Successive Cancellation Decoding of Polar Codes for the Two-User Binary-Input MAC

Saygun Önay
Bilkent University, Ankara, Turkey
Email: saygun@ee.bilkent.edu.tr

***Abstract*—This paper describes a successive cancellation decoder of polar codes for the two-user binary-input multi-access channel that achieves the full admissible rate region. The polar code for the channel is generated from monotone chain rule expansions of mutual information terms.**

I. Introduction

Polarization technique defined in [1] has originally been introduced to develop capacity achieving channel codes. The class of linear block codes, called polar codes, generated by the technique are the first codes with low encoding and decoding complexity that are provably capacity achieving on any discrete memoryless channel (DMC). The essence of the technique is to generate a set of *extremal* channels from repeated uses of a single-user channel. By extremal, it is meant that almost every channel in the set is either almost perfect or almost useless. Although introduced in channel coding context, it has been shown that polarization technique can be used for optimal single-user source coding in [2], [3].

In [4], polarization has been extended to two-user binary-input multi-access channels (MACs). The authors followed a "joint polarization" approach and reached an interesting result stating that there are five types of extremal channels in that case. However, it has also been shown that polar coding cannot achieve the full capacity region of the MAC with the approach used there. Arıkan showed [5] in source coding context that more general classes of polar codes can be defined for multi-user problems by different chain rule expansions. He investigated "monotone" chain rules and proved that at least a specific class of expansions exists that achieves every point on the dominant face of the Slepian-Wolf (SW) achievable rate region [6] with arbitrarily small precision. The results presented in that work are also directly applicable to the dual MAC problem as we pursue in this paper.

Specifically in this paper, we describe a successive cancellation (SC) decoder of the class of polar codes for two-user binary-input multi-access channels that are generated by monotone chain rules introduced in [5]. Furthermore, we implement a *list* decoder to improve the performance and present simulation results for both type of decoders.

II. Preliminaries

We use the notation of [1]. We denote generic variables by upper case letters such as X and their realizations with corresponding lower case letters such as x. We write x^N to denote a vector $(x_1, \ldots, x_N)$ and x_i^j to denote the sub-vector $(x_i, \ldots, x_j)$ for any $1 \leq i \leq j \leq N$. If $j < i$, x_i^j is the null vector.

Let $W\colon \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ be a two-user multiple-access channel with binary input alphabets $\mathcal{X} = \mathcal{Y} = \{0, 1\}$. $\mathcal{Z}$ is an arbitrary discrete output alphabet. The transition probability of the channel is denoted with $W(z|x, y)$, which is the conditional probability of the channel output $z \in \mathcal{Z}$ for each input pair $(x, y) \in \mathcal{X} \times \mathcal{Y}$. The capacity region of such a MAC is the convex hull of all (R_x, R_y) satisfying

$$\begin{aligned} R_x &< I(Z; X|Y) \\ R_y &< I(Z; Y|X) \\ R_x + R_y &< I(Z; X, Y) \end{aligned}$$

for some product distribution $p_X(x)p_Y(y)$ on $\mathcal{X} \times \mathcal{Y}$.

In this paper, rather than the convex hull of all possible rate pairs satisfying above inequalities that are induced by all possible input distributions, we are interested in the *achievable* rate region induced by a specific distribution, namely the uniform distribution on both inputs, i.e. $p_X(x) = p_Y(y) = \frac{1}{2}$. We identify this rate region with $\mathcal{R}(W)$.

Suppose an input block of size $N = 2^n$ with $n \geq 1$. We define W^N as the vector channel corresponding to N independent uses of W. Let $X^N \in \mathcal{X}^N$ and $Y^N \in \mathcal{Y}^N$ denote the inputs of vector channel W^N corresponding to user 1 and 2, respectively. Then, we can write $W^N\colon \mathcal{X}^N \times \mathcal{Y}^N \to \mathcal{Z}^N$ as $W^N(z^N|x^N, y^N) = \prod_{i=1}^N W(z_i|x_i, y_i)$.

