

## Research Article

# A Novel Scheme of Product Polar Codes with High Efficiency and Rate Compatibility

Mohammad Abu Hanif  and Sina Vafi

College of Engineering, IT and Environment, Charles Darwin University, Australia

Correspondence should be addressed to Mohammad Abu Hanif; [mohammad.hanif@cdu.edu.au](mailto:mohammad.hanif@cdu.edu.au)

Received 7 May 2019; Revised 21 October 2019; Accepted 10 December 2019; Published 13 April 2020

Academic Editor: Massimo Condoluci

Copyright © 2020 Mohammad Abu Hanif and Sina Vafi. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a new structure of product codes formed by combining two polar codes. The encoding performance of these codes is verified by implementing an exhaustive search algorithm, which determines their minimum weight specifications. Conducted analysis and simulations confirm that with the equal code length and rate, the newly proposed codes outperform the conventional polar codes in high energy per bit-to-noise ratios ( $E_b/N_0 \leq 5$  dB). This is concluded from punctured and nonpunctured product codes.

## 1. Introduction

Polar codes are promising forward error correction (FEC) codes for the next generation of broadband networks including the fifth generation of wireless networks (5G). These codes are recognized as capacity-achieving codes under a successive cancellation (SC) decoding technique, which were initially constructed for the symmetric binary-input discrete memoryless channel (BI-DMC) [1]. However, at a short code length, their bit error rate (BER) performance is not as reliable as other known codes, such as low-density-parity-check (LDPC) and turbo codes. This limits their reputation as a good code for a wide range of applications. In order to overcome this issue, short length codes are combined with other codes to form a scheme of high-performance concatenated codes [2].

A serially concatenated code was proposed based on an inner polar code and an outer Reed-Solomon (RS-polar) code [3]. It is shown that the block error rate performance of the code is improved at the expense of increasing the length and complexity of the RS code. Eslami and Pishro-Nik showed that a high-performance concatenated code can be formed by combining polar and LDPC codes [4]. In [5], Tal and Vardy considered concatenation of a polar code with an outer cyclic redundancy check (CRC) code. The CRC

code together with the usage of a list decoding algorithm concludes a high-performance code formed by the short length polar code. An alternative scheme of these codes is achieved, when the CRC code is replaced by the parity-check code [6]. In this case, parity-check codes having low multiplicity in their minimum weight are selected [7].

On the other hand, construction of a parallel concatenated code formed by two polar codes was proposed in [8]. Based on the same code length and rate, the designed code outperforms the single one. More recent work demonstrates a parallel construction of multiple polar codes having different rates [9]. Constructed code has high performance at the expense of high complexity. Alternatively, a parallel concatenated code was presented by combining a polar code with a Recursive Systematic Convolutional (RSC) code [10]. Codes obtained by this method conclude a performance better than other well-known codes (turbo codes and LDPC codes), mainly at the error floor region due to utilizing a modified interleaver, which is matched with specifications of the constituent RSC code.

The main similarity between the above-mentioned codes is the serial combination of a polar code with a nonpolar code. On the other hand, a turbo product code is based on multiple polar codes (polar-TPC) introduced in [11]. In this work, nonpunctured constituent polar codes are used to

form a product code, where SCL-Chase decoding algorithm was proposed.

This paper presents a new scheme of serially concatenated codes constituted by two systematic polar codes linked by an interleaver. Codes are constructed on the basis of the message, whose length is the product of the length of constituent codes. This code can be also introduced as a polar code, as every bit of the codeword is properly polarized. Moreover, the puncturing structure of these codes will be verified. Unlike a punctured single polar code, which applies a relatively high number of punctured bits, punctured product polar codes can be constructed by removing a low number of bits applied in one of the constituent codes. This feature will allow decoders to interactively exchange their decoded information with each other and improve the error correcting capability of the code. Conducted simulations confirm that with the same rate and code length, punctured product polar codes have better performance than single punctured polar codes. In addition, it is analysed that the proposed code decoded with BP decoding has similar performance to codes decoded with the SCL-Chase-based algorithm.

The rest of the paper is organized as follows. A brief introduction of polar codes is given in Section 2. In Section 3, the structure of product polar codes and their comparison with single polar codes formed by the same rate and length are explained. Puncturing technique of the proposed code is presented in Section 4. Numerical results are given in Section 5. Finally, Section 6 concludes the paper.

## 2. Basic Structure of Polar Codes

Let  $N = 2^n$  ( $n \in \mathbb{Z}, n > 0$ ) and  $K$  be the length and dimension of a  $(N, K)$  polar code, respectively. The rate of this code is  $R = K/N$ . Let  $\mathbf{G}_N = \mathbf{G}_2^{\otimes \log(N)}$  be the generator matrix of the polar code with rate one and length  $N$ , where  $(\cdot)^{\otimes q}$  denotes

the  $q$ th Kronecker power and  $\mathbf{G}_2 := \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$  as in [1]. It is

shown that as  $N$  approaches to infinity, under successive cancellation (SC) decoding, a fraction of rows of  $\mathbf{G}_N$  leads to good channels suitable for carrying messages with length  $K$ . Other channels are referred to as bad channels. In order to measure how good a binary-input channel  $W$  is, Arikan used the Bhattacharyya parameter of  $W$ , which is defined as [1]

$$Z(W) = \sum_{y \in Y} \sqrt{W(y|x=0)W(y|x=1)}, \quad (1)$$

where  $W(Y|X)$  is the channel transitional probability of input alphabet  $X = \{0, 1\}$  and output alphabet  $Y, x \in X, y \in Y$ . For binary erasure channel (BEC) with  $Z(W_1^{(1)}) = 0.5$ , the reliability of bit channels can be recursively determined by [1]

$$\begin{aligned} Z\left(W_N^{(2^{i-1})}\right) &= 2Z\left(W_{N/2}^{(i)}\right) - Z\left(W_{N/2}^{(i)}\right)^2, \\ Z\left(W_N^{(2^i)}\right) &= Z\left(W_{N/2}^{(i)}\right)^2, \end{aligned} \quad (2)$$

where  $Z(W_N^{(i)})$ ,  $1 \leq i \leq N$ , is the Bhattacharyya parameter of channel  $W_N^{(i)}$ . The code construction rule (3) is channel-independent [12]. This means, the same rule can be used for constructing codes for any binary-input channel [12]. However, it shows that for the additive white Gaussian noise (AWGN) channel, with zero mean and variance  $N_0/2$ , the code works better if equation (2) is initialized by [13–15]

$$Z\left(W_1^{(1)}\right) = \int_{-\infty}^{\infty} \sqrt{W(y|x=-1)W(y|x=1)} = \exp\left(-\frac{E_c}{N_0}\right), \quad (3)$$

where  $E_c = RE_b$  and  $E_b$  is the energy spent per bits. In this case, the coded bits ( $\mathbf{x}$ ) are modulated by binary phase shift keying (BPSK) modulation. Hence, channels with low  $Z(W)$  values are referred to as noiseless channels (or good channels).

Let  $\mathcal{A}_g$  be the set of good channels. Similarly,  $\mathcal{A}_f^c$  denotes the frozen set, which specifies the set of bad channels. Let  $\mathbf{u} = [u_1 u_2 \cdots u_N]$  be a message vector of length  $N$ . In this vector, there are  $K$  message bits ( $\mathbf{u}_{\mathcal{A}_g}$ ) and  $N - K$  frozen bits ( $\mathbf{u}_{\mathcal{A}_f^c} = 0$ ). The codeword vector is generated by  $\mathbf{x} = \mathbf{u}\mathbf{G}_N$ , where  $\mathbf{G}_N$  is a  $N \times N$  generator matrix.

