Permutation Successive Cancellation Decoding of U-UV Codes

Jingyu Lin[†‡], Li Chen[†‡], Fangbo Yi[†‡], Xiaoqian Ye[†‡],
[†] School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou, China
[‡] Guangdong Provincial Key Laboratory of Information Security Technology, Guangzhou, China
Email: linjy228@mail2.sysu.edu.cn, chenli55@mail.sysu.edu.cn, yifb@mail2.sysu.edu.cn, yexq26@mail2.sysu.edu.cn

2025 IEEE Globecom Workshops (GC Wkshps) | 979-8-3315-6741-5/25/$31.00 ©2025 IEEE | DOI: 10.1109/GCWkshps68340.2025.11591018

***Abstract*—U-UV codes are good-performing short-to-medium length channel codes constructed by several small component codes. They are coupled through the $(\mathrm{U}|\mathrm{U}+\mathrm{V})$ recursive structure. With ordered statistic decoding (OSD) for the BCH component codes, the successive cancellation list (SCL) decoding of U-UV codes can outperform that of the cyclic redundancy check (CRC)-polar codes. This paper shows the U-UV codes can be interpreted by block polynomial evaluation and further establishes the equivalence between permutations of component code indices and that of codeword block indices. Based on this understanding, permutation successive cancellation (PSC) decoding of U-UV codes is further proposed. By permuting blocks of received symbol sequence, extra successive cancellation (SC) decoding can be performed, resulting in the proposed PSC decoding. It can outperform the SC decoding by yielding more codeword estimations. A subsequent U-UV code design tailored for PSC decoding is also proposed, which can further increase the decoding performance gain.**

***Index Terms*—generalized concatenated codes, short-to-medium length codes, U-UV codes**

I. Introduction

The next generation communication networks will enable ultra-reliable and low-latency data transmission, in which good-performing short-to-medium length (SML) channel codes are needed. The currently known competent SML channel codes include BCH codes [1] [2], tail-biting convolutional codes [3], cyclic redundancy check (CRC)-polar codes [4]–[6] and polarization adjusted convolutional (PAC) codes [7]. Recently, U-UV codes were proposed as another good-performing SML code [8]–[15]. It is constructed by several component codes of equal length, which are called the U codes and V codes. They are coupled through the $(\mathrm{U}|\mathrm{U}+\mathrm{V})$ recursive structure. This construction is also known as Plotkin construction [8]. For simplicity, we refer as the U-UV construction. This construction also results in polarized subchannels, each of which conveys a component codeword under this U-UV code paradigm. Rates of the component codes can be designed based on the subchannel capacities [11] [13] [16]. The U-UV codes can also be interpreted as the generalized concatenated codes (GCCs), in which the component codes and the polar codes are the outer codes and the inner codes, respectively [16]. It has been shown that successive cancellation list (SCL) decoding of the BCH based U-UV codes can outperform that of the CRC-polar codes [11].

In practical implementations, part of SCL decoding complexity and latency come from decoding path management, i.e., path sorting and pruning. Addressing this challenge, successive cancellation (SC) decoding of polar codes based on permuted factor graphs was proposed in [17]. It runs several instances of SC decoding over the permuted versions of the received symbol sequence in parallel, which is called permutation SC (PSC) decoding. Automorphism group of polar codes, which is a group of permutations that map a codeword into another, was introduced to enhance the PSC decoding performance [18]–[20]. The SC-invariant affine automorphisms were further investigated to improve the PSC decoding efficiency [20]–[23]. For U-UV codes, soft decision decoding based on hidden codewords was proposed in [24]. It utilizes several decoding variants, each of which starts with decoding a different hidden codeword. Different hidden codewords can be uncovered by superimposing different segments of the U-UV codeword, which can be equivalently realized as SC decoding over several modified factor graphs. This paper focuses on SC decoding based on modified factor graphs that are obtained through permuting factor graph layers. They can be equivalently realized by performing SC decoding on permuted received symbol sequence.

This paper proposes the block polynomials for U-UV codes, which help establish the equivalence between permuting the component code indices and permuting the codeword block indices. Based on this understanding, PSC decoding of U-UV codes is further proposed. By permuting blocks of received symbol sequence, extra SC decoding can be performed, resulting in the proposed PSC decoding. It can outperform the SC decoding by yielding more codeword estimations. A subsequent U-UV code design tailored for PSC decoding is also proposed, which can further increase the decoding performance gain.

