



RESEARCH ARTICLE

Reduced-Complexity Decoding of 3D Product Codes for Satellite Communications

Yue Xu¹, Ming Jiang^{1,2*}, Mingyang Zhu¹, Nan Hu³, Chunming Zhao^{1,2}, and Jiangzhou Wang⁴

¹National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China. ²Purple Mountain Laboratories, Nanjing 210096, China. ³Institute of Wireless and Terminal Technology, China Mobile Research Institute, Beijing 100032, China. ⁴School of Engineering, University of Kent, Canterbury CT2 7NT, UK.

*Address correspondence to: jiang_ming@seu.edu.cn

This paper aims to provide a reduced-complexity decoding for satellite communication systems to enhance system performance and transmission efficiency. By addressing the complexity, error rate, and latency issues for satellite communications, a novel 3-dimensional product decoding scheme is proposed. The density evolution algorithm is also introduced to analyze and optimize for the binary message passing decoding, which further reduces the computational complexity and latency. The simulation results show that the proposed method can obtain about 0.3- and 0.15-dB performance gains compared to the similar decodings of the conventional 2-dimensional BCH product codes and staircase codes, respectively. Our proposed coding scheme offers a broad prospect to the critical issues of transmission latency and data throughput in satellite networks.

Introduction

Satellite communication is considered to be a key technology for the next-generation wireless communication system because of its inherent characteristics of wide coverage and high data rate, which make up for the deficiency of the ground relay systems. Although the satellite communication has many advantages, there are also some inherent problems, such as obstacle occlusion [1–4] and long-distance transmission delay.

Because of the vast distance between ground stations and satellites, the data-link delay is a critical issue for satellite communications [5] especially for geosynchronous satellite communication system, where the round trip delay is approximately 240 to 279 ms. In addition, different satellite elevation angles also have an important impact on propagation delay [6]. Therefore, the forward error correction schemes have been widely used in satellite communication instead of automatic request-for-repeat schemes [7]. The classic convolution codes and Reed–Solomon codes have been progressively substituted by some modern advanced coding techniques such as low-density parity-check (LDPC) codes. The paper [8] proposed a practical multirate quasi-cyclic LDPC coding scheme suitable for satellite communications, which has better performance than the LDPC codes in the digital video broadcasting-satellite second-generation extension (DVB-S2X) standard at all code rates. Polar codes [9] and new turbo product codes (TPCs) [10] have capacity-approaching performance under iterative soft-decision decodings (SDDs) [11–13] also as emerging technologies in satellite systems. The traditional SDD method with high complexity is not suitable for satellite applications due to the hardware resource constraints. The hard-decision decodings (HDDs) are more efficient

and easier to use, which have led to their applications in the areas where low complexity and reliable high throughput are greatly desired [14].

Among those modern coding schemes that can be applied with high-throughput HDDs, the product codes have received wide attention. This coding scheme was first proposed by Elias [15] and received renewed interest after about 40 years because of Pyndiah's iterative soft-input soft-output decoding of TPC [16]. By coding each direction of a multidimensional information block separately, the coded blocks in one dimension are equivalent to the codewords in another way after interleaving. The decoding results of any one dimension can give extra benefit to the decoding in the other dimensions, which makes the error correction capability better. The paper [17] introduced a novel decoder for product codes based on error-and-erasure component code decoder and dynamic reliability scores that can perform very closely to the SDD of TPC codes [16].

To achieve ultrahigh-throughput decoding of product codes, there are many studies on HDD algorithms, such as the iterative decodings based on bounded distance decoding (BDD) [18,19]. The traditional HDD makes decisions directly through the channel output, which causes a lot of information lost, and the decoding performance also has about 2-dB loss compared to that of SDD. To balance the error correction capability and decoding complexity, the current practical methods are to use a certain amount of channel information. In [20], it was mentioned that the new binary information can be reconstructed from the output of BDD. In addition, the channel soft information has been used for the product-like codes in [21]. Moreover, the iterative BDD with scaled reliability (iBDD-SR) was proposed in [22],

Citation: Xu Y, Jiang M, Zhu M, Hu N, Zhao C, Wang J. Reduced-Complexity Decoding of 3D Product Codes for Satellite Communications. *Space Sci. Technol.* 2024;4:Article 0096. <https://doi.org/10.34133/space.0096>

Submitted 8 July 2023
Accepted 16 October 2023
Published 23 January 2024

Copyright © 2024 Yue Xu et al.
Exclusive licensee Beijing Institute of Technology Press. No claim to original U.S. Government Works. Distributed under a Creative Commons Attribution License 4.0 (CC BY 4.0).

and the density evolution (DE) was used to optimize the scaling factors. A ternary message passing decoding algorithm for product codes was presented in [23]. The related algorithms were also used for the decoding schemes of staircase codes (SCCs) [24], which are promising to achieve better performance due to the more complicated coupling structure than that of product codes. A new class of SCCs was introduced in [25], where the rearranged precoding blocks and new information blocks were jointly encoded to achieve better waterfall and error floor than the original SCCs [24]. The paper [26] showed that extending the DE analysis for product codes and SCCs to a channel with ternary output and ternary message passing, where the third symbol marks an erasure, can achieve additional coding gains of up to 0.6 dB compared to the traditional binary message passing decoding. Many research studies aiming at improving error correction capabilities from the decoding perspective are achieved by introducing more soft information, at the cost of larger decoding data streams and higher computational complexity [27–29]. To further improve the decoding performance and reduce the delay caused by retransmissions, a higher-order encoding was considered. In [30], a block encoding method based on BCH and single parity check (SPC) construction, which we call BCH²SPC, was proposed. However, the adaptive belief propagation (ABP) decoding proposed is not suitable for the ultralong code length, high code rate, and low delay requirements of satellite communications.

In this letter, we propose a simplified binary passing decoding algorithm, termed iBDDSR-MSA (iBDD with scaled reliability and min-sum algorithm), to reduce the hardware implementation

complexity and minimize the storage overhead, which are also the important issues in satellite communications. By using the iBDD-SR algorithm and studying the DE under the additive white Gaussian noise (AWGN) channel, we calculate the theoretically optimal scaling factors according to the Gaussian approximation (GA) and improve the error correction capability with low additional computational complexity and decoding data stream. Our proposed decoding method can systematically concatenate many conventional 2-dimensional (2D) product codes with large block-lengths, which provide substantial performance gain for satellite communications with quite low complexity in decoding implementations. Our numerical results show that the multiplicative codes and proposed enhanced iterative decoding can achieve ultrahigh throughput with a more flexible parallelism.

Notations: When we define a notation by a lower-case letter (e.g., x), we will use the corresponding uppercase letter to represent the underlying random variable (e.g., X). $|a|$ denotes the absolute value of a . A Gaussian distribution with mean μ and variance σ^2 is denoted by $\mathcal{N}(\mu, \sigma^2)$. $\mathbb{E}[\cdot]$ is the expectation, and $\mathbb{E}_X[\cdot]$ takes the expectation concerning the random variable X .