Let $U^N \in \mathcal{X}^N$ and $V^N \in \mathcal{Y}^N$ denote the user 1 and user 2 data, respectively. Polar coding on such a channel is performed by transforming each of these vectors as follows before applying to channel.

$$X^N = U^N G_N, \qquad Y^N = V^N G_N \tag{1}$$

where $G_N = \left[\begin{smallmatrix}1 & 0\\ 1 & 1\end{smallmatrix}\right]^{\otimes n} B_N$ is the polar transformation matrix. "$\otimes n$" is the nth Kronecker power and B_N is the "bit-reversal" matrix defined in [1]. The above transformation operations produce a combined vector channel $W_N\colon \mathcal{X}^N \times \mathcal{Y}^N \to \mathcal{Z}^N$ with

$$W_N(z^N|u^N, v^N) = W^N(z^N|u^N G_N, v^N G_N). \tag{2}$$

A. Chain rules and rates

Since G_N is a one-to-one mapping, we have

$$I(Z^N; U^N, V^N) = I(Z^N; X^N, Y^N) = NI(Z; X, Y), \tag{3}$$

which states that the total mutual information is conserved. There are various ways one can expand $I(Z^N; U^N, V^N)$. We use the monotone chain rule expansions defined in [5] and present this expansion here for completeness. The expansion is of the form

$$I(Z^N; U^N, V^N) = \sum_{k=1}^{2N} I(Z^N; S_k | S^{k-1}) \tag{4}$$

where $S^{2N} = (S_1, \ldots, S_{2N})$ is a permutation of $U^N V^N$ such that it preserves the relative order of the elements of both U^N and V^N. This class of chain rule expansion can be visualized on a two-dimensional diagram as presented in [5]. Different expansions can be represented as paths on this "chain rule diagram" which start at top-left node $\emptyset$ and terminate at bottom-right node $U^N V^N$. Each move from node to node along the path must be either to the right or down. This rule guarantees the monotony of the expansion on both U^N and V^N. These paths on the chain rule diagram can also be identified by a string $b^{2N} = b_1 b_2 \ldots b_{2N}$ where b_k is 0 if the kth move along the path is in the horizontal direction and 1 otherwise. And $(S_1, \ldots, S_{2N})$ denote the *edge variables* along a given path b^{2N}. Each edge on the diagram corresponds to an *incremental* mutual information gained along the path which is given as follows for an edge terminating at a node denoted by $U^i V^j$.

$$I(Z^N; S_k | S^{k-1}) = \begin{cases} I(Z^N; U_i | U^{i-1} V^j) & \text{if } b_k = 0 \\ I(Z^N; V_j | U^i V^{j-1}) & \text{if } b_k = 1 \end{cases} \tag{5}$$

for $k = i + j$, $0 \le i, j \le N$ and $1 \le k \le 2N$. Then, the partial mutual information accumulated upto node $U^i V^j$ on such a path is $\sum_{k'=1}^{k} I(Z^N; S_{k'} | S^{k'-1}) = I(Z^N; U^i, V^j)$.

We define rates R_u and R_v for any given path b^{2N} with edge variables S^{2N} as follows.

$$R_u = \frac{1}{N} \sum_{k: b_k = 0} I(Z^N; S_k | S^{k-1}) \tag{6}$$

and

$$R_v = \frac{1}{N} \sum_{k: b_k = 1} I(Z^N; S_k | S^{k-1}). \tag{7}$$

R_u and R_v are the sum of the conditional mutual information terms normalized by N on the horizontal and vertical edges along the given path, respectively. Then, clearly for any path b^{2N} these rates satisfy

$$R_u \le \frac{1}{N} I(Z^N; U^N | V^N) = I(Z; X|Y),$$
$$R_v \le \frac{1}{N} I(Z^N; V^N | U^N) = I(Z; Y|X),$$
$$R_u + R_v = \frac{1}{N} I(Z^N; U^N, V^N) = I(Z; X, Y).$$

Let (R_u^1, R_v^1) and (R_u^2, R_v^2) be two rate pairs that satisfy the first and second inequality with equality, respectively. Then they correspond to operating on one of the corner points of the dominant face of $\mathcal{R}(W)$. The rate pairs (R_u^1, R_v^1) and (R_u^2, R_v^2) are achieved with paths $1^N 0^N$ and $0^N 1^N$, respectively.

