Non-Stationary Polar Codes for Resistive Memories

Marwen Zorgui, Mohammed E. Fouda, Zhiying Wang, Ahmed M. Eltawil and Fadi Kurdahi

Electrical Engineering and Computer Science, University of California–Irvine, CA, USA

Abstract—Resistive memories are considered a promising memory technology enabling high storage densities. However, the readout reliability of resistive memories is impaired due to the inevitable existence of wire resistance, resulting in the sneak path problem. Motivated by this problem, we study polar coding over channels with different reliability levels, termed non-stationary polar codes, and we propose a technique improving the bit error rate (BER) performance. We then apply the framework of non-stationary polar codes to the crossbar array and evaluate its BER performance under two modeling approaches, namely binary symmetric channels and binary asymmetric channels. Finally, we propose a technique for biasing the proportion of high-resistance states in the crossbar array and show its advantage in reducing further the BER. Several simulations are carried out using a SPICE-like simulator, exhibiting significant reduction in BER.

Index Terms—Polar codes, non-stationary channels, resistive memories, sneak-path problem, bit-reversal permutation.

I. INTRODUCTION

Emerging nonvolatile memories (NVMs), such as phase change memory (PCRAM), ferroelectric memory (FeRAM), spin transfer torque magnetic memory (STT-MRAM), and resistive memory (RRAM), have shown high potential as alternatives for floating-gate-based nonvolatile memories [1]. RRAMs are considered one of the best candidates for the next generation nonvolatile memory due to their high reliability, fast access speed, multilevel capabilities and stack-ability creating 3D memory architectures.

In resistive memories, the high/low cell resistance represents the stored bit. In order to retrieve the stored data, resistive sensing (reading) techniques are adopted. To achieve higher density, access devices such as transistors, diodes and selectors are removed. However, the main disadvantage of the selectorless (gate-less) crossbar-based memories is *sneak paths* which are undesired current paths and voltage drops limiting the readability of the array. In this paper, a parallel reading of an entire crossbar row [2] is adopted as shown in Fig. 1. It eliminates the multi-path problem in single-cell reading [3], one of the causes of sneak paths. But, the inevitable wire resistances lead to undesired voltage drops, another type of sneak paths. These voltage drops are functions of the stored data and the wire resistance. At the expected feature size of $F = 5nm$ of RRAMs, the wire resistance per cell reaches as high as 90Ω [4], leading to large voltage drops.

Fig. 2a shows the measured current of each cell in a (32×32) array with 25Ω wire resistance, storing random data, which is generated by the SPICE-like simulator of [4]. Due to sneak paths, the sensed current of low resistance state decreases in both vertical and horizontal directions in the array. The top left cells have distinguishable distributions for the stored ones and zeros. On the other hand, bits in the rightbottom cells are indistinguishable due to the read margin
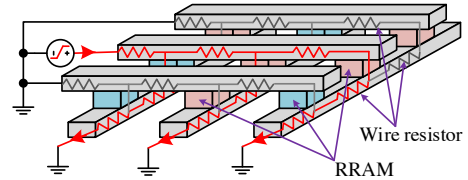

Fig. 1: Parallel reading of the entire row in the crossbar. The columns and rows are grounded, except the row being read. The red arrow shows the sensed current flowing through wire resistances and RRAMs.
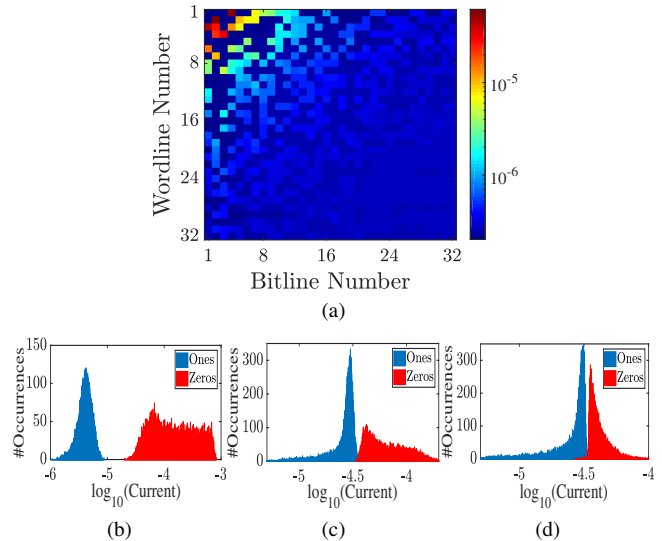

Fig. 2: (a) Measured current per cell, and its histogram for bitline number (b) 1, (c) 16, and (d) 32.

overlap. Figures 2b, 2c, and 2d show the histogram of the measured currents of the 1st, the 16th and the 32nd bitline (column), respectively. Clearly, the larger the bitline index is, the more errors occur. We can see that the channels of the cells have varying reliability.

