Fast Multi-dimensional Polar Encoding and Decoding

Hessam Mahdavifar, Mostafa El-Khamy, Jungwon Lee, Inyup Kang

Mobile Solutions Lab, Samsung Research America
4921 Directors Place, San Diego, CA 92121
{h.mahdavifar, mostafa.e, jungwon2.lee, inyup.kang}@samsung.com

***Abstract*—New decoding methods for polar codes are derived from a multi-dimensional realization of polar codes. The notion of polar transformation is extended into higher dimensional polar transformations. It is shown that two dimensional polar transformation leads to a rate-adaptive doubly concatenated polar code. The results are generalized to n-dimensional construction and decoding of polar codes, where $N = 2^n$ is the length of the code. The proposed structure for successive cancellation decoding, based on the n-dimensional construction of polar codes, results in efficient decoding latency and space complexity. Soft successive decoding of polar codes and iterative detection and decoding of compound polar codes are implemented using the proposed structure for multi-dimensional decoding of polar codes.**

I. Introduction

Polar codes, introduced by Arıkan in 2008 [1], are the first family of capacity-achieving codes with low encoding and decoding complexity for a certain class of channels i.e. binary-input symmetric memoryless (BSM) channels. Construction of polar codes is based on a phenomenon called the *channel polarization*. Arıkan proves that as the block length goes to infinity the channels seen by individual bits through a certain transformation called the *polar transformation* start polarizing which means that they approach either a noise-less channel or a pure-noise channel. Moreover, the fraction of almost noise-less bit-channels approach the capacity of the channel. This phenomenon leads to the following construction for polar codes: transmit information bits on almost noiseless bit-channels, also called good bit-channels, while freezing the rest of them to some known values, say zeros. Arıkan also proposed a low-complex decoder, the successive cancellation (SC) decoder, that guarantees the capacity-achieving property in an asymptotic sense [1]. The decoding complexity of the proposed SC decoder in original Arıkan's work is $O(N \log N)$, where N is the length of polar code.

Polar codes and polarization phenomenon have been successfully applied to various problems such as wiretap channels [2], data compression [3], [4], multiple access channels [5]–[7] and bradcast channels [8], [9]. Also, various methods has been proposed in order to make polar codes suitable for practical implementations [10]–[13]. In this paper, we aim at improving the performance of polar codes by extending the notion of polar transformation into multi-dimensional polar codes and applying soft decoding and iterative detection and decoding methods on top of that. We propose the new multi-dimensional encoding and decoding of polar codes, based on binary trees. It is shown that both the latency and space complexity of the decoder are $O(N)$, where N is the length of the code. Soft successive decoding of polar codes, based on the proposed decoding tree structure, is discussed, where the equations for updating log likelihood ratios (LLR) across the binary tree are derived. These results are extended to the recently proposed compound polar codes for application in bit interleaved coded modulation (BICM) channels. Also an iterative decoding and detection (IDD) is proposed for improving the performance of the BICM detector.

II. Overview of polar codes

In this subsection, we provide a brief overview of the groundbreaking work of Arıkan [1] on polar codes and channel polarization.

Polar codes are constructed based upon a phenomenon called *channel polarization* discovered by Arıkan [1]. The basic polarization matrix is given as

$$G = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \tag{1}$$

The Kronecker powers of G are defined by induction. Let $G^{\otimes 1} = G$ and for any $n > 1$:

$$G^{\otimes(n)} = \begin{bmatrix} G^{\otimes(n-1)} & 0 \\ G^{\otimes(n-1)} & G^{\otimes(n-1)} \end{bmatrix}$$

It can be observed that $G^{\otimes(n)}$ is a $2^n \times 2^n$ matrix. Let $N = 2^n$. Then $G^{\otimes n}$ is the $N \times N$ polarization matrix. Let $(U_1, U_2, \ldots, U_N)$, denoted by U_1^N, be a block of N independent and uniform binary random variables. The polarization matrix $G^{\otimes n}$ is applied to U_1^N to get $X_1^N = U_1^N G^{\otimes n}$. Then X_i's are transmitted through N independent copies of a binary-input discrete memoryless channel (B-DMC) W. The output is denoted by Y_1^N. This transformation with input U_1^N and output Y_1^N is called the polar transformation. In this transformation, N independent uses of W is transformed into N bit-channels, described next. Following the convention, random variables are denoted by capital letters and their instances are denoted by small letters. Let $W^N : \mathscr{X}^N \to \mathscr{Y}^N$

denote the channel consisting of N independent copies of W i.e.

