ERROR CORRECTION IN 5G USING POLAR CODES

Roobasri.S[1], Sadiya Banu.K[2], Dr. Hemajothi.S[3]

[1] *Student, Department of ECE, Prince Shri Venkateshwara Padmavathy Engineering College, Tamilnadu, India*
[2] *Student, Department of ECE, Prince Shri Venkateshwara Padmavathy Engineering College, Tamilnadu, India*
[3] *Professor, Department of ECE, Prince Shri Venkateshwara Padmavathy Engineering College, Tamilnadu, India*

ABSTRACT

Polar coding technique is chosen for 5G New Radio (NR) communication system. Polar codes are believed as prominent breakthrough in channel coding theory. It guarantees apical performance for 5G scenarios and hence it is considered for the 5G New Radio. Polar codes give better performance than LDPC and turbo codes. This coding technique involves capacity achieving than just capacity approaching. It provides the encoding and decoding process foreseen by the 5G standards to facilitate the simulation and implementation of 5G-polar codes.

Keyword : - *Polar codes, Coding techniques in 5G.*

1. INTRODUCTION

Polar codes are a class of capacity-achieving codes. In the past decade, the interest and research effort on polar codes has been constantly rising in academia and industry alike. Within the ongoing 5th generation wireless systems (5G) standardization process of the 3rd generation partnership project (3GPP), polar codes have been adopted as channel coding for uplink and downlink control information for the enhanced mobile broadband (eMBB) communication service.

5G foresees two other frameworks, namely ultra-reliable low-latency communications (URLLC) and massive machine-type communications (mMTC), for which polar codes have been selected as one of the possible coding schemes. The construction of a polar code involves the identification of channel reliability values associated to each bit to be encoded. This identification can be effectively performed given a code length and a specific signal-to-noise ratio.

However, within the 5G framework, various code lengths, rates and channel conditions are foreseen, and having a different reliability vector for each parameter combination is unfeasible. Thus, substantial effort has been put in the design of polar codes that are easy to implement, having low description complexity, while maintaining good error-correction performance over multiple code and channel parameters.

2. POLAR CODES IN 5G

The polar code encoding process foreseen by 5G, from the code concatenation, through interleaving functions, to the polar-code specific subchannel allocation and rate-matching schemes. The purpose of this work is to provide the reader with a straightforward, self-contained guide to the understanding and implementation of 5Gcompliant encoding of polar codes Figure 1. shows the foreseen communication chain and give insights about its operating steps, along with decoding options.

