On the Concatenations of Polar Codes and Non-binary LDPC Codes

Qingping Yu, Zhiping Shi, *Member, IEEE,* Xingwang Li, *Member, IEEE,* Jianhe Du, Jiayi Zhang, *Member, IEEE,* and Khaled M. Rabie, *Member, IEEE,*

***Abstract*—An interleaved concatenation scheme of polar codes with non-binary low-density parity check (NBLDPC) codes is proposed in this paper to improve the error-correcting performance of polar codes with finite code length. The information blocks of inner polar codes are split into several information sub-blocks, and several segment successive cancellation list (S-SCL) decoders are carried out in parallel for all inner polar codes. Moreover, for a better error-correcting performance, an improved SCL decoder with a selective extension is proposed for the concatenated polar codes, which will be referred to selective extended segment SCL (SES-SCL) decoder. The SES-SCL decoder uses soft information of some unreliable information sub-blocks for the decoding of subsequent sub-blocks so as to mitigate the error propagation of premature hard decision of S-SCL decoder. Simulation results show that NBLDPC-polar codes can outperform Reed Solomon (RS)-polar codes. NBLDPC-polar codes when decoded with the proposed SES-SCL algorithm can also be comparable to pure polar codes with cyclic redundancy check aided successive cancellation list (CA-SCL) decoding with list size $L = 4$ in the high SNR, but require lower decoding storage. Therefore, NBLDPC-polar codes may strike a better balance between memory space and performance compared to the state-of-art schemes in the finite length regime.**

***Index Terms*—Polar codes, code concatenation, non-binary low-density parity-check (NBLDPC) code, successive cancellation list (SCL) decoding.**

I. Introduction

POLAR codes, proposed by Arıkan [1], have been selected for the 5th generation (5G) wireless communication standard [2], due to their inherent capacity-achieving property over any binary memoryless symmetric (BMS) channel and low-complexity encoding and decoding. Polar codes with successive cancellation (SC) decoding can achieve Shannon capacity with $O(N \log N)$ complexity, where N is the block-length [1]. However, in practical applications, the error rate performance of polar codes with finite length is unsatisfactory under the SC decoder. To improve the error-correcting performance of finite length polar codes, authors of [3] and [4] have proposed the SC list (SCL) decoder and cyclic redundancy check aided SCL (CA-SCL) decoder, respectively, which approach the performance of maximum-likelihood (ML) decoding at the cost of higher memory requirement and decoding complexity.

Another way to improve the performance of finite length polar codes is to concatenate such codes with other codes. For instance, in [5] concatenated polar codes with outer Reed Solomon (RS) codes are proposed and have shown the ability to reduce error rates nearly exponentially with the code length. However, the cardinality of the RS code increases exponentially with the length of the polar code. To solve this problem and further improve the error correction performance of polar codes, concatenated polar codes with interleaved outer RS codes is proposed in [6] and an improved RS-polar concatenation is proposed in [7]. Besides, the authors of [8] report a concatenation scheme of concatenating convolutional codes with polar codes, which sacrifices decoding complexity for archiving a better performance. Low-density parity-check (LDPC) codes, proposed by Gallager in [9], are block codes with sparse parity check matrix. Such sparsity allows for low complexity belief propagation (BP) decoding algorithm. In [10], Davey and MacKay introduced the nonbinary LDPC (NBLDPC) code and proved that the NBLDPC code has a significant performance improvement compared with the binary LDPC code. In [11], a serial concatenation of polar codes with LDPC codes is discussed, and it is shown that the outer polar codes can remove the error floor of the inner LDPC codes. In [12] and [13], LDPC codes are used to protect intermediate bit-channels of polar codes to get a better performance.

In [14], the concatenation of polar codes and NBLDPC codes (referred to as NBLDPC-polar) is proposed and a segment SCL (S-SCL) decoder is first introduced for the NBLDPC-polar code. However, the performance of NBLDPC-polar codes under the S-SCL decoder is sub-optimal. Moreover, the rate optimization scheme and space requirement analysis are omitted in [14] because of the space limit. In this paper, we extend the results of [14] to propose an improved decoding scheme for the concatenated NBLDPC-polar codes for a better performance. The main contributions of this paper are summarized in the following.

1) NBLDPC-polar concatenated code and the corresponding rate optimization method are discussed. In this proposed concatenation scheme, NBLDPC codes, instead of binary LDPC codes, are chosen as outer codes because at short and medium block lengths, the former offers a better performance [15]. In addition, the non-

Q. Yu and Z. Shi are with the National Key Laboratory of Science and Technology on Communications, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: qpyu@std.uestc.edu.cn; szp@uestc.edu.cn).

X. Li is with the School of Physics and Electronic Information Engineering, Henan Polytechnic University, Jiaozuo, China (email:lixingwangbupt@gmail.com).

J. Du is with the School of Information and Communication Engineering, Communication University of China, Beijing 100024, China (email:dujianhe1@gmail.com).

J. Zhang is the School of Electronics and Information Engineering, Beijing Jiaotong University, Beijing 100044, China (email:zhangjiayi@bjtu.edu.cn).

K. M. Rabie is with the School of Electrical Engineering, Manchester Metropolitan University, Manchester, U.K. (e-mail: K.Rabie@mmu.ac.uk)

binary outer code has the potential to match better with multibit-decision approach which can significantly reduce latency of traditional SCL decoders [16, 17].

2) A new decoding scheme, named selective extended segment SCL (SES-SCL), is proposed for the concatenated polar codes. The proposed SES-SCL decoding algorithm takes adequate consideration of the actual decoding result of previous information sub-blocks in the decoding of subsequent sub-blocks. It uses the parity check matrix to check whether the outer NBLDPC codeword is correctly decoded, thus results a reasonable criterion to judge whether the hard-decision will be made for the corresponding information sub-block. If no valid NBLDPC codeword is detected, symbol probabilities of NBLDPC codeword will be saved to associate the following decoding. This decoding method avoids making final decisions too early for the unreliable information sub-blocks and thus mitigates the error propagation problem. Simulation results verify that the SES-SCL scheme achieves about 0.4dB gain with code length 8192 and code rate $1/3$ at BER 10^{-5} compared to the S-SCL decoding.
3) A comprehensive complexity comparison and analyses in terms of storage and complexity are provided. Simulation results show that NBLDPC-polar codes with the proposed SES-SCL decoding substantially outperform pure polar codes with SC or BP decoding. Besides, NBLDPC-polar codes with SES-SCL decoding perform as well as pure polar codes with CA-SCL decoding in the high SNR regime but with lower memory requirement. Therefore, it can be concluded that, using the proposed SES-SCL decoding for concatenated polar codes may have a better balance compared to the state of art methods.
4) To further explore the advantage of the selective extension method in the decoding of concatenated polar codes, a short RS code is used in place of the NBLDPC code, which again shows a performance gain.

