

# BCH Based U-UV Codes and Its SCL Decoding

Wenhao Chen , Jinjun Cheng , Changyu Wu , Li Chen , *Senior Member, IEEE*,  
and Huazi Zhang , *Senior Member, IEEE*

**Abstract**—U-UV codes are constructed by a number of component codes in the  $(U|U+V)$  recursive structure, where the U codes and V codes are component codes. This construction is known as the Plotkin construction and the U-UV codes are also known as the generalized concatenated codes with inner polar codes. This paper proposes U-UV codes with primitive BCH component codes as a pursuit of designing competent short-to-medium length codes for future ultra low-latency communications. The U-UV code design considers both the finite length rate of the subchannels and the equal error probability rule, yielding a good performing U-UV code that is designed for a targeted transmission rate. The successive cancellation list (SCL) decoding and its complexity reduction variant are proposed to maximize the code's performance. Their decoding complexity and latency are analyzed. Decoding performance of the U-UV codes is further studied, showing that SCL decoding of the U-UV codes can approach its approximated maximum likelihood (ML) decoding bound. They can outperform other competent short-to-medium length codes, including polar codes, BCH codes and tail-biting convolutional (TBC) codes.

**Index Terms**—Generalized concatenated codes, Plotkin construction, successive cancellation list decoding, U-UV codes.

## I. INTRODUCTION

**F**UTURE communication systems do not only require ultra reliable information transmission, but also ultra low latency. Competent short-to-medium length channel codes will play an important role. Modern codes, including turbo codes [1], low-density parity-check (LDPC) codes [2] and polar codes [3], can produce a capacity approaching performance. Among them, polar codes have been proven to achieve the capacity of the binary input symmetric discrete memoryless channel (DMC). However, their advanced error-correction capabilities

are realized based on a large codeword length, which inevitably incurs a large decoding latency. In the short-to-medium length regime, BCH codes [4], [5], tail-biting convolutional (TBC) codes [6] and polar codes [3] are known to be the competent candidates [7]. This paper introduces another promising candidate, the U-UV codes. They are produced by the  $(U|U+V)$  recursive construction, where the U codes and the V codes are component codes. This construction was first proposed by Plotkin [8]. Hence, it is also known as the Plotkin construction. For simplicity, we use U-UV to refer the coding structure and call them the U-UV codes. The U-UV construction can be extended recursively by involving more component codes, resulting in a longer U-UV code.

It is known that both Reed-Muller (RM) codes [9], [10] and polar codes [3] can be interpreted by the U-UV structure. The RM codes have also been proven to achieve the capacity of erasure channels [11]. Polar codes are founded based on channel polarization where the U-UV construction leads to subchannels with polarized capacities. Consequently, the coded transmission can be designed accordingly. That says the subchannels with a capacity close to 1 will be used to transmit the information bits (known as the *unfrozen* bits), and the remaining subchannels will be used to transmit the redundancy (known as the *frozen* bits). Portion of information bits is the code rate. It has been shown that when the codeword length  $n$  is sufficiently large, successive cancellation (SC) decoding of polar codes can achieve the channel capacity with a decoding complexity of  $O(n \log_2 n)$  [3]. However, when the codeword length is limited, channel polarization is incomplete. There exists a significant portion of subchannels without a polarized capacity. This will downgrade the effectiveness of the above mentioned bit wise transmission assignment. To improve the decoding performance, the SC list (SCL) decoding [12], [13] has been proposed. However, it is known that the SC decoding inherits a large latency. For this, the multi-bit SC decoding have been proposed [14], [15]. They have been further explored for reducing the SCL decoding latency [16], [17], [18], [19], [20], [21], [22]. To compensate the effect of incomplete polarization, the polarization-adjusted convolutional (PAC) codes [23] were recently proposed. The additional convolutional transform enables a better utilization of the polarized subchannels. It has been shown that the rate-1/2 length-128 PAC code with Fano decoding [24], [25] can approach the normal approximation (NA) bound. The U-UV coding is an alternative solution for incomplete polarization. Under the U-UV paradigm, each component code is transmitted through a subchannel. Their rates can

Manuscript received 3 May 2023; revised 26 September 2023, 22 December 2023, and 22 February 2024; accepted 22 February 2024. Date of publication 28 February 2024; date of current version 7 March 2024. This work was supported by the National Natural Science Foundation of China (NSFC) under Grant 62071498 and the Guangdong Provincial Natural Science Foundation. An earlier version of this paper was presented in part at the 2021 IEEE International Symposium on Information Theory (ISIT) [DOI: 10.1109/ISIT45174.2021.9517823]. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. An Liu. (*Corresponding author: Li Chen.*)

Wenhao Chen and Li Chen are with the School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou 510006, China (e-mail: chenwh85@mail2.sysu.edu.cn; chenli55@mail.sysu.edu.cn).

Jinjun Cheng and Changyu Wu are with the School of Electronics and Communication Engineering, Sun Yat-sen University, Guangzhou 510006, China (e-mail: chengjj5@mail2.sysu.edu.cn; wuchy28@mail2.sysu.edu.cn).

Huazi Zhang is with Hangzhou Research Center, Huawei Technologies Co., Ltd., Hangzhou 310052, China (e-mail: zhanghuazi@huawei.com).

Digital Object Identifier 10.1109/TSP.2024.3371153

be designed based on the subchannel capacities. Decoding of a U-UV code can be empowered by that of its component codes, making them another good performing short-to-medium length channel code. It has been shown that with the soft decoding of algebraic-geometric component codes, the U-UV codes can attain the capacity of the discrete symmetric channel [26]. This work shows that the U-UV codes can outperform polar codes, but at the cost of decoding complexity. However, the U-UV codes inherit a higher degree of SC/SCL decoding parallelism, yielding a much lower SC/SCL decoding latency.

By specifying the component codes, the U-UV codes have been known as the generalized concatenated codes (GCCs), in which the component codes and polar codes are the outer codes and inner codes, respectively [27], [28], [29], [30], [31], [32], [33]. These existing work have primarily considered BCH codes as the component codes [27], [28], [29], [32], while the extended BCH (eBCH) codes have been considered in [33]. Other component codes such as convolutional codes, Reed-Solomon codes and very small block codes have been considered in [28], [29], [30] and [31], respectively. The design of U-UV codes is realized by determining their component code rates. One straightforward approach arises from multilevel coding design [34], i.e. to estimate the subchannel capacities and design the component code rates accordingly. This design produces a code that aims to achieve a targeted transmission rate. But it cannot guarantee the code's performance. For this, a design algorithm has been proposed in [27] based on equaling the decoding error probabilities of the component codes. Meanwhile, the design of [31] and [33] aim to minimize the overall block error probability and maximize the minimum distance of the concatenated code, respectively. Bounds on the SC decoding error exponent and error rate have been analyzed in [32]. In the above mentioned work, the U-UV codes are mainly decoded by the SC algorithm, except in [33] the SCL algorithm was proposed based on hybrid list decoding of the eBCH codes. To the best of our knowledge, the first SCL decoding of U-UV codes was proposed in [35]. It has been shown that in the short-to-medium length regime, SCL decoding of the U-UV codes can outperform that of the polar codes<sup>1</sup>. Recently, the U-UV codes have also been integrated with bit-interleaved coded modulation (BICM) for realizing the spectrally efficient coded transmission [36], which is an extended research of this proposed work.

This paper presents a more complete research upon the authors' earlier work of [35]. Its major contributions include:

- 1) A combined code design approach is proposed for the U-UV codes, ensuring its decoding performance under a targeted transmission rate. Considering the BCH component codes are of limited length, finite length rates over the subchannels are first calculated. To ensure decoding performance of the U-UV code, component code rates are further adjusted through lining their theoretical decoding error probabilities, i.e., their tangential maximum likelihood (ML) decoding bounds.

<sup>1</sup>In this work, the SCL decoding of polar codes is assisted by a cyclic redundancy check (CRC) code. They are also called the CRC-polar codes.

- 2) SCL decoding of the proposed BCH based U-UV codes is introduced. The exponential list decoding complexity is rationalized by path pruning. To further facilitate the decoding, the reduced complexity SCL (RC-SCL) decoding is proposed. It can effectively eliminate the redundant decoding paths. Both the complexity and latency of the proposed SCL decoding are characterized, which in particular demonstrates the code's feature of yielding a low decoding latency.
- 3) Systematic U-UV codes are also introduced with the characterizations of their encoding and SCL decoding. The systematic U-UV codes have the decoding bit error rate (BER) performance advantage over their non-systematic counterparts.
- 4) Decoding performance of the U-UV codes is further analyzed and studied numerically. The SC decoding upper bound and approximated ML decoding lower bound are presented as our simulation benchmarks. Our simulation results show that with BCH component codes, SCL decoding of the U-UV codes can approach their approximated ML decoding lower bound. The U-UV codes are also compared with other competent short-to-medium length codes, including polar codes, BCH codes and TBC codes. Our simulation results show that the proposed U-UV codes can outperform the BCH codes and TBC codes with a lower decoding complexity as well as latency. In comparison with polar codes, the U-UV codes' performance advantage comes at the cost of decoding complexity. But they still exhibit a significant advantage on terms of decoding latency.

The rest of this paper is organized as follows. Section II introduces the U-UV code construction, including its systematic encoding. Section III proposes the combined U-UV code design. Section IV introduces the SCL decoding for the U-UV codes, and further proposes its complexity reduction variant. Section V analyzes the decoding complexity and latency. Section VI studies performance of the U-UV codes. Finally, Section VII concludes the paper.

## II. CODE CONSTRUCTION

The U-UV code construction is introduced in the context of BCH component codes. However, it should be noted that any linear block code can be utilized for the construction. The systematic U-UV encoding is also introduced.

### A. Code Construction

*Definition 1* [8]: Let the U code and the V code be two linear block codes of length  $n$ , which are parameterized by  $(n, k_U, d_U)$  and  $(n, k_V, d_V)$ , respectively, where  $k_U$  and  $k_V$  are their dimensions, and  $d_U$  and  $d_V$  are their minimum Hamming distances, respectively. The U-UV code is a  $(2n, k, d_{\min})$  linear block code that is constructed by

$$\{(\mathbf{u}|\mathbf{u} + \mathbf{v}); \mathbf{u} \in \mathbf{U} \text{ and } \mathbf{v} \in \mathbf{V}\}, \quad (1)$$

where  $k = k_U + k_V$  and  $d_{\min} = \min\{2d_U, d_V\}$  is the minimum Hamming distance of the U-UV code. Note that for

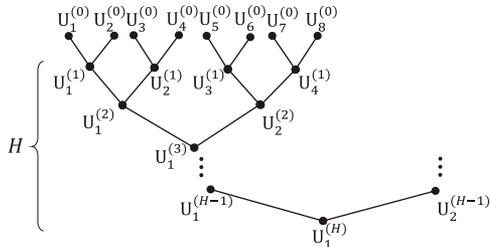


Fig. 1. Recursive construction of an  $H$ -level U-UV code.

convenience,  $U$  and  $V$  are also used to denote the respective codebooks.

In the proposed coding scheme, the above  $U$  code and  $V$  code are primitive BCH codes. This construction can be extended recursively by involving more BCH component codes, forming a larger U-UV code. This becomes a multi-level U-UV construction. Fig. 1 illustrates the recursive construction of an  $H$ -level U-UV code. Please note that in the rest of the paper, unless specifically mentioning, we use  $U_i^{(h)}$  (or  $u_i^{(h)}$ ) to denote the  $i$ th component code (or codeword) at level- $h$ , where  $h = 0, 1, \dots, H$  and  $i = 1, 2, \dots, 2^{H-h}$ . There are  $2^{H-h}$  component codes at level- $h$ . At each level, if  $i$  is odd,  $U_i^{(h)}$  is a  $U$  code. Otherwise, it is a  $V$  code. In particular, when  $h = 0$ , component code  $U_i^{(0)}$  is an  $(n, k_i)$  BCH code with rate  $R_i = k_i/n$ . In general, an  $H$ -level U-UV code consists of  $2^H$  BCH component codes with an overall length of  $2^H n$ . Its rate is  $(\sum_{i=1}^{2^H} R_i)/2^H$ . Based on Definition 1, the  $i$ th component code of level- $h$  ( $h \geq 1$ ) is constructed by two component codes of level- $(h-1)$  as

$$\left\{ \mathbf{u}_i^{(h)} = \left( \mathbf{u}_{2i-1}^{(h-1)} \parallel \mathbf{u}_{2i}^{(h-1)} \right) \right\}. \quad (2)$$

Let  $\mathbf{G}_{2i-1}^{(h-1)}$  and  $\mathbf{G}_{2i}^{(h-1)}$  denote the generator matrices of  $U_{2i-1}^{(h-1)}$  and  $U_{2i}^{(h-1)}$ , respectively. Based on the construction of (2), the generator matrix of  $U_i^{(h)}$  is

$$\mathbf{G}_i^{(h)} = \begin{bmatrix} \mathbf{G}_{2i-1}^{(h-1)} & \mathbf{G}_{2i}^{(h-1)} \\ \mathbf{0} & \mathbf{G}_{2i}^{(h-1)} \end{bmatrix}, \quad (3)$$

where  $\mathbf{0}$  is an all-zero matrix. With the  $2^H$  BCH component codes,  $\mathbf{G}_1^{(0)}, \mathbf{G}_2^{(0)}, \dots, \mathbf{G}_{2^H}^{(0)}$  can be defined. Generator matrix of the  $H$ -level U-UV code, denoted as  $\mathbf{G}_1^{(H)}$ , can be computed recursively as in (3). Let  $\mathbf{m}$  denote the message vector and the U-UV codeword  $\mathbf{u}_1^{(H)}$  is generated by

$$\mathbf{u}_1^{(H)} = \mathbf{m} \mathbf{G}_1^{(H)}. \quad (4)$$

Note that the U-UV codes can also be interpreted as the generalized polar concatenated codes [27], in which  $2^H$  outer BCH component codes are concatenated with  $n$  inner polar codes. The inner codes are of length  $2^H$ . This GCC interpretation is shown as in Fig. 2. Coded bits of an outer codeword are distributed to the same information position of the inner polar encoders. This implies that the U-UV construction

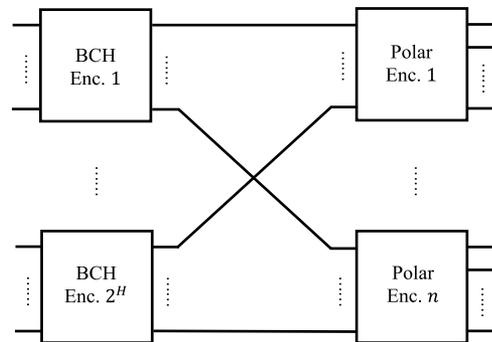


Fig. 2. GCC interpretation of a U-UV code with BCH component codes.

possesses an intrinsic channel polarization effect [3]. The BCH component codes are transmitted through the equivalent polarized subchannels.