II. Preliminaries

A. U-UV *Codes*

Let the U code and V code be two linear block codes of length n with dimensions k_{U} and k_{V}, respectively. The one-level U-UV code of length $2n$ and dimension $k_{\mathrm{U}}+k_{\mathrm{V}}$ is constructed as [8]

$$\{(\boldsymbol{u}\,|\,\boldsymbol{u}+\boldsymbol{v});\boldsymbol{u}\in\mathcal{C}_{\mathrm{U}},\boldsymbol{v}\in\mathcal{C}_{\mathrm{V}}\},\qquad(1)$$

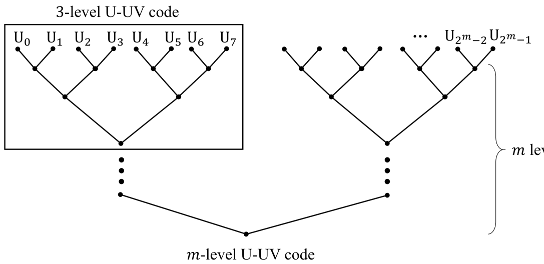

Fig. 1. Recursive construction of an m-level U-UV code.

where $\mathcal{C}_{\mathrm{U}}$ and $\mathcal{C}_{\mathrm{V}}$ denote the codebooks of the U code and V code, respectively. In this work, they are primitive BCH codes. A multi-level U-UV code can be constructed by extending this construction recursively through involving more component U codes and V codes, as is illustrated in Fig. 1. An m-level U-UV code consists of 2^m component codes. Let U_i denote the i-th component code, where $i = 0, 1, \cdots, 2^m - 1$. It is a BCH code of length n and rate $R_i = k_i/n$. The m-level U-UV code has a length of $N = 2^m n$ and a dimension of $K = \sum_{i=0}^{2^m-1} k_i$.

SC decoding of U-UV codes can be realized by recursively computing the received log-likelihood ratio (LLR) vector with respect to each BCH component code and then decoding it with a particular algorithm for the BCH code [11] [13]. To further realize SCL decoding, the component code decoding shall provide a codeword estimation list, e.g., the ordered statistic decoding (OSD) [25] can realize this and is especially competent in decoding BCH codes [11] [13]. In this work, we consider both the Berlekamp-Massey (BM) [26] [27] decoding and the OSD for the BCH component codes. The BM algorithm is selected for its low decoding complexity, while OSD is chosen for its high decoding performance. Integrating with the SC or SCL decoding for the inner polar codes, the decoding for the U-UV codes include the SC-BM, SC-OSD and SCL-OSD decoding.

B. Code Design

The U-UV code design is realized through determining its component code rates. Let W_i denote the i-th polarized subchannel that conveys component code U_i. Let $\mathcal{R}_i$ denote the maximum transmission rate of subchannel W_i. If

$$R_i \leq \mathcal{R}_i, \tag{2}$$

decoding error probability of U_i can be arbitrarily small. When component codeword length n is sufficiently large, $\mathcal{R}_i$ can be characterized by the polarized subchannel capacity $I(\mathrm{W}_i)$. But without this prerequisite on n, $\mathcal{R}_i$ deviates from the capacity due to channel dispersion, resulting in the finite length transmission rate over the subchannel [28]. Hence, it is used to characterize $\mathcal{R}_i$ instead [13]. With equivalent noise variances of the subchannels estimated by Gaussian approximation (GA) [16], subchannel capacity $I(\mathrm{W}_i)$ can be determined. With a desired error probability P_{e}, the maximum transmission rate of subchannel W_i is [28]

$$\mathcal{R}_i = I(\mathrm{W}_i) - \Big(\sqrt{\frac{\mathcal{V}_i}{n}} Q^{-1}(P_{\mathrm{e}}) - \frac{\log_2 n}{2n}\Big), \tag{3}$$

where $Q^{-1}(\cdot)$ is the inverse of Q-function, and $\mathcal{V}_i$ is the subchannel dispersion. Subsequently, the component code rate R_i can be determined based on (2).

To optimize decoding performance, the equal error probability rule should be further adopted to adjust the component code rates [13] [16]. Decoding performances of all component codes should be aligned since the worst performing component code is the decoding bottleneck of the U-UV code. For the SC-OSD or the SCL-OSD decoding, the theoretical maximum likelihood (ML) decoding error probability of the component codes can be applied, since OSD provides a near-ML decoding performance for these component codes. With the subchannel equivalent noise variance and the weight spectrum of BCH codes [29], the tangential bound [30] can be applied to characterize their ML decoding error performances [13]. For the SC-BM decoding, the BM decoding error probability of a BCH component code can also be estimated. Let p_i denote the error probability of an uncoded bit conveyed by subchannel W_i. For a length-n BCH component code correcting up to t_i errors, its decoding error probability is

$$P_{\mathrm{B}i} = \sum_{j=t_i+1}^{n} \binom{n}{j} p_i^j \cdot (1-p_i)^{n-j}. \tag{4}$$

The overall decoding error probability of the U-UV code can be estimated by

$$P_{\text{U-UV}} = 1 - \prod_{i=0}^{2^m-1} (1 - P_{\mathrm{B}i}). \tag{5}$$

Applying the equal error probability rule, the component codes are designed through minimizing $P_{\text{U-UV}}$.