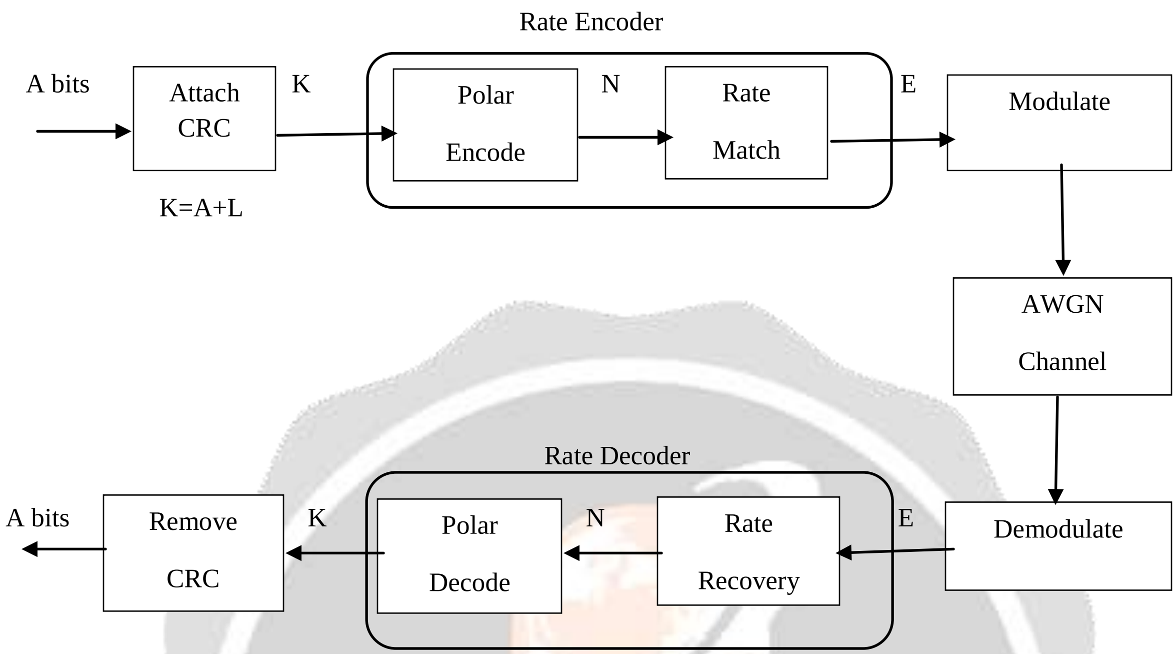

Fig 1: Operation of Polar Codes

2.1 Definition of Polar Codes

Mathematical foundations of polar codes lay on the polarization effect of the matrix $G_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$. In an (N, K) polar code of length N = 2^n, the polarization effect establishes N virtual channels, through which a single bit u_i is transmitted. Each bit-channel, or subchannel, has a different reliability; message bits are allocated to the K most reliable channels. The polar code is hence defined by the transformation matrix $G_N = G_2^{\otimes n}$, i.e. as the n-th Kronecker power of the polarizing matrix, and either the frozen set F of size N − K, or its complementary information set I = F^C of size K, where I and F are subsets of the index set {0, 1,.., 2^{n-1}}. A codeword d = {d_0, d_1, . . ., d_{N-1}} is calculated as

$$d = u \,.\, G_N \tag{1}$$

where the input vector u = {u_0, u_1, . . ., u_{N-1}} is generated by assigning u_i= 0 if i ∈ F, and storing information in the remaining elements. Each index i identifies a different bit-channel.

2.2 Polar code Encoding in 5G

Figure 2 portrays the set of operations that information encoded with polar codes goes through within the 5G framework. In the following, the notation introduced in the 3GPP technical specification will be used. Polar codes in the uplink are used to encode the uplink control information (UCI) over the physical uplink control channel (PUCCH) and the physical uplink shared channel (PUSCH). In the downlink, polar codes are used to encode the downlink control information (DCI) over the physical downlink control channel (PDCCH), and the payload in the physical broadcast channel (PBCH).

A bits have to be transmitted through a code of length E code bits. L CRC bits are added to the information bits, resulting in K bits that will be encoded by a (N, K) mother polar code, with N = 2^n. Rate matching is finally performed to obtain a code of length E and rate R = A/E. Vector a contains the A information bits to be transmitted.

To every A-bit vector, an L-bit CRC is attached. The resulting vector c, constituted of K = A+L bits, is passed through an interleaver.

On the basis of the desired code rate R and codeword length E, a mother polar code of length N is designed, along with the relative bit channel reliability sequence and frozen set

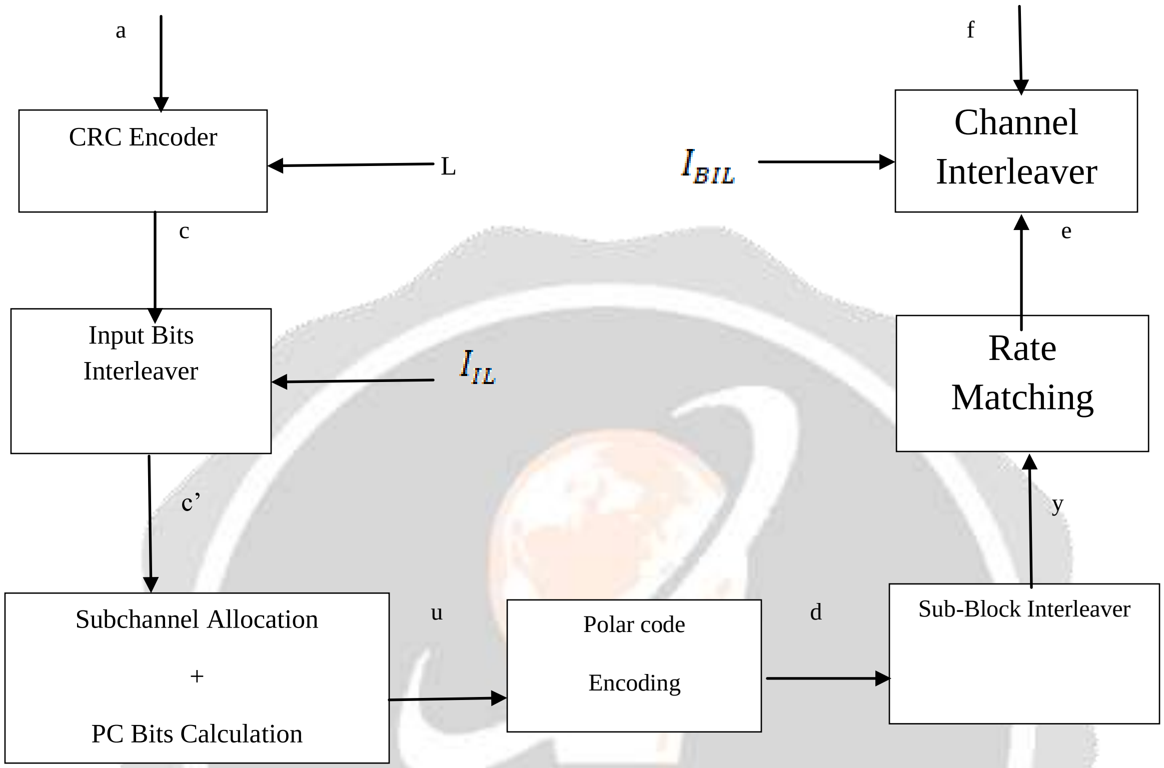

Fig 2: Polar Encoding

Table 1 summarizes the encoding chain parameters depending on channel and code parameters. The n_{max} is used to indicate the uplink and the downlink channel. The flags I_{IL} and I_{BIL} are used to activate and deactivate the input bit interleaver and channel interleaver respectively. The L indicates the length of the CRC polynomial. The n_{PC} denotes the parity-check bit that are inserted with the K information and CRC bits. The n_{PC}^{wm} denotes the weight of rows of the generator matrix.

