Efficient Belief Propagation Polar Decoder With Loop Simplification Based Factor Graphs

Yuqing Ren, *Student Member, IEEE*,
Yifei Shen, *Student Member, IEEE*,
Zaichen Zhang, *Senior Member, IEEE*,
Xiaohu You, *Fellow, IEEE*, and Chuan Zhang, *Member, IEEE*

***Abstract*—The performance of belief propagation list (BPL) decoding of polar codes is related to the selection of L factor graphs (FGs), which have the least number of girths. However, the straightforward search of such FGs is of high complexity. To achieve good performance with reasonable complexity, we propose an efficient method to find FGs with the least number of length-12 loops in all permuted FGs. Since some length-12 loops have been destroyed by redundant decoding operations, the corresponding FGs can be simplified to different numbers of length-12 loops. Thanks to the proposed loop simplification (LS), BPL decoding is now based on more efficient FGs, resulting in better performance and lower average decoding latency than the state-of-the-art. Numerical results have shown that the performance improvement is 0.15 dB when frame error ratio (FER) is 10^{-4}, for $(1024, 512)$ codes with $L = 64$.**

***Index Terms*—Polar codes, length-12 loop, belief propagation list (BPL), loop simplification (LS).**

I. Introduction

Polar codes proposed by Arıkan in [1] have received intensive attention from both academia and industry. They have been standardized to protect control channels for the enhanced mobile broadband (eMBB) scenarios by 3GPP [2]. There exist two main directions to decode polar codes: successive cancellation (SC) decoding [1] and belief propagation (BP) decoding [3]. For SC decoding, the authors of [4] propose the SC list (SCL) decoding that can approach the maximum likelihood (ML) decoding. The cyclic redundancy check (CRC) aided SCL (CA-SCL) decoding can further improve the performance, which is comparable to that of the WiMaX low-density parity-check (LDPC) codes [5]. However, due to the serial manner, the SC family compared to BP decoding suffers from two inherent flaws: hard output and long latency [6].

On the other hand, BP algorithm, which is based on the factor graph (FG) composed of processing elements (PEs), is inherently parallel. Though BP decoding is of higher throughput than SCL decoding, it still has drawbacks. First, due to the slow convergence, BP decoding usually needs lots of iterations. To reduce the iteration number, **G**-matrix [7] and CRC [8] are employed for early termination. Secondly, due to the existence of loops in the FG, the error-correction performance of BP decoding is not comparable to that of SCL. The authors of [9] utilize m cyclic shift permutations from $m!$ permuted FGs for better performance, where $m = \log_2 N$. Then, BP list (BPL) decoding [10] is proposed to approach the bound of SCL decoding. By selecting L permuted FGs randomly, BPL decoding can expand BP decoding into L parallel paths, and choose the most likely path based on the Euclidean distance. However, existing methods ignore the performance difference among permuted FGs. Therefore, an empirical approach to select L good permutations is proposed by [11]. Simulation results show that only the right stages of the original FG need to be permuted for good permutations. Unfortunately, this empirical strategy lacks theoretical analysis to explain detailed reasons and not applicable when the number of right stages to be permuted is too large.

To this end, we propose an efficient method to find L FGs to enhance the error-correction performance of BPL decoding. Similar to LDPC codes [12], length-12 loops defined by [13] affect the performance of BP polar decoding. According to our analysis, the redundant operations in BPL decoding can destroy some length-12 loops. Thus, loop simplification (LS) is proposed to efficiently sort all the permuted FGs by the number of length-12 loops in them. Numerical results have shown that BPL decoding based on the proposed LS method outperforms the state-of-the-art (SOA) [11] with even lower average decoding latency.