Addressing the sneak path problem has attracted a lot of interest from both research and industry communities. Proposed solutions include hardware-based approaches, e.g., transistor gating [5], and/or techniques based on communication and coding theory [6], [7]. The focus of this paper is on the latter approach and our scheme is based on polar codes. Polar codes [8] are the first family of explicit error-correcting codes to provably achieve the capacity of binary symmetric channels, with a low-complexity encoding and successive cancellation decoding. For a code of length N, encoding/decoding has a complexity of $O(N \log N)$. For the above reasons, polar codes constitute an attractive error correction scheme.

Our contributions are summarized as follows:

• We study polar coding for channels with different reliability levels, termed non-stationary polar codes.

- We argue that the ordering of the channels is important and propose an ordering with a numerically competitive performance.
- We then apply the framework of non-stationary polar codes to the sneak path problem and demonstrate significant improvement in terms of bit error rate (BER). In particular, we discuss modeling the array cells as either binary symmetric channels (BSCs) and as binary asymmetric channels (BACs) and compare their BER performances.
- Finally, we propose a technique for biasing the number of high-resistance values in the crossbar through puncturing, and show by simulation that the proposed technique can help reduce the BER in certain scenarios.

The remainder of the paper is organized as follows. In Section II, we analyze the framework for polar codes over channels with different reliability levels and describe our approach for employing such polar codes. In Section III-A, we apply the developed framework to the crossbar array and describe the various techniques proposed to further reduce the BER. Section IV draws conclusions and discusses future research directions.

Notation. Vectors are denoted with lower-case bold letters. A permutation π over the integers $\{0, \ldots, N-1\}$ is denoted as $\pi = [\pi(0), \ldots, \pi(N-1)]$, where $\pi(j)$ is the image of j under π. For a vector $\mathbf{x} = [x_0, \ldots, x_{N-1}]$, $\mathbf{x}_\pi$ denotes the vector $\mathbf{x}_\pi = [x_{\pi(0)}, \ldots, x_{\pi(N-1)}]$.

II. Non-stationary Polar Code Construction

For a binary-input discrete memoryless channel (B-DMC), W, with output alphabet $\mathcal{Y}$, we denote its transition probabilities by $W(y|x), x \in \{0,1\}, y \in \mathcal{Y}$, and define the symmetric channel output probability as $W(y) = \frac{1}{2}W(y|0) + \frac{1}{2}W(y|1)$. Define the symmetric capacity $I(W)$ as

$$I(W) \triangleq \sum_{y \in \mathcal{Y}} \sum_{x \in \{0,1\}} \frac{1}{2}W(y|x) \log \frac{W(y|x)}{W(y)}. \tag{1}$$

Arikan's polar transformation [8] manufactures out of N independent copies of a given B-DMC channel, a second set of N *synthesized* binary-input channels. The channels show a polarization effect, namely, as N becomes large, the symmetric capacities of the synthesized channels tend towards 0 or 1 for all but a vanishing fraction.

In our framework, in contrast to the original polar codes, we consider the extension of polar codes to the setting where the underlying channels are of *varying reliability levels* (see Fig. 3). Using the terminology in [9], we refer to such polar codes as non-stationary polar codes.

Polar Transformation. Let $\mathbf{u} = [u_0, u_1, \ldots, u_{N-1}]$ and $\mathbf{x} = [x_0, x_1, \ldots, x_{N-1}]$ be the input and the output of a length-N polar code, respectively, with $N = 2^n$ for some integer n. The polar transformation is given by

$$\mathbf{x} = \mathbf{u}G, \quad G = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}^{\otimes n},$$

where the symbol $\otimes n$ denotes the n-th Kronecker power operator.

When $n = 1, N = 2$, the basic polarization transformation is applied to two independent channels $W_0 : \{0,1\} \to \mathcal{Y}_0$ and

$W_1 : \{0,1\} \to \mathcal{Y}_1$, resulting in two channels, $W' : \{0,1\} \to \mathcal{Y}_0 \times \mathcal{Y}_1$ and $W'' : \{0,1\} \to \mathcal{Y}_0 \times \mathcal{Y}_1 \times \{0,1\}$, given by

$$W'(y_0, y_1|u_0) = \frac{1}{2} \sum_{u_1 \in \{0,1\}} W_0(y_0|u_0 \oplus u_1)W_1(y_1|u_1),$$

$$W''(y_0, y_1, u_0|u_1) = \frac{1}{2} W_0(y_0|u_0 \oplus u_1)W_1(y_1|u_1), \tag{2}$$

where $y_0 \in \mathcal{Y}_0, y_1 \in \mathcal{Y}_1$, and $u_0, u_1 \in \{0,1\}$. We denote this single-step transformation by $(W_0, W_1) \mapsto (W', W'')$.

Channel Polarization. The transformation preserves the average symmetric capacity, while exhibiting a polarization effect.