Methods

3D product codes using BCH and SPC codes

As shown in Fig. 1, an s -layer 3D product code BCH²SPC is constructed with the first $s - 1$ layers being independent 2D BCH product codes and the s th layer consisting of parity-check bits. Each BCH product code is obtained by encoding

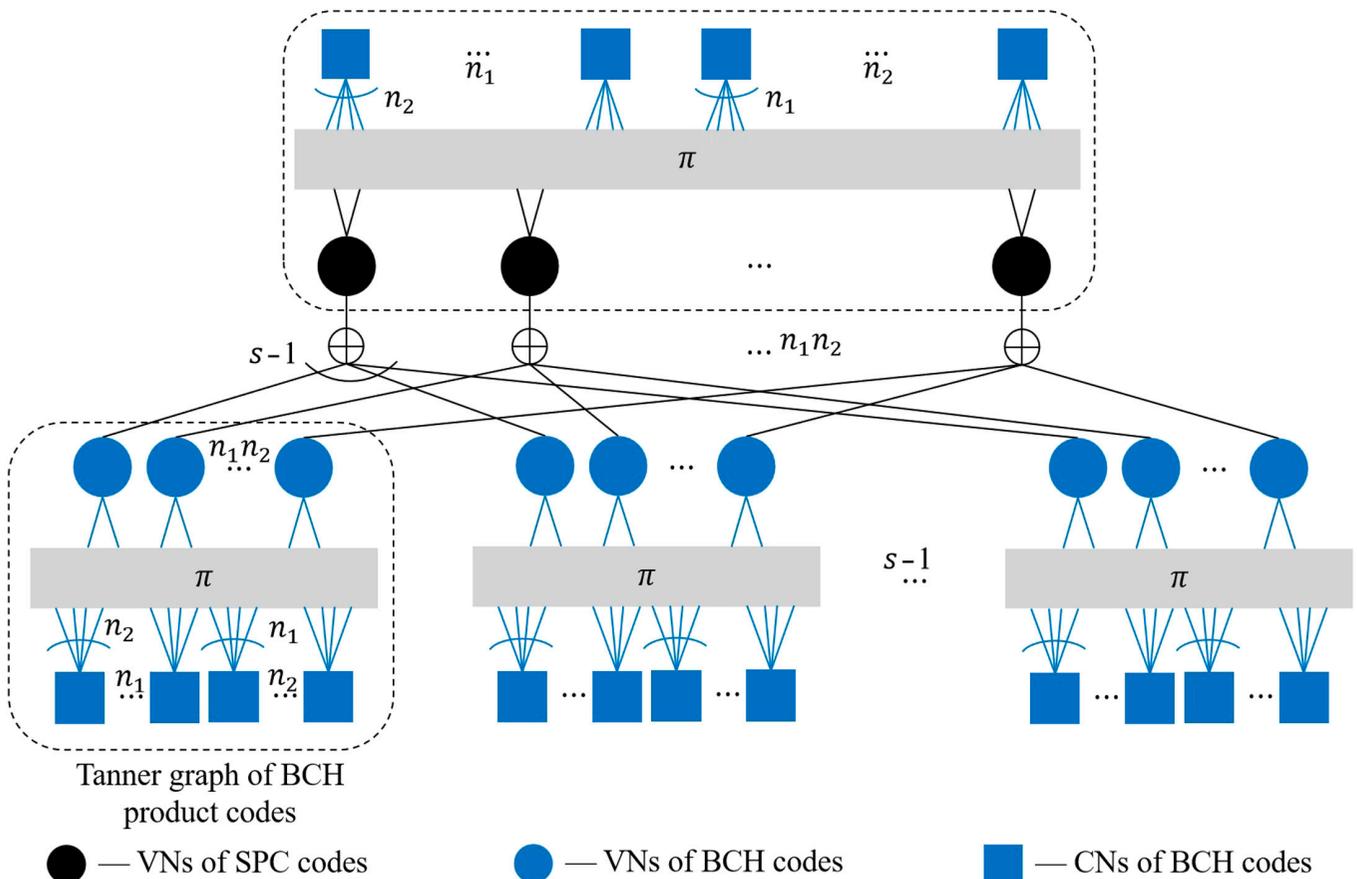


Fig. 1. The Tanner graph of the BCH²SPC product codes.

an (n_d, r_d, t_d) BCH code in 2 dimensions, where n_d , r_d , and t_d are code length, code rate, and error correction capability of the d th dimension coding, $d = 1, 2$, respectively. Then, the 3D product code with a total code length $N = n_1 n_2 \times s$ and code rate $R = r_1 r_2 \times (s - 1) / s$ is generated, which is denoted as $\mathcal{P}_{3D}(N, R)$.

Since the conventional 2D product codes can be viewed as a special kind of general LDPC codes, the connections on Tanner graph between variable nodes (VNs) and constraint nodes (CNs) can be easily determined. Figure 1 depicts the Tanner graph of the 3D product codes, where the black circles, the blue circles, and blue square blocks denote the VNs of SPC codes and the VNs and CNs of BCH codes, respectively. Each VN of BCH codes connects to 3 nodes, 2 of which are the CNs of BCH codes and the third one is a degree- s parity-check nodes of SPC code. The relationship between product codes and general LDPC codes dictates that the DE analysis applied for LDPC codes also can be conveniently used for product codes whose nodes in Tanner graph also have a tree-like neighborhood down to a certain depth.

The generator matrix of the 3D product codes can be described as $\mathbf{G} = \mathbf{G}_1 \otimes \mathbf{G}_2 \otimes \mathbf{G}_3$, where \mathbf{G}_1 , \mathbf{G}_2 , and \mathbf{G}_3 are the generator matrices of 3 component codes (2 BCH codes and 1 SPC code), respectively. For an SPC code of length s , its matrix \mathbf{G}_3 can be represented as

$$\mathbf{G}_3 = \begin{bmatrix} \mathbf{I}_{s-1} & \mathbf{1}_{(s-1) \times 1} \\ \mathbf{0}_{1 \times (s-1)} & \mathbf{1} \end{bmatrix}_{s \times s}. \quad (1)$$

To facilitate a better understanding of our proposed decoding scheme, we first take the 2D product code with BCH codes as an example and introduce the iBDD-SR algorithm here. Consider the decoding of the row component codes at the k th iteration. Let $\Lambda^{r,k}$ be the hard input of the row component BDDs at the k th iteration. Then, map the results according to $0 \mapsto +1$ and $1 \mapsto -1$ if one component decoder is successful. Otherwise, the outputs are all zeros if decoding fails. Then, the matrix of extrinsic information from row decoders \mathbf{U}_r^k is obtained. By combining with the log likelihood ratios (LLRs) \mathbf{L}_{ch} from the channels, the final outputs $\Lambda^{c,k}$ transmitted to the column decoder can be described as the hard decisions of the channel outputs and row BDDs outputs, $\Lambda^{c,k} = H(\mathbf{L}_{ch} + w_r^k \mathbf{U}_r^k)$, where $H(\cdot)$ refers to a bitwise operation on each element of the matrix, $H(\cdot) = \frac{1 - \text{sign}(\cdot)}{2}$, and w_r^k is the scaling factor corresponding to the output reliabilities of row component BDDs in the k th iteration. Similar operations are performed in the BDDs of the column component codes. The iterative process continues until the maximum number of iterations is reached. By exchanging information between row and column component decoders in the iterative process, the error correction capability of 2D product codes will be substantially improved.

Figure 2 shows the block diagram of the proposed iBDDSR-MSA of BCH²SPC. Without loss of generality, assume that decoding starts with the BDDs of the row component decoders at the k th iteration. Let $\Lambda^{r,k}$ of size $n_1 \times n_2 \times s$ with elements in $\{1, 0\}$ denotes the input matrix of the row component BDDs; each layer is a 2D decoding matrix of a BCH product code. It is noteworthy that the input information of the row component decoders consisting of the channel LLRs L_{ch} , z component, and the column component decoders outputs $\Lambda^{r,k} = H(\mathbf{L}_{ch} + \mathbf{L}_z^{k-1} + w_c^{k-1} \mathbf{U}_c^{k-1})$, where w_c^{k-1} is the scaling factor associated with the reliability of \mathbf{U}_c^{k-1} . Initially, $\Lambda^{r,1} = H(\mathbf{L}_{ch})$. Total $n_1 \times s$ row component BDDs

can be carried out in parallel, and then the decoded output matrix \mathbf{U}_r^k is obtained. Similarly, $\Lambda^{c,k}$ is the input information of the column component decoders, $\Lambda^{c,k} = H(\mathbf{L}_{ch} + \mathbf{L}_z^{k-1} + w_r^k \mathbf{U}_r^k)$.