B. Continuity of rates and polarization

It should be obvious by now that the MAC problem considered here is very much related to multi-terminal source coding problem considered in [5]. The results of [5] on continuity of rates and polarization are directly applicable here with a small modification on the problem stated in [5]. Therefore, we do not present the full treatment here, again; however, we sketch the general idea how this relation may be observed. We generalize the setting of source coding problem in [5] by adding an extrinsic "side-information" $Z \in \mathcal{Z}$ on source pair (X, Y), where $\mathcal{Z}$ is an arbitrary discrete alphabet. That is, we assume (Z^N, X^N, Y^N) be independent samples from a source $(Z, X, Y) \sim p_{XY}(x, y) W(z|x, y)$. Then, in effect, all of the entropy terms of (X, Y) and (U, V) in [5] are replaced with the ones conditioned on this new information source Z.

Now, the MAC problem considered here is related to a special case of the generalized SW problem stated above where $p_{XY}(x, y) = p_X(x) p_Y(y) = \frac{1}{4}$, i.e. X and Y are independent and uniformly distributed on $\{0, 1\}$. However, note that X and Y are not independent conditional on Z. Then, incremental mutual information terms of (5) relate to incremental entropy terms in [5] as $I(Z^N; S_k | S^{k-1}) = 1 - H(S_k | S^{k-1}, Z^N)$ which implies that they also polarize to 0 or 1. Similarly, the channel rates (R_x, R_y) relate to source rates (R_x^s, R_y^s) as $R_x = 1 - R_x^s$, $R_y = 1 - R_y^s$. Therefore, it is implied that arbitrary points on the dominant face of $\mathcal{R}(W)$ can be achieved with such chain rule expansions.

C. Coordinate channels and polar coding

Even though the underlying channel is a two-user MAC, the class of chain rule expansions given in (5) gives rise to four types of extremal channels which are *single-user*. This is in contrast to the expansions considered in [4], where the extremal channels themselves are also MACs and they are of five types.

For a given path b^{2N} with edge variables $(s_1, \ldots, s_{2N})$, we write the coordinate channels $W_N^{(b_k, i, j)} : \mathcal{X} \to Z^N \times X^{k-1}$, $0 \le i, j \le N$, $1 \le k = i + j \le 2N$ as

$$\begin{aligned} &W_N^{(b_k, i, j)}(z^N, s^{k-1} | s_k) \\ &\qquad = \begin{cases} W_N^{(0, i, j)}(z^N, u^{i-1}, v^j | u_i) & \text{if } b_k = 0 \\ W_N^{(1, i, j)}(z^N, u^i, v^{j-1} | v_j) & \text{if } b_k = 1 \end{cases} \end{aligned} \tag{8}$$

Although the coordinate channels are single-user binary-input channels, the input may be from user 1 or 2 depending on the value of b_k. The capacity terms of these channels, given by (5), polarize to 0 or 1 as N goes to infinity. Hence, four different types of extremal channels emerge. These channels can be interpreted as the effective channel seen by the successive cancellation decoder (provided that s^{k-1} was decoded correctly) at decoding step k.

Similar to single-user case, we define the polar coding for two-user MAC as accessing individual coordinate channels and sending information only through the ones with capacity terms near 1. We identify these codes with parameter vector $(N, b^{2N}, (K_u, K_v), (\mathcal{A}_u, \mathcal{A}_v), (u_{\mathcal{A}_u^c}, v_{\mathcal{A}_v^c}))$. K_u and K_v

denote the information block size of user 1 and 2, respectively. $\mathcal{A}_u$ and $\mathcal{A}_v$ denote the information index set of user 1 and 2, respectively. $u_{\mathcal{A}_u^c}$ and $v_{\mathcal{A}_v^c}$ denote the frozen bits of user 1 and 2, respectively. b^{2N} denotes the chain rule expansion, which also dictates the order of decoding of user bits.