$$W^N(y_1^N|x_1^N) \stackrel{\text{def}}{=} \prod_{i=1}^{N} W(y_i|x_i) \tag{2}$$

The combined channel $\widetilde{W}$ is defined with transition probabilities given by

$$\widetilde{W}(y_1^N|u_1^N) \stackrel{\text{def}}{=} W^N\big(y_1^N \,|\, u_1^N G^{\otimes n}\big) \tag{3}$$

For $i = 1, 2, \ldots, N$, the bit-channel $W_N^{(i)}$ is defined as follows:

$$W_N^{(i)}\big(y_1^N, u_1^{i-1}|u_i\big) \stackrel{\text{def}}{=} \frac{1}{2^{n-1}} \sum_{u_{i+1}^N \in \{0,1\}^{n-i}} \widetilde{W}\Big(y_1^N \,|\, u_1^N\Big) \tag{4}$$

Intuitively, this is the channel that bit u_i observes through a *successive cancellation (SC)* decoder, deployed at the output. Under this decoding method, proposed by Arıkan for polar codes [1], all the bits u_1^{i-1} are already decoded and are assumed to be available at the time that u_i is being decoded. The channel polarization theorem states that as N goes to infinity, the bit-channels start polarizing meaning that they either become a noise-less channel (good bit-channel) or a pure-noise channel (bad bit-channel).

III. Multi-dimensional polar codes

In this section, we extend the polar transformation introduced by Arıkan in [1], into two-dimensional polar transformation i.e. a transformation on matrices rather than vectors, and then into multi-dimensional polar transformation. As we will see, these two notions turn out to be equivalent. However, the multi-dimensional realization of polar codes along with a binary tree based successive decoding results in an efficient decoding latency and space complexity.

A. Two-dimensional polar transformation

Let m and n be two non-negative integers and let $M = 2^m$ and $N = 2^n$. Consider an $M \times N$ matrix $U \in \mathbb{F}_2^{M\times N}$ containing MN independent uniform binary random variables. Let G be the 2×2 polarization matrix and $G^{\otimes i}$ denote the i-th Kronecker power of G.

Let $W : \{0, 1\} \to \mathscr{Y}$ be a given B-DMC. The two-dimensional polar transformation is defined as follows: apply the $M \times M$ polar transformation matrix $G^{\otimes m}$ to the columns of V and then the $N \times N$ polar transformation matrix $G^{\otimes n}$ to the rows of the resulted matrix to get the $M \times N$ binary matrix X. More precisely,

$$X = (G^{\otimes m})^T U G^{\otimes n}$$

Then all the entries of X are transmitted through MN independent copies of W. The received word is denoted by $Y \in \mathscr{Y}^{M\times N}$. The transformation from U to Y is called the two-dimensional polar transformation and is denoted by $\mathscr{P}^{M\times N}$.

Lemma 1: The two dimensional polar transformation $\mathscr{P}^{M\times N}$, described above, is equivalent to the one-dimensional polar transformation $\mathscr{P}^{MN}$.

The two-dimensional channel polarization can be interpreted as two steps of polarization that are performed consecutively: the main channel W is polarized n levels in each row, then each of the polarized bit-channels is further polarized m more levels along the column. Therefore, roughly speaking we may observe this as $m + n$ levels of polarization.

B. Two-dimensional successive decoding

In this subsection, we discuss the two-dimensional successive cancellation decoding for the two-dimensional polar codes proposed in the previous subsection. Consider a polar code constructed with respect to the $M \times N$ polar transformation $\mathscr{P}^{M\times N}$, with $M \times N$ input U and $M \times N$ output Y. Let V be the matrix after multiplying columns of U by $G^{\otimes m}$ i.e.

$$V = (G^{\otimes m})^T U$$

For each of the M rows, a successive cancellation decoding is invoked for a rate 1 polar code of length N. However, it does not make the hard decision. The hard decisions are made by column decoders. For a fixed $j \leqslant N$, suppose that all the columns indexed by $1, 2, \ldots, j - 1$ are decoded. Each of the row decoders generate soft information LLR (log likelihood ratio) of their j-th bit. These calculations are done independently in parallel. Then these LLRs are passed to j-th column decoder which is another successive cancellation decoder for the polar code of length M. The j-th column decoder knows the positions of good bit-channels in the j-th column of U. It generates the hard decisions vector which is multiplied by $G^{\otimes m}$ to generate the hard decisions for j-th vector of V and is passed to row decoders. The row decoders use these hard decisions to compute the next LLRs.