### B. Systematic Encoding

The systematic generator matrix of a U-UV code can also be recursively constructed from the systematic generator matrices of its BCH component codes. Let  $\tilde{\mathbf{G}}_i^{(h)}$  denote the systematic generator matrix of the  $i$ th component code at level- $h$ , i.e.,  $U_i^{(h)}$ . In order to characterize the systematic generator matrix of a U-UV code, the following Lemma on the subchannel capacities is needed.

*Lemma 1:* At level- $(h-1)$ , capacity of the subchannel that transmits  $U_{2i-1}^{(h-1)}$  is greater than or equal to that of the subchannel that transmits  $U_{2i}^{(h-1)}$ .

*Proof:* This is a property of the channel polarization theorem [3]. Combining and splitting two independent channels, result in a good subchannel with a higher capacity and a bad subchannel with a lower capacity. Referring to Fig. 1,  $U_{2i-1}^{(h-1)}$  and  $U_{2i}^{(h-1)}$  correspond to the good subchannel and the bad subchannel at level- $(h-1)$ , respectively.  $\square$

Let  $k_{2i-1}^{(h-1)}$  and  $k_{2i}^{(h-1)}$  denote the dimension of component codes  $U_{2i-1}^{(h-1)}$  and  $U_{2i}^{(h-1)}$ , respectively. Lemma 1 implies that  $k_{2i-1}^{(h-1)} \geq k_{2i}^{(h-1)}$ . Armed with this, the following Theorem defines the systematic generator matrix of a U-UV code.

*Theorem 2:* Let  $\tilde{\mathbf{G}}_{2i-1}^{(h-1)}$  and  $\tilde{\mathbf{G}}_{2i}^{(h-1)}$  denote the systematic generator matrix of  $U_{2i-1}^{(h-1)}$  and  $U_{2i}^{(h-1)}$ , respectively. The systematic generator matrix of  $U_i^{(h)}$  can be determined by

$$\tilde{\mathbf{G}}_i^{(h)} = \begin{bmatrix} \tilde{\mathbf{G}}_{2i-1}^{(h-1)} & \tilde{\mathbf{G}}_{2i}^{(h-1)} \\ \mathbf{0} & \tilde{\mathbf{G}}_{2i}^{(h-1)} \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \tilde{\mathbf{G}}_{2i}^{(h-1)} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}. \quad (5)$$

*Proof:* Since  $\tilde{\mathbf{G}}_{2i-1}^{(h-1)} = \begin{bmatrix} \mathbf{I}_{2i-1}^{(h-1)} & \mathbf{P}_{2i-1}^{(h-1)} \end{bmatrix}$  and  $\tilde{\mathbf{G}}_{2i}^{(h-1)} = \begin{bmatrix} \mathbf{I}_{2i}^{(h-1)} & \mathbf{P}_{2i}^{(h-1)} \end{bmatrix}$ , where  $\mathbf{I}_{2i-1}^{(h-1)}$  and  $\mathbf{I}_{2i}^{(h-1)}$  are the identity submatrices of  $\tilde{\mathbf{G}}_{2i-1}^{(h-1)}$  and  $\tilde{\mathbf{G}}_{2i}^{(h-1)}$ , respectively, and  $\mathbf{P}_{2i-1}^{(h-1)}$  and  $\mathbf{P}_{2i}^{(h-1)}$  are their parity submatrices. For simplicity, sizes of the all-zero submatrices  $\mathbf{0}$  are not specified. Based on (3), with

the systematic generator matrix of  $U_{2i-1}^{(h-1)}$  and  $U_{2i}^{(h-1)}$ , the systematic generator matrix of  $U_i^{(h)}$  can be defined as

$$\begin{aligned}\tilde{\mathbf{G}}_i^{(h)} &= \begin{bmatrix} \tilde{\mathbf{G}}_{2i-1}^{(h-1)} & \tilde{\mathbf{G}}_{2i-1}^{(h-1)} \\ \mathbf{0} & \tilde{\mathbf{G}}_{2i}^{(h-1)} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{2i-1}^{(h-1)} \mathbf{P}_{2i-1}^{(h-1)} & \mathbf{I}_{2i-1}^{(h-1)} \mathbf{P}_{2i-1}^{(h-1)} \\ \mathbf{0} & \mathbf{I}_{2i}^{(h-1)} \mathbf{P}_{2i}^{(h-1)} \end{bmatrix}.\end{aligned}$$

Based on Lemma 1,  $k_{2i-1}^{(h-1)} \geq k_{2i}^{(h-1)}$ . Hence,

$$\begin{aligned}& \begin{bmatrix} \mathbf{I}_{2i-1}^{(h-1)} \mathbf{P}_{2i-1}^{(h-1)} & \mathbf{I}_{2i-1}^{(h-1)} \mathbf{P}_{2i-1}^{(h-1)} \\ \mathbf{0} & \mathbf{I}_{2i}^{(h-1)} \mathbf{P}_{2i}^{(h-1)} \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{I}_{2i}^{(h-1)} \mathbf{P}_{2i}^{(h-1)} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{2i-1}^{(h-1)} \mathbf{P}_{2i-1}^{(h-1)} & \mathbf{0} & \mathbf{\Omega} \\ \mathbf{0} & \mathbf{I}_{2i}^{(h-1)} \mathbf{P}_{2i}^{(h-1)} & \end{bmatrix},\end{aligned}\quad (6)$$

where the submatrix  $[\mathbf{0} \ \mathbf{\Omega}]$  is generated by

$$[\mathbf{0} \ \mathbf{\Omega}] = [\mathbf{I}_{2i-1}^{(h-1)} \mathbf{P}_{2i-1}^{(h-1)}] + [\mathbf{I}_{2i}^{(h-1)} \mathbf{P}_{2i}^{(h-1)}]. \quad (7)$$

It can be seen that (6) is in a systematic form.  $\square$

Theorem 2 shows that the systematic generator matrix  $\tilde{\mathbf{G}}_1^{(H)}$  of an  $H$ -level U-UV code can be recursively constructed by the systematic generator matrices of its BCH component codes. Given a message  $\mathbf{m} = (\mathbf{m}_1, \mathbf{m}_2)$ , where  $\mathbf{m}_1$  and  $\mathbf{m}_2$  are of length  $k_{2i-1}^{(h-1)}$  and  $k_{2i}^{(h-1)}$ , respectively. Based on (6), the systematic codeword  $\mathbf{u}_i^{(h)}$  can be generated by

$$\begin{aligned}\mathbf{u}_i^{(h)} &= \mathbf{m} \tilde{\mathbf{G}}_i^{(h)} \\ &= (\mathbf{m}_1, \mathbf{m}_2) \begin{bmatrix} \mathbf{I}_{2i-1}^{(h-1)} \mathbf{P}_{2i-1}^{(h-1)} & \mathbf{0} & \mathbf{\Omega} \\ \mathbf{0} & \mathbf{I}_{2i}^{(h-1)} \mathbf{P}_{2i}^{(h-1)} \end{bmatrix} \\ &= \left( \mathbf{m}_1, \mathbf{m}_1 \mathbf{P}_{2i-1}^{(h-1)}, \mathbf{m}_2, \mathbf{m}_1 \mathbf{\Omega} + \mathbf{m}_2 \mathbf{P}_{2i}^{(h-1)} \right),\end{aligned}\quad (8)$$

where the submatrix  $[\mathbf{0} \ \mathbf{\Omega}]$  is defined as in (7). Similarly, the systematic U-UV codeword  $\mathbf{u}_1^{(H)}$  can be generated by  $\tilde{\mathbf{G}}_1^{(H)}$ .

### III. CODE DESIGN

The U-UV code design is realized through determining its component code rates. This work considers both the finite length rate of each subchannel and the equal error probability between all component codes, aiming to construct a U-UV code that achieves a targeted transmission rate while maintaining a good decoding performance.

#### A. Finite Length Rate

With a targeted transmission rate, the BCH component code rates can be designed based on the finite length rates of the subchannels, which is determined through the normal approximation (NA) bound [25]. Note that BCH component codes are of short-to-medium length, they are not capacity approaching codes.

Let  $W_i^{(h)}$  denote an equivalent subchannel at level- $h$  that conveys component code  $U_i^{(h)}$ . At level-0, the BCH component codes are transmitted through subchannels  $W_1^{(0)}, W_2^{(0)}, \dots, W_{2^H}^{(0)}$ . Let  $\mathcal{R}_i$  denote the maximum transmission rate of subchannel  $W_i^{(0)}$ , such that if

$$R_i \leq \mathcal{R}_i, \quad (9)$$

error probability of  $U_i^{(0)}$  can be arbitrarily small. Given an  $H$ -level U-UV code, the average transmission rate of all subchannels is  $\mathcal{R} = (\sum_{i=1}^{2^H} \mathcal{R}_i) / 2^H$ . When codeword length  $n$  is sufficiently large, the maximum transmission rate of subchannel  $W_i^{(h)}$  can be characterized by its capacity  $I(W_i^{(h)})$ . But without this prerequisite on  $n$ , the maximum transmission rate would retreat from the capacity, due to channel dispersion, resulting in the a finite length transmission rate over the subchannel. Gaussian approximation (GA) [27] can be utilized to first estimate the equivalent noise variance. Subsequently, the subchannel capacities and their dispersions can be determined, leading to the characterization of the finite length rate for the subchannel.

Let  $u_{i,j}^{(h)}$  denote the  $j$ th bit of a component codeword  $\mathbf{u}_i^{(h)}$ , where  $j = 0, 1, \dots, 2^h n - 1$ . Let  $L_{i,j}^{(h)}$  further denote its decoding log-likelihood ratio (LLR), which will be defined in Section IV-A. They can be regarded as Gaussian random variables. Assume that the U-UV codeword  $\mathbf{u}_1^{(H)}$  is transmitted through the additive white Gaussian noise (AWGN) channel using binary modulation. Its noise variance is  $\sigma^2$ . The received LLRs exhibit  $\mathbb{E}[L_{1,j}^{(H)}] = 2/\sigma^2$  and  $\text{var}[L_{1,j}^{(H)}] = 4/\sigma^2$ , respectively, which are denoted as  $L_{1,j}^{(H)} \sim \mathcal{N}(2/\sigma^2, 4/\sigma^2)$ . In the U-UV paradigm, means of the decoding LLRs can be computed level-by-level as [27]

$$\mathbb{E}[L_{2i-1,j'}^{(h-1)}] = 2\mathbb{E}[L_{i,j}^{(h)}], \quad (10)$$

$$\mathbb{E}[L_{2i,j'}^{(h-1)}] = \varphi^{-1} \left( 1 - \left( 1 - \varphi \left( \mathbb{E}[L_{i,j}^{(h)}] \right) \right)^2 \right), \quad (11)$$

where  $j = 0, 1, \dots, 2^h n - 1$ ,  $j' = 0, 1, \dots, 2^{h-1} n - 1$  and

$$\varphi(z) = \begin{cases} 1 - \frac{1}{\sqrt{4\pi z}} \int_{-\infty}^{+\infty} \tanh\left(\frac{\nu}{2}\right) \\ \quad \times \exp\left(-\frac{(\nu-z)^2}{4z}\right) d\nu, & z > 0, \\ 1, & z = 0. \end{cases} \quad (12)$$

Finally,  $\mathbb{E}[L_{i,j}^{(0)}]$  can be determined. They are utilized to determine the subchannel capacities  $I(W_i^{(0)})$ . In particular, with  $\mathbb{E}[L_{i,j}^{(0)}]$ , the noise variance of subchannel  $W_i^{(0)}$  is  $\sigma_i^2 = 2/\mathbb{E}[L_{i,j}^{(0)}]$ . Its capacity can be determined by

$$I(W_i^{(0)}) = \frac{1}{2} \sum_{u_{i,j}^{(0)} \in \{0,1\}} \mathbb{E} \left[ \log_2 \frac{P(y_j | u_{i,j}^{(0)})}{\frac{1}{2} \sum_{u_{i,j}^{(0)} \in \{0,1\}} P(y_j | u_{i,j}^{(0)})} \right], \quad (13)$$

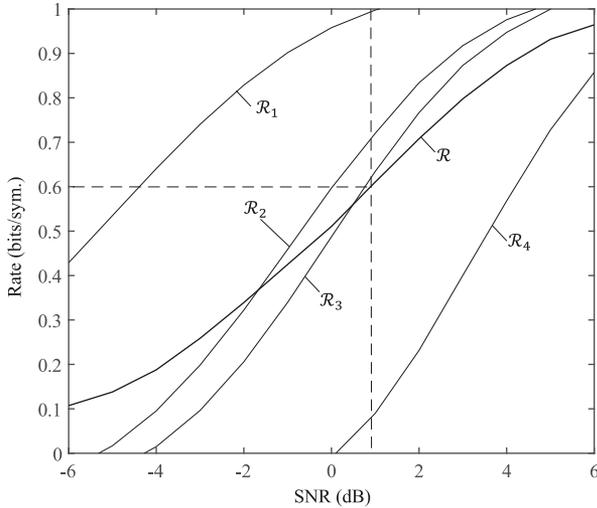


Fig. 3. Finite length rates  $\mathcal{R}_i$  of the 2-level U-UV construction.

where  $y_j$  is the received symbol that carries coded bit  $u_{i,j}^{(0)}$ , and  $P(y_j|u_{i,j}^{(0)})$  is the channel transition probability.

