

Non-Binary Polar Coded System for the Two-User Multiple-Access Channel

Guan-Chen Liu*, Qi-Yue Yu*

*Communications Research Center, Harbin Institute of Technology, Harbin 150001, China

Email: guanchen_liu@hit.edu.cn, yuqiye@hit.edu.cn

Abstract—This paper presents non-binary polar codes for the two-user multiple-access channel (MAC). The bit error rate (BER) performances of the non-binary polar codes with different kernel factors have been investigated in detail to select a proper parameter from $\text{GF}(q)$ for the generator matrix. Furthermore, the successive cancellation decoding for the non-binary polar codes in the two-user MAC is introduced in detail. Simulation results show that the choice of the kernel factors has a significant impact on the block error rate (BLER) performance; moreover, the non-binary polar codes provide a better BLER performance than their binary counterpart in the two-user MAC.

Index Terms—Non-binary polar codes, two-user multiple-access channel, kernel selection, successive cancellation (SC) decoding.

I. INTRODUCTION

Polar codes have attracted widespread attention since they can achieve the Shannon limit [1], and much research has been done for polar codes in many aspects. Since 2012, polar codes have been considered for the multiple-access channel (MAC) [2]-[5]. Authors in [2] present a joint polarization for the two-user MAC, which results in five possible transmission models that achieve the dominant face of the capacity region. [3] extends the two-user case to the m -user case, $m \geq 2$, and deduces the extremal points of the reachable rate region. However, the proposed joint polarization can only reach some of the capacity region instead of all. Paper [4] proposes compound polar codes for the two-user MAC, which can achieve the whole uniform rate region by changing the decoding order of the joint successive cancellation decoder. In [5], the authors utilize the generalized chain rule to construct polar codes for two-user MACs, achieving all the capacity region.

Moreover, the non-binary (NB) polar codes are also an appealing research field because of their flexible structure [6]-[10]. In [6], the authors exploit the randomized construction to polarize the arbitrary input discrete memoryless channel with the binary kernel. In [7], the authors present a non-binary kernel form that can be polarized if the input size is the power of a prime and the kernel's parameter is the primitive element of the Galois Field $\text{GF}(q)$. A polarized mapping scheme is discussed in [8], which is suited to both the sources and channels. It has been pointed out that multilevel polarization arises when the input size is the power of two [9].

The work presented in this paper was supported in part by the National Natural Science Foundation of China under Grand No. 62071148, and partly supported by the Natural Science Foundation of Heilongjiang Province of China (No. YQ2019F009).

Then non-binary kernels are considered in [10] to construct a system for polarized transmission, resulting in a better BLER performance than binary polar codes.

Some literature works have been done on non-binary polar codes for the MAC due to their appealing features. [11] constructs polar codes using the group structure when the MAC input is arbitrary, achieving the symmetric sum capacity except for some points. In [12], the authors present polarization theorems for arbitrary classical-quantum channels with Arıkan style transformation, which can be used to construct polar codes for arbitrary classical-quantum MACs with relatively low complexity of encoding and decoding. A channel upgradation polar construction is generalized to the non-binary input MAC case to choose the polarization channel for data transmission [13][14].

The prior works mainly focus on the binary codes and the achievability of the rate region in theory. Based on the generalized chain rule proposed by [5], this paper presents a non-binary polar coded system for the two-user MAC, which mainly exploits the flexible design of the non-binary kernel and the successive cancellation (SC) decoding implementation.

The structure of this paper is arranged as follows. Section II presents the system model of the proposed scheme. A successive cancellation decoding algorithm of non-binary polar codes in the two-user MAC is presented in Section III. In Section IV, the selection of the kernel factors is discussed in detail. The simulation results of the proposed system are shown in Section V, followed by concluding remarks in Section VI.

II. SYSTEM MODEL

We define \mathbb{B} , \mathbb{N} and \mathbb{R} as the binary, natural, and real field, respectively. The Galois field $\text{GF}(q)$ is denoted by \mathbb{F}_q , where $q = 2^r$ and $r \in \mathbb{N}$. Considering the element set in \mathbb{F}_q as $\{0 = \gamma^{-\infty}, 1 = \gamma^0, \gamma, \gamma^2, \dots, \gamma^{q-2}\}$, where γ is the primitive element of \mathbb{F}_q . We follow the notations defined in [1], denoting random variables and the corresponding samples by upper and lower case letters, respectively. Besides, A_1^N stands for a vector $(A_1, A_2, \dots, A_i, \dots, A_N)$, and A_i^j denotes the subvector (A_i, \dots, A_j) for $1 \leq i \leq j \leq N$. Let $\mathcal{N}(\mu, \sigma^2)$ represent the Gaussian distribution with the mean μ and the variance σ^2 .