The interleaved vector c ′ is assigned to the information set along with ad-hoc parity-check bits, while the remaining bits in the N-bit u vector are frozen. Vector u is encoded with d = uG_N, where $G_N = G_2^{\otimes n}$ is the generator matrix for the selected mother code.

Table 1: Channel Parameters

	PUCCH/PUSCH			PDCCH/PBCH
	A ≥ 20	12 ≤ A ≤ 19		
		E – A ≤ 175	E – A > 175	
n_{max}	10			9
I_{IL}	0			1
I_{BIL}	1			0
L	11	6		24
n_{PC}	0	3		0
n_{PC}^{wm}	0	0	1	0

After encoding, a sub-block interleaver divides d in 32 equal-length blocks, scrambling them and creating y, that is fed into the circular buffer. For rate matching, puncturing, shortening or repetition are applied to change the N-bit vector y into the E-bit vector e.

A channel interleaver is finally applied to compute the vector f, that is now ready to be modulated and transmitted. In the following, detail shows the operations necessary in each step of the 5G encoding

2.3 Code parameters and rate matching selection

The 5G polar code encoding process relies on several parameters that depend on the amount and type of information to be transmitted and on the used channel. The first parameter that needs to be identified is the code length of the mother polar code, N = 2^n . The number n is calculated using equation (2).

$$n = \max(\min(n_1, n_2, n_{max}), n_{min}) \quad (2)$$

where, n_{min} and n_{max} give a lower and an upper bound on the mother code length, respectively. In particular, n_{min}= 5, while n_{max}= 9 for the downlink control channel, and n_{max} = 10 for the uplink. Parameter n_2 gives an upper bound on the code based on the minimum code rate admitted by the encoder, i.e. $\frac{1}{8}$; as a consequence, $n_2 = \lceil log_2 (8K)\rceil$. Finally, the value of n1 is bound to the selection of the rate-matching scheme. It is in fact usually calculated as $n_1 = \lceil log_2 (E)\rceil$, so that 2^{n1} is the smallest power of two larger than E.

However, a correction factor is introduced to avoid a too severe rate matching: if $\{log_2 (E)\} < 0.17$, i.e. if the smallest power of two larger than E is too far from E, the parameter is set to $n_1 = \lfloor log_2 (E)\rfloor$. In this case an additional constraint on the code dimension is added, imposing K < $\frac{9}{16}$ E, to assure that K < N. If a code length N > E is selected, the mother polar code will be punctured or shortened, depending on the code rate, before the transmission. In particular, if $\frac{K}{E} \leq \frac{7}{16}$, the code will be punctured, otherwise it will be shortened.

On the contrary, if N < E, repetition is applied and some encoded bits will be transmitted twice; in this case, the code construction assures that K < N. As shown in Table 3.1, a set of flags and parameters assume different values depending on the type of transmission. The flags I_{IL} and I_{IBL} refer to the activation of the input bits interleaver and the channel interleaver respectively. The number of the two types of assistant PC bits are given by n_{PC} and n_{PC}^{wm}.

The length of the information vector A and the length of the transmitted codeword E are dependent on the type, content, and number of consecutive transmissions, and are thus bound to decisions taken in layers higher than the physical one. The reader can consult for detailed information. This guide enables 5G-compliant encoding for any allowed combination of A, R and E.

2.4 CRC encoding

A CRC of L bits is appended to the A message bits stored in a, resulting in a vector c of A + L bits. The possible CRC generator polynomials are given in equation (3), (4) and (5).

$$g_6(x) = x^6+x^4+1 \tag{3}$$

$$g_{11}(x) = x^{11} + x^{10} + x^9 + x^5 + 1 \tag{4}$$

$$g_{24}(x) = x^{24} + x^{23} + x^{21} + x^{20} + x^{17} + x^{15} + x^{13} + x^{12} + x^8 + x^4 + x^2 + x + 1 \tag{5}$$

The polynomial equation 5 is used for the payload in PBCH and DCIs in the PDCCH, while polynomials equation 3 and equation 4 are used for UCIs, in the case 12≤A≤19 and A ≥ 20, respectively. The CRC shift register is initialized by all zeros for UCIs and for the PBCH payloads, and to all ones for the DCIs. Moreover, for DCIs, the CRC parity bits are "scrambled" according to a radio network temporary identifier (RNTI) $x_0^{rnti}, x_1^{rnti}, \dots, x_{15}^{rnti}$, i.e. the RNTI is masked in the last 16 CRC bits calculated by g_{24} (x) as $c_{A+8+k} = c_{A+8+k} \oplus x_k^{rnti}$ for k = 0, . . . , 15.

