Modified PAC Codes

Zuoxin Cai†, Li Chen†, Wenxin Liu† and Huazi Zhang‡
† School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou, China, 510006
‡ Hangzhou Research Center, Huawei Technologies Co. Ltd., Hangzhou, China, 310052
Email: caizx7@mail2.sysu.edu.cn, chenli55@mail.sysu.edu.cn, liuwx6@mail2.sysu.edu.cn, zhanghuazi@huawei.com

2023 IEEE International Symposium on Information Theory (ISIT) | 978-1-6654-7554-9/23/$31.00 ©2023 IEEE | DOI: 10.1109/ISIT54713.2023.10206642

***Abstract*—Polarization-adjusted convolutional (PAC) codes can approach the normal approximation (NA) bound using Fano decoding. However, when the received information is unreliable, the decoding may linger over the decoding tree, resulting in both a high decoding complexity and latency. This paper proposes the modified PAC (MPAC) codes and their hybrid Fano-SC (HFSC) decoding that prevents an impractical decoding. For the MPAC codes, only a subset of the information bits undergo the convolutional transform. Its output then concatenates the remaining information bits for the inner polar transform. Consequently, the Fano decoding and the successive cancellation (SC) decoding are deployed to recover the information bits that have undergone the convolutional transform and the remaining information bits, respectively. Both the MPAC code design and the HFSC decoding insight are studied. Our simulation results show that with a limited complexity, HFSC decoding of MPAC codes can yield a better performance-complexity tradeoff than Fano decoding of PAC codes and SC list (SCL) decoding of cyclic redundancy check (CRC)-polar codes.**

***Index Terms*—Fano decoding, polarization-adjusted convolutional codes, polar codes, performance-complexity tradeoff**

I. Introduction

Polar codes [1] have been proved to achieve the capacity of any binary input discrete memoryless channel (BI-DMC) using the successive cancellation (SC) decoding. The decoding complexity is $O(N \log_2 N)$, where N is the codeword length. However, the SC decoding performances of short-to-medium length polar codes remain limited. This is primarily due to the suboptimality of the SC decoding. By maintaining L instances of the SC decoders, the SC list (SCL) decoding [2] can approach the maximum likelihood (ML) decoding performance with sufficiently large L and a complexity of $O(LN \log_2 N)$. Further improvement can be obtained by applying an outer transform using the cyclic redundancy check (CRC) codes [3], the parity check (PC) codes [4] or the convolutional codes [5].

Recently, the polarization-adjusted convolutional (PAC) codes [6] that concatenate an outer rate-1 convolutional code and an inner polar code were proposed. It has been shown that with Fano decoding, a length-128 rate-1/2 PAC code can approach the normal approximation (NA) bound. The PAC codes are constructed by the Reed-Muller (RM) rate-profile, which results in a limited rate choice. Several rate-profile methods such as the Monte-Carlo, the RM-polar and the weighted sum (WS) have been investigated in [7] and [8]. They provide a flexible rate choice. Meanwhile, construction approaches of [9] and [10] can both reduce the number of minimum weight codewords of PAC codes, yielding an enhanced decoding performance. However, the Fano decoding may linger over the decoding tree without providing a complete codeword estimation. This causes an infeasible decoding complexity and hence latency, especially when the received information is unreliable. To reduce the Fano decoding complexity, the path metric functions with subchannel cutoff rates as bias and the adaptive bias have been proposed in [11] and [12], respectively. However, it is still challenging to apply Fano decoding for PAC codes of length-512 or above.

This paper proposes the modified PAC (MPAC) codes in order to enable a more practical decoding while maintaining a high decoding performance. For the MPAC codes, a subset of the information bits first undergo the outer convolutional transform. Its output then concatenates the remaining information bits for the inner polar transform. This can be realized by two rate profilings. Consequently, the decoding can be realized by a hybrid Fano-SC (HFSC) mechanism, namely HFSC decoding. The convolutionally transformed information bits are recovered by the Fano decoding, while the remaining information bits are recovered by the SC decoding. This modified design of PAC codes and their decoding can yield an improved performance-complexity tradeoff performance. The MPAC code design and the HFSC decoding insight are studied. Our simulation results show that with a limited complexity, HFSC decoding of MPAC codes delivers a better performance than Fano decoding of PAC codes and SCL decoding of CRC-polar codes.

Notation: Given a set $\mathcal{A} \subset \{1, \ldots, N\}$, its cardinality and complement are denoted by $|\mathcal{A}|$ and $\mathcal{A}^c$, respectively. Let $\mathbb{F}_2 = \{0, 1\}$ denote the binary field. We use $\underline{a}_1^N$ to denote a vector $(a_1, a_2, \ldots, a_N)$. Given $\underline{a}_1^N$, we further use $\underline{a}_i^j$, where $1 \leq i < j \leq N$, to denote its subvector $(a_i, a_{i+1}, \ldots, a_j)$. Given $\underline{a}_1^N$ and $\mathcal{A}$, we also use $\underline{a}_{\mathcal{A}}$ to denote a subvector $(a_i | i \in \mathcal{A})$.