Algorithm 1: The decoding of BCH²SPC.

```

Data:  $L_{ch}, n_1, n_2, s, t, k_{max}, w_r, w_c$ , Encoder, Decoder, the result of channel encoding of the original input information  $B$ 
Result: Decoding result  $D$ 
for  $k \leftarrow 1$  to  $k_{max}$  do
  if  $k = 1$  then
    |  $\Lambda_c^1 \leftarrow H(L_{ch})$ 
  else
    |  $\Lambda_c^k \leftarrow H(L_{ch} + L_z^{k-1} + w_c^{k-1} U_c^{k-1})$ 
  end
  for  $i \leftarrow 1$  to  $s$  do
    for  $j \leftarrow 1$  to  $n_1$  do
      [decoded_info, correct_num]  $\leftarrow$  step(Decoder,  $\Lambda_c^k(j, :, i)^T$ );
      decoded_code  $\leftarrow$  step(Encoder, decoded_info);
      if correct_num  $\leq 0$  then
        |  $U_r^k(j, :, i) \leftarrow \mathbf{0}_{1 \times n_2}$ ;
      else
        |  $U_r^k(j, :, i) \leftarrow 1 - 2 \times \text{decoded\_code}^T$ ;
      end
    end
  end
  if  $k = 1$  then
    |  $\Lambda_r^1 \leftarrow H(L_{ch} + w_r^1 U_r^1)$ 
  else
    |  $\Lambda_r^k \leftarrow H(L_{ch} + L_z^{k-1} + w_r^k U_r^k)$ 
  end
  for  $i \leftarrow 1$  to  $s$  do
    for  $j \leftarrow 1$  to  $n_1$  do
      [decoded_info, correct_num]  $\leftarrow$  step(Decoder,  $\Lambda_r^k(i, j, :)$ );
      decoded_code  $\leftarrow$  step(Encoder, decoded_info);
      if correct_num  $\leq 0$  then
        |  $U_c^k(:, j, i) \leftarrow \mathbf{0}_{n_1 \times 1}$ ;
      else
        |  $U_c^k(:, j, i) \leftarrow 1 - 2 \times \text{decoded\_code}^T$ ;
      end
    end
  end
   $\Lambda_c^k \leftarrow H(L_{ch} + w_r^k L_r^k + w_c^k U_r^k)$ ;
  for  $i \leftarrow 1$  to  $n_1$  do
    for  $j \leftarrow 1$  to  $n_2$  do
      |  $L_z^k(i, j, :) \leftarrow \text{MSA}(\Lambda_c^k(i, j, :), 0, 8)$ 
    end
  end
   $Y_{app} \leftarrow L_{ch} + w_r^k L_r^k + w_c^k U_r^k + L_z^k$ ;
   $D \leftarrow H(Y_{app})$ ;
  if sum(sum(sum( $D \neq B$ )))  $\neq 0$  then
    | break
  end
end
end

```

The middle branch of Fig. 2 describes the decoding process in the z -component decoders. The input and output to the MSA in the z -component decoders are $\Lambda^{z,k} = \mathbf{L}_{ch} + w_r^k \mathbf{U}_r^k + w_c^k \mathbf{U}_c^k$ and \mathbf{L}_z^k , respectively. To achieve binary-only message transfer between the component decoders, the binary inputs to the row and column component decoders are also calculated at the z -component decoders end. At this point, one round of iteration is completed. The iteration process will not stop until the maximum number of iterations is reached. The whole process can be seen in Algorithm 2. In practical decoding implementations, the above processes can be carried out in parallel, which can further reduce decoding latency.

DE of 3D product codes

In this section, we present the Gaussian-approximation-assisted DE analysis of the BCH²SPC using iBDDSR-MSA under AWGN channel. We consider the binary phase shift key (BPSK) transmission and analyze the decoder behaviors by assuming the transmission of all-zero codewords. This approximate analysis helps us to efficiently obtain the optimized scaling factors and avoids an extensive Monte Carlo simulation.

For a binary-input AWGN channel with the variance σ_n^2 , we define $\mu = \frac{2}{\sigma_n^2}$ and $\sigma^2 = 2\mu$. Clearly, the channel LLR $\sim \mathcal{N}(\mu, 2\mu)$ that is computed by $l = \frac{2y}{\sigma_n^2}$. The probability that a randomly

iBDD-SR

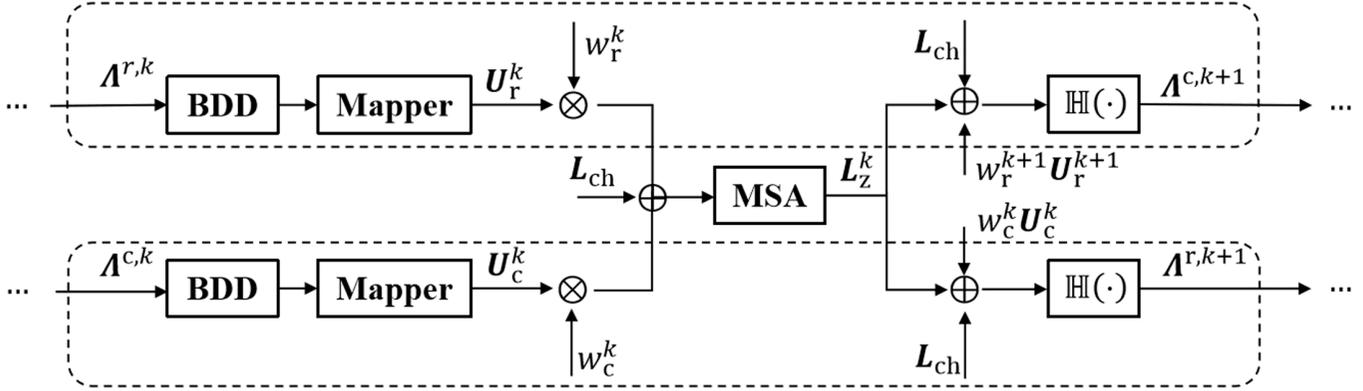


Fig. 2. Schematic of the proposed iBDDSR-MSA of the k th decoding.

selected bit in one component codeword is decoded correctly or incorrectly with t -bit error correction capability is

$$P^{+1}(i) = \begin{cases} 1 & \text{if } 0 \leq i \leq t-1 \\ \sum_{\delta=0}^{t-1} \sum_{j=0}^{\delta} \frac{n-h}{n} A_h F_h & \text{if } t \leq i \leq n-t-1 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Algorithm 2: The calculation of scaling factors.

