Partial Product-LDPC Codes Without Rate Loss

Qianfan Wang*, Suihua Cai*, Fanhui Meng†, Li Chen‡, and Xiao Ma*

*School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China
†School of Systems Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China
‡School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou 510006, China
Email:{wangqf6,mengfh3}@mail2.sysu.edu.cn, {caish23,chenli55,maxiao}@mail.sysu.edu.cn

Abstract—In this paper, we present a new construction of product codes, where all rows of the information array are encoded using the low-density parity-check (LDPC) codes while only partial columns of the information array are encoded using the algebraic codes. Also distinguished from the conventional product codes, the actual code rates of the presented partial product codes are the same as those of the row component LDPC codes, without any rate loss. This is achieved by using the free-ride coding to transmit the additional column checks along with the LDPC codewords. For decoding, an iterative decoding algorithm, which is scheduled as free-ride decoding, row decoding and column decoding, is proposed. At the column decoding stage, the messages associated with those successfully decoded columns are hardened and exploited to improve the free-ride decoding and the row decoding. By assuming a binary symmetric channel model after row decoding, we present an approximated bound on the word error rate (WER) of the doubly-protected (by both the row code and the column code) information bits. To predict the performance of the proposed codes, a genie aided (GA) lower bound derived from the original LDPC coded system with partial known bits is presented. These bounds and the simulation results reveal that the presented product-LDPC codes can have a significant performance improvement over the row component LDPC codes (about one dB at the WER of 10^{-5}). They also reveal that the presented product-LDPC codes can have a flexible trade off between the error-floor and the waterfall performances, by using different row codes and column codes.

Index Terms—Free-ride codes, LDPC codes, product codes.

I. Introduction

Conventional product codes, which were first proposed by Elias in [1], are constructed based on two component block codes. A codeword of the conventional product codes is usually represented by a rectangular coded array, where each row is a codeword of the row code and each column is a codeword of the column code. Evidently, compared to the two component codes, the code length (coded array size) is lengthened and the code rate is reduced, yielding a potential performance improvement. By introducing a turbo-like iterative decoding algorithm, product codes become popular, which are often referred to as block turbo codes (BTCs) [2] or turbo product codes (TPCs) [3]. Due to the inherent parallel structures of their encoding and decoding algorithms, the product codes have been adopted in various communication standards, including the IEEE 802.16 [4], the IEEE 802.20 [5] and the IEEE-1901 [6].

In this paper, we present a new class of product codes, where the whole information array is encoded row-by-row using the low-density parity-check (LDPC) codes but only partial information array is encoded column-by-column using

the high rate algebraic codes. More distinguishingly, the additional column check bits are transmitted implicitly rather than explicitly using the free-ride coding, which is an approach to transmit several extra bits over the existing LDPC coded link but without any bandwidth expansion or transmission power increase [7], [8]. Thanks to the free-ride coding, the proposed partial product-LDPC codes achieve the same code rate as the row component LDPC codes. As an example of applying the free-ride coding to code construction, the main differences between this work and [9], [10] lie in the encoding, decoding and performance analysis. For the encoding, we discuss in more detail of the impact of row codes and column codes on the performance. For the decoding, we present an iterative decoding algorithm, where within one iteration, the free-ride decoding, row decoding and column decoding are performed sequentially. At the column decoding stage, the messages associated with those successfully decoded columns are hardened and exploited for next iteration. For the performance analysis, we present a genie aided (GA) lower bound by assuming both the free-ride codes and the column codes are correctly decoded. We also present an approximated bound on word error rate (WER) of the doubly-protected (by both the row codes and column codes) information. These bounds and simulation results reveal that the proposed partial product-LDPC codes can reduce the WER of the row component LDPC code from 10^{-3} to 10^{-6} at the signal-to-noise ratio (SNR) around 2 dB. They also reveal that the doubly-protected information can achieve a WER of 10^{-15} at the SNR around 2.5 dB.