II. Preliminaries

A. Polar Codes

Due to the channel polarization phenomenon, the capacity of N binary-input coordinate channels is extremely distributed in [1]. The symbol (N, K) denotes a polar code of code length N and information length K, and $\mathcal{A}$ denotes the set of information bits indices. Assuming u_1^N as the input vector, x_1^N is generated by Kronecker power $\mathbf{G}^{\otimes m}$ as Eq. (1) follows:

$$x_1^N = u_1^N \mathbf{G}^{\otimes m}, \mathbf{G} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}. \quad (1)$$

B. BP Decoding of Polar Codes

Pamuk puts forward the min-sum approximation [14] to simplify the BP equations. Then, Yuan finds out scaled min-sum (SMS) decoding [7] by introducing a parameter $\alpha = 0.9375$ to decrease the performance loss. In Eq. (2), the function of every PE is to propagate left-to-right and right-to-left log-likelihood ratios (LLR) messages of the i-th index of layer j, which are expressed by $R_{i,j}$ and $L_{i,j}$, respectively.

$$\begin{aligned} L_{i,j} &= g(L_{i,j+1}, L_{i+2^j,j+1} + R_{i+2^j,j}), \\ L_{i+2^j,j} &= g(L_{i,j+1}, R_{i,j}) + L_{i+2^j,j+1}, \\ R_{i,j+1} &= g(R_{i,j}, L_{i+2^j,j+1} + R_{i+2^j,j}), \\ R_{i+2^j,j+1} &= g(R_{i,j}, L_{i,j+1}) + R_{i+2^j,j}, \end{aligned} \quad (2)$$

where $g(a, b) = 0.9375 * \operatorname{sgn}(a) \operatorname{sgn}(b) \min(|a|, |b|)$, adopting SMS algorithm, and Fig. 1 is the FG of $(8, 5)$ polar codes.

Manuscript received September 19, 2019; revised December 27, 2019 and February 23, 2020; accepted March 1, 2020. Date of publication March 9, 2020; date of current version May 14, 2020. This work was supported in part by NSFC under Grants 61871115 and 61501116, in part by the Jiangsu Provincial NSF for Excellent Young Scholars under Grant BK20180059, in part by the Six Talent Peak Program of Jiangsu Province under Grant 2018-DZXX-001, in part by the Distinguished Perfection Professorship of Southeast University, in part by the Fundamental Research Funds for the Central Universities, and in part by the SRTP of Southeast University. The review of this article was coordinated by Prof. Y. T. Su. *(Corresponding author: Chuan Zhang.)*

The authors are with the LEADS of Southeast University the National Mobile Communications Research Laboratory of Southeast University, Quantum Information Center of Southeast University, Hammond, LA 70402 USA, and also with the Purple Mountain Laboratories, Nanjing 210096, China (e-mail: 15895896213@163.com; 597592813@qq.com; zczhang@seu.edu.cn; xhyu@seu.edu.cn; chzhang@seu.edu.cn).

Digital Object Identifier 10.1109/TVT.2020.2979334

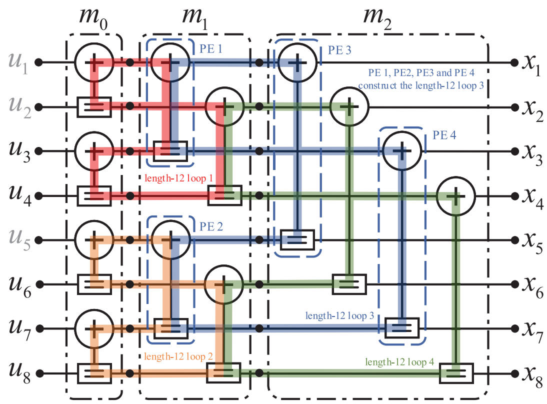


Fig. 1. The original FG of $(8, 5)$ polar codes with $\mathcal{A} = [8, 7, 6, 4, 3]$.

C. BPL Decoding of Polar Codes

Through permuting m stages of the original FG, BPL decoding can extend L decoding paths with different permuted FGs, which is originally from [15]. In addition to the above fully parallel BPL decoding, the authors of [16] propose a serial BPL decoding. If failing to pass the early termination with the maximum number of iteration, the received LLRs will be sent to the next BP decoder with a different FG. Compared to the fully parallel BPL decoding, the serial BPL decoder demands less computational complexity. In this paper, we use the serial BPL decoding combined with CRC early termination.

