International Journal of Advanced Science and Technology
Vol. 29, No. 7s, (2020), pp. 4207-4216

Channel Codıng Technıque In 5G Radıo Network

B.Yamini Pushpa1, Dr. Sunita panda2

Abstract

Eminent LDPC codes have appreciated performance with large block length and high code rate, which makes use in current wireless applications. Although LDPC codes show underlying performance for short block lengths and low code rates which are essential for 5G scenarios. There is inherent requisite for channel coding in 5G scenarios to support variable code rates and code lengths with high throughput, low tolerance, low energy and low decoding efficiency. The codes which satisfy all the above requirements are polar codes. This paper presents comparison of turbo codes, LDPC codes and polar codes at the introduction part and usage of polar codes in different scenarios

1 Introduction:

During transmission of data, error free transmission plays major role. For that we are using some coding techniques to transfer the data. These techniques are available at the source level, as well as at the channel level. The most familiar source coding techniques are Shannon'sfano coding, Huffman coding.. etc. Coming to the channel coding techniques number of block coding techniques are available. With regard to encoding the data in 3G and 4G, Turbo codes and LDPC codes had been handed-down. Nothing left with the turbo codes, as these codes results in low decoding throughput for infinite block lengths accompanying Scanty performance for short block lengths along with decoding entanglement Which consists large error transmission bits and there is no further improvement in the performance. This constitutes unsuitable for reliable communication. On account of these reasons turbo codes are not used in 5G. Eminent LDPC codes have appreciated performance with large block length and high code rate, which makes use in current wireless applications. Although LDPC codes show underlying performance for short block lengths and low code rates which are essential for 5G scenarios.

5G uses channel coding as a pivotal mechanism for advanced technologies in various applications. Inherent requisite for 5G is to support variable code rates and code lengths with high throughput, low quiescent, low energy and low decoding entanglement.New Radio Access Technology(NRAT) for 5G utilizes 3 scenarios namely: enhanced mobile broadband (eMBB), Ultra Reliable and Low Latency Communication (URLLC) and Massive Machine Type Communication (mMTC). The coding technique which has the above mentioned requisite called polar codes.

Polar codes are class of linear block codes because they work on block of symbols/ bits and can be used for error correction, developed by Erdal Arikan. Polar codes have less noise effect in the transmission of accurate signals, which can be achieved from the concept of polarization. Hence the name polar codes. These codes are very good in making variable code rate and code lengths yields in high throughput and BER performance better than the remaining codes. These codes have greater channel capacity and reduced complexity in encoding and decoding methods results more attractive and successful in 5G technology.

2 Polar Codes Encodıng

ISSN: 2005-4238 IJAST

Arikan proposed polar codes have the concept of channel polarization and he carried accurate mathematical proof[1]. To pick exact message bits in polar codes, Bhattacharya parameters and density evolution methods are used. Encoding of polar codes carried in two ways: Non Systemic Polar codes (NSPC) and Systemic Polar codes (SPC)[2].

While implementing polar codes in NSPC, there is hardware compliance and computing resources are more as it uses generator matrix. For that parallel structures made simple encoding. Although it uses more computing resources in view of larger length of polar codes. Compared with NSPC, SPC has transcended in the bit error rate performance. Computational units in NSPC can be reduced by using parity check matrix in SPC. This can be done by using exclusive OR (Ex-OR) operation in the encoding structure.

2.1. Non Systemic Polar Codes:

From the basic knowledge of information theory, any code has the length $N= 2^n$ where n is positive integer. Let the length of message bits K and the code rate is given by $R=\frac{K}{N}$ [4].The length K message bits are in the set of matrix A and its complement is A^1. The selected message bits after encoding is A^1-A, these bits are known as frozen bits of length N-K.

Let arbitrary message vector is $m_i= (m_1,m_2,m_3,......m_N)$ and gives code word vector as

$Y_i= (y_1,y_2,y_3......y_N)$.

Generator matrix G= $F^{\otimes n}$ where $n=log\ N$.