Consider the individual bit-channels observed through the two-dimensional successive decoder mentioned above. Then the next theorem follows:

Theorem 2: There is a one-to-one mapping between the bit-channels observed through $\mathscr{P}^{M\times N}$ and the bit-channels observed through $\mathscr{P}^{MN}$.

IV. Multi-dimensional polar code

In this section, we generalize the results of the previous section to investigate higher dimensional realizations of polar codes. In particular, for a polar code of length $N = 2^n$, we investigate its n-dimensional realization. We show that both the latency and the space complexity of the proposed decoding architecture, based on n-dimensional polar codes, is $O(N)$.

A. n-dimensional polar transformation and encoding

As a generalization to two dimensional polar transformations, discussed in the previous section, an n-dimensional polar transformation $\mathscr{P}^{\overbrace{2\times 2\times\cdots\times 2}^{n \text{ times}}}$ is considered. The input to this transformation is the n-dimensional matrix

$$U = [U_{i_1,i_2,\ldots,i_n}]_{2\times\cdots\times 2} \in \mathbb{F}_2^{\overbrace{2\times 2\times\cdots\times 2}^{n \text{ times}}}$$

wherein all the N coordinates are independent and uniformly distributed. In the first level, the polarization matrix G is multiplied by all the vectors of the form $(U_{0,i_2,\ldots,i_n}, U_{1,i_2,\ldots,i_n})$. In the second level, the vectors of the form $(U_{i_1,0,\ldots,i_n}, U_{i_1,1,\ldots,i_n})$ are multiplied by G. This is done with respect to all the n coordinates in a consecutive order. The resulting matrix X is transmitted through N independent copies of a B-DMC W and the received vector is denoted by Y

Lemma 3: The n-dimensional polar transformation $\mathscr{P}^{\overbrace{2\times 2\times\cdots\times 2}^{n \text{ times}}}$, is equivalent to the one-dimensional polar transformation $\mathscr{P}^N$.

The encoding process can be also explained by a binary tree. For a polar code of length $N = 2^n$, the depth of the binary tree is n. Each internal node has a left child and a right child and there are N leaves in total. Let u_1^N denote the vector to be encoded, wherein u_i is an information bit if the i-th bit-channel is good and otherwise, is set to zero. The binary bits $(u_0, u_2, \ldots, u_{N-1})$ are assigned to the leaves of the tree from left to right. In fact, a vector of length 2^{n-i} is associated to a node at depth or level i of the tree. Consider a node at the i-th level of tree and call it c. Let $u_j^{j+2^{n-i}-1}$ be the sub-vector assigned to the leaves of the tree that are grandsons of c. Then the vector assigned to c is $u_{j+1}^{j+2^{n-i}-1}G^{\otimes(n-i)}$. The encoding can be done in n steps:

- Initialize the tree by assigning u_0^{N-1} to the leaves of tree, from left to right, respectively.
- For $i = 1, 2, \ldots, n$, at time instance i, each node at level $n-i$ of the tree gets two vectors u_L and u_R from its left and right child, respectively. Then it computes $(u_L + u_R, u_R)$ and returns it to its parent in the next time instance.
- At time instance n, the vector computed at the root of the tree is the encoded vector.

This encoding structure is shown for a polar code of length $8 = 2^3$ in Figure 1.

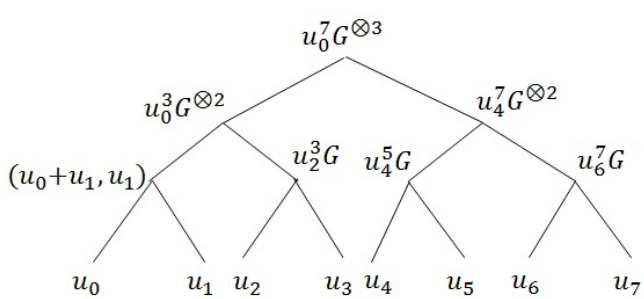

Fig. 1. The binary tree representation of 3-dimensional polar encoder

B. Proposed successive decoding of polar codes based on n-dimensional realization

A new structure for successive cancellation decoding of polar codes based on their multi-dimensional realization is discussed in this subsection. The new decoding structure results in efficient decoding latency and space complexity.