The system model is shown in Fig. 1. At the transmitter, the frozen bits are added to each user's uncoded information while maintaining the sum rate R for the whole system. Then every r bits are converted to a non-binary symbol of \mathbb{F}_q serially. Denote $N = 2^n$, $n \in \mathbb{N}$ as the code length of

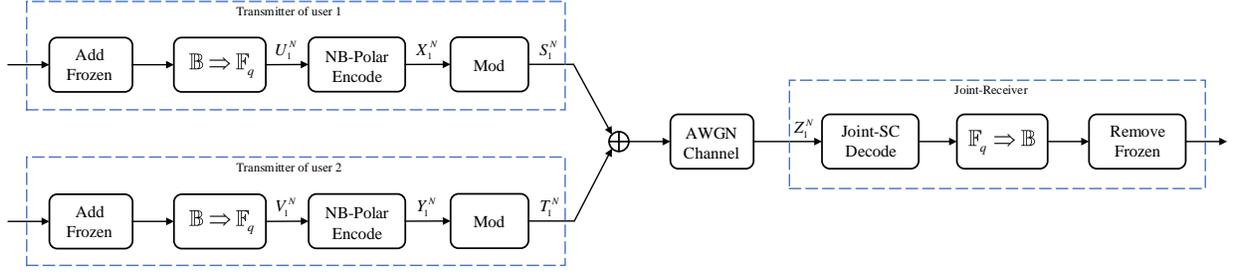


Fig. 1. A diagram of the non-binary polar transmission system in the two-user MAC.

each block in the non-binary symbol form. The non-binary symbol blocks of users 1 and 2 are respectively defined by U_1^N and V_1^N , where $U_i = (U_{i,1}, U_{i,2}, \dots, U_{i,t}, \dots, U_{i,r})$, $V_i = (V_{i,1}, V_{i,2}, \dots, V_{i,t}, \dots, V_{i,r})$, and $U_{i,t}, V_{i,t} \in \mathbb{B}$, $1 \leq i \leq N$, $1 \leq t \leq r$. Define F as the kernel matrix, given by

$$F = \begin{bmatrix} 1 & 0 \\ \alpha & 1 \end{bmatrix}, \quad \alpha \in \mathbb{F}_q. \quad (1)$$

Denote α_u and α_v as the kernel factor for users 1 and 2, respectively. The generator matrix of polar codes is defined by $G_N = B_N F^{\otimes n}$, where $F^{\otimes n}$ denote the n -th Kronecker power of F and B_N is the reverse-shuffle. Note that the operations of addition and multiplication are both based on \mathbb{F}_q .

Let X_1^N and Y_1^N denote the encoded polar codewords of users 1 and 2, given by

$$X_1^N = U_1^N G_N, \quad Y_1^N = V_1^N G_N, \quad (2)$$

with $X_i = (X_{i,1}, X_{i,2}, \dots, X_{i,t}, \dots, X_{i,r})$, $Y_i = (Y_{i,1}, Y_{i,2}, \dots, Y_{i,t}, \dots, Y_{i,r})$, and $X_{i,t}, Y_{i,t} \in \mathbb{B}$. The encoded polar codewords are then converted to bits. By using BPSK modulation, the modulated signals are S_1^N and T_1^N for users 1 and 2, respectively. Two users' modulated signals are then transmitted to the multiple-access channel, and the received signals Z_1^N is given by

$$Z_1^N = S_1^N + T_1^N + K_1^N, \quad (3)$$

where $Z_i = (Z_{i,1}, Z_{i,2}, \dots, Z_{i,t}, \dots, Z_{i,r})$, $K_i = (K_{i,1}, K_{i,2}, \dots, K_{i,t}, \dots, K_{i,r})$, and $Z_{i,t}, K_{i,t} \in \mathbb{R}$. The noise component $K_{i,t}$ satisfies $\mathcal{N}(0, N_0/2)$.

At the receiver, the successive cancellation decoding is used to recover U_1^N and V_1^N , denoted by \hat{U}_1^N and \hat{V}_1^N . After converting \hat{U}_1^N and \hat{V}_1^N to bits and removing the frozen bits, the information bits of two users are obtained. By now, the description of the system framework is accomplished. Some definitions for the channel analysis of this system are given next.

Let $W : \mathcal{X} \times \mathcal{Y} \rightarrow \mathcal{Z}$ be a two-user multiple-access channel. Symbol pair (X_i, Y_i) can be viewed as the input of W , and $W(z_i|x_i, y_i)$ denotes the corresponding transition probability. Set W^N as N independent uses of W . Let \mathcal{X}^N and \mathcal{Y}^N denote the input of W^N from user 1 and user 2, while \mathcal{Z}^N is the output of W^N , then $W^N(z_1^N|x_1^N, y_1^N) = \prod_{i=1}^N W(z_i|x_i, y_i)$.

Like the case in [5], define the combined channel W_N by

$$W_N(z_1^N|u_1^N, v_1^N) = W^N(z_1^N|u_1^N G_N, v_1^N G_N). \quad (4)$$

Since there is no mutual information loss during the polar transform, thus

$$I(Z_1^N; U_1^N, V_1^N) = I(Z_1^N; X_1^N, Y_1^N) = N \cdot I(Z; X, Y). \quad (5)$$

To construct a polarization channel, (5) is expanded as

$$I(Z_1^N; U_1^N, V_1^N) = \sum_{k=1}^{2N} I(Z_1^N; E_k | E_1^{k-1}), \quad (6)$$

where E_1^{2N} is the permutation of $U_1^N V_1^N$ that preserves the monotone order of both U_1^N and V_1^N . Let b_1^{2N} indicate the relative order of E_1^{2N} . When $E_i \in U_1^N$, $b_i = 0$; otherwise, $b_i = 1$.

The coordinate channels $W_N^{(b_k, i, j)}$ that correspond to (6) are defined as

$$W_N^{(b_k, i, j)}(z_1^N, e_1^{k-1} | e_k) = \begin{cases} W_N^{(0, i, j)}(z_1^N, u_1^{i-1}, v_1^j | u_i), & b_k = 0 \\ W_N^{(1, i, j)}(z_1^N, u_1^i, v_1^{j-1} | v_j), & b_k = 1 \end{cases}, \quad (7)$$

where i and j stand for the i -th symbol of user 1 and the j -th symbol of user 2, $0 \leq i, j \leq N$, $1 \leq k = i + j \leq 2N$. The coordinate channel transition probability is used in the decoding process as described in the next section.