III. Proposed LS Algorithm for Permuted FGs

A. Simplification of the Original FG

There are too many loops in the polar FG, which is one of the critical reasons for errors in BP decoding. In the corresponding operations of LDPC, eliminating length-4 cycles in check matrix $\mathbf{H}$ is a standard method to deal with loop effect. In polar codes, the length-12 loop defined by [13] is a vital optimization object in the FG. As shown in Fig. 1, there exist four length-12 loops labeled by different colors. As the "length-12 loop 3" shows in Fig. 1, every length-12 loop consists of four independent PEs.

In general, there are $\frac{N\cdot(m-1)}{4}$ length-12 loops in the original FG of (N, K) polar codes. For permuted FGs of polar codes, stage permutation will only transform the shape and size of loops, instead of the number of loops [10]. However, through the analysis of message propagation process, permuted FGs have changed the number of length-12 loops, which is an important reason for performance improvement in BPL decoding. As outlined in Eq. (2), BP decoding of polar codes has too many redundant operations on the left side of the FG, which makes the speed of convergence slow. The messages of the left side of the FG update actively only when i is the index of frozen bit and $i + 2^j$ is the index of information bit, respectively. In other cases, the messages of L layers and R layers propagate with fixed modules, which can not constitute loops in iterative decoding. For the simplicity of expression, the subscripts of messages of L and R are simplified to 1 and 2 in Fig. 2.

By observing the input and output of three fixed PEs labeled as grey in Fig. 2, message propagation has changed into a restricted mode, instead of two-way form. In the grey PEs, the value of $L_{i,j}$ and $L_{i+2^j,j}$ only depends on the value of $L_{i,j+1}$ and $L_{i+2^j,j+1}$ propagated from the right stage in the current iteration. The output value of $R_{i,j+1}$ and $R_{i+2^{j+1},j+1}$ is the same as the input $R_{i,j}$ and $R_{i+2^{j+1},j}$, which is fixed to [0, 0], $[\infty, 0]$ or $[\infty, \infty]$. Therefore, from the perspective of loops, most of PEs on the left side of the FG seem to be "frozen" in iterative

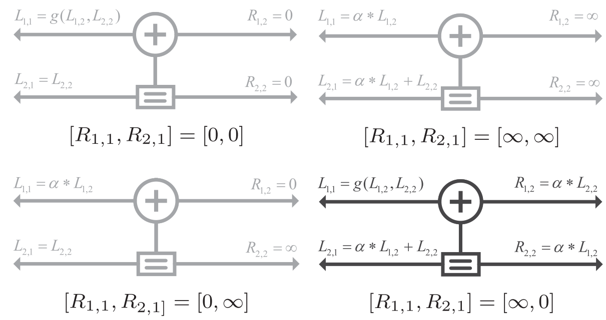


Fig. 2. Message propagation process on the left side of the FG.

