JURNAL INFOTEL
Informatics - Telecommunication - Electronics
Website Jurnal : http://ejournal.st3telkom.ac.id/index.php/infotel
ISSN : 2085-3688; e-ISSN : 2460-0997

Performance Analysis of CRC-Polar Concatenated Codes

Lydia Sari[1]*, Masagus M. Ikhsan Assiddiq U.P.[2], Syah Alam[3], Indra Surjati[4]
[1,2,3,4] Electrical Engineering Department, Universitas Trisakti
[1,2,3,4] Jl. Kyai Tapa 1, Jakarta 11440, Indonesia
*Corresponding Email: lydia_sari@trisakti.ac.id

Received 26 June 2020, Revised 14 August 2020, Accepted 14 September 2020

Abstract — Polar code has been proven to obtain Shannon capacity for Binary Input Discrete Memoryless Channel (BIDMC), and its use has been proposed as the channel coding in 5G technology. However, its performance is limited in a finite block length, compared to Turbo or LDPC codes. This research proposes the use of various CRC codes to complement Polar codes with finite block length and analyses the performance based on Block Error Rate (BLER) to E_s/N_0 (dB). The CRC codes used are of degrees 11 and 24, with 3 different polynomial generators for each degree. The number of bits in the information sequence is 32. The list sizes used are 1, 2, 4, and 8. Simulation results show that CRC and Polar codes' concatenation will yield good BLER vs. E_s/N_0 performance for short blocks of the codeword, with rates 32/864 and 54/864. Concatenating CRC codes with Polar codes will yield a BLER performance of 10^{-2} with E_s/N_0 values of -9.1 to -7.5 dB when CRC codes of degree 11 are used, depending on the SC list used. The use of CRC codes of degree 24 enables a BLER performance of 10^{-2} with E_s/N_0 values of -7 to -6 dB when the SC list used is 1 or 2. The use of CRC codes of degree 24 combined with SC list with sizes 4 or 8 will improve the BLER performance to 10^{-2} with E_s/N_0 values of -8 to -7.5 dB.

Keywords – Polar code, CRC, BLER, SCL decoding

I. INTRODUCTION

The increasing need for wireless telecommunication systems with high data rate and performance has triggered a massive number of researches to address such need. One of the technologies developed to enable high data rate with low latency, wide service area, low power, and high reliability is the 5G technology. The services provided by 5G technology include Enhanced Mobile Broadband (eMBB), Massive Machine-Type Communication (mMTC), and Ultrareliable and Low-Latency Communication (URLLC), all of which require a high bandwidth [1].

Data protection, both in data and control channels in 5G, is of the utmost importance. The data channel of a 5G system uses Low Density Parity Check (LDPC) while the control channel uses Polar code [2]. Polar code is the first channel code that has been proven to obtain Shannon capacity for Binary Input Discrete Memoryless Channel (BIDMC) using a simple decoder [3], [4]. Polar coding's main idea is transforming a pair of identical binary input channels into two channels, of which the characteristics are the polar opposite of each other. It means that one channel is designed to have better performance while the other has worse performance than the original channel. By repeating the polarization operation in pairs to a set of $N = 2^n$ binary input channels, a set of 2^n with different qualities can be achieved for an arbitrary number of integer n [5].

When n is adequately large, part of the channels will be close to error-free, and the rest will be noisy and unusable. The mostly error-free channel will be used to transmit data, while the input to the unusable channels will be set to 0, called the "frozen values", which are known to the receiver. Polar encoding's ability to yield such channel polarization to obtain Shannon capacity has made it into the new standard of 5G technology.

The code's limitation is its low BER performance for finite block length, compared to Turbo or LDPC codes [6], [7]. One of the causes of this limitation is the use of the Successive Cancellation (SC) decoder on the receiver side, which introduces decoding delay [8], [9], [10]. SC is very commonly used to decode Polar codes. The term Successive Interference Cancellation (SIC), as usually found in the literature on multiple access systems is not typically used in the literature on Polar codes. An attempt to improve the decoder for polar code is given in [11], with complexity as the trade-off. It is

https://doi.org/10.20895/infotel.v12i4.494

also noted that the performance of an SC decoder relies on the first decoded codeword stream. If the first decoded stream contains numerous errors, the next decoded stream will also be erroneous, affecting the next decoded stream. Therefore, a method is needed to ensure that the first stream is decoded successfully. One proposed method is to use Parity-Check-Concatenated (PCC) polar code as proposed in [12]. The PCC polar code has parity checks which are scattered in the unfrozen bit sequence which requires a complex decoding process.