III. A JOINT SC DECODING ALGORITHM

In this section, a joint SC decoding is presented for the NB-polar codes in the two-user MAC. Since the generator matrix G_N is determined by the row vector in $F^{\otimes n}$, thus $F_N = F^{\otimes n}$ is used to describe the decoding process, ignoring operation B_N that has no impact on the performance.

To calculate $W_N^{(b_k, i, j)}$, we define the split MAC channel $W_N^{(i, j)}$ by

$$W_N^{(i, j)}(z_1^N, u_1^{i-1}, v_1^{j-1} | u_i, v_j) = \sum_{u_{i+1}^N, v_{j+1}^N} \left(\frac{1}{q^{N-1}} \right)^2 W_N(z_1^N | u_1^N, v_1^N). \quad (8)$$

When considering the decoding order, the single user transition probability is derived as

$$W_N^{(0, i, j)}(z_1^N, \hat{u}_1^{i-1}, \hat{v}_1^j | u_i) = \begin{cases} \sum_{v_1} \frac{1}{q} W_N^{(i, 1)}(z_1^N, \hat{u}_1^{i-1} | u_i, v_1) & \text{if } j = 0 \\ \frac{1}{q} W_N^{(i, j)}(z_1^N, \hat{u}_1^{i-1}, \hat{v}_1^{j-1} | u_i, \hat{v}_j) & \text{if } j > 0 \end{cases}, \quad (9)$$

$$W_N^{(1, i, j)}(z_1^N, \hat{u}_1^i, \hat{v}_1^{j-1} | v_j) = \begin{cases} \sum_{u_1} \frac{1}{q} W_N^{(1, j)}(z_1^N, \hat{v}_1^{j-1} | u_1, v_j) & \text{if } i = 0 \\ \frac{1}{q} W_N^{(i, j)}(z_1^N, \hat{u}_1^{i-1}, \hat{v}_1^{j-1} | \hat{u}_i, v_j) & \text{if } i > 0 \end{cases}. \quad (10)$$

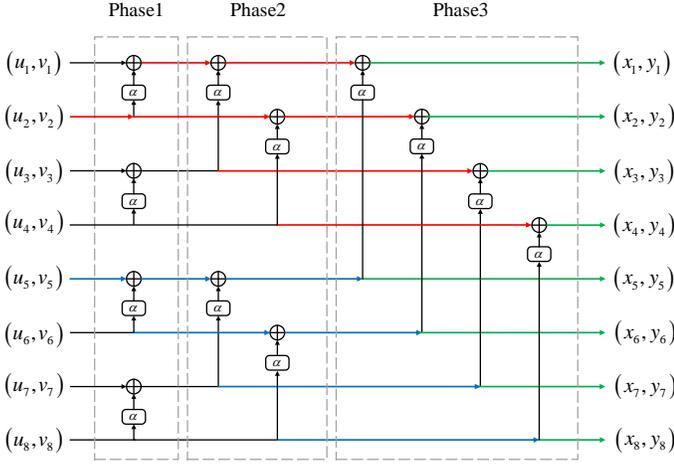


Fig. 2. An example of the decoding path with three phases, where $N = 8$.

The transition probability of the split MAC channel $W_N^{(i,j)}$ can be calculated recursively due to the recursive structure. For simplicity, here we use the notations similar to [5]. Let $\hat{u}_i \triangleq u_{2i-1} + \alpha_{u_1} \cdot u_{2i}$ and $\hat{v}_i \triangleq u_{2i}$, and define $\hat{u}_1^i = u_{1,o}^{2i} + \alpha_{u_1} \cdot u_{1,e}^{2i}$ and $\hat{v}_1^i = u_{1,e}^{2i}$, where the subscripts o and e denote the subvector with odd and even indices that are similar to \hat{u}_1^i and \hat{v}_1^i , respectively. Define

$$D_N^{(i,j)}(z_1^{2N}, u_1^{2i}, v_1^{2j}) \triangleq W_N^{(i,j)}(z_1^{2N}, \hat{u}_1^{i-1}, \hat{v}_1^{j-1} | \hat{u}_i, \hat{v}_j) \cdot W_{N+1}^{(i,j)}(z_{N+1}^{2N}, \hat{u}_1^{i-1}, \hat{v}_1^{j-1} | \hat{u}_i, \hat{v}_j). \quad (11)$$

Then the recursive equations are given by

$$W_{2N}^{(2i-1, 2j-1)}(z_1^{2N}, u_1^{2i-2}, v_1^{2j-2} | u_{2i-1}, v_{2j-1}) = \sum_{u_{2i}, v_{2j}} \frac{1}{q^2} D_N^{(i,j)}(z_1^{2N}, u_1^{2i}, v_1^{2j}), \quad (12)$$

$$W_{2N}^{(2i, 2j-1)}(z_1^{2N}, u_1^{2i-1}, v_1^{2j-2} | u_{2i}, v_{2j-1}) = \sum_{v_{2j}} \frac{1}{q^2} D_N^{(i,j)}(z_1^{2N}, u_1^{2i}, v_1^{2j}), \quad (13)$$

$$W_{2N}^{(2i-1, 2j)}(z_1^{2N}, u_1^{2i-2}, v_1^{2j-1} | u_{2i-1}, v_{2j}) = \sum_{u_{2i}} \frac{1}{q^2} D_N^{(i,j)}(z_1^{2N}, u_1^{2i}, v_1^{2j}), \quad (14)$$

$$W_{2N}^{(2i, 2j)}(z_1^{2N}, u_1^{2i-1}, v_1^{2j-1} | u_{2i}, v_{2j}) = \frac{1}{q^2} D_N^{(i,j)}(z_1^{2N}, u_1^{2i}, v_1^{2j}). \quad (15)$$

