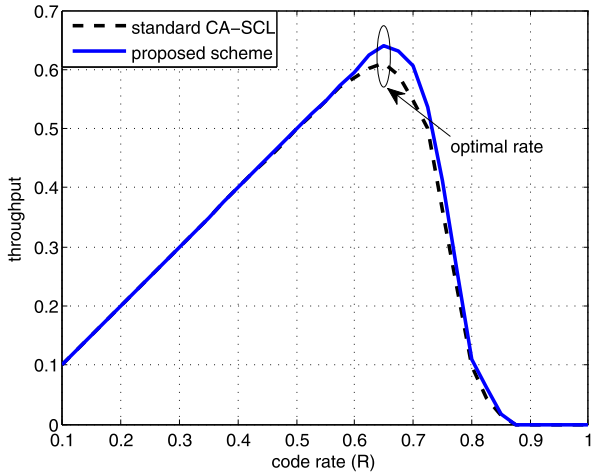


FIGURE 16. Throughput vs. the code rate with $N = 1024$ when $E_b/N_0 = 2$ *dB* for BPSK with standard CA-SCL and the proposed scheme.

Overall, the proposed scheme can provide the best detecting effect, irrespective of the code length, code rate or $|\mathcal{A}_m|$. This reflects the universality of the proposed scheme.

C. THROUGHPUT OF THE PROPOSED SCHEME

When using HARQ or adaptive SCLD technique, throughput is an improtant metric to design coding scheme, where throughput of a decoder can be defined as [23]

$$\eta = R(1 - P_{fer}) \tag{15}$$

where P_{fer} is the FER of the decoder. In Fig.16, we give the throughput of the proposed coding scheme and CA-SCL decoder. The SNR (i.e., E_b/N_0) is 2 dB. The codelength is 1024 and the number of check bits is 8. The list size is 8. We can find that if we take throughput as the metric, the optimal code rate for CA-SCL decoder and the proposed coding scheme are both 0.65. At higher code rate, the proposed scheme can provide better throughput compared with the CA-SCLD.

V. CONCLUSION

In this paper, we proposed a structure of CRC-aided PC polar coding. We designed the number of the two types of the check bits. Moreover, the locations of the check bits are jointly designed. The proposed scheme can terminate a failed decoding attempt early and offer an impressive performance gain compared with the standard CRC-concatenated polar codes with a better error-detecting effect.