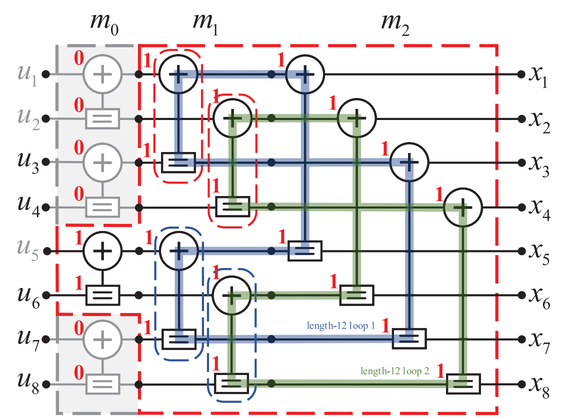


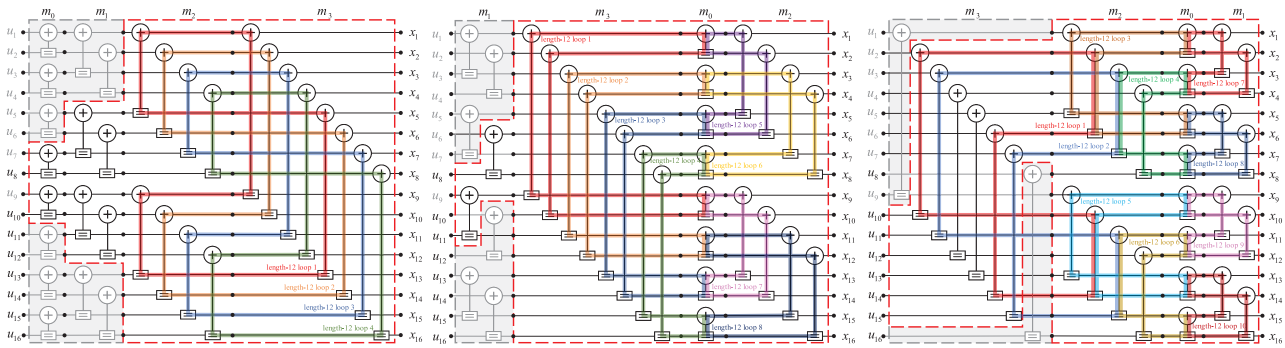
Fig. 3. The simplified FG of $(8, 5)$ polar codes with $\mathcal{A} = [8, 7, 6, 4, 3]$.

decoding, as shown in the three grey PEs in Fig. 2, on the other hand, most of PEs on the right side of the FG show an "active" state, as shown in the black PE in Fig. 2. This frozen state can spread to the interior of the FG gradually from left to right. However, it could be prevented by the following two cases.

1) If the current PE exists any node connected to the "active" node in the previous stage, it will be marked as "active".
2) In the current stage, if satisfying the conditions that i is the index of frozen bit and $i + 2^j$ is the index of information bit, respectively, the current PE could be marked as "active" state too.

As a consequence, by the judgment from left to right, the original FG has been divided into two sections marked as the frozen and active state. As mentioned above, a complete length-12 loop demands four independent and active PEs, so the frozen part of FG destroy some length-12 loops. Moreover, due to *case* 1, if the left couple of PEs in the length-12 loop are active, the right couple of PEs must be active. Therefore, when judging whether the current length-12 loop exists, we only need to judge the state of the left couple of PEs in it. The simplified FG of (8, 5) polar codes is shown in Fig. 3.

As illustrated in Fig. 3, the frozen PEs marked with "0" make up the frozen parts of the FG painted in grey. The remaining PEs marked with "1" compose the active part framed by the red dotted line. Taking the stage m_1 as an example, there exist four PEs which adhere to different cases. The above two PEs framed with the red dotted line follow the case 2 rule, and the others framed with the blue dotted line follow the case 1 rule. Hence, the simplification FG of (8,5) polar codes has only two length-12 loops finally in Fig. 3.



$[m_0, m_1, m_2, m_3]$ with 4 length-12 loops $[m_1, m_3, m_0, m_2]$ with 8 length-12 loops $[m_3, m_2, m_0, m_1]$ with 10 length-12 loops

Fig. 4. The distribution and quantity of length-12 loops in three kinds of permuted FGs for $(16, 8)$ polar codes.

B. Selection of Efficient FGs Based on LS Algorithm

In this section, a method called LS algorithm is proposed to find more efficient FGs from the massive set of $m!$ permuted FGs candidates to construct BPL decoding. Although the authors of [17] propose that there exist $\prod_{l=0}^{m-1}(m-l)^{(2^l)}$ FGs in polar codes with N, it is practical to consider $m!$ FGs in [11]. According to the above research, the number of the remaining length-12 loops which exist in the simplified FG is an important element to influence the error-correction performance. Therefore, taking the least number of length-12 loops as the core principle, LS algorithm of selecting L efficient FGs of polar codes is illustrated as Algorithm 1.

In addition, we denote by $[m_0,\ m_1\ ...m_{m-1}]$ the original FG and denote by $[m_{\pi^0},\ m_{\pi^1}\ ...m_{\pi^{m-1}}]$ the any FG π with permuted order $[\pi^0,\ \pi^1\ ...\pi^{m-1}]$. For a definite frozen set $\mathcal{A}_c$, the selection by Algorithm 1 is off-line, instead of occupying real-time resources. First, the initialization is based on Fig. 2 to divide the first stage $m_{\pi_k^0}$ into the frozen part and active part. Then, judge whether the current length-12 loop is active and record the current number of length-12 loops. Finally, return the l FGs with the least number of length-12 loops. Moreover, the off-line complexity of constructing the proposed set is $\mathcal{O}(N\log_2 N\cdot(\log_2 N)!)$. For instance, $[m_0, m_1, m_2, m_3]$, $[m_1, m_3, m_0, m_2]$ and $[m_3, m_2, m_0, m_1]$ for $(16, 8)$ polar codes are demonstrated in Fig. 4. The numbers of length-12 loops are distinguished by Algorithm 1, which are 4, 8 and 10 respectively.