The rest of the paper is organized as follows. In Section II, we review polar codes and the main decoding methods. In Section III, NBLDPC-polar codes are discussed and a rate optimization scheme is provided. Section IV proposes a selective extended segment SCL decoding for NBLDPC-polar codes. In Section V, space requirements and decoding complexity are discussed. Simulation results are presented and discussed in Section VI followed by concluding remarks in Section VII.

II. Preliminaries

In this section, some necessary background on the polar codes and its associated decoding schemes are presented.

A. Polar codes

Polar codes are constructed based on channel combining and splitting process [1]. Let $W : X \to Y$, Y denote an arbitrary binary input discrete memoryless channel (BDMC) with the input alphabet $X = \{0, 1\}$ and the output alphabet Y. Channel combining means transforming the N copies of W into a vector channel $W_N : X^N \to Y^N$ and channel splitting means splitting the vector channel W_N into N binary-input coordinate channels $\{W_N^{(i)}, i = 1, 2, \ldots, N\}$. The channel polarization theorem states that as N goes to infinity, the bit-channels start polarizing, meaning that one part of bit-channels become completely noisy ones, while the other part become completely noiseless ones.

Let $\mathbf{F}^{\otimes n_0}$ denote a $N \times N$ matrix, where $n_0 = \log_2 N$, $\otimes n_0$ denotes n_0-th Kronecker power, $\mathbf{F} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$ and $\mathbf{F}^{\otimes n_0} = \mathbf{F} \otimes \mathbf{F}^{\otimes (n_0-1)}$. Let $\mathbf{B}_N$ denote the bit-reversal permutation matrix, then the $n_0 \times n_0$ polarization transform matrix is denoted as $\mathbf{G}_N = \mathbf{B}_N \mathbf{F}^{\otimes n_0}$. Hence, the polar code can be encoded by $\mathbf{x}_1^N = \mathbf{u}_1^N \mathbf{G}_N$, where $\mathbf{u}_1^N = (u_1, u_2, \ldots, u_N)$ and $\mathbf{x}_1^N = (x_1, x_2, \ldots, x_N)$ denote the information bit sequence and the encoded bit sequence, respectively. The bit sequence of $\mathbf{u}_1^N$ can be divided into two subsets according to the channel polarization effects, one subset is presented as A carrying all information bits and the other subset is presented as A^c carrying all frozen bits.

B. Decoding of Polar Codes

SC decoder is the first decoder used to decode polar codes [1]. SC decoder provides each bit estimate based on the received $\mathbf{y}_1^N$, the previously estimated bits $\hat{\mathbf{u}}_1^{i-1}$, and the location of information bits A. The formulation is

$$\hat{u}_i = \begin{cases} h_i(\mathbf{y}_1^N, \hat{\mathbf{u}}_1^{i-1}) & i \in A; \\ u_i & i \in A^c, \end{cases} \tag{1}$$

where

$$h_i(\mathbf{y}_1^N, \hat{\mathbf{u}}_1^{i-1}) = \begin{cases} 0, & if\ \frac{W_N^{(i)}(\mathbf{y}_1^N, \hat{\mathbf{u}}_1^{i-1}|u_i=0)}{W_N^{(i)}(\mathbf{y}_1^N, \hat{\mathbf{u}}_1^{i-1}|u_i=1)} \geq 1; \\ 1, & otherwise. \end{cases} \tag{2}$$

$W_N^{(i)}(\mathbf{y}_1^N, \hat{\mathbf{u}}_1^{i-1}|u_i)$ represents the likelihood of u_i given the channel output $\mathbf{y}$ and $\mathbf{u}_1^{i-1}$ considering u_{i+1}, u_{i+2}, . . . , u_N as unknown bits. To calculate $W_N^{(i)}(\mathbf{y}_1^N, \hat{\mathbf{u}}_1^{i-1}|u_i)$, Arıkan's recursive channel transformation [1] is used. A pair of binary channels $W_{2\Lambda}^{(2i-1)}$ and $W_{2\Lambda}^{(2i)}$ are calculated by a one-step transformation of two independent copies of a binary input channel $W_\Lambda^{(i)} : (W_\Lambda^{(i)}, W_\Lambda^{(i)}) \to (W_{2\Lambda}^{(2i-1)}, W_{2\Lambda}^{(2i)})$. The channel transition probabilities are calculated as

$$\begin{aligned} W_{2\Lambda}^{(2i-1)}&(\mathbf{y}_1^{2\Lambda}, \mathbf{u}_1^{2i-2}|u_{2i-1}) \\ &= \frac{1}{2}\sum[W_\Lambda^{(i)}(\mathbf{y}_1^{\Lambda}, \mathbf{u}_{1,o}^{2i-2} \oplus \mathbf{u}_{1,e}^{2i-2}|u_{2i-1} \oplus u_{2i}) \\ &\times W_\Lambda^{(i)}(\mathbf{y}_{\Lambda+1}^{2\Lambda}, \mathbf{u}_{1,e}^{2i-2}|u_{2i})], \end{aligned} \tag{3}$$

$$\begin{aligned} W_{2\Lambda}^{(2i)}&(\mathbf{y}_1^{2\Lambda}, \mathbf{u}_1^{2i-1}|u_{2i}) \\ &= \frac{1}{2}W_\Lambda^{(i)}(\mathbf{y}_1^{\Lambda}, \mathbf{u}_{1,o}^{2i-2} \oplus \mathbf{u}_{1,e}^{2i-2}|u_{2i-1} \oplus u_{2i}) \\ &\times W_\Lambda^{(i)}(\mathbf{y}_{\Lambda+1}^{2\Lambda}, \mathbf{u}_{1,e}^{2i-2}|u_{2i}), \end{aligned} \tag{4}$$

where $0 < i \leq \Lambda = 2^\lambda < N$ and $0 \leq \lambda < n0$.

The decoding process of polar codes can be graphically described as a search problem in the decoding tree. As an

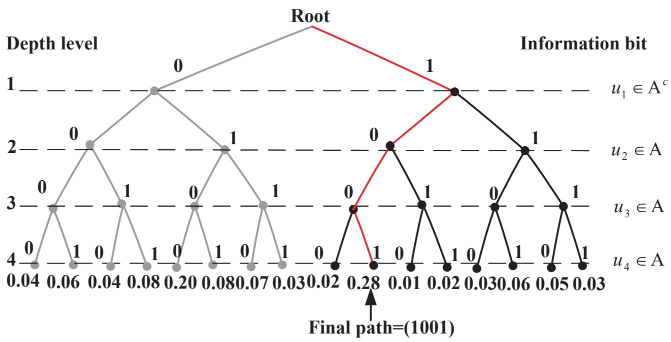

Fig. 1: Decoding process of SC decoder over a code tree.

example, the decoding tree for a polar code length of 4 is shown in Fig. 1, in which the root node represents a null state, the nodes at depth i represent u_i, and the left and right children nodes at depth i represent $u_i = 0$ and $u_i = 1$, respectively. Therefore, the set of all codewords length of N can be viewed as a perfect binary tree with a depth-N, and a particular codeword block is just a path from the root node to the corresponding leaf node. Consequently, the task of decoding is just to, given the received polar code block $\mathbf{y}_1^N$ from the channel, find the original path of the code block so that $\hat{\mathbf{u}}_1^N$, an estimate of $\mathbf{u}_1^N$, can be obtained. Besides, authors of [18] propose a method, which is based on adding extra computations nodes to the SC code tree, to enhance the performance of the SC decoder.