Data: $E_b N_{0,low}, E_b N_{0,up}, E_b N_{0,diff}, n, k, t, s$
Result: $E_b N_0, W_r, W_c$
 $k_{max};$
 $E_b N_0 \leftarrow 0;$
 $w_r, w_c, W_r, W_c \leftarrow \mathbf{0}_{1 \times k_{max}};$
 $v \leftarrow \log_2^{s+1};$
 $R \leftarrow k^2 \times (s-1)/n^2/s;$
while $E_b N_{0,up} - E_b N_{0,low} \leq E_b N_{0,diff}$ **do**
 $snr \leftarrow (E_b N_{0,up} + E_b N_{0,low})/2;$
 $\sigma \leftarrow \sqrt{\frac{1}{2R \times 10^{snr/10}}};$
 $u_{ch} \leftarrow \frac{2}{\sigma^2};$
 $p_{ch} \leftarrow Q(\frac{u_{ch}}{\sigma});$
 $w \leftarrow \mathbf{0}_{1 \times k_{max}};$
 for $k \leftarrow 1$ **to** k_{max} **do**
 if $k = 1$ **then**
 $x_r, x_c \leftarrow p_{ch}$
 else
 $x_r \leftarrow \text{equation(6)};$
 $x_c \leftarrow \text{equation(6) with } \tilde{x}_c^{k-1} \text{ replace } \tilde{x}_c^{k-1} \text{ and } w_c^k \text{ replace } w_c^{k-1};$
 end
 calculate $f^c(x_r), f^e(x_r), w_r(k)$ according to equations(4,5,7);
 calculate $f^c(x_c), f^e(x_c), w_c(k)$ according to equations(4,5,7);
 $\mu_m \leftarrow \text{equation(12)}$
 end
 $x_r;$
 if $x_r \leq 10^{-8}$ **then**
 $E_b N_0, E_b N_{0,up} \leftarrow snr;$
 $W_r \leftarrow w_r;$
 $W_c \leftarrow w_c;$
 else
 $E_b N_{0,low} \leftarrow snr$
 end
end

or

$$P^{-1}(i) = \begin{cases} 0 & \text{if } 0 \leq i \leq t-1 \\ \sum_{\delta=0}^{t-1} \sum_{j=0}^{\delta} \frac{h+1}{n} A_{h+1} F_h & \text{if } t \leq i \leq n-t-1 \\ 1 & \text{otherwise} \end{cases} \quad (3)$$

when it is initialized an erasure bit and there are i errors in the other $n-1$ bits. Here, the superscripts $+1$ and -1 refer to the correct and incorrect decodings due to the assumption that the input is all zeros. Clearly, the probability of this bit being erased is $P^e(i) = 1 - P^{-1}(i) - P^{+1}(i)$, where the definitions of $h, F_h,$ and A_h can be seen in [22].

In each decoding, we take the current decoded bit as an erasure and then determine how likely it is that this position was decoded correctly,

$$f^{+1}(x) = \sum_{i=0}^{n-1} \binom{n-1}{i} x^i (1-x)^{n-1-i} P^{+1}(i) \quad (4)$$

or incorrectly,

$$f^{-1}(x) = \sum_{i=0}^{n-1} \binom{n-1}{i} x^i (1-x)^{n-1-i} P^{-1}(i). \quad (5)$$

given that the input information has an error probability x .

Although only its hard decision is used in the practical decoding process, the soft input is used for the DE analysis. The soft input of a bit position in the row decoder is $l_r^k = l_{ch} + l_z^{k-1} + w_c^{k-1} u_c^{k-1}$.

Clearly, $L_r^k \sim \sum_{a \in \{\pm 1, 0\}} \Pr(U_c^k = a) \mathcal{N}(\mu + \mu_z^{k-1} + a \cdot w_c^{k-1}, 2(\mu + \mu_z^{k-1}))$. So the error probability \tilde{x}_r^k of the row input message is

$$\tilde{x}_r^k = P(l_r^k < 0) = f^{+1}(\tilde{x}_c^{k-1}) Q\left(\frac{\mu + \mu_z^{k-1} + w_c^{k-1}}{\sqrt{2(\mu + \mu_z^{k-1})}}\right) + f^{-1}(\tilde{x}_c^{k-1}) Q\left(\frac{\mu + \mu_z^{k-1} - w_c^{k-1}}{\sqrt{2(\mu + \mu_z^{k-1})}}\right) + (1 - f^{+1}(\tilde{x}_c^{k-1}) - f^{-1}(\tilde{x}_c^{k-1})) Q\left(\sqrt{\frac{\mu + \mu_z^{k-1}}{2}}\right) \quad (6)$$

With \tilde{x}_r^k , we can determine the scaling factor for the row component decodings at the k th iteration as

$$w_r^k = \log \frac{f^{+1}(\tilde{x}_r^k)}{f^{-1}(\tilde{x}_r^k)}. \quad (7)$$

Similarly, from the soft input $l_c^k = l_{ch} + l_z^{k-1} + w_r^k u_r^k$ of a column decoder, we can obtain the error probability \tilde{x}_c^k of the input information and the scaling factor w_c^k for the column component decodings at the k th iteration, respectively.

For the z -component decoders, we consider an SPC code of length s . Using the sum-product algorithm to decode, the extrinsic message is given as

$$\tanh \frac{m}{2} = \prod_{i=1}^{s-1} \tanh \frac{l_{z(i)}^k}{2} \quad (8)$$

where m is the extrinsic message and $l_{z(i)}^k$ are $s - 1$ input messages. It is worth noting that the calculation of w is performed offline and does not increase the decoding complexity. It is well known that, in the DE for LDPC codes, all messages in the above equation can be approximated to Gaussian distributed with the consistency condition (we say consistent Gaussian in the following). However, because of $l_z^k = l_{ch} + w_r^k u_r^k + w_c^k u_c^k$, where w_r^k and w_c^k are 2 constants, the messages in the right-hand side of Eq. 8 are obviously not consistent Gaussian. In this case, we still assume that the message m in the left-hand side of Eq. 8 approximately satisfies the consistency condition. This assumption is verified in the experimental results in Fig. 3. The red curves indicate the actual distribution of m obtained by simulation according to Eq. 8, and the blue curves indicate the distribution curve obtained by assuming that the left-hand of Eq. 8 for m still satisfies the consistent Gaussian. The solid, dashed, and dotted lines are obtained at $\{P^{+1}(i), P^{-1}(i), P^e(i)\} = \{0.7, 0.2, 9, 0.01\}$, $\{0.55, 0.435, 0.015\}$ and $\{0.3, 0.68, 0.02\}$, respectively.

With the Gaussian approximation of m , we proceed to calculate its expected value. Taking the expected value of both sides, it follows that

$$\mathbb{E} \left[\tanh \frac{M}{2} \right] = \prod_{i=1}^{s-1} \mathbb{E} \left[\tanh \frac{L_{z(i)}^k}{2} \right]. \quad (9)$$

Now, we need to determine the distribution of $L_{z(i)}^k$, which is the random variable representation of $l_{z(i)}^k$. Since $l_z^k = l_{ch} + w_r^k \mu_r^k + w_c^k \mu_c^k$, L_z^k will be a Gaussian variable with the density $\mathcal{N}(\mu + w_r^k \mu_r^k + w_c^k \mu_c^k, 2\mu)$ and $\{\mu_r^k, \mu_c^k\} \in \{\pm 1, 0\}^2$. Define $\mu_{(\alpha, \beta)} = \mu + w_r^k \alpha + w_c^k \beta$, where $\alpha, \beta \in \{\pm 1, 0\}^2$. We rewrite the expectation in the right of Eq. 8 as

$$\begin{aligned} \mathbb{E} \left[\tanh \left(\frac{L_{z(i)}^k}{2} \right) \right] &= \mathbb{E} \left[\tanh \left(\frac{l_{ch} + w_r^k U_r^k + w_c^k U_c^k}{2} \right) \right] \\ &= \sum_{\{\alpha, \beta\} \in \{\pm 1, 0\}^2} \Pr\{U_r^k = \alpha, U_c^k = \beta\} \cdot E_L \left[\tanh \left(\frac{l_{ch} + w_r^k \alpha + w_c^k \beta}{2} \right) \right] \\ &= \sum_{\{\alpha, \beta\} \in \{\pm 1, 0\}^2} \Pr\{U_r^k = \alpha, U_c^k = \beta\} \int_{-\infty}^{+\infty} \tanh \left(\frac{\tau + w_r^k \alpha + w_c^k \beta}{2} \right) \mathcal{N}(\mu_{(\alpha, \beta)}, 2\mu) d\tau. \end{aligned} \quad (10)$$

Define

$$\Psi(\mu; w_r^k; w_c^k) \triangleq 1 - \mathbb{E} \left[\tanh \left(\frac{L_{z(i)}^k}{2} \right) \right] \quad (11)$$

We can write that $1 - \Phi(\mu_m) = [1 - \Psi(\mu; w_r^k; w_c^k)]^{s-1}$, and it follows that

$$\mu_m = \Phi^{-1} \left(1 - [1 - \Psi(\mu; w_r^k; w_c^k)]^{s-1} \right). \quad (12)$$

$\Phi^{-1}(\cdot)$ can be computed by a numerical search since $\Phi(\cdot)$ is monotonically decreasing.

