Received 12 February 2020; accepted 27 February 2020. Date of publication 16 March 2020; date of current version 16 April 2020.

Digital Object Identifier 10.1109/OJCOMS.2020.2979529

Transformation of Binary Linear Block Codes to Polar Codes With Dynamic Frozen

CHIEN-YING LIN [1], YU-CHIH HUANG [1] (Member, IEEE), SHIN-LIN SHIEH [2], AND PO-NING CHEN [1] (Senior Member, IEEE)

[1] Institute of Communications Engineering, National Chiao Tung University, Hsinchu 300, Taiwan

[2] Department of Communications Engineering, National Taipei University, Taipei 23741, Taiwan

CORRESPONDING AUTHOR: C.-Y. LIN (e-mail: thisistt.cm05g@g2.nctu.edu.tw)

The work of Chien-Ying Lin and Po-Ning Chen was supported by the Ministry of Science and Technology of Taiwan under Grant MOST 106-2221-E-009-025-MY3. The work of Yu-Chih Huang was supported by the Ministry of Science and Technology of Taiwan under Grant MOST 109-2636-E-009-021. The work of Shin-Lin Shieh was supported in part by the Ministry of Science and Technology of Taiwan under Grant MOST 106-2221-E-009-025-MY3, and in part by the Ministry of Science and Technology of Taiwan under Grant MOST 106-2221-E-305-006-MY3.

ABSTRACT In this paper, a general transformation of binary linear block codes (BLBCs) to (possibly, multi-kernel) polar codes with dynamic frozen bits is proposed. Through a simple matrix permutation operation, a one-to-one connection between the codewords of a BLBC and its transformed polar code can be established. This transformation allows the usage of *any* decoding algorithm of polar codes for efficient soft decoding of BLBCs, including the powerful successive cancellation list (SCL) decoding algorithm. Simulations show that the soft SCL polar decoding of BLBCs can achieve a comparative performance to the order statistic decoding (OSD), as well as the maximum-likelihood decoding (MLD) in certain cases, with a much lower computational complexity.

INDEX TERMS Channel coding, Polar codes, binary linear block codes and soft decoding.

I. INTRODUCTION

THIS binary linear block codes (BLBC), as an important subclass of error correcting codes, have played a pivotal role in modern communications [1]. In applications such as communication systems with strict latency constraints, codes with short blocklength are preferred. In these particular situations, algebraic coding theory provides a powerful means to code construction with a good minimum pairwise Hamming distance among codewords. In spite of their many nice properties, most of algebraic codes suffer from a major drawback that it is difficult to exploit soft information for decoding [1], thereby lacking of soft decoding algorithms with low complexity.

The main theme of this paper is to study the soft decoding of BLBCs with a particular focus on algebraic codes with short blocklength. A well-known soft decoding algorithm for BLBCs is the order statistic decoding (OSD) algorithm [2]. Extensive simulations in the literature have confirmed that OSD performs close to the maximum-likelihood decoding (MLD) algorithm for many BLBCs. This exceptionally good performance, however, can be obtained when OSD with high order is used, which involves a relatively large computational complexity and a high demand on memory.

In this paper, we tackle the soft decoding of BLBCs with a different approach. By transforming a BLBC into a (possibly, multi-kernel) polar code with dynamic frozen bits, a connection between codewords of the BLBC and codewords of the polar code is built such that *any* soft decoding algorithm of polar codes can be used for efficient soft decoding of BLBCs.

Invented by Arikan [3], polar codes are the first provably capacity-achieving codes with explicit construction and with low encoding and decoding complexity. With the establishment of their multi-kernel designs [4], polar codes of arbitrary blocklength can be constructed. However, the small minimum pairwise Hamming distance of polar codes with short block length results in a limitation on their performance in the short blocklength regime. A solution, as proposed by

Trifonov and Miloslavskaya [5], is to dynamically generate the frozen bits of polar codes according to the message bits, as well as the generator matrix of an auxiliary linear code, such that the minimum pairwise distance can be increased. As polar codes have now been included in 5G new radio standard, it can be expected that the development of efficient decoding for polar codes will become an emerging subject of practical interest.

In order to perform polar decoding on BLBCs, we identify a permutation matrix that transforms a BLBC to a (possibly multi-kernel) polar code with dynamic frozen bits. Since the permutation matrix, to be identified and then employed, is only a function of blocklength, the same permutation matrix can be applied to build the required connection from codewords of several BLBCs of the same length to codewords of their corresponding polar code. This design simplifies the implementation of the receiver when multiple codes of equal length but different code rates are specified in the system. Since the transformation does not alter the code space, no adjustment on the encoding procedure of BLBCs is required; in other words, only the receiver needs to be replaced. Efficient soft decoding of BLBCs can thus be implemented via soft decoding of their transformed polar codes. Simulation results show that the proposed approach can perform close to OSD, as well as MLD in certain cases, with a much lower decoding complexity.

It should be mentioned that the idea of polar-decoding specific BLBCs has been addressed in the literature. In [6, Example 1], a polar subcode was constructed, and was noted to be an extended Bose-Chaudhuri-Hocquenghem (BCH) code [1] (see Example 1 in this paper). In [7], by leveraging the specific structure of the extended (24, 12, 8) Golay code, Bioglio and Land showed that the code can be equivalently transformed to a polar code with (traditionally static) frozen bits. It is also conjectured in [7] that a similar transformation might be possible for many algebraic codes. This paper confirms the conjecture, and further exploits and generalizes the notion by introducing a universal matrix transformation of *any* BLBC of the same blocklenth to a (multi-kernel) polar code with *dynamic* frozen bits. Due to the close structural relation to polar codes [3], [8], [9], Reed-Muller (RM) codes have been studied thoroughly from the polar coding aspect and have been known to be efficiently decodable by decoding algorithms of polar codes. From this perspective, this paper serves as a continued effort in expanding polar decoding to other classes of BLBCs.

II. BACKGROUND

In this section, we review the definitions of BLBCs and polar codes with dynamic frozen bits.

