Performance investigation of polar codes employing polar encoder and successive cancellation decoder in 5G systems and cloud computing

[1]Krishna Prasad S J, [2]Dr.B K Sujatha, [3]ThirthaPrashanth K M

[1]Associate Professor Dept. of E&,TCE, MSRIT, Bangalore

[2] Professor Dept. of E&,TCE, MSRIT, Bangalore

[3] Student

Abstract

Information encompasses broad spectrum of data in terms voice, images, sensor and software generated data. Digital computer communication techniques are popular for transmission of information in present days in global scenarios. This is due to the fact that the quality of perception user is bestowed at the receiving end. Channel coding and decoding are important units of these digital computer communication systems. Coding in general and channel coding in particular will ensure secure and reliable communication with the onset of ongoing cyber threats in digital world. Future wireless systems like 5G systems are expected to enhance speed of communication as compared to predeccessor technologies. It demands high performance codes having less complexity of hardware associated with encoder and decoders. These systems operate in exceptionally reliable scenarios, better efficiency for short and long messages, high throughput, low Bit error rate and high code rates. Polar Codes are having capacities of achieving forward error correction which are realized with low complex encoder and decoders. In this work main focus is on development of successive cancellation decoder using $G2$ to $G512$ transforms and reliability sequences of 5G systems used in cloud computing paradigm. In order to minimize the Bit error rate at reduced SNR one has to deploy polar codes. Key area of research in polar code is development of decoder part. Successive cancellation decoder is most popularly used because of its reduced complexity and better efficiency for various code lengths. Bit Error Rate(BER) for turbo code is 10^{-2} at $3dB$ whereas, polar code documents BER of 10^{-5}at $3dB$ in present work.

Keywords: Polar codes; Successive cancellation decoder; ITU; 5G communication; Cloud computing

1. Introduction

Mobile phones nowadays are essential devices for global communities in day to day life. Earlier to mobile phones were circuit switched landline systems but due to inconveniences caused with those systems 1G or first generation mobiles were introduced in the 1980s. It had speed of 2.4 kbps and provided mobile telephony services. 2G or second-generation provided digital voice and short messaging. It was introduced in 1990 with a speed up to 64 kbps. The 3G or third generation was an innovation in the mobile technologies.3G commercial mobile networks were operational based on W-CDMA technologies in Japan around 2001.As a next leap 4G systems were evolved. It evolved because of market needs.3G main function was voice communications on circuit switch systems. People preferred wired communication over wireless because wired voice communication was cheaper. Another driving force was smart phone developments which included keyboard, display, camera, navigation, microphone and sensor varieties. Nowthat we are in the era of 5G systems.5G systems not only focus on system capacity but also on energy specifications, efficiency, connection density and latency.5G has been reality in year 2020 and it achieves practically a speed of 1Gbps.New specific requirements and application cases energized New Radio Access Technology (NRA-T) for 5G. International Telecommunication Union Radio communication (ITU-R) defined three major scenarios of 5G use in September 2015: Enhanced Mobile Broad Band (eMBB), Ultra-Reliable-Low Latency Communications System (URLLC), and Massive Machine Type Communications (mMTC). Various code lengths and speeds with low vitality, low latency, high reliability, and high throughput are possible requirements for 5G channel coding(Kim *et al.*, 2020).

Coding is a method of adding a few data bits to original message before transmitting it in order to recover the original data bits at the receiver's end. Forward error correction (FEC) is nomenclature associated with it. For 4G and 5G wireless communication, there are several error correction coding methods, including Repetition codes, Hamming codes, BCH codes, Reed Solomon code, and Turbo codes. LDPC and Polar codes are two types of standardized error control coding techniques. The field of coding hypothesis rose with the spearheading work of Claude E. Shannon distributed among communities around 1948 with an article published in Bell System Technical Journal(Shannon *et al.*, 1949). In his work, Shannon examines both the transmission and pressure of data. A probabilistic model is proposed for the correspondence channel or for the source, and a numerical measure is proposed to evaluate the measure of commotion in the channel (limit) or the measure of the excess source entropy. Shannon at that point gives two great outcomes. The largest amount of data that can be reliably sent is quantified by channel limit. Not long after Shannon, Richard Hamming, in 1950 created coding plans which can address a fixed number of blunders with likelihood one, as contradicted to Shannon who satisfied himself to address mistakes with high likelihood on the commotion acknowledge. Shannon created a result that has ended up in one of the basic hypothesis of coding. This hypothesis essentially states that mistake free transmission of codes is conceivable inside a most extreme rate depending on the commotion of the communication channel. To begin with Shannon built up the capacity of a given channel, which is the hypothetical exchange rate over that channel and is given by equation1

$$C = B * \log(1 + [S/N]) \quad (1)$$

Where C is Channel Capacity, B is the Bandwidth of the channel, and [S/N] is the ratio of signal to noise. Rate of transmission $R_t$in the channel is given by following constraint as given in below equation 2