Lemma 1 ([9]). *Suppose* $(W_0, W_1) \mapsto (W', W'')$. *Then,*

$$I(W') + I(W'') = I(W_0) + I(W_1),$$
$$I(W') \leq I(W_i) \leq I(W''), i = 0, 1. \tag{3}$$

The Bhattacharya parameter of a binary discrete memoryless channel $W : \{0,1\} \to \mathcal{Y}$, denoted by $Z(W)$, is a measure of the reliability of W, and has been used to bound the error probability of polar codes in [8]. The parameter $Z(W)$ is defined as

$$Z(W) \triangleq \sum_{y \in \mathcal{Y}} \sqrt{W(y|0)W(y|1)}.$$

Lemma 2. *Let* $W_0 : \{0,1\} \to \mathcal{Y}_0$ *and* $W_1 : \{0,1\} \to \mathcal{Y}_1$ *and* $(W_0, W_1) \mapsto (W', W'')$.
(i) The following relations hold:

$$Z(W') \leq Z(W_0) + Z(W_1) - Z(W_0)Z(W_1), \tag{4}$$
$$Z(W'') = Z(W_0)Z(W_1), \tag{5}$$
$$Z(W') \geq \sqrt{Z(W_0)^2 + Z(W_1)^2 - Z(W_0)^2 Z(W_1)^2}. \tag{6}$$

(ii) Equality (4) holds with equality iff W_0 or W_1 is a binary erasure channel.
(iii) Equality (6) holds with equality iff both W_0 and W_1 are binary symmetric channels.

In Lemma 2, parts (i) and (iii) are from [10, Lemma 2.15 and Lemma 2.16] and [11, Lemma 9], and part (ii) can be proved following similar lines as [8, Proposition 5]. Lemma 2 implies that reliability improves under a single-step channel transformation in the sense that

$$Z(W') + Z(W'') \leq Z(W_0) + Z(W_1),$$

with equality iff W_0 or W_1 is a binary erasure channel (BEC).

The single-step transformation, as described in (2), can be generalized to the case of $N = 2^n$ binary, memoryless, but not necessarily stationary channels $\{W_i\}_{i=0}^{N-1}$. In particular, let $W_{m,i}$ denote the i-th bit channel after m levels of polarization of the sequence $\{W_i\}_{i=0}^{N-1}$, where $W_{0,i} \triangleq W_i, i = 0, \ldots, N-1$. Applying Arikan's polar transformation results in a collection of synthesized channels, such that, for any level $1 \leq l \leq n$, for $0 \leq j < 2^{l-1}, 0 \leq m < 2^{n-l}$, we have

$$(W_{l-1,2^l m+j}, W_{l-1,2^l m+2^{l-1}+j}) \mapsto$$
$$(W_{l,2^l m+j}, W_{l,2^l m+2^{l-1}+j}). \tag{7}$$

In [9], it was shown that the fraction of non-polarized channels approach 0, i.e., for every $0 < a < b < 1$,

$$\liminf_{N \to \infty} \frac{1}{N} |0 \leq i < N : I(W_{n,i}) \in [a,b]\}| = 0.$$

The work in [11] proved that in the asymptotic regime where the blocklength N is large, the effective average symmetric capacity $\bar{I}(\{W_i\}_{i=0}^{\infty}) \triangleq \lim_{N \to \infty} \frac{1}{N} \sum_{i=1}^{N} I(W_i)$ is achievable. Moreover, the polar coding scheme is constructed based on Arikan's channel polarization transformation in combination with certain permutations at each polarization level. However, it is not clear whether such permutation choices yield attractive performances for practical lengths. In this work, we are concerned with the performance of non-stationary polar codes in the finite blocklength regime.

Encoding and Decoding. After the polar transformation, one obtains N synthesized channels $\{W^{(i)} \triangleq W_{n,i}, 0 \leq i \leq N-1\}$. A polar code of dimension k transmits k information bits in the k synthesized channels with the highest $I(W^{(i)})$ (we denote the corresponding information set by $\mathcal{I}$), and $N-k$ arbitrary but fixed bits in the remaining $N-k$ synthesized channels (denoted by $\mathcal{F}$). Decoding of polar codes is carried out using successive cancellation decoding as in [8], taking into account the appropriate likelihood ratios of the original channels.

Remark 1. *The information set of a non-stationary polar code can be determined as long as $I(W^{(i)})$ are computed. However, their computational complexities are generally unmanageable. Instead, we propose that the reliability parameters of the synthesized channels can be approximated efficiently by $Z(W^{(i)})$ using (4) and (5) in Lemma 2. We call this method the Bhattacharyya bound approach. It is a generalization of the algorithm in [12], developed for regular polar codes. Let $Z_{n,i}$ denote the Bhattacharyya parameter of channel $W_{n,i}$. Then, we apply the following recursion: for level $1 \leq l \leq n$, for $0 \leq j < 2^{l-1}$ and $0 \leq m < 2^{n-l}$, we have*

$$Z_{l,2^l m+j} = Z_{l-1,2^l m+j} + Z_{l-1,2^l m+2^{l-1}+j}$$
$$- Z_{l-1,2^l m+j} Z_{l-1,2^l m+2^{l-1}+j},$$
$$Z_{l,2^l m+2^{l-1}+j} = Z_{l-1,2^l m+j} Z_{l-1,2^l m+2^{l-1}+j},$$

where $\{Z_{0,i}, 0 \leq i < N\}$ are the initial channel Bhattacharyya parameters. The indices of the lowest $N-k$ values in the set of N final stage values form the set $\mathcal{F}$. This algorithm is an evolution of the Bhattacharyya parameters of channels from right to left (see Fig. 3), preferably applied in log-domain to avoid underflow.