Instead of making a hard decision for each information bit in the SC algorithm, the SCL algorithm [3] creates two paths, in which the information bit is assumed to be 0 and 1, respectively. Therefore, for each information bit, the decoder doubles the number of possible paths up to a predetermined limit L. When the number of paths exceeds L, decoder just keeps L best paths as survival paths while discards the others. At the end of the SCL decoding procedure, the most likely path among the L paths is chosen as the single codeword at the decoder output. Moreover, in [4] a CRC is used as the primary criterion for the path selection, the decoder outputs the most likely path passing the CRC detection as the estimation sequence. If there is no path passing the CRC check, the most reliable path will be selected as the target path.

III. NBLDPC-Polar Codes

In this section, a new class of concatenation schemes, NBLDPC-polar codes, is proposed. We first introduce the concatenation structure and then discuss the rate optimization of outer NBLDPC codes.

A. System Model of NBLDPC Codes

A scheme of concatenating inner polar codes with regular NBLDPC codes is presented in Fig. 2 (a). In the scheme, polar codes are used as inner codes and regular NBLDPC codes as outer codes. It is assumed that the channel input sequences are binary. Data sequences are encoded as non-binary LDPC codewords over $GF(2^t)$, then LDPC code symbols with the same index are converted into binary streams to form one

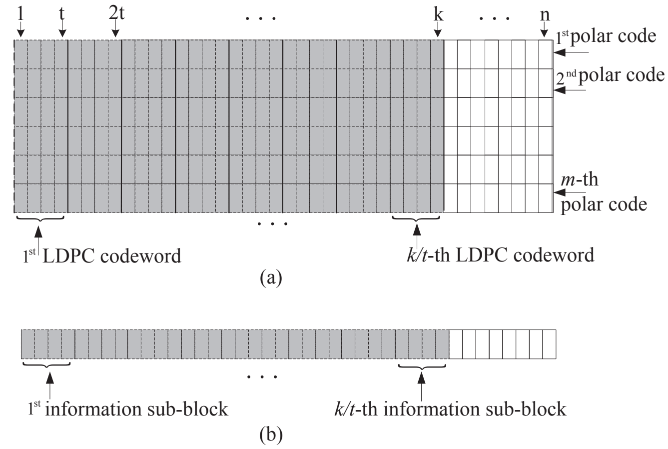

Fig. 2: (a) Encoding structure of concatenate NBLDPC-polar codes, (b) Structure of the inner polar code.

input sequence for the (n, k) polar encoder. Here we assume that all the NBLDPC codewords used are independent. The gray and white parts of each row in Fig. 2 correspond to k information bits and $n-k$ frozen bits of the inner polar code, it is assumed that k is divisible by t. It is shown in Fig. 2 (b) that there are total k/t information sub-blocks in the information block of each polar code. Let m and n denote code lengths of the NBLDPC code over $GF(2^t)$ and the binary polar code, respectively. Assuming the rate of the j-th LDPC code over $GF(2^t)$ is Ro_j, then the overall number of input symbols to outer NBLDPC encoders is $\sum_{j=1}^{k/t} Ro_j m$, the overall number of input bits for the concatenation is $\sum_{j=1}^{k/t} Ro_j mt$. The overall block length of the concatenated code is $N = mn$. Therefore, the overall rate of concatenated code is

$$R = \sum_{j=1}^{k/t} Ro_j mt/N = \sum_{j=1}^{k/t} Ro_j t/n. \tag{5}$$

B. Rate Optimization for Outer NBLDPC Codes

Due to the channel polarization of polar codes, not all bit-channels carrying the information bits have the same reliability. Correspondingly, not all information sub-blocks consisting of different bit channels have the same reliability. Therefore, the rate of each NBLDPC code could be properly assigned to protect the information sub-blocks in such a way that all the information sub-blocks are almost equally protected [6].

Generally, for $j = 1, 2, ..., k/t$, the probability that an information sub-block (or a symbol of the j-th NBLDPC code) is in error, assuming that all previous information sub-blocks (or symbols of previous NBLDPC codewords) were successfully recovered, is denoted as

$$P_{s_j} = 1 - (1 - \pi_{jt-t+1})(1 - \pi_{jt-t+2})...(1 - \pi_{jt}), \tag{6}$$

where, π_{jt} is the probability that an error occurs when decoding the jt-th information bit of an (n, k) inner polar code with SC decoder, assuming that all the first $jt - 1$ information bits were successfully recovered. The block error rate P_{o_j} of the each outer NBLDPC code depends on the channel condition

P_{s_j} and the outer code rate R_{o_j}. Authors in [19] present an efficient method for predicting the performance of finite-length LDPC codes. Here we get the actual error rate performance of the outer NBLDPC codes based on Monte Carlo simulations.

Let P_e represent the frame error rate of the concatenated code, then P_e can be computed as

$$P_e = 1 - \prod_{j=1}^{k/t}(1 - P_{o_j}). \tag{7}$$

Assuming we want to design codes with a fixed overall rate $R^\star$, then the rate of NBLDPC code R_{o_j} can be optimized as follows to achieve the minimum frame error rate.[1]

1) Compute π_j for each information bit of the inner polar code based on density evolution, then compute P_{s_j} for each information sub-block.
2) Set $p1 = 1$, $p2 = 0$.
3) If $p1 - p2 > \epsilon$, set the target block error rate of each outer code is $P^\star = \frac{p1+p2}{2}$, otherwise go to 6).
4) For each sub-block channel, find the maximum R_{o_j} that satisfies $P_{o_j} \leq P^\star$.
5) Compute k as the number of outer codes with $R_{o_j} > 0$. Compute R based on (5). If $R < R^\star$, set $p2 = P^\star$, compute P_{o_j} and then compute P_e based on (7). If $R \geq R^\star$, set $p1 = P^\star$, return to 3).
6) Find out the minimum value $P_{e_{min}}$ among all P_e as the minimum achievable frame error rate (FER) of the code, and output the corresponding $R_{o_j}, j = 1, 2, ..., k/t$ as the optimized rate of the outer NBLDPC codes.

IV. Proposed Decoding Methods for the NBLDPC-polar Concatenated Code

In this section, two decoding techniques are proposed to improve the performance of finite length polar codes. One is S-SCL decoding, in which the decoding process of inner polar codes is divided into several segments, and a SCL decoding is used for each segment. A hard decision is made for each segment and then sent to the next segment for further decoding. The other one is SES-SCL decoding, by checking whether a valid NBLDPC codeword is detected or not, a selection method is set to judge whether a hard decision is made for the current segment. A BP decoding is applied for outer NBLDPC codes. In related works, the partitioned SCL (PSCL) algorithm [21] breaks the code tree into several sub-trees (partitions) and each partition is decoded with the SCL algorithm. PSCL decoder can significantly reduce the memory requirement by increasing the number of partitions. However, this memory saving of PSCL is obtained at the cost of a higher error probability. In [22], a CRC selection scheme is proposed to further improve the error correction performance of PSCL decoder. Furthermore, authors of [23] propose a generalized PSCL (GPSCL) algorithm and a layered PSCL (LPSCL) algorithm to bridge the performance gap between SCL and PSCL decoding. We note that, the proposed S-SCL decoding has a similar path extension for inner polar codes with the PSCL decoding. However, the path selections of our S-SCL and SES-SCL are different from those of [21–23].