II. Preliminary

A. Polar Codes and SC Decoding

Polar codes are founded on channel polarization, which consists of channel combining and splitting. Given a BI-DMC $\mathrm{W} : \mathcal{X} \to \mathcal{Y}$, with an input alphabet $\mathcal{X} \in \{0, 1\}$ and an arbitrary output alphabet $\mathcal{Y} \in \mathbb{R}$, channel combining first combines N copies of independent channels W to produce a vector channel $\mathrm{W}_N : \mathcal{X}^N \to \mathcal{Y}^N$, where $N = 2^n$ and $n \in \mathbb{N}^+$. Channel splitting further splits W_N into a set of N polarized subchannels $\mathrm{W}_N^{(i)} : \mathcal{X} \to \mathcal{Y}^N \times \mathcal{X}^{i-1}$, where $1 \leq i \leq N$. Let $I(\mathrm{W})$ and $I(\mathrm{W}_N^{(i)})$ denote the symmetric capacity of W and $\mathrm{W}_N^{(i)}$, respectively. Channel polarization results in $\mathrm{W}_N^{(i)}$ becoming either noiseless with $I(\mathrm{W}_N^{(i)}) \to 1$,

or heavily noisy with $I(\mathrm{W}_N^{(i)}) \to 0$. The channel polarization theorem [1] has shown that when $N \to \infty$, the fraction of noiseless subchannels converges to $I(\mathrm{W})$.

The generator matrix of a length-N polar code is defined as $\mathbf{G}_\mathrm{p} = \mathbf{F}^{\otimes n}$, where $\mathbf{F} = ((1,0),(1,1))^T \in \mathbb{F}_2^{2\times 2}$ is the kernel matrix and $\otimes$ denotes the Kronecker product. The polar codeword $\underline{c}_1^N = (c_1, c_2, \ldots, c_N) \in \mathbb{F}_2^N$ is generated by $\underline{c}_1^N = \underline{u}_1^N \mathbf{G}_\mathrm{p}$, where the information vector $\underline{u}_1^N \in \mathbb{F}_2^N$ is partitioned by the information set $\mathcal{A}$ and the frozen set $\mathcal{A}^c$, such that $\underline{u}_1^N = (\underline{u}_\mathcal{A}, \underline{u}_{\mathcal{A}^c})$. In this paper, we set $\underline{u}_{\mathcal{A}^c} = \underline{0}$.

Assume that codeword $\underline{c}_1^N$ is transmitted using BPSK modulation. Let $\underline{y}_1^N = (y_1, y_2, \ldots, y_N) \in \mathbb{R}^N$ and $\underline{\hat{u}}_1^N = (\hat{u}_1, \hat{u}_2, \ldots, \hat{u}_N) \in \mathbb{F}_2^N$ denote the received symbol vector and the estimated information vector, respectively. The estimation of $\hat{u}_i$ is made based on the decision log-likelihood ratio (LLR)

$$L_N^{(i)}(\underline{y}_1^N, \underline{\hat{u}}_1^{i-1}) = \ln \frac{P(\underline{y}_1^N, \underline{\hat{u}}_1^{i-1} \mid \hat{u}_i = 0)}{P(\underline{y}_1^N, \underline{\hat{u}}_1^{i-1} \mid \hat{u}_i = 1)}, \tag{1}$$

where $P(\underline{y}_1^N, \underline{\hat{u}}_1^{i-1} \mid \hat{u}_i)$ is the transition probability of $\mathrm{W}_N^{(i)}$. The above decision LLRs can be computed recursively via the SC decoding trellis [1] [13]. Decision $\hat{u}_i$ can be made based on the decision LLR of (1). That says for $i \in \mathcal{A}$, $\hat{u}_i = 0$ if $L_N^{(i)}(\underline{y}_1^N, \underline{\hat{u}}_1^{i-1}) \geq 0$, or $\hat{u}_i = 1$ otherwise. For $i \in \mathcal{A}^c$, $\hat{u}_i = 0$.

B. PAC Codes and Fano Decoding

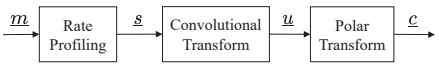

Fig. 1. Block diagram of the PAC codes.

The PAC code concatenates an outer convolutional transform with an inner polar transform, which is shown in Fig. 1. This concatenation improves the weight distribution of the polar code, yielding an improved decoding performance [14] [15].

Let $\underline{m}_1^K = (m_1, m_2, \ldots, m_K) \in \mathbb{F}_2^K$ and $\underline{s}_1^N = (s_1, s_2, \ldots, s_N) \in \mathbb{F}_2^N$ denote the information vector and the input vector of the convolutional transform, respectively. To encoder an (N, K) PAC code, $\underline{m}_1^K$ will be first embedded into $\underline{s}_1^N$ based on the information set $\mathcal{A} \subset \{1, 2, \ldots, N\}$, where $|\mathcal{A}| = K$, such that $\underline{s}_\mathcal{A} = \underline{m}_1^K$ and $\underline{s}_{\mathcal{A}^c} = \underline{0}$. This is the rate profiling for PAC codes. Let $g(x) = g_0 + g_1 x + \cdots + g_m x^m$ denote the generator polynomial of the convolutional transform. Its generator matrix can be written as an upper-triangular Toeplitz matrix $\mathbf{G}_\mathrm{c} \in \mathbb{F}_2^{N\times N}$. Subsequently, the convolutional transform output $\underline{u}_1^N$ is generated by $\underline{u}_1^N = \underline{s}_1^N \mathbf{G}_\mathrm{c}$. The PAC codeword $\underline{c}_1^N$ is further generated by $\underline{c}_1^N = \underline{u}_1^N \mathbf{G}_\mathrm{p}$.

