International Journal of Trend in Scientific Research and Development (IJTSRD)
Volume 4 Issue 6, September-October 2020 Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

MATLAB Implementation of Polar Codes for 5G Systems

Vijaykumar S[1], Vidyarani H J[2]

[1]M. Tech, [2]Asistent Professor, Department of ISE,
[1,2]Dr. Ambedkar Institute of Technology, Bengaluru, Karnataka, India

ABSTRACT
Channel Coding for the 5G wireless networks is facing novel challenges. Turbo and LDPC code doesn't meet the requirements of 5G usage scenarios. Thus polar code are evolved. Different decoding approaches are used, such as SC, SCL at the receiver end of the digital communication system. Depending on the nature of the communication system appropriate decoding procedure is used. The performance of these decoding strategies is analysed by plotting BER v/s SNR graphs. For this purpose MATLAB is used which gives better results and computations.

KEYWORDS: *Polar code*

How to cite this paper: Vijaykumar S | Vidyarani H J "MATLAB Implementation of Polar Codes for 5G Systems" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470, Volume-4 | Issue-6, October 2020, pp.314-318, URL: www.ijtsrd.com/papers/ijtsrd33344.pdf

IJTSRD33344

1. INTRODUCTION

The wireless communication system is improving rapidly nowadays. 1G or first generation was introduced in 1980s and had a practical speed of 2.4 kbps, it provided mobile telephony services. 2G or second generation provided digital voice and short messaging and was introduced in 1990 with a practical achieving speed up to 64 kbps. 3G or third generation was introduced in 2003 with a practical achieving speed up to 2 mbps, and was providing services such as integrated high-quality audio, video and data. 4G or fourth generation was introduced in 2009 with a practical achieving speed of 100 mbps and provided dynamic information access, variable devices services. 5G or fifth generation also provided services such as dynamic information access, variable devices along with AI capabilities and was introduces in the year 2020 and can practically achieve a speed of 1 gbps.

Turbo and LDPC codes doesn't meet the requirements of 5G communication systems. Hence polar codes are used instead of them due to its useful and advantageous properties. Polar code construction is based on channel polarization and this concept is introduced by Erdal Arikan in 2009. Channel polarization is very similar to polarization of light. That is the polarized light have its energy only in one direction and its energy is tends to zero in any other direction. Polarized light can be obtained by passing unpolarized light through a polarizer. In the same way an ordinary channel W is transformed into two extreme channels that is, these channels are either i) Perfect/ Good – A noiseless channel that transmits information without error (max. capacity C(W) = 1) known as reliable channel, or, ii) Useless/ bad – An extremely noisy channel that transmits only random noise (C(W) = 0) known as unreliable channel.

The important characteristics of polar codes are:
- Error correction performance achieves desirable capacity.
- They are easier decode and encode algorithms.
- Clear and straightforward implementation
- Simple to design and straightforward higher performance.
- Efficiency of hardware is high
- Code can be constructed recursively based on channel specificity.
- No error floor.
- Throughput can be further increased by having adaptive decoder with SC, SCL and CRC blocks.

Wide range of research is ongoing in the field of decoders. Here we are mainly concentrating on the decoding methods of polar codes in which we are going to see the brief concepts of encoding and decoding. There are many number decoders available in polar codes, they are successive cancellation decoder, successive cancellation list decoder, adaptive successive cancellation list decoder, adaptive decoder, maximum likelihood decoder, etc. We mainly concentrate on the successive cancellation decoder, adaptive successive cancellation list decoder and adaptive decoder.

Following sections describe polar encoding and several decoding approaches.

A. Polar Transform

Polar codes can be obtained by using the generator matrix G in recursive manner, that is we multiply the generator matrix by itself to obtain higher order generator matrices. The fundamental generator matrix is given by as follows:

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

Here the subscript 2 indicates that only code length (N) of 2 can be encoded and can be transmitted through the polarised channel. Hence it is possible to encode and transmit only powers of 2 number of bits, like N = 2, 4, 8, 16, 32……..

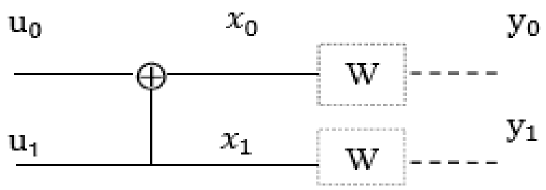

Figure 1: Single phase (N = 2) polar code factor graph

It can be asserted that:

$$x_0 = u_0 + u_1$$
$$x_1 = u_1$$

If $N \neq 2^n$, code can be generated by padding zeros until satisfying the rule $N = 2^n$ and these bits are treated as frozen bits.

In this project we are generating the polar codes of length $N = 2^{10} = 1024$ which is the standard requirement of 5G systems, but for the representation purpose $N = 2^3 = 8$ code is used.

The Generator matrix G to construct a code of length N is calculated as

$$G_N = (G_2)^n$$

A codeword c of length N can be calculated by,

$$c_N = u_i * G_N$$

where u_i is the input vector consisting of both data bits and frozen bits.