IV. Numerical Results

In this section, the error-correcting performance and computational complexity of the proposed LS algorithm are presented. Considering the standards of the 5G eMBB control channel, $(1024, 512)$ polar codes are selected as the main experimental subjects. Moreover, all polar codes signals are modulated by binary phase-shift keying (BPSK) and transmitted over the additive white Gaussian noise (AWGN) channels.

Fig. 5 shows the probability density function (PDF) of the remaining length-12 loops simplified by LS algorithm in all 10! permuted FGs for $(1024, 512)$ + CRC-24 polar codes, which appears a trend of normal distribution roughly and $\mathcal{A}$ includes 24 extra CRC bits. In this paper, we use the CRC24 of 5G New Radio and its generator polynomial is shown in Eq. (3). Every point in Fig. 5 represents a set of permuted FGs with the same number of length-12 loops. Moreover, the priority of all sets of permuted FGs increases as the number of length-12 loops decreases. In the current definite code construction, the least number of length-12 loops is 1260 and the corresponding probability density is 0.005291%, as shown by the point "Q" in Fig. 5, which means there exist 192 permuted FGs with the same priority. Besides, the original FG always belongs to this set "Q". When judging the priority of permuted

Algorithm 1: Selection of FGs based on LS.

Input: $\mathcal{A}_c$, Permuted FGs: $\pi_1, \pi_2 \ldots \pi_{m!}$
Output: Set of FGs: $\pi_{l^1}, \pi_{l^2}, \ldots \pi_{l^l}$
Initilization: LP_num $\leftarrow \{\mathbf{0}\}$;
for $k = 1 : 1 : m!$ **do**
  `// Process` π_k `with` $[m_{\pi_k^0}, m_{\pi_k^1} \ldots m_{\pi_k^{m-1}}]$
  for *all PEs in the first stage* $m_{\pi_k^0}$ **do**
    `// Initialization the first stage`
    if $i \in \mathcal{A}_c$ ***and*** $i + 2^{m_{\pi_k^0}} \in \mathcal{A}$ **then**
      `// "active" state`
      Set the state of PE to 1;
    else
      `// "frozen" state`
      Set the state of PE to 0;
  for $j = 0 : 1 : m-2$ **do**
    `// Frozen and active states propagate`
    for *all couples of PEs in a same length-12 loop* **do**
      if *a couple of PEs are both* 1 **then**
        LP_num[k] = LP_num[k] + 1;
        Set the state of PEs which connected in the next stage $m_{\pi_k^{j+1}}$ to 1;
      else if *existing any node is* 1 **then**
        `// case 1`
        Set the state of PEs which connected in the next stage $m_{\pi_k^{j+1}}$ to 1;
      else if $i \in \mathcal{A}_c$ ***and*** $i + 2^{m_{\pi_k^{j+1}}} \in \mathcal{A}$ **then**
        `// case 2`
        Set the state of PE which re-satisfying in the next stage $m_{\pi_k^{j+1}}$ to 1;
return *l FGs with the least* ***LP_num***: $\pi_{l^1} \ldots \pi_{l^l}$;

FGs in the same set, our strategy is to pick out the FGs which have more fixed stages on the left side. To express concisely, we denote the fixed left stage as $[m_0,\ m_1\ ...m_{k-1}]$, $1 \leq k \leq m$. For the FGs with the same number of length-12 loops, the value of k is larger, the possibility of selecting is higher, which is also consistent with the numerical results in [11]. Hence, the proposed LS algorithm can realize the priority of all permuted FGs sorting more accurately.

$$g_{CRC24}(D) = [D^{24} + D^{23} + D^{21} + D^{20} + D^{17} + D^{15} + D^{13} + D^{12} + D^{8} + D^{4} + D^{2} + D]. \quad (3)$$

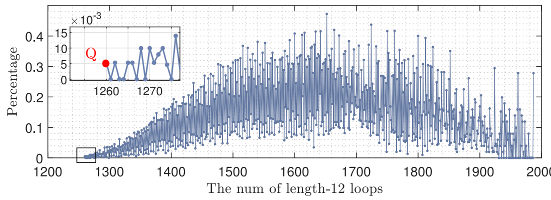


Fig. 5. The PDF of the remaining length-12 loops by LS algorithm in all 10! permuted FGs for $(1024, 512)$ + CRC-24 polar codes.