This research proposes the use of various CRC codes to complement Polar codes and analyses the performance based on Block Error Rate (BLER) to E_s/N_0 (dB) similar to [13] and [14]. Unlike the approach in [13], where the CRC codes' positions are varied within the block code, in our research CRC codes are appended at the end of a codeword, and hence the decoding process can be simplified. We also focus on using short block length to analyze the Polar code performance for finite blocklength instead of research [14]. The block length is varied as the focus was on finding the CRC code's optimal minimum distance.

This paper is organized as follows. The second part of this paper discusses Polar Transform, Polar Encoding, and SC decoding. The system model and description of the CRC codes used will be given in the third part. Simulation results are given in part four, while part five concludes the paper.

II. RESEARCH METHOD

A. *Polar Transform, Polar Coding, and SC Decoding*

The basic of Polar coding is polar transform, where the input bits are transformed using a basic generator matrix which dimension is 2 × 2 [15], [16]. The basic generator matrix or standard kernel for polar code is

$$G_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \qquad (1)$$

Polar transformation is a modulo-2 multiplication between input data and the generator matrix. The polar transformation of 2 input bits is the modulo-2 multiplication between those bits with $\mathbf{G}_2$, which is

$$\begin{bmatrix} u_1 \; u_2 \end{bmatrix} \mathbf{G}_2 = \begin{bmatrix} u_1 \; u_2 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} u_1 + u_2 & u_2 \end{bmatrix} \qquad (2)$$

The polar transformation result can be represented in a binary tree, as illustrated in Fig.1.

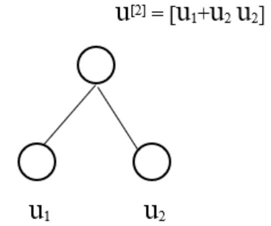

Fig.1. Binary Tree Representation for Polar Transformation with G₂

The kernel matrix can be notated as $\mathbf{G}^{\otimes n}$ where $\mathbf{G}^{\otimes n} = \mathbf{G} \otimes \ldots . \otimes \mathbf{G}$ (n times), which is an n-fold Kronecker product of the standard kernel given in (1) [6],. The value of n is $log_2(N)$, where N is the number of bits in the codeword. Therefore, $\mathbf{G}_2$ can be represented as $\mathbf{G}^{\otimes 1}$. If the input stream consists of 4 bits, the transformation matrix dimension is 4 × 4, notated as $\mathbf{G}_4$ or $\mathbf{G}^{\otimes 2}$. The $\mathbf{G}_4$ matrix has the form of

$$\mathbf{G}_4 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \qquad (3)$$

The polar transformation for four information bits using $\mathbf{G}_4$ can be stated as

$$\begin{aligned} &\begin{bmatrix} u_1 \; u_2 \; u_3 \; u_4 \end{bmatrix} \mathbf{G}_4 = \\ &\begin{bmatrix} u_1 + u_2 + u_3 + u_4 & u_2 + u_4 & u_3 + u_4 & u_4 \end{bmatrix} \end{aligned} \qquad (4)$$

From (3), it is apparent that the polar transformation using $\mathbf{G}_4$ is a repetition of the transformation using $\mathbf{G}_2$. A similar process can be done to perform polar transformation for 2^n input bits. For instance, to transform eight input bits, we use $\mathbf{G}_8$ or $\mathbf{G}^{\otimes 3}$, which is the product of Kronecker multiplication between the kernel $\mathbf{G}_2$ and $\mathbf{G}_4$.

A Polar code (N, K) is formed from K information bits which have been transformed using generator matrix $\mathbf{G}^{\otimes n}$ to yield N bits in the codeword. The steps to form a Polar codeword is as follows:

a. For[13], a codeword which form is a vector "u" with length N, state N-K channels with the worst condition based on the reliability sequence as given in the standard
b. Set the value = 0 for u_i in the N-K worst channels. The bits with 0 in the worst condition channels are said to be in the frozen positions.
c. For the rest of the bits, set the bit values according to the information bits.
d. The codewords are formed by multiplying u by $\mathbf{G}_N$.