In systematic polar encoding, the codeword is split into two parts [16]. This can be written as  $\mathbf{x} = (\mathbf{x}_{\mathcal{B}} \mathbf{x}_{\mathcal{B}^c})$ , where  $\mathcal{B}$  is an arbitrary subset of  $\{1, 2, \dots, N\}$ . These parts are recognized as message and parity parts of the codeword, respectively. In this case,

$$\mathbf{x}_{\mathcal{B}} = \mathbf{u}_{\mathcal{A}_g} \mathbf{G}_{\mathcal{A}_g \mathcal{B}} + \mathbf{u}_{\mathcal{A}_f^c} \mathbf{G}_{\mathcal{A}_f^c \mathcal{B}}, \quad (4)$$

$$\mathbf{x}_{\mathcal{B}^c} = \mathbf{u}_{\mathcal{A}_g} \mathbf{G}_{\mathcal{A}_g \mathcal{B}^c} + \mathbf{u}_{\mathcal{A}_f^c} \mathbf{G}_{\mathcal{A}_f^c \mathcal{B}^c}, \quad (5)$$

where  $\mathbf{G}_{\mathcal{P}\mathcal{Q}}$  is a  $\mathcal{P} \times \mathcal{Q}$  generator matrix formed by taking rows of  $\mathcal{P}$  indices and columns of  $\mathcal{Q}$  indices of  $\mathbf{G}_N$ . In general, frozen bits are set to the value of zero. Hence,  $\mathbf{u}_{\mathcal{A}_f^c}$  is a zero vector and  $\mathbf{x}_{\mathcal{B}^c}$  can be calculated as

$$\mathbf{x}_{\mathcal{B}^c} = \mathbf{x}_{\mathcal{B}} \left( \mathbf{G}_{\mathcal{A}_g \mathcal{B}} \right)^{-1} \mathbf{G}_{\mathcal{A}_g \mathcal{B}^c}. \quad (6)$$

*2.1. Effect of the Minimum Weight on the Performance of Polar Code.* The minimum weight ( $d_{\min}$ ) of a systematic polar code is determined based on the weights of its rows, whose indices are elements of  $\mathcal{A}_g$ . This is given by [10, 17]

$$d_{\min} = \min_{b \in \mathcal{A}_g} \left\{ 2^{\omega(b)} \right\}, \quad (7)$$

where  $\omega(b)$  is the number of ones in the binary expansion of  $b$ . In polar codes, the lowest expected parity weight of the codeword is one [4]. This means that  $d_{\min}$  of the code is obtained from messages, whose weights are not greater than  $d_{\min} - 1$ . Based on this property and using equation (7), Table 1 gives minimum weight specifications of different polar codes. In this table,  $N_{d_{\min}}$  gives the multiplicity of the minimum weight determined by an exhaustive search algorithm, which considers all possible messages with weight no greater than  $d_{\min} - 1$  of the code.

As a class of linear block codes, performance of polar codes can be analysed by its Input-output weight enumerating function (IOWEF), which is given by [18]

$$A^C(W, H) = \sum_{\omega, h} A_{\omega, h} W^\omega H^h, \quad (8)$$

where  $A_{\omega, h}$  is the number of codewords with weight  $h$  obtained from messages with weight  $\omega$ . Based on IOWEF, the conditional weight enumerating function (CWEF) of the code is defined by

$$A_\omega^C(H) = \frac{1}{\omega!} \frac{\partial^\omega [A^C(W, H)]}{\partial W^\omega} \Big|_{W=0}. \quad (9)$$

This leads to expressing the upper bound of the probability of error for the code decoded by a maximum likelihood decoding technique as follows:

$$P_b(e) \leq \sum_{\omega=1}^k \frac{\omega}{k} A_\omega^C(H) \Big|_{H=e^{-RE_b/N_0}}, \quad (10)$$

where  $E_b/N_0$  is the signal-to-noise ratio per bit. The above bound can also be represented as

$$P_b(e) \leq \sum_{d_\Pi} D d_\Pi H^{d_\Pi} \Big|_{H=e^{-RE_b/N_0}}, \quad (11)$$

where

$$D_{d_\Pi} = \sum_{j+\omega=d_\Pi} \frac{\omega N_{\omega, j}}{K}. \quad (12)$$

Here,  $N_{\omega, j}$  gives the number of codewords generated by an input message with weight  $\omega$ , whose parity-check bits have weight  $j$ . With a good approximation, the above probability is given by

$$P_b(e) \approx \frac{1}{2} \sum_{d_\Pi} D_{d_\Pi} \operatorname{erfc} \left( \sqrt{d_\Pi \frac{RE_b}{N_0}} \right). \quad (13)$$

TABLE 1: Minimum weight specifications of polar codes with rates 1/2 and 1/4.

Polar code	$d_{\min} (N_{d_{\min}})$	Polar code	$d_{\min} (N_{d_{\min}})$
(8, 4)	4 (14)	(8, 2)	4 (2)
(16, 8)	4 (28)	(16, 4)	8 (14)
(32, 16)	4 (8)	(32, 8)	8 (12)
(64, 32)	8 (662)	(64, 16)	16 (364)
(128, 64)	8 (1724)	(128, 32)	16
(256, 128)	8 (2330)	(256, 64)	16
(512, 256)	8	(512, 128)	32
(1024, 512)	8	(1024, 256)	32
(2048, 1024)	16	(2048, 512)	64
(4096, 2048)	16	(4096, 1024)	64
(8192, 4096)	16	(8192, 2048)	64

Let  $P_{d_\Pi}(\gamma_b) = D_{d_\Pi} \operatorname{erfc}(\sqrt{d_\Pi \gamma_b})$ , where  $\gamma_b = RE_b/N_0$ . The relative contribution of each weight to the overall BER is given by

$$\bar{P}_{d_\Pi}(\gamma_b) = \frac{P_{d_\Pi}(\gamma_b)}{\sum_{d_\Pi} P_{d_\Pi}(\gamma_b)}. \quad (14)$$

*Example 1.* The IOWEF of (8, 4) systematic polar code is given by

$$A^{C_{sp}}(W, H) = 1 + 4WH^4 + 6W^2H^4 + 4W^3H^4 + W^4H^8. \quad (15)$$

The CWEF of this code is  $A_0^{C_{sp}}(H) = 1, A_1^{C_{sp}}(H) = 4H^4, A_2^{C_{sp}}(H) = 6H^4, A_3^{C_{sp}}(H) = 4H^4, A_4^{C_{sp}}(H) = H^8$ .

From (11) and (12), probability of error of the code is  $P_b(e) \approx \frac{4}{8} \operatorname{erfc} \left( \sqrt{4 \frac{RE_b}{N_0}} \right) + \frac{12}{8} \operatorname{erfc} \left( \sqrt{4 \frac{RE_b}{N_0}} \right) + \frac{12}{8} \operatorname{erfc} \left( \sqrt{4 \frac{RE_b}{N_0}} \right) + \frac{4}{8} \operatorname{erfc} \left( \sqrt{8 \frac{RE_b}{N_0}} \right) = 3.5 \operatorname{erfc} \left( \sqrt{4 \frac{RE_b}{N_0}} \right) + 0.5 \operatorname{erfc} \left( \sqrt{8 \frac{RE_b}{N_0}} \right)$ .

Considering (12) and CWEF of the code, it is evident that the effect of codewords with weight 4 on the performance of the code is much higher than that with weight 8.