$\otimes$is Kronecker operation,

$F=\begin{bmatrix}1 & 0\\ 1 & 1\end{bmatrix}$

For the length N, the generator matrix is given as $G_N = \begin{bmatrix}G_{\frac{N}{2}} & 0\\ G_{\frac{N}{2}} & G_{\frac{N}{2}}\end{bmatrix}$

Now the code word vector for NSPC is $Y=m_iG_N = m_AG_A + m_{A^1}G_{A^1}$

Where $m_A$and $G_A$are sub matrices of $m_i\ and\ G_N$ and G_A is constructed by the rows of indices in A. For simple 2-bit transmission has message bits $m_i = (m_1, m_2)$ and generator matrix of order 2× 2 is given as $G_2 = \begin{bmatrix}1 & 0\\ 1 & 1\end{bmatrix}$.

Now the encoded data is $Y_2 = m_iG_2$

$$= (m_1m_2)\begin{bmatrix}1 & 0\\ 1 & 1\end{bmatrix}.$$

$$=[m_1 + m_2m_2]$$

In the same manner, for 16-bit transmission message bits are $m_i = (m_1, m_2, m_3, m_4, \dots \dots \dots, m_{16})$ and generator matrix of order 16×16 is given as $G_{16} =$

ISSN: 2005-4238 IJAST

$$
\begin{pmatrix}
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
1&1&0&0&1&1&0&0&0&0&0&0&0&0&0&0\\
1&0&1&0&1&0&1&0&0&0&0&0&0&0&0&0\\
1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0\\
1&1&0&0&0&0&0&0&1&1&0&0&0&0&0&0\\
1&0&1&0&0&0&0&0&1&0&1&0&0&0&0&0\\
1&1&1&1&0&0&0&0&1&1&1&1&0&0&0&0\\
1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0\\
1&1&0&0&1&1&0&0&1&1&0&0&1&1&0&0\\
1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1
\end{pmatrix}
$$

Now the encoded data is given as $Y_{16} = (Y_1, Y_2, \dots \dots . Y_{16})$. These bits can be calculated from simple mod-2 operation from the following equations

$$y_1 = (m_1 + m_2 + m_3 + m_4 + m_5 + m_6 + m_7 + m_8 + m_9 + m_{10} + m_{11} + m_{12} + m_{13} + m_{14} + m_{15} + m_{16})$$

$$y_2 = (m_2 + m_4 + m_6 + m_8 + m_{10} + m_{12} + m_{14} + m_{16})$$

$$y_3 = (m_3 + m_4 + m_7 + m_8 + m_{11} + m_{12} + m_{15} + m_{16})$$

$$y_4 = (m_4 + m_8 + m_{12} + m_{16})$$

$$y_5 = (m_5 + m_6 + m_7 + m_8 + m_{13} + m_{14} + m_{15} + m_{16})$$

$$y_6 = (m_6 + m_8 + m_{14} + m_{16})$$

$$y_7 = (m_7 + m_8 + m_{15} + m_{16})$$

$$y_8 = (m_8 + m_{16})$$

$$y_9 = (m_9 + m_{10} + m_{11} + m_{12} + m_{13} + m_{14} + m_{15} + m_{16})$$

$$y_{10} = (m_{10} + m_{12} + m_{14} + m_{16})$$

$$y_{11} = (m_{11} + m_{12} + m_{15} + m_{16})$$

$$y_{12} = (m_{12} + m_{16})$$

$$y_{13} = (m_{13} + m_{14} + m_{15} + m_{16})$$

$$y_{14} = (m_{14} + m_{16})$$

$$y_{15} = (m_{15} + m_{16})$$

$$y_{16} = (m_{16})$$

2.2. Systematic Polar codes:

NSPC are little bit confusing compared with SPC. PC uses simple butterfly structure to encode the data in hardware implementations. The computational complexity in NSPC can be reduced by using XOR operations. SPC structure is butterfly structure.