For ease of notation, for any two real numbers a and b, let

$$a \ast b \stackrel{\text{def}}{=} \frac{ab+1}{a+b}$$

This operation can be interpreted as follows. Assuming that a and b are the likelihood ratios of two independent binary random variables X and Y e.g.

$$a = \frac{P(X=0)}{P(X=1)} \quad \text{and} \quad b = \frac{P(Y=0)}{P(Y=1)}$$

Then $a \ast b$ is the likelihood ratio for $X \oplus Y$. The definition of this operation is also extended to vectors of the same size, in which case the operation is applied in a component-wise manner.

The simplest way to describe the proposed decoding structure is via binary tree. This is shown in Figure 2. The proposed polar decoder is initialized at the root of the tree, where the input is the soft channel output. The other input to the decoder is the indicator vector G for the good bit-channels. If the i-th bit-channel is good, then $G(i)$ is 1 and otherwise, $G(i)$ is zero.

The procedure for passing the hard information to the top of the tree is similar to that of the encoding process discussed in the previous subsection. At the end of decoding, each node at depth i of the tree, will have a hard information as a vector of length 2^{n-i+1}. The hard information calculation across the tree is done by the following rules:

- If a leaf corresponds to an information bit, then its hard decision is made as soon as its LLR is calculated (explained in the soft information calculation), otherwise its hard decision is equal to its frozen value.
- For any internal node, its hard information is calculated once the decoding process for both of its children is finished. Let vectors H_L and H_R be the hard infromation calculated at its left and right children, respectively. Then the hard information of the parent is the vector $(H_L + H_R, H_R)$.

The computation of soft information for the left and right child follows the same rule as in simple 2×2 polarization block, extended to a vector of likelihood ratios (LRs). Each node at depth i of the tree will have a vector of length 2^{n-i+1} of LRs.

- The LR of the root of tree is a vector of length N, provided by the N channel observations.
- Let LR of an internal node be $L = (L_1, L_2)$, where the length of L_1 and L_2 is half of the length of L. Then the LR of the left child is $L_L = L_1 \ast L_2$.
- The LR of a right child L_R is calculated after the decoding of its left sibling is finished. Given the LR of its parent $L = (L_1, L_2)$ and the hard information vector H_L calculated at its left sibling, which has the same length as L_1 and L_2,

$$L_R = L_1^{1-2H_L} L_2$$

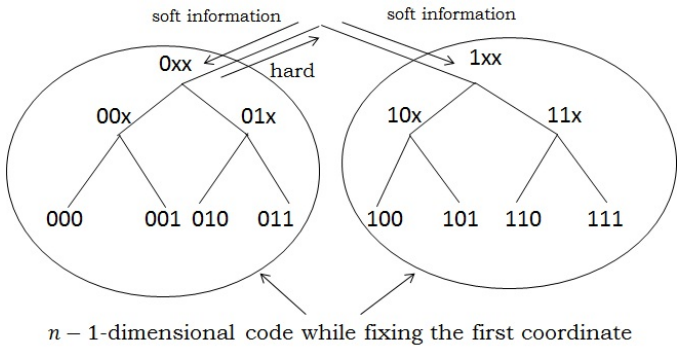

Fig. 2. The proposed structure for SC decoding

Theorem 4: The decoding latency and space complexity of the proposed n-dimensional polar decoder is $O(N)$, where $N = 2^n$ is the length of polar code.

V. Soft successive decoding

In our proposed n-dimensional tree structure for polar decoder, the soft information generated by the channel outputs are passed down the tree, while the hard decisions are passed up in the tree. In this section, we describe how to run a soft successive decoder, wherein the soft information are passed up in the tree instead of the hard information.

A. Soft information combining

The idea is to combine the soft information at all the internal nodes to pass to the parents. These soft information are called lower-LR (or lower-LLR in the log domain) and are denoted by H, whereas the LRs computed recursively from the top of the tree will be called upper-LR and are denote by L.