$$R_t \leq C \quad (2)$$

Historically many error correcting codes are being designed. First one was Hamming codes which are block of linear error-correction codes. Second one was BCH codes which were invented by French mathematician Alexis Hocquenghem in 1959, and delivered to the general population in the year 1960 by Raj Bose and D. K. Pillar Chaudhari termed as BCH codes, otherwise called Bose–Chaudhuri–Hocquenghem codes. Next code in that direction was Reed - Solomon Codes which is a collection of code review errors introduced by Irving S. Reed and Gustavo Solomon in 1960. The turbo codes, which are termed as advanced errors correction codes (FEC), had been developed in 1990-91, but were released in 1993. They are the first codes to return near a large channel capacity. This coding scheme is termed as a landmark milestone developed by Berrou, Glavieux, Thitimajshima by which the gap between capacity limit and practically feasible channel utilization is almost closed. The turbo code has excessive complexity results for low frequency repetition and has excessive latency(Vucetic *et al.*, 2012). For sufficiently large interleaving sizes the correcting performance of turbocodes investigated by simulation done in research works appears to close theoretical limit as proposed by Shannon(Berrou *et al.*, 1996). A low-density parity-check (LDPC) correction code is a way to transmit a message through a noisy channel. LDPC codes are capacity acquisition codes. Which means present constructs that permit the limit to be set very near to the imaginary size (Shannon limit) of the symmetric memory less channel. The execution technique of LDPC codes has been left in some codes, especially in turbo codes(Shokrollahi *et al.*, 2004).

1.1 Polar codes

The area of coding is undergoing extensive study. Polar code is outcome of an investigation into forward error correction code(FEC). The construction of the code is based on repetition of a short code which has ability to convert the physical channel into external channels of varied capacities. When the repetition rate is high, visual channels lean to have either high or low fidelity and pieces of data are assigned to more closely related channels having uniform fidelity. It is first code with a clear structure that achieves the capacity of the channel for equivalent installation of Binary discrete memoryless channel(BDMC) with a polynomial belief on the capacity.

Other attractive features of Polar code is that it has very low latency of 1ms, good error performance, reliability of 99.999%, has a target to achieve throughput of 20Gbps and has a high gain with less error and simple decoding methods which are suitable for 5G systems(Vaz *et al.*, 2019).

The polar codes are a class of error rectification codes that depends on the phenomenon of channel polarization. For Symmetric Binary discrete memory-less channels (BDMC), polar code is the first identified to provably reach Shannon ability(Arikan, Erdal , 2009).

Coming up next are the three crucial structure components of a polar code. Channel Polarization is utilized to assemble the code. The channel coding issue is improved by changing a common channel W into two outrageous channels as follows:

- Complete / Good - A channel that sends data exclusive of error (limit C (W) = 1), is called a trusted channel or reliable channel.
- Useless / Bad - A noise channel that transmits improper data (C (W) = 0), is called an unreliable channel.

The AWGN noise model is often used in communication systems. Arikan used the term ”polar codes” to describe a novel idea based on the concept of channel polarization. One can divide the whole channel into two categories: noisy channel and noise less channel. Information is sent via highest-capacity channel, minimizing the impact of noise. The main difficult effort is to locate channels with the highest capacity. Polar codes are very efficient due to their low error rate(Arikan, Erdal , 2009).In polar codes the number of channels are created based on N, which is given by $N = 2^n$. N number of channels is created based on the concept of channel splitting using Arikan transform with the help of Kronecker product method. Channels are classified into two extreme ends, either too high noise or too low noise channels and there is no concept of moderate noise in the channel hence the name polarization.

In a Binary symmetric discrete memoryless channel(B-DMC) there are two parameters of interest the symmetric capacity I(w) and Bhattacharya Parameter Z(w). These parameters are employed as measures of rate and reliability, respectively. Using the inputs of W with equal frequency I(W) is the highest rate at which communication reliably is possible across W. When W is used only once to transmit a 0 or 1, Z(W) is an upper bound on the probability of maximum-likelihood (ML) decision error. The symmetric capacity I(W) equals the Shannon capacity when W is a symmetric channel. The Binary Erasure Channel (BEC) and the Binary Symmetric Channel (BSC) are examples of symmetric channels. Channel polarization is an operation by which one manufactures out of N independent copies of a given B-DMC W a second set of N channels W(i)N : 1≤ i ≤ N that show a polarization effect in the sense that, as N becomes large, the symmetric capacity terms I(W(i)N) tend towards 0 or 1 for all but a vanishing fraction of indices i. This operation consists of a channel combining phase and a channel splitting phase. Researchers have taken advantage of the polarization effect to construct codes that achieve the symmetric channel capacity I(W) by a method called polar coding. The basic idea of polar coding is to create a coding system where one can access each coordinate channel W(i) N individually and send data only through those for which Z(W(i)N) is near 0(Arikan, Erdal , 2009). A Brief comparison of the LDPC codes, Turbo codes and Polar codes are given in the following Table1 .In present work ,the consideration is the Binary discrete memoryless channel. Because this channel has no capacity to store value due to a lack of registers, it is known as Binary Discrete Memory Less Channel(BMDC). In polar codes, there are a variety of number decoders, including successive cancellation decoders, successive cancellation list decoders,