Role of the Channel Ordering. The performance of polar codes depends on the synthesized channels of the information set $\sum_{i \in \mathcal{I}} I(W^{(i)})$ [8]. As illustrated in Fig. 3, to construct a non-stationary polar code, we propose to apply a permutation π to the vector $\mathbf{x}$ in order to enhance the overall performance. Ideally, we want to find a permutation π^* such that for all permutation π,

$$\sum_{i \in \mathcal{I}_{\pi^*}} I(W_{\pi^*}^{(i)}) \geq \sum_{i \in \mathcal{I}_{\pi}} I(W_{\pi}^{(i)}), \tag{8}$$
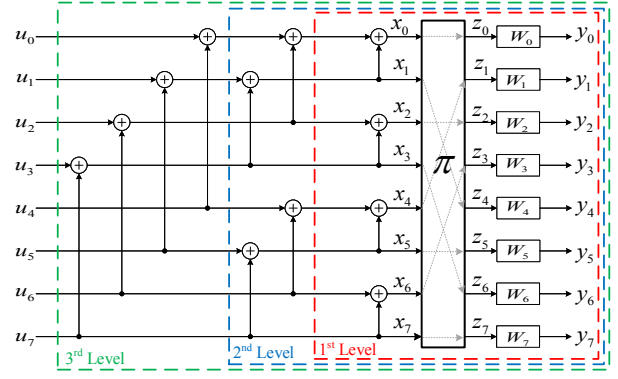

Fig. 3: Polar encoding with $N = 8$ channels, with permutation $\pi = [0, 4, 2, 6, 1, 5, 3, 7]$.

where $I(W_{\pi}^{(i)})$ is the symmetric capacity of the i-th synthesized channel under permutation π and $\mathcal{I}_{\pi}$ is its information set.

Remark 2. *See Fig. 4 for a schematic picture of the full encoder and decoder. The permutation π is defined such that $\mathbf{z}_{\pi} = \mathbf{x}$, or equivalently $\mathbf{z} = \mathbf{x}_{\pi^{-1}}$. Then, $z_{\pi(i)} = x_i$, implying that symbol x_i goes through channel $W_{\pi(i)}$, for $0 \leq i < N$. Correspondingly, a reverse permutation is required for the output of the channels. The polar decoder receives as input the vector $\mathbf{y}' = [y_{\pi(0)}, \ldots, y_{\pi(N-1)}] = \mathbf{y}_{\pi}$.*

Remark 3. *In general, the channel symmetric capacities can be given in any arbitrary order. In this case, to study the effect of permutations in a canonical way, we apply the composition permutation $\pi_{ord} \circ \pi$ to the output $\mathbf{x}$, where π_{ord} is a permutation that orders the channels in increasing order of symmetric capacities: that is $I(W_{\pi_{ord}(i)}) \leq I(W_{\pi_{ord}(j)})$, for $i < j$. Thus, the output of the polar encoder x_i is mapped to channel $W_{\pi_{ord} \circ \pi(i)}, i.e., x_i = z_{\pi_{ord} \circ \pi(i)}.$*

We next show that some channel orderings are equivalent after polarization.

Definition 3. *Given N B-DMC channels $\{W_i\}_{i=0}^{N-1}$, an equivalence class of permutations over $\{0, \ldots, N-1\}$ consists of all channel permutations π such that the n-level polarization transformation of the channels $\{W_{\pi(i)}\}_{i=0}^{N-1}$ results in the same symmetric capacities.*

Lemma 4. *The number of permutation classes is upper bounded by $\frac{N!}{2^{N-1}}$.*

Proof: Consider the two single-step transformations $(W_0, W_1) \mapsto (W_1', W_1'')$ and $(W_1, W_0) \mapsto (W_2', W_2'')$. From (1) and (2), we have

$$I(W_1'') = \sum_{y_0, y_1, u_0} \sum_{u_1} \frac{1}{4} W_0(y_0|u_0 \oplus u_1) W_1(y_1|u_1)$$
$$\times \log \frac{\frac{1}{2} W_0(y_0|u_0 \oplus u_1) W_1(y_1|u_1)}{W_1''(y_0, y_1, u_0)}$$
$$= \sum_{y_0, y_1, u_0} \sum_{u_1} \frac{1}{2} W_0(y_0|u_1) W_1(y_1|u_0 \oplus u_1)$$
$$\times \log \frac{W_0(y_0|u_1) W_1(y_1|u_0 \oplus u_1)}{W_1''(y_0, y_1, u_0)}, \tag{9}$$
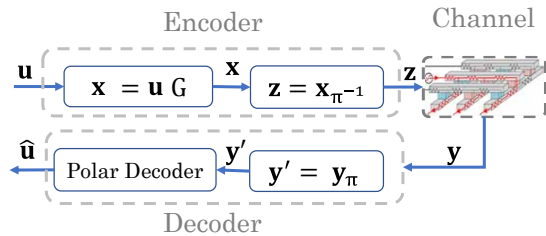
Fig. 4: Full system model.