2.5 Input bits Interleaver

The K bits obtained from the CRC encoder are interleaved before being inserted into the information set of the mother polar code. Figure 3.6 represents the input bit interleaving process. The interleaver is activated through a flag I_{IL}. In particular, the input bit interleaver is activated for PBCH payloads and PDCCH DCIs (I_{IL} = 1). In the case of PUCCH and PUSCH UCIs (I_{IL}= 0). The input bit interleaver interleaves up to K ≤ 164. The interleaving function is applied to c, and the K-bit vector c' is obtained.

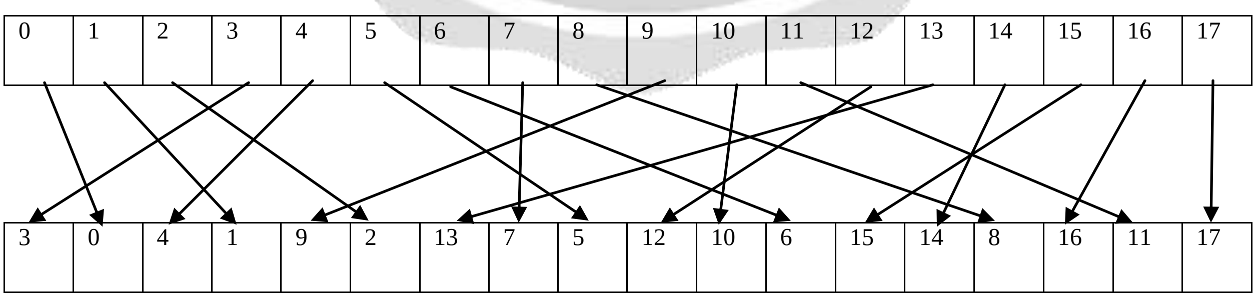

Fig 3: Input Bit Interleaver

2.6 Subchannel allocation and PC bits calculation

In this procedure, vector c ′ is expanded in the N-bit input vector u with the addition of assistant bits and frozen bits.

2.6.1 Assistant bits design

It has been noticed that the introduction of an outer CRC code improves the error-correction performance, for example when used to help the selection of the correct candidate in SCL decoding. In general, it has been proven that the minimum distance of polar codes can be dramatically improved by adding an outer code to polar codes. This improved code spectrum is fully used by SCL decoders, and it has contributed to the selection of polar codes for 5G. Assistant bits can be broadly identified as additional bits that help the decoding of the polar code, either increasing the error-correction performance or improving a metric, like speed or complexity.

2.6.2 Frozen set design

As N goes toward infinity, the polarization phenomenon influences the reliability of bit-channels, that are either completely noisy or completely noiseless; even more, the fraction of noiseless bit-channels equals the channel capacity . More formally, let W be a binary memoryless symmetric channel with input alphabet X = {0,1} and output alphabet Y, and let {W(y | x) : x ∈ X, y ∈ Y} be the transition probabilities. In order to quantify the reliability, i.e. the goodness, of the channel W, we use the Bhattacharyya parameter Z(W)∈[0,1], that is defined in equation (6).

$$Z(W) = \sum_{y \epsilon y} \sqrt{W(y|0)W(y|1)}$$

(6)

Hence, the good bit-channels are the ones that have the lowest Bhattacharyya parameter. For finite practical code lengths, the polarization of bit channels is incomplete, therefore, there are bit-channels that are partially noisy. The polar encoding process consists in the classification of the bit-channels in u into two groups as the K good bit-channels that will carry the information bits and are indexed by the information set I, and the N − K bad bit channels that are fixed to a predefined value (usually 0) and are indexed by the frozen set F.

As a first step, n_{PC} parity-check bits are inserted within the K information and CRC bits. The mother polar code is hence a (N, K ′) code, with K ′ = K + n_{PC}. To create the input vector u to be encoded, the frozen set of subchannels needs to be identified. The number and position of frozen bits depend on N, E, and the selected rate-matching scheme. Initially, the frozen set $\bar{Q}_F^N$ and the complementary information set $\bar{Q}_I^N$ are computed based on the polar reliability sequence $\bar{Q}_0^{N_{max}-1}$ and the rate matching strategy. Later, information bits are assigned to u according to the information set.

Finally, assistant parity check bits are calculated and stored in u, if necessary. In the following, we examine every step of the creation of the input vector u in more detail.

Frozen set $\bar{Q}_F^N$: The first bits identified in the frozen set correspond to the indices of the U = N − E untransmitted bits, i.e. the bits eliminated from the codeword by the ratematching scheme. These indices correspond to the first U or the last U codeword bits in the case of puncturing and shortening, respectively. If $\frac{K}{E} \leq \frac{7}{16}$ and hence the mother polar code has to be punctured.