According to (12)-(15), the recursive transform implies a decoding path for each split channel. For example, the decoding path of $W_8^{(2,5)}(z_1^8, \hat{u}_1, \hat{v}_1^4 | u_2, v_5)$ is shown in Fig. 2, where red lines, blue lines, and green lines represent the decoding path for user 1, user 2, and both two users, respectively. Assuming that the phase index p increases from the input side to the output side in the polar encoder, $1 \leq p \leq n$. In each phase, two users' decoding paths are combined to calculate the corresponding probabilities.

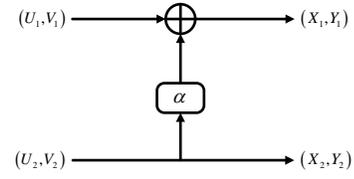


Fig. 3. The basic kernel (or combined) structure of the NB-polar generator.

When we decode symbol $U_i \in U_1^N$, $1 \leq i \leq N$, all the known decisions \hat{V} are viewed as auxiliary symbols, similarly to the symbol $V_i \in V_1^N$. Since the decoding process of U and V are reciprocity, we use the decoding process of U with the auxiliary information V for the analysis in the following.

The transition probability transform corresponding to the decoding path of each phase is based on the combined basic structure, as shown in Fig. 3. The calculation of probability is divided into two categories: f calculation and g calculation, which correspond to decode U_1 and U_2 of the combined basic structure, respectively. According to (12)-(15), considering the number of auxiliary symbols, there are six types of calculations, denoted by $f_{0,v}(P_1, P_2)$, $f_{1,v}(P_1, P_2, V_1)$, $f_{2,v}(P_1, P_2, V_1, V_2)$ and $g_{0,v}(P_1, P_2, U_1)$, $g_{1,v}(P_1, P_2, U_1, V_1)$, $g_{2,v}(P_1, P_2, U_1, V_1, V_2)$, respectively, as shown in Fig. 4. The subscript represents the number of the available auxiliary symbols of V_1^2 . Based on the combined basic structure and calculations, the decoding process of each phase is similar to the single-user case.

A complete binary tree \mathbb{T} of depth $n+1$ is defined first to indicate the SC decoding process [15]. Given a node m , define its location of the decoding tree by the vector (d_m, c_m) , representing the c_m -th node of the d_m -th depth, $1 \leq d_m \leq n+1$, $1 \leq c_m \leq 2^{d_m-1}$. Let the node's parent node, left and right child node be p_m , m_l , and m_r , respectively.

There are four kinds of information stored in the (d_m, c_m) -th node m : the probability matrices Φ_m , decision symbol vector β_m , auxiliary symbol vector θ_m , and child nodes set \mathbb{C}_m . Let $\Phi_m[l]$ be the l -th row in Φ_m , and $\beta_m[l]$, $\theta_m[l]$ be the l -th element in β_m , θ_m , where $1 \leq l \leq \rho_m$, $\rho_m = 2^{(n-d_m+1)}$.

For a two-user decoding algorithm, two decoding trees are defined as \mathbb{T}_u and \mathbb{T}_v , respectively. As stated in [5], for simplicity, $b_{2N} = (0^M 1^N 0^{N-M})$ is defined as our decoding order, where $1 \leq M \leq N$. The decoding order can be naturally divided into three stages: decode U_1^M in stage I, decode V_1^N in stage II, and decode U_{M+1}^N in stage III. Next, we will give a detailed description of the decoding process, consisting of six steps.

Step 1: Initialize Φ_m and \mathbb{C}_m for stage I.

The transition probability $P(z|x, y)$ is initialized to the root node's Φ_m of \mathbb{T}_u , denoted as

$$\Phi_m[l] = \frac{1}{(\sqrt{2\pi}\sigma)^r} \prod_{t=1}^r \exp\left(-\frac{(z_{l,t} + 2x_{l,t} + 2y_{l,t} - 2)^2}{2\sigma^2}\right), \quad (16)$$

where $1 \leq l \leq N$. \mathbb{C}_m is initialized with a set for the internal node, including m_l and m_r . For the leaf node, \mathbb{C}_m is initialized with an empty set ϕ .