At each step of decoding the decoder decodes a single bit of either user 1 or 2 in increasing order of indices. The decision functions $h_{(b_k,i,j)} : \mathcal{Z}^N \times \mathcal{X}^{k-1} \to \mathcal{X}$, $k = i + j$ are defined as

$$h_{(b_k,i,j)}(z^N, s^{k-1}) = \begin{cases} 0, & \frac{W_N^{(b_k,i,j)}(z^N, s^{k-1}|0)}{W_N^{(b_k,i,j)}(z^N, s^{k-1}|1)} \geq 1 \\ 1, & \text{otherwise} \end{cases} \quad (9)$$

and valid for $1 \leq k \leq 2N$ and either $i \in \mathcal{A}_u$ if $b_k = 0$ or $j \in \mathcal{A}_v$ if $b_k = 1$. Then, the decoder generates its decisions as

$$\hat{u}_i = \begin{cases} u_i, & \text{if } i \in \mathcal{A}_u^c \\ h_{(0,i,j)}(z^N, s^{k-1}), & \text{otherwise} \end{cases} \quad (10)$$

if $b_k = 0$ or

$$\hat{v}_j = \begin{cases} v_j, & \text{if } j \in \mathcal{A}_v^c \\ h_{(1,i,j)}(z^N, s^{k-1}), & \text{otherwise} \end{cases} \quad (11)$$

if $b_k = 1$.

III. Recursive Channel Transformations and Successive Cancellation Decoder

Suppose a given path b^{2N} with edge variables $(s_1, \ldots, s_{2N})$, again. Instead of calculating single user coordinate channel probabilities directly we calculate transition probabilities for channels $W_N^{(i,j)} : \mathcal{X}^2 \to Z^N \times X^{k-2}$, $1 \leq k = i + j \leq 2N$ defined by

$$W_N^{(i,j)}(z^N, u^{i-1}, v^{j-1}|u_i, v_j) \triangleq \sum_{u_{i+1}^N, v_{j+1}^N} \left(\frac{1}{2^{N-1}}\right)^2 W_N(z^N|u^N, v^N) \quad (12)$$

Although these channels are two-user, we never decide on the values of both u_i and v_j at a single step. We obtain coordinate channel probabilities of (8) from (12) as follows.

$$W_N^{(0,i,j)}(z^N, \hat{u}^{i-1}, \hat{v}^j|u_i) = \begin{cases} \sum_{v_1} \frac{1}{2} W_N^{(i,j)}(z^N, \hat{u}^{i-1}|u_i, v_1) & \text{if } j = 0 \\ \frac{1}{2} W_N^{(i,j)}(z^N, \hat{u}^{i-1}, \hat{v}^{j-1}|u_i, \hat{v}_j) & \text{if } j > 0 \end{cases} \quad (13)$$

and

$$W_N^{(1,i,j)}(z^N, \hat{u}^i, \hat{v}^{j-1}|v_j) = \begin{cases} \sum_{u_1} \frac{1}{2} W_N^{(i,j)}(z^N, \hat{v}^{j-1}|u_1, v_j) & \text{if } i = 0 \\ \frac{1}{2} W_N^{(i,j)}(z^N, \hat{u}^{i-1}, \hat{v}^{j-1}|\hat{u}_i, v_j) & \text{if } i > 0 \end{cases} \quad (14)$$

In (13) and (14), we explicitly denoted the variables whose values have already been decided with a "ˆ" over them.

Similar to single-user case, the block-wise channel combining and splitting operations of (2) and (12) can be broken into single-step channel transformations. This is crucial to implementing a recursive low complexity decoding. First, we write a couple of definitions to simplify the notation of recursion equations.

Definition 1. *Let $\bar{u}_i \triangleq u_{2i-1} \oplus u_{2i}$ and $\bar{\bar{u}}_i \triangleq u_{2i}$. Then it also follows that $\bar{u}^i = u_o^{2i} \oplus u_e^{2i}$ and $\bar{\bar{u}}^i = u_e^{2i}$. We define $\bar{v}_j$ and $\bar{\bar{v}}_j$ similarly.*

Definition 2. *For any $n \geq 1, N = 2^n, 1 \leq i, j \leq N$,*

$$A_N^{(i,j)}(z^{2N}, u^{2i}, v^{2j}) \triangleq W_N^{(i,j)}(z_1^N, \bar{u}^{i-1}, \bar{v}^{j-1}|\bar{u}_i, \bar{v}_j) \cdot W_N^{(i,j)}(z_{N+1}^{2N}, \bar{\bar{u}}^{i-1}, \bar{\bar{v}}^{j-1}|\bar{\bar{u}}_i, \bar{\bar{v}}_j). \quad (15)$$

Now, we can write the recursion equations which are of four types.