II. Rate-Lossless Partial Product-LDPC Codes

A. Free-Ride Coding

Let $\mathbb{F}_2 = \{0,1\}$ be the binary field. Let $\boldsymbol{u} = (\boldsymbol{u}^{(0)}, \boldsymbol{u}^{(1)}, \cdots, \boldsymbol{u}^{(L-1)})$ with $\boldsymbol{u}^{(i)} \in \mathbb{F}_2^k$ be the original payload data need to be transmitted. Let $\mathscr{C}_p[n,k]$ be an LDPC code, referred to as payload code, whose parity-check matrix is denoted as $\mathbf{H}$. Without extra bits, $\boldsymbol{u}$ is encoded by the encoding algorithm of $\mathscr{C}_p[n,k]$, resulting in L LDPC codewords $\boldsymbol{v} = (\boldsymbol{v}^{(0)}, \boldsymbol{v}^{(1)}, \cdots, \boldsymbol{v}^{(L-1)})$ such that $\boldsymbol{v}^{(i)}\mathbf{H}^{\mathrm{T}} = \mathbf{0}$ and $\boldsymbol{v}\widetilde{\mathbf{H}}^{\mathrm{T}} = \mathbf{0}$, where

$$\widetilde{\mathbf{H}} = \mathrm{diag}\underbrace{\{\mathbf{H}, \cdots, \mathbf{H}\}}_{L}$$

and $\mathrm{diag}\{\mathbf{H}, \cdots, \mathbf{H}\}$ is a block diagonal matrix with $\mathbf{H}$ on the diagonal.

Now we assume that, in addition to the payload data $\boldsymbol{u}$, a few extra bits $\boldsymbol{f} \in \mathbb{F}_2^{\ell L}$ with $\ell L \ll kL$ need to be transmitted. This can be achieved by using free-ride coding [7], [8] based on the original payload coded link. In this paper, we employ Reed-Muller (RM)-based free-ride codes, denoted as $\mathscr{C}_f[nL, \ell L]$. Let $\mathbf{G}_{\mathrm{RM}}$ denote a generator matrix of an RM code. A generator matrix $\widetilde{\mathbf{G}}_f$ of the free-ride code $\mathscr{C}_f[nL, \ell L]$ can be found from

$$\widetilde{\mathbf{G}}_f \widetilde{\mathbf{H}}^{\mathrm{T}} = \widetilde{\mathbf{G}}_{\mathrm{RM}} \Pi, \tag{1}$$

where $\widetilde{\mathbf{G}}_{\mathrm{RM}} = \mathrm{diag}\{\mathbf{G}_{\mathrm{RM}}, \cdots, \mathbf{G}_{\mathrm{RM}}\}$ with L $\mathbf{G}_{\mathrm{RM}}$ and Π is a row-column interleaver of size nL. Simply, by assuming that the first $m = n - k$ columns (denoted collectively as $\mathbf{B}$) of $\mathbf{H}$ are linearly independent, we have $\widetilde{\mathbf{G}}_f = \mathrm{diag}\{\mathbf{G}_f, \cdots, \mathbf{G}_f\}$, where $\mathbf{G}_f$ can be found as

$$\mathbf{G}_f = [\mathbf{G}_{\mathrm{RM}}(\mathbf{B}^{-1})^{\mathrm{T}}, \mathbf{0}_{\ell \times k}]. \tag{2}$$

Given the generator matrix $\widetilde{\mathbf{G}}_f$, the encoding is implemented in a superposition manner. The extra data $\boldsymbol{f} \in \mathbb{F}_2^{\ell L}$ is encoded by the free-ride code $\mathscr{C}_f[nL, \ell L]$, resulting in a free-ride codeword as $\boldsymbol{g} = \boldsymbol{f}\widetilde{\mathbf{G}}_f \in \mathbb{F}_2^{nL}$. Then the transmitted codeword $\boldsymbol{c}$ can be calculated as

$$\boldsymbol{c} = \boldsymbol{v} + \boldsymbol{g}, \tag{3}$$

where "+" is the addition over $\mathbb{F}_2^{nL}$. The decoding is performed in a successive cancellation manner, where the extra bits are firstly recovered and then the payload data is recovered. For more detail, see [8]. From the encoding, we see that the extra bits are transmitted along with the payload, costing no extra bandwidth. This figure of merit can be exploited to construct new coupling LDPC codes, say, implicit globally-coupled LDPC codes, implicit product-LDPC codes and terminated spatially-coupled LDPC codes [9]–[11].