Let $\underline{\hat{u}}_1^N \in \mathbb{F}_2^N$ denote the estimated input of the polar transform. Presenting the SC decoding over the binary decoding tree, the Fano decoding improves the error-correction capability by moving backwards and forwards to pursue a more likely decoding path, as illustrated by Fig. 2. This requires the assistance of a new path metric which can be computed layer-by-layer as [11]

$$\begin{aligned}\mathcal{M}(\underline{\hat{u}}_1^i) = \mathcal{M}(\underline{\hat{u}}_1^{i-1}) + 1.0 + \log_2 P(\hat{u}_i \mid \underline{y}_1^N, \underline{\hat{u}}_1^{i-1}) \\ - E_0(1, \mathrm{W}_N^{(i)}),\end{aligned} \tag{2}$$

where $P(\hat{u}_i \mid \underline{y}_1^N, \underline{\hat{u}}_1^{i-1})$ and $E_0(1, \mathrm{W}_N^{(i)})$ are the *a posteriori* probability of $\hat{u}_i$ and the cutoff rate of subchannel $\mathrm{W}_N^{(i)}$, respectively. The path metric is initialized as $\mathcal{M}\left(\underline{\hat{u}}_1^0\right) = 0$. $\mathcal{M}(\underline{\hat{u}}_1^i)$ and $\mathcal{M}(\underline{\hat{u}}_1^{i-1})$ are the path metrics of the nodes at layers i and $i-1$ of the tree, respectively. Furthermore, $\mathcal{M}(\underline{\hat{u}}_1^i)$ can be computed by updating the current node of $\mathcal{M}(\underline{\hat{u}}_1^{i-1})$ through exploring the branches that correspond to $\hat{u}_i = 0$ and $\hat{u}_i = 1$, respectively. We further let $\{\mathcal{M}(\underline{\hat{u}}_1^i)\}_\mathrm{max}$ and $\{\mathcal{M}(\underline{\hat{u}}_1^i)\}_\mathrm{min}$ denote the larger and smaller $\mathcal{M}(\underline{\hat{u}}_1^i)$ realizations, respectively. Their corresponding decoding paths are denoted by $\{\underline{\hat{u}}_1^i\}_\mathrm{max}$ and $\{\underline{\hat{u}}_1^i\}_\mathrm{min}$. Note that a larger $\mathcal{M}(\underline{\hat{u}}_1^i)$ indicates a more reliable path for $\underline{\hat{u}}_1^i$.

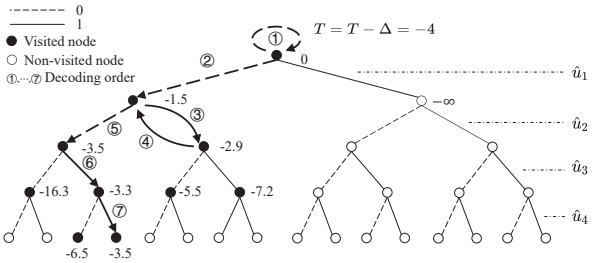

Fig. 2. Fano decoding binary tree with $T = 0$ and $\Delta = 4$ for a polar code with $N = 4$, $\mathcal{A} = \{2, 3, 4\}$ and $\underline{u}_1^4 = (0, 0, 1, 1)$.

Fig. 2 demonstrates the Fano decoding of a polar code with a threshold $T = 0$ and a step size $\Delta = 4$. Note that at current node $\hat{u}_i$, the Fano decoding compares $\mathcal{M}(\underline{\hat{u}}_1^i)$ with T in order to decide whether to move forward or backward as follows:

1) If $\{\mathcal{M}(\underline{\hat{u}}_1^i)\}_\mathrm{max} \geq T$, move forward to $\{\underline{\hat{u}}_1^i\}_\mathrm{max}$;

2) If $\{\mathcal{M}(\underline{\hat{u}}_1^i)\}_\mathrm{max} < T$, move backward to the root node of the tree, or the node $\hat{u}_{i'}$ which satisfies $i' \leq i$, $\{\mathcal{M}(\underline{\hat{u}}_1^{i'})\}_\mathrm{min} > T$ and the path $\{\underline{\hat{u}}_1^{i'}\}_\mathrm{min}$ has not been visited.

Note that T is dynamically adjusted by Δ [16] , so that a root-to-leaf complete path can be found. A node explored in the above procedure is called *a node visit of Fano decoding*.

In decoding a PAC code, the Fano decoding is integrated with the trellis constraint of the convolutional transform. Let $\mathcal{S}^i$ denote the state of the convolutional transform shift registers with an input of $\underline{s}_1^i$. $\mathcal{S}^0$ is initialized to be all zero. With the state $\mathcal{S}^{i-1}$ and an input s_i, the generation of output u_i is denoted by $s_i \underset{\mathcal{S}^{i-1}}{\longmapsto} u_i$. Moreover, state $\mathcal{S}^{i-1}$ is updated to $\mathcal{S}^i$. Assuming that $\hat{u}_i$ ($i \in \mathcal{A}$) has been estimated by Fano decoding, the information bit $\hat{s}_i$ that satisfies the mapping

$$\hat{s}_i \underset{\mathcal{S}^{i-1}}{\longmapsto} \hat{u}_i \tag{3}$$

can be obtained. Meanwhile, for the frozen bit $\hat{s}_i$ where $i \in \mathcal{A}^c$, set $\hat{s}_i = 0$ and obtain $\hat{u}_i$ by mapping $0 \underset{\mathcal{S}^{i-1}}{\longmapsto} \hat{u}_i$. Once the decoding reaches a leaf node of the tree, the complete estimation of $\underline{\hat{u}}_1^N$ and $\underline{\hat{s}}_1^N$ have been obtained.