Figure 1 shows the relative contribution of the three lowest weights of (16, 8) and (32, 16) polar codes. It is clearly observed that by increasing  $(E_b/N_0)$ s, the minimum weight has the highest effect on the performance of code. Figure 2 shows the relative contribution of low weights of (32, 8) and (64, 16) polar codes with  $R=1/4$ . The weight distributions of these short length codes conclude minimum weights with high multiplicities. Considering the probability of error given in equation (10), the existence of either the minimum weight with a high multiplicity

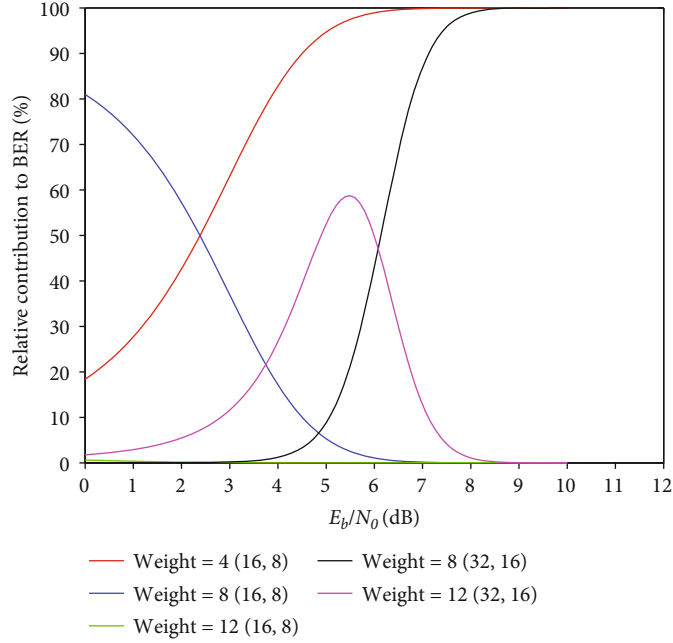


FIGURE 1: Relative contribution of different weights to the BER performance of (16, 8) and (32, 16) polar codes.

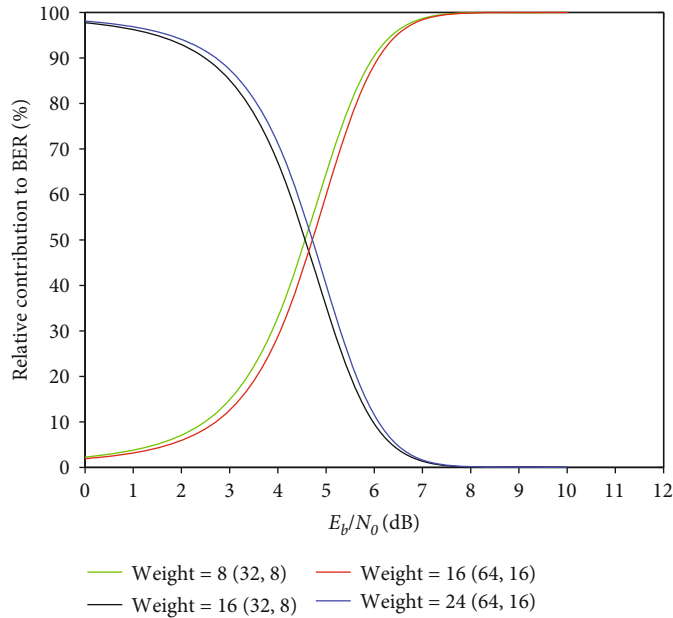


FIGURE 2: Relative contribution of different weights to the BER performance of (32, 8) and (64, 16) polar codes.

or a low minimum weight relative to the length of the codeword will deteriorate performance of the polar code. As shown in the figures, this is more sensible at the medium to high ( $E_b/N_0$ )s.

### 3. Structure of Product Polar Codes (PPCS)

Product codes are represented as one of the most well-known codes, whose high error-correcting capability is guaranteed due to their high minimum weight. Figure 3(a)

shows the basic structure of product codes. In this code, every message with length  $K = K_1 \times K_2$  is converted to  $K_2$  blocks with length  $K_1$ . The  $(N_1, K_1)$  outer code encodes every block and generates  $K_2$  codewords with length  $N_1$ . Codewords obtained from the first encoder are transposed to construct  $N_1$  blocks of length  $K_2$ , which are considered as the message for the  $(N_2, K_2)$  (inner) code. Finally, codewords obtained from the second encoding are multiplexed to form a codeword with length  $N = N_1 \times N_2$  for the concatenated code. The minimum weight of product codes

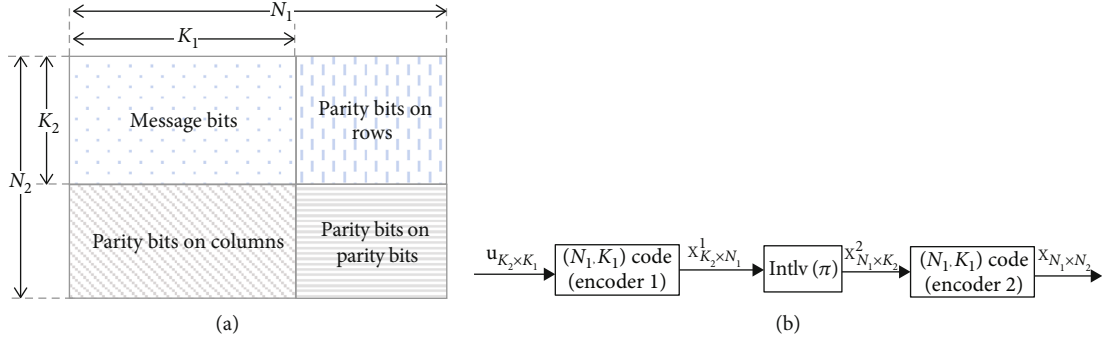


FIGURE 3: (a) Two-dimensional codeword structure of product polar codes. (b) The encoding structure of product polar codes.

is calculated by  $d_{\min 1} \times d_{\min 2}$  [19], where  $d_{\min 1}$  and  $d_{\min 2}$  are minimum weights of the  $(N_1, K_1)$  and  $(N_2, K_2)$  constituent codes, respectively.

The product code can be viewed as a serially concatenated code, which is formed by  $(N_1 * K_2, K_1 * K_2)$  outer and  $(N_1 * N_2, N_1 * K_2)$  inner codes. This structure is shown in Figure 3(b). Indeed, the  $(N_1, K_1)$  outer code encodes  $K_2$  blocks of messages with length  $K_1 \cdot K_2 * N_1$  codewords obtained from this encoding which are interleaved to form  $N_1$  messages with length  $K_2$ . Finally, every  $K_2$  bit message is encoded by the  $(N_2, K_2)$  code to conclude a codeword with a length of  $N_1 * N_2$  [20].

The behaviour of the product code allows us to analyse its performance by utilizing the uniform interleaver with length  $N_1 K_2$ , which maps information with length  $\ell$  into all distinct  $\binom{N_1 K_2}{\ell}$  permutations of it with equal probability of  $\frac{1}{\binom{N_1 K_2}{\ell}}$  as seen in [18]. The analysis conducted by the uniform interleaver represents the average performance of the code. Indeed, for each value of the signal-to-noise ratio, there will always be an interleaver, which provides better performance for the code compared to the uniform interleaver. In this case, the CWF of the product code is expressed by [18]

$$A_{\omega}^{C_{PR}}(H) = \sum_{\ell=0}^{N_1 K_2} \frac{A_{\omega, \ell}^{C_o^{K_2}} \cdot A_{\ell}^{C_i^{N_1}}(H)}{\binom{N_1 K_2}{\ell}}, \quad (16)$$

where  $A_{\omega, \ell}^{C_o^{K_2}}$  and  $A_{\ell}^{C_i^{N_1}}(H)$  are the number of codewords with weight  $\ell$  obtained to form messages with weight  $\omega$  of the outer code and CWF of the inner code, respectively. As  $A^{C_i^{N_1}}(L, H) = [A^{C_i}(L, H)]^{N_1}$  from (9), we have

$$A_{\ell}^{C_i^{N_1}}(H) = \left[ \frac{1}{\ell!} \frac{\partial^{\ell} [A^{C_i}(L, H)]}{\partial L^{\ell}} \Big|_{L=0} \right]^{N_1}. \quad (17)$$