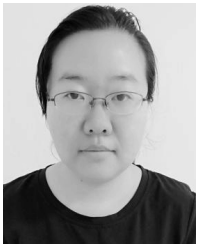
REFERENCES

[1] E. Arıkan, "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. Niu and K. Chen, "CRC-aided decoding of polar codes," *IEEE Commun. Lett.*, vol. 16, no. 10, pp. 1668–1671, Oct. 2012.
[4] 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, Jun. 2017.
[5] Z. Qingshuang, L. Aijun, and P. Xiaofei, "Efficient CRC concatenation scheme for polar codes," *Electron. Lett.*, vol. 53, no. 13, pp. 860–862, Jun. 2017.
[6] F. Cheng, A. Liu, Y. Zhang, and J. Ren, "CRC location design for polar codes," *IEEE Commun. Lett.*, vol. 22, no. 11, pp. 2202–2205, Nov. 2018.
[7] J. Guo, Z. Shi, Z. Liu, Z. Zhang, and Q. Liu, "Multi-CRC polar codes and their applications," *IEEE Commun. Lett.*, vol. 20, no. 2, pp. 212–215, Feb. 2016.
[8] S. Hashemi, A. Balatsoukas-Stimming, P. Giard, C. Thibeault, and W. J. Gross, "Partitioned successive-cancellation list decoding of polar codes," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Mar. 2016, pp. 957–960.
[9] S. A. Hashemi, M. Mondelli, S. H. Hassani, C. Condo, R. L. Urbanke, and W. J. Gross, "Decoder partitioning: Towards practical list decoding of polar codes," *IEEE Trans. Commun.*, vol. 66, no. 9, pp. 3749–3759, Sep. 2018.
[10] T. Wang, D. Qu, and T. Jiang, "Parity-check-concatenated polar codes," *IEEE Commun. Lett.*, vol. 20, no. 12, pp. 2342–2345, Dec. 2016.
[11] P. Trifonov and V. Miloslavskaya, "Polar subcodes," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 2, pp. 254–266, Feb. 2016.
[12] H. Zhang, R. Li, J. Wang, S. Dai, G. Zhang, Y. Chen, H. Luo, and J. Wang, "Parity-check polar coding for 5G and beyond," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–7.
[13] J. E. Mazo and B. R. Saltzberg, "Error-burst detection with tandem CRCs," *IEEE Trans. Commun.*, vol. 39, no. 8, pp. 1175–1178, Aug. 1991.
[14] 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, Dec. 2012.
[15] O. Afisiadis, A. Balatsoukas-Stimming, and A. Burg, "A low-complexity improved successive cancellation decoder for polar codes," in *Proc. 48th Asilomar Conf. Signals, Syst. Comput.*, Nov. 2014, pp. 2116–2120.
[16] X. Ma, J. Liu, and B. Bai, "New techniques for upper-bounding the ML decoding performance of binary linear codes," *IEEE Trans. Commun.*, vol. 61, no. 3, pp. 842–851, Mar. 2013.
[17] P. Koopman and T. Chakravarty, "Cyclic redundancy code (CRC) polynomial selection for embedded networks," in *Proc. IEEE Int. Conf. AINA*, Jun./Jul. 2004, pp. 145–154.
[18] A. Balatsoukas-Stimming, M. B. Parizi, and A. Burg, "LLR-based successive cancellation list decoding of polar codes," *IEEE Trans. Signal Process.*, vol. 63, no. 19, pp. 5165–5179, Oct. 2015.
[19] G. He, J.-C. Belfiore, I. Land, G. Yang, X. Liu, Y. Chen, R. Li, J. Wang, Y. Ge, R. Zhang, and W. Tong, "Beta-expansion: A theoretical framework for fast and recursive construction of polar codes," in *Proc. IEEE GLOBECOM*, Dec. 2017, pp. 1–6.
[20] F. Cheng, A. Liu, Y. Zhang, and J. Ren, "Bit-flip algorithm for successive cancellation list decoder of polar codes," *IEEE Access*, vol. 7, pp. 58346–58352, 2019.
[21] Y. Yongrun, P. Zhiwen, L. Nan, and Y. Xiaohu, "Successive cancellation list bit-flip decoder for polar codes," in *Proc. 10th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Oct. 2018, pp. 1–6.
[22] K. Chen, K. Niu, and J. Lin, "A hybrid ARQ scheme based on polar codes," *IEEE Commun. Lett.*, vol. 17, no. 10, pp. 1996–1999, Oct. 2013.
[23] H. Khoshnevis, I. Marsland, and H. Yanikomeroglu, "Throughput-based design for polar-coded modulation," *IEEE Trans. Commun.*, vol. 67, no. 3, pp. 1770–1782, Mar. 2019.
[24] K. Niu, K. Chen, and J.-R. Lin, "Beyond turbo codes: Rate-compatible punctured polar codes," in *Proc. IEEE ICC*, Jun. 2013, pp. 3423–3427.

FENGYI CHENG was born in China, in 1988. He received the B.S. degree from the Nanjing University of Posts and Telecommunication NUPT, in 2011, and the M.S. degree in communication engineering and information systems from the College of Communications Engineering, Army Engineering University of PLA, Nanjing, China, in 2017, where he is currently pursuing the Ph.D. degree with the Institution of Communications Engineering. His research interests include satellite communication, coded modulation techniques, channel coding, and information theory.

AIJUN LIU (M'15) received the B.S. degree in microwave communications and the M.S. and Ph.D. degrees in communications engineering and information systems from the College of Communications Engineering, Army Engineering University of PLA, Nanjing, China, in 1990, 1994, and 1997, respectively. He is currently a Full Professor with the Army Engineering University of PLA. His research interests include satellite communication system theory, signal processing, space heterogeneous networks, channel coding, and information theory.

JING REN was born in China, in 1992. She received the B.S. degree from the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications (BUPT), in 2014, and the M.S. degree in communication and information systems from the Army Engineering University of PLA, in 2017. She is currently pursuing the Ph.D. degree with the Institution of Communications Engineering, Army Engineering University of PLA. Her research interests include cognitive radios, sensor networks, and energy harvesting.



KAI FENG was born in China, in 1983. She received the B.S. degree from the Harbin Institute of Technology (HIT), in 2006, and the M.S. degree from the University of Bordeaux, France, in 2010. She is currently a Lecturer with the Army Engineering University of PLA. Her research interests include engineering graphics, production information, automation, and signal processing.