A. BINARY LINEAR BLOCK CODES

An (n, k) BLBC codeword space $\mathcal{C}$ can be defined through its generator matrix $G \in \mathbb{F}_2^{k\times n}$ as

$$\mathcal{C} \triangleq \left\{\boldsymbol{c} = \boldsymbol{m}G|\boldsymbol{m} \in \mathbb{F}_2^k\right\},$$

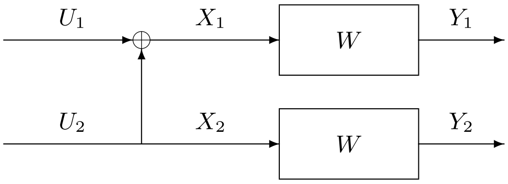


FIGURE 1. Basic transformation.

where $\boldsymbol{m}$ is called message. It can be alternatively defined via a parity-check matrix $H \in \mathbb{F}_2^{n\times(n-k)}$, which gives

$$\mathcal{C} \triangleq \left\{\boldsymbol{c} \in \mathbb{F}_2^n|\boldsymbol{c}H = \boldsymbol{0}\right\}.$$

A basic property of BLBCs is that the mapping between messages and codewords is one-to-one; hence, a message is successfully recovered if its corresponding codeword is identified by the decoder.

B. CHANNEL POLARIZATION OF POLAR CODES

Consider $W : \mathcal{X} \to \mathcal{Y}$ a binary-input discrete memoryless channel with input alphabet $\mathcal{X} = \{0, 1\}$, output alphabet $\mathcal{Y}$, and transition probability $W(y|x)$ for $x \in \mathcal{X}$ and $y \in \mathcal{Y}$. A quality measure of W, which is particularly used for polar codes, is the Bhattacharyya parameters [3] defined as

$$Z(W) \triangleq \sum_{y\in\mathcal{Y}} \sqrt{W(y|0)W(y|1)}.$$

The smaller the Bhattacharyya parameter $Z(W)$, the better the quality of the channel W.

Channel polarization of polar codes is done in two phases [3]. The first *channel combining* phase combines n uses of W into a vector channel W_n of length n in a recursive manner. The recursion is based on the basic transformation in Fig. 1, whose input-output relationship is given by

$$W_2\left(y_1^2|x_1^2\right) \triangleq W(y_1|u_1 \oplus u_2)W(y_2|x_2).$$

The general construction of W_n can be characterized by the so-called Arikan kernel:

$$F_2 \triangleq \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}.$$

The second *channel splitting* phase splits W_n back to n binary-input coordinate channels recursively via the following manipulation from basic transformation:

$$\begin{cases} W^- : U_1 \to (Y_1, Y_2) \\ W^+ : U_2 \to (Y_1, Y_2, U_1) \end{cases}$$

It was shown in [3] that

$$Z\left(W^+\right) \le Z(W) \le Z\left(W^-\right)$$

and

$$Z\left(W^-\right) + Z\left(W^+\right) \le 2Z(W).$$

Hence, W^+ is considered a better channel than W, while W^- is worse than W. The recursions at each phase are performed $\ell = \log_2(n)$ times.

As a linear code, codewords $\boldsymbol{c}_{\mathrm{p}}$ of an $(n = 2^{\ell}, k)$ polar code can also be obtained from the formula $\boldsymbol{c}_{\mathrm{p}} = \boldsymbol{u}G_{\mathrm{p}}$, where $\boldsymbol{u}$ consists of k message bits and $(n-k)$ frozen bits, and $G_{\mathrm{p}} = F_2^{\otimes \ell}$ is the ℓth Kronecker power of Arikan kernel F_2. The positions of the frozen bits are preferably those that are associated with worse channels after channel polarization.

In order to construct polar codes of arbitrary blocklength n (not necessarily a power of 2), a multi-kernel polarization technique was proposed in [4]. As its name reveals, the technique uses a combination of multiple kernels of possibly different sizes for polarization. For example, a polar code of blocklength $n = 6$ can be constructed by a combination of the Arikan kernel and a kernel of size 3 as follows:

$$G_{\mathrm{p}} \triangleq \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

As proposed originally by Arikan [3], successive cancellation (SC) decoding is employed for polar codes, which results in $O(n \log n)$ decoding complexity. The performance can be substantially improved by adopting SC list (SCL) decoding [10], [11], [12], [13], which accepts as many as L paths, rather than just one path in SC decoding, and hence its resultant computational complexity is L times larger than SC decoding. In an extreme case with $L = 2^k$, SCL can achieve the performance of MLD.

C. POLAR CODES WITH DYNAMIC FROZEN BITS

The small minimum pairwise Hamming distance of polar codes with small blocklength makes the design less attractive in the small blocklength regime. In order to resolve the issue, a recent contribution from [5] proposed to dynamically generate the frozen bits according to the message bits in a way that can result in a large minimum pairwise distance. Specifically, by incorporating an auxiliary linear code with specific $k \times n$ upper trapezoidal generator matrix M_{DF}, codewords of a polar code can be generated via $\boldsymbol{c}_{\mathrm{p}} = \boldsymbol{u}G_{\mathrm{p}} = \boldsymbol{m}_{\mathrm{p}} M_{\mathrm{DF}}\, G_{\mathrm{p}}$, based on which the dynamic frozen bits can be determined by its previous bits. An example of dynamic frozen bits that are determined based on an extended BCH code is provided in [5], which has been shown to yield a large minimum distance and an excellent performance. Another example is that in CRC-aided polar codes, the CRC check bits are generated according to previous message bits, which can also be regarded as a setup of dynamic frozen bits [5].

III. TRANSFORMATION FROM BLBCS TO POLAR CODES

In this section, a procedure that transforms a BLBC to an equivalent polar code with dynamic frozen bits is established, based on which the soft decoding algorithms of polar codes can be applied to decoding the BLBC.

Proposition 1: For an (n, k) BLBC $\mathcal{C}$ and a permutation matrix P, there exists a polar code $\mathcal{C}_{\mathrm{p}}$ with dynamic frozen bits such that the one-to-one connection between codeword $\boldsymbol{c}$ in $\mathcal{C}$ and codeword $\boldsymbol{c}_{\mathrm{p}}$ in $\mathcal{C}_{\mathrm{p}}$ is given by $\boldsymbol{c} = \boldsymbol{c}_{\mathrm{p}}P$.