Let  $f(L, H) = A^{C_i}(L, H)$ . Based on the Leibnitz theorem, the above equation is given by

$$\begin{aligned} A_{\ell}^{C_i^{N_1}}(H) &= \frac{1}{\ell!} \left\{ \sum_{i=0}^{\ell} \binom{\ell}{i} \frac{\partial^{\ell-i} f^{(N_1-1)}(L, H)}{\partial L^{\ell-i}} \frac{\partial^i f(L, H)}{\partial L^i} \right\} \Big|_{L=0} \\ &= \frac{1}{\ell!} \left\{ \frac{\partial^{\ell} f^{(N_1-1)}(L, H)}{\partial L^{\ell}} \cdot f(L, H) \right. \\ &\quad \left. + f^{(N_1-1)}(L, H) \frac{\partial^{\ell} f(L, H)}{\partial L^{\ell}} \right. \\ &\quad \left. + \sum_{i=1}^{\ell-1} \binom{\ell}{i} \frac{\partial^{\ell-i} f^{(N_1-1)}(L, H)}{\partial L^{\ell-i}} \frac{\partial^i f(L, H)}{\partial L^i} \right\} \Big|_{L=0}. \end{aligned} \quad (18)$$

The third term is expressed as

$$\begin{aligned} &\frac{1}{\ell!} \left\{ \sum_{i=1}^{\ell-1} \binom{\ell}{i} \frac{\partial^{\ell-i} f^{(N_1-1)}(L, H)}{\partial L^{\ell-i}} \frac{\partial^i f(L, H)}{\partial L^i} \right\} \Big|_{L=0} \\ &= \sum_{i=1}^{\ell-1} A_{\ell}^{C_i^{(N_1-1)}}(H) A_i^{C_i}(H). \end{aligned} \quad (19)$$

In product polar codes with the  $R = 1/4$  constituent by two half rate polar codes, the minimum weight obtained from messages with weight of  $d_{\min}$  is  $2d_{\min}$  [4]. Hence, the lowest order of  $H$  obtained from the third term will be  $2d_{\min}$ . By contrast, for the first and second terms, the expected lowest order of  $H$  is  $d_{\min}$  as they are only formed by one derivative. Therefore, for different  $(E_b/N_0)$ s, the third term will not be as effective as the first and second terms on the performance of the code. This will be more evident, when the value of  $E_b/N_0$  increases. Continuing the Leibnitz theorem for the first and second terms, CWF of the inner code is approximately given by

$$\begin{aligned} A_{\ell}^{C_i^{N_1}}(H) &\approx \frac{1}{\ell!} \left\{ f^{N_1-1}(L, H) \frac{\partial^{\ell} f(L, H)}{\partial L^{\ell}} \cdot f(L, H) \right. \\ &\quad \left. + (N_1 - 1) f^{N_1-1}(L, H) \frac{\partial^{\ell} f(L, H)}{\partial L^{\ell}} \right\} \Big|_{L=0}. \end{aligned} \quad (20)$$

For  $L = 0$ ,  $f^{N_1-1}(L, H) = 1$ . Hence,

$$A_\ell^{C_i^{N_1}}(H) = \frac{1}{\ell!} \left[ (N_1) \frac{\partial^\ell f(L, H)}{\partial L^\ell} \right] \Big|_{L=0} = N_1 A_\ell^{C_i}(H). \quad (21)$$

Similarly,  $A_{\omega, \ell}^{C_o^{K_2}} = K_2 A_{\omega, \ell}^{C_o}$ . Therefore, CWEF of the product code is finalised as follows:

$$A_\omega^{C_{PR}}(H) = \sum_{\ell=0}^{N_1 K_2} \frac{N_1 K_2 A_{\omega, \ell}^{C_o} \cdot A_\ell^{C_i}(H)}{\binom{N_1 K_2}{\ell}}. \quad (22)$$

As shown in Section 1.1, codewords with the minimum weight have the most effect on the performance of the code at the medium to high signal-to-noise ratios. Therefore, equation (24) can be approximately given by

$$A_\omega^{C_{PR}}(H) \cong \frac{N_1 K_2 A_{\omega, d_{\min}}^{C_o} \cdot A_{d_{\min}}^{C_i}(H)}{\binom{N_1 K_2}{d_{\min}}}. \quad (23)$$

As the lowest order of  $H$  obtained from  $A_{d_{\min}}^{C_i}(H)$  is  $2d_{\min}$ , more simplified CWEF of the product code is given by

$$A_\omega^{C_{PR}}(H) \approx \frac{N_1 K_2 A_{\omega, d_{\min}}^{C_o} \cdot A_{d_{\min}, 2d_{\min}}^{C_i} H^{2d_{\min}}}{\binom{N_1 K_2}{d_{\min}}}. \quad (24)$$

On the other hand, the CWEF of a single polar code with rate  $R'$  and minimum weight of  $d'_{\min}$  is approximately given by

$$A_\omega^{C_{SP}}(H) \approx A_{\omega, d'_{\min}}^{C_{SP}} H^{d'_{\min}}, \quad (25)$$

where  $A_{\omega, d'_{\min}}^{C_{SP}}$  is number of codewords with the weight of  $d'_{\min}$  generated from messages with weight  $\omega$ .

Based on  $R' = R$  and  $d'_{\min} = (d_{\min})^2$ , Table 2 shows conditions for different product codes with  $R = 1/4$ , which satisfy  $A_\omega^{C_{PR}}(H) < A_\omega^{C_{SP}}(H)$ . Considering (10), this inequality concludes that the product polar code can produce a lower upper bound of probability of error than that obtained from the single polar code. Conditions are formed by  $A_{\omega, d_{\min}}^{C_o} \cdot A_{d_{\min}, 2d_{\min}}^{C_i} = 10 A_{\omega, d'_{\min}}^{C_{SP}}$ , which assumes that the multiplicity of the minimum weight obtained from the product code is much greater than that generated by the single code. It is observed that the expected lower probability of error for the product code is guaranteed by  $(E_b/N_0)$ s that are applicable in communication systems.

In particular cases, multiplicity of minimum weight of the product code can be similar or even smaller than

the multiplicity of the minimum weight of the single code. This will provide a better condition for the product code to outperform the single code at the error floor region.

Figure 4 shows the probability of error for the (64, 16) code. In these graphs, only the minimum weight specifications of the code are considered in (13). Based on the exhaustive search algorithm, multiplicity of 196 for the minimum weight of the (64, 16) product code is obtained. From Table 1, it is observed that codes have the same minimum weight but the product code generates lower multiplicity for the minimum weight. This consequently provides a better error correcting compared to the single polar code. Again, the improvement should be mainly observed at the medium to high  $(E_b/N_0)$ s, where the practical behaviour of codes is matched with the calculated  $P_b(e)$ s [18].

**3.1. Reliability of Bit Channels for Product Polar Codes.** The codeword obtained from the product polar encoding can be expressed as a  $N_1 \times N_2$  matrix. This matrix can also be represented by two  $K_1 \times N_1$  and  $(N_2 - K_2) \times N_1$  submatrices, which denote the codeword of the outer code and parity bits of the inner code, respectively. The reliability of bits for the first and second submatrices is directly determined from equation (2). In every row of the first submatrix, computation is initialized by the value given in equation (3). Therefore, bits positioned at the same column will have the same reliability value. The reliability of bits at the inner code is determined on the basis of reliability of interleaved bits obtained from the first submatrix. Indeed, reliabilities calculated from  $K_1$  rows of the first submatrix are considered as the initialized values for determining the reliability of bits positioned at the second submatrix.

Table 3 shows the reliability of bit channels for the (16, 4) product polar code constituted by two (4, 2) polar codes, which are recursively calculated by equation (2). In this case, channels are considered as binary AWGN and followed the polar code construction method proposed in [13–15] with  $E_b/N_0 = 0$  dB. In the first encoding, reliability of bits positioned at the first two rows is obtained. This is accomplished when  $\alpha \leq i \leq n_2$  ( $K_2 = 2^\alpha$ ,  $N_2 = 2^{n_2}$ ,  $n_2 = \{1, 2, 3, \dots\}$ ,  $\alpha = n_2 - 1$ ) is applied for equation (2). In this case,  $Z(W_{2^{\alpha-1}}^{(j)})$ s,  $1 \leq j \leq K_2$ , are channel reliabilities of bits placed in one column of the first submatrix. Then, from these calculated values, those reliabilities positioned at one column of the matrix are applied as  $Z(W_2^{j'})$ ,  $1 \leq j' \leq 2$ , to determine reliabilities of parity bits for codewords of the  $(N_2, K_2)$  polar code. All of the reliabilities relevant to the inner code are expressed as  $Z(W_4^{j'})$ ,  $1 \leq j' \leq 4$  in equation (2).