B. Partial Product-LDPC Codes

Let $\mathscr{C}[n, k]$ be an LDPC code (chosen as the row component code) with length n and dimension k, whose parity-check matrix is denoted as $\mathbf{H}$. Let $\mathscr{C}_1[n_1, k_1]$ be an algebraic code (chosen as the column component code) with length n_1 and dimension k_1, whose parity-check matrix is denoted as $\mathbf{H}_1$. Let $\mathscr{C}_f[nn_1, \ell n_1]$ be a free-ride code with length nn_1 and dimension ℓn_1, whose generator matrix is denoted as $\widetilde{\mathbf{G}}_f$. Let $\boldsymbol{u}$ denote an information array with n_1 rows (each of which denoted as $\boldsymbol{u}^{(i)}$) and k columns (each of which denoted as $\boldsymbol{u}_j$) to be transmitted. The proposed partial product code is illustrated in Fig. 1. Each row $\boldsymbol{u}^{(i)} \in \mathbb{F}_2^k$ for $0 \leqslant i \leqslant n_1 - 1$ is protected using $\mathscr{C}[n, k]$, resulting in n_1 LDPC codewords, denoted as $\boldsymbol{v} = (\boldsymbol{v}^{(0)}, \boldsymbol{v}^{(1)}, \cdots, \boldsymbol{v}^{(n_1-1)})$ with $\boldsymbol{v}^{(i)} \in \mathbb{F}_2^n$. Different from the conventional product codes, only k_s columns[1] $\boldsymbol{u}_j \in \mathbb{F}_2^{n_1}$ for $0 \leqslant j \leqslant k_s - 1$ are protected using $\mathscr{C}_1[n_1, k_1]$ as

$$\boldsymbol{u}_j \mathbf{H}_1^{\mathrm{T}} = \boldsymbol{s}_j, \tag{4}$$

resulting in k_s syndromes, denoted as $\boldsymbol{s} = (\boldsymbol{s}_0, \boldsymbol{s}_1, \cdots, \boldsymbol{s}_{k_s-1})$ with $\boldsymbol{s}_j \in \mathbb{F}_2^{n_1-k_1}$. Also distinguished from the conventional product codes, the syndromes are transmitted implicitly rather than explicitly by using the free-ride coding. For notational
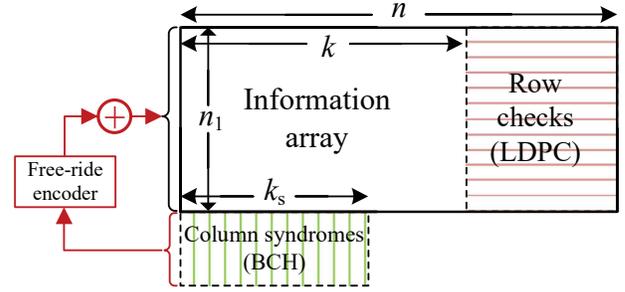
Fig. 1. An illustration of the rate-lossless partial product code, where the column syndromes are transmitted implicitly rather than explicitly by using the free-ride coding over all LDPC codewords.

convenience, the column syndromes $\{\boldsymbol{s}_j, 0 \leqslant j \leqslant k_s - 1\}$ are written as (if necessary, $\ell n_1 - k_s(n_1 - k_1)$ zeros are padded) $\boldsymbol{f} \in \mathbb{F}_2^{\ell n_1}$, which is encoded by free-ride codes and transmitted as extra bits. The encoding algorithm of the proposed partial product-LDPC codes is summarized in Algorithm 1.

Remark: For the presented partial product code, the information length is $n_1 k$ and the transmitted codeword length is $n_1 n$, implying an actual code rate k/n which is the same as that of the row LDPC code. For this reason, we say that the proposed partial product codes are *rate-lossless*.

Algorithm 1 Encoding of the Rate-Lossless Partial Product-LDPC Codes

Input: Information array $\boldsymbol{u}$ of size $n_1 \times k$.
1) **Row Encoding**: Take as input each row $\boldsymbol{u}^{(i)} \in \mathbb{F}_2^k$ of the information array $\boldsymbol{u}$ to the payload LDPC encoder and deliver as output an LDPC codeword $\boldsymbol{v}^{(i)} \in \mathbb{F}_2^n$, resulting in n_1 LDPC codewords $\boldsymbol{v} = (\boldsymbol{v}^{(0)}, \boldsymbol{v}^{(1)}, \cdots, \boldsymbol{v}^{(n_1-1)})$.
2) **Column Encoding**: Take as input each column $\boldsymbol{u}_j \in \mathbb{F}_2^{n_1}$ for $0 \leqslant j \leqslant k_s - 1$ of the information array $\boldsymbol{u}$ to the algebraic code and deliver as output $\boldsymbol{s}_j = \boldsymbol{u}_j \mathbf{H}_1^{\mathrm{T}}$, resulting in k_s column syndromes, denoted as $\boldsymbol{f} \in \mathbb{F}_2^{\ell n_1}$.
3) **Free-Ride Encoding**: Encode the column syndromes $\boldsymbol{f}$ by the free-ride code $\mathscr{C}_f[nn_1, \ell n_1]$, resulting in a free-ride codeword $\boldsymbol{g} = \boldsymbol{f}\widetilde{\mathbf{G}}_f \in \mathbb{F}_2^{nn_1}$. Then the transmitted codeword $\boldsymbol{c} \in \mathbf{F}_2^{nn_1}$ is calculated by

$$\boldsymbol{c} = \boldsymbol{v} + \boldsymbol{g}, \tag{5}$$

where "+" is the addition over $\mathbb{F}_2^{nn_1}$.