III. PERMUTATION DECODING

A. Block Polynomials of U-UV Codes

Constructed via the U-UV recursive structure, the codeword of an m-level U-UV code can be decomposed into 2^m blocks of length n. Each block is formed through the superposition of specific component codewords, which can be represented by the evaluation of a polynomial. This polynomial is defined as the block polynomial of the U-UV code. Let the monomial set be

$$\mathcal{M} \triangleq \{x_0^{a_0} \cdots x_{m-1}^{a_{m-1}} | (a_0, \cdots, a_{m-1}) \in \mathbb{F}_2^m\}. \tag{6}$$

A one-level U-UV code constructed as in (1) can be denoted by

$$\{(g(0) \,|\, g(1)); g(x_0) = \boldsymbol{u} + \boldsymbol{v} x_0, \boldsymbol{u} \in \mathcal{C}_{\mathrm{U}}, \boldsymbol{v} \in \mathcal{C}_{\mathrm{V}}\}, \tag{7}$$

where $g(x_0)$ is the block polynomial of the U-UV code. For $m \geq 2$, the block polynomial of an m-level U-UV code is

$$\begin{aligned} &g(x_0, \cdots, x_{m-1}) \\ &\quad= g_1(x_0, \cdots, x_{m-2}) + g_2(x_0, \cdots, x_{m-2}) x_{m-1}, \end{aligned} \tag{8}$$

where g_1 and g_2 are the block polynomials of two $(m-1)$-level U-UV codes. Hence, based on induction, the block polynomial of an m-level U-UV code is

$$g(x_0, \cdots, x_{m-1}) = \sum_{i=0}^{2^m-1} \boldsymbol{u}_i x_0^{i_0} \cdots x_{m-1}^{i_{m-1}}, \tag{9}$$

where $\boldsymbol{u}_i \in \mathrm{U}_i$ is the i-th component codeword, and $(i_0, \cdots, i_{m-1}) \in \mathbb{F}_2^m$ is the binary composition of i. The U-UV codeword $\boldsymbol{c}$ can be represented by

$$\boldsymbol{c} = (\boldsymbol{c}_0, \boldsymbol{c}_1, \cdots, \boldsymbol{c}_{2^m-1}), \tag{10}$$

where each length-n coded block $\boldsymbol{c}_j = g(j_0, \cdots, j_{m-1})$ and $(j_0, \cdots, j_{m-1}) \in \mathbb{F}_2^m$ is the binary composition of j. Fig. 2 shows the block polynomial of a 2-level U-UV code.

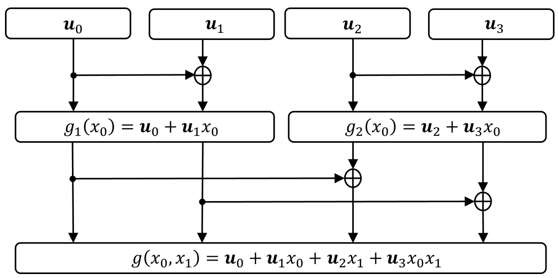

Fig. 2. Block polynomial of a 2-level U-UV Code.

B. From Permutations of Component Codes to Permutations of Codeword Blocks

Let π denote a permutation of the binary composition $(i_0, \cdots, i_{m-1})$. Let $\Pi(i)$ further denote the integer that corresponds to the permuted binary composition $(i_{\pi(0)}, \cdots, i_{\pi(m-1)})$. The following lemma establishes the relationship between permuting the component code indices and permuting the codeword block indices of U-UV codes.

Lemma 1. For an m-level U-UV code $\mathcal{C}$ with length-n component codes, applying bit permutation Π on its component code indices is equivalent to applying the same permutation on its length-n codeword block indices.

Proof: Applying permutation Π on the component code indices of $\mathcal{C}$, one can obtain another U-UV code $\mathcal{C}'$ with block polynomial

$$\begin{aligned} g'(x_0, \cdots, x_{m-1}) &= \sum_{i=0}^{2^m-1} \boldsymbol{u}_{\Pi(i)} x_0^{i_0} \cdots x_{m-1}^{i_{m-1}} \\ &= \sum_{i=0}^{2^m-1} \boldsymbol{u}_i x_0^{i_{\pi^{-1}(0)}} \cdots x_{m-1}^{i_{\pi^{-1}(m-1)}} \\ &= \sum_{i=0}^{2^m-1} \boldsymbol{u}_i x_{\pi(0)}^{i_{\pi(\pi^{-1}(0))}} \cdots x_{\pi(m-1)}^{i_{\pi(\pi^{-1}(m-1))}} \\ &= \sum_{i=0}^{2^m-1} \boldsymbol{u}_i x_{\pi(0)}^{i_0} \cdots x_{\pi(m-1)}^{i_{m-1}} \\ &= g(x_{\pi(0)}, \cdots, x_{\pi(m-1)}), \end{aligned} \tag{11}$$