At the beginning, the set of upper-LRs are only non-zero at the root of the tree which are given by the channel output. The set of lower-LRs at the leaves of the tree that correspond to bit-channels are initialized as follows: for the leaves that are frozen to zero, the lower-LR is set to $+\infty$. Otherwise, the bit-channel carries an information bit for which we do not have any a priori information. Therefore, we set the lower-LRs of the information bit-channel to 1. The lower-LRs of all the other nodes in the tree is also set to 1. The equations for updating the lower-LRs and upper-LRs in the tree are discussed next.

Consider a node V in the decoding tree, at depth i (the depth of the root is 0 and the depth of the leaves is n). Then a vector L of length 2^{n-i} of upper-LRs is passed to V from its parent. Let $L = (L_1, L_2)$, where L_1 and L_2 have the same size 2^{n-i-1}. Let also H_L and H_R be the a priori lower-LRs of the children of V. The length of H_L and H_R is also 2^{n-i-1}. Let V_L denote the left child of V and V_R denote the right child of V. Then the upper-LR L_L of V_L is calculated as

$$L_L = L_1 * (L_2 H_R) \tag{5}$$

where the product of vectors is considered as component-wise product. The vector L_L is passed to V_L and the soft successive decoder is called for the node V_L and its subtree with input L_L. Let H'_L be the computed vector of lower-LRs at V_L. In fact, H'_L replaces H_L as the a priori lower-LR of V_L. Then the upper-LR L_R of the right child V_R is computed as

$$L_R = L_2(L_1 * H'_L) \tag{6}$$

Then the vector L_R is passed to V_R and the soft successive decoder is called again for V_R with input L_R. Let H'_R denote the computed vector of lower-LRs at V_R at the end of the soft successive decoder for V_R. H'_R replaces H_R as the set of a priori lower-LRs of V_R.

At the last step of the call to soft successive decoder for node V, the set of lower-LRs of V needs to be updated. The vector of lower LRs H of the node V is computed as follows:

$$H = \left[H'_L * (L_2 H'_R), H'_R(H'_L * L_1)\right]$$

In the next iteration, the soft successive decoding with the given channel output is run again. However, the difference is that the already computed lower-LLRs of all nodes in the decoding tree are stored from the previous iteration. They can help to boost the performance of the soft successive decoder.

The performance of the proposed soft successive decoding of polar codes is simulated to compare with Arıkan's hard SC decoding. The block length is $N = 1024$ and the rate of the code is 0.5. The soft successive decoding is performed with 8 iterations ($i = 8$). As shown in Figure 3, soft successive decoding slightly improves the performance of polar codes, in terms of the frame error rate FER, upon hard SC decoding by 0.1dB SNR. The bit error rates are also compared where the improvement of soft successive decoding is somewhat larger. This can be interpreted as follows. In soft successive decoding, the bit errors are less effective to the proceeding ones which reduce the error propagation through the whole block, comparing to hard SC decoder.

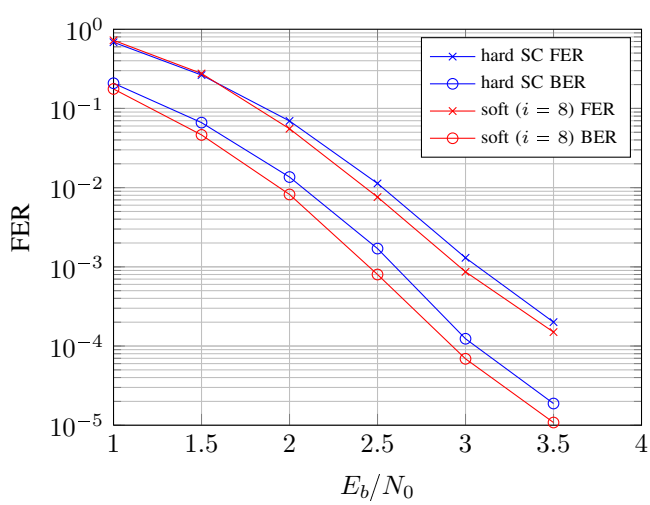

Fig. 3. Performance of soft successive decoding of polar codes

B. Soft successive decoding with iterative detection over BICM channels

In a previous work, we proposed capacity achieving polar-based schemes for transmission over multi-channels, called compound polar codes [14]. Compound polar codes [14] are proposed for reliable communication over the general model of multi-channels, where the encoded bits are transmitted across a certain set of binary-input channels. Bit interleaved code modulation (BICM) channels are good examples of multi-channels. For instance, BICM with 16-QAM can be modeled as a multi-channel with two constituent channels, called a 2-multi-channel and BICM with 64-QAM can be modeled as a multi-channel with three constituent channels, called a 3-multi-channel.