This table also gives the reliabilities of bit channels for the single (16, 4) polar code. Similar to the single code, the product code effectively polarizes the channel as either good or bad channels. Such recognition allows us to construct polar codes with arbitrary lengths and  $R > 1/4$  originated from the main product polar code when a number of codeword bits are punctured.

TABLE 2: Minimum weight specifications of polar codes with rates 1/2 and 1/4.

Product code	$N_1 * K_2$	$d_{\min}$	$X = \binom{N_1 K_2}{d_{\min}}$	Exponential form of $(10 * (1/X))$	$A_{\omega}^{C^{PR}}(H) < A_{\omega}^{C^{SP}}(H)$ if
$(8, 4) \times (8, 4)$	32	4	35960	$e^{-8.1876}$	$E_b/N_0 < 3.75$ dB
$(8, 4) \times (16, 8)$	64	4	635376	$e^{-11.0594}$	$E_b/N_0 < 5.38$ dB
$(16, 8) \times (16, 8)$	128	4	$1.0668 \times 10^7$	$e^{-13.8802}$	$E_b/N_0 < 6.50$ dB
$(32, 16) \times (32, 16)$	512	4	$2.8299 \times 10^9$	$e^{-19.4609}$	$E_b/N_0 < 8.20$ dB
$(64, 32) \times (64, 32)$	2048	8	$7.5714 \times 10^{21}$	$e^{-48.0761}$	$E_b/N_0 < 5.28$ dB

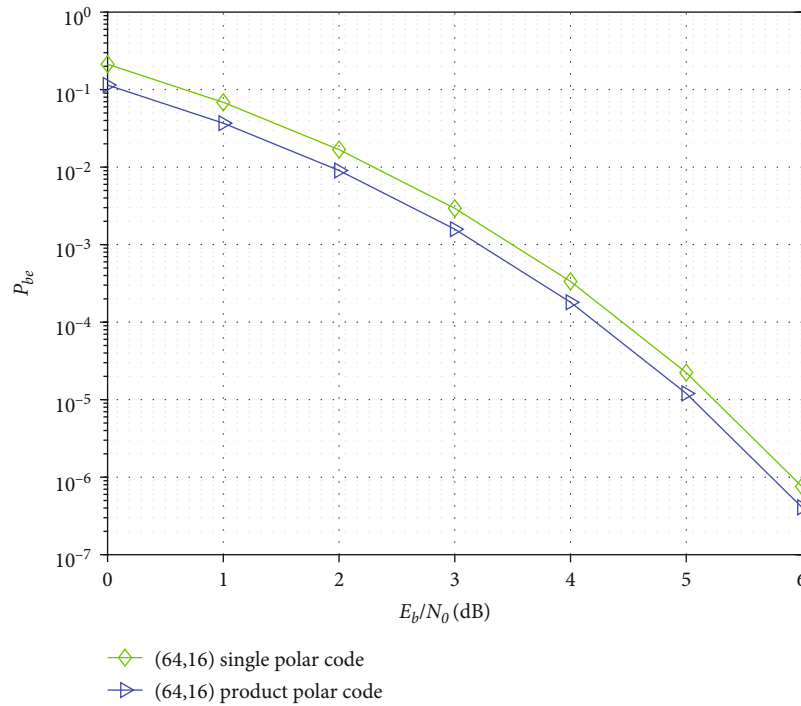


FIGURE 4: Probability of error for (64, 16) polar code.

TABLE 3: Reliability of bits for (16, 4) polar codes.

First encoding			Product polar code					Second encoding					(16, 4) single polar code		
0.84	0.25	0.36	0.018	0.9745	0.4410	0.5911	0.0363	0.9994	0.6875	0.8328	0.0713				
0.84	0.25	0.36	0.018	0.7062	0.0637	0.1300	0.0003	0.9137	0.1233	0.2430	0.0007				
				0.9745	0.4410	0.5911	0.0363	0.9497	0.1945	0.3493	0.0013				
				0.7062	0.0637	0.1300	0.0003	0.4987	0.0041	0.0169	$1.12e - 07$				

#### 4. A Puncturing Technique for Product Polar Codes

A  $(N, K)$  polar code is punctured by removing a number of rows and columns from the generator matrix. Let  $X$  be the number of punctured bits. The codeword length and rate of the newly designed code are  $N - X$  and  $K/(N - X)$ , respec-

tively. In product polar codes, puncturing can be conducted on one of its constituent codes. The rate of these codes formed by two constituent  $(N_1, K_1)$  and  $(N_2, K_2)$  codes is determined by

$$R_p = \frac{K_1 \times K_2}{(N_1 - \varphi)N_2}, \quad (26)$$

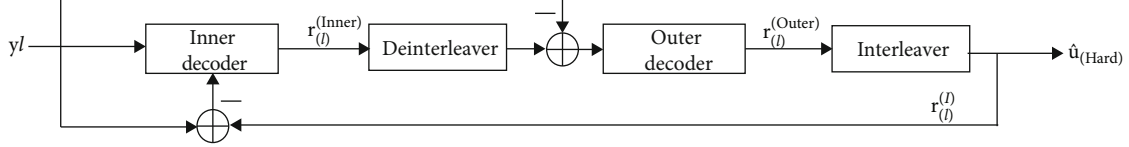


FIGURE 5: Structure of interactive iterative decoding of product codes.

where  $\varphi$  is the number of bits removed from the  $(N_1, K_1)$  code.

In order to have  $R_p$  equal to the rate of the  $(N - X, K)$  single polar code ( $N = N_1 \times N_2$ ,  $K = K_1 \times K_2$ ), the following condition should be satisfied:

$$\varphi = \frac{X}{N_2}. \quad (27)$$

This means a lower number of bits punctured from one of the constituent codes are required in comparison with the number of punctured bits applied for the single code.

Based on the two-dimensional structure of the codeword,  $\varphi$  rows or columns representing the most unreliable bits of a constituent code will be removed. The structure of punctured codewords for the (64, 16) product code is given by

$$\begin{pmatrix} \times & C_{(1,2)} & C_{(1,3)} & C_{(1,4)} & \times & C_{(1,6)} & C_{(1,7)} & C_{(1,8)} \\ \times & C_{(2,2)} & C_{(2,3)} & C_{(2,4)} & \times & C_{(2,6)} & C_{(2,7)} & C_{(2,8)} \\ \times & C_{(3,2)} & C_{(3,3)} & C_{(3,4)} & \times & C_{(3,6)} & C_{(3,7)} & C_{(3,8)} \\ \times & C_{(4,2)} & C_{(4,3)} & C_{(4,4)} & \times & C_{(4,6)} & C_{(4,7)} & C_{(4,8)} \\ \times & C_{(5,2)} & C_{(5,3)} & C_{(5,4)} & \times & C_{(5,6)} & C_{(5,7)} & C_{(5,8)} \\ \times & C_{(6,2)} & C_{(6,3)} & C_{(6,4)} & \times & C_{(6,6)} & C_{(6,7)} & C_{(6,8)} \\ \times & C_{(7,2)} & C_{(7,3)} & C_{(7,4)} & \times & C_{(7,6)} & C_{(7,7)} & C_{(7,8)} \\ \times & C_{(8,2)} & C_{(8,3)} & C_{(8,4)} & \times & C_{(8,6)} & C_{(8,7)} & C_{(8,8)} \end{pmatrix}, \quad (28)$$

where  $C_{(i,j)}$ ,  $(i, j) \in \{\mathcal{A}_g\}$ , and  $\mathcal{A}_g = \{4, 6, 7, 8\}$  are message bits and the rest are parity bits. In this matrix, “ $\times$ ” denotes position of punctured bits. This code is constructed by two (8, 4) polar codes. For a punctured code formed based on a low value of  $\varphi$ , it is possible to do the puncturing by selecting bits that are not adjacent to each other [21]. In this code, the first and fifth bits of the (8, 4) code have the lowest reliabilities. Hence, the information positioned at the mentioned columns (first and fifth columns) of the matrix is punctured. In this case, at every decoding step of the outer code, there would be at least one valid (nonpunctured) information, which assists the decoder to recover the original codeword.