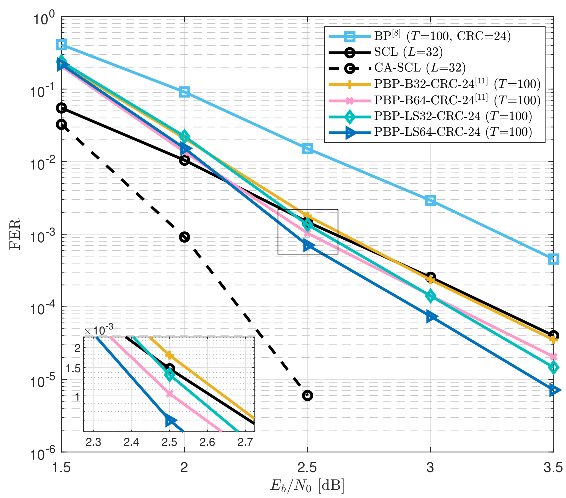


Fig. 6. FER performance of BP, SCL, CA-SCL, PBP-B and PBP-LS for $(1024, 512)$ polar codes. All iterative decoders based on SMS decoding use the CRC-based early termination and CRC-24 is selected by 5G New Radio.

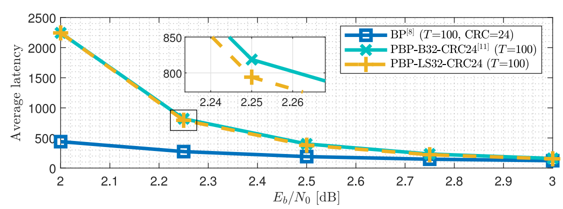


Fig. 7. Average decoding latency comparison between BP, PBP-B and PBP-LS for $(1024, 512)$ aided CRC-24 polar codes.

Fig. 6 illustrates that the FER performance of $(1024, 512)$ polar codes, which focuses on the difference between the proposed LS algorithm and the permuted BP (PBP) in [11]. Considered the implementation of hardware, all iterative decoders are based on SMS decoding and we set the fixed stage $k = 5$ to simulate the PBP-B decoding. When combined with CRC-24 to make early termination and maximum iteration $T = 100$, PBP-B32-CRC-24 can just be level with SCL-32, and the proposed PBP-LS32-CRC-24 has 0.12 dB improvement in comparison with both PBP-B32-CRC-24 and SCL-32 at FER$= 10^{-4}$. With the scale of paths increasing, the gap between PBP-LS64-CRC24 and PBP-B64-CRC24 is expanding gradually, which shows 0.15 dB improvement at FER$= 10^{-4}$. Fig. 7 shows the average decoding latency comparison between BP, PBP-B and PBP-LS for $(1024, 512)$ + CRC24 polar codes. Based on the single column construction in [18], the average decoding latency can be denoted by $\mathcal{L}_{av} = 2mT_{av}$, where T_{av} is the average number of iterations. At $E_b/N_0 = 3$ dB, the $\mathcal{L}_{av}$ of PBP-LS32-CRC24 could approach to 153 clock cycles, and the SCL32 decoder in [19] demands 2592 clock cycles in comparison. Through the enlarged window, we can see that the proposed LS algorithm compared with [11] has a slight advantage in decoding latency, and after 2.75 dB, and PBP-LS32-CRC24 has no significant difference with SMS BP decoding.

V. Conclusion

In this paper, we propose LS to choose a set of efficient FGs from $m!$ permuted FGs candidates. Based on the current code construction, this method can sort the priority of all permuted FGs accurately by the number of the remaining length-12 loops. Therefore, L efficient FGs with the least number of length-12 loops can be provided to BPL decoding. Numerical results show that BPL decoding based on LS method outperforms the SOA in terms of both error-correcting performance and decoding latency.