Table 1. Comparison between Polar Codes, LDPC and Turbo Codes based on code nature

Code parameters	Polar codes	LDPC codes	Turbo codes
Latency	The polar code has an overall latency of less than 1ms.	The LDPC code has a latency of more than 1ms	The turbo code has a latency of more than 1ms.
Information throughput	High in terms of pipeline structure.	High for in terms of parallel structure.	Low in terms of parallel structure
Computational complexity	Less due to its fast function	moderate	more
Flexibility	Very high flexibility	Partial flexibility	High flexibility
Bit Error Rate (BER)	Low BER (10-5)ˆ at 3dB Eb/No	Moderate BER	Moderate BER (10-2)ˆ at 3dB Eb/No
Error correction performance	Superior error correction performance of 99.999%	The performance of error correction is average.	Performance in error correction is average.

adaptive decoders, and maximum likelihood decoders. The successive cancellation decoder is the focus of the present study.(Hu *et al.*, 2017).

2. Related Works

A.C.Vaz et.al. (Vaz *et al.*, 2019)have compared Turbo codes and Polar codes. They have compared Turbo Code and Polar Code with respect to channel capacity by considering short and long length messages. This work centers on the assessment and investigation of spotting bit errors rates for various block lengths and coding rates. M.Hu et.al. in their work(Hu *et al.*, 2017)have compared different Polar decoding algorithm and they have found Successive Cancellation Decoder gives best results. And they also have compared both systematic and non-systematic polar code. They have also said that Polar code is the first scientifically proven technology to reach effective channel coding and capabilities.

E.Arikan in his article (Arikan, Erdal , 2009) discussed about channel polarization to construct code sequence. Performance is attainable by encoders and decoders with complexity O (N log N) for each. Author has done theoretical findings to prove versatility of channel polarization.

Balatsoukas-Stimming et .al. in their article (Balatsoukas-Stimming, 2017)discussed about comparison of different error control coding schemes like Polar Code, LDPC and Turbo Code. This comparison allows polar encoders to identify situations that outperform existing error-correcting encoding methods, as well as a better search way to implement polar decoders. B.Tahir et.al. in their works (Tahir et al, 2017)they have taken up a first execution of Turbo, low-density parity check (LDPC) and polar codes that cover a wide scope of dependability and throughput conditions as far as Bit error rate (BER) for various informational index lengths and code. A.Cyriac et.al. in their work (Cyriac*et al.*, 2018) have discussed that over the years, various methods have been proposed for creating, encoding and decoding polar codes. The decoder section is an important area of study, and the Successive cancellation decoder is the most widely used.

N.F.Muhammed et.al in their works (Mohammed *et al.*, 2020) have discussed the authenticity of source is an important requirement in e health care systems. It is required to maintain the privacy of patient's data in medical records. The Polar code can be considered as coding requirement especially in case of cloud computing. This cloud computing makes use of 5G wireless systems. Hence polar codes can also be used as security mechanisms to maintain privacy in the wireless network transactions.

After analysing all of these above said works research gap can be stated as follows:

In evolving and realized systems like 6G and 5G respective wireless networks, Latency of the transmission is having highest priority. So also the security and error free transmissions. In that sense the Polar code has got vast utility in such systems. Also looking into decoding techniques in Polar code, M.Hu et.al. in their works(Hu *et al.*, 2017)have compared different Polar decoding algorithms and they have found Successive Cancellation Decoder gives best results. Out of simulation environments MATLAB is a natural choice because of vast communication tool boxes available for engineers. Accordingly consolidating all these factors described, this work has been organized into different sections. Section2 describes related works priory done. Section3 describes methods and materials of the work undertaken, Section 4 describes implementation part of the work, Section 5 documents the results and discussion/ analysis of results obtained are highlighted in Section6. Finally, conclusions of the work done are documented. The problem taken up is to design, develop and simulate Polar Code in MATLAB environment and analyse results.

3. Methods and materials

A typical communication system diagram is as shown in Figure1. System includes a data source where information in terms of data bits is originating. Analog to digital transformation is done before source. Data from source is encrypted (using a polar transformer encoder in the present work) into binary data streams. These encoded streams are modulated using digital modulation schemes like BPSK, QPSK, and M-ary PSK, aiding in data transmission over long distances. Modulated signal is then sent through an AWGN/ BSC channel, directing signals from the transmitter to receiver. Signal is then pulled down at the receiver's output using a BPSK demodulator employed. Message is transmitted to destination after decryption using a SC decoder. In this work modulation scheme adopted is BPSK (Binary Phase Shift