The NA bound [25] is further utilized to determine the finite length rates. For subchannel  $W_i^{(0)}$  with capacity  $I(W_i^{(0)})$  and a desired error probability  $P_e$ , its maximum transmission rate is [25]

$$\mathcal{R}_i = I(W_i^{(0)}) - \left( \sqrt{\frac{\mathcal{V}_i}{n}} Q^{-1}(P_e) - \frac{\log_2 n}{2n} \right), \quad (14)$$

where  $Q^{-1}(\cdot)$  is the inverse of  $Q$ -function, and  $\mathcal{V}_i$  is the dispersion of subchannel  $W_i^{(0)}$ . Subsequently, the BCH component code rates can be determined based on (9).

Considering a 2-level U-UV construction using length 63 BCH component codes and  $P_e = 10^{-4}$ , Fig. 3 shows the finite length rates of the 4 subchannels. If the coded system aims to achieve a targeted transmission rate of 0.6 bits/sym., Fig. 3 shows the signal-to-noise ratio (SNR) threshold is 1dB, at which the finite length rates of the subchannels are  $\mathcal{R}_1 = 1$ ,  $\mathcal{R}_2 = 0.72$ ,  $\mathcal{R}_3 = 0.61$  and  $\mathcal{R}_4 = 0.09$ , respectively. Therefore,  $U_1^{(0)}$ ,  $U_2^{(0)}$ ,  $U_3^{(0)}$  and  $U_4^{(0)}$  are chosen as the (63, 63), (63, 45), (63, 36) and the (63, 0) BCH codes, respectively. They result in a 2-level (252, 144) U-UV code. However, the above mentioned finite length rate design cannot ensure a good decoding performance for the U-UV code. The following equal error probability rule needs to be further applied to optimize the code's performance.

### B. Equal Error Probability

In order to ensure a good decoding performance for the U-UV code, the decoding performances of all component codes should be aligned. Intuitively, the worst performing component code becomes the decoding bottleneck of the overall U-UV code. The component codes will be decoded by the SCL algorithm which will be introduced in Section. IV. The decoding performance alignment is realized through further adjusting the component code rates. With the subchannel equivalent noise variance

and the weight spectrum of a BCH component code [37], we use the tangential bound [38] to characterize their decoding error probabilities.

Fig. 4(a) shows the tangential bounds of the 4 above mentioned BCH component codes, which are estimated over the underlying AWGN channel. Note that the (63, 0) BCH code does not carry information, resulting in its decoding error probability being zero. Decoding error probabilities of the 4 component codes do not align. Their rate adjustment is needed. For this, rate of the component codes that yield a worse decoding error probability should be reduced, and vice versa for the other component codes that yield a better decoding error probability. In this example, we can further reduce the rates of  $U_1^{(0)}$  and  $U_2^{(0)}$ , and increase the rate of  $U_4^{(0)}$ . This leads to  $U_1^{(0)}$ ,  $U_2^{(0)}$ ,  $U_3^{(0)}$  and  $U_4^{(0)}$  being the (63, 57), (63, 39), (63, 36) and (63, 7) BCH codes, respectively, resulting in the (252, 139) U-UV code. Fig. 4(b) shows the decoding error probabilities of the 4 component codes are better aligned.

Please note that a component code rate design algorithm was also proposed in [27] under the paradigm of the generalized polar concatenated codes. However, this design is formulated at the cost of overlooking the component codes' performance over a wider SNR regime. We will compare these two design approaches in Section VI-B.

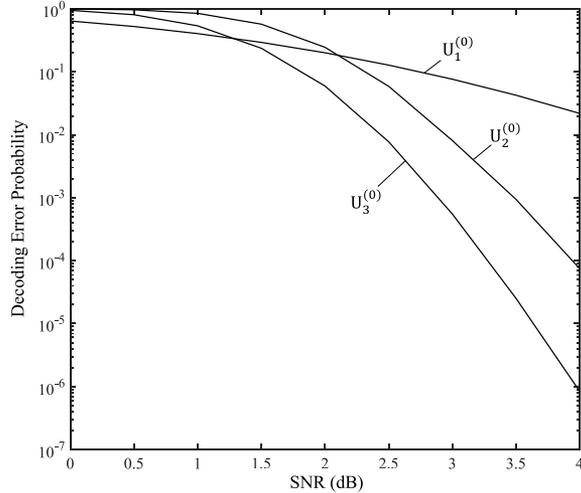
## IV. THE SCL DECODING

The SCL decoding of a U-UV code is substantiated by the list decoding of its component codes. In this work, the OSD [39] is employed to decode the BCH component codes. It not only yields a near ML decoding performance for the component codes, but also provides a list of estimated codewords. The SCL decoding complexity is rationalized through list pruning and its complexity can be further reduced by eliminating the redundant decoding paths. Finally, SCL decoding of systematic U-UV codes is introduced.

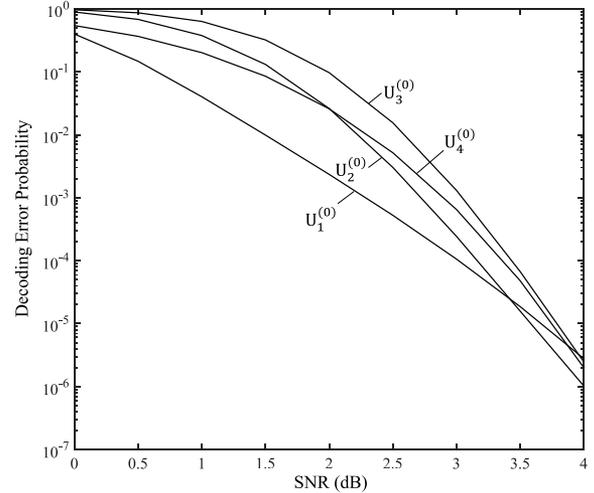
### A. The Algorithm

In the SCL decoding mechanism, the component codes are decoded successively. Decoding of a component code is based on the estimations of the earlier decoded ones. This leads to the number of component codeword estimations growing exponentially, resulting in a prohibitive decoding complexity. In order to rationalize the decoding complexity, it is assumed that  $l$  decoding estimations are kept for each component code, which parameterizes the SCL decoding. Note that the OSD can yield a decoding output list size greater than  $l$ .

Assume that codeword  $\mathbf{u}_1^{(H)}$  is transmitted through a memoryless channel using the binary phase shift keying (BPSK) modulation. The received symbol vector is  $\mathbf{y} = (y_0, y_1, \dots, y_{2^H n-1}) \in \mathbb{R}^{2^H n}$ . Let us define  $\Omega(h) = \{1, 2, \dots, 2^{H-h}\}$  as the set of the indices of level- $h$  component codes, and  $\Omega_i(h) = \{i+1, i+2, \dots, 2^{H-h}\}$  as a subset of  $\Omega(h)$ . At level- $h$ , the component codes are decoded in the order of  $U_{2^{H-h}}^{(h)} \rightarrow U_{2^{H-h-1}}^{(h)} \rightarrow \dots \rightarrow U_1^{(h)}$ . Hence,  $\Omega_i(h)$  indicates the set of level- $h$  component codes which are decoded prior to  $U_i^{(h)}$ . Note



(a) The (252, 144) U-UV code



(b) The (252, 139) U-UV code

Fig. 4. Equivalent error probabilities of U-UV component codes.

that at level- $H$ , the U-UV code is formed and  $\Omega_i(H) = \emptyset$ . For  $\mathbf{u}_i^{(h)} = (u_{i,0}^{(h)}, u_{i,1}^{(h)}, \dots, u_{i,2^h n-1}^{(h)})$ , LLR of  $u_{i,j}^{(h)}$  is defined as [40]

$$L_{i,j}^{(h)} = \ln \frac{P\left(\{y_{j+(\eta-1)2^h n}\}_{\eta \in \Omega_i(h)}, \{\hat{u}_{\rho,j}^{(h)}\}_{\rho \in \Omega_i(h)} \mid u_{i,j}^{(h)} = 0\right)}{P\left(\{y_{j+(\eta-1)2^h n}\}_{\eta \in \Omega_i(h)}, \{\hat{u}_{\rho,j}^{(h)}\}_{\rho \in \Omega_i(h)} \mid u_{i,j}^{(h)} = 1\right)}, \quad (15)$$

where  $P(\{y_{j+(\eta-1)2^h n}\}_{\eta \in \Omega_i(h)}, \{\hat{u}_{\rho,j}^{(h)}\}_{\rho \in \Omega_i(h)} \mid u_{i,j}^{(h)})$  are the transition probabilities of subchannel  $W_i^{(h)}$ ,  $\hat{u}_{\rho,j}^{(h)}$  is an estimation of  $u_{\rho,j}^{(h)}$  and  $j = 0, 1, \dots, 2^h n - 1$ . Since  $\Omega_i(H) = \emptyset$ ,

$$L_{1,j}^{(H)} = \ln \frac{P(y_j \mid u_{1,j}^{(H)} = 0)}{P(y_j \mid u_{1,j}^{(H)} = 1)}. \quad (16)$$

The LLRs will be updated level-by-level until  $L_{i,j}^{(0)}$  are produced. The level- $(h-1)$  LLRs can be determined by [40]

$$L_{2i,j'}^{(h-1)} = f\left(L_{i,j'}^{(h)}, L_{i,j'+2^{h-1}n}^{(h)}\right), \quad (17)$$

$$L_{2i-1,j'}^{(h-1)} = L_{i,j'}^{(h)} + (-1)^{\hat{u}_{2i,j'}^{(h-1)}} L_{i,j'+2^{h-1}n}^{(h)}, \quad (18)$$

where  $f(\mathcal{X}, \mathcal{Y}) \triangleq \ln \frac{e^{\mathcal{X}} e^{\mathcal{Y}} + 1}{e^{\mathcal{X}} + e^{\mathcal{Y}}}$ ,  $\mathcal{X}, \mathcal{Y} \in \mathbb{R}$  and  $j' = 0, 1, \dots, 2^{h-1}n - 1$ . Eq. (18) shows that the LLRs update of component code  $U_{2i}^{(h-1)}$  requires the estimations of  $U_{2i}^{(h-1)}$ .

Fig. 5 illustrates the SCL decoding of a 2-level U-UV code. Without loss of generality, the SCL decoding of a component code  $U_i^{(h)}$  can be performed by the following key operations.

**Code Decomposition:** If  $h > 0$ ,  $U_i^{(h)}$  is still a U-UV structural code. It should be decomposed into  $U_{2i}^{(h-1)}$  and  $U_{2i-1}^{(h-1)}$ , with their LLRs updated by (17) and (18), respectively.

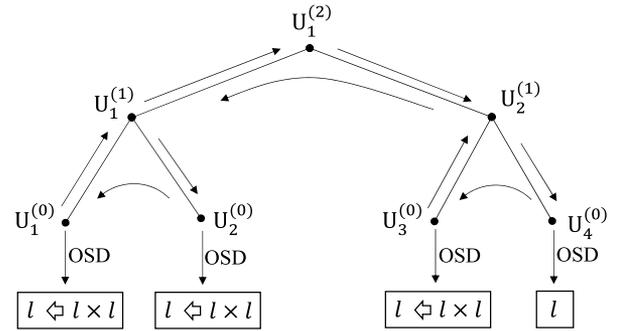


Fig. 5. The SCL decoding process of a 2-level U-UV code.

**Component Code Decoding:** The above decomposition continues until level-0, where the BCH component codes  $U_4^{(0)}, U_3^{(0)}, U_2^{(0)}, U_1^{(0)}$  are decoded by the OSD.

**Code Reconstruction:** Once the estimations of  $\hat{\mathbf{u}}_{2i-1}^{(h-1)}$  and  $\hat{\mathbf{u}}_{2i}^{(h-1)}$  have been obtained,  $\hat{\mathbf{u}}_i^{(h)}$  can be obtained as in (2).

The above SCL decoding of the U-UV code can be illustrated by a decoding tree that is shown in Fig. 6(a). In the tree, each layer corresponds to a BCH component code and nodes of the layer represent its estimations. With  $l$  estimations for each component BCH codeword, the number of decoding paths will grow exponentially, leading to a prohibitive decoding complexity. Therefore, decoding path pruning is needed. For this, the following proposition needs to be introduced.

**Proposition 3:** In the SCL decoding of U-UV codes, the  $l$  most likely estimations of the  $i$ th BCH component code  $U_i^{(0)}$  are kept, where  $i = 1, 2, \dots, 2^H$ . They are further denoted as  $\hat{\mathbf{u}}_i^{(0)}(1), \hat{\mathbf{u}}_i^{(0)}(2), \dots, \hat{\mathbf{u}}_i^{(0)}(l)$ , respectively.

