An Optimized Method for Polar Code Construction

Issame El Kaime[1], Reda Benkhouya[2], Abdessalam Ait Madi[3], Hassane Erguig[4]
Advanced Systems Engineering Laboratory-National School of Applied Sciences, Ibn Tofail University Kenitra, Morocco[1]
MISC Laboratory-Faculty of Sciences, Ibn Tofail University, Kenitra, Morocco[2]
Advanced Systems Engineering Laboratory-National School of Applied Sciences, Ibn Tofail University Kenitra, Morocco[3]
Materials Physics and Subatomics Laboratory-Faculty of Sciences, Ibn Tofail University, Kenitra, Morocco[4]

***Abstract*—Polar codes are traditionally constructed by calculating the reliability of channels, then sorting them by intensive calculations to select the most reliable channels. However, these operations can be complicated especially when, the polar code length, N becomes great. This paper proposes a new low-complexity procedure for polar codes construction over binary erasure and additive white Gaussian noise (AWGN) channels. Using the proposed algorithm, the code construction complexity is reduced from O(Nlog N) to O(N), where $N=2^n$ ($n\geq1$). The proposed approach involves storing the classification of channels by reliabilities in a vector of length L, and then deriving the classification of M channels for every M where M<=L. The proposed method is consistent with Bhattacharya parameter based Construction and Density Evolution with Gaussian Approximation (DEGA) based construction. In this paper, the Successive Cancellation Decoding algorithm (SCDA) is used. Thanks to its low complexity and its high error-correction capability.**

Keywords—Polar codes; SNR; successive cancellation decoding; error correction; low-complexity; code construction; additive white Gaussian noise; Bhattacharya parameter; density evolution with Gaussian approximation

I. Introduction

It is usual for communication links to suffer from errors due to random noise, interference and malfunctioning devices, etc. To correct errors in channel coded data streams, a set of algorithmic operations is applied to the original data stream at the transmitter. A second set of algorithmic operations is applied to the received data stream at the receiver. Encoding and decoding operations at the transmitter and receiver are collectively called channel coding operations in channel coding terminology. Research in channel coding is focused on developing high performance channel codes that mitigate the effects of errors in communication links. A real challenge here is to accomplish this with sufficient simplicity to allow practical implementation in silicon technology. Everything depends on the complexity of a code, including its power consumption, memory requirements, computation power requirements, and latency, which determine whether or not a code is appropriate for any given scenario. Channel coding is somewhat revolutionized by polar codes.

The polar codes proposed by Arikan in 2008 can achieve that capacity of any binary discrete memoryless channel [1], researchers from all over the world have been interested in polar codes ever since their introduction in [1], polar codes can be used in lot of applications like cryptgraphy [2], speech communication [3], data storage [4]. Polar codes are also used to develop new block [5]. Polar codes are composed of three main stages; namely the construction, the coding and finally the decoding. The construction of polar codes is a crucial step as they affect the performance of polar codes. In polar codes construction, synthetic channels are evaluated by reliability. Good channels are selected for information transmission and the bad channels are frozen. Where polar codes are capable to achieving channel capacity for any binary-input discret memoryless channel [1].

Polar codes construction step encounters the following problem: given N code lengths and K information bit length,what is the best way to select K channels out of all the available ones, knowing that the remaining N-K bit channels are frozen and provided to the transmitter and receiver. An indication of the quality of a virtual bit channel $W_N^i()$ can be determined by using a variety of metrics. Polar codes take effect when the length of the code is very large, which implies a large number of computes to know the capacity of each channel, As a result, building polar codes is extremely difficult and requires a lot of resources. This paper considers the construction of polar codes over symmetric binary discrete memory-less channels by using a new method to reduce the complexity of the construction. Many methods have been developed previously in literature to build polar codes, Monte-Carlo simulations are proposed in [1] with a high complexity of O(TNlogN) where T indicates the number of iterations of Monte-Carlo simulations. In [6] and [7], polar codes construction is based on density evolution, where convolutions of functions are performed and numerical calculation precision is limited by the complexity of the process. In [8] bit-channel approximations are proposed with a Complexity under controlled conditions of $O(N.\mu^2 \log \mu)$ (μ a user-defined parameter that limits the number of output alphabets at each step of the approximation process). Another type of algorithm can construct polar codes using Gaussian approximation (GA) of additive white Gaussian noise (AWGN) channels [9]–[11], this approximation function [11] inherently limits the GA method, with some restrictions on the length of blocks [11]. Bhattacharya paremeters are used to construct polar codes in[1], other constructions methods with variable performance and complexity are located in [12]–[14].