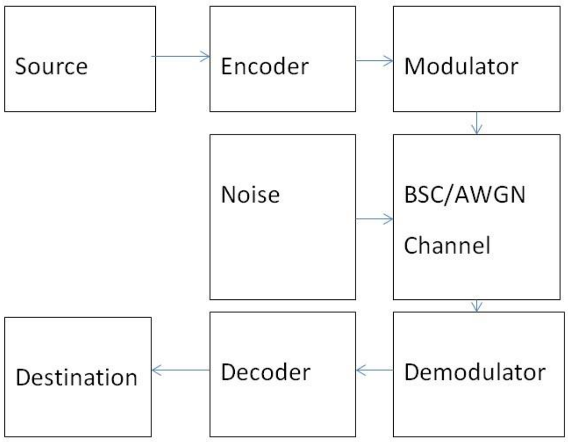

Fig. 1. Block Diagram of a General Digital Communication System

Keying), widely utilized in digitally. BPSK modulation is advanced and powerful of all PSK modulation techniques. Carrier signal must be removed on the receiver side in order to decipher real signal. Carrier recovery block is termed as demodulator. BPSK signal is recovered with carrier and transmitted via integrator, with decision block providing final signal. In BPSK, 1 is denoted by letter '1' whereas 0 is denoted by letter '-1' during the time of decoding, there will be two types of decision making they are soft decision and hard decision. In the hard decision, anything less than '0' is decoded as 0 and anything greater than '0' is decoded as 1. We would be particularly simulating hard-decision decoders.

3.1 Polar Transform encoder

Polar Transform Encoder is used to encode information before transmitting through channel. It is based on Arikan Transform as discussed in many literatures(Mahdavifar*et al.*, 2013)(Cyriac*et al.*, 2018).Polar code construction is based on the following observations by Arikan which is termed as channel polarization. Let G be denoted by equation 3

$$G = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad (3)$$

The i-th Kronecker power of G, which is denoted by $G^{\otimes i}$, is defined by induction i.e. $G^{\otimes i}$,= G and for any i ≥ 1: This transformation is given by equation4

$$G^{\otimes i} = \begin{bmatrix} G^{\otimes(i-1)} & 0 \\ G^{\otimes(i-1)} & G^{\otimes(i-1)} \end{bmatrix} \quad (4)$$

Next, for all $N = 2^n$ let us define the Arikan transform matrix $G_n \overset{\text{def}}{=} R_N G \otimes i$ where RN is the bit reversal permutation matrix.

Arikan defines Bhattacharya parameter Z(W) and also defines theorem for successive cancellation decoding .It is easy to show that Bhattacharyya parameter Z(W) is always between 0 and 1. Intuitively, Z(W)shows how good the channel W is been. Channels with Z(W) close to zero are almost noiseless, while channels with Z(W) close to one are almost pure-noise channels.

3.1.1 G2 Transformation

In G2 transform when MATLAB code is developed consideration is that matrix U1 defined by equation5 and U2 defined by 6 and using EXOR operation between U1 and U2. The output of G2 transform is U1 EXOR U2 given by equation7 and technically method of obtaining G2 retaining U2 as given in equation8. Binary representation of G2 transform is shown in Figure2

$$U_1 = \begin{bmatrix} 1 & 0 \end{bmatrix} \quad (5)$$

(6)

$$U_2 = \begin{bmatrix} 1 & 1 \end{bmatrix} \quad (7)$$

$$G_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad (8)$$

$$\begin{bmatrix} U_1 & U_2 \end{bmatrix} G_2 = U_1 + U_2 U_2$$

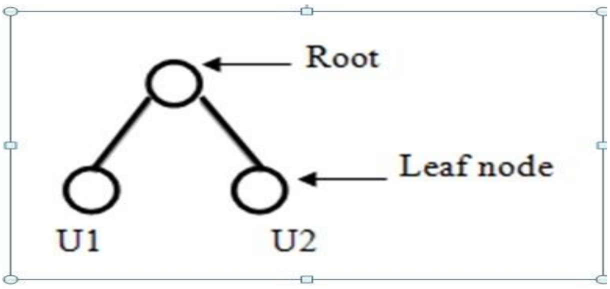

Fig. 2. G2 transform Graphical representation

3.1.2 G4 Transformation

In G4 transform we are multiplying G2 matrix with G2 matrix. First G2 matrix consists of U1 given by equation9 and U2 given by equation10 and second G2 matrix consist of U3 given by equation 11 and U4 given by equation 12. We get G4 transform using EXOR operation between U1 and U2 and also between U3and U4. These transformations are given by equation13 and equation14 respectively. Then technique of getting the output of G4 transform as shown in equation 15. Overall transformation of G4 is given by equation16.Binary representation of G4 transform is shown in Figure3

$$U_1 = \begin{bmatrix}1 & 0\end{bmatrix} \quad (9)$$

$$U_2 = \begin{bmatrix}1 & 1\end{bmatrix} \quad (10)$$

$$U_3 = \begin{bmatrix}1 & 0\end{bmatrix} \quad (11)$$

U4=[1,1] (12)

[U1 U2]G2= U1+ U2U2 (13)

$$\begin{bmatrix}U_3 & U_4\end{bmatrix} G_2 = U_3 + U_4U_4 \quad (14)$$

(15)

$$G_4 = \begin{bmatrix}1 & 0\\1 & 1\end{bmatrix} \bigotimes \begin{bmatrix}1 & 0\\1 & 1\end{bmatrix} \quad (16)$$

$$G_4 = \begin{bmatrix}1 & 0 & 0 & 0\\1 & 1 & 0 & 0\\1 & 0 & 1 & 0\\1 & 1 & 1 & 1\end{bmatrix}$$