2.6.3 PC bit calculation

The calculation of the PC bits is performed using equation 7 through a cyclic shift register of length 5, initialized to 0. Each PC bit is calculated as the XOR of the message bits assigned to preceding subchannels, modulo 5, excluding the previously calculated parity check bits. To summarize, a PC bit u_i , with i ∈ Q_{pc}^N, is calculated using equation (7).

$$u_i = \bigoplus_{j=\lfloor i_{pc}/5 \rfloor}^{q-1} u_{5j+p} \quad (7)$$

where q = ⌊i/5⌋, p = i mod 5 and $i_{PC} \in Q_{PC}^{N}$ is the highest index smaller than i for which i_{PC} mod 5 = p. If no such index exists, i_{PC} = 0.

2.7 Encoding

The encoding is performed by the multiplication using equation (3.1) where, $G_N = G_2^{\otimes n}$, with $G_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$. Encoding complexity can be proved to be $O(N \log N)$. The Encoding operation for both uplink and downlink is shown in the figure 4 and figure 5.

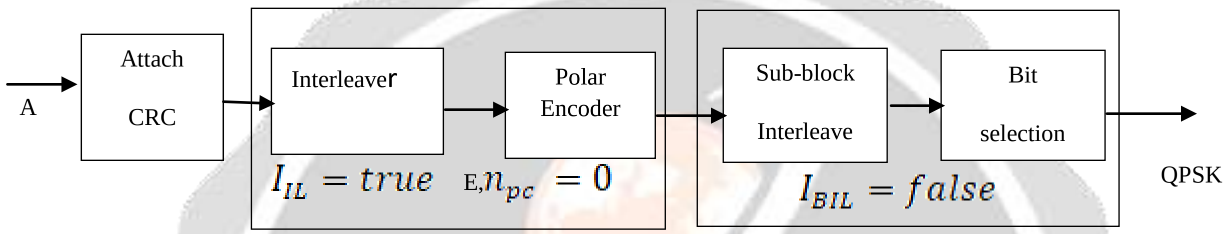

Fig 4: Polar Encoding for Downlink

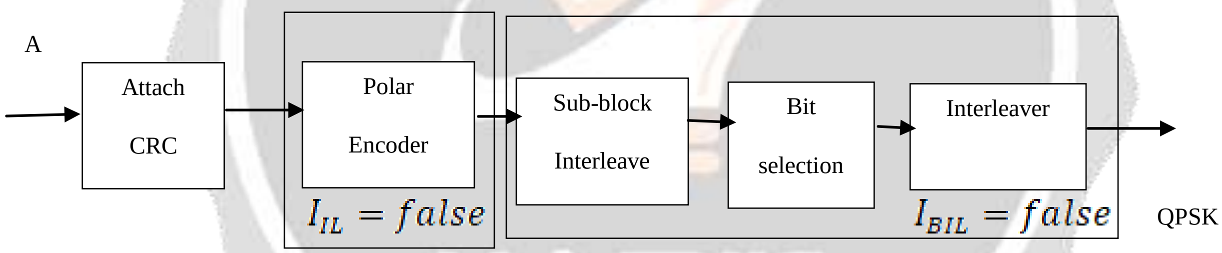

Fig 5: Polar Encoding for Uplink

2.8 Sub-Block Interleaver

The N encoded bits are then interleaved before performing the rate matching. This interleaver divides the N encoded bits stored in d into 32 blocks of length B $=\frac{N}{32}$ bits, interleaving the blocks according to a list of 32 integers P and obtaining the vector.

2.9 Rate matching

According to the definition, the length N of a polar code is limited to powers of two, while the code dimension K can assume any value smaller than N, since only the K most reliable bits will be used to carry the information. This is a limitation for typical 5G applications, where the amount of information K is fixed and a codeword of length N is needed to achieve the desired rate R = K/N.

Rate matching for polar codes becomes thus a length matching problem, and can be faced through classical coding theory techniques as puncturing and shortening. Both puncturing and shortening reduce the length of a mother code by not transmitting code bits in a predetermined pattern, called matching pattern; the difference lies in the meaning of the code bits belonging to the matching pattern. In puncturing, one or more code bits are not transmitted, which are treated as erased at the decoder.

In shortening, a sub-code is introduced such that one or more code bits assume a fixed value, typically zero, and not transmitted since they are known at the decoder. Rate matching alters the reliability of the subchannels, with an impact that depends on the strategy adopted. As a rule of thumb, it has been observed that for polar codes, shortening works better for high rates, puncturing for low rates. Puncturing deteriorates subchannel reliabilities; moreover, the erasures introduced by puncturing cause some bit channels, called incapable bits, to be completely unreliable.