The decoding process for a polar codeword is always done successively, starting from u_1, u_2, and subsequently until the whole codeword is successfully decoded. This process is called *Successive Cancellation* (SC). The codeword arriving in the receiver will be processed using the belief propagation concept. The illustration of a codeword arriving in the receiver where N = 2 is given in Fig. 2.

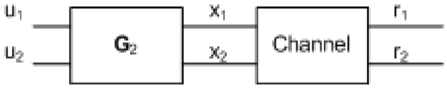

Fig.2. A Received Codeword with N = 2

According to Fig. 2, the codewords consist of 2 bits, namely x_1 and x_2, where each consists of a combination of u_1 and u_2 as follows:

$$x = \left[x_1 \;\; x_2\right] = \left[u_1 + u_2 \quad u_2\right] \qquad (5)$$

The codewords are subsequently modulated and subjected to Additive White Gaussian Noise (AWGN) in the channel, which may cause errors in the codewords that arrive in the receiver. About Fig. 2, the received codewords are r_1 and r_2. The codeword r_1 is used to estimate x_1, while the codeword r_2 is used to estimate x_2. The codewords r_1 and r_2 are called the beliefs for u_1 and u_2. The belief for u_1 is calculated as follows:

$$L(u_1) = f(r_1, r_2) = \operatorname{sgn}(r_1) \cdot \operatorname{sgn}(r_2) \min\left(|r_1|, |r_2|\right) \qquad (6)$$

where $L(u_1)$ is the belief for u_1, min($|r_1|$, $|r_2|$) denotes the minimum value taken from r_1 or r_2, and *sgn* denotes the following function:

$$\operatorname{sgn}(x) = \begin{cases} -1 & x < 0 \\ 0 & x = 0 \\ 1 & x > 0 \end{cases} \qquad (7)$$

The value of u_1 is determined using the hard decision as follows:

$$\begin{aligned} \hat{u}_1 &= 0 \;\; if \;\; L(u_1) \geq 0 \\ \hat{u}_1 &= 1 \;\; if \;\; L(u_1) < 0 \end{aligned} \qquad (8)$$

Subsequently, if $\hat{u}_1 = 0$ then x = $\left[u_1 + u_2 \quad u_2\right]$ = $\left[0 + u_2 \quad u_2\right]$ = $\left[u_2 \quad u_2\right]$. It is apparent that when $\hat{u}_1 = 0$ x only consists of u_2. This means the belief calculation for u_2 need not be performed using (6) but instead can be done using

$$L(u_2) = r_1 + r_2 \qquad (9)$$

If $\hat{u}_1 = 1$, the x = $\left[u_1 + u_2 \quad u_2\right]$ = $\left[1 + u_2 \quad u_2\right]$ = $\left[\bar{u}_2 \quad u_2\right]$, where $\bar{u}_2$ denotes the complement of u_2. Therefore, the belief for u_2 if $\hat{u}_1 = 1$ can be stated as

$$L(u_2) = r_1 - r_2 \qquad (10)$$

SC decoding can obtain optimum BIDMC channel capacity, but its performance is worse than LDPC or Turbo codes when the number of information bits is finite [13]. *Successive Cancellation List* (SCL) is a decoding method suggested for Polar codes. It basically equals SC, but in SCL, every decoding step will generate a list of all possible decoding paths to find u_i [13], [17].

CRC is a binary cyclic code in which every codeword *c* [*m*, *r*] consists of *k*+*p* binary digits. It is generated by appending parity bits to the information bits following a certain pattern [18]. The total number of parity bits is *p*, and the parity bits can be stated as

$$s = \left[s_0, s_1, \ldots, s_{p-1}\right] \qquad (11)$$

A CRC code with *p* parity bits is defined using a generator polynomial *g(x)* with degree *p*. The parity bits are calculated from message bits *m(x)* so that the following equations are satisfied

$$z(x) = \left(x^p \cdot m(x)\right) \bmod g(x) \qquad (12)$$

The codeword obtained by using CRC is

$$c(x) = x^p m(x) + z(x) \qquad (13)$$

For Equation (13), all CRC codewords are divisible by the code polynomial. To determine whether a codeword arriving at the receiver is erroneous, the receiver can check whether said codewords are divisible by the generator polynomial used.

B. *System Model*

The proposed system model used in this research is given in Fig. 3. The message bits are coded using CRC as the first step. After the message bits are transformed into CRC bits given in (13), they are processed using Polar encoding to obtain Polar codewords consisting of *N* bits. The Polar codewords are then transmitted via the channel. The first decoding process in the receiver is SCL decoding, where there are *M* possibilities of decoding paths to determine u_i.