Proposition 1. *For any $n \geq 0$, $N = 2^n$, $1 \leq i, j \leq N$*

$$W_{2N}^{(2i-1,2j-1)}(z^{2N}, u^{2i-2}, v^{2j-2}|u_{2i-1}, v_{2j-1}) = \sum_{u_{2i}, v_{2j}} \frac{1}{2^2} A_N^{(i,j)}(z^{2N}, u^{2i}, v^{2j}) \quad (16)$$

$$W_{2N}^{(2i,2j-1)}(z^{2N}, u^{2i-1}, v^{2j-2}|u_{2i}, v_{2j-1}) = \sum_{v_{2j}} \frac{1}{2^2} A_N^{(i,j)}(z^{2N}, u^{2i}, v^{2j}) \quad (17)$$

$$W_{2N}^{(2i-1,2j)}(z^{2N}, u^{2i-2}, v^{2j-1}|u_{2i-1}, v_{2j}) = \sum_{u_{2i}} \frac{1}{2^2} A_N^{(i,j)}(z^{2N}, u^{2i}, v^{2j}) \quad (18)$$

$$W_{2N}^{(2i,2j)}(z_1^{2N}, u_1^{2i-1}, v_1^{2j-1}|u_{2i}, v_{2j}) = \frac{1}{2^2} A_N^{(i,j)}(z^{2N}, u^{2i}, v^{2j}). \quad (19)$$

We omit the proof which is very similar to single-user case given in [1] Proposition 3. The four types of recursions correspond to indices of u and v being odd and even.

The main loop of the decoding algorithm is presented in Algorithm 1 as pseudo-code. The decoder runs for $2N$ steps, deciding on the bit value of u_i or v_j depending on the value of b_k being 0 or 1, respectively. Thus, at each step either i or j is incremented. To make the decision, decoder calculates one of the *a posteriori* probabilities (APP) $W_N^{(0,i,j)}(z^N, u^{i-1}, v^j|u_i)$ or $W_N^{(1,i,j)}(z^N, u^i, v^{j-1}|v_j)$. Then, it decides on the value of the information bit as given in (10) or (11). The APP is calculated from $W_N^{(i,j)}(z^N, u^{i-1}, v^{j-1}|u_i, v_j)$ using (13) or (14). The calculations of $W_N^{(i,j)}(z^N, u^{i-1}, v^{j-1}|u_i, v_j)$ are performed recursively using one of the four equations (16),(17),(18),(19), depending on the values of i and j being odd or even. At each step of recursion the problem is split into two half size problems, just like in single-user SC decoder. Recursions stop when the problem size is reduced to 1 at which point the channel probability $W(z|u, v)$ is returned.

Algorithm 1: Description of the SC MAC decoder

input : Received vector z^N, path vector b^{2N}
output: Decoded user bits $(\hat{u}^N, \hat{v}^N)$

// Initialization
$i \leftarrow 0,\ j \leftarrow 0$
// Main Loop
for $k = 1, \ldots, 2N$ **do**
  if $b_k = 0$ **then**
    // Horizontal step
    $i \leftarrow i + 1$
    calculate $P_i^u[u_i] \leftarrow W_N^{(0,i,j)}(z^N, \hat{u}^{i-1}, \hat{v}^j | u_i)$ from (13)
    if u_i is frozen **then**
      set $\hat{u}_i$ to the frozen value
    else
      if $P_i^u[1] > P_i^u[0]$ **then**
        set $\hat{u}_i \leftarrow 1$
      else
        set $\hat{u}_i \leftarrow 0$
  else
    // Vertical step
    $j \leftarrow j + 1$
    calculate $P_j^v[v_j] \leftarrow W_N^{(1,i,j)}(z^N, \hat{u}^i, \hat{v}^{j-1} | v_j)$ from (14)
    if v_j is frozen **then**
      set $\hat{v}_j$ to the frozen value
    else
      if $P_j^v[1] > P_j^v[0]$ **then**
        set $\hat{v}_j \leftarrow 1$
      else
        set $\hat{v}_j \leftarrow 0$