It can be shown that U punctured code bits make exactly U subchannels incapable; the position of these bits can be calculated on the basis of the matching pattern. In order to avoid catastrophic error-correction performance degradation under SC decoding, incapable bits must be frozen in order to avoid random decisions at the decoder. Shortening improves the bit channel reliabilities by introducing overcapable bits, i.e. bits with (theoretically) infinite reliability: those bits are surely correctly decoded under SC, if the previous bits have been correctly decoded.

However, code bits in the matching pattern must depend on frozen bits only, which forces to freeze the usually most reliable subchannels. Three main strategies have been proposed to design the matching pattern. The first option is to design the frozen set of the mother polar code based on the matching pattern. According to this strategy, the matching pattern is initially generated according to some heuristic, then the subchannel reliabilities are calculated in order to find the optimal frozen set. This method is used for shortening and for puncturing of polar codes.

Where the DE/GA algorithm is run to find the optimal frozen set on the basis of different matching patterns. The result is a code with good block error rate (BLER) performance, at the cost of an higher complexity due to the reliability calculation. An alternative approach is to design the matching pattern on the basis of the frozen set. This significantly reduces the code design complexity at the cost of an increased BLER.

This approach is selected by 3GPP for the standardization of polar codes in 5G. Finally, joint optimization is used to design frozen set and matching pattern at the same time. Symmetries in the polar code structure reduce the number of matching patterns to be tested, however not enough to make this technique practical for application in 5G. Rate matching is performed by a circular buffer, and the codeword e of length E bits is calculated. Three possible rate-matching schemes are foreseen:

- Puncturing : If $E \leq N$ and $R \leq \frac{7}{16}$, the mother code is punctured. In this case, the first $U = N - E$ bits are not transmitted, hence $e_i = y_{i+U}$ for $i = 0, \ldots, E$.
- Shortening: if $E \leq N$ and $R > \frac{7}{16}$, , the mother code is shortened. In this case, the last $U = N - E$ bits are not transmitted, hence $e_i = y_i$ for $i = 0, \ldots, E$.
- Repetition: if $E > N$, the first $U = N - E$ bits are transmitted twice, hence $e_i = y_{i \bmod N}$ for $i = 0, \ldots, E$.

The operating principle of the rate-matcher based on the circular buffer.

2.10 Channel interleaver

Before passing the rate-matched codeword to the modulator, the bits in e are interleaved one more time using a triangular bit interleaver. This interleaver has been considered necessary to improve the coding performance of the coding scheme for high-order modulation. This interleaver is not applied for every use case, and it is triggered by the parameter I_{BIL}. In particular, the channel interleaver is activated for PUCCH and PUSCH UCIs (I_{BIL} = 1), while it is bypassed in the case of PBCH payloads and PDCCH DCIs (I_{BIL}= 0).

3. POLAR CODED MODULATION

If low-order modulation schemes like 4QAM are used, the BLER is not affected by the modulation scheme since all bits in modulated symbols have uniform reliability. Polar-coded modulation (PCM) was introduced for larger constellations, exploiting the polarization effect in the construction of polar codes for higher-order modulation. However, the canonical PCM requires the introduction of an additional polarization matrix whose size depends on the modulation scheme used. This results in increased latency due to a further decoding step.

The introduction of a channel interleaver emerged as a low complexity alternative to PCM; this technique, termed as bit interleaved polar-coded modulation (BIPCM), proved to improve the diversity gain under high-order modulation without increasing the code complexity. In BIPCM, the channel is considered as a set of parallel bit channels which can be combined with polar coded bits. Carefully mapping coded bits to modulation symbols offers a certain gain over the conventional random interleaving for high-order QAM over AWGN channels.

Moreover, interleavers designed to be adaptive to channel selectivity can achieve a remarkable diversity gain compared to random interleaving for polar coded OFDM transmission. The correlation between coded bits mapped into the same symbol allows to combine the demapping and deinterleaving units with the SC decoder to perform the decoding directly on the LLRs of the received symbols instead of the ones of the coded bits.

4. DECODING

The first decoding algorithm for polar codes is the successive cancellation algorithm, with which polar codes achieve channel capacity at infinite code length. It can be represented as a binary tree search, where the leaf nodes are the N bits to be estimated, and soft information about the received vector is input at the root node. The soft values received from the channel and the internally exchanged information to be logarithmic likelihood ratios (LLRs).

The SC algorithm can be implemented in both software and hardware with low complexity but its error-correction performance is mediocre when decoding practical code lengths. Thus, many attempts have been made to overcome this shortcoming. Eventually, a list-based decoding approach to polar codes (SCL) was introduced.

5. RESULTS AND DISCUSSION

The following results are the graph plot of SNR vs BER. The simulation output shows decoding operation with different list sizes.

5.1 Simulation outputs

The output BER vs E_b/N_t of LDPC, Convolutional LDPC and Systematic polar code is shown in Figure 6.