The complete analytical calculation can be seen in Algorithm 2. First, the error probability of the input of row component decoders

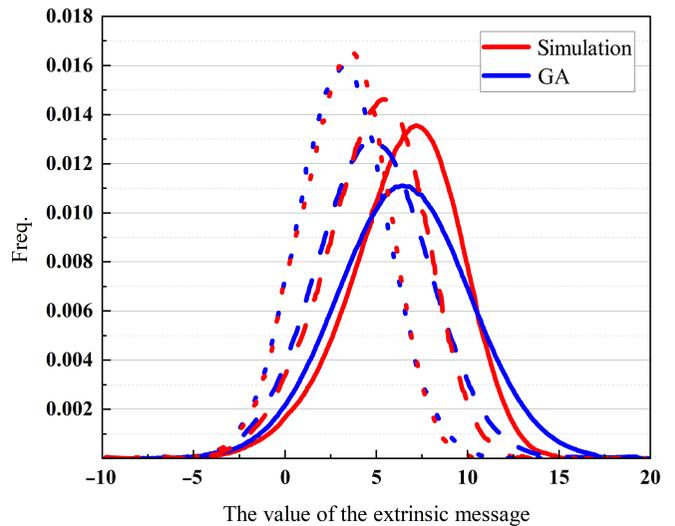


Fig. 3. The distribution of variable m in Eq. 8.

is calculated via Eq. 6 with $x_r^1 = P(l_r^k < 0) = P(l_{ch} < 0)$, after which the correct and error probabilities of BCH decoding can be calculated via Eqs. 4 and 5, then the factor of the iterative row decoders can be derived from Eq. 7. With w_r^1 and U_r^1 , the error probability of the input of column component decoders can be obtained analogously to Eq. 6 with $\mu_z = 0$. Repeat the above operation to obtain w_c^k and U_c^k . In the z -component decoders, the mean value μ_m of the soft information output by MSA is calculated by Eqs. 10 and 12. After that, move on to the next iteration. The set of scaling factors used is such that the error probability of input for row component decoders after 10 times iterative operations is less than 10^{-4} . There is $w_r^k = w_c^k \triangleq w^k$ in the simulation process of this article because the decoding process is decoded in parallel in multiple dimensions, i.e., the column dimension and the SPC dimension use the output information of the previous iteration.

Results

Complexity analysis

Storage complexity comparison

Consider a $\mathcal{P}(n^2 \times s, r^2 \times \frac{s-1}{s})$ decoded over k_{max} iterations.

Using q bits for quantization of L_{ch} , L_z , w_r^k , and w_c^k ($w_r^k = w_c^k$),

the additional memory required for iBDDSR-MSA compared to iBDD is $(k_{max} + 2n^2s)q/(s-1) \approx 2n^2q$ bits, which just twice as much as iBDD-SR. It is worth noting that since only the current l_z is used in each iteration, new data can directly overwrite the original data without the need for additional memory units, and only a slight of bit computation is added by MSA. In iBDDSR-MSA, only binary messages are passed between component decoders. Everything else is done within the component decoders. In practice, the prior information used in each iteration is generated by the previous iteration of the component decoder. This means that, compared to iBDD-SR, the present method does not have extra decoding delay and increase in data flow. The proposed scheme is therefore more suitable for high-throughput systems.

Coding complexity comparison

We measure the coding complexity in terms of the number of binary operations per bit, which is called the normalized coding complexity, denoted by $\overline{\mathcal{N}}$.

For the coding of a (n, k, t) BCH code, the number of operations is $\mathcal{N}_{BCH} = k(n - k)$. Then, the number of operations of the encoding process for BCH-TPC can be obtained by multiplying \mathcal{N}_{BCH} by the number of BCH codes, that is, $\mathcal{N}_{BCH-TPC} = 2nk(n - k)$ and $\overline{\mathcal{N}}_{BCH-TPC} = 2n(n - k) / k$.

We then consider the number of the operations of an SCC with a matrix size of $\binom{n}{2}, n$ and a staircase number of m , which is theoretically infinite. The total number of binary operations and normalized coding complexity are $\mathcal{N}_{SCC} = \binom{mn - \frac{n}{2}}{2} \times k(n - k)$ and $\overline{\mathcal{N}}_{SCC} = \frac{(2m-1)k(n-k)}{m(2k-n)} \sim \frac{2k(n-k)}{2k-n}$. There will be a slight difference between k here and the k in BCH-TPC, but it can basically be ignored at higher bit rates. When $k/n > 0.5$, the complexity of SCC is always greater than that of BCH-TPC.

For the encoding scheme we proposed, assuming that the SPC length is $c \triangleq s + 1$, then $\overline{\mathcal{N}}_{BCH^2SPC} = \frac{2nk(n-k)+cn^2}{c^2} = \frac{2n(n-k)}{k} + \frac{n^2}{k}$. The greater the code rate, the closer this value is to the $\overline{\mathcal{N}}_{BCH-TPC}$.

Decoding complexity comparison

We measure the computational complexity in terms of the number of operations per bit, including the real number computations (additions and comparisons) and the finite field computations (multiplications). Similarly, we have normalized the 2 computations, denoted by $\overline{\mathcal{N}}_{RE}$ (for real number) and $\overline{\mathcal{N}}_{GF}$ (for finite field), respectively.

It is assumed that the decodings of component codes are all carried out in parallel. When the number of iterations is the same and the code lengths of the BCH component codes are equal, the overall delay of decoding can be approximately considered to be the same. Here, we only consider the computational complexity in the case of these assumptions.

When using the BDD decoding of BCH codes, ignoring the finite field additions, and only measuring by the number of finite field multiplications, the decoding complexity is $\mathcal{N}_{GF,BCH} \approx 2t^2 + t(2n - 1)$. Defining a window size of SCC as W_s and the iteration number as I , then the decoding complexity of SCC can be calculated as

$$\overline{\mathcal{N}}_{GF,SCC} = \frac{\left[\frac{n}{2}m + \frac{n}{2}(m-1)\right] \times [2t^2 + t(2n-1)] \times I}{\frac{n}{2} \times \frac{n}{2} \times R} \approx \frac{2[2t^2 + t(2n-1)] \times I}{nR} 2W_s \tag{13}$$

The decoding complexity of BCH^2SPC comes from 2 aspects, including the BDD decoding of BCH codes and the MSA decoding of SPC codes. Then, we can conclude that $\overline{\mathcal{N}}_{3D} = \overline{\mathcal{N}}_{GF,BCH} + \overline{\mathcal{N}}_{RE,SPC}$.

$$\overline{\mathcal{N}}_{GF,BCH} = \frac{[2n(c+1)] \times [2t^2 + t(2n-1)] \times I}{(c+1) \times n^2 \times R} = \frac{2[2t^2 + t(2n-1)] \times I}{nR} \tag{14}$$

It was mentioned in [23] that the normalized computational complexity for the MSA decoding of LDPC codes is $\overline{\mathcal{N}}_{RE,LDPC} \approx 3\rho(N' - K')I$, where ρ , N' , and K' are the density of parity-check matrix, the code blocklength (includes punctured/shortened bits), and the length of information bits, respectively. Then, we can obtain that $\overline{\mathcal{N}}_{RE,SPC} \approx 3I$. From this, it follows that $\overline{\mathcal{N}}_{3D}$ is almost independent of c and $\overline{\mathcal{N}}_{3D} \ll \overline{\mathcal{N}}_{SCC}$.