It should be pointed out that, when the received information is unreliable, the path metrics decrease, leading to excessive backward and threshold-adjusting operations. This makes the Fano decoding linger over the decoding tree, causing an infeasible complexity and latency. It prevents Fano decoding to be applied for larger PAC codes. To overcome this, the following MPAC codes and their HFSC decoding are proposed.

III. The MPAC Codes

A. Encoding

For the MPAC codes, only a subset of the information bits undergo the convolutional transform. Its output together with the remaining information bits then undergo the inner polar transform. The Fano decoding and SC decoding will recover the information bits that have undergone the convolutional transform and those that have not, respectively. They constitute the HFSC decoding. It limits the number of nodes that allow backward computation, reducing the decoding complexity.

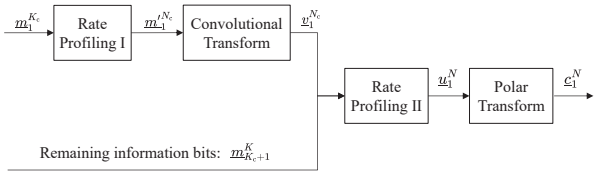

Fig. 3. Block diagram of the MPAC codes.

Fig. 3 shows the block diagram of the MPAC codes. For an (N, K) MPAC code, it can be further specified by the parameters (N_c, K_c), where N_c is the dimension of the convolutional transform as $\mathbf{G}'_c \in \mathbb{F}_2^{N_c \times N_c}$ and K_c is the number of information bits that undergo the transform, respectively. They satisfy $K - K_c + N_c \leq N$. Hence, an (N, K) PAC code can be viewed as a special MPAC code with $N_c = N$ and $K_c = K$.

To encode an (N, K)-(N_c, K_c) MPAC code, the information vector $\underline{m}_1^K$ is partitioned into $\underline{m}_1^K = (\underline{m}_1^{K_c}, \underline{m}_{K_c+1}^K)$, where $\underline{m}_1^{K_c}$ and $\underline{m}_{K_c+1}^K$ denote the information bits that will undergo the convolutional transform and the remaining information bits, respectively. Two rate profilings are needed for the MPAC code. Rate profiling I specifies an index set $\mathcal{C}$, and $|\mathcal{C}| = K_c$. Information bits of $\underline{m}_1^{K_c}$ will be embedded into the input vector of the convolutional transform by $\underline{m}'^{N_c}_1 = (\underline{m}'_{\mathcal{C}}, \underline{m}'_{\mathcal{C}^c})$, where $\underline{m}'_{\mathcal{C}} = \underline{m}_1^{K_c}$ and $\underline{m}'_{\mathcal{C}^c} = \underline{0}$. The convolutionally transformed vector is further generated by $\underline{v}_1^{N_c} = \underline{m}'^{N_c}_1 \mathbf{G}'_c$, where $\underline{v}_1^{N_c} = (v_1, v_2, \ldots, v_{N_c}) \in \mathbb{F}_2^{N_c}$. The vector $\underline{v}_1^{N_c}$ and information $\underline{m}_{K_c+1}^K$ are transmitted through the polarized subchannels, which will be specified by rate profiling II. Let $\mathcal{P}$ and $\mathcal{A}$ denote the index sets of rate profiling II, where $|\mathcal{P}| = N_c$ and $|\mathcal{A}| = K - K_c + N_c$. The input vector $\underline{u}_1^N$ of the inner polar transform are partitioned into $(\underline{u}_{\mathcal{P}}, \underline{u}_{\mathcal{A}\setminus\mathcal{P}}, \underline{u}_{\mathcal{A}^c})$, where $\underline{u}_{\mathcal{P}} = \underline{v}_1^{N_c}$, $\underline{u}_{\mathcal{A}\setminus\mathcal{P}} = \underline{m}_{K_c+1}^K$ and $\underline{u}_{\mathcal{A}^c} = \underline{0}$ are the convolutionally transformed bits, the remaining information bits and the frozen bits, respectively. Finally, the MPAC codeword is generated by $\underline{c}_1^N = \underline{u}_1^N \mathbf{G}_p$. It can be seen that the index sets $\mathcal{A}$, $\mathcal{P}$ and $\mathcal{C}$ determine the design of the MPAC codes, which will be introduced in the following subsection.

B. Code Design

Fig. 4 shows the index sets of the two rate profilings. In particular, for the most reliable subchannels with $I(\mathrm{W}_N^{(i)}) \to 1$, they are used to transmit the information bits of $\underline{m}_{K_c+1}^K$. These subchannels are indexed by $\mathcal{A}\setminus\mathcal{P}$. The less reliable subchannels indexed by $\mathcal{P}$ are used to transmit the convolutionally transformed bits of $\underline{v}_1^{N_c}$. Rate profiling I, which is specified by $\mathcal{C}$, is further designed to provide a good weight distribution for the MPAC codes. To realize this design idea, the Gaussian approximation (GA) [17] and an improved RM-polar (iRMP) ordering that evolves from the RM-polar rule [18] are needed.

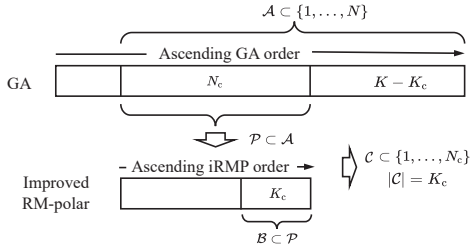

Fig. 4. Index sets of the MPAC codes.