return $(\hat{u}^N, \hat{v}^N)$

Note that, a special case occurs in calculating the probabilities $W_N^{(0,i,j)}$ and $W_N^{(1,i,j)}$ that is indicated by the first lines of cases in (13) and (14). For example, let $b_1 = 0$, then at step 1 the decoder must calculate the probability $W_N^{(0,1,0)}(z^N|u_1)$. First, $W_N^{(1,1)}(z^N|u_1, v_1)$ is calculated recursively. Then, the probability we are interested is calculated as $W_N^{(0,1,0)}(z^N|u_1) = \sum_{v_1} \frac{1}{2} W_N^{(1,1)}(z^N|u_1, v_1)$, using the first line of case statement in (13). This equation is used until the first time a decision is to be made on v_1, i.e. the first time b_k is 1. At that point the APP calculation is of the form $W_N^{(1,i,1)}(z^N, \hat{u}^i|v_1) = \frac{1}{2} W_N^{(i,1)}(z^N, \hat{u}^{i-1}|\hat{u}_i, v_1)$. From that point on, only the second line of case statement in (13) is invoked.

The running time complexity order of the decoder is the same as the single-user decoder which is $O(N \log N)$. The space complexity can be made as low as $O(N)$ using the space efficient implementation introduced in [7].

IV. Simulations

Similar to single-user case, *list* decoding is also possible for MAC decoder. Here, we present results for both which are based on efficient implementation of [7]. The list decoder improves the performance considerably. We present the performance of the decoders on two different types of MACs.

The first one is the binary erasure MAC (BE-MAC). The channel output is given as $Z = X + Y$ which is of a ternary alphabet, $Z \in \mathcal{Z} = \{0, 1, 2\}$. The capacity region of this channel is well known [8, Sec. 15.3] and shown in Fig.1. The figure shows results for three different code classes labelled by A, B and C which target three different rate pairs $(0.75, 0.75)$, $(0.625, 0.875)$ and $(0.5, 1)$ on the dominant face of the MAC region, respectively. The results are shown for four different block sizes from each code class yielding 12 points on the figure. We adjust the rates of the codes on straight lines yielding operating rate pairs $(R_u^A, R_v^A) = \rho_A \cdot (0.75, 0.75)$, $(R_u^B, R_v^B) = \rho_B \cdot (0.625, 0.875)$ and $(R_u^C, R_v^C) = \rho_C \cdot (0.5, 1)$ with $0 \leq \rho_A, \rho_B, \rho_C \leq 1$ for code classes A, B and C, respectively. And we mark the points where block error rate (BLER) reaches 10^{-4}. The shown results are for list size (L) of 32. The results show that the performance of code class B is the best while A comes second and C is the worst. Fig.2 shows BLER w.r.t. sum rate for code class B, comparing the performances of list sizes 1 (no list) and 32.

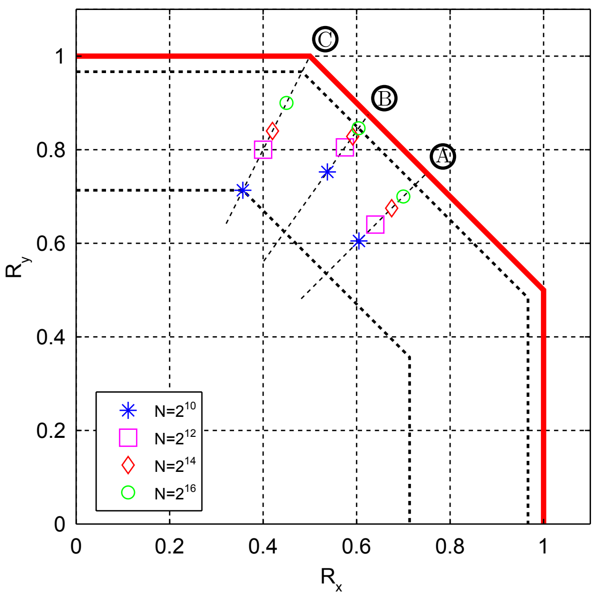

Fig. 1. Operating rates for BE-MAC at BLER=10^{-4}.