Proof: Let G be a generator matrix of BLBC $\mathcal{C}$. Noting that G_{p} is an $n \times n$ full rank matrix, we can rewrite the code space of this (n, k) BLBC as

$$\begin{aligned} \mathcal{C} &\triangleq \left\{ \boldsymbol{c} = \boldsymbol{m}G | \boldsymbol{m} \in \mathbb{F}_2^k \right\} \\ &= \left\{ \boldsymbol{c} = \boldsymbol{m}\left(E^{-1}E\right) G \left(P^{-1}G_{\mathrm{p}}^{-1}G_{\mathrm{p}}P\right) | \boldsymbol{m} \in \mathbb{F}_2^k \right\}, \end{aligned}$$

where E is the (invertible) elimination matrix that transforms $GP^{-1}G_{\mathrm{p}}^{-1}$ into reduced row echelon form. Hence, $EGP^{-1}G_{\mathrm{p}}^{-1}$ is upper-trapezoidal and can be regarded as M_{DF}. As a result, the code space $\mathcal{C}$ can be further rewritten as

$$\begin{aligned} \mathcal{C} &= \left\{ \boldsymbol{c} = \left(\boldsymbol{m}E^{-1}\right)\left(EGP^{-1}G_{\mathrm{p}}^{-1}\right)G_{\mathrm{p}}P | \boldsymbol{m} \in \mathbb{F}_2^k \right\} \\ &= \left\{ \boldsymbol{c} = \left(\boldsymbol{m}_{\mathrm{p}}\, M_{\mathrm{DF}}\, G_{\mathrm{p}}\right)P | \boldsymbol{m}_{\mathrm{p}} \in \mathbb{F}_2^k \right\} \\ &= \left\{ \boldsymbol{c} = \boldsymbol{c}_{\mathrm{p}}P | \boldsymbol{c}_{\mathrm{p}} \in \mathcal{C}_{\mathrm{p}} \right\}, \end{aligned}$$

where $\boldsymbol{m}_{\mathrm{p}} \triangleq \boldsymbol{m}E^{-1}$ and $\boldsymbol{c}_{\mathrm{p}} \triangleq \boldsymbol{m}_{\mathrm{p}}M_{\mathrm{DF}}\, G_{\mathrm{p}}$.[1] The one-to-one permutation-based connection between codeword $\boldsymbol{c}$ in $\mathcal{C}$ and codeword $\boldsymbol{c}_{\mathrm{p}}$ in $\mathcal{C}_{\mathrm{p}}$ is therefore established with $\mathcal{C}_{\mathrm{p}}$ being the transformed polar code with dynamic frozen bits. ■

With Proposition 1, we can transform the decoding space from $\mathcal{C}$ to $\mathcal{C}_{\mathrm{p}}$ by performing the inverse permutation P^{-1} onto the noisy received vector at the receiver and then decode $\hat{\boldsymbol{m}}_{\mathrm{p}}$ using a decoder tailored for polar codes. The estimate of $\boldsymbol{m}$ can then be recovered via $\hat{\boldsymbol{m}} = \hat{\boldsymbol{m}}_{\mathrm{p}}E$. Note that for given G and P (as well as G_{p}), the corresponding elimination matrix E can be obtained from the Gauss-Jordan elimination with amortized cost. One can thus focus on devising a suitable permutation matrix P that facilitates the polar decoding of $\mathcal{C}_{\mathrm{p}}$.

A block diagram of the proposed polar decoding system is given in Fig. 2. Although $E^{-1}, E, P^{-1}, G_{\mathrm{p}}^{-1}, G_{\mathrm{p}}$ and P are depicted in the encoding process in Fig. 2, only the BLBC encoder is necessarily realized because $E^{-1}E$ and $P^{-1}G_{\mathrm{p}}^{-1}G_{\mathrm{p}}P$ are both identity matrices. Hence, an advantage of the proposed polar decoding system is that no adjustment on the usual encoding procedure of BLBCs is needed. As a result, the system in Fig. 2 still has the freedom to switch back to a traditional BLBC decoder, as illustrated in the upper path of the decoding process in Fig. 2.

On the other hand, the lower path of the decoding process shows that we can alternatively perform P^{-1} at the receiver and then use a polar code decoder to decode the BLBC. As

[1] It is worth mentioning that since E is invertible, both generator matrices G and $G' \triangleq EG$ generate the same BLBC, and both G and G' fulfill $GH = G'H (= EGH) = \mathbf{0}$, where H is a parity-check matrix of this BLBC. Therefore, E can be absorbed into the determination of $M_{\mathrm{DF}} (= EGP^{-1}G_{\mathrm{p}}^{-1}) = G'P^{-1}G_{\mathrm{p}}^{-1}$ when the generator matrix G' is assumed to be adopted in the proof. Since a particular generator matrix G such as a systematic one may be required in certain applications, a general proof for arbitrary G is provided in Proposition 1.

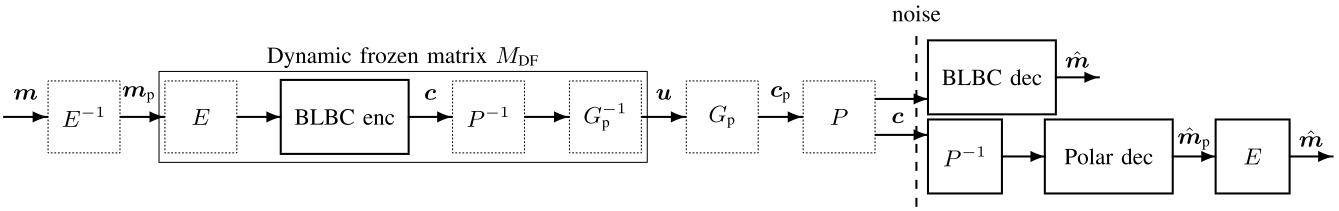


FIGURE 2. Block diagram of the proposed polar decoding. Note that only the BLBC encoding process in the solid box is required at the transmitter, and the operations in dashed boxes are unnecessary because $E^{-1}E = P^{-1}G_{\mathrm{p}}^{-1}G_{\mathrm{p}}P = I$. At the decoder end, the proposed system offers an alternative path, where a soft-decoding algorithm that is tailored for polar codes can be used.

long as the channel is memoryless, the statistical pattern of the noise samples will not be affected by the permutation operation; hence, the same recursive computations of soft information for polar code decoder such as SCL can be applied [10], [11], [12], [13], provided that the locations and relations of information bits and dynamic frozen bits can be correctly regained via the chosen P. The output $\hat{\boldsymbol{m}}_{\mathrm{p}}$ of the polar code decoder can then be used to recover an estimate of the message via $\hat{\boldsymbol{m}} = \hat{\boldsymbol{m}}_{\mathrm{p}}E$.

When the polar code decoder adopted guarantees a maximum-likelihood (ML) decoding output, the proposed transformation renders an ML decoding system for BLBCs, regardless of the choice of P. Yet, when a suboptimal (or even near-optimal) decoder for polar codes is employed, the decoding performance varies with P. A question that follows is how to find a permutation matrix P that guarantees a good polar decoding performance with respect to a suboptimal polar code decoder. To this end, the decoding performance corresponding to a chosen P may be a straightforward criterion for computer search of an acceptably good P. However, since obtaining the decoding error probability via system simulations for each candidate P is operationally intensive, the next proposition provides an alternative criterion that is much more numerically efficient and hence will be adopted in this paper.