On the other hand, based on transposing the received information, at the second decoder, some adjacent information is recognized as punctured information. However, as shown in Figure 5, due to interactive iterative decoding of the product code, in every iteration, decoded information obtained from two decoders is exchanged with each other. This allows both decoders to effectively recover the original

information transmitted over the noisy channel. Regardless the error floor effect, existence of a good error correcting performance is expected from the medium to high  $(E_b/N_0)$ s, where the first decoder has a better chance to correctly decode the received information.

By contrast, in a single polar code formed by a high number of punctured bits, it is possible to puncture adjacent bits or those having high reliability. This is mainly evident for codes with  $X \geq K/2$  [21]. Moreover, as interactive iterative decoding is not conducted for a single polar code, it is expected that with the same rate and length, punctured product polar codes outperform punctured single polar codes.

**4.1. Puncturing Effect on the Channel Reliability for SPC and PPC.** The reliability of the bit channel can affect error performance of the punctured code. Puncturing a bit ultimately deletes its bit-channel reliability. This consequently reduces the overall bit-channel reliability ( $Z_{\text{sum}}$ ) of the code and mitigates decoding performance. The overall reliability of a polar code is defined by

$$Z_{\text{sum}} = \sum_{i \in N} Z(W_N^{(i)}). \quad (29)$$

As mentioned above, the  $Z_{\text{sum}}$  value is reduced, when bits are punctured. Therefore, it is essential to keep  $Z_{\text{sum}}$  as high as possible to have a good performance for the punctured code. Let  $\zeta$  be the overall reliability of nonpunctured bits of a punctured code, which is given by

$$\zeta = Z_{\text{sum}} - \sum_{j \in \mathcal{A}_p} Z(W_N^{(j)}), \quad (30)$$

where  $Z(W_N^{(j)})$  is the bit-channel reliability value of the  $j$ -th bit of the codeword and  $\mathcal{A}_p$  represents the set of indices applied for punctured codeword bits. Obviously, for  $\mathcal{A}_p = 0$ , the code is considered as a nonpunctured one. From (29) and (30), the below inequality can be observed

$$\zeta \leq Z_{\text{sum}}. \quad (31)$$

Therefore, if  $\zeta$  is tightly upper bounded, punctured bits have less effect on the code performance. This requires careful selection of puncturing bits so that the calculated  $\zeta$  remains close to  $Z_{\text{sum}}$  of the code.

As mentioned before, in product polar codes, a lower number of bits are punctured from their constituent codes compared to a single punctured code. Hence, less effect of punctured bits on the performance of product code compared to the single code is expected. Table 4 confirms this



TABLE 4: Comparison of  $\zeta$  for different codes. The channel reliability is calculated for binary AWGN channel with SNR = 0 dB.

$N$	Product code		Single code	
	$Z_{\text{sum}}$	$\zeta$	$Z_{\text{sum}}$	$\zeta$
48	23.5443	10.0989	23.5443	8.0056
192	94.1771	40.3955	94.1771	30.5400
768	376.7085	161.5818	376.7085	120.8185
3072	$1.5068 \times 10^3$	646.3274	$1.5068 \times 10^3$	482.8405
12288	$6.0273 \times 10^3$	$2.5853 \times 10^3$	$6.0273 \times 10^3$	$1.9313 \times 10^3$

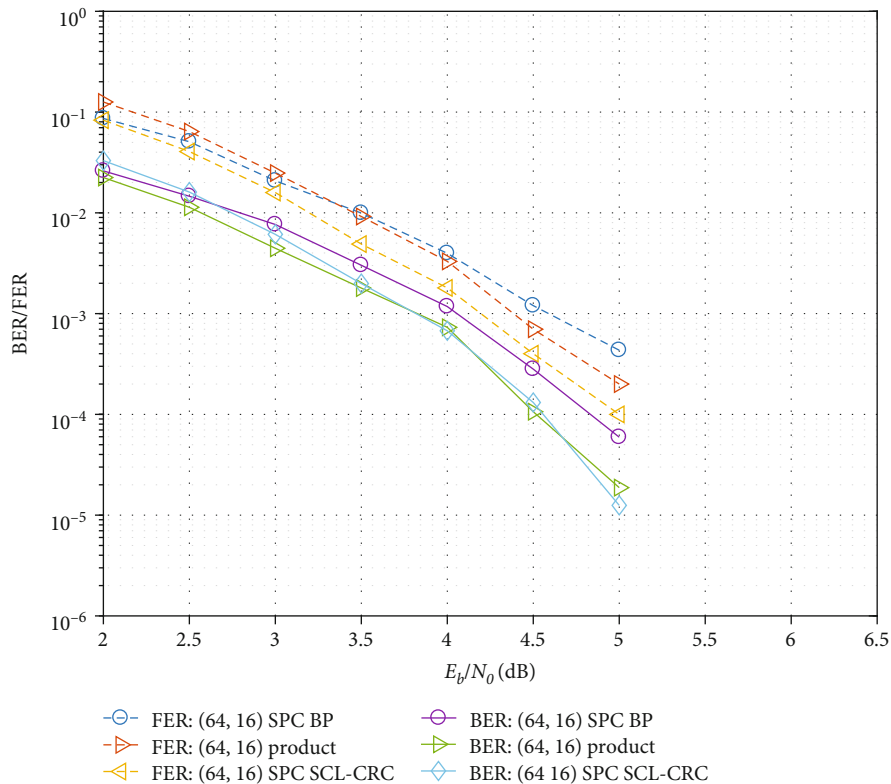


FIGURE 6: Performance of (64, 16) codes decoded by different algorithms.

subject for the number of punctured product polar and single codes formed by the equal code length and rate.

## 5. Numerical Results

The bit error rate (BER) and frame error rate (FER) performances of different product and single polar codes transmitted over the AWGN channel are verified. Codes are modulated by BPSK and decoded by the soft-in soft-out belief propagation method mentioned in [16]. For product codes, four iterations are considered for the interactive decoding between two constituent decoders. For single systematic polar codes, decoding is accomplished by 60 iterations. A successive cancellation list with cyclic redundancy check (SCL-CRC) decoding is performed with the list size of either  $L=4$  or  $L=8$ . Moreover,  $m$ -bit CRC code is applied. This means  $m$  bits added to the message bits.

Figure 6 shows the performance of (64, 16) codes. It is observed that the product code has a similar BER performance to the single code with SCL-CRC ( $L=8$  and  $m=4$ ) decoding. Constituent half rate codes are decoded by 30 iterations. In addition, at  $E_b/N_0 = 4.5$  dB, this code shows a performance better than the single polar code. The product code provides significantly better performance than SPC with BP decoding.