Let $\mathcal{L}(\mathrm{W}_N^{(i)})$ denote the mean LLR value of $\mathrm{W}_N^{(i)}$, which is estimated by GA. Note that the GA is conducted under all-zero codeword assumption. The $\mathrm{W}_N^{(i)}$ with a larger $\mathcal{L}(\mathrm{W}_N^{(i)})$ has a higher reliability and vice versa. Hence, subchannels can be ordered by their reliabilities. Given subchannel indices i_1 and i_2, the GA ordered pair $(i_1, i_2)_{\mathrm{GA}}$ denotes $\mathcal{L}(\mathrm{W}_N^{(i_1)}) > \mathcal{L}(\mathrm{W}_N^{(i_2)})$. Let $\mathrm{r}(i) = \mathrm{b}(i-1)$, where $\mathrm{b}(i-1)$ is the number of 1s in the binary representation of $i-1$ and $1 \leq i \leq N$. The iRMP ordered pair $(i_1, i_2)_{\mathrm{iRMP}}$ is further defined as follows. Given indices i_1 and i_2, it is claimed that $(i_1, i_2)_{\mathrm{iRMP}}$, if $\mathrm{r}(i_1) > \mathrm{r}(i_2)$, or $\mathrm{r}(i_1) = \mathrm{r}(i_2)$ and $(i_1, i_2)_{\mathrm{GA}}$. Note that the iRMP ordering can provide a better weight distribution than the RM-polar ordering for the MPAC codes.

Under the GA order, a refreshed subchannel index tuple $(j_1, j_2, \ldots, j_N)_{\mathrm{GA}}$ can be obtained, which implies

$$\mathcal{L}(\mathrm{W}_N^{(j_1)}) > \mathcal{L}(\mathrm{W}_N^{(j_2)}) > \cdots > \mathcal{L}(\mathrm{W}_N^{(j_N)}). \tag{4}$$

Let $\mathcal{A}$ denote the index set of the $K - K_c + N_c$ most reliable subchannels, where

$$\mathcal{A} = \{j_1, j_2, \ldots, j_{K-K_c+N_c}\} \tag{5}$$

is used in rate profiling II. This is similar to the GA rate-profile in polar coding, except that $|\mathcal{A}| = K - K_c + N_c$. Let

$$\mathcal{P} = \{j_{K-K_c+1}, j_{K-K_c+2}, \ldots, j_{K-K_c+N_c}\} \subset \mathcal{A}, \tag{6}$$

denote the index set of the N_c least reliable subchannels in $\mathcal{A}$. Furthermore, the iRMP order is used to reorganize set $\mathcal{P}$, yielding an N_c-tuple $(j_{[1]}, j_{[2]}, \ldots, j_{[N_c]})_{\mathrm{iRMP}}$, such that

$$(j_{[1]}, j_{[2]})_{\mathrm{iRMP}}, (j_{[2]}, j_{[3]})_{\mathrm{iRMP}}, \cdots, (j_{[N_c-1]}, j_{[N_c]})_{\mathrm{iRMP}} \tag{7}$$

and the $[\,\cdot\,]$ is a mapping $\{1, \ldots, N_c\} \to \{K-K_c+1, \ldots, K-K_c+N_c\}$. For the iRMP ordering, let

$$\mathcal{B} = \{j_{[1]}, j_{[2]}, \ldots, j_{[K_c]}\} \subset \mathcal{P} \tag{8}$$

denote the index set of the information bits that undergo the convolutional transform, where $|\mathcal{B}| = K_c$.

Index set $\mathcal{C}$ characterizes the rate profiling I for the information bits $\underline{m}_1^{K_c}$. In order to construct $\mathcal{C}$, elements of $\mathcal{P}$ and $\mathcal{B}$ need to be sorted in an ascending order, yielding vectors

$\underline{p}_1^{N_c} = (p_1, p_2, \ldots, p_{N_c})$ and $\underline{b}_1^{K_c} = (b_1, b_2, \ldots, b_{K_c})$, respectively, where $p_1 < p_2 < \cdots < p_{N_c}$ and $b_1 < b_2 < \cdots < b_{K_c}$. The index set $\mathcal{C}$ can be further constructed as

$$\mathcal{C} = \{k \mid p_k = b_i, i \in \{1, \ldots, K_c\}\}, \tag{9}$$

where $|\mathcal{C}| = K_c$. Furthermore, the mapping $f_{\mathcal{B}}$ is defined as $\mathcal{B} = \{b_i : 1 \le i \le K_c\} \to \{1, \ldots, K_c\}$. For $b_i \in \mathcal{B}$, we have $f_{\mathcal{B}}(b_i) = i \in \{1, \ldots, K_c\}$, which will be used to determine the indices of $\underline{m}_1^{K_c}$ in the HFSC decoding. Note that all the above mentioned index sets and mapping functions are computed offline for a specific (N, K)-(N_c, K_c) MPAC code.

