Research on Polar Code Construction Algorithms under Gaussian Channel

Jianping Li
Communication University of China
College of Information Engineering
Beijing, China
Turbo_code@163.com

Man Hu
Communication University of China
College of Information Engineering
Beijing, China
532695504@qq.com

Zhiyuan Cheng
Communication University of China
College of Information Engineering
Beijing, China
Sylvia9645@163.com

Abstract— Polar code is widely accepted due to its excellent error correction performance, low complexity coding and decoding structure. However, the imperfect performance in the short code length of polar codes limits its further application. A significant way to improve the performance of short polar code is to choose fit sub-channels for information bits accurately and effectively. After analyzing some construction algorithms for polar codes under Gaussian channel, comparing their complexity and their BER performance, this paper proposes an improved polarization weight (PW) algorithm, which can reduce the bit error rate.

Keywords—polar codes construction; Density Evolution; Gaussian approximate (GA); polarization weight (PW) algorithm

I. INTRODUCTION

Arikan proposed an encoding method for a group of independent binary symmetric input discrete memoryless channels (B-DMCs), where each sub-channel exhibits different reliability [1]. As the code length increases, the sub-channels show a polarization phenomenon called channel polarization. Recently, polar codes have been regarded as one of the channel coding schemes in the 5G wireless communication system due to its excellent performance [2].

Reference [1] pointed out that using good channels to transmit information bits, and setting a fixed value (for example, set to 0) for poor channels. The combination and splitting of polar code channel can make N independent channels correlate with each other with the same parameters. Although each sub-channel capacity changes, the total channel capacity remains unchanged. This provides an effective method by selecting some sub-channels with big channel capacity to transmit information bits, which is called as polar construction.

The selection of sub-channels can directly determine the performance of successive cancellation decoding algorithm and the asymptotic behavior of polar codes. Arikan gives a detailed construction method of polar codes in BEC channels, which can easily find good or bad channels by calculation [3]. However, the complexity of this method exponentially increases with the increase of code length in Gaussian channel. Some researches on polar code construction under Gaussian channel have been done, such as Monte Carlo algorithm [1], Density Evolution algorithm [4], Gaussian Approximation algorithm [5], PW algorithm [6], etc. We have presented a comprehensive survey of these well-known construction algorithms. We then proposed an improved PW algorithm after comparing the BER of four channel construction algorithms. Numerical results showed that improved PW algorithm performs better than Original PW algorithm. When BER=10^{-3}, the system can obtain about 0.3 dB SNR gain.

II. POLAR CODE CONSTRUCTION ALGORITHMS UNDER GAUSSIAN CHANNEL

A. Monte Carlo method(MC)

Let W: X → Y denote a B-DMC with input alphabet X and output alphabet Y. The channel transition probabilities are given by W (y |x), x ∈ X and y ∈ Y. Given the code length N = 2^n, n = 1, 2, · · ·, the information length K, the polar coding is described as [1]. After the channel combining and splitting operations on N independent duplicates of W, we obtain N successive uses of synthesized binary input channels $W_N^{(j)}$, j = 1, 2, · · ·, N, with transition probabilities $W_N^{(j)}(y_1^N, u_1^{j-1} \mid uj)$.

The literature [1] pointed out that an approximate code construction algorithm can be used. So we can use the average of formula (1) as Bhattacharyya value of the ith sub-channel, that is $Z(W_N^{(i)})$. The specific algorithm is as follows:

$$\sqrt{\frac{W_N^{(i)}(Y_1^N, U_1^{i-1} | U_i \oplus 1)}{W_N^{(i)}(Y_1^N, U_1^{i-1} | U_i)}} \quad (1)$$

1) All 0-code words are transmitted into the Gaussian channel after BPSK modulation.

2) Calculate the initial likelihood based on the output of the channel, that is

$$L_N^{(i)}(y_i)=\frac{W(y_i|0)}{W(y_i|1)} \tag{2}$$

3) Calculate $L_N^{(i)}(y_1^N,\hat{u}_1^{i-1})$ according to the formula below :

$$L_N^{(2j-1)}(y_1^N,\hat{u}_1^{2j-2}) \overset{\Delta}{=} \frac{L_{N/2}^{(j)}(y_1^{N/2},\hat{u}_{1,o}^{2j-2}\oplus\hat{u}_{1,e}^{2j-2})L_{N/2}^{(j)}(y_{N/2+1}^{N},\hat{u}_{1,e}^{2j-2})+1}{L_{N/2}^{(j)}(y_1^{N/2},\hat{u}_{1,o}^{2j-2}\oplus\hat{u}_{1,e}^{2j-2})+L_{N/2}^{(j)}(y_{N/2+1}^{N},\hat{u}_{1,e}^{2j-2})}$$

$$L_N^{(2j)}(y_1^N,\hat{u}_1^{2j-1}) \overset{\Delta}{=} L_{N/2}^{(j)}(y_1^{N/2},\hat{u}_{1,o}^{2j-2}\oplus\hat{u}_{1,e}^{2j-2})^{1-2\hat{u}_{2j-1}}\bullet L_{N/2}^{(j)}(y_{N/2+1}^{N},\hat{u}_{1,e}^{2j-2}) \tag{3}$$