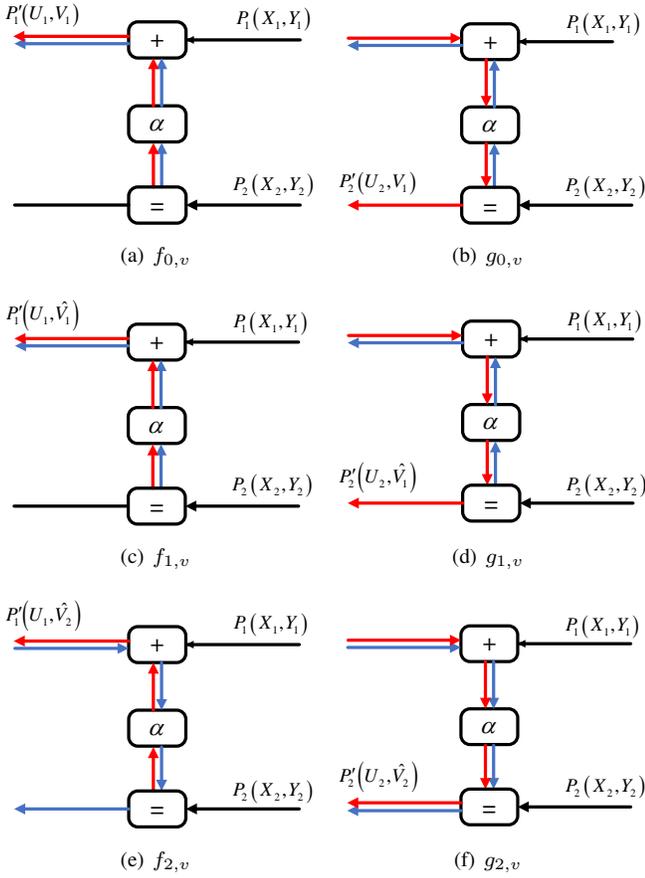


Fig. 4. Six different f and g calculations.

Step 2: The decoding process for stage I.

Update $\Phi_m, \beta_m, \mathbb{C}_m$ in \mathbb{T}_u . The node passing routine and update operation are relayed on the cardinality of \mathbb{C}_m , denoted as $|\mathbb{C}_m|$, and there are three cases.

If $|\mathbb{C}_m| = 2$, change to its left child node m_l and remove m_l from \mathbb{C}_m . Let $1 \leq l \leq \frac{\rho_m}{2}$ and update Φ_{m_l} as

$$\Phi_{m_l}[l] = f_{0,v} \left(\Phi_m[l], \Phi_m \left[l + \frac{\rho_m}{2} \right] \right). \quad (17)$$

If $|\mathbb{C}_m| = 1$, change to its right child node m_r and remove m_r from \mathbb{C}_m . Update Φ_{m_r} as

$$\Phi_{m_r}[l] = g_{0,v} \left(\Phi_m[l], \Phi_m \left[l + \frac{\rho_m}{2} \right], \beta_{m_l}[l] \right). \quad (18)$$

If $|\mathbb{C}_m| = 0$, change to its parent node p_m . There are two situations for this case. On one hand, when m is an internal node, let m_l denote the left child node of p_m , update β_{p_m} as

$$\beta_{p_m}[l] = \begin{cases} \beta_{m_l}[l] + \alpha \cdot \beta_m[l] & 1 \leq l \leq \rho_m \\ \beta_m[l - \rho_m] & \rho_m + 1 \leq l \leq 2\rho_m \end{cases}. \quad (19)$$

On the other hand, when m is a leaf node, make a decision on Φ_m and update β_m as

$$\beta_m = \arg \max_{u \in \mathbb{F}_q} \sum_{v \in \mathbb{F}_q} \Phi_m. \quad (20)$$

This stage is terminated when U_M has been decoded.

Step 3: Initialize θ_m for stage II.

Some of β_m in \mathbb{T}_u is initialized to θ_m in \mathbb{T}_v . Let $J = M$. For the leaf node, initialize θ_m as

$$\theta_m(n+1, J) = \beta_m(n+1, J). \quad (21)$$

Then for the non-leaf node, let $0 \leq k \leq n-1$, θ_m is updated from the leaf side to the root side recursively

$$\theta_m(n-k, J) = \beta_m(n-k, J), J = \left\lceil \frac{J}{2} \right\rceil. \quad (22)$$

Step 4: The decoding process for stage II.

This process is similar to step 2. The difference exists in update Φ_m with θ_m .

If $|\mathbb{C}_m| = 2$, update Φ_{m_l} as

$$\Phi_{m_l}[l] = f \left(\Phi_m[l], \Phi_m \left[l + \frac{\rho_m}{2} \right], \theta_{m_l}[l], \theta_{m_r}[l] \right). \quad (23)$$

If $|\mathbb{C}_m| = 1$, update Φ_{m_r} as

$$\Phi_{m_r}[l] = g \left(\Phi_m[l], \Phi_m \left[l + \frac{\rho_m}{2} \right], \beta_m[l], \theta_{m_l}[l], \theta_{m_r}[l] \right), \quad (24)$$

where $1 \leq l \leq \frac{\rho_m}{2}$. We choose different calculation types to calculate (23) and (24) according to the value of θ_{m_l} and θ_{m_r} . This stage is terminated when V_N is decoded successfully.

Step 5: Initialize θ_m for stage III.

Some of β_m in \mathbb{T}_v is initialized to θ_m in \mathbb{T}_u . Set $J = N$ and execute the recursive update in step 3, then the initialization for θ_m is accomplished.

Step 6: The decoding process for stage III.

This process is similar to step 4, and this stage is terminated when U_N is decoded successfully.

The algorithm is summarized in Algorithm 1, which yields $O(q \cdot N \cdot \log N)$ operations, a slight increase in contrast with the $O(N \cdot \log N)$ complexity of its binary counterpart.

IV. THEORETICAL ANALYSIS OF THE KERNEL

This section focuses on the effect of the kernel. Since the reliability is relatively low in stage I, the performance is mainly determined by the kernels of stages II and III. First, we consider α_u in stage III since α_v has no effect when all the auxiliary symbols V_1^N are already known. Then we optimize α_v in stage II, assuming α_u is fixed.

A. The Kernel of Stage III

The probability transform of stage III is equivalent to that of the single-user basic structure, as shown in Fig. 3, where U_1, U_2 and X_1, X_2 are the input and output, respectively.