Let  $\hat{\mathbf{u}}_i^{(0)}(s') = (\hat{u}_{i,0}^{(0)}(s'), \hat{u}_{i,1}^{(0)}(s'), \dots, \hat{u}_{i,n-1}^{(0)}(s'))$  and  $\hat{\mathbf{u}}_{i+1}^{(0)}(s) = (\hat{u}_{i+1,0}^{(0)}(s), \hat{u}_{i+1,1}^{(0)}(s), \dots, \hat{u}_{i+1,n-1}^{(0)}(s))$  denote an

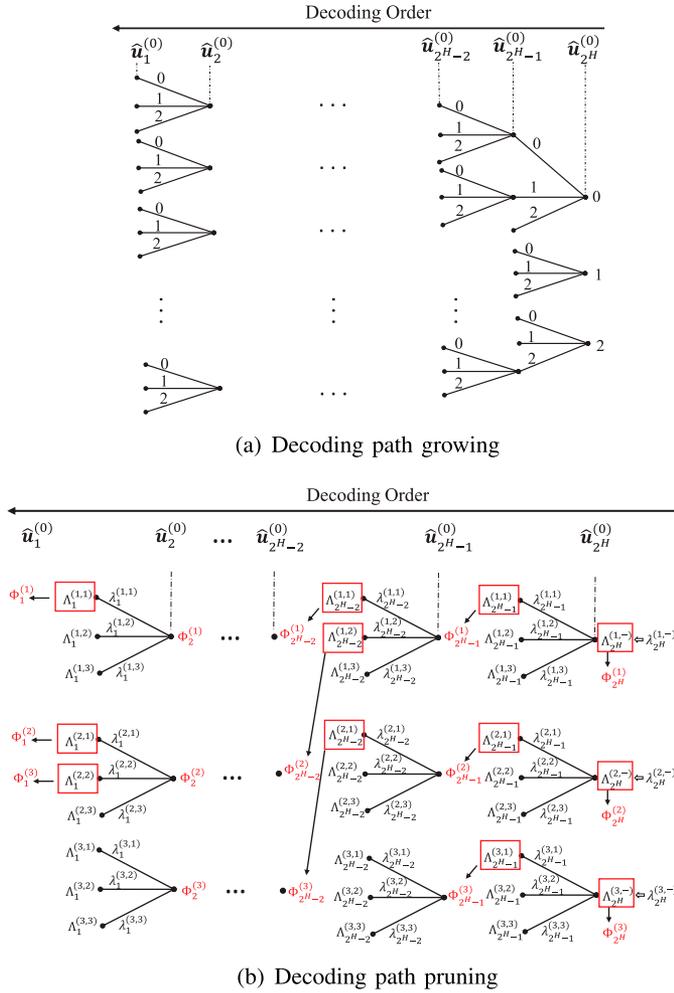


Fig. 6. The SCL decoding tree,  $l = 3$ .

estimation of  $U_i^{(0)}$  and  $U_{i+1}^{(0)}$ , respectively, where  $s, s' = 1, 2, \dots, l$ . Further let

$$\mathbf{L}_i^{(0)}(s') = \left( L_{i,0}^{(0)}(s'), L_{i,1}^{(0)}(s'), \dots, L_{i,n-1}^{(0)}(s') \right) \quad (19)$$

denote the corresponding LLR vector that leads to the estimation  $\hat{\mathbf{u}}_i^{(0)}(s')$ . The correlation distance between  $\mathbf{L}_i^{(0)}(s')$  and  $\hat{\mathbf{u}}_i^{(0)}(s')$  is defined as

$$\begin{aligned} \lambda_i^{(s,s')} &= \lambda \left( \mathbf{L}_i^{(0)}(s'), \hat{\mathbf{u}}_i^{(0)}(s') \right) \\ &= \sum_{j \in \Psi_i^{(s,s')}} \left| L_{i,j}^{(0)}(s') \right|, \end{aligned} \quad (20)$$

where  $\Psi_i^{(s,s')} = \{j \mid L_{i,j}^{(0)}(s') \cdot (1 - 2\hat{u}_{i,j}^{(0)}(s')) < 0\}$ . Please note that in (20),  $\mathbf{L}_i^{(0)}(s')$  is obtained based on the estimation  $\hat{\mathbf{u}}_i^{(0)}(s)$  as in (17) and (18). This correlation distance is a metric for assessing the likelihood of the decoding estimations. A smaller correlation distance indicates the estimation is more likely to be the transmitted codeword. Over the SCL decoding tree in Fig. 6(a),  $\lambda_i^{(s,s')}$  is the metric of the branch that emancipates from node  $s$  of layer  $i+1$  and ends at node

### Algorithm 1 SCL Decoding of $U_i^{(h)}$ , $\text{SCLD}(U_i^{(h)})$

**Input:**  $\{L_{i,j}^{(h)} \mid j = 0, 1, \dots, 2^h n - 1\}$ ,  $l$ ;

**Output:**  $\{\hat{\mathbf{u}}_i^{(h)}(s)\}$ ;

- 1: **If**  $h > 0$
- 2: Determine the LLRs of  $U_{2^i}^{(h-1)}$  as in (17);
- 3: Perform SCLD( $U_{2^i}^{(h-1)}$ ) and produce  $\{\hat{\mathbf{u}}_{2^i}^{(h-1)}(s)\}$ ;
- 4: **For** each estimation  $\hat{\mathbf{u}}_{2^i}^{(h-1)}$  **do**
- 5: Determine the LLRs of  $U_{2^{i-1}}^{(h-1)}$  as in (18);
- 6: Perform SCLD( $U_{2^{i-1}}^{(h-1)}$ ) and produce  $\{\hat{\mathbf{u}}_{2^{i-1}}^{(h-1)}(s)\}$ ;
- 7: **For** each pair of  $\hat{\mathbf{u}}_{2^{i-1}}^{(h-1)}$  and  $\hat{\mathbf{u}}_{2^i}^{(h-1)}$  **do**
- 8: Reconstruct  $\hat{\mathbf{u}}_i^{(h)}$  as in (2);
- 9: Form the estimation list  $\{\hat{\mathbf{u}}_i^{(h)}(s)\}$ ;
- 10: **Else**
- 11: Decode  $U_i^{(0)}$ , producing  $\{\hat{\mathbf{u}}_i^{(0)}(s)\}$ ;
- 12: Determine  $\Lambda_i^{(s,s')}$  as in (20) and (21);
- 13: Keep the  $l$  most likely estimations of  $U_i^{(0)}$ ;

$s'$  of layer  $i$ . Metrics of the nodes are accumulated along the SCL decoding path, resulting in the accumulated correlation distances (ACDs) that are stored at the paths. In particular, the ACD at node  $s'$  of layer  $i$  is defined as

$$\Lambda_i^{(s,s')} = \Phi_{i+1}^{(s)} + \lambda_i^{(s,s')}, \quad (21)$$

where  $\Phi_{i+1}^{(s)}$  denote the ACDs that are kept at layer  $i+1$ . Based on Proposition 3,  $\Phi_{i+1}^{(1)}, \Phi_{i+1}^{(2)}, \dots, \Phi_{i+1}^{(l)}$  are the  $l$  smallest ACDs at layer  $i+1$ . Similarly, an SCL decoding path with a smaller ACD indicates it is more likely to be the correct path. Since  $U_{2^H}^{(0)}$  is the first component code to be decoded, its estimations are the roots of the SCL decoding tree. Therefore, their branch metrics and node metrics are denoted as  $\lambda_{2^H}^{(s,-)}$  and  $\Lambda_{2^H}^{(s,-)}$ , respectively.

At the beginning,  $\Phi_{2^H}^{(s)}$  are initialized as

$$\Phi_{2^H}^{(s)} = \Lambda_{2^H}^{(s,-)} = \lambda_{2^H}^{(s,-)}. \quad (22)$$

By processing from layer  $i+1$  to layer  $i$ ,  $l^2$  ACDs are determined by (20) and (21). Based on Proposition 3, these  $l^2$  ACDs will be sorted, and only the  $l$  smallest ones will be kept and denoted as  $\Phi_i^{(s')}$ . This path pruning is shown over the SCL decoding tree as in Fig. 6(b) where  $l = 3$ .

The above SCL decoding is summarized as in **Algorithm 1**. After the last BCH component code  $U_1^{(0)}$  is decoded, the  $l$  smallest ACDs  $\Phi_1^{(1)}, \Phi_1^{(2)}, \dots, \Phi_1^{(l)}$  will be kept. Let

$$s^* = \arg \min \left\{ \Phi_1^{(s)}, \forall s \right\}. \quad (23)$$

The SCL decoding path that corresponds to  $\Phi_1^{(s^*)}$  will be identified. The estimated U-UV codeword and its message can then be retrieved from the path.

### B. Complexity Reduction

The above SCL decoding complexity can be further reduced by eliminating the unpromising decoding paths. This is realized

by estimating the branch metrics  $\lambda_i^{(s,s')}$ , such that the decoding paths that are unlikely to yield the  $l$  smallest ACDs can be eliminated. Let us first introduce the the following proposition.

*Proposition 4:* The  $l$  smallest ACDs kept at layer  $i + 1$  are reordered as

$$\Phi_{i+1}^{(1)} \leq \Phi_{i+1}^{(2)} \leq \dots \leq \Phi_{i+1}^{(l)}. \quad (24)$$

Subsequently, in decoding component code  $U_i^{(0)}$ , we prioritize the elaboration of the node with a smaller ACD. That says  $U_i^{(0)}$  will first be decoded based on the estimation of  $U_{i+1}^{(0)}$  that corresponds to  $\Phi_{i+1}^{(1)}$ , yielding the ACDs  $\Lambda_i^{(1,1)}$ ,  $\Lambda_i^{(1,2)}$ ,  $\dots$ ,  $\Lambda_i^{(1,l)}$  at layer  $i$ . They are sorted as in (24) and denoted as  $\Phi_i^{(1)}$ ,  $\Phi_i^{(2)}$ ,  $\dots$ ,  $\Phi_i^{(l)}$ , respectively. The decoder then continues to decode  $U_i^{(0)}$  based on the node with  $\Phi_{i+1}^{(2)}$ . However, if the path starting from the node is unlikely to yield an ACD that is smaller than  $\Phi_i^{(l)}$ , it will be discarded. Its inherited component code decoding can be skipped. This results in a complexity reducing variant of the above mentioned SCL decoding.

Assuming the branch metric  $\lambda_i^{(s,s')}$  are Gaussian distributed with  $\lambda_i^{(s,s')} \sim \mathcal{N}(\mu_{\lambda_i}, \sigma_{\lambda_i}^2)$  [41], [42], its lower bound  $\Upsilon_i$  can be estimated through

$$\frac{1}{\sqrt{2\pi}\sigma_{\lambda_i}} \int_{\Upsilon_i}^{\infty} e^{-\frac{(\lambda_i - \mu_{\lambda_i})^2}{2\sigma_{\lambda_i}^2}} d\lambda_i = 1 - \theta, \quad (25)$$

where  $\theta \in (0, 1)$ . Note that the mean  $\mu_{\lambda_i}$  and variance  $\sigma_{\lambda_i}^2$  can be computed based on Theorem 4 of [42]. Therefore, in decoding component code  $U_i^{(0)}$ , its correlation distance lower bound  $\Upsilon_i$  can be obtained beforehand. Once the  $l$  smallest ACDs  $\Phi_i^{(1)}$ ,  $\Phi_i^{(2)}$ ,  $\dots$ ,  $\Phi_i^{(l)}$  have been obtained from the decoding that starts from node  $\Phi_{i+1}^{(1)}$ , the following decoding of  $U_i^{(0)}$  will be assessed before their execution. For  $s > 1$  and if

$$\Phi_{i+1}^{(s)} + \Upsilon_i > \Phi_i^{(l)}, \quad (26)$$

it indicates nodes of  $\Phi_{i+1}^{(s)}$ ,  $\Phi_{i+1}^{(s+1)}$ ,  $\dots$ ,  $\Phi_{i+1}^{(l)}$  cannot lead to a more likely decoding path than existing ones. Their decoding path elaboration can be skipped. Otherwise, the decoder continues to decode  $U_i^{(0)}$  based on the node with  $\Phi_{i+1}^{(s)}$ , yielding the ACDs  $\Lambda_i^{(s,1)}$ ,  $\Lambda_i^{(s,2)}$ ,  $\dots$ ,  $\Lambda_i^{(s,l)}$  at layer  $i$ . Afterward, the  $l$  smallest ACDs that are kept at layer  $i$  are updated and sorted again as in (24). This process continues until the condition of (26) occurs or all the decoding path elaborations from layer  $i + 1$  have been completed. The decoder then moves onto decode  $U_{i-1}^{(0)}$ . Note that in practice, the above  $\theta$  of (25) needs to be optimized empirically.

This RC-SCL decoding also results in a smaller ACD sorting complexity. In the proposed SCL, an  $l^2$ -to- $l$  path sorting is required at each layer. In this RC-SCL, the  $l^2$ -to- $l$  path sorting is replaced by several sequential  $2l$ -to- $l$  path sortings. When the received information is reliable, this  $2l$ -to- $l$  path sorting rarely occurs. The simulation results in Section VI-B will demonstrate this property and the complexity reduction effect of the RC-SCL decoding.

### C. Decoding of Systematic U-UV Codes

SCL decoding of systematic U-UV codes can be processed similarly as above. However, after the component code decoding, the U-UV codeword needs to be reconstructed in order to retrieve the intended message. The following Lemma shows the relationship between different portions of the systematic codeword of (8).

*Lemma 5:* Given a systematic codeword  $(\mathbf{m}_1, \mathbf{m}_1 \mathbf{P}_{2i-1}^{(h-1)}, \mathbf{m}_2, \mathbf{m}_1 \mathbf{\Omega} + \mathbf{m}_2 \mathbf{P}_{2i}^{(h-1)})$  of  $U_i^{(h)}$  as in (8),  $(\mathbf{m}_2, \mathbf{m}_1 \mathbf{\Omega} + \mathbf{m}_2 \mathbf{P}_{2i}^{(h-1)}) - (\mathbf{m}_1, \mathbf{m}_1 \mathbf{P}_{2i-1}^{(h-1)})$  is a codeword of  $U_{2i}^{(h-1)}$ .