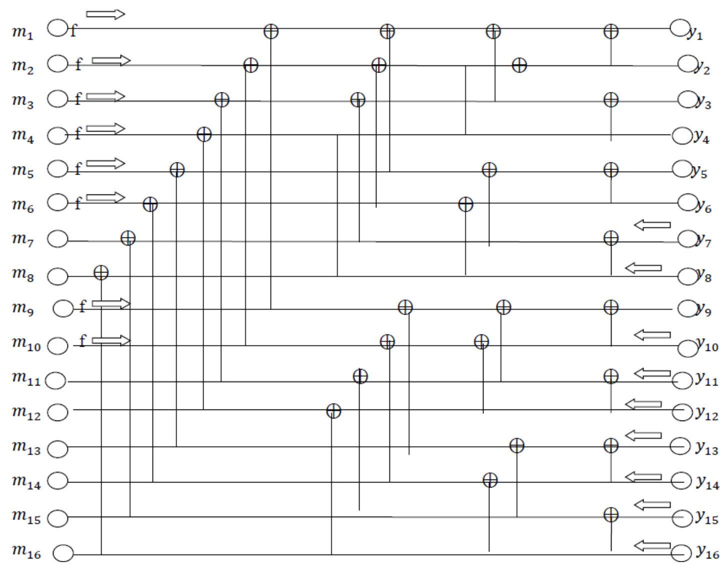

Fig:2.2.1 systemic polar codes structure

The above fig:2 shows systematic polar codes structure. From this figure we can say that number of XOR operations reduced. To encode any bit, by seeing the structure we can write the equation straight away. This is the simplicity of encoding polar codes using SPC.

Number of frozen bits are calculated from (N,K) code word format, where N is number of bits after encoding, and K is number of message bits. Now frozen bits are N-K bits. Exact the position of the bits can be known from 5G standard. Let the number of message bits are 8, and code word length is 16, then frozen bits are 16-8=8. These 8 frozen bits are {1,2,3,5,9,4,6,10}.

No need to encode these frozen bits as these bits are accurate and by replacing these bits to 0,we can start computing for the remaining bits. Encoding direction starts from bottom to top and encoding process for message bits starts from right to left, for frozen bits it is from left to right. The arrows in the above figure represents computing direction of message bits, as well as frozen bits. The value of frozen bits doesn't affect the encoding results; we set those values to zero. After making frozen bits to zeros, the remaining bits can be calculated from the following equations, which are obtained from the butterfly structure.

$$y_7 = (m_7 + m_8 + m_{15} + m_{16})$$

$$y_8 = (m_8 + m_{16})$$

$$y_9 = (m_9 + m_{10} + m_{11} + m_{12} + m_{13} + m_{14} + m_{15} + m_{16})$$

$$y_{11} = (m_{11} + m_{12} + m_{15} + m_{16})$$

$$y_{12} = (m_{12} + m_{16})$$

$$y_{13} = (m_{13} + m_{14} + m_{15} + m_{16})$$

ISSN: 2005-4238 IJAST

$$y_{14} = (m_{14} + m_{16})$$

$$y_{15} = (m_{15} + m_{16})$$

$$y_{16} = (m_{16})$$

3 Decodıng Of Polar Codes

Sequential decoding of polar codes known as successive decoding. Polarization of these code can be implemented at the receiver stage. As we know code word can be generated using Kronecker matrix multiplied with message bits. In the next step this code word split into n bit channels, with code word length n[2]. Response from the each bit channel is different. From the first bit channel it is complete received vector r_n and from the second bit channel, it is received vector and previous message bit. In this manner output from the each channel can be generated. Determination of previous bits takes tedious work. Suppose the bits are decoded precisely, then we can say all the channels polarized very good, else if polarized very badly. Here polarization plays crucial role from very bad to very good based on quality of bit channels.