Proposition 2: When the locations of dynamic frozen bits are decided via $M_{\mathrm{DF}} = EGP^{-1}G_{\mathrm{p}}^{-1}$ after fixing a permutation matrix P, the probability of block errors due to SC decoding is upper-bounded by the sum of Bhattacharyya parameters of the information bits.

Proof: This is a consequence of [3, Proposition 2]. ■

IV. SOFT POLAR DECODING OF BCH CODES, GOLAY CODES AND LDPC CODES

In this section, we examine the proposed transformation and its respective polar decoding for BCH codes (Example 1), Golay codes (Example 2) and LDPC codes (Example 3).

Example 1: Consider the (16, 7) extended BCH code with parity check matrix given by

$$H_{\mathrm{eBCH}} = \begin{bmatrix} \vec{1} & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 & \alpha^7 & \alpha^8 & \alpha^9 & \alpha^{10} & \alpha^{11} & \alpha^{12} & \alpha^{13} & \alpha^{14} & \vec{0} \\ \vec{1} & \alpha^3 & \alpha^6 & \alpha^9 & \alpha^{12} & \vec{1} & \alpha^3 & \alpha^6 & \alpha^9 & \alpha^{12} & \vec{1} & \alpha^3 & \alpha^6 & \alpha^9 & \alpha^{12} & \vec{0} \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}^{\mathrm{T}}, \tag{1}$$

where in order to differentiate from 0 and 1 in $\mathbb{F}_2$, we use $\vec{0}$ and $\vec{1} = \alpha^0$ to denote respectively the additive and multiplication identities in $\mathbb{F}_{2^4}$. Here, $\alpha \triangleq [0\ 1\ 0\ 0]^{\mathrm{T}} = 2$ [1, Tab. 2.8] is a primitive element of $\mathbb{F}_{2^4}$. Set the permutation matrix P as[2]:

$$P_{i,j} = \begin{cases} 1, & i = n-1-\alpha^j \text{ for } 0 \le j \le n-2 \\ & \text{or} i = j = n-1; \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

With the above choice of P, we perform the Gauss-Jordan elimination upon $GP^{-1}G_{\mathrm{p}}^{-1}$ for the systematic generator matrix G and $G_{\mathrm{p}} = F_2^{\otimes 4}$, and obtain

$$E = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

One can verify that

$$M_{\mathrm{DF}} = EGP^{-1}G_{\mathrm{p}}^{-1}$$
$$= \begin{bmatrix} 0&0&0&1&0&0&1&0&0&0&1&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0&0&1&1&0&1&0&0&0 \\ 0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1 \end{bmatrix}$$

is an upper-trapezoidal matrix and can be used to define the BLBC-equivalent polar code with dynamic frozen bits. Specifically, $\boldsymbol{m}_{\mathrm{p}}M_{\mathrm{DF}} = \boldsymbol{u}$ implies that u_0, u_1, u_2, u_4 and u_8 must be zero frozen bits, and u_3, u_5, u_7, u_{11}, u_{13}, u_{14} and u_{15} are message bits because the 4th, 6th, 8th, 12th, 14th, 15th and 16th columns of M_{DF} only consist of a single one. The remaining bits are dynamic frozen bits whose values are decided by the message bits with smaller indices.

Alternatively, we can define this BLBC-equivalent polar code from the perspective of the parity-check matrix. For a given parity-check matrix H_{eBCH} of the (16, 7) extended BCH code, the codeword $\boldsymbol{c}$ must satisfy

$$\mathbf{0} = \boldsymbol{c}H_{\mathrm{eBCH}}F = \boldsymbol{m}E^{-1}\underbrace{EGP^{-1}G_{\mathrm{p}}^{-1}}_{M_{\mathrm{DF}}}G_{\mathrm{p}}PH_{\mathrm{eBCH}}F$$

[2] Here we index the entries of permutation matrix P from 0 to $n-1$ (rather than from 1 to n).

$$
\begin{aligned}
&= \underbrace{\boldsymbol{m}_{\mathrm{p}} M_{\mathrm{DF}}}_{\boldsymbol{u}} G_{\mathrm{p}} P H_{\mathrm{eBCH}} F \\
&= \boldsymbol{u} \begin{bmatrix}
0&0&0&0&0&0&0&0&1\\
0&0&0&0&0&0&0&1&0\\
0&0&0&0&0&0&1&0&0\\
1&1&0&0&1&0&0&0&0\\
0&0&0&0&0&1&0&0&0\\
1&1&1&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0\\
0&0&0&1&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0
\end{bmatrix}, \qquad (3)
\end{aligned}
$$

where

$$
F = \begin{bmatrix}
0&1&0&1&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0\\
1&1&0&0&1&0&1&0&0\\
0&0&1&0&0&0&0&1&0\\
0&1&0&0&1&0&0&0&0\\
0&0&1&0&0&0&0&0&0\\
1&1&1&0&1&0&0&0&0\\
0&0&1&0&1&0&0&0&0\\
1&1&0&1&0&1&1&1&1
\end{bmatrix}
$$

is the (invertible) elimination matrix that transforms $G_{\mathrm{p}}PH_{\mathrm{eBCH}}$ into *backward* reduced row echelon form.[3] The polar code that has the same code space as the $(16, 7)$ extended BCH code is accordingly specified as follows. First, we know from the 4th, 6th, 7th, 8th and 9th columns of $G_{\mathrm{p}}PH_{\mathrm{eBCH}}F$ that u_0, u_1, u_2, u_4 and u_8 must be zero frozen bits. Second, as the 8th, 12th, 14th, 15th, 16th rows of $G_{\mathrm{p}}PH_{\mathrm{eBCH}}F$ contain only zero entries, u_7, u_{11}, u_{13}, u_{14} and u_{15} can be arbitrary; so, they correspond to message bits. Third, we have to comply with the 1st, 2nd, 3rd and 5th columns of $G_{\mathrm{p}}PH_{\mathrm{eBCH}}F$, which dictates

$$
\begin{cases}
u_3 \oplus u_5 \oplus u_{12} = 0\\
u_3 \oplus u_5 \oplus u_{10} = 0\\
u_5 \oplus u_9 = 0\\
u_3 \oplus u_6 = 0
\end{cases} \qquad (4)
$$

Designating the one with the largest index in each of the four equations in (4) to be the dynamic frozen bit, which is guaranteed to be distinct for all four equations because $G_{\mathrm{p}}PH_{\mathrm{eBCH}}F$ is in backward row reduced echelon form, we obtain that u_3 and u_5 are message bits and u_6, u_9, u_{10} and u_{12} are dynamic frozen bits. The BLBC-equivalent polar code is therefore established.