*Proof:* Recall that for  $\mathbf{m} = (\mathbf{m}_1, \mathbf{m}_2)$ ,  $\mathbf{m}_1$  and  $\mathbf{m}_2$  are of lengths  $k_{2i-1}^{(h-1)}$  and  $k_{2i}^{(h-1)}$ , respectively. Since

$$(\mathbf{m}_2, \mathbf{m}_1 \mathbf{\Omega} + \mathbf{m}_2 \mathbf{P}_{2i}^{(h-1)}) = (\mathbf{m}_1, \mathbf{m}_2) \begin{bmatrix} \mathbf{0} & \mathbf{\Omega} \\ \mathbf{I}_{2i}^{(h-1)} & \mathbf{P}_{2i}^{(h-1)} \end{bmatrix}$$

and

$$\begin{bmatrix} \mathbf{0} & \mathbf{\Omega} \\ \mathbf{I}_{2i}^{(h-1)} & \mathbf{P}_{2i}^{(h-1)} \end{bmatrix} = \begin{bmatrix} \mathbf{I}_{2i-1}^{(h-1)} & \mathbf{P}_{2i-1}^{(h-1)} \\ \mathbf{I}_{2i}^{(h-1)} & \mathbf{P}_{2i}^{(h-1)} \end{bmatrix} + \begin{bmatrix} \mathbf{I}_{2i}^{(h-1)} & \mathbf{P}_{2i}^{(h-1)} \\ \mathbf{0} & \end{bmatrix},$$

we have

$$\begin{aligned} & (\mathbf{m}_2, \mathbf{m}_1 \mathbf{\Omega} + \mathbf{m}_2 \mathbf{P}_{2i}^{(h-1)}) \\ &= (\mathbf{m}_1, \mathbf{m}_2) \left( \begin{bmatrix} \mathbf{I}_{2i-1}^{(h-1)} & \mathbf{P}_{2i-1}^{(h-1)} \\ \mathbf{I}_{2i}^{(h-1)} & \mathbf{P}_{2i}^{(h-1)} \end{bmatrix} + \begin{bmatrix} \mathbf{I}_{2i}^{(h-1)} & \mathbf{P}_{2i}^{(h-1)} \\ \mathbf{0} & \end{bmatrix} \right). \end{aligned}$$

Based on Lemma 1, we know  $k_{2i-1}^{(h-1)} \geq k_{2i}^{(h-1)}$ , and

$$\begin{aligned} & (\mathbf{m}_2, \mathbf{m}_1 \mathbf{\Omega} + \mathbf{m}_2 \mathbf{P}_{2i}^{(h-1)}) \\ &= (\mathbf{m}_1, \mathbf{m}_1 \mathbf{P}_{2i-1}^{(h-1)}) + (\mathbf{m}_2, \mathbf{m}_2 \mathbf{P}_{2i}^{(h-1)}) \\ & \quad + \mathbf{m}_1|_{1 \sim k_{2i}^{(h-1)}} \left[ \mathbf{I}_{2i}^{(h-1)} \mathbf{P}_{2i}^{(h-1)} \right], \end{aligned}$$

where  $\mathbf{m}_1|_{1 \sim k_{2i}^{(h-1)}}$  denotes a subvector of  $\mathbf{m}_1$  that contains its first  $k_{2i}^{(h-1)}$  entries. Therefore,

$$\begin{aligned} & (\mathbf{m}_2, \mathbf{m}_1 \mathbf{\Omega} + \mathbf{m}_2 \mathbf{P}_{2i}^{(h-1)}) - (\mathbf{m}_1, \mathbf{m}_1 \mathbf{P}_{2i-1}^{(h-1)}) \\ &= (\mathbf{m}_2, \mathbf{m}_2 \mathbf{P}_{2i}^{(h-1)}) + \mathbf{m}_1|_{1 \sim k_{2i}^{(h-1)}} \left[ \mathbf{I}_{2i}^{(h-1)} \mathbf{P}_{2i}^{(h-1)} \right] \\ &= (\mathbf{m}_2 + \mathbf{m}_1|_{1 \sim k_{2i}^{(h-1)}}) \left[ \mathbf{I}_{2i}^{(h-1)} \mathbf{P}_{2i}^{(h-1)} \right] \\ &= (\mathbf{m}_2 + \mathbf{m}_1|_{1 \sim k_{2i}^{(h-1)}}) \mathbf{G}_{2i}^{(h-1)}. \end{aligned}$$

It can be seen that  $(\mathbf{m}_2, \mathbf{m}_1 \mathbf{\Omega} + \mathbf{m}_2 \mathbf{P}_{2i}^{(h-1)}) - (\mathbf{m}_1, \mathbf{m}_1 \mathbf{P}_{2i-1}^{(h-1)})$  is a codeword of  $U_{2i}^{(h-1)}$ .  $\square$

The above Lemma shows that a systematic U-UV codeword still maintains the recursive U-UV structure. At level- $h$ , both message  $\mathbf{m}_1$  and  $\mathbf{m}_2$  can be retrieved from the systematic codeword  $\mathbf{u}_i^{(h)}$ . Hence, systematic U-UV codes can be decoded in the same manner as their non-systematic counterparts. But the systematic U-UV codeword needs to be reconstructed after all BCH component codes have been decoded. The intended message can be further retrieved from therein.

TABLE I  
DECODING COMPLEXITY OF DIFFERENT CODING SCHEMES

	U-UV, SCL	U-UV, RC-SCL	Polar, SCL
LLR Update	$O(\ln H 2^H)$	$O(\bar{l} \ln H 2^H)$	$O(\ln 2^H \log_2(n 2^H))$
Component Code Decoding	$O(l 2^H k_i^{\tau_{i^*}})$	$O(\bar{l} 2^H k_i^{\tau_{i^*}})$	–
Path Sorting	$O(l 2^H (\Gamma^* + l) \log_2 l)$	$O(\bar{l} 2^H (\Gamma^* + l) \log_2 l)$	$O(2kl \log_2 l)$

## V. COMPLEXITY AND LATENCY ANALYSES

The SCL decoding complexity and latency for the U-UV codes are characterized.

### A. Decoding Complexity

In SCL decoding of U-UV codes, the complexity attributes to the LLR updates between levels, the decoding of component codes and the decoding path sorting.

With a decoding output list size of  $l$ , the complexity of LLR updates of an  $H$ -level U-UV code can be characterized as that of SCL decoding  $n$  polar codes of length  $2^H$ . Hence, it is  $O(\ln 2^H \log_2 2^H) = O(\ln H 2^H)$ .

In this work, the BCH component codes are decoded by the OSD. With an  $(n, k)$  BCH code, the complexity of an order- $\tau$  OSD is  $O(k^\tau)$  [39]. Let  $\tau_i$  denote the OSD order for component code  $U_i^{(0)}$ , and

$$i^* = \arg \max\{k_i^{\tau_i} \mid i = 1, 2, \dots, 2^H\}. \quad (27)$$

The OSD complexity in decoding all  $2^H$  BCH component codes can be approximated as  $O(l 2^H k_i^{\tau_{i^*}})$ .

The SCL decoding path sorting exists in identifying the  $l$  most likely estimations from the OSD output and the subsequent  $l^2$ -to- $l$  decoding path sorting. Let  $\Gamma_i$  denote the total number of estimations that are generated by the OSD of  $U_i^{(0)}$ , and

$$\Gamma_i = \sum_{j=0}^{\tau_i} \binom{k_i}{j}. \quad (28)$$

In order to identify the  $l$  most likely estimations from the OSD of  $U_i^{(0)}$ , the sorting complexity is  $O(\Gamma_i \log_2 l)$ . Further let

$$\Gamma^* = \max\{\Gamma_i \mid i = 1, 2, \dots, 2^H\}, \quad (29)$$

the sorting complexity for identifying the  $l$  most likely estimations from the OSD of all component codes can be approximated as  $O(l 2^H \Gamma^* \log_2 l)$ . Similarly, the complexity of the  $l^2$ -to- $l$  decoding path sorting is  $O(2^H l^2 \log_2 l)$ .

Complexity of the RC-SCL decoding can be further analyzed as follows. In the RC-SCL decoding, some of the  $l$  stored nodes of the layers may not be explored. Let  $\bar{l}$  denote the average number of explored nodes at each layer, and  $\bar{l} \leq l$ . Similar to the above analysis, complexity of the LLR updates is  $O(\bar{l} \ln H 2^H)$ , and the OSD complexity can be approximated as  $O(\bar{l} 2^H k_i^{\tau_{i^*}})$ . The sorting complexity for identifying the  $l$  most likely estimations from the OSD would be  $O(\bar{l} 2^H \Gamma^* \log_2 l)$ . Furthermore, the  $2l$ -to- $l$  decoding path sorting complexity is  $O(\bar{l} 2^H l \log_2 l)$ .

The above analysis is summarized in Table I, which also shows the SCL decoding complexity of a polar code with length

of  $n 2^H$  and dimension of  $k$ . It shows that the RC-SCL decoding reduces the SCL decoding complexity by a factor of  $1 - \bar{l}/l$ . Table I also shows that the SCL decoding of a polar code yields an extra LLR update complexity of  $O(\ln 2^H \log_2 n)$  over that of a U-UV code. However, the latter requires extra complexity in decoding the component codes and sorting. Note that the component code decoding complexity will become dominant if the OSD order is large. But there also exist several methods to reduce the OSD complexity, including the box-and-match algorithm [43]. Moreover, list decoding of the component codes can also be realized through other more algebraic approaches [44], [45], [46], which can be considered if other component codes are employed.

### B. Decoding Latency

In this work, the decoding latency is characterized in terms of the number of clock cycles (CCs) required to decode a U-UV codeword. It is a metric that has been widely adopted for analyzing the decoding latency [21]. Similarly, the SCL decoding latency attributes to the LLR updates between levels, the component code decoding and the subsequent path sorting.

Let us assume that the maximum degrees of parallelism in the LLR updates and the re-encoding of OSD are  $P_1$  and  $P_2$ , respectively, where  $P_1$  is a multiple of  $n$ . At level- $h$ , the LLR updates are performed  $2^{H-h}$  times, each of which processes an LLR vector of length  $n 2^h$ . If  $n 2^h \leq P_1$ , only one CC is required for an update. Otherwise,  $\lceil n 2^h / P_1 \rceil$  CCs are needed. Hence, the LLR update requires [47]

$$\begin{aligned} T_1 &= \sum_{h=0}^{\lfloor \log_2(P_1/n) \rfloor} 2^{H-h} + \sum_{h=\lfloor \log_2(P_1/n) \rfloor + 1}^{H-1} 2^{H-h} \cdot \lceil n 2^h / P_1 \rceil \\ &\approx \frac{n 2^H}{P_1} \left( H - 2 - \log_2 \frac{P_1}{n} \right) + 2^{H+1} \end{aligned} \quad (30)$$

CCs. In decoding the  $i$ th BCH component code, OSD is performed for  $l$  decoding paths in parallel. Sorting of the input LLRs requires one CC, and Gaussian elimination (GE) requires  $k'_i$  CCs, where  $k'_i = \min\{k_i, n - k_i\}$  [48]. The re-encoding requires  $\lceil l \Gamma_i / P_2 \rceil$  CCs. Based on a pipeline structure, calculating the correlation distances, identifying the  $l$  most likely estimations and sorting the  $l^2$  decoding paths would require one CC each. Hence, decoding a component code requires  $k'_i + \lceil l \Gamma_i / P_2 \rceil + 4$  CCs. Further let  $K = \sum_{i=1}^{2^H} k'_i$ , decoding all component codes yields a latency of

$$T_2 = K + 4 \cdot 2^H + \sum_{i=1}^{2^H} \lceil l \Gamma_i / P_2 \rceil. \quad (31)$$

TABLE II  
DECODING LATENCY OF DIFFERENT CODING SCHEMES

	Latency (CCs)
U-UV, SCL	$K + 6 \cdot 2^H + Q_1 + Q_2$
U-UV, RC-SCL (best)	$K + 6 \cdot 2^H + Q_1 + Q'_2$
U-UV, RC-SCL (worst)	Eq. (34)
Polar, SCL	$k + n2^{H+1} + Q_1$

Therefore, decoding a U-UV code requires

$$\begin{aligned} T_{\text{SCL}} &= T_1 + T_2 \\ &\approx K + 6 \cdot 2^H + Q_1 + Q_2 \end{aligned} \quad (32)$$

CCs, where  $Q_1 = \frac{n2^H}{P_1}(H - 2 - \log_2 \frac{P_1}{n})$  and  $Q_2 = \sum_{i=1}^{2^H} \lceil l\Gamma_i/P_2 \rceil$ . In particular, if  $P_1 \geq n2^{H-1}$  and  $P_2 \geq l\Gamma^*$ , the LLR updates and the decoding of each component code can be fully parallelized, resulting in a decoding latency of  $T_{\text{SCL}} = K + 7 \cdot 2^H - 2$ . Therefore, OSD dominates the SCL decoding latency.