Example 1. Given an $(8, 5)$-$(4, 2)$ MPAC code, the index sets are determined as below. Table I lists the mean LLR values $\mathcal{L}(\mathrm{W}_N^{(i)})$ of the subchannels and their $\mathrm{r}(i)$ values. An 8-tuple $(8, 7, 6, 4, 5, 3, 2, 1)_{\mathrm{GA}}$ can be obtained. The index set of the 7 most reliable subchannels is $\mathcal{A} = \{8, 7, 6, 4, 5, 3, 2\}$, among which the 4 least reliable subchannels are indexed by $\mathcal{P} = \{4, 5, 3, 2\}$. Further, we have the 4-tuple $(4, 5, 3, 2)_{\mathrm{iRMP}}$, where $(4, 5)_{\mathrm{iRMP}}$, $(5, 3)_{\mathrm{iRMP}}$, $(3, 2)_{\mathrm{iRMP}}$. The index set $\mathcal{B} = \{4, 5\}$ can be further obtained. Sorting the elements of $\mathcal{P}$ and $\mathcal{B}$, we have $\underline{p}_1^4 = (p_1, p_2, p_3, p_4) = (2, 3, 4, 5)$ and $\underline{b}_1^2 = (b_1, b_2) = (4, 5)$. Hence, index set $\mathcal{C} = \{3, 4\}$. Finally, with mapping $f_{\mathcal{B}}$, we have $f_{\mathcal{B}}(4) = 1$ and $f_{\mathcal{B}}(5) = 2$.

TABLE I
$\mathcal{L}(\mathrm{W}_N^{(i)})$ AND $\mathrm{r}(i)$ OF THE SUBCHANNEL INDEX i

index i	1	2	3	4	5	6	7	8
$\mathcal{L}(\mathrm{W}_N^{(i)})$	0.29	2.01	2.74	9.13	3.79	11.57	13.51	32.00
$\mathrm{r}(i)$	0	1	1	2	1	2	2	3

IV. The HFSC Decoding

A. Decoding Mechanism

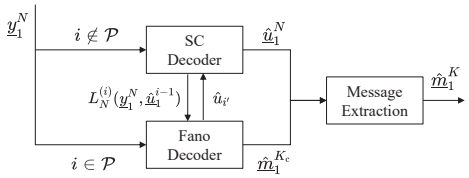

Fig. 5. Block diagram of the HFSC decoding.

Fig. 5 shows the block diagram of the HFSC decoding. It deploys the SC decoding to recover the information bits of $\underline{m}_{K_c+1}^K$ that are transmitted through the most reliable subchannels indexed by $\mathcal{A} \backslash \mathcal{P}$. The Fano decoding will only be deployed to recover the information bits of $\underline{m}_1^{K_c}$ that have undergone the convolutional transform. The convolutionally transformed bits of $\underline{v}_1^{N_c}$ are transmitted through the less reliable subchannels indexed by $\mathcal{P}$. During the HFSC decoding, the decision LLRs are computed by the SC decoder. In order to estimate $\underline{\hat{m}}_1^{K_c}$, the Fano decoder requests the SC decoder for the decision LLRs $L_N^{(i)}(\underline{y}_1^N, \underline{\hat{u}}_1^{i-1})$ and estimates $\hat{u}_{i'}$, where $i' \le i$. If $i' \in \mathcal{B}$, bit $\hat{m}_j$ can be obtained, where $j = f_{\mathcal{B}}(i')$. Note that the estimated $\hat{u}_{i'}$ will be fed back to the SC decoder. Meanwhile, the information bits of $\underline{\hat{m}}_{K_c+1}^K$ are determined based on their decision LLRs $L_N^{(i)}(\underline{y}_1^N, \underline{\hat{u}}_1^{i-1})$. Once the last bit $\hat{u}_N$ is estimated, information vector $\underline{\hat{m}}_1^K$ has been obtained. Algorithm 1 summarizes the HFSC decoding.

Algorithm 1 The Hybrid Fano-SC Decoding

Input: $\underline{y}_1^N$, $\mathcal{A}$, $\mathcal{P}$, $\mathcal{B}$;
Output: $\underline{\hat{m}}_1^K$;
Initialize $j = 1$ and $i = 1$;
While $i \neq N + 1$
  Compute decision LLR $L_N^{(i)}(\underline{y}_1^N, \underline{\hat{u}}_1^{i-1})$ as in (1);
  If $i \in \mathcal{P}$
    Compute path metric $\mathcal{M}(\underline{\hat{u}}_1^i)$ as in (2);
    Perform a node visit of Fano decoding;
    Estimate $\hat{u}_{i'}$ with $i' \le i$ and let $i = i' + 1$;
    If $i' \in \mathcal{B}$
      Let $j = f_{\mathcal{B}}(i')$ and determine $\hat{m}_j$ as in (3);
  Else
    Estimate $\hat{u}_i$ with $i \in \mathcal{A}$ based on $L_N^{(i)}(\underline{y}_1^N, \underline{\hat{u}}_1^{i-1})$;
    Compute $\mathcal{M}(\underline{\hat{u}}_1^i)$ as in (2) and let $i = i + 1$;
Retrieve $\underline{\hat{m}}_1^K$ from $\underline{\hat{m}}_1^{K_c}$ and $\underline{\hat{u}}_1^N$, and terminate decoding.

Note that a computation threshold can also be applied for the HFSC decoding to bound the decoding latency, which will be introduced in the Section V.