Throughput comparison

The throughput rate is defined as the number of output bits/decoding time. Here, we use the number of operations of complex calculations as time, so it is defined as the number of output bits/number of operations. At this time, it can be considered that it has an inverse relationship with the computational complexity of the ‘‘Decoding complexity comparison’’ section. Thus, the proposed scheme has lower computational complexity and delay. If we only focus on the number of output bits of an iteration rather than the computational complexity, since each position of SCC will be iterated $W_s \times I$ times, the throughput rate of SCC will be $\frac{1}{m}$ of BCH^2SPC .

Numerical results

We consider an iterative decoder with a maximum of 12 iterations, consisting of 10 times iBDDSR-MSA, followed by 2 iterations for iBDD-MSA. To compare with the performance in [22,28], we choose $\mathcal{P}_{3D}^1 (255^2 \times 13, 0.81)$ and $\mathcal{P}_{3D}^2 (511^2 \times 28, 0.90)$ BCH^2SPC product codes, respectively. The DE-optimized scaling factors for \mathcal{P}_{3D}^1 and \mathcal{P}_{3D}^2 over 10 iterations are listed in Table at $E_b/N_0 = 3.975$ and 4.710 dB, respectively. Considering that we make approximations during the calculation and that the first iteration should have little impact on the overall iterative decoding, we set different initial iteration factors and find that the best performance is achieved when we set $w^1 = 0.5$ under DE algorithm, which can be seen in Fig. 4. On the basis of experience, we also select 3 other sets of factors and find that the DE algorithm we proposed consistently lead to better scaling factors. In the subsequent simulations, we also use the optimized factors in Table.

In Figs. 5 and 6, we show the performance of iBDDSR-MSA, iterative bounded distance decoding with combined reliability (iBDD-CR) [28], iBDD-SR [22], and anchor decoding (AD) proposed in [31]. All the simulations are implemented over the bi-AWGN channel. In addition, the thresholds are represented by dashed lines. Further, in Fig. 7, we can see that the proposed decoding method can achieve a performance gain of 0.35 dB compared to AD and 0.3 and 0.25 dB compared to iBDD-SR and iBDD-CR at bit error rate (BER) = 10^{-6} , respectively.

In Fig. 6, we can see that the proposed method still has a performance gain of 0.15 and 0.12 dB compared to iBDD-SR and iBDD-CR of 2D code, respectively. We also note that compared to SCCs with higher latency and greater computational complexity, the proposed scheme also has slightly performance gain. In addition, it is worth noting that our proposed decoding algorithm uses parallel decoding in the simulation, i.e., different dimensions are decoded at the same time. This greatly improves the decoding speed, but it also leads to the decoding performance not being fully utilized. This is because the information from the previous iteration is used in the decoding process. If sequential decoding is used, its performance will be better.

Table. The evolution of w^k for the proposed 3D product codes $\mathcal{P}_{3D}^1, \mathcal{P}_{3D}^2$, and some other factors over 10 iterations

Different methods	w^2	w^3	w^4	w^5	w^6	w^7	w^8	w^9	w^{10}
$DE(\mathcal{P}_{3D}^1), w^1 = 2.0$	3.7	4.1	4.5	4.8	5.3	5.8	6.5	7.8	10.6
$DE(\mathcal{P}_{3D}^1), w^1 = 0.5$	3.7	4.1	4.5	4.8	5.2	5.8	6.5	7.8	10.6
$DE(\mathcal{P}_{3D}^2), w^1 = 0.5$	4.4	4.8	5.2	5.5	5.9	6.4	7.1	8.3	10.7
Constant, $w^1 = 0.5$	2	2	2	2	2	2	2	2	2
Increase ₁ , $w^1 = 0.5$	1	1.5	2	2.5	3	5	8	9	10
Increase ₂ , $w^1 = 0.5$	1	1	2	2	3	3	3	5	10

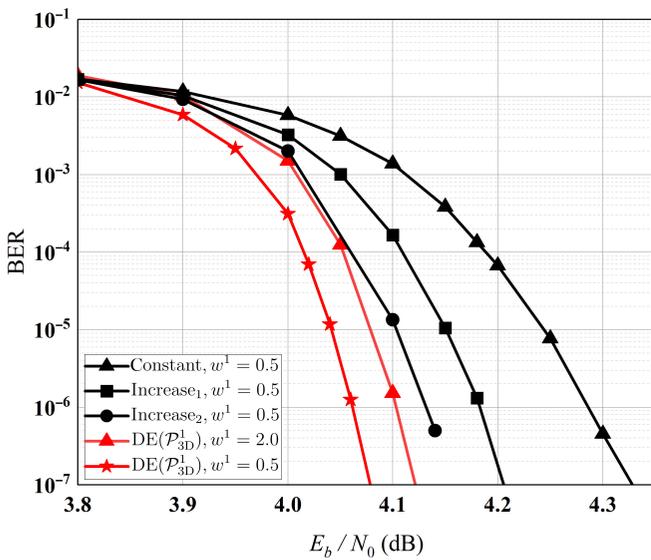


Fig. 4. Performance of iBDDSR-MSA for \mathcal{P}_{3D}^1 under different factors.

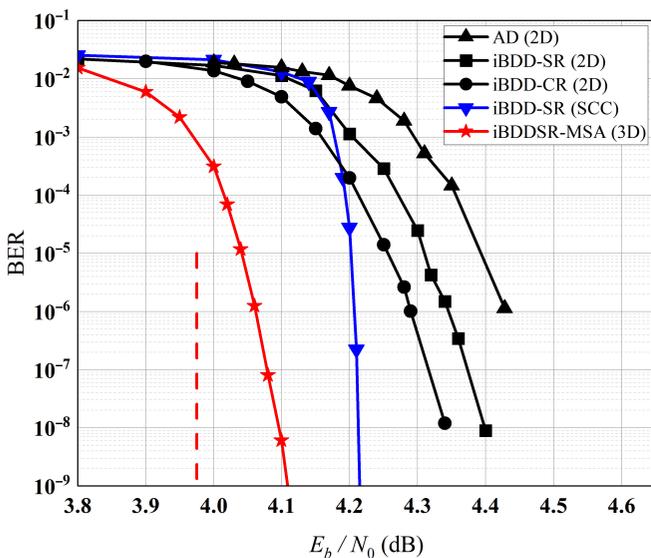


Fig. 5. Performance of iBDDSR-MSA for \mathcal{P}_{3D}^2 ($255^2 \times 13.0.81$) BCH²SPC, AD, iBDD-SR, and iBDD-CR for a \mathcal{P}_{2D}^2 ($255^2, 0.82$) product code and iBDD-SR for a \mathcal{P}_{SCC}^1 ($254, 0.81$) SCC that has a window size of 7.

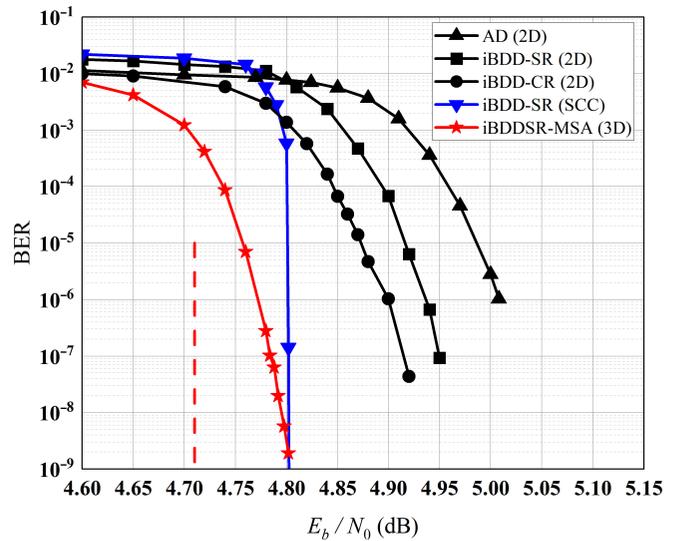


Fig. 6. Performance of iBDDSR-MSA for \mathcal{P}_{3D}^3 ($511^2 \times 28.0.9$) BCH²SPC, AD, iBDD-SR, and iBDD-CR for a \mathcal{P}_{2D}^2 ($511^2, 0.90$) product code and iBDD-SR for a \mathcal{P}_{SCC}^2 ($510, 0.89$) SCC that has a window size of 7.