Note that the same code has been used in [5] to demonstrate how the dynamic frozen bits can be obtained. Here, we reuse it and show that through the identification of P, the receiver can decode the extended BCH code by combining the operation of multiplying P^{-1} with a soft polar code decoder (see Fig. 2).

In Section V, we will examine the polar decoding performances with respect to the parity check matrix and permutation matrix similarly specified as in (1) and (2) via primitive element $\alpha = 2$ for extended BCH codes of length $n = 64$.

Complementing to the proof of Proposition 1, Eq. (3) in Example 1 provides an alternative way to determine the BLBC-equivalent polar code with dynamic frozen bits. In other words, one can identify the BLBC-equivalent polar code from the parity-check matrix H of the BLBC according to

$$
\boldsymbol{u} G_{\mathrm{p}} P H F = \boldsymbol{0},
$$

where the indices of dynamic frozen bits are determined by the last non-zero components of multiple-one columns of $G_{\mathrm{p}}PHF$, and the zero frozen bits are those corresponding to the non-zero component of single-one columns of $G_{\mathrm{p}}PHF$.

In the next two examples, we will take the parity-check matrix perspective when devising the BLBC-equivalent polar codes.

Example 2: We turn to the $(24, 12)$ extended Golay code. It comprises the $(23, 12)$ Golay code with generator polynomial $\mathrm{x}^{11} + \mathrm{x}^{10} + \mathrm{x}^{6} + \mathrm{x}^{5} + \mathrm{x}^{4} + \mathrm{x}^{2} + 1$ and with an extra bit equal to the exclusive-or sum of all the previous bits. Its parity check matrix is given by

$$
H_{\mathrm{eGolay}} = \begin{bmatrix}
1&1&0&0&0&1&1&1&0&1&0&1\\
0&1&1&0&0&0&1&1&1&0&1&1\\
1&1&1&1&0&1&1&0&1&0&0&0\\
0&1&1&1&1&0&1&1&0&1&0&0\\
0&0&1&1&1&1&0&1&1&0&1&0\\
1&1&0&1&1&0&0&1&1&0&0&1\\
0&1&1&0&1&1&0&0&1&1&0&1\\
0&0&1&1&0&1&1&0&0&1&1&1\\
1&1&0&1&1&1&0&0&0&1&1&0\\
1&0&1&0&1&0&0&1&0&1&1&1\\
1&0&0&1&0&0&1&1&1&1&1&0\\
1&0&0&0&1&1&1&0&1&0&1&1\\
1&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&1&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0\\
0&0&0&0&0&0&0&1&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&0&0&0&0&1&0\\
0&0&0&0&0&0&0&0&0&0&0&1
\end{bmatrix}.
$$

[3] A matrix A is said to be in backward reduced row echelon form if AJ is in reduced row echelon form, where J is the backward identity matrix.

Consider the multi-kernel polar code with generator matrix:

$$G_{\mathrm{p}} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix},$$

and let the permutation matrix P be chosen as

$$P = \begin{bmatrix}
0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0 \\
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0 \\
0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0 \\
0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1 \\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0 \\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0
\end{bmatrix}.$$

The dynamic frozen bits can thus be determined by $\boldsymbol{u}G_{\mathrm{p}}PH_{\mathrm{eGolay}}F = \mathbf{0}$ with

$$F = \begin{bmatrix}
1&0&0&1&1&1&1&1&1&1&1&1 \\
1&1&0&1&1&0&1&1&0&0&0&0 \\
1&0&0&1&0&0&0&0&1&1&1&1 \\
0&1&0&1&0&0&0&0&1&1&0&1 \\
1&0&0&1&1&1&1&1&0&0&0&0 \\
0&1&1&1&0&0&0&0&0&1&1&1 \\
0&0&1&1&0&0&0&0&1&0&0&1 \\
0&1&0&1&1&1&0&1&0&0&0&0 \\
0&1&1&1&0&1&1&1&0&0&0&0 \\
1&1&0&1&0&0&0&0&1&0&1&1 \\
1&1&1&1&0&0&0&0&0&1&0&1 \\
1&0&1&1&0&0&0&0&0&0&1&1
\end{bmatrix}.$$

Note that although targeting the same Golay code, the proposed decoder in this work and the one proposed in [7] are different.

Example 3: In [14], Gallager introduces the following parity-check matrix to elucidate the design principle of LDPC codes:

$$H_{\mathrm{LDPC}} = \begin{bmatrix}
1&0&0&0&1&0&0&0&1&0&0&0 \\
1&0&0&0&0&1&0&0&0&1&0&0 \\
1&0&0&0&0&0&1&0&0&0&1&0 \\
1&0&0&0&0&0&0&1&0&0&0&1 \\
0&1&0&0&1&0&0&0&0&0&0&1 \\
0&1&0&0&0&1&0&0&1&0&0&0 \\
0&1&0&0&0&0&1&0&0&1&0&0 \\
0&1&0&0&0&0&0&1&0&0&1&0 \\
0&0&1&0&1&0&0&0&0&0&1&0 \\
0&0&1&0&0&1&0&0&0&0&0&1 \\
0&0&1&0&0&0&1&0&1&0&0&0 \\
0&0&1&0&0&0&0&1&0&1&0&0 \\
0&0&0&1&1&0&0&0&0&1&0&0 \\
0&0&0&1&0&1&0&0&0&0&1&0 \\
0&0&0&1&0&0&1&0&0&0&0&1 \\
0&0&0&1&0&0&0&1&1&0&0&0
\end{bmatrix}.$$

Because H_{LDPC} does not have full rank, it corresponds actually to a $(16, 7)$ code. For $G_{\mathrm{p}} = F_2^{\otimes 4}$, we search over 10^7 permutation matrices that randomly drawn from $16! \approx 2 \cdot 10^{13}$ possibilities, and adopt the one with the smallest Bhattacharyya parameter sum of the information bits, which is given by

$$P = \begin{bmatrix}
0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0 \\
0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0 \\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0 \\
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0 \\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1 \\
0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0
\end{bmatrix}.$$

With the elimination matrix

$$F = \begin{bmatrix}
1&0&0&1&0&0&0&1&1&1&0&0 \\
1&0&1&0&1&1&1&0&1&1&1&1 \\
0&0&0&0&1&0&0&0&1&1&0&0 \\
0&1&1&1&0&1&1&1&1&1&1&1 \\
0&0&0&0&1&1&1&1&0&1&1&0 \\
0&0&1&0&1&0&0&0&0&1&0&1 \\
0&1&1&1&0&1&1&0&0&1&1&0 \\
1&0&0&1&0&0&0&1&0&1&0&1 \\
1&0&0&0&0&1&0&0&0&0&1&0 \\
1&0&0&1&0&0&0&1&0&0&0&1 \\
0&0&0&0&0&0&0&0&0&0&1&0 \\
0&0&0&0&0&0&0&0&0&0&0&1
\end{bmatrix},$$