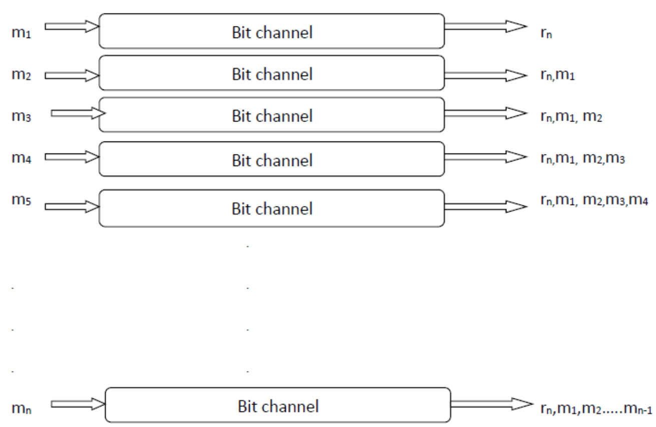

Fig3.1 transmission of message bits through channel

From the above figure, it is clear that from the first bit channel response is r_n, and from the second bit channel it is r_n, m_1. Correspondingly from the third bit channel r_n, m_1,m_2. This process continues to last bit channel. Where m_1,m_2.. are the previous message bits. From the received bits, finding of previous bit is very tedious work. For that expected bit taken as previous bit. Consider the case, if the bit is frozen, then no need to decode, since frozen bit is 0. If the bit is not frozen, decoding of the bit must be done. The decoding process accommodates SISO (Soft Input Soft Output) and repetition algorithms. By using SISO previous bit can be estimated, and to get next bit repetition code will be employed along with estimated bit. This procedure repeats up to acquisition of the last bit. If the selected decoder is good, then decoding vector also error free, if not frozen bits help to decode correctly. This is the processes of sequential decoding of polar codes.

Let's start our algorithm by considering bit length N=2 . Basic building block of SC decoder: N =2 is shown below.

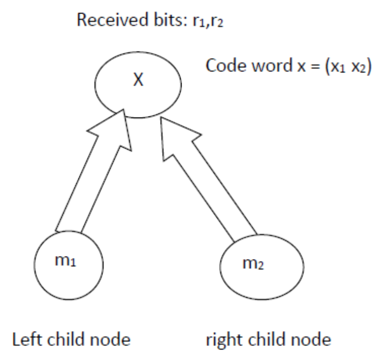

Fig3.2: binary tree representation

For bit 2 length, depth is 1, it consists of root nod and two child nodes as shown in the figure. At the child node available bits are m_1 and m_2.The bits which are ready to transmit are m_1 and m_2. Now these bits are encoded using Kronecker matrix G to obtain code word of length-2 as x_1 and x_2. These encoded bits are transmitted through BPSK and AWGN channel and the resultant bits are r_1,r_2.

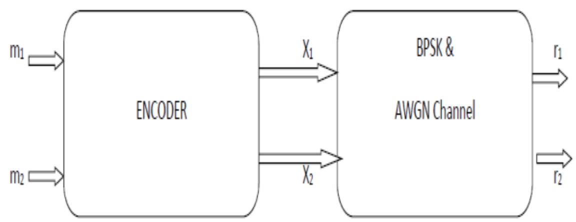

Fig.3.3. transmission of message bits of length-2

$X = (x_1 \ \ x_2)$

$=(m_1+m_2 \ \ \ m_2)$

On transmitting these bits through channel we get r_1 and r_2.In another way, r_1 was received when x_1 transmitted and r_2 was received when x_2 was transmitted (r_1 is belief for x_1 and r_2 is belief for x_2). Which means that r_1 corresponds to x_1 and r_2 corresponds to x_2.Now our intention is to find beliefs for m_1 and m_2.Here decoding is exactly reverse process of encoding. Using x_1 and x_2, by applying single parity check (SPC) code m_1 can be obtained and by using repetition code m_2 can be obtained. That is $m = (m_1 \ \ m_2)$

$=(x1+x2 \ \ x2)$

For SPC minsum can be utilised. i.e;

$L(m_1)= f(r_1, r_2) = \text{sign}(r_1)\ \text{sign}(r_2)\ \min(|r_1|,|r_2|)$

$m_1=0$ if $L(m_1)>=0$ and $m_1=1$ if $L(m_1)<=0$

From the above equations, considering threshold values m_1 can be obtained. i.e; if belief is positive m_1 is 0 and if belief is negative m_1 is 1. Then m_2 can be decoded using repetition code based on previous message bit m_1 .