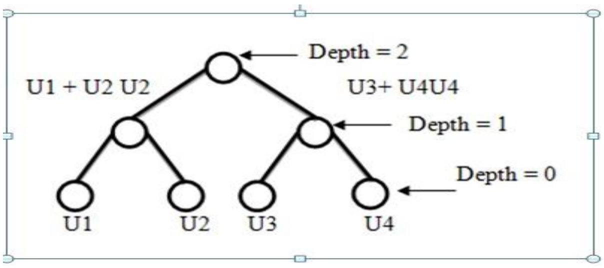

Fig. 3. G4 transform binary tree representation

3.1.3 G8 Transformation

For G8 transform we have to multiply G2 transform and G4 transform as shown in equation 17. Output matrix G8 is given in the equation 18. Tree transform of G8 is shown in Figure4. Here there are 8 channels (from X1 to X8) are created as shown in Figure 4.

When significant numbers of repeated recursions are used, the virtual channels will have either high or low reliability, and the data will be organized according to channel reliabilities.

$$G_4 = \begin{bmatrix}1 & 0 & 0 & 0\\1 & 1 & 0 & 0\\1 & 0 & 1 & 0\\1 & 1 & 1 & 1\end{bmatrix} \bigotimes G_2 = \begin{bmatrix}1 & 0\\1 & 1\end{bmatrix} \quad [17]$$

$$G_8 = \begin{bmatrix}1 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\1 & 1 & 0 & 0 & 0 & 0 & 0 & 0\\1 & 0 & 1 & 0 & 0 & 0 & 0 & 0\\1 & 1 & 1 & 1 & 0 & 0 & 0 & 0\\1 & 0 & 0 & 0 & 1 & 0 & 0 & 0\\1 & 1 & 0 & 0 & 1 & 1 & 0 & 0\\1 & 0 & 1 & 0 & 1 & 0 & 1 & 0\\1 & 1 & 1 & 1 & 1 & 1 & 1 & 1\end{bmatrix} \quad [18]$$

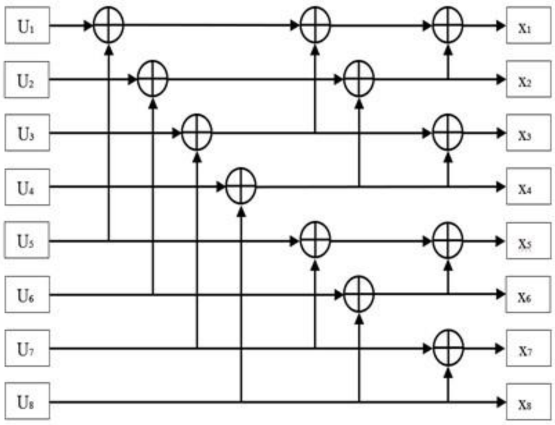

Fig. 4. G8 transform binary tree representation

3.1.4 Reliability Sequence

From channel polarization we know that there are N number of channels created with each having different probabilities. Few channels will be having more noise in the channel as compared to other few channels.

According to the 5G standard reliability series for different N=8,16 and 32 channel values are given by following equations 19 20 and 21respectively.

[1 2 3 5 4 6 7 8] Reliability sequence for N=8 channels (19)

[1 2 3 5]
[9 4 6 10]
[7 11 13 18]
[12 14 15 16]
Reliability sequence for N=16 channels (20)

$$\begin{bmatrix} 1 & 2 & 3 & 5 & 9 & 17 \\ 4 & 6 & 10 & 7 & 18 & 11 \\ 19 & 13 & 21 & 25 & 8 & 12 \\ 20 & 14 & 15 & 22 & 27 & 26 \\ 23 & 29 & 16 & 24 & 28 & 30 \\ 31 & 32 & & & & \end{bmatrix} \text{Reliability sequence for } N = 32 \text{ channels} \quad (21)$$

3.2 Successive Cancellation Decoder

Successive Cancellation (SC) Decoder have been discussed in many works(Yuan *et al.*, 2015) makes one-by-one choices on bits. As a result, only one choice may be taken at any one moment. It takes LLR of channel output y and partial modulo sums of previously calculated bits to decode the ith bit. The guidelines employed for decoding are as Illustrated below

- The estimate ui will be forecasted as 0 if the ith bit belongs to the frozen bit set.
- If the ith bit is no longer part of the frozen bit set, the LLR value from the previous step is compared to a threshold value of 0 to estimate the final results.

Successive cancellation decoder is sequential in nature that is it decodes bit by bit one after the other sequentially. During decoding it uses minimum sum method and normal addition and subtraction operation for decision making. These are described in following subsections

3.2.1 Minimum sum calculation

In min sum calculation we have to take sign (+ or -) according to the received bit at the output and we have to consider the minimum value of two bits which is considered at the output as shown in Figure 9. Minimum (Least) Value of U1 L(U1) is calculated from equation22

$$L(U1) = sign(r1) * sign(r2) * min(|r1|,|r2|) \tag{22}$$

Where, r1 and r2 are received bits with noise variable.