A. Segment SCL Decoding for Inner Polar codes

As introduced in Section III part A, the information block of inner polar codes is split into k/t sub-blocks. Correspondingly, the decoding process of inner polar codes is divided into k/t segments, and a SCL decoding with list size $L = 2^t$ is used for each segment. Then with the help of the LDPC decoder, a single correct codeword is selected for each segment and then sent to the next segment for further decoding. Moreover, according to coding theory, the parity-check matrix H of LDPC code can be commonly used for codeword detection by [10]

$$\tilde{x}H^T = 0. \tag{8}$$

If the condition in (8) is satisfied, then $\tilde{x}$ is equivalent to the right codeword with high probability. Therefore, (8) can be used as an early stopping criterion to reduce decoding latency and complexity.

More precisely, for $j = 1, 2, ..., k/t$, let $W(y, \hat{u}_1^{(j-1)t}|u_{(j-1)t+1}^{jt})$ denote the path likelihood of the j-th sub-block.[2] For each consecutive t bits, polar decoder computes the likelihoods of all the 2^t paths, and then symbol probabilities of the outer NBLDPC codeword can be obtained by traversing all the possible 2^t paths [6]. Initialize the algorithm with $j = 1$, then the S-SCL decoding algorithm is described as follows.

A1) Do path extension for each of m inner polar codes, and compute path likelihoods $W(y, \hat{u}_1^{(j-1)t}|u_{(j-1)t+1}^{jt})$ of all 2^t paths. These operations can be done in parallel for all m polar codewords.

A2) Do path selection with early stopping. Regard the m most likely paths as m symbols over $GF(2^t)$ to form a hard-decisioned LDPC codeword x^*. If $x^* H_j^T = 0$ is satisfied, x^* is output and used to update the decoded bits of all m polar codewords, then go to A4. Else go to A3.

A3) Do path selection without early stopping. Pass the $m2^t$ paths and path likelihoods to the LDPC decoder as the input messages of the BP decoding. Do BP decoding and then update the decoded bits of all m polar codewords using the LDPC decoder output.

A4) If $j < k/t$, $j = j + 1$ and return to $A1$.

Fig. 3 shows the decoding procedure of the S-SCL algorithm with list size $L = 16$. When the decoding of segment-1 is finished, path 0101 in the first polar code tree is saved and is used to continue the decoding of the second segment. Finally, the path 0110 in blue color is saved for segment-2, the other 15 paths are discarded. Similarly operations can be done in parallel for all inner polar codewords. We note that, the log-likelihood ratio (LLR)-based SCL decoder [24, 25] can be also applied to our proposed S-SCL decoder. In that case, polar decoder calculates all the possible LLRs of all the 2^t paths

[1] Note that the rate optimization algorithm is a modified version of the algorithm in [20].

[2] Since all the polar codes used are identical, for sake of simplicity, here we don't identify the index of polar codes.

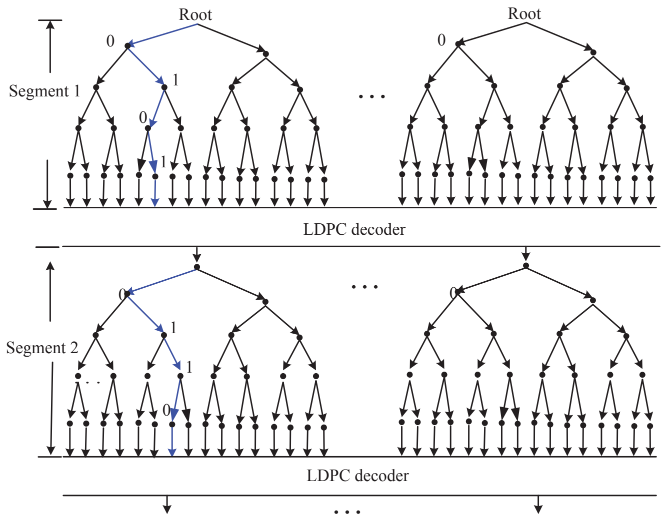

Fig. 3: Decoding process of S-SCL decoder for NBLDPC-polar codes.

for each segment, and passes path LLRs to the LDPC decoder as symbol LLRs. Correspondingly, a LLR-BP decoding [26] can be used for the outer NBLDPC codes.

B. Selective Extended Segment SCL Decoding

A main drawback of SC decoding for polar codes is the error propagation. The S-SCL decoder introduced above can potentially mitigate the error propagation problem of polar code because of two reasons, one is offering protection with outer codes, thus errors in the decoded sub-blocks may be corrected using the outer code, and the other one is searching optimal path within a larger search space [16]. However, as S-SCL decoder makes segment-level hard decision for each information sub-block, the segment-level error propagation still exists. Assuming that the LDPC decoder can not correct errors in the decoded sub-blocks, one might expect that the wrong chosen hard-decision sub-block will propagate the following decoding process. To address this issue, we propose a selective extended segment SCL (SES-SCL) decoding algorithm.

In a specific construction, a set $F_j, j = 1, 2, ..., k/t$ is used to hold the flags to indicate the decoding result of the j-th segment, so that $F_j = 0$ and $F_j = 1$ indicate successful decoding and failed decoding of segment j, respectively. Here we assume that errors incurred in the decoded bit sequence of the $j-1$-th segment can not be corrected by the outer LDPC decoder, and thus $F_{j-1} = 1$. Let X^q_{j-1} and Λ^q_{j-1}, where $q = 2^t$, $2 \leq j \leq k/t$, denote the estimated erroneous symbol sequence and symbol probabilities sequence of the $j-1$-th LDPC codeword, respectively. According to total probability Theorem, the path likelihood of the j-th segment can be calculated as

$$\tilde{W}(y, \hat{u}_1^{(j-1)t} | u_{(j-1)t+1}^{jt}) = \sum_{a=1}^{2^t} \Lambda^{s_a}_{(j-1)} W_{s_a}(y, \hat{u}_1^{(j-1)t} | u_{(j-1)t+1}^{jt}), \tag{9}$$

where $W_{s_a}(y, \hat{u}_1^{(j-1)t} | u_{(j-1)t+1}^{jt})$ denotes the path likelihood $W(y, u_1^{(j-1)t} | u_{(j-1)t+1}^{jt})$ with $\hat{u}_{(j-2)t+1}^{(j-1)t} = s_a$.