where the second equality is obtained by a change of variable. On the other hand, we have

$$
\begin{aligned}
W_1''(y_0, y_1, u_0) &= \sum_{u_1 \in \{0,1\}} W_0(y_0|u_0 \oplus u_1) W_1(y_1|u_1) \\
&= \sum_{u_1 \in \{0,1\}} W_0(y_0|u_1) W_1(y_1|u_0 \oplus u_1) \\
&= W_2''(y_0, y_1, u_0).
\end{aligned}
\tag{10}
$$

From (9) and (10), we obtain $I(W_1'') = I(W_2'')$. From (3), it follows $I(W_1') = I(W_2')$. Thus, the single-step transformation is symmetric in the sense that exchanging the order of W_0 and W_1 does not impact the symmetric capacities of the synthesized channels. It follows that for $N = 2^n$ channels, permuting channel W_{2i} and channel W_{2i+1}, for $0 \le i \le \frac{N}{2} - 1$ does not change the symmetric capacities of the synthesized channels obtained at the end of the polarization process. Extending the above reasoning to level 2, permuting (W_{4i}, W_{4i+1}) with (W_{4i+2}, W_{4i+3}), $0 \le i \le \frac{N}{4} - 1$ results in the same symmetric capacities of the synthesized bit channels. At level 3, we can permute $(W_{8i}, W_{8i+1}, W_{8i+2}, W_{8i+3})$ with $(W_{8i+4}, W_{8i+5}, W_{8i+6}, W_{8i+7})$, $0 \le i \le \frac{N}{8} - 1$ without changing the symmetric capacities. Similar observation applies to all polarization levels l, for $1 \le l \le n$. Thus, the number of permutation classes is upper bounded by $\frac{N!}{2^{1+2+\dots+\frac{N}{2}}} = \frac{N!}{2^{1+2+\dots+2^{n-1}}} = \frac{N!}{2^{2^n-1}} = \frac{N!}{2^{N-1}}$. ∎

Remark 4. *Equivalent classes of permutations can be equally defined in terms of the Battacharaya parameters instead of symmetric capacities, and the statement of Lemma 4 holds.*

For $N = 4$ BECs with different erasure probabilities, Proposition 1 determines the best ordering depending on the target rate. The proof can be found online in [13].

Proposition 1. *Consider $N = 4$ BECs with parameters $\epsilon_i, 1 \le i \le 4$. Then, the permutation $\pi_{ord} \circ \pi$, where $\pi = [0, 3, 1, 2]$, satisfies (8).*

Extending the analysis in Proposition 1 is not tractable. In the following, we consider the bit-reversal permutation defined below. We note that such permutation was used in [14] in the context of rate-compatible punctured polar codes and was shown to yield attractive performance compared to other existing or random puncturing patterns. We explore the performance under bit-reversal permutation through various numerical examples.

Definition 5. *We define ψ to be the bit-reversal permutation. For each integer $i \in \{0, \dots, 2^n - 1\}$, $\psi(i)$ is the integer obtained by reversing the binary representation of i. That is, let*
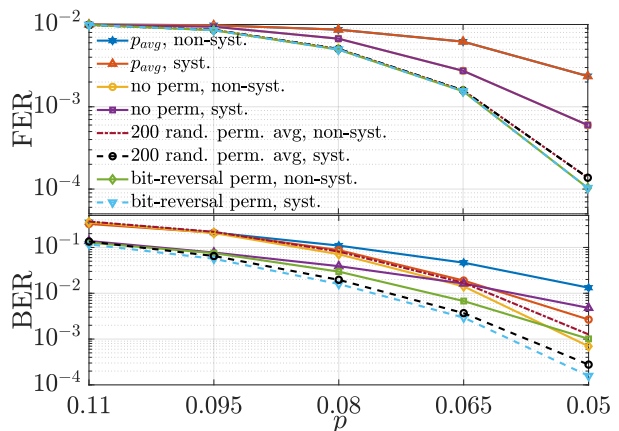

Fig. 5: Performance evaluation for BSCs with linearly spaced cross-over probabilities. $N = 1024, k = 512$.

$i = \sum_{j=1}^n b_j 2^{j-1}$, *then* $\psi(i) = \sum_{j=1}^n b_j 2^{n-j}$. *As an example, when $N = 8$, $\psi = [0, 4, 2, 6, 1, 5, 3, 7]$, as illustrated in Fig. 3.*

From the proof of Proposition 1 in [13] and Lemma 4, when $\epsilon_0 = \epsilon_1$ or $\epsilon_2 = \epsilon_3$, then ψ satisfies (8) and is optimal.