If $m_1=0$, $L(m_2) = r_1+r_2$ $\quad$ $x=(m_2\ m_2)$

If $m_2=1$, $L(m_2)= r_1-r_2$. $\quad$ $X=(m_2\ m_2)$

From the above discussion, it is clear that decoded bits can be obtained one after another as we are not decoding all at a time. This is the method of successive cancelation. From this we can extend this decoding algorithm into message passing for larger lengths. As we know from the above r_1 and r_2 are beliefs for x_1 and x_2.

Step 1: The received the bits r_1 and r_2, These bits are passed to left child to get soft decision by using minsum algorithm. i.e;

$f(r_1, r_2) = \text{sign}(r_1)\ \text{sign}(r_2)\ \min(|r_1|,|r_2|)$

With this message bit m_1 taken as hard decision, that bit value transfer to the root node.

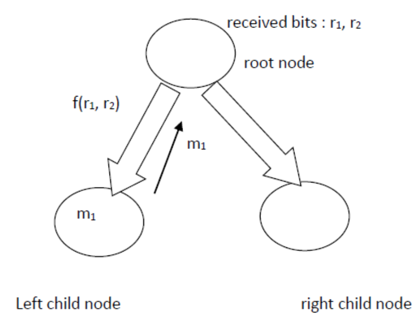

Fig.3.4.hard decision from left child node

Step 2: In this step message m_1 and received bits r_1,r_2 sent to second child, that is to the right child. The hard decision can be calculated from

$g(r_1, r_2, m_1)= r_1+(1-2(m_1))r_2$. M_2 can be decided by hard decision.

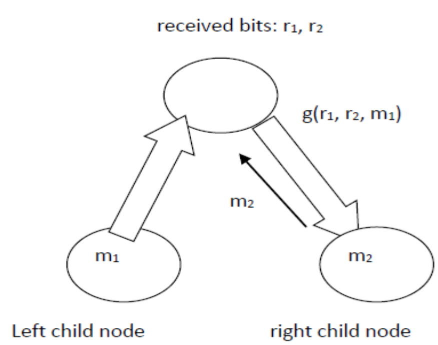

Fig:3.5. hard decision from right child node

Step 3: once hard decisions from both the child's received, then root node will decides the code word with message bits. X= $(x_1\, x_2)$

$$=(m_1+m_2\ m_2)$$

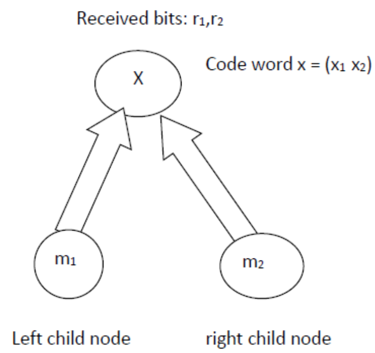

Fig.3.6.hard decision at root node

Let implement the same procedure for N=4. Here depth is 2.

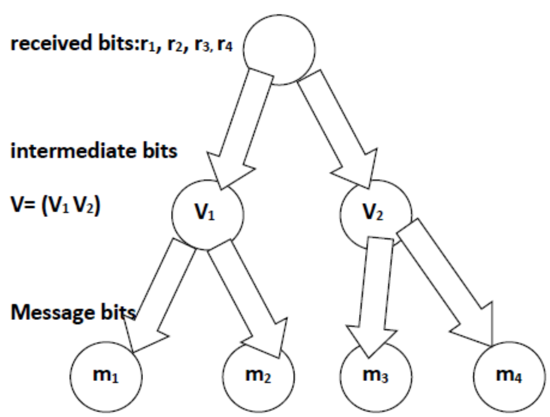

Fig:3.7.binary tree representation for message bits of length 4

In stage 1 bits are m_1, m_2, m_3 and m_4, and at stage 2 intermediate bits are v_1 and v_2.

V1= $(m_1+m_2\ m_2)$

V2=$(m_3+m_4\ m_4)$

from these intermediate bits x can be calculated as $x=(v_1+v_2\ v_2)$

v_1+v_2 is of 2 bit size and v_2 is also 2bit size, finally code word X is 4 bit size. r_1, r_2, r_3, r_4 are the beliefs for x_1, x_2, x_3 and x_4 . Our intention is to find belief for m_1,m_2,m_3, m_4.