Although the calculation of the path likelihood in (9) is the most accurate, it's too complicated. To maintain a low decoding complexity, in each estimated message X^q_{j-1}, we reserve l, $1 < l < L$ symbols with the largest probabilities as selective survival paths. Let $X^{s_a}_{j-1}$ and $\Lambda^{s_a}_{j-1}$, where $s_a \in \{1, ..., q\}$ and $a = 1, 2, ..., l$, denote these selected symbols and symbol probabilities, respectively. LDPC decoder passes $X^{s_a}_{j-1}$ to the polar decoder, polar decoder then uses $X^{s_a}_{j-1}$ as survival paths for the decoding process of the next segment, and thus lL paths are generated during the decoding process for the j-th segment (see Fig. 4). Upon finishing the computation of all the path likelihoods for lL paths, we use (10) to optimize the path likelihood of segment j.

$$\tilde{W}(y, \hat{u}_1^{(j-1)t} | u_{(j-1)t+1}^{jt}) \approx \sum_{a=1}^{l} \Lambda^{s_a}_{j-1} W_{s_a}(y, \hat{u}_1^{(j-1)t} | u_{(j-1)t+1}^{jt}). \tag{10}$$

Select the most likely path as the hard-decisioned output bits, then mt hard-decisioned output sub-blocks are passed as m symbols over $GF(2^t)$ to form an LDPC codeword. Then a codeword detection corresponding to the j-th NBLDPC code is performed based on (8). If the condition in (8) is satisfied, end the current segment decoding. Otherwise a BP decoding is performed. Initialize $\{F_j\}_{j=1}^{k/t} = 0$, and $j = 1$. Then the SES-SCL decoding algorithm is described as follows.

- B1) If $j = 1$ or $j \geq 2$, $F_{j-1} = 0$, go to B2, otherwise go to B4.
- B2) Do path extension for all inner polar codes in parallel, which is the same as A1. Then do path selection with early stopping, the process is almost the same as A2 except that, "A4" and "A3" in step A2 are changed to "B7" and "B3", respectively.
- B3) Do path extension without early stopping, which is the same as A3. Set $F_j = 1$ if no valid NBLDPC codeword is detected. Then go to B7.
- B4) For each of the m polar codewords, do path expansion based on l selective survival paths and compute path likelihoods $W(y, \hat{u}_1^{(j-1)t} | u_{(j-1)t+1}^{jt})$ of all $l2^t$ paths. Optimize path likelihoods of segment j based on (10). These operations can be done in parallel for all m inner polar codewords.
- B5) Do path selection with early stopping. The process is almost the same as A2 except that "A4" and "A3" in step A2 are changed to "B7" and "B6", respectively.
- B6) Do path extension without early stopping which is the same as A3. Set $F_j = 1$ if no valid NBLDPC codeword is detected. Then go to B7.
- B7) If $j < k/t$, $j = j + 1$ and return to B1.

Fig. 4 shows the decoding process of SES-SCL algorithm with 2 segments, the list size is $L = 16$ and the selective survival paths number is $l = 2$. Here we assume that the codeword detection for segment-1 of the first polar code is failed (the condition in (8) is not satisfied with H_1 and $\hat{x}_1$), then two selective survival paths 0100 and 1110 are

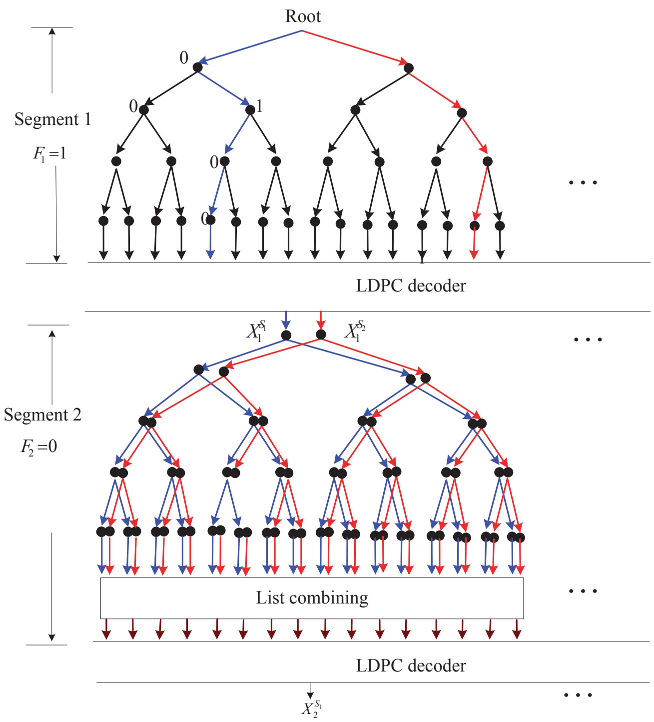

Fig. 4: Decoding process of SES-SCL decoder for NBLDPC-polar codes with list size $L = 16$ and selective survival paths number is $l = 2$.

saved for segment-1, with associated probabilities P_{0100} and P_{1110}, respectively. Correspondingly, in the path extension of the second segment, two groups of 16 candidate paths are generated. One group with blue color is computed based on path 0100, and the group with red color is computed based on path 1110. Calculate path likelihoods of the second segment based on (10), for example the path likelihood of 0101 is represented by $P(u_5^8 = 0101) \approx P_{0100} \times W(y, u_1^4 = 0100|u_5^{8t} = 0101) + P_{1110} \times W(y, u_1^4 = 1110|u_5^8 = 0101)$. For the sake of comparison, we assume that the decoding of the second segment of concatenated polar codes is successful and $F_2 = 0$, hence at the end of decoding for segment-2, one path $X_2^{S_1}$ is saved as the final decoded results and the other 15 paths are discarded. Similar operations are processed in parallel for other polar codes.

V. Analyses and Discussions

In this section, we give analyses and comparisons of required memory size and decoding complexity of NBLDPC-polar codes with S-SCL decoding and SES-SCL decoding.

A. Memory Space Analysis

SCL decoding requires a large memory to store the intermediate values, i.e. the high area requirement of SCL decoding is mostly dominated by its memory usage [25]. In S-SCL decoding, on the other hand, the decoding process is broken into segments and SCL decoding is performed on each segment. Each segment outputs a single path which is selected with the help of the LDPC decoder and then sent to the next segment for further decoding. Therefore, similar to the PSCL decoder in [21], it is possible to share the memory between segments since only one candidate survives after decoding each segment. It should be noted that, as all possible 2^t paths are selected for each segment, there is no need to store the path metric (PM) [25] for inner polar codes. Following the same reasoning for PSCL decoder, the memory requirement of each inner polar code length of n with S-SCL decoding is denoted as[3]

$$M_1 = \left(n + \sum_{r=1}^{P-1} \frac{n}{2^r} + \left(\frac{n}{2^{P-1}} - 1\right) L\right) Q_\alpha + \sum_{r=1}^{P-2} \frac{n}{2^r} + \left(\frac{n}{2^{P-2}} - 1\right) L, \tag{11}$$