Figure 7 shows the performance of two product codes and their performance with BCH-polar [22] and RS-polar [23] codes. Constituents of both product codes are based on non-half rate polar codes. It shows that the (256, 42) product code with rate  $R=1/6$  has close FER performance with BCH-based polar code. In addition, the (512, 171) product code is constituted by (32, 19) and (16, 9) polar codes. At  $E_b/N_0 = 4.0$  dB, this code has 2.0 dB and 0.85 dB gain in BER performance compared to RS-polar and LDPC-polar codes,

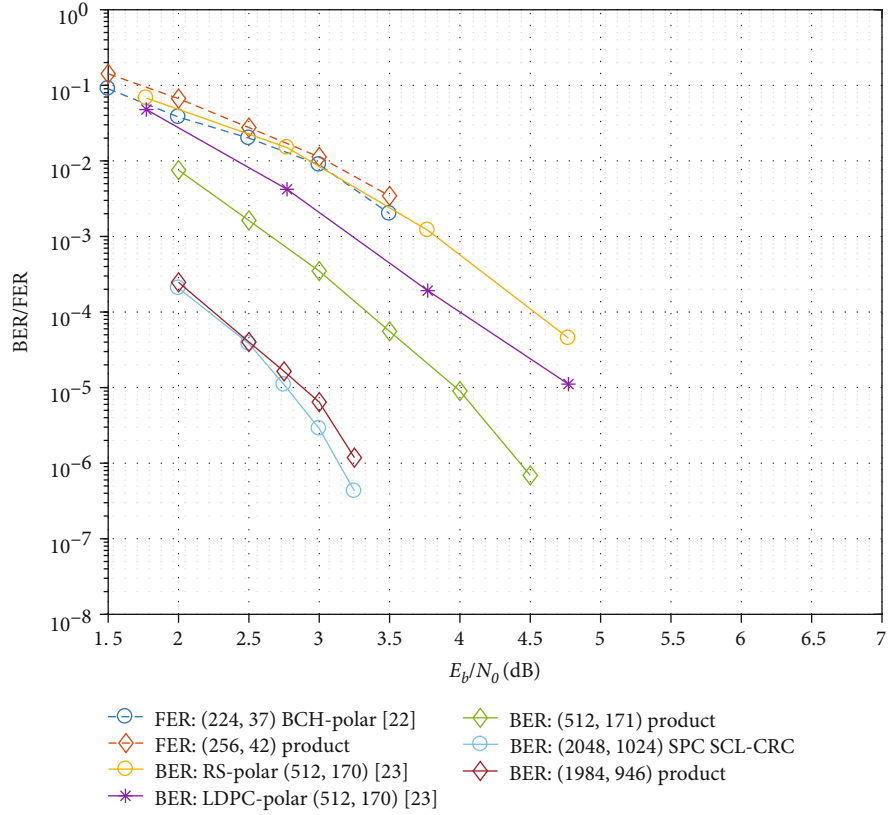


FIGURE 7: Performance of product polar codes and their comparison with BCH-polar, RS-polar, and LDPC-polar codes.

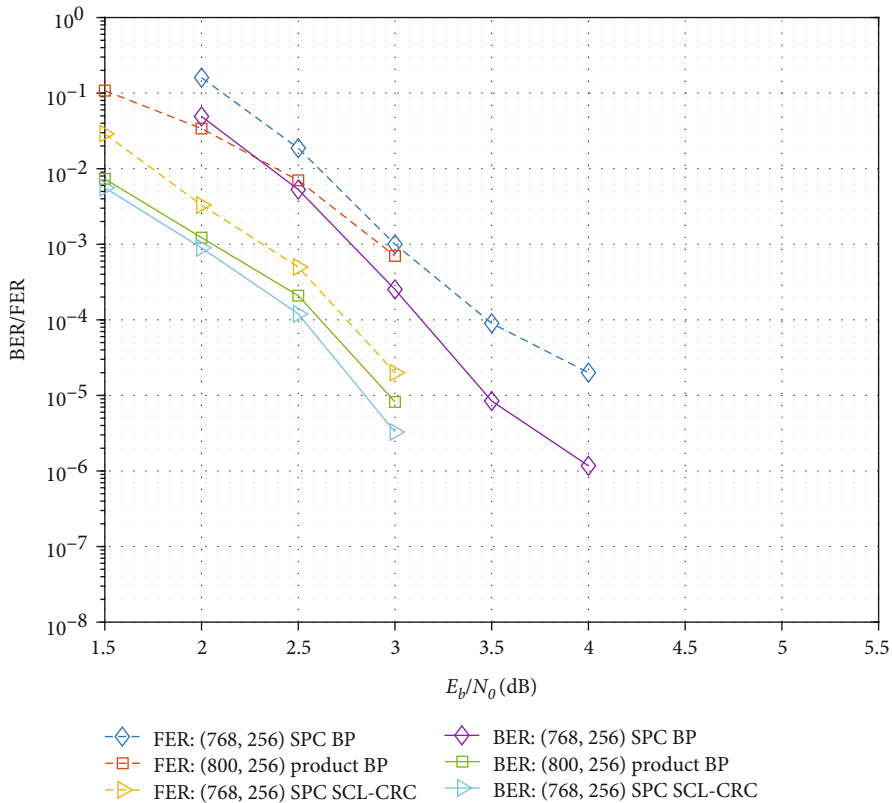


FIGURE 8: BER and FER performance of (800, 256) product polar code and its comparison with other codes.

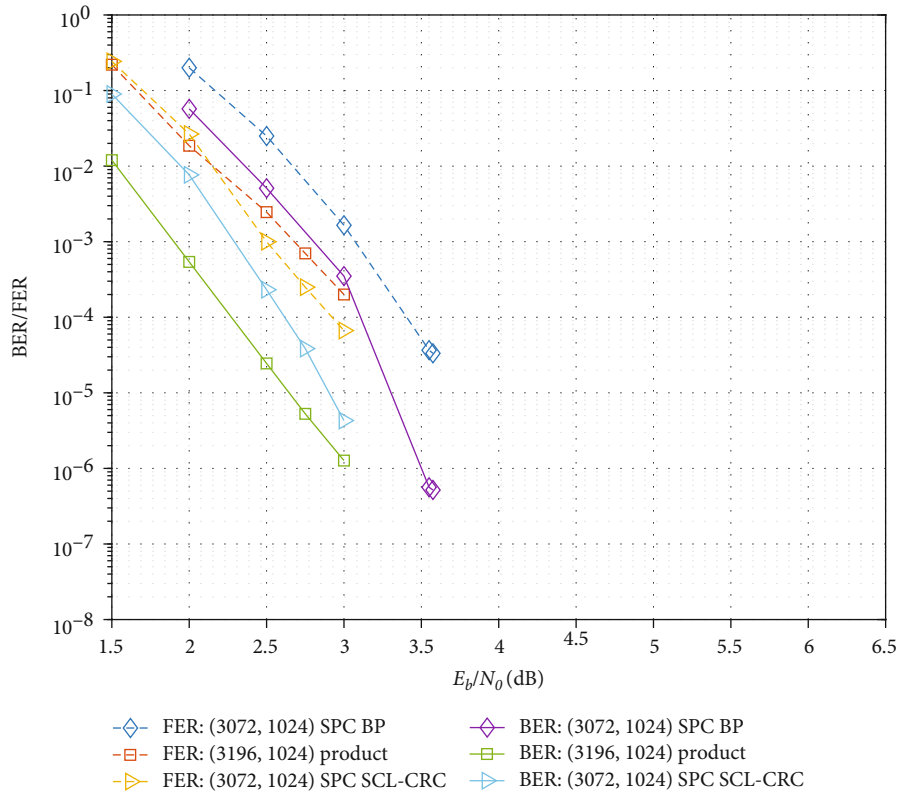


FIGURE 9: BER and FER performances of (3196, 1024) product and (3072, 1024) single polar codes.