This paper describes an efficient method of constructing polar codes to reduce their computational complexity, if the method suggested in this paper is compared to other ones in the literature, the method presented here is characterised by a reduced complexity.

In the following sections of this paper, polar codes are described, including their background, with a focus on the concept of channel polarization. Section III presents the traditional method of constructing polar codes, while Section IV provides numerical results that demonstrate the effectiveness of the proposed method compared to traditional methods and state-of-the-art techniques. Finally, the conclusion summarizes the key findings of the study and discusses potential future research directions.

II. Polar Codes

In general, all channel coding technologies work in quite similar ways, even if the excellent performance of turbo codes and LDPC (Low Density Parity Check) codes in practice, none of the last codes can be demonstrated to attain the capacity of channels exempt the binary erasure channel (BEC). Polar codes are members of the block code family since they operate on blocks of symbols/bits. To construct polar code, two key operations are required: channel combining and channel splitting. The channel combining process, carefully selected combinations of bits are mapped to specific channels.

Let consider N bits to be sent over discrete binary channel without memory (B-DMC) W. Each transmission represents a use of W, which means that each bit is passed through a copy of W as shown in Fig. 1.

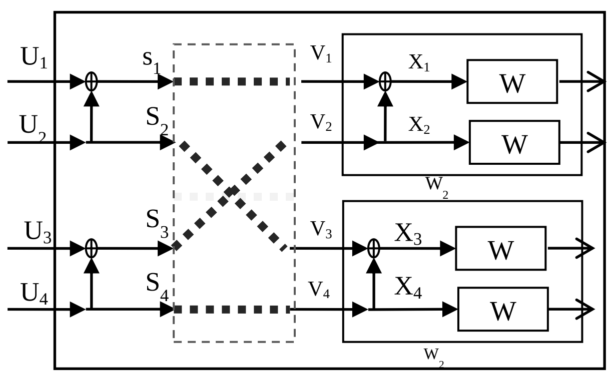


Fig. 1. Combination of two copies of W to form W2.

The map in Fig. 1 between the input u_2^1 and the vector X_2^1 can be represented by the equation (1):

$$X_2^1 = u_2^1 * G_2 \tag{1}$$

Where G_2 is the basic matrix:

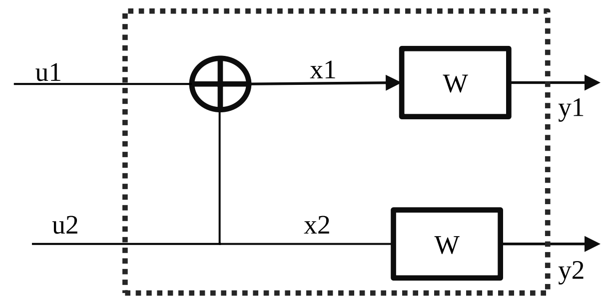
$$G_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \tag{2}$$

The same operation is repeated with W_2 as the basic element to produce W_4 like presented in Fig. 2. Generally At this step, the combined channels W_N is the virtual channel that map the input data u_N^i to the output y_N^1. W_N^i are divided into a set of N bit input channels $W_N^i()$.

These virtual channels $W_N^i()$ are characterized by the transition probabilities provided by equation (3).

$$W_N^{(i)}\left(y_1^N, u_1^{i-1} \middle| u_i\right) \triangleq \sum_{u_{i+1}^N \in X^{N-i}} \frac{1}{2^{N-1}} W_N\left(y_1^N \middle| u_1^N\right) \tag{3}$$

$$(W_N^{(i)}) : X \rightarrow Y^N \times X^{i-1}, 1 \leq i \leq N \tag{4}$$



$$W_2^{(1)}: u_1 \rightarrow (y_1, y_2)$$

Fig. 2. Combination of two copies of W2 to form W4.