where π^{-1} is the inverse of π. Therefore, given a U-UV codeword $\boldsymbol{c} \in \mathcal{C}$ as in (10), there exists another codeword $\boldsymbol{c}' \in \mathcal{C}'$, which is

$$\begin{aligned} \boldsymbol{c}' &= (\boldsymbol{c}'_0, \boldsymbol{c}'_1, \cdots, \boldsymbol{c}'_{2^m-1}) \\ &= (\boldsymbol{c}_{\Pi(0)}, \boldsymbol{c}_{\Pi(1)}, \cdots, \boldsymbol{c}_{\Pi(2^m-1)}), \end{aligned} \tag{12}$$

as

$$\begin{aligned} \boldsymbol{c}'_j &= g'(j_0, \cdots, j_{m-1}) \\ &= g(j_{\pi(0)}, \cdots, j_{\pi(m-1)}) \\ &= \boldsymbol{c}_{\Pi(j)}. \end{aligned} \tag{13}$$

Hence, applying permutation Π on the indices of the length-n blocks of $\boldsymbol{c}$ transforms it into a codeword of $\mathcal{C}'$.

□

Based on Lemma 1, any codeword from one U-UV code can be converted into a codeword of another U-UV code that is constructed through permuting the same set of component codes. This implies that SC decoding can be performed on multiple trials, each of which under a different component code order. The inheriting diversity in codeword estimations enables a new list decoding approach.

C. PSC Decoding of U-UV *Codes*

Assume that a U-UV codeword $\boldsymbol{c} = (c_0, c_1, \cdots, c_{2^m n-1}) \in \mathbb{F}_2^{2^m n}$ is transmitted through a discrete memoryless channel using binary phase shift keying (BPSK) modulation of mapping: $0 \mapsto 1; 1 \mapsto -1$. Let $\boldsymbol{L} = (L_0, L_1, \cdots, L_{2^m n-1}) \in \mathbb{R}^{2^m n}$ denote the received LLR vector. It can be partitioned into 2^m blocks of length n, i.e.,

$$\boldsymbol{L} = (\boldsymbol{L}_0, \boldsymbol{L}_1, \cdots, \boldsymbol{L}_{2^m-1}), \tag{14}$$

where $\boldsymbol{L}_i \in \mathbb{R}^n$ and $0 \le i \le 2^m - 1$. Applying Π on the block indices of $\boldsymbol{L}$, one can obtain

$$\begin{aligned} \boldsymbol{L}_{\mathrm{p}} &= \Pi(\boldsymbol{L}) \\ &= (\boldsymbol{L}_{\Pi(0)}, \boldsymbol{L}_{\Pi(1)}, \cdots, \boldsymbol{L}_{\Pi(2^m-1)}). \end{aligned} \tag{15}$$

It can be regarded as the received LLR vector of the U-UV code with component codes

$$\mathrm{U}_{\Pi(0)}, \mathrm{U}_{\Pi(1)}, \cdots, \mathrm{U}_{\Pi(2^m-1)}. \tag{16}$$

It should be pointed out that the component code rates of the original U-UV code are designed based on the maximum transmission rates of the polarized subchannels. Permuting the indices of component codes will result in a mismatch between the component code rates and the maximum transmission rates of the polarized subchannels. Hence, SC decoding with $\boldsymbol{L}_{\mathrm{p}}$ yields inferior performance compared to that with $\boldsymbol{L}$. SC decoding with $\boldsymbol{L}_{\mathrm{p}}$ shall perform as an addition to the original SC decoding, aiming to enhance the decoding performance by providing extra decoding estimations.

Let $\mathbb{P} = \{\Pi_s | s = 0, 1, \cdots, m! - 1\}$ denote the set of all bit permutations of component code indices. The SC decoding performances with each permutation can be evaluated through simulation. For PSC decoding with a designed output list size of l, one can select l best-performing permutations from $\mathbb{P}$ to form a permutation subset $\mathbb{P}_{\mathrm{b}}$, where $2 \le l \le m!$. Based on each permutation in $\mathbb{P}_{\mathrm{b}}$, l instances of SC-BM or SC-OSD decoding can be performed in parallel with the permuted LLR vectors. They will produce at most l codeword estimations, i.e., $\hat{\boldsymbol{c}}^{(0)}, \hat{\boldsymbol{c}}^{(1)}, \cdots, \hat{\boldsymbol{c}}^{(l-1)}$. Their correlation distances with $\boldsymbol{L}$ are defined as

$$\lambda(\boldsymbol{L}, \hat{\boldsymbol{c}}^{(s)}) = \sum_{j: L_j \cdot (1-2\hat{c}_j^{(s)})<0} |L_j|. \tag{17}$$

The codeword estimation that yield the minimum correlation distance with $\boldsymbol{L}$ is selected as the decoding output $\hat{\boldsymbol{c}}$. The above PSC decoding is summarized as in Algorithm 1. Note that SCDecoding$(\cdot)$ is SC-BM or SC-OSD decoding and Π_s^{-1} is the inverse of Π_s.