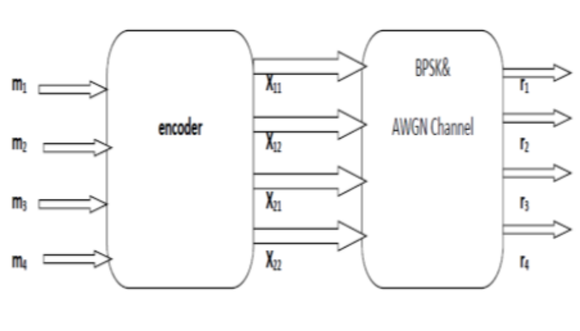

fig:3.8. transmission of message bits over the channel

Initially find belief for m_1 and by using this find belief for m_2, and the processes continues to m_4. This is the method of successive cancelation decoding.

Step1: In the binary tree representation, r1 and r_3 are beliefs for x1, r_2 and r_4 are beliefs for x2. Next is to get beliefs for x2 and x2. For that, use minsum algorithm.

$L = (f(r_1, r_3)\ f(r_2,r_4))$

$X = (x_1\ \ x_2) = (x_{11}+x_{21}\ \ x_{12}+x_{22})$

Now v= x1+x2 , it becomes N=2 processe X_{21}.

step2: In the same manner for right child to get hard decision using $g(x_1\ x_2)$

step3: bits will be sent to the right child.

$L= (g(r_1,r_2, m_1+m_2)\ \ g(r_2,r_4,u_2))$

$L=(L_1\ L_2)$

These are the beliefs going to the lower child of left to get m_3 by taking hard decision $f(L_1,L_2)$, by using m_3. Using $g(L_1,L_2,m_3)$, m_4 can be computed. Once we calculate m_1, m_2, m_3 and m_4 then no need to go further.

Step 4: if the length of the code word is larger than we need to go estimate of code word.

4 Applications

1. Polar codes in speech communication can achieve greater capacity with low encoding and decoding entanglement compared with LDPC codes in both AWGN and Rayleigh channels and also the system is successful in low code rate and high code length.

2. In recent years for free space communication, satellite laser technologies achieved larger transmission capacity from the ground to satellite transmission, where as it is not for satellite to ground. By using polar codes considerable channel capacity can be achieved.

3. Because of variable block sizes, polar codes are more attractive in industrial applications. Huawei Company declared that, by using polar codes in channel coding in 5G achieved 27Gbps in field trial.

5 Conclusıon

Polar codes have self-deprecating encoding and decoding complexity, for that they are used in many applications including 5G wireless transmitter and receiver. As implementation of polar codes very simple and straight forward. They are also acceptable channel coding in controlling the channel. Finally polar codes are examine to solve previous problems such as complexity, network infrastructure, latency and quality of service on comparing turbo codes and LDPC codes.

References

1. E.Arikan "Channel polarization: a method for constructing capacity achieving codes for symmetric binary-input memory less channels " IEEE Trans. on Inform. Theory, vol. 55, no. 7, pp. 3051 –3073, Jul. 2009.
2. I. Tal and A. Vardy, "How to construct polar codes," submitted to IEEETrans. Inform. Theory, available online as arXiv:1105.6164v2, 2011.
3. E. Sasoglu, E. Telatar, and E. Arikan, "Polarization for arbitrary discrete memoryless channels," in Proc. IEEE Information Theory Workshop ITW 2009, 2009, pp. 144–148.
4. H. Mahdavifar and A. Vardy, "Achieving the secrecy capacity of wiretap channels using polar codes," in IEEE ISIT 2010, Jun. 2010, pp. 913 –917
5. Ido Tal and Alexander Vardy, "List decoding of polar codes," in Proc.(ISIT) Symp. IEEE Int Information Theory, 2011.
6. C. Zhang, J. Yang, X. You and S. Xu, Pipelined implementations of polar encoder and feed-back part for SC polar decoder, in IEEE International Symposium on Circuits and Systems (Lisbon), pp. 3032–3035 (2015).
7. Xiumin Wang, Zhihong Zhang, Liang Shan "anoptimised encoding algorithm for systematic polar codes" in EURASIP journal on wireless communications and networking, 2019.