3.2.2 Decision making

The decision is taken based on the output of the min sum calculation if the value is less than or equal to 0 then we have to consider r2+r1, if the value is greater than to 0 then we have to consider r2 - r1 to find the value of U2. Decision making is done by following equations23 and 24

$$IF \quad \hat{U}_1 \leq 0 \text{ then } L(U2) = r2 + r1 \tag{23}$$

Else if

$$IF \quad \hat{U}_1 > 0 \quad then \quad L(U2) = r2 - r1 \tag{24}$$

In general, considering $\hat{U}_1 = b$ we get the next generated bit by equation 25

$$g(r1,r2,b) = r2 + (1–2b)r1 \tag{25}$$

Eventually equations23 and 24 are the mathematical techniques to decode in Successive cancellation (SC) decoder. Let us consider Binary tree of decoding for the SC decoder as shown in the figure5.This decoding tree is experimentally verified decoding tree in implementation steps given in the present work As shown in Figure 5 consider the received bit [r1to r8] = [-0.8207 -1.7354 0.7562 1.1927 3.0123 2.5574 -1.7591 -0.7067] Consider the sign and minimum value of r1 (-0.8207 by comparing 1st and 5th received bit), r2 (-1.7354 by comparing 2nd and 6th received bit), r3 (0.7562 by comparing 3rd and 7th received bit), r4 (-0.7067 by comparing 4th and 8th received bit), to find U1 as shown in Figure 6 and 5.Decoding will be done in following steps:

- STEP1: Now consider minimum value and sign of received value of r1 (0.7562) and r2 (-0.7067) we will get minimum value of -0.7067 as shown in Figure 5

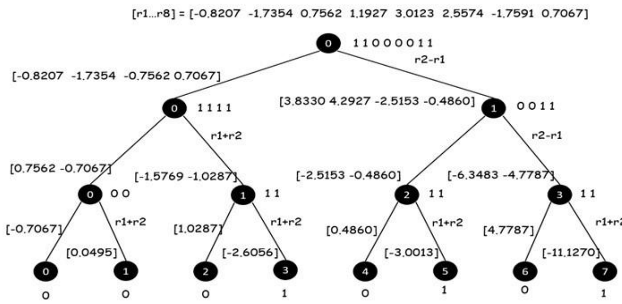

Fig. 5. Error corrected output of SC decoder an example

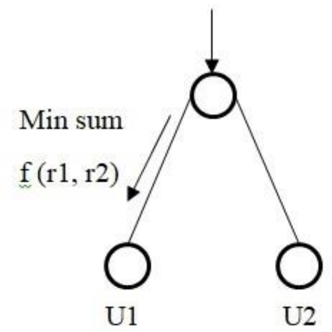

Fig. 6. Binary tree representation for min sum calculation

- STEP2: From Figure5 received value is -0.7067 and is less than 0, then the value of U1 is 1 but channel U1 is frozen for that the value is 0. To calculate U2 consider U1, if value of U1 is 0 then go for r1+r2, if value of U1 is 1 then go for r1-r2. Here value is 0 hence r1+r2 and value is 0.0495 since channel U2 is also frozen channel value of U2 has 0.
- STEP3: Implementation of Step 3 is mathematically shown in the equation26 and tree diagram is depicted in the Figure7

$$x = [\hat{U}_1] + [\hat{U}_2] * [\hat{U}_2] \quad (26)$$

After finding the value of $\hat{U}_1$and $\hat{U}_2$we have to go for EXOR operation between U1 and U2 to find the

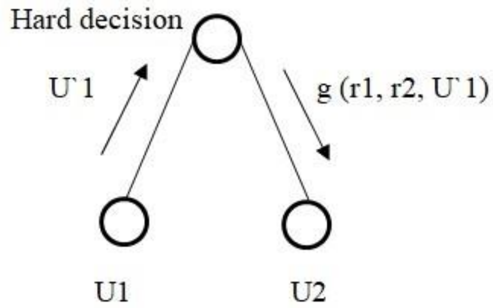

Fig. 7. Binary tree representation for decision

Value of higher node. Since the value U1 and U2 is 0 we will get the higher node value has [0 0] as shown in Figure8 Like this at each node it follows these three steps at each node next higher level bit is

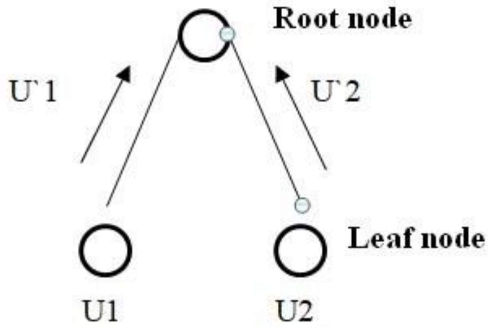

Fig. 8. Binary tree representation to find the value of higher node

Calculated by this technique

4. Implementation steps in MATLAB

In the present work MATLAB software running on Windows 10 platform is the simulation platform used for developing practically implemented and tested software code. Input parameters are n, N, K and Reliability sequence for Polar encoder algorithm/program .In the developed program, input power in decibels (dB) and BER are also inputs given .Generation of message bits using exclusive or operation and construction of Polar code with Arikan transform are steps in program. Conversion into BPSK signal and transmission through channel are next steps realized. In channel synthesized AWGN noise is added. For Polar decoder algorithm, Successive cancellation algorithm has been realized in MATAB. As in encoding we need to Input to the Polar decoder algorithm/program parameters n, N, K and Reliability sequence. Input power in decibels (dB) and BER are also inputs given to this program. Logic of program is to create for loops for depth of the code tree needed and to test node is frozen or not. If node is not frozen then decision is made on Hard decision technique.