Output: Transmitted codeword $\boldsymbol{c}$ of length nn_1.

At the receiver, upon receiving the noisy version $\boldsymbol{y} = (\boldsymbol{y}^{(0)}, \boldsymbol{y}^{(1)}, \cdots, \boldsymbol{y}^{(n_1-1)})$ of $\boldsymbol{c}$, we can calculate the corresponding LLRs $\boldsymbol{\Lambda} = (\boldsymbol{\Lambda}^{(0)}, \boldsymbol{\Lambda}^{(1)}, \cdots, \boldsymbol{\Lambda}^{(n_1-1)})$ and perform an iterative decoding, as described in Algorithm 2. Particularly, at the column decoding stage, if the hard decision decoding (HDD) of $\boldsymbol{u}_j$ is successful, update the LLRs associated with $\boldsymbol{u}_j$ as fixed values (either positive or negative with large amplitude) and exploit these "hardened" messages to improve the free-ride decoding and the row decoding.

---

[1]Considering that each LDPC codeword can carry ℓ extra bits, we choose k_s such that $k_s(n_1 - k_1) \leqslant \ell n_1$.

Algorithm 2 Decoding of the Rate-Lossless Partial Product-LDPC Codes

Input: LLRs $\mathbf{\Lambda}$.

repeat

1) **Free-Ride Decoding**: Take as input the LLRs $\mathbf{\Lambda} = (\mathbf{\Lambda}^{(0)}, \mathbf{\Lambda}^{(1)}, \cdots, \mathbf{\Lambda}^{(n_1-1)})$ to the free-ride decoder of $\mathscr{C}_f[nn_1, \ell n_1]$ and deliver as output the estimation of column syndromes $\widehat{\boldsymbol{f}}$. Remove the effects of $\widehat{\boldsymbol{g}} = \widehat{\boldsymbol{f}}\widetilde{\mathbf{G}}_f$ on $\mathbf{\Lambda}$ to obtain the updated LLRs $\widetilde{\mathbf{\Lambda}}$ of $\boldsymbol{v}$.

2) **Row Decoding**: For the unrecovered $\boldsymbol{v}^{(i)}$ indicated by the parity check, take as input $\widetilde{\mathbf{\Lambda}}^{(i)}$ of $\widetilde{\mathbf{\Lambda}}$ to the LDPC decoder and deliver as output the estimation $\widehat{\boldsymbol{v}}^{(i)}$.

3) **Column Decoding**: Perform an HDD of the algebraic code for $\boldsymbol{u}_j$ $(0 \leqslant j \leqslant k_s - 1)$ based on $\widehat{\boldsymbol{f}}$. For each $\boldsymbol{u}_j$, if the HDD is successful, update the LLRs associated with $\boldsymbol{u}_i$ as fixed values (either positive or negative with large amplitude), resulting in an updated LLRs stored in $\mathbf{\Lambda}$.

until $\widehat{\boldsymbol{v}}^{(i)}\mathbf{H}^{\mathrm{T}} = \mathbf{0}$ *for* $0 \leqslant i \leqslant n_1 - 1$ *or a preset global maximum iteration number* $I_{\max}$ *is reached*;

Output: Estimation of the information array $\widehat{\boldsymbol{u}}$.

III. PERFORMANCE ANALYSIS AND CODE DESIGN

A. Free-Ride Codes: High Reliable Column Syndromes

From the decoding algorithm, we can see that the free-ride decoding is crucial for the presented product-LDPC codes. Let WER_p be the word error rate of the original payload link without extra bits, WER_f be the word error rate of the extra bits, and WER_{pf} be the word error rate of the payload data with extra bits. Then, as shown in [9], we have the upper bound for WER_{pf} as

$$\mathrm{WER}_{pf} \leqslant \mathrm{WER}_p + \mathrm{WER}_f. \tag{6}$$

***Example** 1*: To illustrate the performance of the free-ride codes, we take a rate-$1/2$ irregular LDPC code of length 1024 as the payload code, which is constructed based on the variable node degree distribution (from the perspective of the edge) $\lambda(x) = 0.32660x + 0.11960x^2 + 0.18393x^3 + 0.36988x^4$ [12]. The length of extra bits for each LDPC codeword is $\ell = 6, 8$ or 10. The simulation results by assuming additive white Gaussian noise (AWGN) channels and binary phase-shift keying (BPSK) modulation are shown in Fig. 2, where we observe that the extra bits (labeled as WER_f) are more reliable than the payload data (labeled as WER_p). We also observe that the extra bits become more reliable with smaller ℓ (the average number of extra bits carried by one payload codeword) or larger L (introducing interleaving gain). The upper bound and performance of WER_{pf} reveal that transmitting not too many extra bits has negligible effect on the payload data.