where Q_α denotes the number of quantized bits for the path likelihood, $P = k/t$ denotes the number of segments and $2 \leq P < n$, the list size is $L = 2^t$. The space required by m polar codes with S-SCL decoding is mM_1. The outer LDPC codes are decoded in order, which enables us to reuse the memory. The memory requirement of NBLDPC codes over $GF(2^t)$ is $M_L = (cd_c + md_v)2^t Q_\alpha$ [27], where c denotes the number of check nodes. Therefore, the total memory requirement for NBLDPC-polar codes with S-SCL decoding is $mM_1 + M_L$. The memory requirements of SES-SCL decoder are higher than that of S-SCL decoder since l times of path likelihoods and hard-decision bits need to be stored for the segments with extension (see Fig. 5). The memory requirement of each inner polar code with SES-SCL decoding is denoted as

$$M_2 = \left(n + l\sum_{r=1}^{P-1} \frac{n}{2^r} + \left(\frac{n}{2^{P-1}} - 1\right) lL\right) Q_\alpha + l\sum_{r=1}^{P-2} \frac{n}{2^r} + \left(\frac{n}{2^{P-2}} - 1\right) lL. \tag{12}$$

The overall space required by NBLDPC-polar codes with SES-SCL decoding is $mM_2 + M_L$ units for SES-SCL decoding. Besides, we note that the memory requirements of S-SCL decoding is higher than that of the PSCL decoding. This is mainly because we use NBLDPC codes, other than CRC bits as the outer codes, thus more memories are required for the LDPC decoder. However, as the outer LDPC codes are very short block codes and $m \ll N$, the area overhead for extra memory requirement of NBLDPC decoder is negligible.

B. Decoding Complexity Analysis

Next we analyze the decoding complexity of NBLDPC-polar codes taking both polar codes and NBLDPC codes into consideration. For the S-SCL scheme, the decoding complexity of m polar codes is $m(Ln \log n) = LN \log n = 2^t N \log n$, and the decoding complexity of k/t NBLDPC

[3]In S-SCL decoding, the information bits are divided (partitioned). This is different from the PSCL decoding in [21] where the coded bits are partitioned. However, the S-SCL decoder has the same message schedule for inner polar codes as PSCL decoder. If we consider swapping storage locations of frozen bits and information bits in the decoding process, the memory requirement analysis of PSCL decoding can also be applied to the S-SCL decoding.

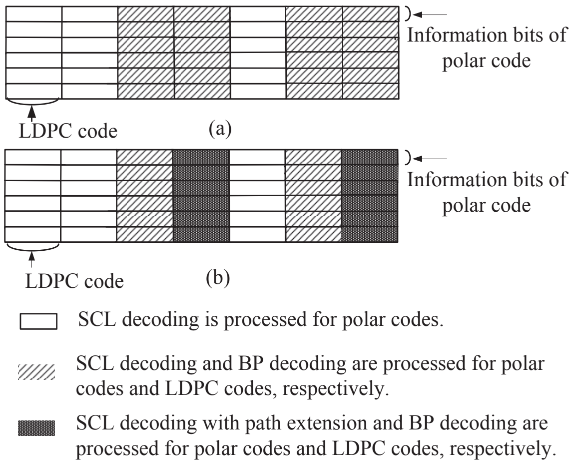

Fig. 5: Decoding of NBLDPC-polar codes with different decoders. (a) S-SCL decoder is used. (b) SES-SCL decoder is used.

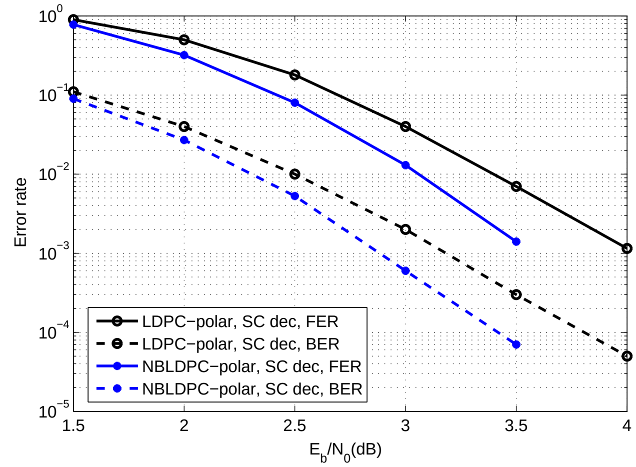

Fig. 6: Performances of concatenated polar codes with binary LDPC codes and non-binary LDPC codes, the overall coded length is $N = 8192$ and the overall code rate is $R = 1/3$.

codes is $k/t(n_{iter}cu2^t(t+u))$ [27], c and u are the number of check nodes and the mean row weight of parity check matrix H, respectively. Therefore, the overall complexity for NBLDPC-polar codes with S-SCL decoding is $2^tN\log n + (k/t)n_{iter}cu2^t(t+u))$, and with SES-SCL decoding is $l2^tN\log n + (k/t)n_{iter}cu2^t(t+u))$.

VI. Simulation Results

In this section, we present simulation results to compare the error correction performance of NBLDPC-polar codes over the additive white Gaussian noise (AWGN) channel with binary phase-shift-keyed (BPSK) modulation. All the NBLDPC codes used in this section are constructed by using progressive edge-growth (PEG) algorithm [28] over $GF(16)$, the maximum iteration number of BP decoding for LDPC codes is $n_{iter} = 20$. For SES-SCL decoder, the selective survival path number is $l = 2$. The target E_b/N_0 that we use to construct polar codes is 2dB.

To show the superiority of the proposed NBLDPC-polar codes, Fig. 6 presents the FER and bit error rate (BER) comparison between binary LDPC-Polar codes and NBLDPC-Polar codes. The LDPC-polar scheme is constructed with binary LDPC $(32, 28)$ codes and binary polar $(256, 96)$ codes, SC algorithm is used for decoding inner polar code. NBLDPC-polar scheme is constructed with NBLDPC $(32, 28)$ codes over $GF(16)$ and binary polar $(256, 96)$ codes. SC decoding is used for inner polar codes, and the symbol probabilities of NBLDPC codes are approximately calculated using the bit probabilities generated by the polar decoder. As can be seen, the NBLDPC-polar scheme provides a better performance than LDPC-polar scheme and gets a gain of approximately 0.4dB at FER of 10^{-3}. We analyze one possible reason of the superiority of NBLDPC-polar scheme is that, NBLDPC codes offer a better performance than binary LDPC codes [15].

Fig. 7 shows the FER performance of the NBLDPC-polar codes with S-SCL and SES-SCL decoding. The overall block length is $N = 8192$ and the overall code rate is $R = 1/3$. For the NBLDPC-polar code, N is split into $m = 16$ and $n = 512$.