4) If $L_N^{(i)}(y_1^N,\hat{u}_1^{i-1})\geq 1.0$, the value of this bit is 0. According to formula (1), the ith sub-channel:

$$Z(W_N^{(i)})=\sqrt{\frac{W_N^{(i)}(Y_1^N,U_1^{i-1}|0\oplus 1)}{W_N^{(i)}(Y_1^N,U_1^{i-1}|0)}} =\sqrt{\frac{W_N^{(i)}(Y_1^N,U_1^{i-1}|1}{W_N^{(i)}(Y_1^N,U_1^{i-1}|0)}}=\sqrt{1/L_N^{(i)}(y_1^N,\hat{u}_1^{i-1})} \tag{4}$$

If $L_N^{(i)}(y_1^N,\hat{u}_1^{i-1})<1.0$, the value of this bit is 1. According to formula (1), the ith sub-channel:

$$Z(W_N^{(i)})=\sqrt{\frac{W_N^{(i)}(Y_1^N,U_1^{i-1}|1\oplus 1)}{W_N^{(i)}(Y_1^N,U_1^{i-1}|1)}} =\sqrt{\frac{W_N^{(i)}(Y_1^N,U_1^{i-1}|0}{W_N^{(i)}(Y_1^N,U_1^{i-1}|1)}}=L_N^{(i)}(y_1^N,\hat{u}_1^{i-1}) \tag{5}$$

5) Repeat calculation several times, for example 1000 times, that means, 1000 code words are transmitted, and take the average of 1000 Z values, then we can obtain the approximate Bhattacharyya value of the sub-channel. Sort these values in ascending sequence. Select the k sub-channels with the smallest z value as the information channels.

B. Density Evolution(DE)

Using the Density Evolution algorithm to select sub-channels is as follows:

1) All 0-code words are transmitted under the Gaussian channel after BPSK modulation.

2) Calculate the initial likelihood based on the channel output.

$$.L_l^{(i)}(y_i)=\log\frac{W(y_i|0)}{W(y_i|1)} \tag{6}$$

3) Calculate $L_N^{(i)}(y_1^N,\hat{u}_1^{i-1})$, which

$$L_N^{(2j-1)}(y_1^N,\hat{u}_1^{2j-2})=2\tanh^{-1}(\tanh(L_{N/2}^{(i)}(y_1^{N/2},\hat{u}_{1,e}^{2j-2}\oplus\hat{u}_{1,o}^{2i-2})/2) \times\tanh(L_{N/2}^{(i)}(y_{N/2+1}^{N},\hat{u}_{1,e}^{2i-1})+(-1)^{\hat{u}2i-1}L_{N/2}^{(j)}(y_1^{N/2},\hat{u}_{1,o}^{2j-2}\oplus\hat{u}_{1,e}^{2j-2}\oplus\hat{u}_{1,o}^{2i-2}) \tag{7}$$

4) Calculate the error probability function value of these sub-channels according to $L_N^{(i)}(y_1^N,\hat{u}_1^{i-1})$, the formula is as follows:

$$P_e(i)=\frac{1}{2}\int_{-\infty}^{+\infty}l_i(x)\bullet e^{-\frac{1}{2}(x+|x|)}dx \tag{8}$$

5) Sort the error probability of sub-channels. And select the k sub-channels with the lowest bit error rate as the information channels.

C. Gaussian Approximation(GA)

1) Under the Gaussian channel (AWGN), the initial likelihood of the channel is calculated according to the output of the channel, which is

$$L_N^{(i)}(y_i)=\frac{W(y_i|0)}{W(y_i|1)} \tag{9}$$

In above formula, $L_N^{(i)}(y_i)\sim N(\frac{2}{\sigma^2},\frac{4}{\sigma^2})$.