In Fig. 7A and B, we also carry out simulations for the 3 coding methods with the same total code length and equal code rate. The total code lengths in Fig. 7A and B are approximately 65,000 and 260,000 bits, respectively, and the total code rates are 0.52 and 0.62, respectively. It can be seen from the simulation results that the method we proposed has more reliable decoding capabilities. Under BER = 10⁻⁶, the 3D decoding method we proposed can achieve performance gains of 0.4 to 0.7 dB and 0.8 to 0.9 dB, respectively, compared to the iBDD-SR decoding algorithm of 2D product codes and SCC. At the same time, the decoding method we proposed is very suitable for decoding long blocklength codes. Through the 3D construction and parallel decoding algorithm, our scheme can achieve lower latency and higher reliability.

Figure 8 shows the BDD decoding results of the BCH component codes used in Fig. 7, and it is clear that our construction obtains overall better performance in the case of poor individual BCH decoding. At the same time, shorter BCH component codes imply lower decoding delay, which is exactly what we want.

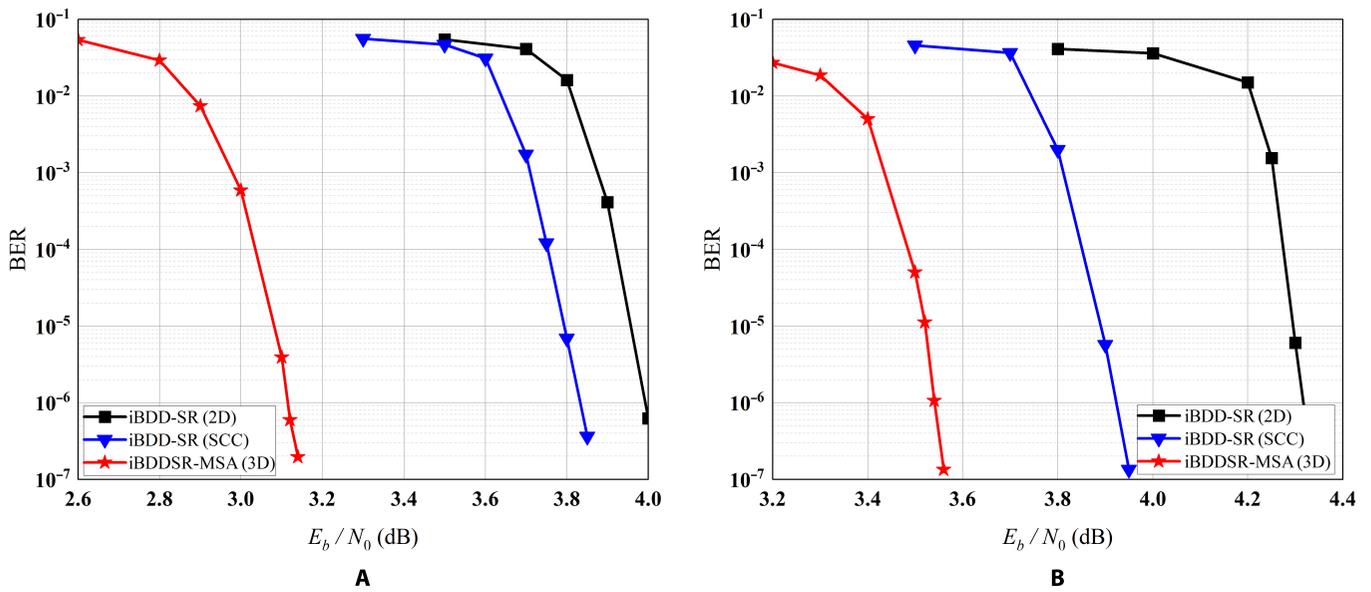


Fig. 7. Performance comparison of 3D BCH²SPC, 2D BCH product codes, and 2D SCC with equal total code length and bit rate. (A) and (B) represent the 2 groups. The 3 encodings in (A) are $\mathcal{P}_{33}^3(127^2 \times 4.052)$, $\mathcal{P}_{25}^3(255^2, 0.54)$, and $\mathcal{P}_{\text{SCC}}^3(126^2, 0.55)$; and the 3 encodings in (B) are $\mathcal{P}_{33}^4(255^2 \times 4.062)$, $\mathcal{P}_{25}^4(511^2, 0.62)$, and $\mathcal{P}_{\text{SCC}}^4(254^2, 0.62)$. The decoding window size of SCC is 8.

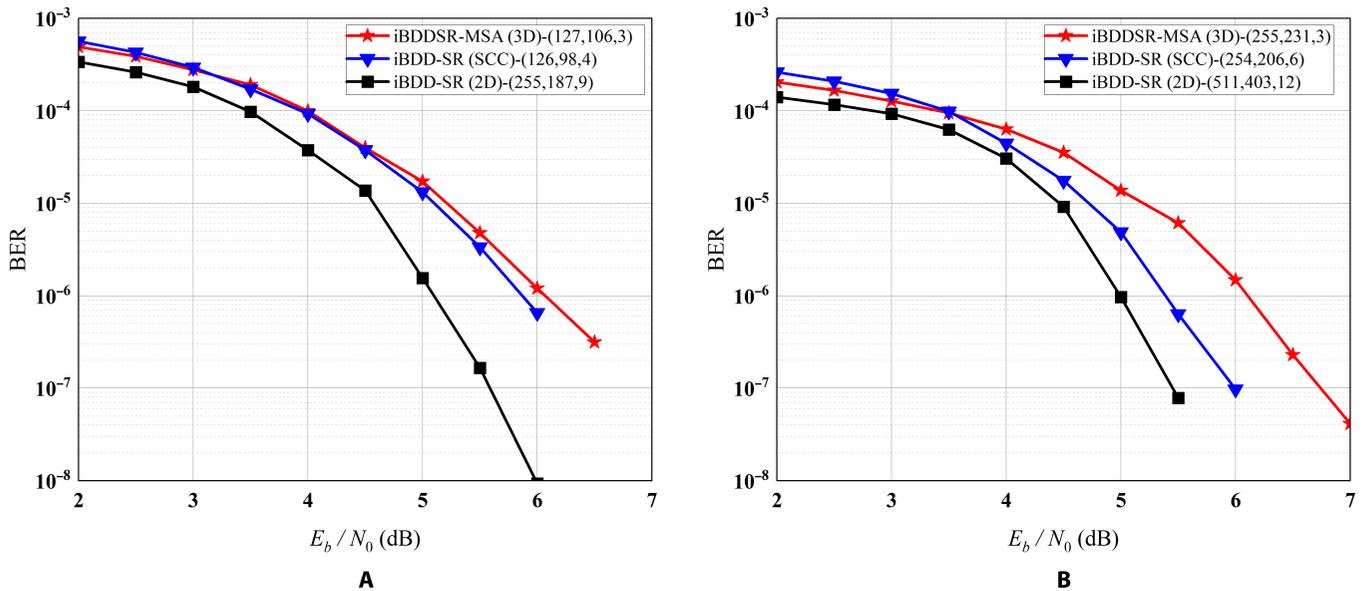


Fig. 8. BDD decoding result of the component code of Fig. 7.