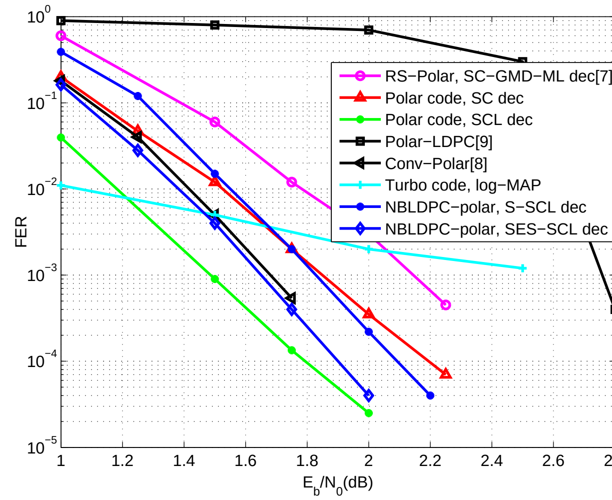

Fig. 7: FER performance of NBLDPC-Polar concatenated codes compared to the RS-Polar code and stand-alone polar code, with $N = 8192$ and $R = 1/3$.

The maximum degrees of outer NBLDPC codewords are $d_{v_{max}} = 4$, $d_{c_{max}} = 32$. For the sake of comparison, RS-polar codes with SC-generalized minimum distance-maximum likelihood (SC-GMD-ML) decoding [6], Conv-polar codes with soft-output multistage iterative decoding (SOMID) scheme [8], Turbo code with log-maximum a posteriori (MAP) algorithm, pure polar code with SC scheme [1], and pure polar code with CA-SCL scheme [4] with list size 4 are also shown in Fig. 7. Note that the performance of RS-polar code is from Fig. 5 of [6] with the overall code length 7680. It is observed that, there is about 0.2dB SNR gain using our proposed SES-SCL decoding on top of the S-SCL decoding for NBLDPC-

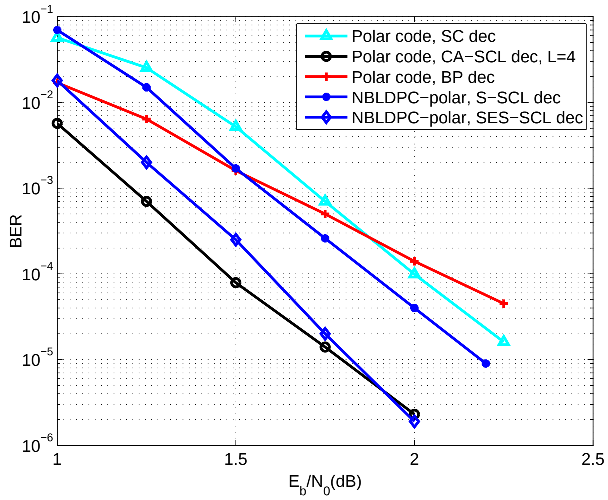

Fig. 8: BER performances of LDPC-polar concatenated codes with other codes with code length $N = 8192$ and code rate $R = 1/3$.

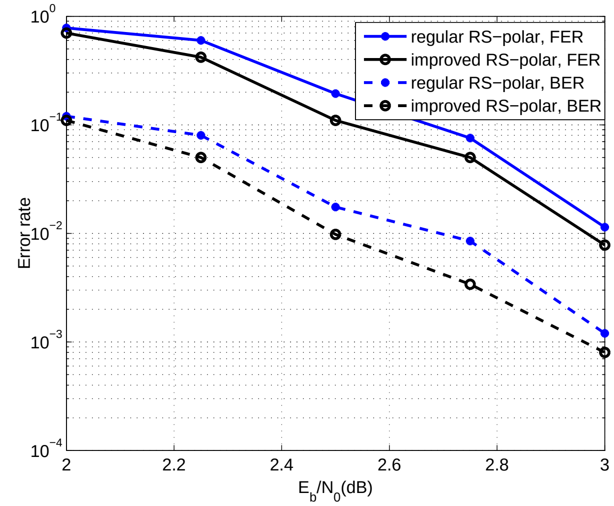

Fig. 9: Performances of concatenated codes with code length $N = 7680$ and code rate $R = 1/3$.

polar concatenated codes. At FER of 10^{-3}, the NBLDPC-polar code with the proposed SES-SCL scheme has nearly 0.2dB and 0.5dB gains over pure polar code with SC scheme and RS-polar code with SC-GMD-ML scheme, respectively. NBLDPC-polar code performs slightly better than Conv-polar code. It is also observed that, pure polar code with CA-SCL scheme provides better FER performance than NBLDPC-polar code with SES-SCL scheme.

Fig. 8 shows the BER performance of NBLDPC-polar codes along with other codes. As shown in Fig. 8, NBLDPC-polar code with the proposed SES-SCL scheme outperforms pure polar code with BP or SC scheme. The SES-SCL scheme has about 0.4dB gain over S-SCL scheme at BER of 10^{-5}. Also according to Fig. 8, compared with pure polar code with CA-SCL scheme with list size 4, NBLDPC-polar code with SES-SCL scheme has about 0.2dB BER performance degradation at low SNR. However, when SNR reaches 1.8dB or higher, the degradation becomes negligible. Even at $E_b/N_0 = 2dB$, NBLDPC-polar code has a better BER performance. Moreover, the proposed S-SCL decoder requires only about 50% of the memory required by the CA-SCL decoder, the proposed SES-SCL decoder requires about 68% of the memory required by the CA-SCL decoder with $Q_\alpha = 6$ bits, and $Q_{PM} = 8$ bits.

Besides, to show the advantage of the SES-SCL scheme, simulations with concatenated polar codes with other non-binary block codes are also conducted. Simulations with RS codes are shown in Fig. 9. The inner polar code is a $(512, 232)$ code and the outer RS code is a $(15, 11)$ code. The blue line is the traditional regular RS-polar scheme in [6], SC-GMD-ML scheme is used for decoding. Whereas the black line with circle represents the improved RS-polar scheme, in which the selective extension method is used for the decoding of inner polar codes. As there is no soft information offered by the RS decoder, here we use the difference of path likelihoods between the first hard-decision and the secondary hard-decision symbol to approximately represent the RS symbol reliability [7]. It can be seen that, the improved RS-polar scheme is able to improve the BER and FER performance by about 0.1dB. The performance gain is not as good as in Fig. 7 and 8, possibly because of the fact that the RS code can not provide soft information, the path extension of sequent sub-blocks is based on the approximately estimated information. The results once again suggest that the SES-SCL decoder can improve the performance of concatenated polar codes with outer non-binary block codes.

From the above description, we can draw a conclusion that the concatenated NBLDC-polar codes outperform pure polar codes with SC or BP decoding, while the required memory space is much less than that of pure polar codes with BP decoding. NBLDPC-polar codes can also outperform RS-polar codes and Conv-polar codes. Pure polar codes with CA-SCL decoding provides better performance at the cost of much larger memory space. At high SNR, the proposed SES-SCL scheme has comparable BER performance with CA-SCL decoding with list size 4, but with lower memory space. Hence we can conclude that NBLDPC-polar concatenated codes with the proposed SES-SCL decoding scheme strikes a better balance between performance and storage than the existing schemes.