Set $u_1 = 0$, the transition probability can be written as

$$P_i(z_i|x_i) = \frac{1}{(\sqrt{2\pi}\sigma)^r} \prod_{t=1}^r \exp \left(-\frac{(z_{i,t} + 2x_{i,t} - 1)^2}{2\sigma^2} \right), \quad (25)$$

where $1 \leq i \leq 2$, $\sigma^2 = N_0/2$. The coordinate channel transition probability of u_2 is

$$\begin{aligned} P(z_1^2, u_1|u_2) &= P_1(z_1|x_1) P_2(z_2|x_2) \\ &= \frac{1}{(\sqrt{2\pi}\sigma)^{2r}} \prod_{i=1}^2 \prod_{t=1}^r \exp \left(-\frac{(z_{i,t} + 2x_{i,t} - 1)^2}{2\sigma^2} \right). \end{aligned} \quad (26)$$

Algorithm 1 A Successive Cancellation decoding algorithm of NB-polar in the two-user MAC.

```

1: Input:  $\mathbb{T}_u, \mathbb{T}_v, n, M$ 
2: Initialize for stage I.
3: Initialize  $\Phi_m, \mathbb{C}_m$ 
4: Decoding process for stage I
5: for  $i = 1 : M$  do
6:   while  $d_m \neq n + 1$  do
7:     Update  $\Phi_m, \beta_m, \mathbb{C}_m$  in  $\mathbb{T}_u$ .
8:     Change to the next node.
9:   end while
10:   $\hat{u}_i = \arg \max_{u \in \mathbb{F}_q} \sum_{v \in \mathbb{F}_q} \Phi_m$ .
11:  Update  $\beta_m$  and change to the next node.
12: end for
13: Initialize for stage II
14: Initialize  $\Phi_m, \mathbb{C}_m, \theta_m$ 
15: Decoding process for stage II
16: for  $i = 1 : N$  do
17:   while  $d_m \neq n + 1$  do
18:     Update  $\Phi_m, \beta_m, \mathbb{C}_m$  in  $\mathbb{T}_u$ .
19:     Change to the next node.
20:   end while
21:    $\hat{v}_i = \arg \max_{v \in \mathbb{F}_q} \Phi_m$ .
22:   Update  $\beta_m$  and change to the next node.
23: end for
24: Initialize for stage III
25: Initialize  $\theta_m$ 
26: Decoding process for stage III
27: for  $i = M + 1 : N$  do
28:   while  $d_m \neq n + 1$  do
29:     Update  $\Phi_m, \beta_m, \mathbb{C}_m$  in  $\mathbb{T}_u$ .
30:     Change to the next node.
31:   end while
32:    $\hat{u}_i = \arg \max_{u \in \mathbb{F}_q} \Phi_m$ .
33:   Update  $\beta_m$  and change to the next node.
34: end for
35: Output:  $\hat{u}_1^N, \hat{v}_1^N$ .

```

According to the maximum likelihood detection, the decoded \hat{u}_2 is

$$\hat{u}_2 = \arg \max_{u_2 \in \mathbb{F}_q} P(z_1^2, u_1 | u_2), \quad (27)$$

Define $L_s = P(z_1^2, u_1 | \gamma^s)$, $\gamma^s \in \mathbb{F}_q$. Assuming $\tilde{u}_2 = \gamma^s$ is transmitted, the probability of a correct decision is given by

$$P_c = P[L_s > L_a, \text{ for all } \gamma^a \in \mathbb{F}_q \setminus \{\gamma^s\} | u_2 = \gamma^s]. \quad (28)$$

Assume the detection of γ^s is independent, (28) becomes

$$P_c = \prod_{\gamma^a \in \mathbb{F}_q, \gamma^a \neq \gamma^s} P[L_s > L_a | u_2 = \gamma^s]. \quad (29)$$

Let \bar{u}_2 denote γ^a , the term in (29) can be organized as

$$\sum_{i=1}^2 \sum_{t=1}^r (\bar{x}_{i,t}^2 - \tilde{x}_{i,t}^2 + \bar{x}_{i,t} z_{i,t} - \tilde{x}_{i,t} z_{i,t} + \tilde{x}_{i,t} - \bar{x}_{i,t}) > 0, \quad (30)$$

where \bar{x}_1, \bar{x}_2 and \tilde{x}_1, \tilde{x}_2 correspond to \bar{u}_2 and \tilde{u}_2 , respectively. Since $z_{i,t} \sim \mathcal{N}(-2\tilde{x}_{i,t} + 1, \sigma^2)$, the left side of (30) is also a Gaussian variable $Q \sim \mathcal{N}(\mu_q, \sigma_q^2)$, where

$$\mu_q = \sum_{i=1}^2 \sum_{t=1}^r (\tilde{x}_{i,t} - \bar{x}_{i,t})^2, \sigma_q^2 = \sum_{i=1}^2 \sum_{t=1}^r (\tilde{x}_{i,t}^2 + \bar{x}_{i,t}^2) \sigma^2, \quad (31)$$

and the probability is calculated by integrating the probability density function (PDF) of Q . The average BER is calculated by going through all u_2 . Then we choose the kernel factor α_u that can provide the best BER performance.