Now same operations are continued in left and right of tree and until required steps of message bits. After executing every node algorithms stops by stopping rule defined and calculated parameters are buffered and graphically depicted

5. Simulation Results

Terminologies associated with the simulation results obtained are as follows: For practical purposes design of polar codes is done for N = 8 channels initially. The MATLAB output obtained in command window at the transmission end. Conventions are Q1 is reliability sequence for N = 8 channels is frozen positions msg is message bits, u is encoded sequence, cword is encoded message. After passing the message through AWGN channel demodulation and decoding are done in the receiver end.

In decoding steps rcv is received message msg cap is decoded bits. Bit error rate hypothetically assumes zero at value of Eb/No= 3dB.Though it is zero in simulation In reality results of BER can be lower than that documented standard values of wireless communication systems of 4G and 5G. Simulations are further carried out for values of input parameters for both Turbo and Polar codes as indicated in the Table2 and results of simulation have been illustrated. Results of the simulations have been obtained graphically as shown in the Figure 9 and Figure 10 for Polar code and Turbo codes respectively.

6. Analysis of Results

Decoder performance is measured by plotting Eb/No v/s BER graphs. Figure9 shows decoding performance of successive cancellation decoder for 5G standard N = 1024, K = 512, for 2000 / 5000 iterations. Figure 10shows the graph of BER v/s Eb/No for the 850 message bits with data rate of 0.83 for 34/50/85 iterations for Turbo codes. From simulation results, it is observed that the bit error rate (BER) performance of the polar code is $10^{-5}$and BER for turbo code is 10^{-2}at 3dB Eb/No. One can analyse that

Table 2. Input Simulation parameters for Polar and Turbo codes

Sl No	Parameters of Simulation	Polar Codes	Turbo codes
1.	Number of iterations	2000/5000	34/50/85
2.	Coding rate R	0.5	0.83
3.	Modulation	BPSK	BPSK
4.	Channel	AWGN	AWGN
5.	Message Bits K	512	850
6.	Number of Channels	1024	1024
7.	Data Rate R (Mbps)	100	100
8.	Power Input (dB)	3	3

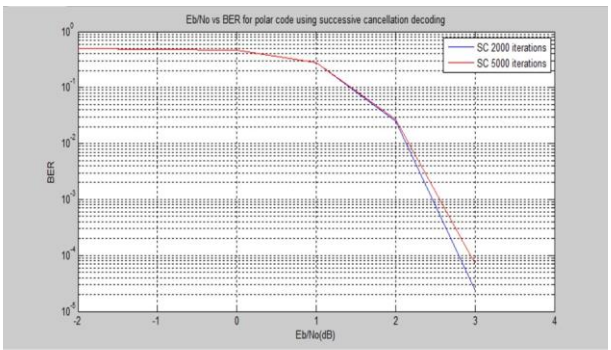

Fig. 9. Successive Cancellation Decoder comparison for 2000 and 5000 iterations for N = 1024, K =512, R = 0.5

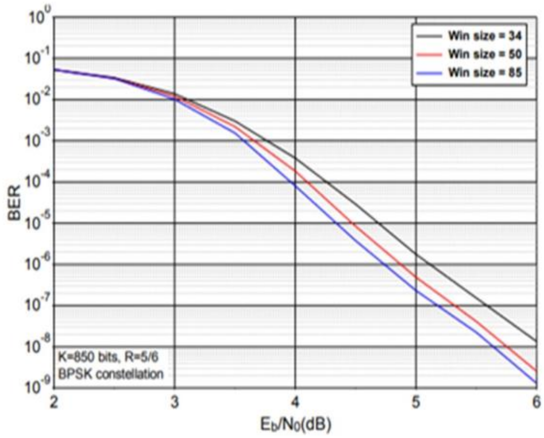

Fig. 10. Turbo Code results for K = 850, R = 0.83 and 34/50/85 iterations

the BER value for polar code 10^{-5} value drops to minimum at 3db (Eb/No), for turbo codes the BER value 10^{-9} drops to minimum at 6db (Eb/No). Thus polar codes performance is better as it achieves desirable BER at less signal power compared to turbo codes.

6.1 Digital Image transmission-A Polar code application

To test an application, case study of digital image transmission through a BSC channel and reconstruction at receiver has been realized in MATLAB. In this case study performance parameter is not mathematical but on visualization aspects.