Next we consider a class of MACs defined as follows. The output of the channel is quaternary but decomposed into two binary components: $Z = (Z_x, Z_y) \in \mathcal{Z} = \{0, 1\} \times \{0, 1\}$. We consider an "additive" channel $W : \mathcal{X}^2 \rightarrow \mathcal{Z}$ with $Z_x = X \oplus E_x$ and $Z_y = Y \oplus E_y$, where $E_x, E_y \in \mathcal{X}$ and $(E_x, E_y) \sim p_{E_x,E_y}(e_x, e_y)$ is an arbitrary distribution on $\mathcal{Z}$ and independent of X and Y. Thus, $E = (E_x, E_y)$ is an additive correlated *error* vector on the inputs of the MAC. We name this channel as additive binary noise MAC (ABN-MAC). In the simulations, we use the distribution

$$p_{E_x,E_y} = \begin{bmatrix} 0.1286 & 0.0175 \\ 0.0175 & 0.8364 \end{bmatrix}.$$

This distribution results in a MAC with an achievable rate region $\{(R_x, R_y) : R_x \leq 0.4,\ R_y \leq 0.4,\ R_x + R_y \leq 1.2\}$ which is shown in Fig.3 along with the simulation results marked. Code class B performs slightly better while the performances of A and C are almost the same. Fig.4 shows BLER w.r.t. sum rate for code class B.

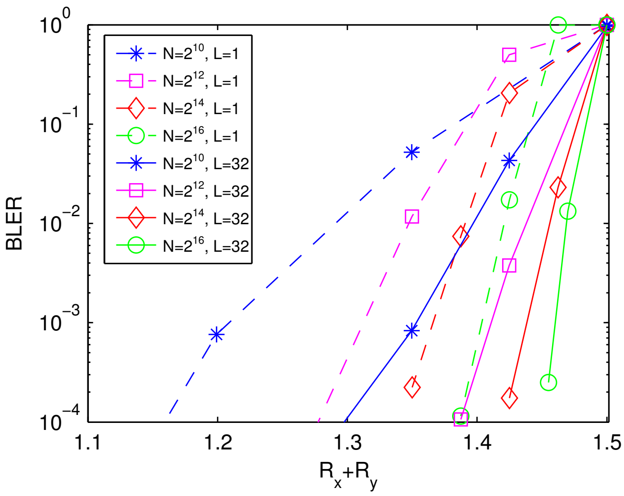

Fig. 2. BLER performance for BE-MAC w.r.t. sum rate.

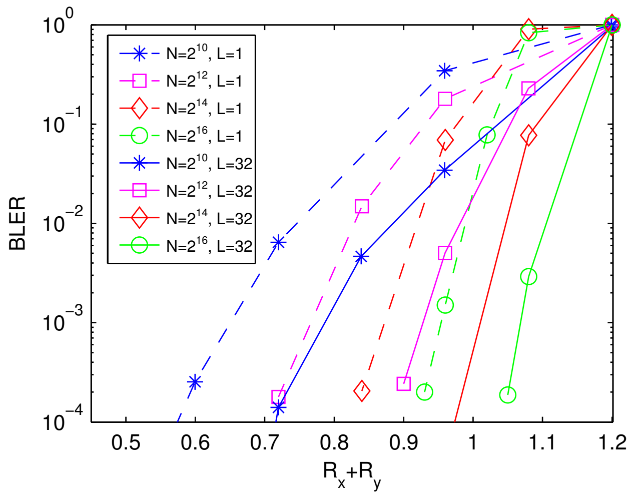

Fig. 4. BLER performance for ABN-MAC w.r.t. sum rate.