B. Decoding Insight

Let $R_c = \frac{K_c}{N_c}$ denote the information rate of the convolutional transform. We consider the MPAC codes with the convolutional generator polynomial $g(x) = 1 + x^2 + x^3 + x^5 + x^6$. They are designed by the GA at the signal-to-noise ratio (SNR) of 0 dB. Fig. 6 shows how the HFSC decoding FER is affected by K_c and R_c. They are rate-$1/2$ MPAC codes with length $N = 128$, and $K_c = 16$, 32 and 48, respectively. The results are obtained over the additive white Gaussian noise (AWGN) channel with a SNR of 3.5 dB. It can be seen that the performance of an MPAC code can be improved by increasing K_c. Note that the polar coding can be seen as the codewords are generated through the linear combination of the rows in $\mathbf{G}_p$. It has been shown that the convolutional transform results in rows in $\mathbf{G}_p$ being included in the minimum weight codewords that generated by polar coding [14] [15]. This inclusion leads to more codewords with large weights, yielding a better weight distribution for the code. In MPAC codes, the number of rows in $\mathbf{G}_p$ that may be included is $N_c - K_c = K_c(\frac{1}{R_c} - 1)$. Given an R_c, $N_c - K_c$ increases with K_c, which leads to more inclusion of rows in $\mathbf{G}_p$. In this way, a larger K_c can result in a better weight distribution for the MPAC codes. Meanwhile, a larger K_c also results in fewer information bits being transmitted through the unreliable subchannels, yielding a better HFSC decoding performance. Empirically, it can be seen that the R_c interval of $(0.45, 0.55)$ can yield a relatively good decoding FER performance, especially when $K_c = 16$. This is due to the fact that when R_c is small, more frozen bits are embedded between the information bits in rate profiling I. It may result in blocks of 0s in $\underline{m'}_1^{N_c}$ and hence the unequal error protection of the convolutional transform [14]. When R_c is large, which

means $N_c - K_c \to 0$, the inclusion of rows is less likely to occur. They both weaken the weight distribution improvement brought by the convolutional transform, resulting in a loss in the decoding FER performance.

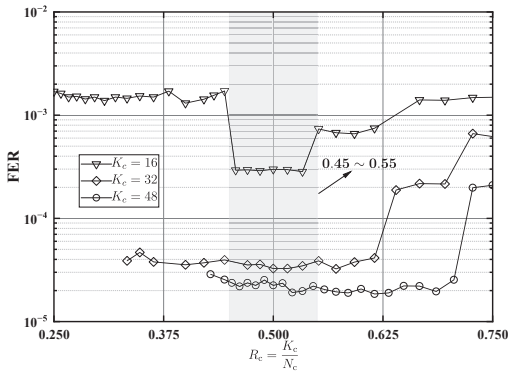

Fig. 6. FER performance of the MPAC codes with different values of K_c and R_c , where $N = 128$ and $K = 64$.

V. Simulation Results

Our simulation results on decoding and complexity performances are obtained over the AWGN channel using BPSK. The MPAC codes with HFSC decoding are compared with relevant coding schemes, including PAC codes with the Fano decoding, CRC-polar codes with the SCL decoding and polar codes with the SC decoding. The MPAC codes with the generator polynomial of $g(x) = 1+x^2+x^3+x^5+x^6$ are designed using GA at the SNR of 0 dB. Based on the above decoding insight, we choose $R_c = 0.5$. Both the CRC-polar codes and the polar codes are designed by the reliability sequence in 5G standard [19]. The CRC generator polynomial is $g_c(x) = 1+x+x^2+x^8$. The PAC codes are constructed by the RM rate-profile. Both the HFSC decoding and Fano decoding have $\Delta = 2$ which is chosen out of the performance-complexity tradeoff.

The decoding complexity is measured as the average number of LLR computations in decoding a codeword. A decoding LLR computation threshold Φ will be introduced to bound the decoding complexity and latency. It is defined as a multiplicity of the SC decoding complexity, i.e., $\Phi = \eta N \log_2 N$, where $\eta \in \mathbb{N}^+$. For both the HFSC decoding and Fano decoding, once the decoding complexity reaches Φ, the SC decoding will be employed to obtain a complete codeword estimation. This prevents the decoding from lingering over the decoding tree. Note that the complexity of all the decoding algorithms are normalized by $N \log_2 N$, namely normalized complexity.

Fig. 7 shows the decoding FER and complexity performances of different coding schemes with $N = 128$ and $K = 64$. Given N and K, the MPAC-(N_c, K_c) code refers to the (N, K)-(N_c, K_c) MPAC code. A decoding computation threshold of $\eta = 128$ is applied. It can be seen that as K_c increases, the FER performances of the MPAC codes improve, which also incurs a slightly increased decoding complexity. In particular, the MPAC-$(96, 48)$ code can slightly outperform the $(128, 64)$ PAC code with a reduced complexity. When K_c reaches K, the MPAC code evolves into the PAC code. However, with a computation threshold, the MPAC code would be more likely to recover the correct information. Furthermore, when compared with the CRC-polar code, the MPAC-$(96, 48)$ code yields 0.25 dB coding gain at the FER of 10^{-4} with a far small decoding complexity.

Fig. 8 shows the decoding FER and complexity performances of different coding schemes with $N = 512$ and $K = 256$. A decoding computation threshold of $\eta = 1024$ is applied. It can be seen that the MPAC-$(128, 64)$ code yields 0.7 dB coding gain over the $(512, 256)$ PAC code at the FER of 10^{-4}, but with a significantly reduced complexity. This again demonstrates that the proposed MPAC codes and the HFSC decoding can achieve a better performance-complexity tradeoff, especially for larger codes. Furthermore, the MPAC-$(128, 64)$ code yields a similar performance as the SCL decoding of the CRC-polar code with $L = 8$. Its HFSC decoding complexity starts to show advantage at the SNR of 2.25 dB. The MPAC-$(128, 64)$ code also substantially outperforms the polar code under the SC decoding with a converging normalized complexity of 1.