Technique here is to convert given digital image to binary image and transmit through AWGN channel, reconstruct buffered image at the receiver and examine the quality of the image at receiver. Data rate is 100Mbps.This case study helps technicians to examine the applicability of Polar code to 5G systems. Digital image example is considered as shown in the Figure11 at receiver end after transmission through

Fig. 11. Input Digital Image (URL, 2015)

AWGN BSC channel digital output obtained in MATLAB command window after due mathematical

Fig. 12. Image reconstructed at the receiver

A transformation example image transmitted and reconstructed is shown in Figure12image reconstructed is visualized to be a good quality one. It is almost similar to the original image transmitted at the transmitter end.

Conclusion

Polar codes have been successfully simulated in MATLAB. Polar encoder and SC decoders were also successfully implemented in MATLAB. Simulation parameters are for iterations 2000 and 5000, for N = 1024 and K = 512 at the rate of R = 0.5. Successive cancellation decoder is most popularly used because of its reduced complexity and better efficiency for various code lengths. Bit Error Rate(BER) for turbo code is 10^{-2} at $3dB$ whereas, polar code documents BER of 10^{-5}at $3dB$ in present work. There is scope for polar code development. Research focus in work can be in polar encoders, decoders, error performance analysis, decoding, convergence, diverse modulation techniques.

Acknowledgment

In this work First author contributions are on code development. Second author / Third author contributions are in development of Polar code in MATLAB and on documentation works. Authors acknowledge Support of Management and Principal of MSRIT. This work is carried out on funding Support obtained from VGST K-Fist II scheme of Government of Karnataka, to MSRIT Bangalore with Sanction number: Ksteps/VGST-K-FIST /L2/2018-19/GRN No 759/315. This fund is to develop Centre of Learning for setting up Microwave and Antenna Lab.

References

Kim, Haesik(2020). Design and Optimization for 5G Wireless Communications. In *John Wiley* &*Sons 2020*.'16.John wiley US

Shannon & Claude E (1949). Communication theory of secrecy systems *The Bell system technical journal 1949*, 28 (4), 656-715.'16. Published by Nokia Bell laboratories

Vucetic, Branka & Yuan (2012). Turbo codes: principles and applications. In *Springer Science* & *Business Media*, 559 ,.'16.Publsihed by Springer Science & Media

Berrou, Claude & Glavieux, Alain (1996). Near optimum error correcting coding and decoding: Turbocodes IEEE Transactions on communications,1996, 44 (10)pp(1261-1271).16 :IEEE publication. IEEE Press.

Shokrollahi, &Amin. (2004). LDPC codes: An introduction. Coding, cryptography and combinatorics. In Springer Press 2004, pp(85–110)16 :Springer publication. Springer Press.

Vaz, Aldrin Claytus & Nayak, C Gurudas & Nayak, Dayananda (2019) Performance comparison between turbo and polar codes. 2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA),2019 pp(1072–1075)16 :IEEE publication. IEEE.

Arikan, Erdal (2009). Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels In Proceed- ings IEEE Transactions on information Theory 2009Vol(55)(7) pp(3051-3073)16 :IEEE publication. IEEE.

Hu, Man and Li, Jianping & Lv, Yansong (2017). A comparative study of polar code decoding algorithms 2017 In Proceedings IEEE 3rd Information Technology and Mechatronics Engineering Conference (ITOEC) 2017 pp(1221-1225)16 :IEEE publication. IEEE.

Balatsoukas-Stimming, Alexios & Giard, Pascal and Burg, Andreas(2017). Comparison of polar decoders with existing low-density parity-check and turbo decoders In: 2017 IEEE Wireless Communications and Networking Conference Workshops (WCNCW); 2017 pp(1-6)16 :IEEE publication. IEEE.

Tahir, Bashar and Schwarz, Stefan &Rupp, Markus, (2017). BER comparison between Convolutional, Turbo, LDPC, and Polar codes,In;2017 24th international conference on telecommunications (ICT),2017 pp(1-7) 16 :IEEE publication. IEEE.

Cyriac, Ajith & Narayanan, Gayathri.(2018). Polar code encoder and decoder implementation,In;2018 3rd International Conference on Communication and Electronics Systems (ICCES),2018.pp(294–302) 16 :IEEE publication. IEEE. (pp. 1–9).

Mahdavifar, Hessam & El-Khamy, Mostafa & Lee, Jungwon and Kang, Inyup . (2013). Compound polar codes,In;2013 Information Theory and Applications Workshop (ITA),2013 pp(1-6) 16 :IEEE publication. IEEE

Mohammed, Nada Fadhil &Ali, Suhad Ahmed & Jawad, Majid Jabbar(2020). Biometric-based medical watermarking system for verifying privacy and source authentication,In;2020 Kuwait Journal of Science(47) (3)pp(2-13) 16 :Kuwait Journal Publication.

Yuan, Bo ,& Parhi, Keshab K. (2013). Successive cancellation decoding of polar codes using stochastic computing In;2015 IEEE International Symposium on Circuits and Systems (ISCAS),2015 pp(30403043) 16 :IEEE publication. IEEE

Weblink:https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQuykPglusC_Hp-pB0tkTh1hTyOQr9Cu7ISng&usqp=CAU