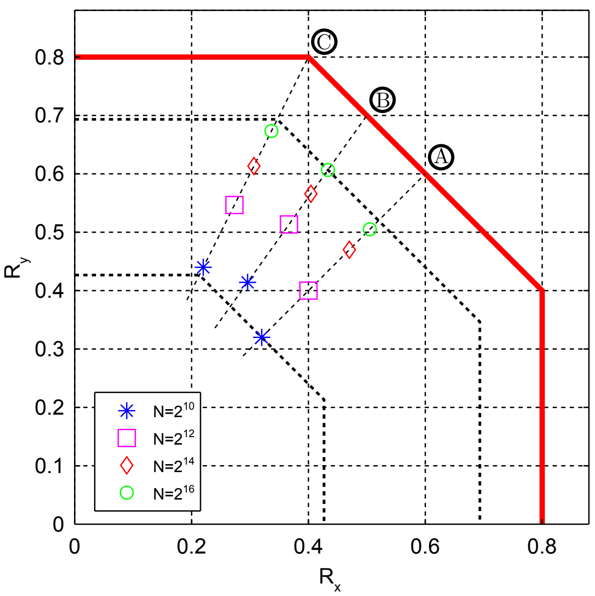

Fig. 3. Operating rates for ABN-MAC at BLER=10^{-4}.

We designed the polar codes by Monte-Carlo simulations using the SC MAC decoder. Our decoder outputs soft likelihood ratios for both u^N and v^N which are averaged over large number of simulations and used as reliability values. The code design is specific to the underlying MAC and involves finding a path b^{2N} and information index sets $\mathcal{A}_u$ and $\mathcal{A}_v$ for a desired target rate pair (R_x, R_y). Although there may be many possible paths satisfying the required rate pair we restricted ourselves to a class of paths of the form $0^i1^N0^{N-i}$ for $0 \le i \le N$. These paths produce rate pairs that span the entire dominant face of the MAC region.

V. Conclusions

We presented a SC decoder of polar codes for two-user binary-input MAC. The class of polar codes considered are generated by monotone chain rules of [5]. We considered paths of the form $0^i1^N0^{N-i}$ for $0 \le i \le N$. By adjusting i we could reach any point on the dominant face of $\mathcal{R}(W)$ with sufficient precision. We increased performance by employing list decoding and presented simulation results for two different MACs. The decoder was implemented based on efficient structure of [7] yielding time complexity of $O(L \cdot N \log N)$ and space complexity of $O(L \cdot N)$ for a list size L.

The second type of MAC we used in our simulations was selected that way to serve a secondary purpose: The decoder optimized for such a MAC with a given distribution $p_{E_x E_y}$ is also the decoder of the SW problem in [5] for source pair (E_x, E_y). The source decoding would be performed by setting decoder input z^N to $(0,0)^N$ and frozen bits vectors $u_{\mathcal{A}_u^c}$ and $v_{\mathcal{A}_v^c}$ to the compressed vectors from encoders of user 1 and 2, respectively. Due to space limitations we do not present results on source compression here.

Acknowledgment

Helpful discussions with E. Arıkan are gratefully acknowledged. This work was supported in part by the European Commission in the framework of the FP7 Network of Excellence in Wireless COMmunications NEWCOM# (contract n.318306) and in part by TÜBİTAK under grant 110E243.

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, pp. 3051 –3073, July 2009.
[2] S. B. Korada, *Polar codes for channel and source coding*. PhD thesis, EPFL, Lausanne, Switzerland, July 2009.
[3] E. Arıkan, "Source polarization," in *Proc. IEEE Int'l Symp. Inform. Theory (ISIT'2010)*, (Austin, TX, U.S.A.), pp. 899–903, June 2010.
[4] E. Şaşoğlu and E. Telatar, "Polar codes for the two-user binary-input multiple-access channel," in *Proc. IEEE Inform. Theory Workshop (ITW)*, (Dublin, Ireland), p. 1–5, 2010.
[5] E. Arıkan, "Polar coding for the slepian-wolf problem based on monotone chain rules," in *Proc. IEEE Int'l Symp. Inform. Theory (ISIT'2012)*, (Cambridge, MA, U.S.A.), pp. 566 –570, July 2012.
[6] J. D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inf. Theory*, vol. IT-19, pp. 471–480, July 1973.
[7] I. Tal and A. Vardy, "List decoding of polar codes," *arXiv:1206.0050*, May 2012.
[8] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Wiley-Interscience, 2 ed., July 2006.