2) Calculate E[Ln(i)], according to the formula below:

3) If i is odd, then

$$E[L_n^{(2i-1)}]=\varphi^{-1}(1-(1-\varphi(E[L_{n/2}^{(i)}]))^2) \tag{10}$$

If i is even, then

$$E[L_n^{(2i)}]=2\,E[L_{n/2}^{(i)}] \tag{11}$$

Which,

$$\varphi(x)=\begin{cases}1, & x=0\\ 1-\frac{1}{\sqrt{4\pi x}}\int_{-\infty}^{\infty}tanh\frac{u}{2}e^{-\frac{(u-x)^2}{4x}}dx, & x>0\end{cases} \tag{12}$$

During the simulation, we can use an approximate function φ(x), which

$$\varphi(x)=\begin{cases}\exp(-0.4527x^{0.86}+0.0218), & 0<x<10\\ \sqrt{\frac{\pi}{x}}\exp\left(-\frac{x}{4}\right)\left(1-\frac{10}{7x}\right), & x\geq 10\end{cases} \tag{13}$$

$E[L_1^{(1)}]=2/\sigma^2$, σ is the noise variance of the channel.

4) Calculate the error probability of each sub-channel.

5) Sort the error probability of sub-channels. And select the k sub-channels with the lowest bit error rate as the information channels. *probability of each sub-channel.*

D. polarization weight (PW) algorithm

Literature [7] proposed a new polar code construction algorithm called PW algorithm. The literature [6] introduced two important properties of Universal Partial Order (UPO): nesting and symmetry. These two properties are the basis for the construction of polar code sequences in a recursive manner. Then the PW algorithm is introduced, in which the β parameter need to be carefully selected, and the value of β directly determined the order of the synthesized channels. Finally, literature [6] concluded that the best value of β is 1.1892, that is, $\beta = 2^{1/4}$. Both progressive analysis and numerical simulation results demonstrate that the PW algorithm achieved the same performance as GA but with much lower complexity.

B-expansion: Principle of PW algorithm

Definition: (PW algorithm) Consider a synthetic channel index x whose n-bit binary extension is $B = (b_{n-1}, ..., b_1, b_0)$ with the most significant bit on the left. The polarization weight is defined as

$$f^{pw}: x \to \sum_{i=1}^{n} b_i \beta^i \tag{14}$$

The value of β need to choose carefully. The literature [6] suggests $\beta=2^{1/4}$.

A large value of w_x indicates a higher reliability of the synthetic channel. By arranging w_x in increasing order, we can obtain the polar code sequence.

The advantage of the PW algorithm is that it provides a simple and low-complexity full-alignment algorithm for the reliability of the synthetic channel, and it can always maintain a nesting when the code length 2n increases from n=1 to $n\to+\infty$. Literature [6] reached a conclusion that: PW algorithm performs equally well as GA but with much lower complexity.

E. Complexity comparison

The Monte Carlo algorithm has the greatest complexity O(M· N· logN) among all, where M is the number of iterations of the Monte-Carlo simulation. When the code length is large, the calculation amount is very large. The Density Evolution algorithm has the second largest complexity O (N· μ2logμ) when the channel output has μ symbols. The Gaussian Approximation algorithm takes a complexity of O(N) function computations, but involves relatively higher complexity function computations, it enjoys the second least complexity. The PW algorithm takes a complexity of O(N) function computations, too. But its calculation is relatively simple without high-dimensional calculation. As shown in figure 1, polar code has almost the same BER performance under the four construction algorithms. And the PW algorithm has the least complexity. So, it is more valuable to improve PW algorithm.

III. Improved PW Algorithm

We use Gaussian Approximation algorithm and PW algorithm to select the sub-channels respectively, which have some different sub-channels. These different sub-channels appear around the N/2th sub-channel. These sub-channels are the least reliable of the selected channels. Therefore, information bits transmitted by these sub-channels are more prone to errors. Thus, an improved PW algorithm is proposed in this paper.

Among information sub-channels obtained by PW algorithm, sub-channels around the N/2th sub-channel have the lowest reliability, which are the easiest to get wrong. Use these error-selected sub-channels to transmit information bits results in bit error rate. Therefore, we propose an improved PW algorithm. First, we use the original PW algorithm to calculate W_X of all sub-channels. Secondly, sort W_X in descending order, and select the first 0.461*N sub-channels to transmit information bits, the last 0.461*N sub-channels to transmit frozen bits. The third step, select the sub-channels during [0.461*N, 0.539*N] and calculate the error probability of these sub-channels. The fourth step, sort the error probability of sub-channels which are selected by step 3 in increasing order. Last, select the first 0.039*N sub-channels as information channels. The above 0.078*N, 0.461*N, 0.539*N, etc. If decimal, it is rounded to an integer.

The steps of improved PW algorithm are as follows:

1) Calculate W_X of all sub-channels.

2) Sort W_X in descending order, select the first 0.461*N sub-channels to transmit information bits, and the last 0.461*N sub-channels to transmit frozen bits.

3) Select the sub-channels during [0.461*N, 0.539*N], and calculate the error probability of these sub-channels.