III. The Native Construction

As mentioned earlier, building polar codes is like finding all the best positions of reliable bits positions. The selection of the information bits is one of the most important steps in polar coding. Polar codes are originally constructed using simple bounds on Bhattacharyya parameter bit channels [1], The Bhattacharyya parameter is linked to the Bhattacharyya distance, which measures the similarity of the probability distributions of two symbols [15]. It has been widely used to produce good polar codes because of its simplicity. By creating a system allowing one to access each bit-channel individually, one can send data only through those for which the Bhattacharyya parameter is close to 0. Bhattacharyya's parameter Z( ) defined in (5) provides an upper bound on the error probability of transmission over W with maximum likelihood (ML) decisions when the channel is used only once.

$$Z(W_N^{(i)}) = \sum_{y_1^N \in Y^N} \left| X_{i+1}^N, y_1^N \right| \sqrt{W_N^{(i)}(y_1^N, 0_1^{i-1}|0) W_N^{(i)}(y_1^N, 0_1^{i-1}|1)} \tag{5}$$

Accordingly, channels with Z(W_N^i)<ε are almost noiseless, whereas channels with Z W(W_N^i)>1-ε are almost pure-noise channels where 0<ε<1 [15]. However, it has been cited that the parameters' updates succeeded with equality just for the Binary Erasure Channel.

A. Density Evolution with Gaussian Approximation (DEGA) Construction

Over AWGN channel the Density Evolution with Gaussian Approximation (DEGA) is the famous construction for polar codes. From the channel stage towards the decision stage, DEGA attempts to evolve the densities of the LLRs via the decoder. Regardless, precise if the begin is with Gaussian densities at the channel stage, the consequent densities at the following stages are not Gaussian anymore. The DEGA decoder relaxes this by supposing they are approximately Gaussian, and consequently it only tracks their mean and variance throughout the approach. The relationship between mean and variance should survive at all stages [10]. The variance and the mean is defined in (6) where the LLR is defined in (7).

$$m = E\{L\} = \frac{2}{\sigma^2}\ , \quad \mathrm{var}\{L\} = \frac{4}{\sigma^2} = 2m \qquad (6)$$

The next update is accomplished by DEGA[9].

$$m_i^{(s)} = \begin{cases} \phi^{-1}\left(1-\left(1-\phi\left(m_i^{(s-1)}\right)\right)^2\right), if\ \ node\ f, \\ 2m_i^{(s-1)}, if\ node\, g, \end{cases} \qquad (7)$$

The function ϕ is given by (8).

$$\phi(x) = \begin{cases} \exp(0.4527x^{0.86}+0.0218),\ 0<x<10 \\ \sqrt{\frac{\Pi}{x}}\exp\left(-\frac{x}{4}\right)\left(1-\frac{10}{7x}\right), x \geq 10 \end{cases} \qquad (8)$$

IV. Numerical Results and Discussion

This section analyses the classifications of polar codes virtual channels by reliability for different code lengths over AWGN and BEC, the information length is not tested because the construction of polar codes depends on the code length.

A. *Classification of Polarizing Channels over BEC*

This part starts by the comparison by reability of 8 BEC polarizing channels with erasure probability 0.1,0.2,0.3,0.4,0.5,0.6,0.7 and 0.8, the results are shown in Fig. 3.

The simulated results that are calculated using the relation (17) of [1] indicates that virtual channels are classified in the same way independently of the erasure probability. Fig. 3 shows that for all erasure probabilities, virtual channels are classified as below 8->7 ->6 ->4 ->5 ->3->2 ->1 (from the most reliable to the least accessible).

Fig. 4 shows the reliability of 16 virtual channels for the following erasure probabilities [0,1,...,0,9]. Fig. 4 shows that the classification of 16 virtual channels is the same for all erasure probabilities, the classification by consequence is

16-> 15 ->14 -> 12 -> 8 -> 13 -> 11 -> 10 -> 7 -> 6 -> 4 -> 9 -> 5 ->3->2->1.

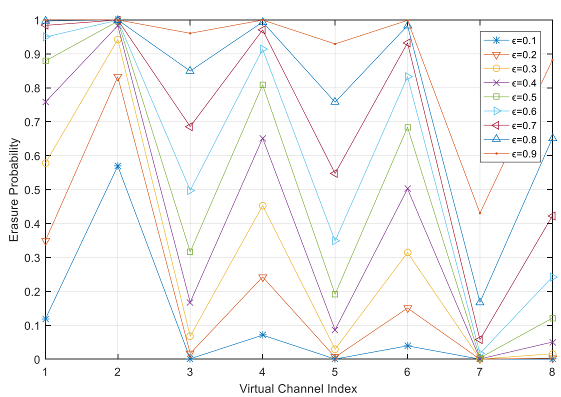


Fig. 3. The Bhattacharyya for 8 BEC polarizing channels.