B. Row Codes: LDPC Codes with Known Bits as Lower Bounds

Obviously, the performance of the row decoding can not be better than that in the case when both the free-ride codes and the column codes are perfectly decoded. This motivates us to present a GA lower bound by simulating the LDPC codes with known bits. Without loss of generality, we assume that
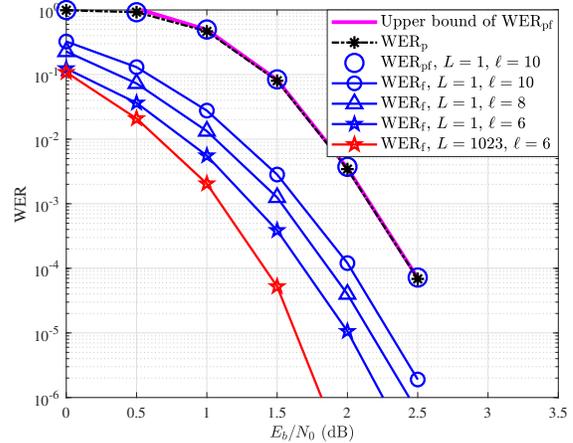
Fig. 2. Performance of the payload data (with or without extra bits) and the extra bits, where ℓL extra bits are first encoded by the RM-based free-ride codes and then carried by L payload LDPC codewords. The payload code is a rate-$1/2$ irregular LDPC code of length 1024.
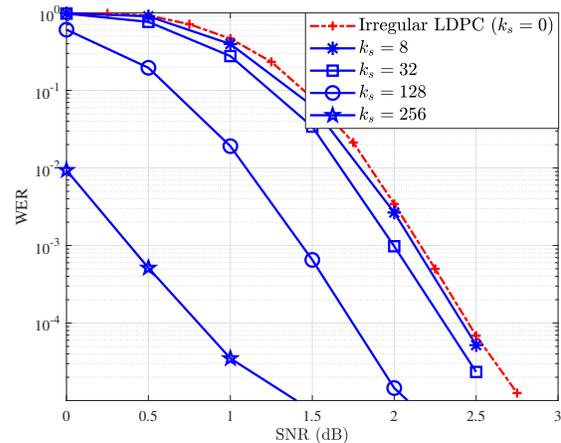
Fig. 3. Performance of an LDPC code with k_s known bits at the decoder. The LDPC code is a rate-$1/2$ irregular LDPC code of length 1024.

the first k_s coded bits, denoted as $\boldsymbol{v}_L$, are told by a genie to the decoder, and the remained $n - k_s$ coded bits, denoted as $\boldsymbol{v}_R$, need to be recovered. The parity-check matrix $\mathbf{H}$ of the LDPC code is correspondingly divided into two parts, expressed as $\mathbf{H} = [\mathbf{H}_L, \mathbf{H}_R]$. With this assumption, we can perform an iterative belief propagation (BP) decoding algorithm (say, the sum-product algorithm (SPA) [13]) over a partial Tanner graph specified by

$$\boldsymbol{v}_R\mathbf{H}_R^{\mathrm{T}} = \boldsymbol{v}_L\mathbf{H}_L^{\mathrm{T}}, \tag{7}$$

which is equivalent to $\boldsymbol{v}_R\mathbf{H}_R^{\mathrm{T}} = \mathbf{0}$ over binary input output symmetric (BIOS) channels.

***Example** 2*: We take a rate-$1/2$ irregular LDPC code of length 1024 used in Example 1 for the LDPC coded transmission. The remained coded bits $\boldsymbol{v}_R$ are transmitted over AWGN channels using BPSK modulation. At the receiver, the SPA over a partial Tanner graph specified by (7) with a maximum iteration number of 50 is employed for the decoding. The WER performance with different k_s is shown
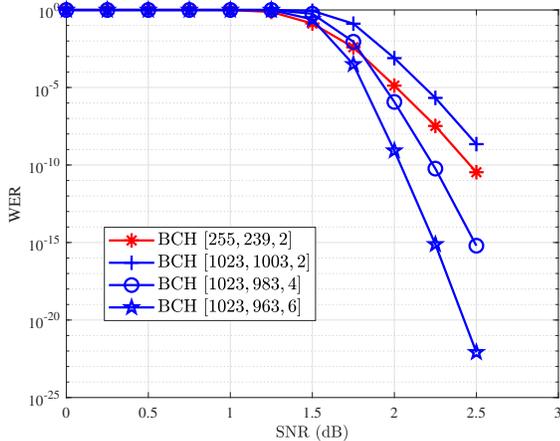
Fig. 4. Approximated bounds on WER for the doubly-protected column information with different BCH codes.