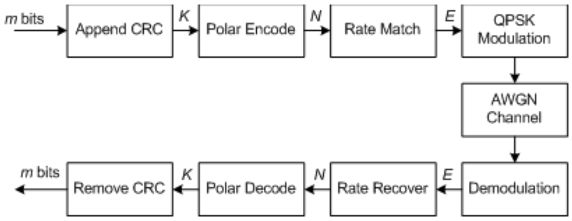

Fig.3. Proposed System Model

In this research, the CRC appended to the information bits are 11 bits and 24 bits, with polynomial patterns as given in [14], [19]. The CRC polynomials are given in Table 1. Rate matching is a process to modify the number of bits with the chosen transmission rate, consisting of a sub-block interleave and bit selections.

The simulations were done using the CRC polynomials given in Table 1. The number of bits in the information bit sequence is 32; therefore, after appending the CRC polynomials, the codeword length is 43 or 54. These short blocks were chosen to analyze the Polar code performance for finite block length. The codeword length after Polar encoding is kept at 864 bits, meaning the code rates are kept at either 43/864 or 54/864.

The simulations were carried out for E_s/N_0 points that started from -0.25 dB with 0.5 increments. The simulations were repeated until the target Block Error Rate of 100 was reached. Simulations are done with L = 1, 2, 4, and 8 for each CRC code. E_s/N_0 as the performance parameter is chosen because E_s depicts the amount of energy needed to transmit one coded symbol. Therefore, it is more appropriate for this research as all the transmitted bits are inherently part of a coded symbol, compared to E_b/N_0 where E_b is the energy needed to transmit one bit.

The CRC polynomial is stated in the hexadecimal notation for simplification. For instance, the polynomial $x^{11} + x^{10} + x^9 + 1$ is stated as E21.

Table 1. CRC polynomial used

Hexadecimal Notation	Degree
E21	11
93F	11
A0F	11
1B2B117	24
11175B7	24
15D6DCB	24

III. RESULT

The simulation results of CRC with 11 parity bits are given in Fig.4. The algorithms will check for all simulations for 1, 2, 4, and 8 possible paths consecutively for all simulations. Hence, the list sizes used are L =1, 2, 4, 8.

The simulation results of CRC-Polar concatenated codes with CRC of degree 24 are given in Fig.5 for L = 1 and 2 and in Fig.6 for L = 4 and 8. These results are depicted in separate figures for clarity since the performances of L = 2 and 4 are close to each other.

IV. DISCUSSION

Figure 4 shows that the use of CRC-Polar concatenated codes enables a BLER performance of 10^{-2} in E_s/N_0 of less than -8.5 dB for L = 4 and L = 8. For L = 1 and L =2, the required E_s/N_0 to obtain BLER = 10^{-2} is less than -7 dB. This means the BLER performance of 10^{-2} for a finite number of bits per coded block (E = 864 bits) can be reached with a low E_s/N_0. It is shown that all three CRC codes of degree-11 used will yield comparable results. The CRC code A0F will give slightly better performance compared to codes E21 and 93F.

Figure 5 shows that the BLER performance of 10^{-2} can be reached at a maximum E_s/N_0 of -6 dB when L = 1 and 2. When L = 4 or 8, the BLER performance of 10^{-2} can be attained with an E_s/N_0 of -7 dB or lower.

Figure 5 and 6 show that the CRC codes used will slightly affect the performance. The best performance is obtained when CRC = 11175B7 is used, which can lower the E_s/N_0 required to reach BLER of 10^{-2} by 0.5 dB lower than when CRC = 15D6DCB is used. The worst performance can be observed when CRC = 15D6DCB is used with L = 8, as its performance is comparable to CRC = 11175B7 with L = 4.

The CRC code of degree 11 provides better performance than that of the CRC code of degree 24. When L = 1 and 2, a polar code equipped with 11-bit CRC will outperform a polar code with 24-bit CRC by 1.5 dB. For L = 4 and 8, the 11-bit CRC yields better performance by 1.5 dB and 1 dB, subsequently, compared to 24-bit CRC.

All simulation results show that the decoding list size will affect the system performance, where the use of large list size will give better performance compared to that of small list size. It can also be observed from the results that CRC-Polar concatenated codes enable the attainment of good BLER performance with low E_s/N_0 requirement for finite block length; in this case, the block length is 864 bits, including the parity bits.