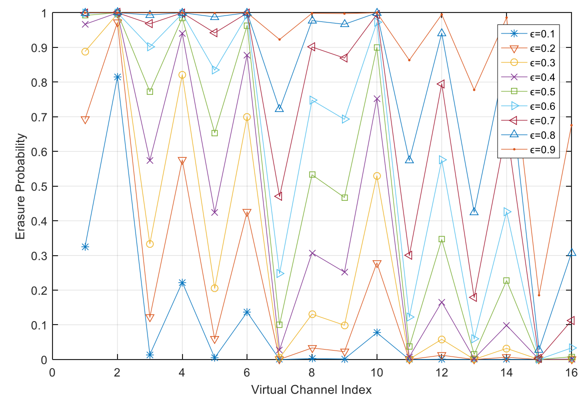


Fig. 4. Comparison of 16 BEC polarizing channels in terms of reliability with erasure probability [0.1, 0.2, …, 0.9].

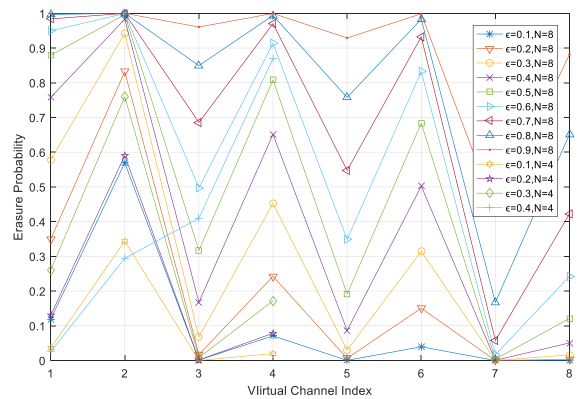


Fig. 5. Classification of channel by reliability for N=8 and N=4.

Fig. 5 compares the classification of virtual channels for different code lengths N=8 and N=4.

The results shown in Fig. 5 proof that the classification of virtual channels with index less than 4 is the same in both codes.

Table I provides the classification of virtual channels by reliability for several code lengths (N=1,2,4,8,16,32) where the erasure probability ε=0,25. For ease of analysis, each channel index is characterized by the same color in each code length. The table shows that the classification is the same when the length of the code changes.

B. *Classification of Polarizing Channels under AWGN*

This section analyses channel classifications by reliability for different code lengths over AWGN, the Bhattacharyya parameters are used to evaluate the reability for each virtual channel, note that the same tests can be applied with Gaussian density approximation method.

In the beginning, reliability of eight virtual channels are compared under different values of signal to noise ratio (SNR). Fig. 6 shows the reliability of eight polarizing channels using SNR -2, -1, 0, 1, 2 and 3. It also worth nothing that the classification of the eight polarizing channels by reliability remains the same when SNR changes.

TABLE I. CLASSIFICATION OF VIRTUAL CHANNEL BY BHATTACHARYYA FOR DIFFERENT CODE LENGTH N=2,4,8,16,32 WHERE $\varepsilon = 0.25$

N	Channel index																														
2	1	0																													
4	3	2	1	0																											
8	7	6	5	3	4	2	1	0																							
16	15	14	13	11	7	12	10	9	6	5	3	8	4	2	1	0															
32	31	30	29	27	23	15	28	26	25	22	21	19	14	13	11	24	7	20	18	12	17	10	9	6	5	16	3	8	4	2	1	0

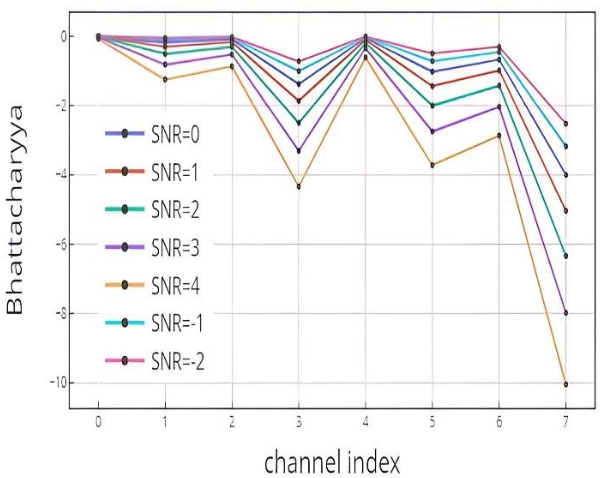


Fig. 6. AWGN polarizing channels reliability for N=8.