The RC-SCL decoding would however incur latency. In the best case, there is only one path elaboration at decoding each component code. It requires  $\lceil \Gamma_i/P_2 \rceil$  CCs to generate  $\Gamma_i$  estimations. Hence, based on (32), the decoding latency is

$$T_{\text{RC-SCL}}^{(\text{best})} = K + 6 \cdot 2^H + Q_1 + Q'_2, \quad (33)$$

where  $Q'_2 = \sum_{i=1}^{2^H} \lceil \Gamma_i/P_2 \rceil$ . In the worst case, another  $l-1$  path elaborations are needed for decoding each component code except  $U_{2^H}^{(0)}$ . This leads to an increased decoding latency of

$$\begin{aligned} T_{\text{RC-SCL}}^{(\text{worst})} &= T_{\text{RC-SCL}}^{(\text{best})} + (l-1) \cdot \\ &\left( K - k'_{2^H} + 4(2^H - 1) + \sum_{i=1}^{2^H-1} \lceil \Gamma_i/P_2 \rceil \right). \end{aligned} \quad (34)$$

Again, if  $P_1 \geq n2^{H-1}$  and  $P_2 \geq \Gamma^*$ ,  $T_{\text{RC-SCL}}^{(\text{best})}$  and  $T_{\text{RC-SCL}}^{(\text{worst})}$  converge to  $T_{\text{RC-SCL}}^{(\text{best})} = K + 7 \cdot 2^H - 2$  and  $T_{\text{RC-SCL}}^{(\text{worst})} = T_{\text{RC-SCL}}^{(\text{best})} + (l-1)(K - k'_{2^H} + 5(2^H - 1))$ , respectively. It should be pointed out that, the RC-SCL decoding consumes less hardware resources, since the  $l^2$ -to- $l$  path sorting is replaced by several sequential  $2l$ -to- $l$  sortings.

Table II summarizes the above decoding latency analysis. They are compared with the SCL decoding latency for the polar codes of length  $n2^H$  and dimension  $k$  [40]. Note that for most practical codes,  $K \leq k$  and  $6 \cdot 2^H < n2^{H+1}$ . If the OSD can be realized with a large degree of parallelism, the SCL decoding latency of U-UV codes would be much lower than that of polar codes. More numerical results will be provided in Section VI-B.

## VI. PERFORMANCE ANALYSIS

This section first studies the theoretical performance benchmarks for the U-UV codes, which will be utilized to evaluate our simulation results. Our simulations show that the U-UV codes can outperform several key good performing short-to-medium length codes.

### A. Theoretical Benchmarks

The SC decoding upper bound and the approximated ML decoding lower bound of the U-UV codes are characterized, which are utilized as the performance benchmarks for our simulations.

1) *SC Decoding Upper Bound*: Decoding error probability of a U-UV code can be determined by the equivalent error probabilities of its BCH component codes. For an  $H$ -level U-UV code, there are  $2^H$  component decoding events. The erroneous decoding (including the decoding errors and failures) of any component code will lead to the erroneous decoding of the U-UV code. Let  $T_i$  denote the event that erroneous decoding first occurs in decoding  $U_i^{(0)}$ . The decoding error probability of an  $H$ -level U-UV code is determined by

$$P_e = \sum_{i=1}^{2^H} P(T_i). \quad (35)$$

For BCH component codes,  $P(T_i)$  can be characterized theoretically. In this work, we utilize the tangential bound [38] which requires the code's weight spectrum. Let  $P_{\text{tan},i}$  denote the tangential bound of component code  $U_i^{(0)}$  that is transmitted through the equivalent polarized subchannel. Hence,  $P(T_i) \leq P_{\text{tan},i}$ . The decoding error probability of an  $H$ -level U-UV code is upper bounded by

$$P_e \leq \sum_{i=1}^{2^H} P_{\text{tan},i}. \quad (36)$$

2) *Approximated ML Decoding Lower Bound*: The SCL decoding performance can be lower bounded by the code's ML decoding performance. Let  $E_b$  and  $N_0 = 2\sigma^2$  denote the energy per message bit and the noise power spectral density, respectively. Note that a code's ML decoding error probability can be characterized by the union bound. The truncated union bound that is determined by the code's minimum weight can be seen as an approximation of the ML decoding lower bound at the high SNR regime [49]. Therefore, the ML decoding error probability lower bound of a U-UV code can be approximated as

$$P_e \geq \frac{1}{2} A_{d_{\min}} \text{erfc} \left( \sqrt{\frac{d_{\min} R E_b}{N_0}} \right), \quad (37)$$

where  $\text{erfc}(\cdot)$  is the complementary error function and  $A_{d_{\min}}$  is the number of U-UV codewords with the minimum weight  $d_{\min}$ . Based on Definition 1,  $d_{\min}$  can be determined by knowing the minimum Hamming distances of the BCH component codes. The following Lemma is further introduced to determine  $A_{d_{\min}}$ . For simplicity, the component codes are resumed the name of U code and V code in this Lemma and its proof.

*Lemma 6*: Given a U-UV code, let  $A_U$  and  $A_V$  denote the number of the minimum weight codewords of the U code and the V code, respectively. The number of the minimum weight codewords of the U-UV code is

$$A_{d_{\min}} = \begin{cases} A_U, & \text{if } 2d_U < d_V; \\ A_U + A_V + A', & \text{if } 2d_U = d_V; \\ A_V + A'', & \text{if } 2d_U > d_V, \end{cases} \quad (38)$$

where  $A'$  and  $A''$  are nonnegative integers.

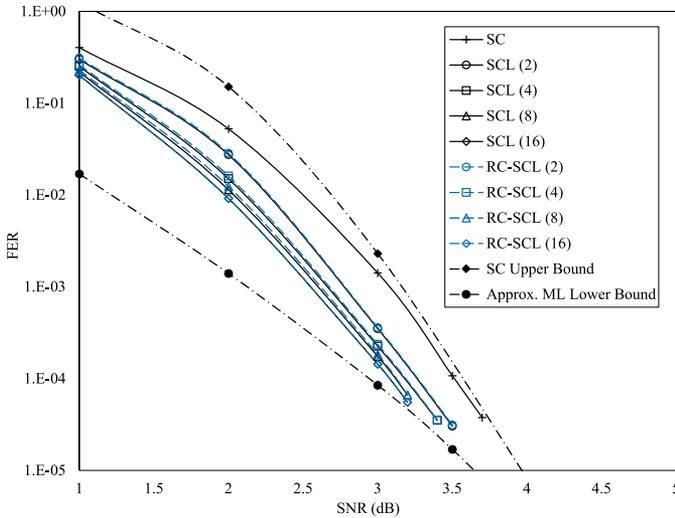


Fig. 7. Performance of the 2-level (252, 139) U-UV code.

*Proof:* The proof is provided in Appendix A.  $\square$

The above Lemma shows that  $A_{d_{\min}}$  of a U-UV code can be partially characterized by the number of the minimum weight codewords of its component codes, i.e.,  $A_U$  and  $A_V$ . But determining  $A'$  and  $A''$  remains challenging. However, if  $2d_U < d_V$ ,  $A_{d_{\min}} = A_U$ . In this case, the ML decoding performance lower bound of the U-UV code can be characterized by only knowing the distance property and weight distribution of the U code. For the 2-level (255, 139) U-UV code,  $2d_U < d_V$  holds for its BCH component codes. The U-UV code has  $d_{\min} = 12$  and  $A_{d_{\min}} = 651$ .

### B. Simulation Results

The SCL decoding performance of the BCH based U-UV codes are shown together with some numerical results on decoding complexity and latency. With an SCL decoding output list size of  $l$ , the OSD orders are chosen such that a near ML decoding performance of the BCH component codes can be achieved. In particular, the (63, 57), (63, 51), (63, 45), (63, 39), (63, 36), (63, 24), (63, 18), (63, 10) and (63, 7) BCH codes are utilized as the component codes for constructing the presented U-UV codes. Their decoding orders are 1, 1, 2, 2, 2, 3, 3, 3 and 3, respectively. The BCH component codes are designed by the combined approach introduced in Section III. The simulation results are obtained over the AWGN channel using BPSK. The U-UV codes are also compared with other key short-to-medium length channel codes, including polar codes, BCH codes and TBC codes.

1) *Performance of U-UV Codes:* Fig. 7 shows the SCL decoding performance of the 2-level (252, 139) U-UV code. The U-UV code is constructed by the (63, 57), (63, 39), (63, 36) and (63, 7) BCH codes. It can be seen that the SCL decoding outperforms the SC decoding. A greater coding gain can be achieved by increasing the SCL decoding output list size  $l$ . The SC decoding upper bound and approximated ML decoding lower bound envelop the SC and SCL decoding performances.

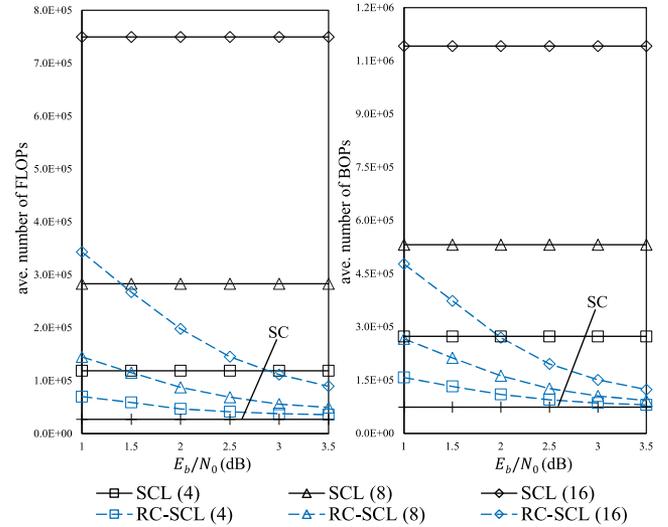


Fig. 8. Decoding complexity of the 2-level (252, 139) U-UV code.

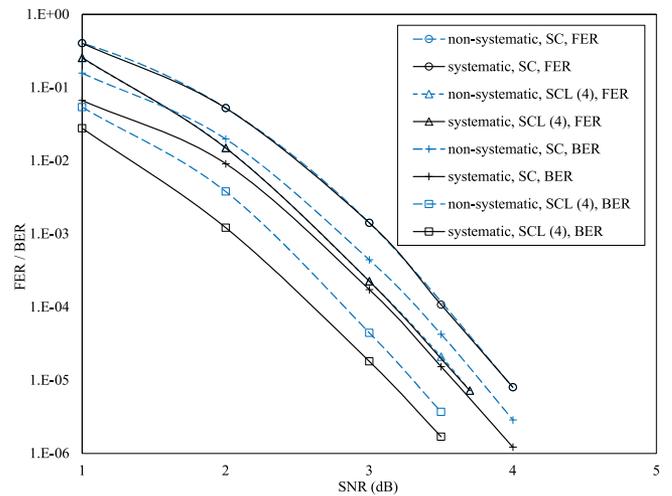


Fig. 9. Comparison between the systematic and non-systematic U-UV codes.

For this U-UV code, its SCL decoding with  $l = 16$  can approach the approximated ML lower bound at the FER of  $10^{-4}$ . Fig. 7 also shows that the RC-SCL decoding maintains a similar performance as its SCL prototype. Note that for the RC-SCL,  $\theta = 10^{-4}$ . Fig. 8 further shows the complexity reduction brought by the RC-SCL decoding. The average numbers of decoding floating point operations (FLOPs) and binary operations (BOPs) are shown. It can be seen that the RC-SCL decoding can reduce the SCL decoding complexity significantly. Especially as the SNR increases, it converges to the SC decoding complexity. This implies that  $\bar{l}$  converges to 1 as the channel condition improves.

Fig. 9 further compares the (252, 139) U-UV code with its systematic counterpart. Both the decoding BER and FER are shown. It can be seen that the systematic U-UV code yields the same FER performance as the non-systematic one, but provides a better BER performance. This BER performance disparity

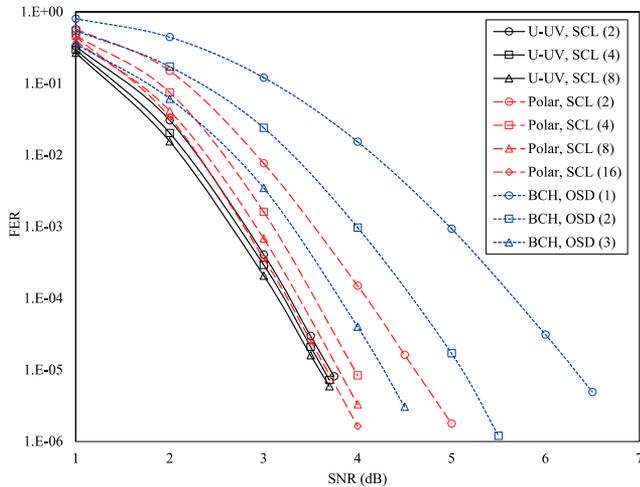


Fig. 10. Comparison between U-UV, BCH and polar codes.

comes from the U-UV codeword reconstruction that is required by the systematic code. This reconstruction will reinterpret some wrongly estimated coded bits, resulting in an enhanced BER performance. A similar decoding phenomenon has also been observed for systematic polar codes [50].

2) *Comparison With Other Coding Schemes:* The BCH based U-UV codes closely relate to both BCH codes and polar codes. In particular, when there is only one component code, the U-UV structure dissolves, resulting in a single BCH code. To the opposite, when the component codes are of length-1, the U-UV code evolves into a polar code. Hence, the proposed U-UV codes are compared with both BCH codes and polar codes. They are also compared with the other good performing short-to-medium length codes, the TBC codes.