References

[1] E. Arıkan, "Channel polarization: A method for constructing capacity-achieving codes," *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3051–3073, Jul. 2008.

[2] Ad-Hoc chair (Nokia), "Chairman's notes of agenda item 7.1.5 on channel coding and modulation for NR," 3GPP TSG RAN WG1 Meeting #87, R1-1613710, Reno, USA, Nov. 2016.

[3] E. Arıkan, "Polar codes: A pipelined implementation," in *Proc. Int. Symp. Broadband Commun.*, Jul. 2010, pp. 11–14.

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

[5] TurboBest, *IEEE 802.16e LDPC Encoder/Decoder Core*, Mar. 2015. [Online]. Available: http://www.turbobest.com/tb_ldpc80216e.htm

[6] C. Zhang, Y.-H. Huang, F. Sheikh, and Z. Wang, "Advanced baseband processing algorithms, circuits, and implementations for 5G communication," *IEEE J. Emer. Top. Circuits Syst.*, vol. 7, no. 4, pp. 477–490, Dec. 2017.

[7] B. Yuan and K. K. Parhi, "Early stopping criteria for energy-efficient low-latency belief-propagation polar code decoders," *IEEE Trans. Signal Process.*, vol. 62, no. 24, pp. 6496–6506, Dec. 2014.

[8] Y. Ren, C. Zhang, X. Liu, and X. You, "Efficient early termination schemes for belief-propagation decoding of polar codes," in *Proc. IEEE Int. Conf. ASIC*, Nov. 2015, pp. 1–4.

[9] N. Hussami, S. B. Korada, and R. Urbanke, "Performance of polar codes for channel and source coding," in *Proc. IEEE Int. Symp. Inf. Theory*, 2009, pp. 1488–1492.

[10] A. Elkelesh, M. Ebada, S. Cammerer, and S. T. Brink, "Belief propagation list decoding of polar codes," *IEEE Commun. Lett.*, vol. 22, no. 8, pp. 1536–1539, Aug. 2018.

[11] N. Doan, S. A. Hashemi, M. Mondelli, and W. J. Gross, "On the decoding of polar codes on permuted factor graphs," in *Proc. IEEE Global Commun. Conf.*, Dec. 2018, pp. 1–6.

[12] S. Lin and D. J. Costello, *Error Control Coding, Second Edition*, New Jersey, NJ, USA: Prentice-Hall, Inc., 2004.

[13] A. Eslami and H. Pishro-Nik, "On bit error rate performance of polar codes in finite regime," in *Proc. IEEE Annual Allerton Conf. Commun., Control, Comput.*, 2011, pp. 188–194.

[14] A. Pamuk, "An FPGA implementation architecture for decoding of polar codes," in *Proc. IEEE Int. Symp. Wireless Commun. Syst.*, 2012, pp. 437–441.

[15] A. Elkelesh, S. Cammerer, M. Ebada, and S. T. Brink, "Mitigating clipping effects on error floors under belief propagation decoding of polar codes," in *Proc. Int. Symp. Wireless Commun. Syst.*, Aug. 2017, pp. 384–389.

[16] A. Elkelesh, M. Ebada, S. Cammerer, and S. T. Brink, "Belief propagation decoding of polar codes on permuted factor graphs," in *Proc IEEE Wireless Commun. Net. Conf.*, Apr. 2018, pp. 1–6.

[17] S. A. Hashemi, N. Doan, M. Mondelli, and W. J. Gross, "Decoding reed-muller and polar codes by successive factor graph permutations," in *Proc. IEEE 10th Inter. Symp. Turbo Codes Itera. Infor. Process.*, Dec. 2018, pp. 1–5.

[18] Y. S. Park, Y. Tao, S. Sun, and Z. Zhang, "A 4.68 Gb/s belief propagation polar decoder with bit-splitting register file," in *Proc. Symp. VLSI Circuits Dig. Tech. Papers*, 2014, pp. 1–2.

[19] B.-S. Alexios, 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.