VII. Conclusions

In this paper, a concatenation scheme of polar codes with interleaved NBLDPC codes has been considered. We have then proposed a new SCL decoding, namely SES-SCL, for the NBLDPC-polar concatenated codes. We have also compared polar codes with various decoders in terms of the memory space requirement and decoding complexity. Analysis and numerical results have shown that the performance of NBLDPC-polar codes is better than RS-polar codes and Conv-polar codes. Furthermore, NBLDPC-polar codes with the proposed SES-SCL decoding algorithm can outperform stand-alone polar codes decoded with BP or SC decoding. The SES-SCL scheme has comparable performance with CA-SCL scheme at high SNR while having smaller memory size and lower memory space. Also, the selective extension SCL decoding method described in this paper can be applied to the concatenation of polar codes with non-binary block codes.

In future work, the design of symbol-decision SCL decoding will be considered for the concatenated polar codes to improve throughput and hardware efficiency. Also the concatenation structure will be extended over non-Gaussian channels [29].

VIII. Acknowledgment

This work was supported by the National Natural Science Foundations of China (no. 61671128 and no. 61601414), and the Fundamental Research Funds for the Central Universities (no. 010403302/CUC18A007-2 and 3132018XNG1808).

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, Jul. 2009.

[2] Reno, NV, USA. 3GPP. (Nov. 2016). *Final Report of 3GPP TSG RAN WG1 #87 v1.0.0.* http://www.3gpp.org/ftp/tsg_ran/WG1_RL1/TSGR1_87/Report/Final_Minutes_report_RAN1%2387_v100.zip.

[3] I. Tal and A. Vardy, "List decoding of polar codes," in *IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2011, pp. 1–5.

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

[5] M. Bakshi, S. Jaggi, and M. Effros, "Concatenated polar codes," in *IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2010, pp. 918–922.

[6] H. Mahdavifar, M. El-Khamy, J. Lee, and I. Kang, "Performance limits and practical decoding of interleaved Reed-Solomon polar concatenated codes," *IEEE Trans.Commun.*, vol. 62, no. 5, pp. 1406–1417, May 2014.

[7] Y. Wang, W. Zhang, Y. Liu, L. Wang, and Y. Liang, "An improved concatenation scheme of polar codes with Reed-Solomon codes," *IEEE Commun. Lett.*, vol. 21, no. 3, pp. 468–471, Mar. 2017.

[8] Y. Wang, K. R. Narayanan, and Y. C. Huang, "Interleaved concatenations of polar codes with BCH and convolutional codes," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 2, pp. 267–277, Feb. 2016.

[9] R. Gallager, "Low density parity check codes," *IRE Trans. Info. Theory, IT-8:pp. 2129, Jan. 1962.*

[10] M. C. Davey and D. MacKay, "Low-density parity check codes over GF(q)," *IEEE Commun.Lett.*, vol. 2, no. 6, pp. 165–167, Jun. 1998.

[11] A. Eslami and 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, Mar. 2013.

[12] J. Guo, M. Qin, A. G. i Fbregas, and P. H. Siegel, "Enhanced belief propagation decoding of polar codes through concatenation," in *IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2014, pp. 2987–2991.

[13] Q. P. Yu, Z. P. Shi, L. Deng, and X. Li, "An improved belief propagation decoding of concatenated polar codes with bit mapping," *IEEE Commun.Lett.*, vol. 22, no. 6, pp. 1160–1163, Jun. 2018.

[14] X. Li, Q. Yu, Z. Shi, Y. Li, and Q. Yan, "Concatenations of polar codes with outer nonbinary LDPC codes," in *17th IEEE Int. Conf. Commun. Technol. (ICCT)*, Oct. 2017, pp. 117–121.

[15] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 619–637, Feb. 2001.

[16] B. Yuan and K. K. Parhi, "Low-latency successive-cancellation list decoders for polar codes with multibit decision," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 23, no. 10, pp. 2268–2280, Oct. 2015.

[17] C. Xiong, J. Lin, and Z. Yan, "Symbol-decision successive cancellation list decoder for polar codes," *IEEE Trans. Signal Process.*, vol. 64, no. 3, pp. 675–687, Feb. 2016.

[18] A. Hadi, E. Alsusa, and K. M. Rabie, "A method to enhance the performance of successive cancellation decoding in polar codes," in *10th Int. Symp. on Communication Systems, Networks & Digital Signal Processing (CSNDSP)*, Jul. 2016, pp. 1–5.

[19] R. Yazdani and M. Ardakani, "An efficient analysis of finite-length LDPC codes," in *IEEE Int. Conf. Commun. (ICC)*, Jun. 2007, pp. 677–682.

[20] Y. Wang and K. R. Narayanan, "Concatenations of polar codes with outer BCH codes and convolutional codes," in *52nd Annu. Allerton Conf. Commun., Control, Comput.*, Sep. 2014, pp. 813–819.

[21] S. A. Hashemi, A. Balatsoukas-Stimming, P. Giard, C. Thibeault, and W. J. Gross, "Partitioned successive-cancellation list decoding of polar codes," in *IEEE Inter. Conf. on Acoustics, Speech and Sig. Process.(ICASSP)*, Mar. 2016, pp. 957–960.

[22] S. A. Hashemi, C. Condo, F. Ercan, and W. J. Gross, "Memory-efficient polar decoders," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 7, no. 4, pp. 604–615, Dec. 2017.

[23] S. A. Hashemi, M. Mondelli, S. H. Hassani, C. Condo, R. L. Urbanke, and W. J. Gross, "Decoder partitioning: Towards practical list decoding of polar codes," *IEEE Trans.Commun.*, pp. 1–1, 2018.

[24] B. Yuan and K. K. Parhi, "Successive cancellation list polar decoder using log-likelihood ratios," in *48th Asilomar Conference on Signals, Systems and Computers*, 2014, pp. 548–552.

[25] A. Balatsoukas-Stimming, M. B. Parizi, and A. Burg, "LLR-based successive cancellation list decoding of polar codes," *IEEE Trans. Signal Process.*, vol. 63, no. 19, pp. 5165–5179, Oct. 2015.

[26] H. Wymeersch, H. Steendam, and M. Moeneclaey, "Log-domain decoding of LDPC codes over GF(q)," in *IEEE Intern. Conf. on Commun. (ICC)*, vol. 2, Jun. 2004, pp. 772–776.

[27] H. Song and J. R. Cruz, "Reduced-complexity decoding of Q-ary LDPC codes for magnetic recording," *IEEE Trans. Magn.*, vol. 39, no. 2, pp. 1081–1087, Mar. 2003.

[28] J. Xu, L. Chen, I. Djurdjevic, S. Lin, and K. Abdel-Ghaffar, "Construction of regular and irregular LDPC codes: Geometry decomposition and masking," *IEEE Trans. Inf. Theory*, vol. 53, no. 1, pp. 121–134, Jan. 2007.

[29] A. Hadi, K. M. Rabie, and E. Alsusa, "Polar codes based OFDM-PLC systems in the presence of middleton class-a noise," in *10th Int. Symp. on Communication Systems, Networks & Digital Signal Processing (CSNDSP)*, Jul. 2016, pp. 1–6.