4) Sort the error probability of sub-channels which are selected by step 3) in increasing order.

5) Select the first 0.039*N sub-channels as information channels.

The TABLE Ⅰ shows the different sub-channels obtained by Gaussian Approximation and PW algorithm. X indicates that there are X different sun-channels when the code length is N. Y indicates that the maximum position of these sub-channels minus the minimum position.

TABLE I. Different sub-channels of information bits obtained by the two algorithms

N	256	512	1024	2048	N
x	1	8	16	32	x
Y	20	40	80	160	Round(0.078* N)

IV. Simulations

A. BER Simulation under Gaussian Channel

Fig. 1 shows that polar code has almost the same BER performance under the four construction algorithms.

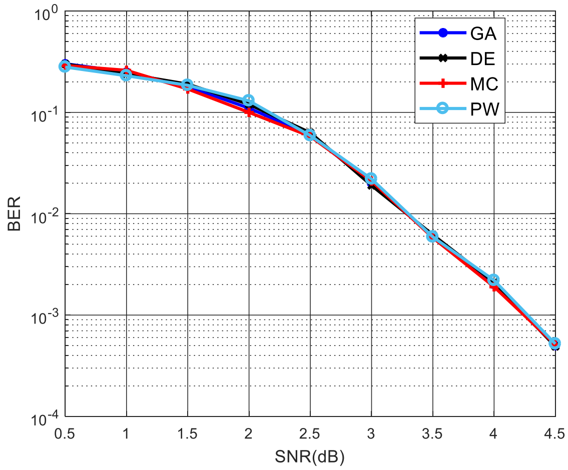

Fig. 1. BER vs. SNR for different polar construction algorithms under Gaussian channel

B. *PW algorithm and improved PW algorithm*

Fig. 2 shows that improved PW algorithm performs better than Original PW algorithm. As the SNR increases, the BER of the improved PW algorithm is effectively improved. When BER=10^{-3}, the system can obtain about 0.3 dB SNR gain.

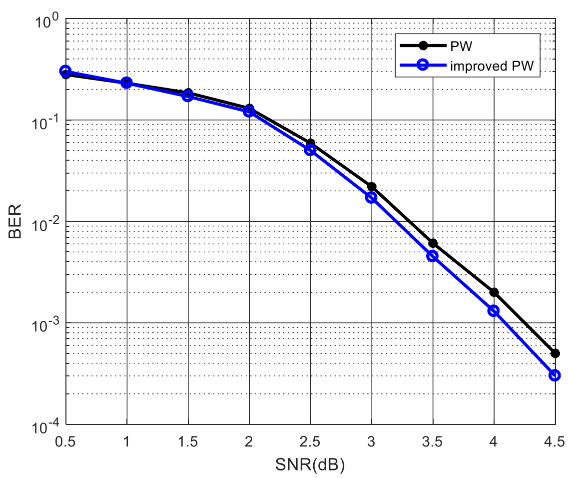

Fig. 2. BER vs. SNR under Gaussian channel using PW algorithm and improved PW algorithm

V. CONCLUSIONS

In this paper, we researched four algorithms for polar code construction under Gaussian channel, detailed steps and formulas for these algorithms are presented. It was demonstrated that polar code has almost the same BER performance by four construction algorithms, among which PW algorithm has the lowest complexity, therefore PW algorithm is chosen to construct polar code. According to the simulation results, we proposed an improved PW algorithm, which lower the bit error compared with the original algorithm.

REFERENCES

[1] E. Arıkan, “Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels,” IEEE Trans. Inf. Theory, vol. 55, no. 7, pp. 3051–3073, Jul. 2009

[2] Final Report of 3GPP TSG RAN WG1 #87 v1.0.0, document R1-1701552, MCC Support, Athens, Greece, Feb. 2017.

[3] E. Arıkan, “A performance comparison of polar codes and reed-muller codes,” IEEE Commulett, vol.12, no.5, pp.447-449, june 2008.

[4] Mori R., Tanaka T., Performance of polar codes with the construction using density evolution, IEEE Communications Letters, 13(7), 2009: 519-521.

[5] Chung S , Thomas J R ichardson , Urbanke R . Analysis of sumproduct decoding of low-density parity-check codes using a Gaussian approximation [J] . IEEE Transactions on Information Theory , 2001 , 47(2) : 657 - 670 .

[6] Yiqun Ge, Ran Zhang, Wen Tong. “β-expansion: A Theoretical Framework for Fast and Recursive Construction of Polar Codes”. IEEE Global Communication Conference,2017

[7] Hussami N, Korada S, Urbanke R. Performance of polar codes for channel and source coding. [C] //IEEE International Sympousion on Information Theory (ISIF), 2009