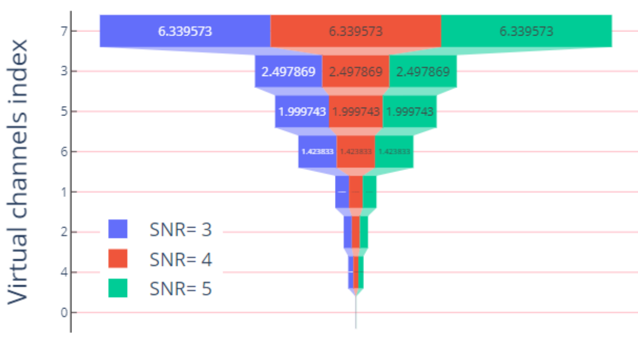


Fig. 7. Reliability based classification of polarizing channels.

Fig. 7 shows also the classification of eight polarizing channels according to their reliability where SNR = 3, 4 and 5. It is clear from Fig. 7 that the classification of polarizing channels remains the same when SNR changes. Next, to evaluate the effect of code length in the classification of virtual channels by reliability, the simulation examines the classification of reliability for the virtual channels when the code length is equal to 32.

The simulation in Fig. 8 shows the classification of 32 polarized channels reliability where the SNR changes from 1 to 5. From the last figure, the classification of channels by reabilities remains the same when the SNR changes.

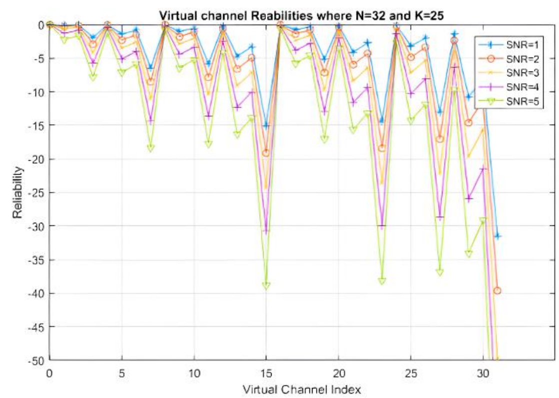


Fig. 8. Classification of 32 virtual channel for different SNR.

Fig. 9 compares the reliability of virtual channels for the different length codes N=8 and N=16. The result shows that the classification of virtual channels with an index less than 8 is the same in both polar codes.

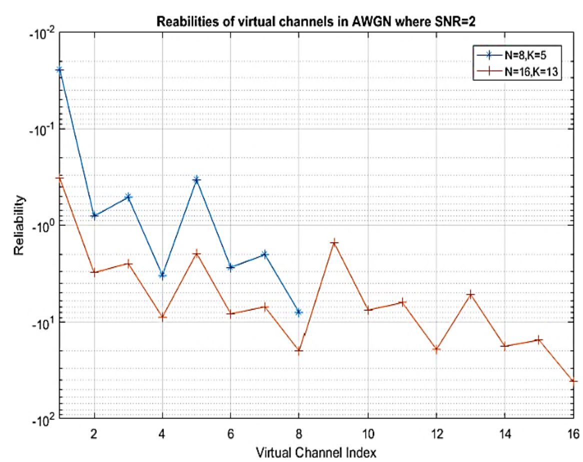


Fig. 9. Classification of channel reliability for N=8 and N=16.

TABLE II. CLASSIFICATION OF VIRTUAL CHANNEL BY RELIABILITY FOR DIFFERENT CODE LENGTH N=2,4,8,16

N=2	0	1														
N=4	0	2	1	3												
N=8	0	4	2	1	6	5	3	7								
N=16	0	8	4	2	1	12	10	6	9	5	3	14	13	11	7	15

The Table II generalizes the simulation of Fig. 9 by presenting the classification of virtual channels by reliability for several code lengths (N=2,4,8,16) where the SNR=2. For ease of analysis, each channel index is characterized by the same color in each code length. The table shows that the classification is the same when the length of the code changes.

Mathematically the purposed procedure to construct polar codes with low complexity well be described below in algorithm 1. This algorithm uses linear search where every element in a reliability vector is checked, starting at 0 and going through each element until the desired element is found. Otherwise, the search continues until the end of the list.

Algorithm 1: Find out all good channels index and bad ones

```
Input :
        Channel index vector classified by reliability I with length N_I
        polar code length N where N≤N_I
output:
        S:virtual channel classified by reliability for polar
        code with length Nr
for k=0 to N do
        for J=0 to N_I do
                if the value of N_I[J] less or equal N
                        S[k]= N_I[J]
                endif
        endfor
endfor
Return S;
```

The latest algorithm uses linear search where each item in a list is checked, starting at 0 and going through each item until the desired item is found. Otherwise, the search continues until the end of the list. There is no better search algorithm. The complexity of the newly developed build algorithm is the same as the complexity of the search algorithm, which is O(N).