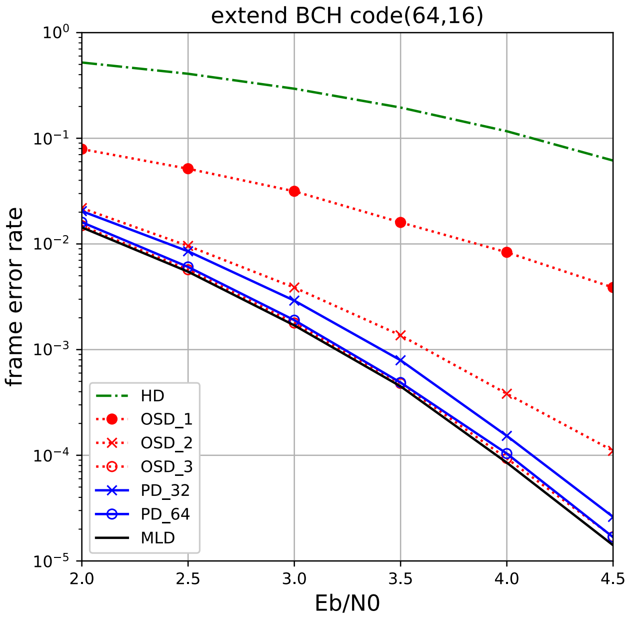


FIGURE 3. FERs for the (64, 16) extended BCH code.

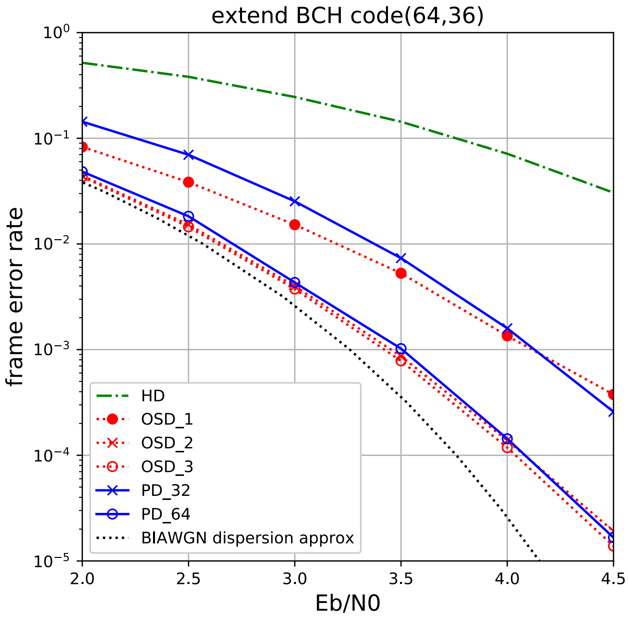


FIGURE 4. FERs for the (64, 36) extended BCH code.

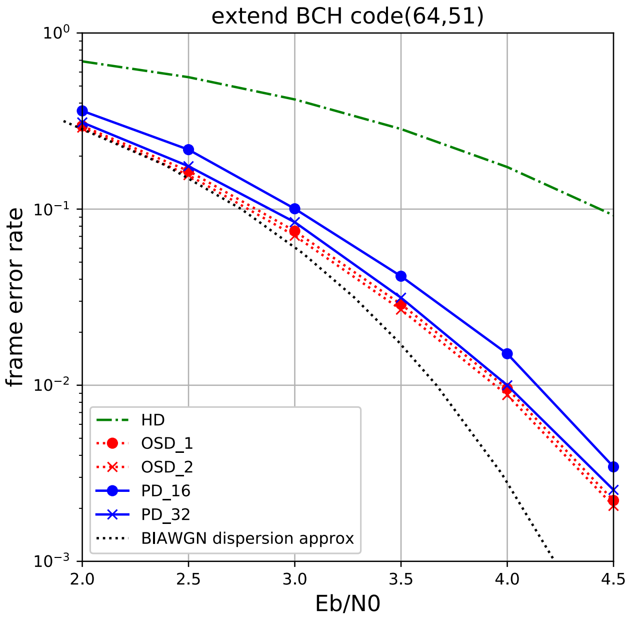


FIGURE 5. FERs for the (64, 51) extended BCH code.

the LDPC-equivalent polar code is defined. This LDPC code can thus be polar-decoded based on the proposed structure in Fig. 2.

We close this section by remarking that although the computation of the Bhattacharyya parameter sum of the information bits for a given permutation matrix P is numerically efficient, it is still operationally infeasible to examine the Bhattacharyya parameter sums of the information bits for *all* candidate permutation matrices for the above examples, except for the (16, 7) extended BCH code in Example 1. Statistically brute-force searching was used instead (as stated in Example 3). The decoding error performances of the permutation matrices that result in the smallest Bhattacharyya parameter sum among all tested ones will be examined in the next section.

V. SIMULATION RESULTS

In this section, frame error rates (FERs), obtained by three soft-decision decoders, i.e., MLD, OSD and the proposed SCL polar decoder, for antipodally transmitting a BLBC over the AWGN channel will be accounted via simulations. In what follows, we will abbreviate OSD with order r and SCL polar decoding with list size L as OSD_r and PD_L, respectively. Also shown are the FERs of a hard-decision bounded distance decoder, abbreviated as HD. As a reference, for an (n, k) BLBC, the decoding complexity of OSD_r is $O(nk^2+2^r nk)$ [2], while that of PD_L is $O(Ln\log(n))$ [10], [11], [12]. Furthermore, when performing PD_L, the survivor path with the best metric in the L-list will be outputted as the final decision [5].

Based on Example 1, we simulate three length-64 extended BCH codes of rates $\frac{1}{4}$, $\frac{9}{16}$ and $\frac{51}{64}$ in Figs. 3–5, respectively. The same permutation matrix as specified in (2) is adopted in their SCL polar decoding. This reflects a side benefit for the proposed SCL polar decoding, namely, the same receiver structure can be used for multiple codes of equal length but of different rates. Notably, as the columns that participate in the Gaussian elimination of OSD vary with the generator matrices, which are apparently different for the three eBCH codes of different rates, a code-by-code adjustment needs to be performed for OSD.