We have extended the proposed n-dimensional decoding and the soft successive decoding to compound polar codes. The iterative detection and decoding (IDD) can be carried as follows. In each iteration, the soft information in the root of the tree can be used as a priori LLRs for demodulating BICM symbols. This is simulated assuming the same parameters in the previous subsection. The simulation results show about 0.1 dB improvement.

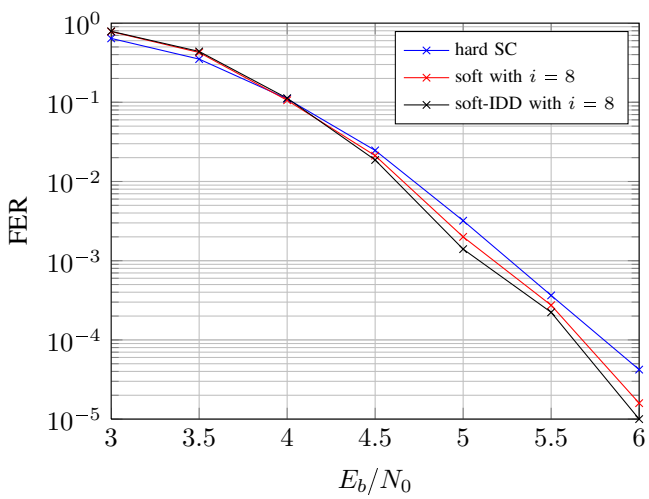

Fig. 4. Soft decoding and IDD for 16-QAM BICM

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, July 2009.

[2] H. Mahdavifar and A. Vardy, "Achieving the secrecy capacity of wiretap channels using polar codes," *IEEE Trans. Inform. Theory*, vol. 57, no. 10, pp. 6428–6443, October 2011.

[3] E. Arikan "Source polarization," in *Proc. IEEE Intern. Symp. on Information Theory*, pp. 899–903, Austin, TX., June 2010.

[4] E. Abbe, "Extracting randomness and dependencies via a matrix polarization," arXiv:1102.1247v1, 2011.

[5] E. Şaşoğlu,E. Telatar, and Edmund Yeh, "Polar codes for the two-user binary-input multiple-access channels," *Proc. 2010 Information Theory Workshop (ITW)*, pp. 1–5, Dublin, Ireland, 2010.

[6] E. Abbe, E. Telatar, "Polar codes for the m-user multiple access channel," *IEEE Trans. Inform. Theory*, vol. 58, no. 8, pp. 5437–5448, August 2012.

[7] H. Mahdavifar, M. El-Khamy, J. Lee and I. Kang, "Achieving the uniform rate region of multiple access channels using polar codes," *submitted to IEEE Trans. Inform. Theory*, available online at http://arxiv.org/abs/1307.2889

[8] N. Goela, E. Abbe, M. Gastpar, " Polar codes for broadcast channels," *Proc. IEEE Intern. Symp. Information Theory*, pp. 1127-1131, Istanbul, Turkey, July 2013.

[9] M. Mondelli, S. H. Hassani, I. Sason, R. Urbanke, "Achieving the superposition and binning regions for broadcast channels using polar codes," arxiv preprint http://arxiv.org/abs/1401.6060

[10] I. Tal and A. Vardy, "List decoding of polar codes," *Proc. IEEE Intern. Symp. Information Theory*, Saint Petersburg, Russia, 2011.

[11] P. Trifonov, "Efficient design and decoding of polar codes," *IEEE Trans. Comm.*, vol. 60, no. 11, pp. 3221-3227, November 2012.

[12] A. Eslami, H. Pishro-Nik "On finite-length performance of polar codes: stopping sets, error floor, and concatenated design", *IEEE Trans. Commun.*, vol. 61, no. 3, pp. 919-929, March 2013.

[13] H. Mahdavifar, M. El-Khamy, J. Lee and I. Kang, "On the construction and decoding of concatenated polar codes," *Proc. IEEE Intern. Symp. Information Theory*, pp. 952-956, Istanbul, Turkey, July 2013.

[14] H. Mahdavifar, M. El-Khamy, J. Lee and I. Kang, "Compound polar codes," *Proc. of Information Theory and Applications Workshop (ITA)*, pp. 1–6, San Diego, California, February 2013.