To better understand the algorithm designed, let’s take an example of building polar codes, assumes that Gaussian density was adopted firstly to construct polar codes. The new purposed algorithm for constructing polar codes can be applied here by storing in a particular vector (R), the classification of virtual channels by reliability from the least reliable to the most reliable. Next the reliability calculation for each virtual channel will not be done in the next polar code construction, there is a need to select the most reliable channels of the R vector with an index less than the length of the new polar codes.

V. Conclusion

The previous approaches of polar codes construction require one to one compute of the reliability for all synthetic channels and use only those that are sufficiently reliable [2]. For the gain in complexity and resources, it is worthwhile to perform an optimized construction polar codes algorithm. the proposed method decreases the computation complexity in the construction of Polar Codes. Clearly, the complexity is reduced to O(N). Note that when using this optimized construction approach, extra memory is allocated to store the vector of reliability.

The memory ressources play a crucial part in the outcome of this study, these limitations do have a significant impact on the primary finding, future research could seek to reduce memory's impact by choosing the best value of the vector reliability length for every application.

References

[1] E. Arikan, « Channel Polarization: A Method for Constructing Capacity-Achieving Codes ».

[2] J. Liu, Y. Wang, Z. Yi, et Z. Lin, « polarRLCE: A New Code-Based Cryptosystem Using Polar Codes », Security and Communication Networks, vol. 2019, p. e3086975, 2019, doi: 10.1155/2019/3086975.

[3] S. Zhao, P. Shi, et B. Wang, « Polar codes and its application in speech communication », in 2011 International Conference on Wireless Communications and Signal Processing (WCSP), 2011, p. 1-4. doi: 10.1109/WCSP.2011.6096731.

[4] K. N. Tunuguntula, « Polar codes for data storage and communication network applications », UC San Diego, 2022.

[5] D. Khebbou, R. Benkhouya, I. Chana, et H. Ben-azza, « Finding Good Binary Linear Block Codes based on Hadamard Matrix and Existing Popular Codes », IJACSA, vol. 12, no 11, 2021, doi: 10.14569/IJACSA.2021.0121150.

[6] R. Mori et T. Tanaka, « Performance and construction of polar codes on symmetric binary-input memoryless channels », in 2009 IEEE International Symposium on Information Theory, Seoul, South Korea, 2009, p. 1496-1500. doi: 10.1109/ISIT.2009.5205857.

[7] R. Mori et T. Tanaka, « Performance of Polar Codes with the Construction using Density Evolution », IEEE Commun. Lett., vol. 13, no 7, p. 519-521, 2009, doi: 10.1109/LCOMM.2009.090428.

[8] Ido Tal and Alexander Vardy, How to Construct Polar Codes.

[9] P. Trifonov, « Efficient design and decoding of polar codes », IEEE Transactions on Communications, vol. 60, no 11, p. 3221-3227, 2012.

[10] D. Wu, Y. Li, et Y. Sun, « Construction and block error rate analysis of polar codes over AWGN channel based on Gaussian approximation », IEEE Communications Letters, vol. 18, no 7, p. 1099-1102, 2014.

[11] J. Dai, K. Niu, Z. Si, et J. Lin, « Evaluation and optimization of Gaussian approximation for polar codes », in Proceedings of the American Society for Information Science and Technology, 2016, vol. 51, no 1, p. 1-2.

[12] R. Pedarsani, S. H. Hassani, I. Tal, et E. Telatar, « On the construction of polar codes », in 2011 IEEE International Symposium on Information Theory Proceedings, 2011, p. 11-15.

[13] D. Kern, S. Vorkoper, et V. Kuhn, « A new code construction for polar codes using min-sum density », in 2014 8th International Symposium on Turbo Codes and Iterative Information Processing (ISTC), Bremen, Germany, 2014, p. 228-232. doi: 10.1109/ISTC.2014.6955119.

[14] H. Li et J. Yuan, « A practical construction method for polar codes in AWGN channels », in IEEE 2013 Tencon-Spring, 2013, p. 223-226.

[15] R. Benkhouya, I. Chana, et Y. Hadi, « Study of the operational SNR while constructing polar codes », IJECE, vol. 10, no 3, p. 3200, 2020, doi: 10.11591/ijece.v10i3.pp3200-3207.