Algorithm 1: PSC Decoding of U-UV Codes

Input: $\boldsymbol{L}$, $\mathbb{P}_\mathrm{b}$;
Output: $\hat{\boldsymbol{c}}$;
For $\Pi_s \in \mathbb{P}_\mathrm{b}, s = 0, \ldots, l-1$ **do**
  $\boldsymbol{L}_\mathrm{p}^{(s)} = \Pi_s(\boldsymbol{L})$;
  $\hat{\boldsymbol{u}}_\mathrm{p}^{(s)} = \mathrm{SCDecoding}(\boldsymbol{L}_\mathrm{p}^{(s)})$;
  $\hat{\boldsymbol{u}}^{(s)} = \Pi_s^{-1}(\hat{\boldsymbol{u}}_\mathrm{p}^{(s)})$;
  Reconstruct $\hat{\boldsymbol{c}}^{(s)}$ as in (1);
$\hat{\boldsymbol{c}} = \arg\min_{s=0,\ldots,l-1} \lambda(\boldsymbol{L}, \hat{\boldsymbol{c}}^{(s)})$;

IV. Code Redesign Based on Permutation Decoding

In Section II-B, the component code rates of U-UV codes are designed such that they do not exceed the maximum transmission rates of their respective polarized subchannels. Hence, to improve the performance of the proposed PSC decoding, the component code rates should be redesigned, since some component code rates can be beyond the maximum transmission rates of their relocated subchannels in some permutation decoding instances. In particular, for a U-UV code with component codes $\mathrm{U}_{\Pi(0)}, \mathrm{U}_{\Pi(1)}, \cdots, \mathrm{U}_{\Pi(2^m-1)}$, $\mathrm{U}_{\Pi(i)}$ is transmitted through subchannel W_i. Since the rate of $\mathrm{U}_{\Pi(i)}$ was designed based on the maximum transmission rate of subchannel $\mathrm{W}_{\Pi(i)}$, it may exceed that of subchannel W_i, limiting the U-UV code's performance. Hence, the rate of $\mathrm{U}_{\Pi(i)}$ should be readjusted based on the maximum transmission rate of subchannel W_i.

For PSC decoding with a designed output list size of l and a permutation set $\mathbb{P}_\mathrm{b}$, if

$$R_i \leq \min_{s=0,\cdots,l-1} \mathcal{R}_{\Pi_s(i)}, \tag{18}$$

where $\Pi_s \in \mathbb{P}_\mathrm{b}$, error probability of U_i can be arbitrarily small in all permutation decoding instances. Let R'_i denote the rate of the i-th component code of an m-level U-UV code with rate R'. If the U-UV code is designed based on the approach introduced in Section II-B,

$$R'_i \leq \mathcal{R}_i. \tag{19}$$

Hence, to design another m-level U-UV code that satisfies the conditions of (18), one can set its i-th component code rate as

$$R^*_i = \min_{s=0,\cdots,l-1} R'_{\Pi_s(i)}. \tag{20}$$

As a result, the overall code rate $R^* \leq R'$. Rather than selecting permutations based on their performances in the original U-UV codes, the permutation set $\mathbb{P}_\mathrm{b}$ for the newly designed U-UV code should instead be optimized to minimize this rate loss. Hence, permutations that result in least variation in the component code rates should be adopted. Define

$$D_s = \sum_{i=0}^{2^m-1} |R'_{\Pi_s(i)} - R'_i|, \tag{21}$$

where $\Pi_s \in \mathbb{P}$. $\mathbb{P}_\mathrm{b}$ should be formed by l permutations in $\mathbb{P}$ with smallest D_s.

Therefore, to design an m-level U-UV code with a targeted code rate of R^* for the PSC decoding with a output list size of l, the design approach can be reformulated as:

1) Design an m-level U-UV code with rate $R' \geq R^*$ based on the approach of Section II-B.
2) Select l permutations from $\mathbb{P}$ that yield the smallest D_s, forming the permutation subset $\mathbb{P}_\mathrm{b}$.
3) Set $R^*_i = \min_{s=0,\cdots,l-1} R'_{\Pi_s(i)}$, where $\Pi_s \in \mathbb{P}_\mathrm{b}$.

Note that this approach may not produce a U-UV code with the exact rate of R^*. One can perform it several times with different R' to obtain a U-UV code with a rate that is close to R^*.

V. Simulation Results

This section shows our simulation results on PSC decoding of U-UV codes. They are obtained over the additive white Gaussian noise (AWGN) channel using BPSK modulation, where noise variance is $\frac{N_0}{2}$ and N_0 is the single side-band power spectrum density. The signal-to-noise ratio (SNR) is defined as $\frac{E_\mathrm{b}}{N_0}$, where E_b is the transmitted energy per information bit. Note that PSC decoding with l SC-BM decoding instances are denoted as PSC-BM (l) and that with l SC-OSD decoding instances are denoted as PSC-OSD (l).