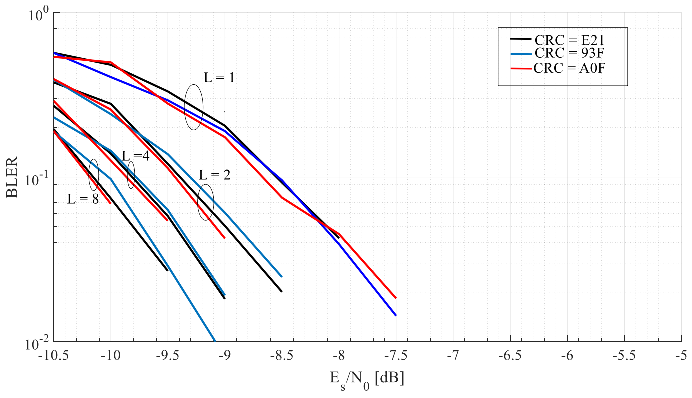

Fig.4. Simulations Results of Polar Coding with 11-Degree CRC, L = 1, 2, 4, 8

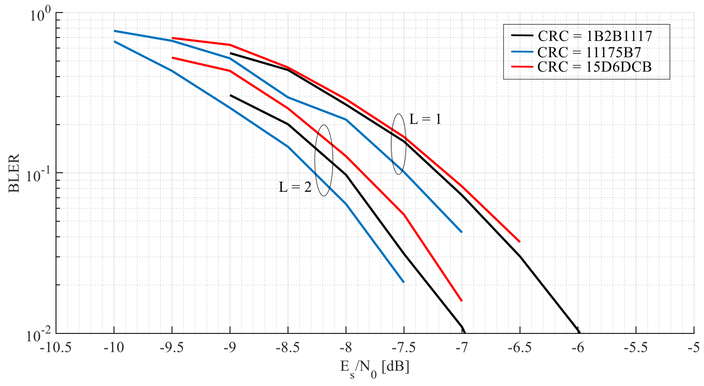

Fig.5. Simulations Results of Polar Coding with 24-degree CRC, *L* = 1 and 2

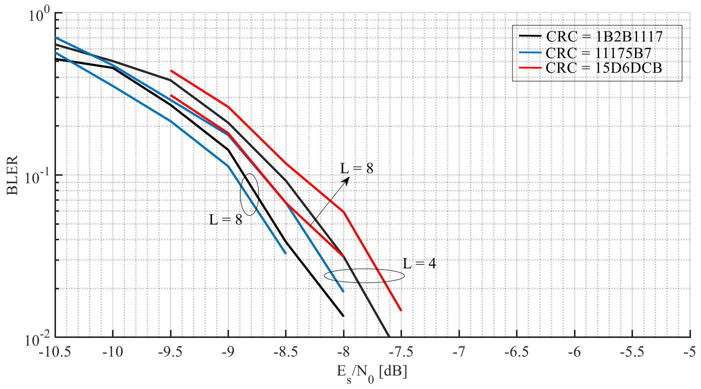

Fig.6. Simulations Results of Polar Coding with 24-degree CRC, *L* = 4 and 8

V. CONCLUSION

Polar code can provide good data transmission reliability and obtain Shannon capacity for Binary Input Discrete Memoryless Channel (BIDMC). However, its performance is poorer than Turbo or LDPC codes for finite block length. In this research, various CRC codes have been coupled with Polar code, and the system performance for finite block length (864 bits) has been analyzed using simulations. The CRC codes used are 11 degrees and 24 degrees, each with 3 different polynomial generators. The Successive Cancellation List's listed size on the receiver side is *L* = 1, 2, 4, and 8. The performance parameter observed is BLER vs. E_s/N_0. Simulations are repeated for E_s/N_0 = -25 dB until 100 errors have been obtained. Results show that the BLER performance of 10^{-2} can be obtained with an E_s/N_0 value of less than -7 dB when CRC codes of degree 11 are used, with *L* = 8 giving the best performance compared to other sizes of *L*. When CRC codes of degree 24 are used, the BLER performance of 10^{-2} is attained with E_s/N_0 value below -6 dB. It is also shown that when CRC = 15D6DCB is used with *L* = 8, its performance is comparable to the use of CRC = 11175B7 with *L* = 4. Simulation results show that CRC and Polar codes' concatenation will yield good BLER vs. E_s/N_0 performance for short codewords with rates 32/864 and 54/864 where the BLER performance of 10^{-2} is achieved at E_s/N_0 ranging from -9.1 to -6 dB.