Fig. 10 compares the 2-level (252, 139) U-UV code with the (255, 139) BCH code and the (256, 140) polar code. The BCH code is decoded by the order- $\tau$  OSD, denoted as OSD ( $\tau$ ). The polar code is designed by the 5th generation new radio (5G NR) standard. Its SCL decoding is assisted by a length-8 CRC code. Fig. 10 shows that the U-UV code can substantially outperform the BCH code. Meanwhile, with the same SCL decoding output list size, the U-UV code can also achieve a remarkable coding gain over the polar code. However, it is noticed that unlike the polar code, limited coding gain can be achieved by increasing the SCL decoding output list size of the U-UV code. This suggests performance of a U-UV code heavily depends on the decoding of its component codes, and requires a rather small SCL decoding output list size. Moreover, it shall be aware that unlike polar codes, the high SCL decoding performance of a U-UV code does not need to be ensured by the use of CRC. This is because the OSD can provide a near ML decoding performance for its component codes. Consequently, the ML U-UV codeword is more likely to be included in the SCL decoding output list. Table III further compares the decoding complexity and latency of the three coding schemes. Note that the latency is measured in terms of the number of required CCs for decoding a codeword. We consider both full parallelism and partial parallelism. It shows that the U-UV code can outperform

TABLE III  
DECODING COMPLEXITY AND LATENCY OF  
DIFFERENT CODING SCHEMES

Scheme	FLOPs	Latency (CCs)
U-UV, SCL (2)	$5.5 \times 10^4$	90
U-UV, SCL (4)	$1.2 \times 10^5$	93 <sup>†</sup>
U-UV, SCL (8)	$2.8 \times 10^5$	103 <sup>‡</sup>
BCH, OSD (1)	$1.1 \times 10^4$	143
BCH, OSD (2)	$5.6 \times 10^5$	145 <sup>†</sup>
BCH, OSD (3)	$2.6 \times 10^7$	149 <sup>‡</sup>
Polar, SCL (2)	$5.1 \times 10^3$	650
Polar, SCL (4)	$9.8 \times 10^3$	652 <sup>†</sup>
Polar, SCL (8)	$1.9 \times 10^4$	660 <sup>‡</sup>
Polar, SCL (16)	$4.1 \times 10^4$	660 <sup>‡</sup>
Polar, MSCL (2)* [17]	$1.5 \times 10^5$	126
Polar, MSCL (4)* [17]	$3.2 \times 10^5$	140 <sup>†</sup>
Polar, MSCL (8)* [17]	$6.7 \times 10^5$	156 <sup>‡</sup>
Polar, FSSCL (2) [18]	$3.1 \times 10^3$	128
Polar, FSSCL (4) [18]	$1.4 \times 10^4$	158
Polar, FSSCL (8) [18]	$5.4 \times 10^4$	192

<sup>†</sup> A half of the required maximum degree of parallelism.

<sup>‡</sup> A quarter of the required maximum degree of parallelism.

\* Size of the multi-bit decision is  $M = 8$ .

a single BCH code with a lower complexity and smaller latency. This implies that it is beneficial to replace a large BCH code by several smaller ones that are coupled through the U-UV structure. As in comparison with the polar code, SCL decoding of the U-UV code is more complex in general, even when the two codes yield a similar decoding performance. E.g., decoding the U-UV code with  $l = 2$  performs similarly as that of the polar code with  $l = 16$  under a similar complexity. Decoding the U-UV code is still slightly more complex. However, as Table III shows, decoding the U-UV code exhibits a great advantage in latency. This primarily thanks to the U-UV code's block wise SC (or SCL) decoding manner, where the number of the component codes (blocks) is very limited. Moreover, Table III also shows the decoding complexity and latency of the facilitated SCL decoding, including the multi-bit SCL (MSCL) decoding [17] and the fast simplified SCL (FSSCL) decoding [18] for the polar code. They can yield the same decoding performance as their prototype. It shows that latency of decoding the U-UV code remains advantage when comparing with the MSCL and the FSSCL decoding. Note that both the MSCL and the FSSCL decoding trade the decoding complexity for latency gains. Latency merit of the U-UV code can also be understood in this aspect. Table III shows SCL decoding the U-UV code yields a better tradeoff than the MSCL decoding and the FSSCL decoding of the polar code. E.g., Fig. 10 shows the U-UV code with  $l = 2$  performs similarly as the polar code with  $l = 8$ . Table III shows SCL decoding the U-UV code exhibits a smaller latency than both the MSCL decoding and the FSSCL decoding the polar code. It is also less complex than the MSCL decoding. This is due to the block wise (component code) decoding in a U-UV code is more efficient than that in the MSCL decoding of a polar code.

Table IV further shows RC-SCL decoding latency for the (252, 139) U-UV code, in which both the worst case and the average latency are measured. Note that the best case latency

TABLE IV  
RC-SCL DECODING LATENCY OF THE (252,139) U-UV CODE

SNR (dB)	$l = 2$		$l = 4$		$l = 8$	
	ave.	worst	ave.	worst	ave.	worst
1.0	122	162 <sup>‡</sup>	193	306 <sup>‡</sup>	330	594 <sup>‡</sup>
2.0	102	162 <sup>‡</sup>	144	306 <sup>‡</sup>	229	562
3.0	95	162 <sup>‡</sup>	113	306 <sup>‡</sup>	160	466
4.0	91	133	99	201	123	370

<sup>‡</sup> The result is consistent with eq. (34).

TABLE V  
DECODING COMPLEXITY AND LATENCY COMPARISON

Scheme	FLOPs	Latency (CCs)
U-UV, SCL (4)	$5.1 \times 10^5$	155
U-UV, SCL (8)	$1.2 \times 10^6$	155
Polar, SCL (8)	$4.2 \times 10^4$	1276
Polar, MSCL (8)* [17]	$1.2 \times 10^6$	254
Polar, FSSCL (8) [18]	$1.0 \times 10^5$	308
TBC (8), WAVA	$3.8 \times 10^5$	655
TBC (11), WAVA	$3.1 \times 10^6$	670
TBC (14), WAVA	$2.5 \times 10^7$	698

\* The size of the multibit decision is  $M = 8$ .

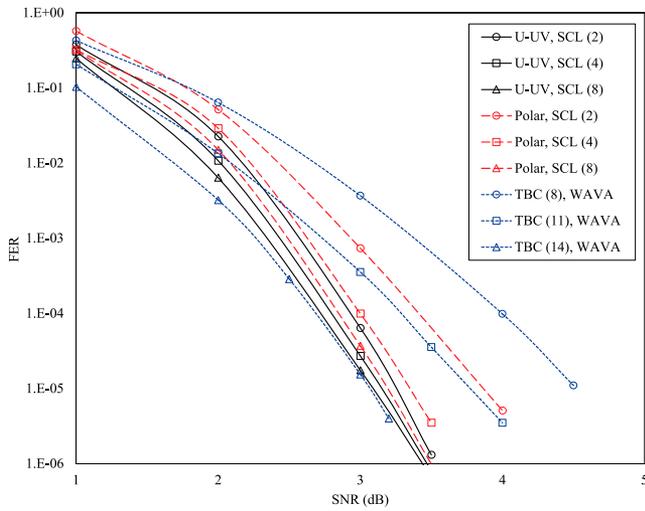


Fig. 11. Comparison between U-UV, polar and TBC codes.

is the same as its prototype, i.e., 90 CCs for the code. It can be seen that the average latency converges to the SCL decoding latency as the SNR increases. Both Tables III and IV verify the SCL decoding latency advantage of the U-UV codes. They also validates our latency analysis of Section V-B.

Furthermore, Fig. 11 compares the 3-level (504, 250) U-UV code with the (512, 254) polar code that is assisted by a length-8 CRC code and the (512, 256) TBC codes with memory sizes of  $m = 8, 11$  and  $14$  [7]. The TBC codes are decoded by the wrap-around Viterbi algorithm (WAVA) [51]. The (504, 250) U-UV code is constructed by the (63, 57), (63, 51), (63, 45), (63, 24), (63, 45), (63, 18), (63, 10) and (63, 0) BCH codes. It shows that with the same SCL decoding output list size, the U-UV code can also outperform the polar code. It also shows that this U-UV code can significantly outperform the TBC codes with  $m = 8$  and  $11$ . Table V further compares their decoding complexity and latency. Fig. 11 shows that SCL decoding of the U-UV code with  $l = 8$  performs similarly as the WAVA of the TBC code with  $m = 14$ . Table V shows that decoding the U-UV code requires smaller numbers of FLOPs and CCs than decoding the TBC code. It also shows that the SCL decoding of the U-UV code with  $l = 4$  can already realize a better complexity-latency tradeoff than the three SCL decoding of the polar code with  $l = 8$ .

3) *Comparison With the Existing Design:* Finally, Fig. 12 compares the FER performance of the U-UV code designed by

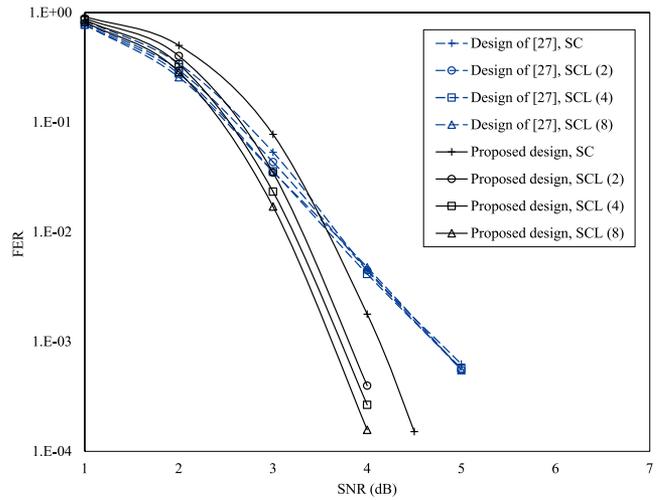


Fig. 12. Comparison of different U-UV code designs.

our combined approach and the other designed by the algorithmic approach of [27]. They both result in the 2 levels (252, 183) U-UV code. Our design yields the (63, 57), (63, 51), (63, 51) and (63, 24) BCH component codes. The design of [27] yields the (63, 63), (63, 51), (63, 51) and (63, 18) BCH component codes. It can be seen that our proposed design results in a better performing U-UV code. Performance of the U-UV code designed by [27] has been limited by its worst performing component code. As pointed earlier in Section III-B, the design of [27] can only provide the component code rates for a specific SNR, limiting its performance over a wider SNR region.

## VII. CONCLUSION

This paper has proposed the U-UV structural codes and its SCL decoding, as another competent short-to-medium length channel code. Systematic encoding of a U-UV code has also been introduced. The recursive U-UV coding structure leads to capacities of the subchannels being polarized. Each of the subchannels conveys a BCH component codeword with a rate capped by the subchannel transmission limit. The U-UV code design jointly considers the finite length rate and the equal error probability rule. It yields a U-UV code that cannot only achieve a targeted transmission rate but also competent decoding performance. The SCL decoding of U-UV codes is also

introduced based on list decoding the BCH component codes. The RC-SCL decoding has been further proposed to facilitate the decoding. Complexity and latency of the SCL decoding have been analyzed. The theoretical decoding performance benchmarks for U-UV codes have been characterized. Our simulation results have shown that with SCL decoding, U-UV codes can outperform BCH codes and TBC codes with a lower decoding complexity as well as latency. In comparison with polar codes, the U-UV codes' performance advantage comes at the cost of decoding complexity. But they still exhibit a significant advantage on terms of decoding latency.

## APPENDIX

## PROOF OF LEMMA 5

*Proof:* Let  $\mathbf{u}_0$  and  $\mathbf{v}_0$  denote the minimum weight codeword of the U code and the V code, respectively. It can be seen that the minimum weight codeword of the U-UV code will be in one of the following forms,  $(\mathbf{u}_0|\mathbf{u}_0)$ ,  $(\mathbf{0}|\mathbf{v}_0)$  and  $(\mathbf{u}_0|\mathbf{u}_0 + \mathbf{v}_0)$ , where  $\mathbf{0}$  is an all zero vector. Let  $w(\mathbf{c})$  denote the Hamming weight of codeword  $\mathbf{c}$ . We know  $w(\mathbf{u}_0|\mathbf{u}_0) = 2d_U$  and  $w(\mathbf{0}|\mathbf{v}_0) = d_V$ . Subsequently,  $w(\mathbf{u}_0|\mathbf{u}_0 + \mathbf{v}_0)$  can be characterized in the following three cases.

Case I: If  $2d_U < d_V$ ,  $w(\mathbf{u}_0|\mathbf{u}_0) = 2d_U$ ,  $w(\mathbf{0}|\mathbf{v}_0) > 2d_U$ , and

$$\begin{aligned} w(\mathbf{u}_0|\mathbf{u}_0 + \mathbf{v}_0) &= w(\mathbf{u}_0) + w(\mathbf{u}_0 + \mathbf{v}_0) \\ &\geq w(\mathbf{u}_0) + (w(\mathbf{v}_0) - w(\mathbf{u}_0)) \\ &= w(\mathbf{v}_0) \\ &> 2d_U. \end{aligned}$$

Therefore, the minimum weight codeword of the U-UV code will be in the form of  $(\mathbf{u}_0|\mathbf{u}_0)$ . That says  $A_{d_{\min}} = A_U$ .

Case II: If  $2d_U = d_V$ ,  $w(\mathbf{u}_0|\mathbf{u}_0) = w(\mathbf{0}|\mathbf{v}_0)$ , and

$$\begin{aligned} w(\mathbf{u}_0|\mathbf{u}_0 + \mathbf{v}_0) &\geq w(\mathbf{u}_0) + (w(\mathbf{v}_0) - w(\mathbf{u}_0)) \\ &= w(\mathbf{v}_0) \\ &= d_V. \end{aligned} \quad (39)$$

Note that if  $w(\mathbf{u}_0 + \mathbf{v}_0) = w(\mathbf{v}_0) - w(\mathbf{u}_0)$ ,  $w(\mathbf{u}_0|\mathbf{u}_0 + \mathbf{v}_0) = d_V$ . This requires the support of "1" in  $\mathbf{u}_0$  is a subset of the support of "1" in  $\mathbf{v}_0$ . E.g.,  $\mathbf{u}_0 = (10010)$  and  $\mathbf{v}_0 = (11010)$ . Let  $A'$  denote the number of U-UV codewords that satisfy  $w(\mathbf{u}_0|\mathbf{u}_0 + \mathbf{v}_0) = d_V$ . Therefore, the number of minimum weight codewords of a U-UV code is  $A_U + A_V + A'$ .