Fig. 5. Performance of the proposed partial product-LDPC code, where $I_{\max}$ is the global iteration number of Algorithm 2. The partial product code is constructed from a rate-1/2 irregular LDPC code $[1024, 512]$ and a BCH code $[1023, 983, 4]$.

in Fig. 3, where we observe that the decoding performance can be improved with the known bits. We also observe that the LDPC coded transmission with more known bits can be more reliable, as is consistent with our intuition.

C. Column Codes: BCH codes for Approximated Bound on the Doubly-Protected Information

In this paper, we take the primitive BCH codes $\mathscr{C}_1[n_1, k_1]$ with length $n_1 = 2^m - 1$ and dimension $k_1 = n_1 - mt$ as the column codes, which are defined over the finite field $\mathbb{F}_{2^m}$ and can correct up to t errors. We need to point out that, if the syndrome is perfectly known, the error correction capability is the same as the original BCH code.

To evaluate the performance of the decoding of the first k_s columns (referred to as doubly-protected information), we assume that the BCH decoder sees a binary symmetric channel (BSC) with crossover probability $\mathrm{BER_{pf}}$, where $\mathrm{BER_{pf}}$ is the bit error rate of the row component LDPC code with extra bits. Similar to [9], we have the following bounds on $\mathrm{BER_{pf}}$,

$$\mathrm{BER_p} \leqslant \mathrm{BER_{pf}} \leqslant \mathrm{BER_p} + \mathrm{WER_f} \qquad (8)$$

by assuming that all information bits are erroneous when free-ride decoding is unsuccessful, where $\mathrm{BER_p}$ is the bit error rate of the row component LDPC code without extra bits and $\mathrm{WER_f}$ is the word error rate of the free-ride code. As indicated by Fig. 2, $\mathrm{WER_f}$ is usually much less than $\mathrm{BER_p}$ for small number of extra bits, resulting in $\mathrm{BER_{pf}} \gtrsim \mathrm{BER_p}$. Under these assumptions, the WER of the column decoder for the doubly-protected information can be estimated as

$$\mathrm{WER} = \sum_{i=t+1}^{n_1} \binom{n_1}{i} \mathrm{BER_p}^i (1 - \mathrm{BER_p})^{n_1 - i}, \qquad (9)$$

where the summation variable i denotes the total number of erroneous bits in a column.

Example 3: Consider the rate-1/2 irregular LDPC code that was used in Example 1. The approximated bounds on WER for the doubly-protected column information with different BCH codes are shown in Fig. 4, where we observe that the approximated bounds can be improved by increasing the error correction capability t/n_1, as is consistent with our intuition.
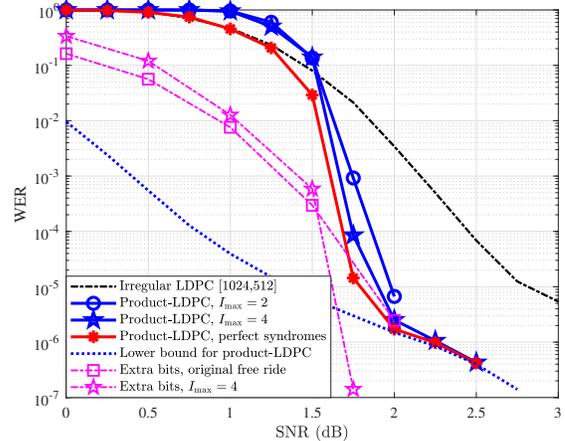
IV. NUMERICAL RESULTS

In this section, we present the numerical results of the proposed partial product-LDPC codes. We take the rate-1/2 irregular LDPC codes or $(3,6)$-regular LDPC codes as the row codes and the primitive BCH codes as the column codes. The iterative decoding algorithm (Algorithm 2) is employed for the decoding. Unless otherwise stated, the maximum iteration number of the SPA is 50 for the benchmark LDPC codes and the row component LDPC codes. In all examples, the codewords $\boldsymbol{c}$ are transmitted using BPSK signaling over AWGN channels. Each LDPC codeword carries $\ell = 10$ extra bits in all simulations.