B. The Kernel of Stage II

Assuming that α_u is fixed, then we select the optimal α_v . Set $u_1 = 0$ and $v_1 = 0$, the transition probability is

$$P_i(z_i | x_i, y_i) = \frac{1}{(\sqrt{2\pi}\sigma)^r} \prod_{t=1}^r \exp\left(-\frac{(z_{i,t} + 2x_{i,t} + 2y_{i,t} - 2)^2}{2\sigma^2}\right), \quad (32)$$

Define \tilde{x}_1, \tilde{x}_2 by $\{\tilde{x}_1, \tilde{x}_2 \in \mathbb{F}_q | \tilde{x}_1 + \alpha_u \cdot \tilde{x}_2 = u_1\}$, the split channel transition probability of u_1, v_2 is given by

$$P(z_1^2, v_1 | u_1, v_2) = \sum_{\tilde{x}_1, \tilde{x}_2} \frac{1}{(\sqrt{2\pi}\sigma)^{2r}} \prod_{i=1}^2 \prod_{t=1}^r \exp\left(-\frac{(z_{i,t} + 2\tilde{x}_{i,t} + 2y_{i,t} - 2)^2}{2\sigma^2}\right). \quad (33)$$

Define $L'_s = P(z_1^2, v_1 | u_1, \gamma^s)$. Assuming $\tilde{v}_2 = \gamma^s$ is transmitted, the probability of a correct decision is

$$P_c = \prod_{\gamma^a \in \mathbb{F}_q, \gamma^a \neq \gamma^s} P[L'_s > L'_a | v_2 = \gamma^s]. \quad (34)$$

It is hard to derive the term in (34) directly. Thus (33) is simplified by omitting the relatively small items. Assuming \tilde{u}_2 is transmitted, and \tilde{x}_1, \tilde{x}_2 correspond to \tilde{u}_2 . Let \bar{v}_2 denote γ^a . Define $\tilde{w}_i = \tilde{x}_i + \tilde{y}_i$ and $\bar{w}_i = \tilde{x}_i + \bar{y}_i$, $1 \leq i \leq 2$, besides

$$\tilde{x}_1^2 = \arg \min_{\tilde{x}_1, \tilde{x}_2 \in \mathbb{F}_q} \sum_{i=1}^2 \|\tilde{w}_i - \bar{w}_i\|^2, \quad (35)$$

where \bar{y}_1, \bar{y}_2 and \tilde{y}_1, \tilde{y}_2 correspond to \bar{v}_2 and \tilde{v}_2 , respectively. Then (33) is approximated by

$$P(z_1^2, v_1 | u_1, \bar{v}_2) \approx \frac{1}{(\sqrt{2\pi}\sigma)^{2r}} \prod_{i=1}^2 \prod_{t=1}^r \exp\left(-\frac{(z_{i,t} + 2\tilde{x}_{i,t} + 2y_{i,t} - 2)^2}{2\sigma^2}\right). \quad (36)$$

The situation degraded to the single-user case, and the average BER is calculated by going through all u_2 and v_2 . Then the kernel factor α_v that provides the best BER is chosen.

V. SIMULATION RESULTS

In this section, Monte Carlo method is used to simulate the BLER performances of the proposed system. In this work, we take the GF(16) as an example to do analysis, and the theoretical BERs of different kernels are shown in Figs. 5 and 6. According to Fig. 5, $\alpha_u = 5, 10$ is the optimal choice for the single-user case in GF(16). Thus let $\alpha_u = 5$ be the kernel factor for user 1. According to Fig. 6, $\alpha_v = 3$ is the optimal choice in the two-user case when $\alpha_u = 5$. Thus let $\alpha_v = 3$ be the kernel factor for user 2.

Fig. 7 compares the BLER performances between the proposed non-binary and binary systems in the two-user MAC

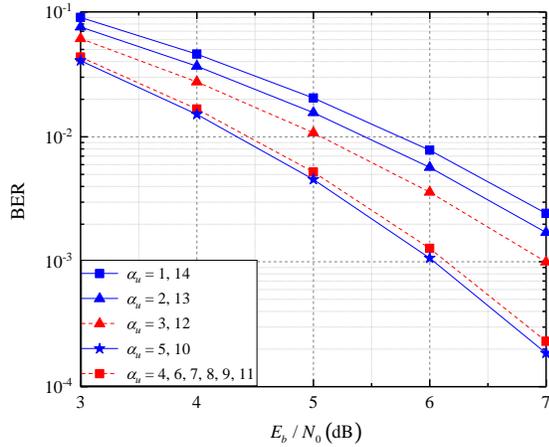


Fig. 5. Different kernels' impact on the single-user basic structure.

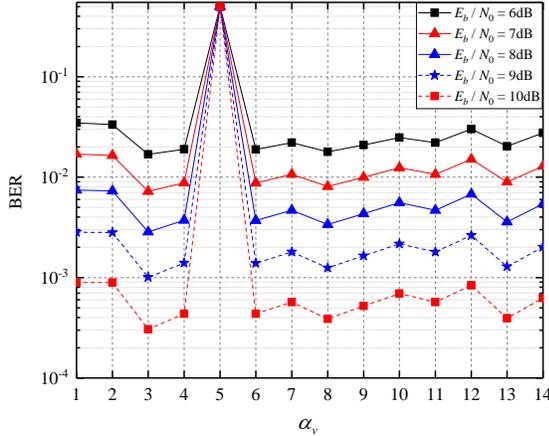


Fig. 6. Different kernels' impact on the combined basic structure.