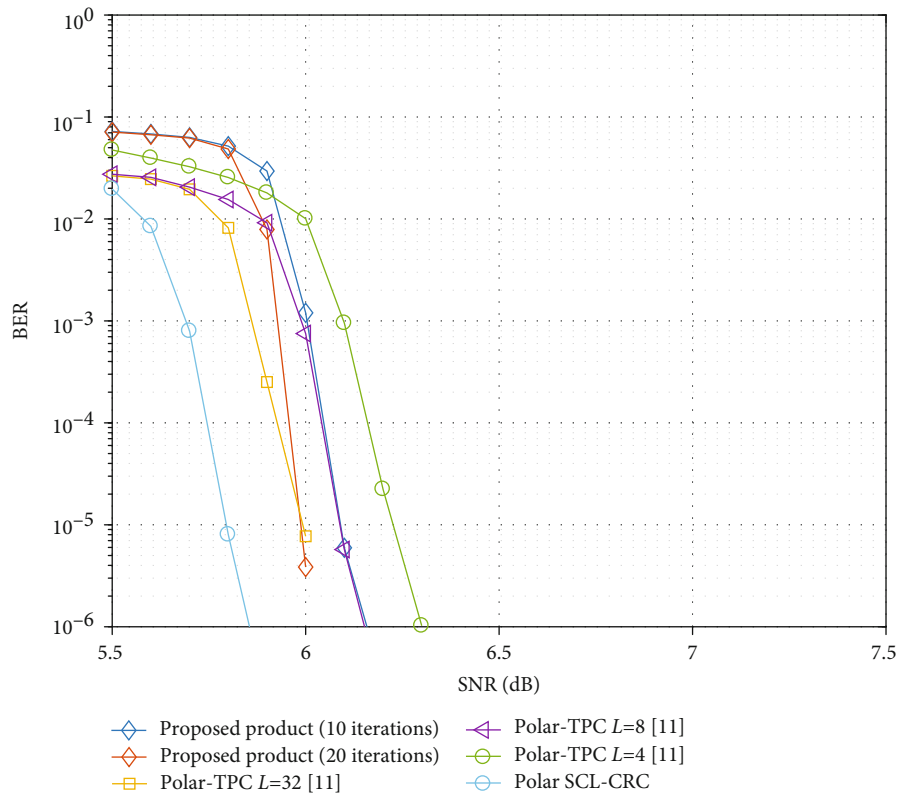


FIGURE 10: BER performance of  $(256^2, 239^2)$  codes.

respectively. Similarly, the (1984, 946) product code constituted by the (62, 43) and (32, 22) polar codes has close performance to the single (2048, 1024) polar code decoded by SCL-CRC decoding with  $L = 4$  and  $m = 4$ . All product codes apply 30 iterations in their BP decoding.

Figure 8 shows the performance of the (800, 256) product and (768, 256) single polar codes. In this case, the product code is constructed by two (32, 16) polar codes, which are decoded by BP decoding, including 90 iterations. Due to interactive iterative decoding applied in the product code, one decoder can properly recover punctured information from a block of information and deliver them to the second decoder for the next step of decoding. It is observed that the product code provides similar performance to decoding based SPC, while outperforming the single polar code decoded by BP decoding.

Figure 9 shows the BER and FER performances of the (3196, 1024) product and (3072, 1024) single polar codes. Again, 90 iterations were considered for BP decoding of constituent codes. It is observed that the product code has a better BER performance compared to the single code. It also has similar FER performance with the single code decoded by the SCL-CRC technique with  $L = 4$  and  $m = 4$ .

Furthermore, BER performance of the  $(256^2, 239^2)$  product polar code and polar-TPC proposed in [11] are analysed in Figure 10. Codes are modulated by quadrature phase shift keying (QPSK). Sixty (60) iterations are applied for BP decoding of the product polar code. It is concluded that the product code decoded by ten and twenty interactive iterations has similar performance to polar-TPC decoded by the SCL-Chase algorithm with the list size of  $L = 8$  and  $L = 32$ , respectively.

## 6. Conclusions and Future Work

The paper presented a product coding scheme of polar codes formed by two different rates of polar codes. The conducted analysis and simulation results confirmed that the constructed product codes outperform the conventional polar codes. This is mainly evident in the medium to high signal-to-noise ratios, which is applicable for both nonpunctured and punctured codes. In the future work, a modification of the structure of punctured product polar codes will be followed aimed at improving their performance.

### Data Availability

Data sharing not applicable to this article as no data sets were generated or analysed during the current study.

### Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

### Acknowledgments

This work is supported by the Research Training Program (RTP) scholarship under Charles Darwin University.

## References

- [1] E. Arıkan, "Channel polarization: a method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Transactions on Information Theory*, vol. 55, no. 7, pp. 3051–3073, 2009.
- [2] Y. Wang, K. R. Narayanan, and Y.-C. Huang, "Interleaved concatenations of polar codes with BCH and convolutional codes," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 2, pp. 267–277, 2016.
- [3] M. Bakshi, S. Jaggi, and M. Effros, "Concatenated polar codes," in *2010 IEEE International Symposium on Information Theory*, Austin, TX, USA, June 2010.
- [4] A. Eslami and H. Pishro-Nik, "On finite-length performance of polar codes: stopping sets, error floor, and concatenated design," *IEEE Transactions on Communications*, vol. 61, no. 3, pp. 919–929, 2013.
- [5] I. Tal and A. Vardy, "List decoding of polar codes," *IEEE Transactions on Information Theory*, vol. 61, no. 5, pp. 2213–2226, 2015.
- [6] T. Wang, D. Qu, and T. Jiang, "Parity-check-concatenated polar codes," *IEEE Communications Letters*, vol. 20, no. 12, pp. 2342–2345, 2016.
- [7] J. Park, I. Kim, and H. Y. Song, "Construction of parity-check-concatenated polar codes based on minimum hamming weight codewords," *Electronics Letters*, vol. 53, no. 14, pp. 924–926, 2017.
- [8] D. Wu, A. Liu, Y. Zhang, and Q. Zhang, "Parallel concatenated systematic polar codes," *Electronics Letters*, vol. 52, no. 1, pp. 43–45, 2016.
- [9] S.-N. Hong, D. Hui, and I. Maric, "Capacity-achieving rate-compatible polar codes," *IEEE Transactions on Information Theory*, vol. 63, no. 12, pp. 7620–7632, 2017.
- [10] Q. Zhang, A. Liu, Y. Zhang, and X. Liang, "Practical design and decoding of parallel concatenated structure for systematic polar codes," *IEEE Transactions on Communications*, vol. 64, no. 2, pp. 456–466, 2016.
- [11] T. Koike-Akino, C. Cao, and Y. Wang, "Turbo product codes with irregular polar coding for high-throughput parallel decoding in wireless OFDM transmission," in *2018 IEEE International Conference on Communications (ICC)*, pp. 1–7, Kansas, MO, USA, May 2018.
- [12] E. Arkan, "A performance comparison of polar codes and Reed-Muller codes," *IEEE Communications Letters*, vol. 12, no. 6, pp. 447–449, 2008.
- [13] S. Zhao, P. Shi, and B. Wang, "Designs of Bhattacharyya parameter in the construction of polar codes," in *2011 7th International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 1–4, Wuhan, China, September 2011.
- [14] H. Li and J. Yuan, "A practical construction method for polar codes in AWGN channels," in *IEEE 2013 Tenccon - Spring*, pp. 223–226, Sydney, NSW, Australia, April 2013.
- [15] H. Vangala, E. Viterbo, and Y. Hong, "A comparative study of polar code constructions for the AWGN channel," <https://arxiv.org/abs/1501.02473>.
- [16] E. Arıkan, "Systematic polar coding," *IEEE Communications Letters*, vol. 15, no. 8, pp. 860–862, 2011.
- [17] N. Hussami, S. B. Korada, and R. Urbanke, "Performance of polar codes for channel and source coding," in *2009 IEEE International Symposium on Information Theory*, Seoul, South Korea, June–July 2009.

- [18] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Serial concatenation of interleaved codes: performance analysis, design and iterative decoding," *IEEE Transactions on Information Theory*, vol. 44, no. 3, pp. 909–926, 1998.
- [19] W. Ryan and S. Lin, *Channel Codes: Classical and Modern*, Cambridge University Press, Cambridge, United Kingdom, 2009.
- [20] S. Lin and D. Costello, *Error Control Coding: Fundamentals and Applications*, Printice Hall, New Jersey, USA, 2004.
- [21] M. Hanif and S. Vafi, "An efficient puncturing method for the short and long length polar codes," in *2017 11th International Conference on Signal Processing and Communication Systems (ICSPCS)*, Gold Coast, QLD, Australia, December 2017.
- [22] Y. Wang, L. Chen, C. Liu, and Z. Xing, "An improved concatenation scheme of BCH-Polar codes with low-latency decoding architecture," *IEEE Access*, vol. 7, pp. 95867–95877, 2019.
- [23] S. M. Zhao, S. P. Xu, and C. Xing, "Concatenated polar-coded multilevel modulation," in *2015 10th International Conference on Communications and Networking in China (ChinaCom)*, Shanghai, China, August 2015.