Fig. 3 shows the frame error rate (FER) of PSC-BM decoding of a 3-level $(248, 122)$ U-UV code, which was defined as $\mathcal{C}_1$ in Table I. The BM and Chase-BM [31] decoding performance of the $(255, 123)$ BCH code is also provided, such that both the U-UV code and the single BCH code have similar length and rate. Note that Chase-BM decoding that flips the η least reliable positions, denoted as Chase-BM (η), will perform BM decoding on 2^η test-vectors. Fig. 3 shows that PSC-BM decoding with $l = 2$ yields 0.2 dB performance gain at the FER of 10^{-5} over the SC-BM decoding, while decoding with $l = 4$ further increases the gain to 0.25 dB. PSC decoding of $\mathcal{C}_1$ can substantially outperform Chase-BM decoding of the $(255, 123)$ BCH code in low-to-moderate SNR region. But in high SNR region, the BCH code has better performance due to its minimum distance advantage. Table II further compares the decoding complexity of the two coding schemes at the SNR of 5 dB. It shows that with the same decoding output list size, decoding the U-UV code requires less than a tenth of finite field operations than decoding the BCH code, which is also conducted over a far smaller finite field.

TABLE I
EXAMPLES OF BCH BASED U-UV CODES

	(N, K)	U_0	U_1	U_2	U_3	U_4	U_5	U_6	U_7	Code design
$\mathcal{C}_1$	$(248, 122)$	$(31, 26)$	$(31, 26)$	$(31, 26)$	$(31, 11)$	$(31, 21)$	$(31, 6)$	$(31, 6)$	$(31, 0)$	[13]
$\mathcal{C}_2$	$(504, 201)$	$(63, 57)$	$(63, 39)$	$(63, 45)$	$(63, 7)$	$(63, 39)$	$(63, 7)$	$(63, 7)$	$(63, 0)$	Proposed
$\mathcal{C}_3$	$(504, 200)$	$(63, 57)$	$(63, 45)$	$(63, 39)$	$(63, 16)$	$(63, 36)$	$(63, 7)$	$(63, 0)$	$(63, 0)$	[13]
$\mathcal{C}_4$	$(504, 270)$	$(63, 57)$	$(63, 51)$	$(63, 51)$	$(63, 24)$	$(63, 51)$	$(63, 18)$	$(63, 18)$	$(63, 0)$	Proposed
$\mathcal{C}_5$	$(504, 268)$	$(63, 57)$	$(63, 51)$	$(63, 51)$	$(63, 30)$	$(63, 45)$	$(63, 18)$	$(63, 16)$	$(63, 0)$	[13]

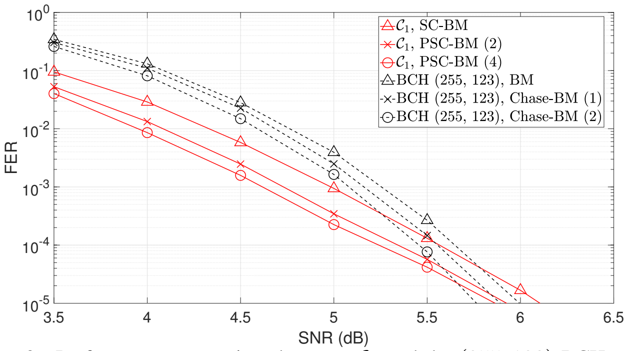

Fig. 3. Performance comparison between $\mathcal{C}_1$ and the $(255, 123)$ BCH code.

TABLE II
DECODING COMPLEXITY COMPARISON BETWEEN $\mathcal{C}_1$ AND THE $(255, 123)$ BCH CODE

$\mathcal{C}_1$	$\mathbb{F}_{2^5}$ oper.	BCH $(255, 123)$	$\mathbb{F}_{2^8}$ oper.
SC-BM	1.36×10^3	BM	1.31×10^4
PSC-BM(2)	2.70×10^3	Chase-BM(1)	3.57×10^4
PSC-BM(4)	5.38×10^3	Chase-BM(2)	7.14×10^4

Fig. 4 shows the FER performance of two U-UV codes with different designs under the SC-OSD, SCL-OSD and PSC-OSD decoding. $\mathcal{C}_2$ is a $(504, 201)$ U-UV code constructed based on the proposed design approach in Section IV. $\mathcal{C}_3$ is a $(504, 200)$ U-UV code constructed based on the design approach in Section II-B. Their component codes are illustrated in Table I. They are all decoded by the OSD with orders that produce a near ML decoding performance of them individually. It shows that $\mathcal{C}_2$ under the SC-OSD decoding exhibits a performance degradation of 0.35 dB compared to $\mathcal{C}_3$. However, greater performance gain can be achieved by increasing the output list size of the PSC-OSD decoding of $\mathcal{C}_2$ compared to $\mathcal{C}_3$. It can be seen that PSC-OSD decoding of $\mathcal{C}_2$ with $l = 2$ yields a 0.3 dB performance gain over SC-OSD decoding at a FER of 10^{-5}, while decoding with $l = 4$ further increases the gain to 0.75 dB. Moreover, PSC-OSD decoding of $\mathcal{C}_2$ with $l = 4$ yields a 0.15 dB performance gain over SCL-OSD decoding of $\mathcal{C}_3$ with the same output list size at a FER of 10^{-5} .