REFERENCES

[1] G. Ancans and V. Bobrovs, “Spectrum Usage for 5G Mobile Communication Systems and Electromagnetic Compatibility with Existent Technologies,” in *Broadband Communications Networks - Recent Advances and Lessons from Practice*, 2018, pp. 27–41.

[2] J. H. Bae, A. Abotabl, H. P. Lin, K. B. Song, and J. Lee, “An overview of channel coding for 5G NR cellular communications,” *APSIPA Trans. Signal Inf. Process.*, vol. 8, pp. 1–14, 2019.

[3] E. Arikan, “Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels,” *IEEE Trans*, vol. 55, no. 7, pp. 3051–3073, 2009.

[4] P. Akuon and H. Xu, “Polar coded spatial modulation,” *IET Commun.*, vol. 8, no. 9, pp. 1459–1466, 2014.

[5] O. O. Koyluoglu and H. El Gamal, “Key agreement,” *IEEE Trans. Inf. Forensics Secur.*, vol. 7, no. 5, pp. 1472–1483, 2012.

[6] H. Vangala, E. Viterbo, and Y. Hong, “A Comparative Study of Polar Code Constructions for the AWGN Channel,” *arXiv:1501.02473*, 2015.

[7] I. Tal and A. Vardy, “List Decoding of Polar Codes,” *IEEE Trans. Inf. Theory*, vol. 61, no. 5, pp. 2213–2226, 2015.

[8] C. Zhang and K. K. Parhi, “Low-Latency Sequential and Over- lapped Architectures for Successive,” *IEEE Trans. Signal Process.*, vol. 61, no. 10, pp. 2429–2441, 2013.

[9] B. Yuan, S. Member, and K. K. Parhi, “Low-Latency Successive-Cancellation Polar Decoder Architectures Using 2-Bit Decoding,” *IEEE Trans. Circuits Syst.*, vol. 61, no. 4, pp. 1241–1254, 2014.

[10] A. V. and W. J. G. C. Leroux, A. J. Raymond, G. Sarkis, I. Tal, “Hardware implementation of successive cancellation decoders for polar codes,” *J. Signal Process. Syst.*, vol. 69, no. 3, pp. 305–315, 2012.

[11] S. Sets and E. Floor, “On Finite-Length Performance of Polar Codes : Concatenated Design,” *IEEE Trans. Commun.*, vol. 61, no. 3, pp. 919–929, 2013.

[12] T. Wang, D. Qu, and T. Jiang, “Parity-check-concatenated polar codes,” *IEEE Commun. Lett.*, vol. 20, no. 12, pp. 2342–2345, 2016.

[13] J. H. Kim, S. H. Kim, J. W. Jang, and Y. S. Kim, “Low complexity list decoding for polar codes with multiple CRC Codes,” *Entropy*, vol. 19, no. 4, pp. 1–13, 2017.

[14] T. Baicheva and P. Kazakov, “CRC selection for decoding of CRC-polar concatenated codes,” in *Proceedings of 2019 Balkan Conference in Informatics*, 2019, no. 26-28 September, pp. 1–5.

[15] H. Vangala, Y. Hong, and E. Viterbo, “Efficient algorithms for systematic polar encoding,” *IEEE Commun. Lett.*, vol. 20, no. 1, pp. 17–20, 2016.

[16] H. Vangala, E. Viterbo, and Y. Hong, “A new multiple folded successive cancellation decoder for polar codes,” *2014 IEEE Inf. Theory Work. ITW 2014*, pp. 381–385, 2014

[17] X. Wang, J. He, J. Li, Z. Wu, L. Shan, and B. Hong, “Improved Adaptive Successive Cancellation List Decoding of Polar Codes,” *Entropy*, vol. 21, no. 9, p. 899, 2019.

[18] Y. S. K. J.H. Kim, S.H. Kim, J.W. Jang, “Low complexity list decoding for polar codes with multiple CRC codes,” *Entropy*, vol. 19, no. 4, 2017.

[19] T. Specification, “TS 138 212 - V15.2.0 - 5G; NR; Multiplexing and channel coding (3GPP TS 38.212 version 15.2.0 Release 15),” vol. 0, p. 100, 2018.