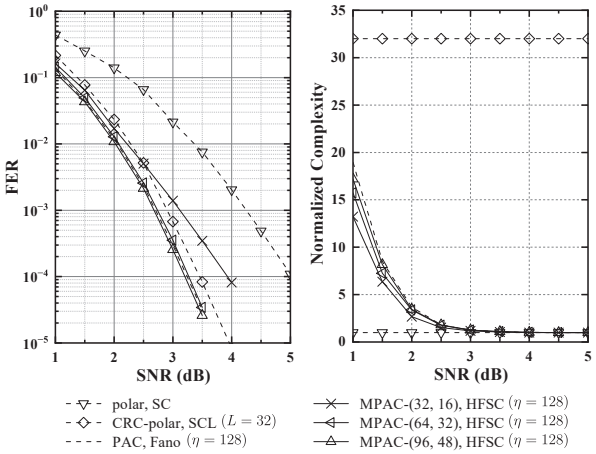

Fig. 7. Performance comparison of codes with $N = 128$ and $K = 64$.

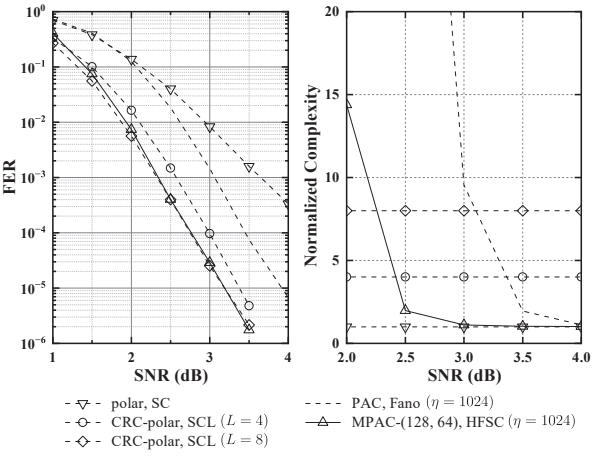

Fig. 8. Performance comparison of codes with $N = 512$ and $K = 256$.

Acknowledgement

This work is sponsored by the National Natural Science Foundation of China (NSFC) with project ID 62071498.

REFERENCES

[1] E. Arıkan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Inform. Theory*, vol. 55, no. 7, pp. 3051–3073, Jul. 2009.

[2] I. Tal and A. Vardy, "List decoding of polar codes," *IEEE Trans. Inform. Theory*, vol. 61, no. 5, pp. 2213–2226, Jul. 2015.

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

[4] H. Zhang, R. Li, J. Wang, S. Dai, G. Zhang, Y. Chen, H. Luo, and J. Wang, "Parity-check polar coding for 5G and beyond," in *Proc. 2018 IEEE Int. Conf. Commun. (ICC)*, Kansas City, MO, USA., May 2018.

[5] A. Fazeli, K. Tian, and A. Vardy, "Viterbi-aided successive-cancellation decoding of polar codes," in *Proc. 2017 IEEE Glob. Commun. Conf. (GLOBECOM)*, Singapore, Singapore, Dec. 2017.

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

[7] M. Moradi and A. Mozammel, "A Monte-Carlo based construction of polarization-adjusted convolutional (PAC) codes," *preprint available as arXiv:2106.08118v1*, Jun. 2021.

[8] W. Liu, L. Chen, and X. Liu, "A weighted sum based construction of PAC codes," *IEEE Commun. Lett.*, vol. 27, no. 1, pp. 28–31, Jan. 2023.

[9] M. Rowshan, S. H. Dau, and E. Viterbo, "Improving the error coefficient of polar codes," in *Proc. 2022 IEEE Inform. Theory Workshop (ITW)*, Mumbai, India, Nov. 2022.

[10] S. Gelincik, P. Mary, A. Savard, and J.-Y. Baudais, "A pre-transformation method to increase the minimum distance of polar-like codes," *preprint available as arXiv:2202.04366*, Feb. 2022.

[11] M. Moradi, "On sequential decoding metric function of polarization-adjusted convolutional (PAC) codes," *IEEE Trans. Commun.*, vol. 69, no. 12, pp. 7913–7922, Dec. 2021.

[12] M. Rowshan, A. Burg, and E. Viterbo, "Polarization-adjusted convolutional (PAC) codes: sequential decoding vs list decoding," *IEEE Trans. Veh. Technol.*, vol. 70, no. 2, pp. 1434–1447, Feb. 2021.

[13] A. Balatsoukas-Stimming, M. B. 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.

[14] M. Rowshan and E. Viterbo, "On convolutional precoding in PAC codes," in *Proc. 2021 IEEE GLOBECOM Workshops*, Madrid, Spain, Dec. 2021.

[15] B. Li, H. Zhang, and J. Gu, "On pre-transformed polar codes," *preprint available as arXiv:1912.06359*, Dec. 2019.

[16] R. Fano, "A heuristic discussion of probabilistic decoding," *IEEE Trans. Inform. Theory*, vol. 9, no. 2, pp. 64–74, Apr. 1963.

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

[18] B. Li, H. Shen, and D. Tse, "A RM-polar codes," *preprint available as arXiv:1407.5483*, Jul. 2014.

[19] "5G NR: Multiplexing and Channel Coding," 3rd Generation Partnership Project (3GPP), TS 38.212 version 15.2.0, Jul. 2018.