Similarly, Fig. 5 compares the FER performance of $\mathcal{C}_4$ and $\mathcal{C}_5$. Their component codes are also illustrated in Table I. It shows that $\mathcal{C}_4$ under the SC-OSD decoding exhibits a performance degradation of 0.1 dB compared to $\mathcal{C}_5$. But PSC-OSD decoding of $\mathcal{C}_4$ with $l = 2$ yields a 0.5 dB performance gain over SC-OSD decoding at a FER of 10^{-5}, while decoding with $l = 4$ further increases the gain to 0.65 dB. Moreover, the list-two PSC-OSD decoding performance of $\mathcal{C}_4$ is even better than the list-four SCL-OSD decoding performance of both codes. For $\mathcal{C}_4$, PSC-OSD decoding with $l = 4$ yields a 0.25 dB performance gain over SCL-OSD decoding with the same output list size at a FER of 10^{-5} .

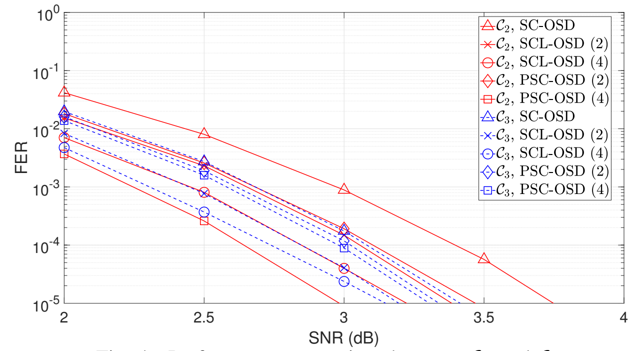

Fig. 4. Performance comparison between $\mathcal{C}_2$ and $\mathcal{C}_3$.

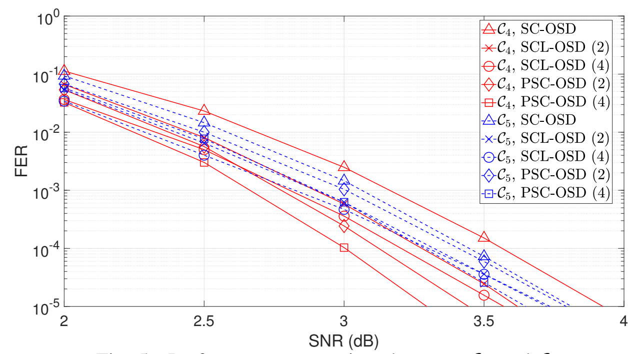

Fig. 5. Performance comparison between $\mathcal{C}_4$ and $\mathcal{C}_5$.

It should be noted that the list-l PSC-OSD decoding runs l independent SC-OSD decoding instances in parallel, without any decoding path management. In contrast, SCL-OSD with the same output list size must maintain and sort l SC-OSD decoding paths. Hence, PSC-OSD decoding, combined with the proposed component code rates design, provides an effective alternative decoding scheme for U-UV codes, offering competitive performance without the overhead of path management.

ACKNOWLEDGEMENTS

This work is supported in part by the National Natural Science Foundation of China (NSFC) with grant ID 62471503 and in part by the Natural Science Foundation of Guangdong Province with grant ID 2024A1515010213.

References

[1] R. Bose and D. Ray-Chaudhuri, “On a class of error correcting binary group codes,” *Inf. and Contr.*, vol. 3, no. 1, pp. 68–79, 1960.

[2] A. Hocquenghem, “Codes correcteurs d’erreurs. chiffres (paris), 2, 147-156,” *Math. Rev*, vol. 22, p. 652, 1959.

[3] H. Ma and J. Wolf, “On tail biting convolutional codes,” *IEEE Trans. Commun.*, vol. 34, no. 2, pp. 104–111, Jan. 1986.

[4] 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, Jun. 2009.

[5] K. Niu and K. Chen, “CRC-aided decoding of polar codes,” *IEEE Commun. Lett.*, vol. 16, no. 10, pp. 1668–1671, Oct. 2012.

[6] I. Tal and A. Vardy, “List decoding of polar codes,” *IEEE Trans. Inf. Theory*, vol. 61, no. 5, pp. 2213–2226, Mar. 2015.