Example 1. *We consider N binary symmetric channels (BSCs) with varying cross-over probabilities, linearly spaced and centered at value $p \in \{0.05, 0.065, 0.08, 0.095, 0.11\}$, with maximum deviation of 0.045. We consider a rate $\frac{1}{2}$ and a block size $N = 2^{10}$. We evaluate the performance of a regular polar code designed for the average BSC channel, with p_{avg} satisfying $N(1 - h(p_{avg})) = \sum_{i=0}^{N-1}(1 - h(p_i))$, where $h(\cdot)$ is the binary entropy function and p_i's are the cross-over probabilities of the channels. The default ordering of the channels is such that the channels' Bhattacharya parameters are ordered in a descending order (i.e., decreasing order of p_i's). We evaluate the performance with the default ordering (no permutation) and the performance with the bit-reversal permutation. We also evaluate the performance obtained by averaging 200 random permutations. For each scenario, we evaluate the frame error rate (FER) and the bit error rate (BER) with non-systematic encoding and systematic encoding [15], for 10^4 runs. Fig. 5 illustrates the results. Similar to [15], we observe that systematic encoding of non-stationary polar codes enhances the BER performance compared to non-systematic encoding, while keeping the FER unchanged. The regular polar code with p_{avg} exhibits the worst BER, while the non-stationary polar code under systematic encoding with $\pi = \psi$ performs the best. In particular, the latter scenario outperforms all 200 random permutations for all values of p.*

Based on the observation that bit-reversal permutation achieves competitive BER numerically, we choose to apply $\pi = \psi$ for our non-stationary polar codes, so as to mitigate the sneak path problem in crossbar arrays.

III. APPLICATION TO RESISTIVE CROSSBAR ARRAYS

A. General Framework

As outlined in the introduction, the crossbar array cells exhibit different reliability levels. For this reason, we propose the application of non-stationary polar codes to address the problem. We apply a two-step approach:
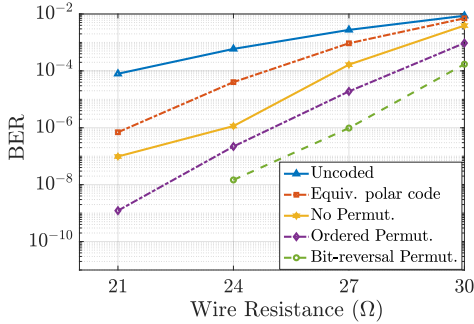
Fig. 6: Performance evaluation for a (32×32) crossbar array, code rate 0.8.

Fig. 7: BER performance under BSC and BAC modeling for a (32×32) crossbar, code rate 0.8.

Step 1). We first estimate a single detection threshold for each wordline (row) to minimize the overall uncoded BER per word. The threshold for each wordline is estimated by generating large training data and then applying a good binary classifier. For instance, we observe that a logistic regression-based classifier gives superior performance in terms of accuracy and speed.

Step 2). Based on the estimated thresholds in Step 1), we model the read channels as BSCs or BACs. In particular, we estimate the cross-over probabilities of each cell in the array. Our simulation shows that the channels are indeed varying. We then apply non-stationary polar codes using the cell characterizations.

In a preliminary version of this work [16], we proposed to encode each row separately. In this paper, we focus on encoding the entire array. This can be suitable for applications such as archival data and image storage. Assuming the crossbar array size is $(N_1 \times N_2)$, then, the blocklength is $N = N_1 N_2$. The encoded output symbol $z_{in+j}, 0 \leq i < m, 0 \leq j < n$, is stored at the (i,j)-th entry in the crossbar array (i.e., we vectorize the array row by row). Instead of using high-level models for the sneak path problem, such as in [6], [7], we use a SPICE-like simulator that is built based on accurate modeling of the resistive crossbar array [4]. This numerical simulator offers a fast alternative to SPICE simulators while maintaining the same simulation accuracy. In our simulations, the high resistance state, representing 1, and the low resistance state, representing 0, are set to $1M\Omega$ and $1k\Omega$, respectively.

B. BSC and BAC Channel Modeling

First, the array cells are modeled as BSCs. The cross-over probability is found to match the BER of each cell.

Example 2. *In Fig. 6, we simulate the BER performance of systematic polar codes for four cases: (i) equivalent regular polar codes correspond to BSCs with parameter p_{avg}, similar to Example 1, (ii) no permutation, (iii) permutation π_{ord}, and (iv) permutation $\pi = \pi_{ord} \circ \psi$, where π_{ord} is as defined in Remark 3. Clearly, the BER permutation under $\pi = \pi_{ord} \circ \psi$ outperforms the other permutations.*

Analyzing further the uncoded error distribution, as one may infer from Figures 2b, 2c, and 2d, we find that the conditional error distributions under 0's and 1's are different, i.e., $P(\text{error}|0) \neq P(\text{error}|1)$. Taking this observation into consideration, we model the crossbar array cells as binary
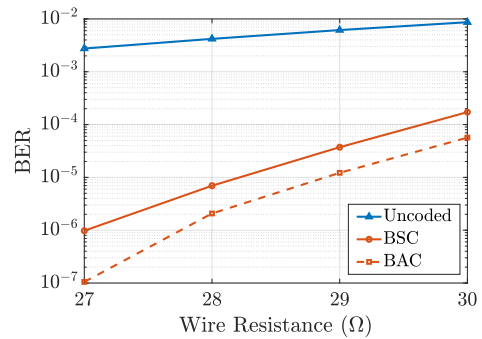
asymmetric channels (BACs) and apply the non-stationary polar codes.