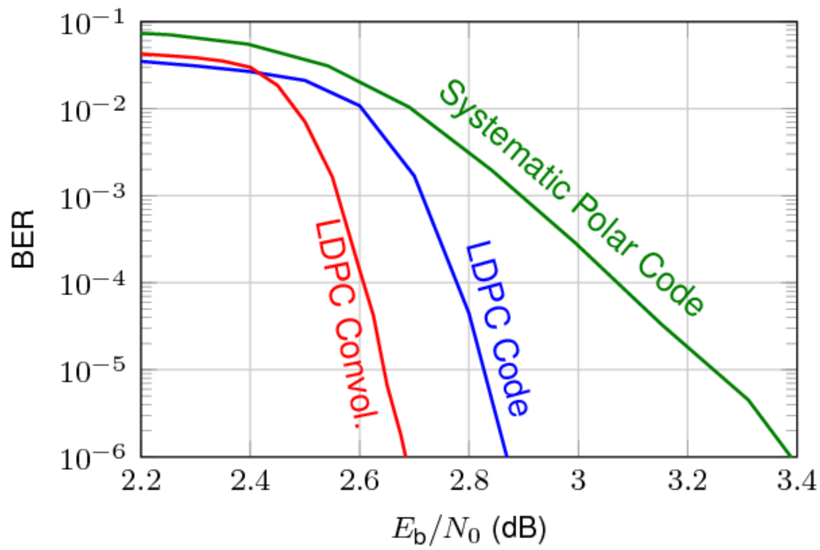

Fig 6: Comparison between LDPC Code, LDPC Convolutional Code and Systematic Polar Code

This figure shows that BER value of systematic polar codes is small than the LDPC code and conventional LDPC codes. The BER performance results of indicate the suitability of polar codes in a communication link and their implicit support for rate-compatibility at the bit-level granularity.The SNR vs BER plot for LDPC code with list size L=32, 16, 8, 4, 2, 1 is shown in Figure 7.

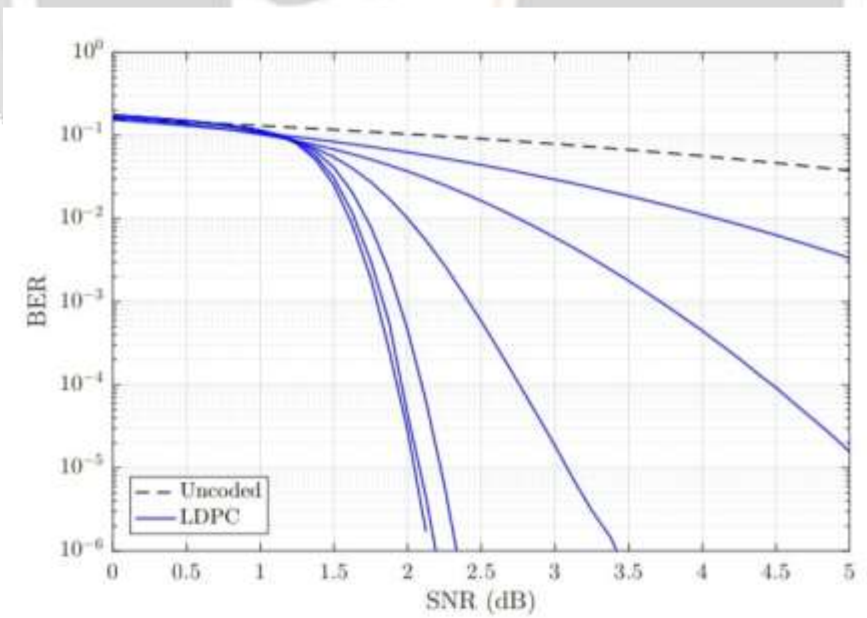

Fig 7: SNR vs BER for LDPC Code with L=32, 16, 8, 4, 2, 1

The SNR vs BER plot for polar code for L=1 is shown in Figure 8.

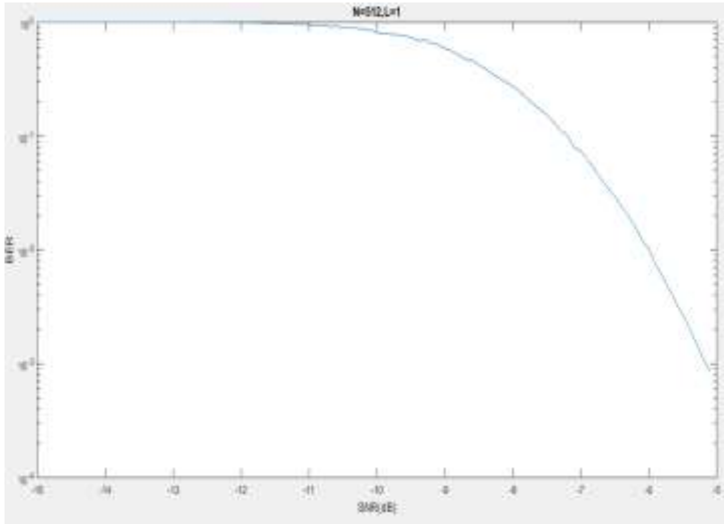

Fig 8: SNR vs BER for Polar Codes with L=1

The SNR vs BER plot for polar code for L=2 is shown in Figure 9.