Example 4 (Different $I_{\max}$): Take the rate-1/2 irregular LDPC code $[1024, 512]$ used in Example 1 as the row code and the BCH code $[1023, 983]$ as the column code. The first $k_s = 255 = \lfloor 1023 \times 10/40 \rfloor$ columns of $\boldsymbol{u}$ are chosen to generate the column syndromes $\boldsymbol{f}$. The WER performances of the proposed partial product-LDPC code with different $I_{\max}$ are shown in Fig. 5, where we observe that the performance can be improved by increasing $I_{\max}$, approaching the performance with perfectly known column syndromes. The WER performance of the row component LDPC code can be reduced from 10^{-3} to 10^{-6} at the SNR around 2 dB, yielding an extra coding gain of up to 0.9 dB at WER $= 10^{-5}$. Moreover, the WER performances of the proposed codes match well with the GA bounds in the high SNR region, implying that the error-floor performance can be predicted by the proposed lower bound. We also observe that, in comparison with the original free-ride codes, the extra bits (column syndromes) performance can be improved by the presented iterative decoding algorithm.

In addition, the WER performance of the doubly-protected information after BCH decoding with $I_{\max} = 2$ and its approx-
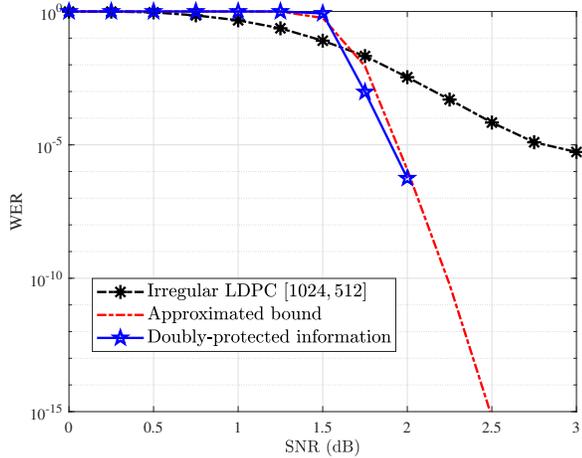
Fig. 6. Performance of the benchmark component LDPC code, the doubly-protected information, and the corresponding approximated bound. The first $k_s = 255$ columns of the information array is protected by both the irregular LDPC code $[1024, 512]$ and the BCH code $[1023, 983]$.

Fig. 7. Performance of the proposed partial product-LDPC code with different column codes, where $I_{\max} = 4$. The row code is a rate-1/2 irregular LDPC code $[1024, 512]$ and the column code is a BCH code of $[255, 239]$, $[1023, 983]$ or $[1023, 963]$.

imated bound are shown in Fig. 6, where we observe that the WER performance of the doubly-protected information after BCH decoding is close to the presented approximated bound, implying that the doubly-protected information can reach a WER of 10^{-15} at SNR around 2.5 dB.

Example 5 (Different BCH Codes): Take the rate-1/2 irregular LDPC code $[1024, 512]$ used in Example 1 as the row code. The column code is a BCH code $[255, 239]$, $[1023, 983]$ or $[1023, 963]$, corresponding to the number of doubly-protected columns $k_s = 159, 255$ or 170, respectively. The WER performances of the proposed partial product-LDPC codes with different BCH codes are shown in Fig. 7, where we observe that the presented product code with the stronger BCH code can have a better waterfall performance but a slightly poorer error-floor performance since k_s is smaller when the BCH code is stronger. The presented product codes with properly selected column codes can achieve a good trade off between the waterfall performance and error-floor performance.

Example 6 (Different LDPC Codes): Take the rate-1/2 irregular LDPC code $[1024, 512]$ used in Example 1 or the $(3, 6)$-regular LDPC code $[1024, 512]$ as the row code. The column code is a BCH code $[1023, 983]$. The number of doubly-protected columns is $k_s = 255 = \lfloor 1023 \times 10/40 \rfloor$. The WER performances of the proposed partial product-LDPC codes with different LDPC codes are shown in Fig. 8, where we observe that the presented product code with the regular LDPC code has better error-floor performance and with the irregular LDPC code has better waterfall performance.