[7] E. Arıkan, “From sequential decoding to channel polarization and back again,” *preprint available as arXiv:1908.09594*, Aug. 2019.

[8] M. Plotkin, “Binary codes with specified minimum distance,” *IRE Trans. Inf. Theory*, vol. 6, no. 4, pp. 445–450, Sep. 1960.

[9] Y. Wang, K. Narayanan, and Y. Huang, “Interleaved concatenations of polar codes with BCH and convolutional codes,” *IEEE J. Select. Areas Commun.*, vol. 34, no. 2, pp. 267–277, Feb. 2016.

[10] I. Márquez-Corbella and J. Tillich, “Attaining capacity with iterated $(\mathrm{U}|\mathrm{U}+\mathrm{V})$ codes based on AG codes and Koetter-Vardy soft decoding,” in *IEEE Int. Symp. Inf. Theory (ISIT)*, Aachen, Germany, Jun. 2017, pp. 6–10.

[11] J. Cheng and L. Chen, “BCH based U-UV codes and its decoding,” in *IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2021.

[12] D. Bailon, M. Bossert, J. Thiers and *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.

[13] W. Chen, J. Cheng, C. Wu and *et al.*, “BCH based U-UV codes and its SCL decoding,” *IEEE Trans. on Signal Process.*, vol. 72, pp. 1286–1300, Feb. 2024.

[14] C. Wu, W. Chen and *et al.*, “U-UV coding for bit-interleaved coded modulation,” *IEEE Trans. Commun.*, vol. 71, no. 9, pp. 5065–5077, Jun. 2023.

[15] W. Chen, L. Chen, J. Lin and *et al.*, “Low complexity successive cancellation list decoding of U-UV codes,” *China Commun.*, pp. 1–20, Jan. 2024.

[16] P. Trifonov, “Efficient design and decoding of polar codes,” *IEEE Trans. Commun.*, vol. 60, no. 11, pp. 3221–3227, Nov. 2012.

[17] N. Doan, S. A. Hashemi, M. Mondelli and *et al.*, “On the decoding of polar codes on permuted factor graphs,” in *IEEE Global Commun. Conf. (GLOBECOM)*, Abu Dhabi, UAE, Dec. 2018, pp. 1–6.

[18] M. Bardet, V. Dragoi, A. Otmani and *et al.*, “Algebraic properties of polar codes from a new polynomial formalism,” in *IEEE Int. Symp. Inf. Theory (ISIT)*, Barcelona, Spain, Jul. 2016, pp. 230–234.

[19] M. Kamenev, Y. Kameneva, O. Kurmaev and *et al.*, “Permutation decoding of polar codes,” in *2019 XVI Int. Symp. Problems Redundancy Inf. Control Syst. (REDUNDANCY)*, Moscow, Russia, Oct. 2019, pp. 1–6.

[20] M. Geiselhart, A. Elkelesh, M. Ebada and *et al.*, “On the automorphism group of polar codes,” in *IEEE Int. Symp. Inf. Theory (ISIT)*, Melbourne, Australia, Jul. 2021, pp. 1230–1235.

[21] C. Pillet, V. Bioglio, and I. Land, “Polar codes for automorphism ensemble decoding,” in *IEEE Inf. Theory Workshop (ITW)*, Kanazawa, Japan, Oct. 2021, pp. 1–6.

[22] C. Pillet, V. Bioglio, and I. Land, “Classification of automorphisms for the decoding of polar codes,” in *ICC 2022 - IEEE Int. Conf. on Commun.*, Seoul, Korea, May 2022, pp. 110–115.

[23] Z. Ye, Y. Li, H. Zhang and *et al.*, “Affine automorphism group of polar codes,” *IEEE Trans. Inf. Theory*, vol. 70, no. 9, pp. 6280–6296, Jul. 2024.

[24] M. Bossert, “Soft decision decoding of recursive Plotkin constructions based on hidden code words,” *IEEE Trans. Inf. Theory*, vol. 71, no. 6, pp. 4228 – 4249, Jun. 2025.

[25] 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, Sept. 1995.

[26] E. R. Berlekamp, *Algebraic coding theory*. New York: McGraw-Hill, 1968.

[27] J. Massey, “Shift-register synthesis and bch decoding,” *IEEE trans. Inf. Theory*, vol. 15, no. 1, pp. 122–127, Jan. 2003.

[28] Y. Polyanskiy, H. V. Poor, and S. Verdu, “Channel coding rate in the finite blocklength regime,” *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2307–2359, Apr. 2010.

[29] M. Terada, J. Asatani, and T. Koumoto, “Weight Distribution.” [Online]. Available: isec.ec.okayama-u.ac.jp/home/kusaka/wd.

[30] 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.

[31] D. Chase, “Class of algorithms for decoding block codes with channel measurement information,” *IEEE Trans. Inf. Theory*, vol. 18, no. 1, pp. 170–182, Jan. 1972.