Conclusion

This paper explores the use of BCH²SPC concatenated coding technique and its reduced-complexity decoding scheme in satellite communications. The results show that this method offers substantial improvements in error correction performance with less data flow and computational complexity, making it a promising approach for use in satellite communication systems. Specifically, the decoding method we proposed outperforms similar message passing decodings of 2D BCH-TPC codes and SCCs by approximately 0.3 and 0.15 dB, respectively, which enhances the reliability and robustness of the communication system. These

findings suggest that the BCH²SPC-concatenated coding has great potential for applications in satellite communications.

Acknowledgments

Funding: This work was supported by Jiangsu Province Basic Research Project under grant BK20192002, the National Natural Science Foundation of China under grant no. 62331002, and the Southeast University China Mobile Research Institute Joint Innovation Center.

Author contributions: Y.X. contributed to the study design and the overall draft of the text. M.J. contributed to the idea,

revisions, and suggestions. M.Z. and J.W. contributed to the structure of the draft, modifications, and suggestions. N.H. and C.Z. contributed to reviewing, editing and reference searching. **Competing interests:** The authors declare that they have no competing interests.

Data Availability

The data are available from the authors upon a reasonable request.

References

- Kato N, Fadlullah ZM, Tang F, Mao B, Tani S, Okamura A, Liu J. Optimizing space-air-ground integrated networks by artificial intelligence. *IEEE Wirel Commun.* 2019;26(4):140–147.
- Kong H, Lin M, Zhu W-P, Amindavar H, Alouini M-S. Multiuser scheduling for asymmetric FSO/RF links in satellite-UAV-terrestrial networks. *IEEE Wireless Commun Lett.* 2020;9(8):1235–1239.
- Guo K, Lin M, Zhang B, Wang JB, Wu Y, Zhu WP, Cheng J. Performance analysis of hybrid satellite-terrestrial cooperative networks with relay selection. *IEEE Trans Veh Technol.* 2020;69:9053–9067.
- Guo K, Li X, Alazab M, Jhaveri RH, An K. Integrated satellite multiple two-way relay networks: Secrecy performance under multiple eaves and vehicles with non-ideal hardware. *IEEE Trans Intell Veh.* 2023;8(2):1307–1318.
- Sweeting MN. Modern small satellites-changing the economics of space. *Proc IEEE.* 2018;106(3):343–361.
- Lin X, Hofström B, Wang YPE, Masini G, Maattanen HL, Rydén H, Sedin J, Stattin M, Liberg O, Euler S, et al. 5G new radio evolution meets satellite communications: Opportunities, challenges, and solutions. In: *5G and beyond: Fundamentals and standards*. Cham: Springer International Publishing; 2021. p. 517–531.
- Roddy D. *Satellite communications*. New York (NY): McGraw-Hill Education; 2006.
- Zhang C, Mu X, Yuan J, Li H, Bai B. Construction of multi-rate quasi-cyclic LDPC codes for satellite communications. *IEEE Trans Commun.* 2021;69(11):7154–7166.
- Arikan E. Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels. *IEEE Trans Inf Theory.* 2009;55(7):3051–3073.
- Koike-Akino T, Cao C, Wang Y. Turbo product codes with irregular polar coding for high-throughput parallel decoding in wireless OFDM transmission. Paper presented at: 2018 IEEE International Conference on Communications (ICC); 2018 May 20–24; Kansas City, MO, USA.
- Leven A, Schmalen L. Status and recent advances on forward error correction technologies for lightwave systems. *J Lightw Technol.* 2013;32(16):2735–2750.
- Matsushita A, Nakamura M, Hamaoka F, Okamoto S. Transmission technologies beyond 400-Gbps/carrier employing high-order modulation formats. Paper presented at: 2018 European Conference on Optical Communication (ECOC); 2018 Sep 23–27; Rome, Italy.
- Barakatain M, Lentner D, Böcherer G, Kschischang FR. Performance-complexity tradeoffs of concatenated FEC for higher-order modulation. *J Lightw Technol.* 2020;38(11):2944–2953.
- Sheikh A, Amat AGI, Liva G. Achievable information rates for coded modulation with hard decision decoding for coherent fiber-optic systems. *J Lightw Technol.* 2017;35(23):5069–5078.
- Elias P. Error-free coding. *Trans IRE Prof Group Inf Theory.* 1954;4(4):29–37.
- Pyndiah R, Glavieux A, Picart A, Jacq S. Near-optimum decoding of product codes: Block turbo codes. Paper presented at: 1994 IEEE GLOBECOM. Communications: The Global Bridge; 1994 Nov 28–Dec 02; Francisco, CA, USA.
- Miao S, Rapp L, Schmalen L. Improved soft-aided decoding of product codes with dynamic reliability scores. *J Lightw Technol.* 2022;40(22):7279–7288.
- Jian Y-Y, Pfister HD, Narayanan KR, Rao R, Mazahreh R. Iterative hard-decision decoding 361 of braided BCH codes for high-speed optical communication. Paper presented at: 2013 IEEE Global Communications Conference (GLOBECOM); 2013 Dec 9–13; Atlanta, GA, USA.
- Sheikh A, Amat AGI, Liva G, Steiner F. Probabilistic amplitude shaping with hard decision decoding and staircase codes. *J Lightw Technol.* 2018;36(9):1689–1697.
- Lechner G, Pedersen T, Kramer G. Analysis and design of binary message passing decoders. *IEEE Trans Commun.* 2012;60(3):601–607.
- Sheikh A, Amat AGI, Liva G. Iterative bounded distance decoding of product codes with scaled reliability. Paper presented at: 2018 European Conference on Optical Communication (ECOC); 2018 Sep 23–27; Rome, Italy.
- Sheikh A, Amat AGI, Liva G. Binary message passing decoding of product-like codes. *IEEE Trans Commun.* 2019;67(12):8167–8178.
- Zhu M, Jiang M, Zhao C. Ternary message passing decoding of RS-SPC product codes. Paper presented at: 2022 IEEE International Symposium on Information Theory (ISIT); 2022 Jun 26–Jul 01; Espoo, Finland.
- Smith BP, Farhood A, Hunt A, Kschischang FR, Lodge J. Staircase codes: FEC for 100 Gb/s OTN. *J Lightw Technol.* 2011;30(1):110–117.
- Qiu M, Yuan J. Sub-block rearranged staircase codes. *IEEE Trans Commun.* 2022;70:5695–5710.
- Rapp L, Schmalen L. Error-and-erasure decoding of product and staircase codes. *IEEE Trans Commun.* 2022;70(1):32–44.
- Miao S, Rapp L, Schmalen L. Improved soft-aided decoding of product codes with adaptive performance-complexity trade-off. Paper presented at: 2022 European Conference on Optical Communication (ECOC); 2022 Sep 18–22; Basel, Switzerland.
- Sheikh A, Amat AGI, Liva G, Alvarado A. Refined reliability combining for binary message passing decoding of product codes. *J Lightw Technol.* 2021;39(15):4958–4973.
- Lei Y, Chen B, Liga G, Balatsoukas-Stimming A, Sun K, Alvarado A. A soft-aided staircase decoder using three-level channel reliabilities. *J Lightw Technol.* 2021;39(19):6191–6203.
- Deng L, Guan YL, Liu Z, Shi Z, Zhang Z. BCH-SPC based cubic turbo product coding. Paper presented at: 2022 10th International Workshop on Signal Design and Its Applications in Communications (IWSDA); 2022 Aug 1–5; Colchester, UK.
- Häger C, Pfister HD. Approaching miscorrection-free performance of product codes with anchor decoding. *IEEE Trans Commun.* 2018;66(7):2797–2808.