Case III: If  $2d_U > d_V$ ,  $w(\mathbf{u}_0|\mathbf{u}_0) > d_V$ ,  $w(\mathbf{0}|\mathbf{v}_0) = d_V$ , and  $w(\mathbf{u}_0|\mathbf{u}_0 + \mathbf{v}_0) \geq d_V$  as in (39). Similarly, codeword  $(\mathbf{u}_0|\mathbf{u}_0 + \mathbf{v}_0)$  may also have the minimum weight if  $w(\mathbf{u}_0 + \mathbf{v}_0) = w(\mathbf{v}_0) - w(\mathbf{u}_0)$ . Let  $A''$  denote the number of such minimum weight codewords. The number of minimum weight U-UV codewords is  $A_V + A''$ .  $\square$

## REFERENCES

- [1] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. IEEE Int. Conf. Commun.*, Geneva, Switzerland, May 1993, pp. 1064–1070.
- [2] R. Gallager, "Low-density parity-check codes," *IRE Trans. Inf. Theory*, vol. 8, no. 1, pp. 21–28, Jan. 1962.
- [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] A. Hocquenghem, "Codes correcteurs d'Erreurs," *Chiffres (Paris)*, vol. 2, pp. 147–156, Sep. 1959.
- [5] R. Bose and D. Ray-Chaudhuri, "On a class of error correcting binary group codes," *Inf. Control*, vol. 3, no. 1, pp. 68–79, 1960.
- [6] H. Ma and J. Wolf, "On tail biting convolutional codes," *IEEE Trans. Commun.*, vol. COM-34, no. 2, pp. 104–111, Feb. 1986.
- [7] M. Coskun et al., "Efficient error-correcting codes in the short block-length regime," *Phys. Commun.*, vol. 34, pp. 66–79, Mar. 2019.
- [8] M. Plotkin, "Binary codes with specified minimum distance," *IRE Trans. Inf. Theory*, vol. 6, no. 4, pp. 445–450, Sep. 1960.
- [9] I. Reed, "A class of multiple-error-correcting codes and the decoding scheme," *IRE Trans. Inf. Theory*, vol. 4, no. 4, pp. 38–49, Sep. 1954.
- [10] D. Muller, "Application of Boolean algebra to switching circuit design and to error detection," *IRE Trans. Electron. Comput.*, vol. EC-3, no. 3, pp. 6–12, Sep. 1954.
- [11] S. Kudekar et al., "Reed–Muller codes achieve capacity on erasure channels," *IEEE Trans. Inf. Theory*, vol. 63, no. 7, pp. 4298–4316, Jul. 2017.
- [12] K. Niu and K. Chen, "CRC-aided decoding of polar codes," *IEEE Commun. Lett.*, vol. 16, no. 10, pp. 1668–1671, Oct. 2012.
- [13] I. Tal and A. Vardy, "List decoding of polar codes," *IEEE Trans. Inf. Theory*, vol. 61, no. 5, pp. 2213–2226, May 2015.
- [14] A. Alamdar-Yazdi and F. Kschischang, "A simplified successive-cancellation decoder for polar codes," *IEEE Commun. Lett.*, vol. 15, no. 12, pp. 1378–1380, Dec. 2011.
- [15] G. Sarkis and W. Gross, "Increasing the throughput of polar decoders," *IEEE Commun. Lett.*, vol. 17, no. 4, pp. 725–728, Apr. 2013.
- [16] C. Xiong, J. Lin, and Z. Yan, "Symbol-decision successive cancellation list decoder for polar codes," *IEEE Trans. Signal Process.*, vol. 64, no. 3, pp. 675–687, Feb. 2016.
- [17] B. Yuan and K. Parhi, "LLR-based successive-cancellation list decoder for polar codes with multibit decision," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 64, no. 1, pp. 21–25, Jan. 2017.
- [18] S. Hashemi, C. Condo, and W. Gross, "Fast and flexible successive-cancellation list decoders for polar codes," *IEEE Trans. Signal Process.*, vol. 65, no. 21, pp. 5756–5769, Nov. 2017.
- [19] D. Kim and I. Park, "A fast successive cancellation list decoder for polar codes with an early stopping criterion," *IEEE Trans. Signal Process.*, vol. 66, no. 18, pp. 4971–4979, Sep. 2018.
- [20] H. Zhang et al., "A flip-syndrome-list polar decoder architecture for ultra-low-latency communications," *IEEE Access*, vol. 7, pp. 1149–1159, 2019.
- [21] M. H. Ardakani et al., "Fast successive-cancellation-based decoders of polar codes," *IEEE Trans. Commun.*, vol. 67, no. 7, pp. 4562–4574, Jul. 2019.
- [22] H.-Y. Lee et al., "A node-reliability based CRC-aided successive cancellation list polar decoder architecture combined with post-processing," *IEEE Trans. Signal Process.*, vol. 68, pp. 5954–5967, 2020.
- [23] E. Arikan, "From sequential decoding to channel polarization and back again," *arXiv:1908.09594*, 2019.
- [24] R. Fano, "A heuristic discussion of probabilistic decoding," *IEEE Trans. Inf. Theory*, vol. IT-9, no. 2, pp. 64–74, Apr. 1963.
- [25] Y. Polyanskiy, 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.
- [26] I. Márquez-Corbella and J. Tillich, "Attaining capacity with iterated  $(U|U + V)$  codes based on AG codes and Koetter-Vardy soft decoding," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Aachen, Germany, Jun. 2017, pp. 6–10.
- [27] P. Trifonov, "Efficient design and decoding of polar codes," *IEEE Trans. Commun.*, vol. 60, no. 11, pp. 3221–3227, Nov. 2012.
- [28] Y. Wang and K. Narayanan, "Concatenations of polar codes with outer BCH codes and convolutional codes," in *Proc. 52nd Annu. Allerton Conf. Commun., Control, Comput. (ALLERTON)*, 2014, pp. 813–819.
- [29] Y. Wang, K. Narayanan, and Y. Huang, "Interleaved concatenations of polar codes with BCH and convolutional codes," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 2, pp. 267–277, Feb. 2016.
- [30] H. Mahdavifa et al., "Performance limits and practical decoding of interleaved Reed-Solomon polar concatenated codes," *IEEE Trans. Commun.*, vol. 62, no. 5, pp. 1406–1417, May 2014.
- [31] H. Saber and I. Marsland, "Design of generalized concatenated codes based on polar codes with very short outer codes," *IEEE Trans. Veh. Technol.*, vol. 66, no. 4, pp. 3103–3115, Apr. 2017.

- [32] D. Goldin and D. Burshtein, "Performance bounds of concatenated polar coding schemes," *IEEE Trans. Inf. Theory*, vol. 65, no. 11, pp. 7131–7148, Nov. 2019.
- [33] D. Bailon et al., "Concatenated codes based on the Plotkin construction and their soft-input decoding," *IEEE Trans. Commun.*, vol. 70, no. 5, pp. 2939–2950, May 2022.
- [34] U. Wachsmann, R. Fischer, and J. Huber, "Multilevel codes: Theoretical concepts and practical design rules," *IEEE Trans. Inf. Theory*, vol. 45, no. 5, pp. 1361–1391, Jul. 1999.
- [35] J. Cheng and L. Chen, "BCH based U-UV codes and its decoding," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Melbourne, Victoria, Jul. 2021, pp. 1433–1438.
- [36] C. Wu et al., "U-UV coding for bit-interleaved coded modulation," *IEEE Trans. Commun.*, vol. 71, no. 9, pp. 5065–5077, 2023.
- [37] M. Terada, J. Asatani, and T. Koumoto, "Weight distribution." Accessed: May 17, 2022. [Online]. Available: [isec.ec.okayama-u.ac.jp/home/kusaka/wd](http://isec.ec.okayama-u.ac.jp/home/kusaka/wd)
- [38] G. Poltyrev, "Bounds on the decoding error probability of binary linear codes via their spectra," *IEEE Trans. Inf. Theory*, vol. 40, no. 4, pp. 1284–1292, Jul. 1994.
- [39] M. 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.
- [40] A. Balatsoukas-Stimming, M. Parizi, and A. Burg, "LLR-based successive cancellation list decoding of polar codes," *IEEE Trans. Signal Process.*, vol. 63, no. 19, pp. 5165–5179, Oct. 2015.
- [41] C. Yue et al., "A revisit to ordered statistics decoding: Distance distribution and decoding rules," *IEEE Trans. Inf. Theory*, vol. 67, no. 7, pp. 4288–4337, Jul. 2021.
- [42] F. Wang et al., "Adjustable ordered statistic decoder for short block length code towards URLLC," in *Proc. 13th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Changsha, China, Oct. 2021, pp. 1–5.
- [43] A. Valembois and M. Fossorier, "Box and match techniques applied to soft-decision decoding," *IEEE Trans. Inf. Theory*, vol. 50, no. 5, pp. 796–810, May 2004.
- [44] D. Chase, "Class of algorithms for decoding block codes with channel measurement information," *IEEE Trans. Inf. Theory*, vol. IT-18, no. 1, pp. 170–182, Jan. 1972.
- [45] T. Hashimoto, "A list-type reduced-constraint generalization of the Viterbi algorithm," *IEEE Trans. Inf. Theory*, vol. IT-33, no. 6, pp. 866–876, Nov. 1987.
- [46] V. Guruswami and M. Sudan, "Improved decoding of Reed-Solomon and algebraic-geometry codes," *IEEE Trans. Inf. Theory*, vol. 45, no. 6, pp. 1757–1767, Sep. 1999.
- [47] C. Leroux et al., "A semi-parallel successive-cancellation decoder for polar codes," *IEEE Trans. Signal Process.*, vol. 61, no. 2, pp. 289–299, Jan. 2013.
- [48] R. P. Jasinski et al., "An improved GF(2) matrix inverter with linear time complexity," in *Proc. Int. Conf. Reconfig. Comput. FPGAs (ReConFig)*, Cancun, Mexico, 2010, pp. 322–327.
- [49] J. Proakis, *Digital Communications*. New York, NY, USA: McGraw-Hill, 2011.
- [50] E. Arıkan, "Systematic polar coding," *IEEE Commun. Lett.*, vol. 15, no. 8, pp. 860–862, Aug. 2011.
- [51] R. Shao et al., "Two decoding algorithms for tailbiting codes," *IEEE Trans. Commun.*, vol. 51, no. 10, pp. 1658–1665, Oct. 2003.

**Wenhao Chen** received the B.Sc. degree in information engineering from Sun Yat-sen University, Guangzhou, China, in 2021. He is currently working toward the M.Sc. degree in information and communication engineering with Sun Yat-sen University, Guangzhou, China. His research interests include channel coding and data communications.

**Jinjun Cheng** received the B.Sc. degree in communication engineering and the M.Sc. degree in information and communication engineering from Sun Yat-sen University, Guangzhou, China, in 2018 and 2021, respectively. His research interests include channel coding and data communications.

**Changyu Wu** received the B.Sc. degree in electronic information science and technology and the M.Sc. degree in communication and information system from Sun Yat-sen University, Guangzhou, China, in 2019 and 2022, respectively. His research interests include channel coding and data communications.

**Li Chen** (Senior Member, IEEE) received the B.Sc. degree in applied physics from Jinan University, China, in 2003, and the M.Sc. degree in communications and signal processing and the Ph.D. degree in communications engineering from Newcastle University, U.K., in 2004 and 2008, respectively. From 2007 to 2010, he was a Research Associate with Newcastle University. In 2010, he returned to China as a Lecturer with the School of Information Science and Technology, Sun Yat-sen University, Guangzhou. From 2011 to 2012, he was a Visiting Researcher with the Institute of Network Coding, The Chinese University of Hong Kong, where he was an Associate Professor and a Professor from 2011 and 2016. Since 2013, he has been the Associate Head with the Department of Electronic and Communication Engineering (ECE). From July 2015 to October 2015, he was a Visitor with the Institute of Communications Engineering, Ulm University, Germany. From October 2015 to June 2016, he was a Visiting Associate Professor with the Department of Electrical Engineering, University of Notre Dame, USA. From 2017 to 2020, he was the Deputy Dean with the School of Electronics and Communication Engineering. His research interests include information theory, error-correction codes, and data communications. He is a Senior Member of the Chinese Institute of Electronics (CIE). He is a member of the IEEE Information Theory Society Board of Governors and its External Nomination Committee and the Chair of its Conference Committee. He is also the Chair of the IEEE Information Theory Society Guangzhou Chapter. He is an Associate Editor of IEEE TRANSACTION ON COMMUNICATIONS. He has been organizing several international conferences and workshops, including the 2018 IEEE Information Theory Workshop (ITW) at Guangzhou and the 2022 IEEE East Asian School of Information Theory (EASIT) at Shenzhen, for which he is the General Co-Chair. He is also the TPC Co-Chair of the 2022 IEEE/CIC International Conference on Communications in China (ICCC) at Foshan.

**Huazi Zhang** (Senior Member, IEEE) received the B.Sc. and Ph.D. degrees from the Institute of Information and Communication Engineering, Zhejiang University, in 2008 and 2013, respectively. From 2011 to 2013, he was a Visiting Researcher with the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC, USA. From 2013 to 2014, he was a Research Fellow with the School of Computer Engineering, Nanyang Technological University, Singapore. From 2014 to 2015, he was a Research Scholar with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL, USA. He joined Huawei Technologies Company, Ltd. in 2015. Since then, he has engaged in research projects on advanced wireless communications involving channel coding and signal processing techniques, and engaged in multiple standardization activities. His contribution led to the adoption of many state-of-the-art research results into 5G standards, as well as the subsequent commercial rollout. His research interests include channel coding, information theory, and signal processing for wireless communications, with focus on theoretical analysis, algorithm design, and hardware implementations for 5G and beyond.