V. CONCLUSIONS

This paper has presented the rate-lossless product-LDPC codes, where the partial column syndromes of the doubly-protected information are transmitted implicitly using the free-ride coding. For the code design, we have discussed in more detail of the impact of row LDPC codes and column BCH
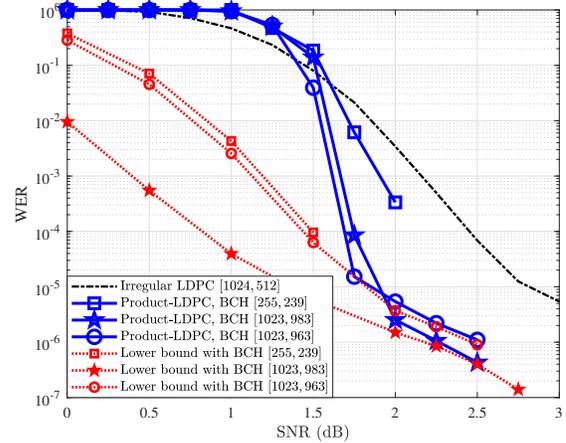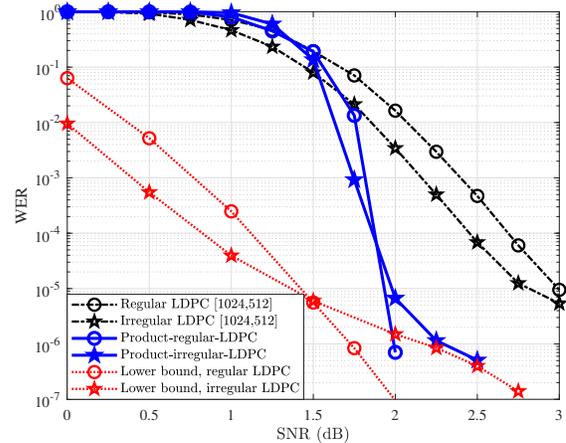
Fig. 8. Performance of the proposed partial product-LDPC code with different row codes, where $I_{\max} = 2$. The column code is a BCH code $[1023, 983]$ and the row code is a $(3, 6)$-regular LDPC code or a irregular LDPC code.

codes on the performance. For the performance analysis, we have proposed a lower bound which can predict the error-floor performance of the proposed product-LDPC codes, and an approximated bound which can estimate the high reliability of the doubly-protected information. These bounds together with the simulation results have shown that the presented product-LDPC codes can outperform the component LDPC codes, yielding an extra coding gain of up to 1 dB. One of our future works is to investigate the choice of the doubly-protected columns (which directly affect the lower bound and the decoding performance), possibly yielding a further performance improvement.

REFERENCES

[1] P. Elias, "Error-free coding," *Trans. IRE Prof. Group Inf. Theory*, vol. IT-4, no. 4, pp. 29–37, Sept. 1954.
[2] R. Pyndiah, "Near-optimum decoding of product codes: Block turbo codes," *IEEE Trans. Commun.*, vol. 46, no. 8, pp. 1003–1010, Aug. 1998.

[3] H. Mukhtar, A. Al-Dweik, and A. Shami, "Turbo product codes: Applications, challenges, and future directions," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 4, pp. 3052–3069, 4th Quart. 2016.

[4] IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Broadband Wireless Access Systems, IEEE Standard 802.16, May 2009.

[5] IEEE Standard for Local and Metropolitan Area Networks Part 20: Air Interface for Mobile Broadband Wireless Access Systems Supporting Vehicular Mobilityphysical and Media Access Control Layer Specification, IEEE Standard 802.20, Aug. 2008.

[6] IEEE Standard for Broadband Over Power Line Networks: Medium Access Control and Physical Layer Specifications, IEEE Standard 1901, Dec. 2010.

[7] S. Cai, S. Zhao, and X. Ma, "Packing additional bits into LDPC coded data," *Electron. Lett.*, vol. 55, no. 18, pp. 997–998, Sept. 2019.

[8] S. Cai, S. Zhao, and X. Ma, "Free ride on LDPC coded transmission," *IEEE Trans. Inf. Theory*, vol. 68, no. 1, pp. 80–92, Jan. 2022.

[9] Q. Wang, S. Cai, and X. Ma, "Free-ride coding for constructions of coupled LDPC codes," *submitted to IEEE Trans. Commun.*, 2022.

[10] X. Ma, Q. Wang, S. Cai, and X. Xie, "Implicit partial product-LDPC codes using free-ride coding," in *Proc. IEEE Int. Conf. Commun.*, 2022, accepted.

[11] X. Ma, Q. Wang, M. Xie, and S. Cai, "Implicit globally-coupled LDPC codes using free-ride coding," in *Proc. IEEE Wireless Commun. Network Conf.*, 2022, pp. 1117–1122.

[12] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 619–637, Feb. 2001.

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