Three observations are made from Figs. 3–5. First, it can be observed from Fig. 3 that PD_32 performs better than OSD_2, and both PD_64 and OSD_3 achieve the MLD performance. Second, unable to obtain the MLD

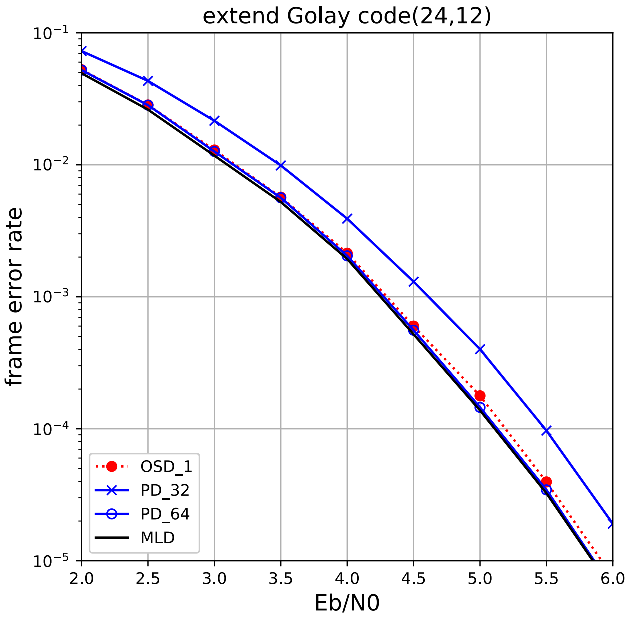


FIGURE 6. FERs for the (24, 12) extended Golay code.

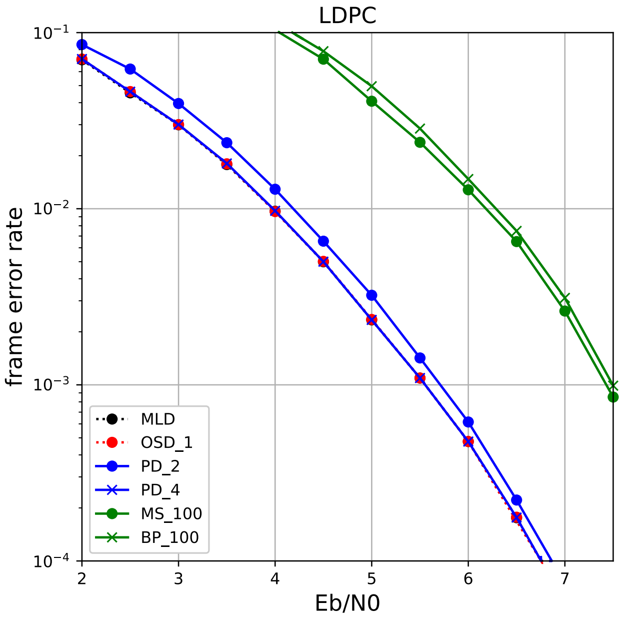


FIGURE 7. FERs for the (16, 7) LDPC code.

performance due to its excessive complexity in Fig. 4, we use the saturated performance of OSD_3 as a reference. The Polyanskiy-Poor-Verdú meta converse of the ML decoding [15] is also included as an alternative reference, where the bi-awgn MATLAB code devised based on the *normal* approximation in [16] is used. We observe that PD_64 has already achieved the saturated performance of OSD_3, and as inferred from the meta converse, is at most 0.5 dB away from the MLD at FER $= 10^{-4}$. Third, in Fig. 5, we can no longer obtain the performance of OSD_3 due to its intensive complexity. Noting that OSD_2 has a comparable (and hence saturated) performance to OSD_1, we observe from Fig. 5 that PD_32 has achieved the saturated performance of OSD_2, and the meta converse is much deviated from the performances of both PD_32 and OSD_2 at high rate.

Based on Example 2, FERs for the (24, 12) extended Golay code are plotted in Fig. 6. Although both OSD_1 and PD_64 achieve the MLD performance in this figure, OSD_1 requires extra effort to monitor its decoding flow according to the sorting results of symbol reliabilities.

Finally, Fig. 7 illustrates FERs for the (16, 7) LDPC code in Example 3. We observe that the performances of MLD, OSD_1 and PD_4 coincide to each other, and all three decoders significantly outperform the belief-propagation (BP) and min-sum (MS) decoding algorithms with 100 iterations, which are abbreviated respectively as BP_100 and MS_100 in the figure. It should be pointed out that BP and MS decodings are inherently unsuitable for a code with such a short blocklength as the corresponding graph is not sparse enough and contains a lot of short cycles. Hence, it is not entirely fair to compare BP and MS decodings with the proposed one here. The main purpose of this simulation is to demonstrate that the proposed technique can be used to transform codes other than BCH and Golay codes.

VI. CONCLUDING REMARK

For certain BLBCs, it could happen that none of the permutation matrices can give a better sum of Bhattacharyya parameters of the information bits than that before polarization. As an example, consider a (8, 3) code with generator matrix:

$$G = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}.$$

Under a target raw bit error rate (BER) of 0.01 over AWGN channels, the sum of Bhattacharyya parameters of three information bits before polarization is $0.198997 \times 3 = 0.596991$. However, among $8! = 40,320$ possible choices of permutation matrices, the minimum Bhattacharyya-parameter sum of the three information bits is $0.830539 + 0.149237 + 0.003134 = 0.98291$, which is much larger than 0.596991. In particular, all permutation matrices are forced to include the bit corresponding to the first row of $G_{\mathrm{p}} = F_2^{\otimes 4}$ into the set of information bits, of which the Bhattacharyya parameter is already as large as 0.830539. This brings up two future directions of this study: *i*) how to devise a systematic and efficient approach for the selection of well-performed permutation matrices for codes of larger sizes, and *ii*) for what class of BLBCs the proposed transformation cannot be well applied from the aspect of having a better Bhattacharyya-parameter sum than that before polarization.

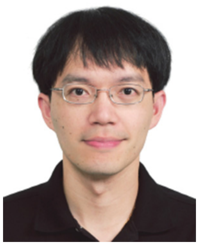
REFERENCES

[1] S. Lin and D. J. Costello *Error Control Coding: Fundamentals and Applications*, 2nd ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 2004.