with different code lengths, where the Monte Carlo construction is used to pick the transmission channels. The decoding order parameter M is set as $N/4$, with $R = 0.5$. First of all, the BLER performance comparisons between two non-binary cases are considered, where $(\alpha_u, \alpha_v) = (5, 3)$ and $(\alpha_u, \alpha_v) = (5, 5)$, corresponding to the best and the worst case in GF(16), given by Fig. 6. When $\text{BLER} = 1 \times 10^{-3}$, the required E_b/N_0 of $N = 64, 128$ are respectively 6.3dB and 5.4dB for (5, 3) kernel case. Obviously, the BLER performance improves with the increase of N . Moreover, it is found that the BLER of (5, 3) case is better than that of (5, 5) case, e.g., there is 0.4dB gain when $\text{BLER} = 1 \times 10^{-3}$ and $N = 128$, which is in line with the theoretical analysis. Secondly, it is found that the proposed (5, 3) non-binary polar system provides a much lower BLER performance than that of the binary polar system, e.g., when $\text{BLER} = 1 \times 10^{-3}$ and $N = 64, 128$, the E_b/N_0 of (5, 3) case has 0.5dB and 0.8dB gain, respectively. In summary, the non-binary polar system with the optimal kernel achieves relatively superior performance compared to the classical binary system.

VI. CONCLUSION

This paper proposes a non-binary polar coding scheme in the two-user MAC and the corresponding SC-decoding

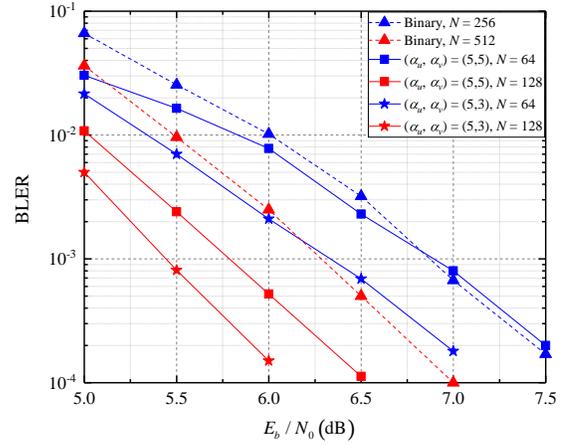


Fig. 7. BLER performance between non-binary and binary polar system.

algorithm. The choice of the kernel factors is discussed in detail. Simulation results show that there is a vast improvement between the worst and the best kernel choice. Moreover, the non-binary polar codes in the two-user MAC achieve much better BLER performances than the classical binary codes.

REFERENCES

- [1] E. Arıkan, "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, July 2009.
- [2] E. Şaşıođlu, E. Telatar and E. M. Yeh, "Polar Codes for the Two-User Multiple-Access Channel," *IEEE Trans. Inf. Theory*, vol. 59, no. 10, pp. 6583-6592, Oct. 2013.
- [3] E. Abbe and E. Telatar, "Polar Codes for the m -User Multiple Access Channel," *IEEE Trans. Inf. Theory*, vol. 58, no. 8, pp. 5437-5448, Aug. 2012.
- [4] H. Mahdaviyar, M. El-Khany, J. Lee and I. Kang, "Techniques for polar coding over multiple access channels," in *Proc. 2014 48th Annu. Conf. Inf. Sci. and Sys. (CISS)*, 2014, pp. 1-6.
- [5] S. Önyar, "Successive cancellation decoding of polar codes for the two-user binary-input MAC," in *Proc. 2013 IEEE Int. Symp. Inf. Theory*, 2013, pp. 1122-1126.
- [6] E. Şaşıođlu, E. Telatar and E. Arıkan, "Polarization for arbitrary discrete memoryless channels," in *Proc. 2009 IEEE Inf. Theory Workshop*, 2009, pp. 144-148.
- [7] R. Mori and T. Tanaka, "Channel polarization on q -ary discrete memoryless channels by arbitrary kernels," in *Proc. 2010 IEEE Int. Symp. Inf. Theory*, 2010, pp. 894-898.
- [8] E. Şaşıođlu, "Polar codes for discrete alphabets," in *Proc. 2012 IEEE Int. Symp. Inf. Theory Proceedings*, 2012, pp. 2137-2141.
- [9] W. Park and A. Barg, "Polar Codes for Q -Ary Channels, $q = 2^r$," *IEEE Trans. Inf. Theory*, vol. 59, no. 2, pp. 955-969, Feb. 2013.
- [10] P. Yuan and F. Steiner, "Construction and Decoding Algorithms for Polar Codes based on 2×2 Non-Binary Kernels," in *Proc. 2018 IEEE 10th Int. Symp. Turbo Codes & Iter. Inf. Process. (ISTC)*, 2018, pp. 1-5.
- [11] R. Nasser and E. Telatar, "Polar Codes for Arbitrary DMCs and Arbitrary MACs," *IEEE Trans. Inf. Theory*, vol. 62, no. 6, pp. 2917-2936, June 2016.
- [12] R. Nasser and J. M. Renes, "Polar Codes for Arbitrary Classical-Quantum Channels and Arbitrary cq-MACs," *IEEE Trans. Inf. Theory*, vol. 64, no. 11, pp. 7424-7442, Nov. 2018.
- [13] I. Tal, A. Sharov and A. Vardy, "Constructing polar codes for non-binary alphabets and MACs," in *Proc. 2012 IEEE Int. Symp. Inf. Theory Proceedings*, 2012, pp. 2132-2136.
- [14] U. Pereg and I. Tal, "Channel Upgradation for Non-Binary Input Alphabets and MACs," *IEEE Trans. Inf. Theory*, vol. 63, no. 3, pp. 1410-1424, March 2017.
- [15] A. Alamdar-Yazdi and F. R. Kschischang, "A Simplified Successive-Cancellation Decoder for Polar Codes," *IEEE Commun. Lett.*, vol. 15, no. 12, pp. 1378-1380, December 2011.