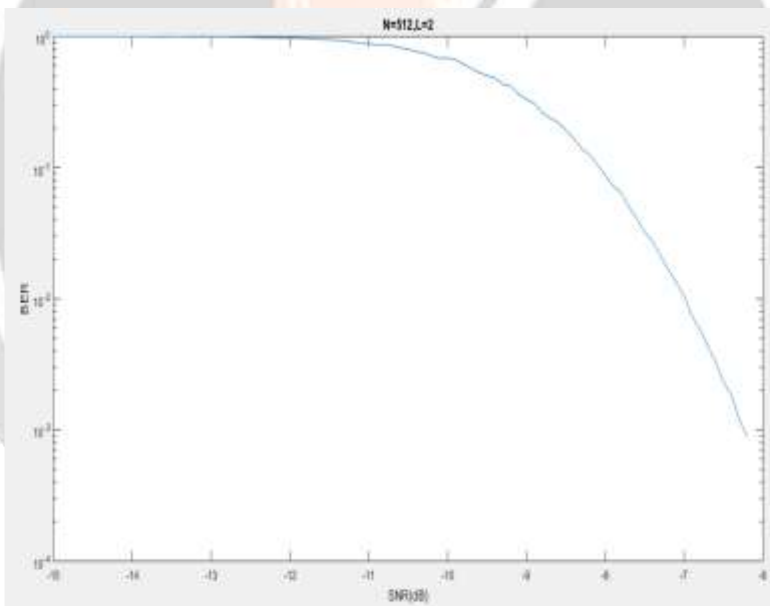

Fig 9: SNR vs BER for Polar Code with L= 2

The SNR vs BER plot for polar code for L=3 is shown in Figure 10.

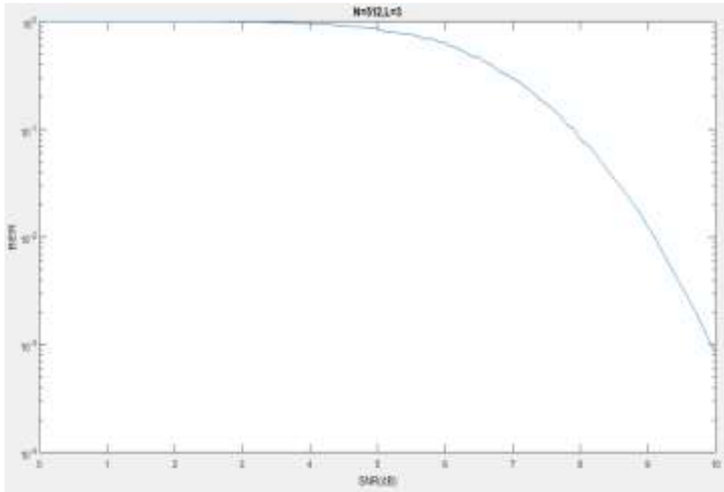

Fig 10: SNR vs BER for Polar Code with L= 3

6. CONCLUSION

The design of polar codes achieves capacity when block sizes are asymptotically large with successive cancellation list decoder. However, in block sizes that industry applications are operating, the performance of the successive is poor compared to the well-defined and implemented coding schemes such as LDPC and Turbo. Polar performance can be improved with successive cancellation list decoding which reduces the complexity in code construction, reduces the latency in the date transmission. The polar codes in 5G New Radio reduces the errors, complexity and increases rate of the channel. It enhances the mobile broad band communication system. Thus, polar coding technique is chosen to be the best choice for 5G signal transmission.

7. REFERENCES

1. Arikan, E., (2009) ‘Channel Polarization: A method for constructing Capacity-Achieving codes for Symmetric Binary-Input Memoryless Channels’, IEEE Transactions on information Theory, vol.55, No.7, pp.3051-3073.

2. Eslami, A., and Pishro-Nik, H., (2013) ‘On finite-length performance of polar codes: stopping sets, error floor, and concatenated design’, IEEE Transaction on Communictions, vol.61, No.3,pp.919-929.

3. Hanif, M., and Ardakani, (2017) M., ‘Fast successive-cancellation decoding of polar codes: identification and decoding of new nodes’, IEEE Communications Letters, vol. 21, No.11,pp.2360-2363.

4. Hashemi, S.A., Condo, C., and Gross, W.J., (2016) ‘Fast and flexible successive-cancellation list decoders for polar codes’, IEEE Transactions on Signal Processing , vol.65, No.21, pp.5756-5769.

5. Niu, K., Chen, K., (2012) ‘CRC-Aided Decoding of Polar Codes’, IEEE Communications Letters, vol. 16, No.10, pp.1668-1671.