[2] M. P. C. Fossorier and S. Lin, “Soft-decision decoding of linear block codes based on ordered statistics,” *IEEE Trans. Inf. Theory*, vol. 41, no. 5, pp. 1379–1396, Sep. 1995.
[3] E. Arikan, “Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels,” *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3051–3073, Jul. 2009.
[4] F. Gabry, V. Bioglio, I. Land, and J.-C. Belfiore, “Multi-kernel construction of polar codes,” in *Proc. IEEE Int. Conf. Commun. Workshops (ICC)*, Paris, France, May 2017, pp. 761–765.
[5] P. Trifonov and V. Miloslavskaya, “Polar codes with dynamic frozen symbols and their decoding by directed search,” in *Proc. IEEE Inf. Theory Workshops (ITW)*, Sevilla, Spain, Sep. 2013, pp. 1–5.
[6] P. Trifonov and V. Miloslavskaya, “Polar subcodes,” *IEEE J. Sel. Areas Commun.*, vol. 34, no. 2, pp. 254–266, Feb. 2016.
[7] V. Bioglio and I. Land, “Polar-code construction of Golay codes,” *IEEE Commun. Lett.*, vol. 22, no. 3, pp. 466–469, Mar. 2018.
[8] E. Arikan, “A performance comparison of polar codes and reed-muller codes,” *IEEE Commun. Lett.*, vol. 12, no. 6, pp. 447–449, Jun. 2008.
[9] E. Arikan, “A survey of reed-muller codes from polar coding perspective,” in *Proc. IEEE Inf. Theory Workshop (ITW)*, Cairo, Egypt, Jan. 2010, pp. 1–5.
[10] I. Tal and A. Vardy, “List decoding of polar codes,” *IEEE Trans. Inf. Theory*, vol. 61, no. 5, pp. 2213–2226, May 2015.
[11] K. Chen, K. Niu, and J. Lin, “List successive cancellation decoding of polar codes,” *Electron. Lett.*, vol. 48, no. 9, pp. 500–501, Apr. 2012.
[12] B. Li, H. Shen, and D. Tse, “An adaptive successive cancellation list decoder for polar codes with cyclic redundancy check,” *IEEE Commun. Lett.*, vol. 16, no. 12, pp. 2044–2047, Dec. 2012.
[13] K. Niu and K. Chen, “CRC-aided decoding of polar codes,” *IEEE Commun. Lett.*, vol. 16, no. 10, pp. 1668–1671, Oct. 2012.
[14] R. G. Gallager, *Low-Density Parity-Check Codes*. Cambridge, MA, USA: MIT Press, 1963.
[15] Y. Polyanskiy, H. V. Poor, and S. Verdú, “Channel coding rate in the finite blocklength regime,” *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2307–2359, May 2010.
[16] T. Erseghe, “Coding in the finite-blocklength regime: Bounds based on Laplace integrals and their asymptotic approximations,” *IEEE Trans. Inf. Theory*, vol. 62, no. 12, pp. 6854–6883, Dec. 2016. [Online]. Available: https://github.com/yp-mit/spectre

CHIEN-YING LIN received the B.S. degree from the Department of Mathematics, National Tsing Hua University, Taiwan, in 2011, and the M.S. degree from the Department of Communication Engineering, National Taipei University, Taiwan, in 2015. He is currently pursuing the Ph.D. degree with the Institute of Communications Engineering, National Chiao Tung University, Taiwan; the research topic is about polar codes and their applications. In 2015, he joined Realtek Semiconductor Corporation, where he was involved in designing error correction coding algorithms in storage, e.g., LDPC and polar codes.

YU-CHIH HUANG (Member, IEEE) received the Ph.D. degree in electrical and computer engineering from Texas A&M University in 2013, where he was a Postdoctoral Research Associate from 2013 to 2015. In 2015, he joined the Department of Communication Engineering, National Taipei University, Taiwan, as an Assistant Professor and was promoted to an Associate Professor in 2018. Since 2020, he has been with the Institute of Communications Engineering, National Chiao Tung University, Taiwan, where he is currently an Associate Professor. His research interests are in information theory, coding theory, wireless communications, and statistical signal processing. He received the 2018 IEEE Information Theory Society Taipei Chapter and IEEE Communications Society Taipei/Tainan Chapter’s Best Paper Award for Young Scholars. He is a recipient of the MOST Young Scholar Fellowship from 2020 to 2023. He is currently serving as an Editor for IEEE COMMUNICATIONS LETTERS.



SHIN-LIN SHIEH received the B.Sc. and M.Sc. degrees in electrical engineering from National Tsinghua University, Hsinchu, Taiwan, in 1999 and 2001, respectively, and the Ph.D. degree from the Department of Communications Engineering, National Chiao Tung University, Hsinchu. He joined the Industrial Technology Research Institute, Taiwan, in 2002, where he was engaged in several projects on the development of baseband communication algorithms. In 2005, he joined Sunplus Technology Company Ltd., where he was involved in designing baseband algorithms of EDGE, WCDMA, HSDPA, LTE, and other communication systems. In 2010, he joined the Department of Communication Engineering, National Taipei University, New Taipei City, Taiwan, where he is currently an Associate Professor.

PO-NING CHEN (Senior Member, IEEE) was born in Taipei in 1963. He received the B.S. and M.S. degrees in electrical engineering from National Tsing Hua University (NTHU), Taiwan, in 1985 and 1987, respectively, and the Ph.D. degree in electrical engineering from the University of Maryland at College Park, College Park, in 1994.

From 1985 to 1987, he was with NTHU Image Processing Laboratory, where he worked on the recognition of printed Chinese characters. In 1989, he was with Star Tech. Inc., where he focused on the development of fingerprint recognition systems. Upon the reception of the Ph.D. degree in 1994, he joined Wan Ta Technology Inc., as a Vice General Manager, manufacturing products for retail point-of-sale systems. In 1995, he became a member of the Research Staff with the Advanced Technology Center, Computer and Communication Laboratories, Industrial Technology Research Institute, Taiwan, where he led a project on Java-based Network Managements. Since 1996, he has been an Associate Professor with the Department of Communications Engineering, National Chiao Tung University (NCTU), Taiwan, and became a Full Professor in 2001. In 2017, he became the Associate Dean of the College of Electrical and Computer Engineering, NCTU. His research interests lie in information and coding theory, large deviations theory, distributed detection, and sensor networks.

Dr. Chen received the award for Junior Research Investigators, Academia Sinica, in 2000. He was selected as an NCTU Outstanding Tutor Teacher in 2002, 2013, and 2014. He was also the recipient of the Distinguished Teaching Award from the College of Electrical and Computer Engineering, NCTU, in 2003 and 2014. He was the Chair of the IEEE Communications Society (ComSoc) Taipei Chapter in 2006, and the IEEE ComSoc Taipei Chapter received the IEEE ComSoc Chapter Achievement Awards and IEEE ComSoc Chapter of the Year in the following year. From 2007 to 2009, he has served as the Chairman of the Department of Communications Engineering, NCTU. From 2012 to 2015, he was the Associate Chief Director of NCTU Microelectronics and Information Systems Research Center, Taiwan.