Example 3. *In Fig. 7, we compare the systematic BER performance under both BSC and BAC modeling using the bit-reversal permutation. As expected, the BER performance under the more accurate BAC model is better. The improvement is 3X – 9X for wire resistance between 27Ω and 30Ω.*

C. Punctured Polar Codes

In this subsection, we propose a technique that can enhance the BER performance in some scenarios by biasing the fraction of high resistance cells. The sneak paths exist through the cells having low resistances, causing inter-cell interference [6]. Intuitively, having more high resistances in the array helps mitigate the sneak path problem. To leverage this intuition, we investigate the use of a (punctured) polar code of a shorter length, say $N - N_p$, while storing high resistances in the corresponding punctured N_p cells in the array. Clearly, there is a trade-off between two opposite factors: puncturing reduces the number of redundant codeword symbols, hence degrading the performance of polar codes, while high resistances decrease the sneak path effect resulting in fewer read errors. In the following, we investigate the application of the above approach to the crossbar array.

A *punctured* polar code is obtained from the N-length parent polar code using a binary puncturing vector $\mathbf{w} = [w_0, w_1, \ldots, w_{N-1}]$, where the zeros imply the punctured positions. We note that the information set $\mathcal{I}$ should be recomputed as in Remark 1, when we consider puncturing.

Punctured polar codes have been investigated by many works, and several efficient puncturing patterns have been proposed in the literature [14]. In [14], the authors proposed an empirically good puncturing algorithm, termed quasi-uniform puncturing (QUP). QUP-polar codes were shown through simulations to outperform the performance of turbo codes in WCDMA (Wideband Code Division Multiple Access) or LTE (Long Term Evolution) wireless communication systems in the large range of code lengths. We adopt QUP as our puncturing pattern and we highlight its advantages below.

Definition 6 ([14]). *The QUP is described as follows:*
1) *Initialize the vector $\mathbf{w}$ as all ones, and then set the first N_p bits of $\mathbf{w}$ to zeros;*
2) *Perform bit-reversal permutation on the vector $\mathbf{w}$ and obtain the puncturing pattern.*
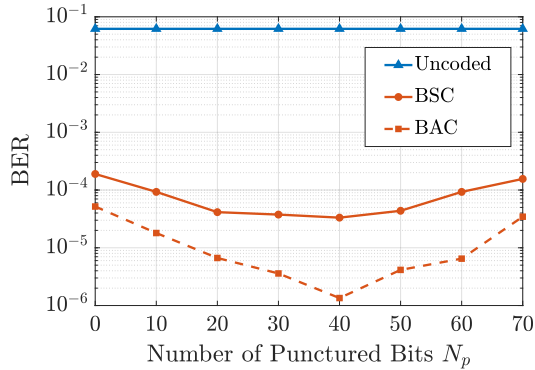
Fig. 8: Punctured polar encoding over the (32×32) crossbar array for code rate 0.8.

Example 4. *Let $N = 8, N_p = 3$. The initial puncturing vector is $\mathbf{w} = [0, 0, 0, 1, 1, 1, 1, 1]$. After bit-reversal permutation, the puncturing vector becomes $\mathbf{w} = [0, 1, 0, 1, 0, 1, 1, 1]$.*

By employing QUP in conjunction with the permutation $\pi_{\mathrm{ord}} \circ \psi$, puncturing N_p positions corresponds to not using the N_p cells with the worst reliability levels for data storage. By placing high-resistance values (i.e., 1's) in the punctured locations, we increase the frequency of 1's in the codeword by $\frac{N_p}{2N}$.

Lemma 7. *For a punctured polar code length of $N - N_p$, the frequency of 1's in the array is given by $\frac{1}{2} + \frac{N_p}{2N}$.*

Proof: Let i be chosen uniformly at random in $\{0, \ldots, N-1\}$, and let z_i be the corresponding codeword symbol. Let $\mathcal{B}_p$ be the set of punctured locations. The size of $\mathcal{B}_p$ is $N - N_p$. Then, we write

$$
\begin{aligned}
P(z_i = 1) &= P(i \in \mathcal{B}_p) P(z_i = 1 | i \in \mathcal{B}_p) \\
&\quad + P(i \notin \mathcal{B}_p) P(z_i = 1 | i \notin \mathcal{B}_p) \\
&= \frac{N_p}{N} + \frac{1}{2} \frac{N - N_p}{N} = \frac{1}{2} + \frac{N_p}{2N}.
\end{aligned}
$$

∎

Example 5. *Fig. 8 illustrates the systematic BER performance of QUP for a (32×32) array with 35Ω wire resistance, $\pi = \pi_{ord} \circ \psi$. We observe that the BER decreases as more bits (N_p) are punctured and as the frequency of 1's increases. This gain is reversed for $N_p > 40$ and the BER increases as N_p and the codeword redundancy decrease. The BER is improved by a factor of 5.7 for a symmetric channel model and by a factor of 38.5 for an asymmetric channel model.*

IV. CONCLUDING REMARKS

In this paper, motivated by the sneak path problem in resistive memories, we studied polar coding over channels with different reliability levels. In particular, we argued that the channels' ordering is important and proposed a channel ordering whose attractive performance was shown numerically. We then applied our framework to the sneak path problem in resistive memories. Simulation results on SPICE-like resistive crossbar showed significant bit-error rate performance improvement, especially for low uncoded BER. Additionally, using a more accurate binary assymetric channel modeling, the

BER is further reduced. Finally, it is shown that biasing the frequency of high-resistance values in the array can mitigate the sneak path occurrences. We note that while in this work, we modeled each cell individually as a BSC (or a BAC), the cost of such modeling is amortized by using the same characterization over several crossbar arrays, which makes the model parameter costs justifiable from a practical perspective. Moreover, it is possible to *cluster* multiple cells together in a way to reduce the number of overall crossbar model parameters.

The present work represents another step toward coding for the resistive crossbar arrays. The performance of polar codes can be improved by using more enhanced decoding algorithms (e.g. list-decoding [17]), at the expense of higher complexity. Moreover, by the sneak path nature, errors in neighboring cells are not independent. A future direction for research is to investigate enhanced polar decoding algorithms, taking into consideration such correlations. Another avenue for research is to investigate other techniques for biasing the distribution of high-resistance values, along with techniques of coding for asymmetric channels as in [18].

REFERENCES

[1] D. C. Daly, L. C. Fujino, and K. C. Smith, "Through the looking glass-the 2017 edition: Trends in solid-state circuits from isscc," *IEEE Solid-State Circuits Magazine*, vol. 9, no. 1, pp. 12–22, 2017.

[2] M. E. Fouda, A. M. Eltawil, and F. Kurdahi, "On resistive memories: One step row readout technique and sensing circuitry," *arXiv preprint arXiv:1903.01512*, 2019.

[3] M. Zidan, H. Omran, R. Naous, A. Sultan, H. Fahmy, W. Lu, and K. N. Salama, "Single-readout high-density memristor crossbar," *Scientific reports*, vol. 6, p. 18863, 2016.

[4] M. E. Fouda and et.al, "Modeling and analysis of passive switching crossbar arrays," *IEEE Trans. on Circuits and Systems I: Regular Papers*, vol. 65, no. 1, pp. 270–282, 2018.

[5] M. A. Zidan, H. A. H. Fahmy, M. M. Hussain, and K. N. Salama, "Memristor-based memory: The sneak paths problem and solutions," *Microelectronics Journal*, vol. 44, no. 2, pp. 176–183, 2013.

[6] Z. Chen, C. Schoeny, and L. Dolecek, "Coding assisted adaptive thresholding for sneak-path mitigation in resistive memories," in *IEEE Information Theory Workshop (ITW)*. IEEE, 2018, pp. 1–5.

[7] Y. Ben-Hur and Y. Cassuto, "Detection and coding schemes for sneak-path interference in resistive memory arrays," *IEEE Transactions on Communications*, 2019.

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

[9] M. Alsan and E. Telatar, "A simple proof of polarization and polarization for non-stationary memoryless channels," *IEEE Trans. Inf. Theory*, vol. 62, no. 9, pp. 4873–4878, 2016.

[10] S. B. Korada, "Polar codes for channel and source coding," 2009.

[11] H. Mahdavifar, "Fast polarization for non-stationary channels," in *IEEE International Symposium on Information Theory*, 2017, pp. 849–853.

[12] E. Arikan, "A performance comparison of polar codes and reed-muller codes," *IEEE Communications Letters*, vol. 12, no. 6, pp. 447–449, 2008.

[13] M. Zorgui et. al, "Non-stationary polar codes for resistive memories," 2019. [Online]. Available: http://faculty.sites.uci.edu/zhiying/publications/

[14] K. Niu, K. Chen, and J.-R. Lin, "Beyond turbo codes: Rate-compatible punctured polar codes," in *IEEE Int. Conf. on Communications (ICC)*. IEEE, 2013, pp. 3423–3427.

[15] E. Arikan, "Systematic polar coding," *IEEE communications letters*, vol. 15, no. 8, pp. 860–862, 2011.

[16] M. Zorgui et al., "Polar coding for selector-less resistive memories," in *10th Annual Non-Volatile Memories Workshop*, 2019.

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

[18] R. Gabrys and L. Dolecek, "Coding for the binary asymmetric channel," in *Int. Conf. Comp., Net. and Comm. (ICNC)*, Jan 2012, pp. 461–465.