B. Successive Cancellation (SC) Decoder

Successive Cancellation (SC) Decoder makes its decisions on bits one by one. As seen in Fig 2 $\hat{u}_1$ is obtained by estimating the received code vector $\hat{u}_1$ is required to decode $\hat{u}_2$ along with the received code vector and $\hat{u}_1$ and $\hat{u}_2$ are required to decode$\hat{u}_3$ along with the received code vector So only one decision can be made at a time. This process continues in a similar fashion for higher order codes until all the nodes are processed [2]. Thus to decode the i^{th} bit, it requires partial modulo sums of previously estimated bits and LLR of channel output. The rules used for decoding are given below:

I. The estimate u_i will be estimated as zero, if the i^{th} bit belongs to the frozen bit set. [4]
II. The LLR value of the last stage is compared with a threshold value to estimate the result, which is generally zero, if the i^{th} bit does not belong to the frozen bit set. [4]

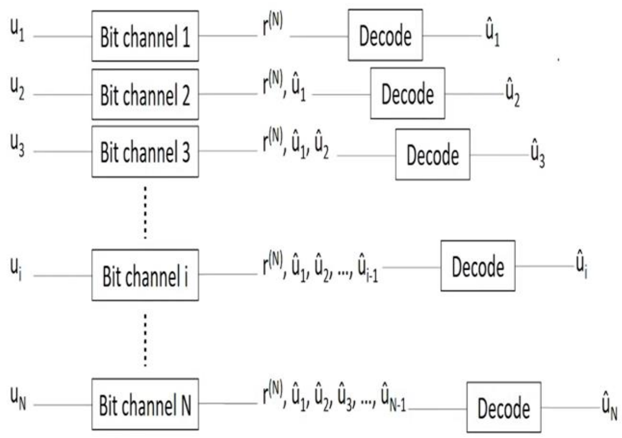

Figure 2: Illustration of SC decoder working principle

Algorithm: SC Decoder

Input: received codeword y^N_1, the code block length N, frozen bit vector u_{Ac} and the information set A

Output: estimated bits $\hat{u}_A$

```
Begin
    for i = 1 to N do
        if i ∉ A then
            û_i = u_i
    else
        if f(r1,r2) >= 0 then
            û_i = 0
    else
            û_i = 1
    return û_A
```

Three different functions are defined to illustrate the behaviour of the SC decoder. They are,

The function 'f' computes top channel splitting operation

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

The function 'g' calculates bottom channel splitting operation

$$g(r_1, r_2, \hat{u}) = r_1 (-1)^{\hat{u}} + r_2$$

Finally, the function 'd' is the decision function, that uses soft decisions to compute hard decisions such that,

$$u_i = \begin{cases} u_i & \text{if } u_i \in Ac \\ 0 & \text{if } u_i \in A \text{ and } f(r1, r2) \geq 0 \\ 1 & \text{others} \end{cases}$$

C. Successive Cancellation List (SCL) Decoder

A more powerful SC-List decoder is proposed and gives better performs than SC decoder. []The SC List decoder keeps L survival paths, instead of keeping only one survival path as in the SC decoder. The SC-List decoder discards the worst (least probable) paths, and only keeps the L best paths when the number of paths exceeds a predefined threshold L. In order to increase the error detection CRC is added. Only that path codeword which passes the CRC is considered as a valid codeword among the list of codewords. If multiple path codewords passes the CRC, then only that codeword which has highest path metric (PM) is treated as a valid codeword. In order to obtain valid codewords, two terms need to be determined first. They are Decision Metric (DM) and Path Metric (PM).

While parsing the decoder tree, consider both decision that is, '0' and '1' for each bit. Decision metric (DM) can take values as per the beliefs oragainst the beliefs of previously detected bits.

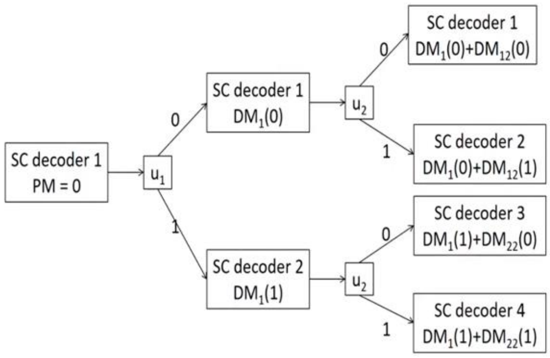

Figure 3: Illustration of path metric (PM) calculation

Assign decision metric for all decisions as given by the following rule:

if $L(u_i) \geq 0$: $\hat{u}_i = 0$ has $DM_i = 0$, $\hat{u}_i = 1$ has $DM_i = | L(U_i) |$

if $L(u_i) < 0$: $\hat{u}_i = 1$ has $DM_i = 0$, $\hat{u}_i = 0$ has $DM_i = | L(U_i) |$

Algorithm: SCL Decoder

Input: received codeword y^N_1, the code block length N, frozen bit vector u_{Ac} , list size L and the information set A

Output: estimated bits $\hat{u}_A$

1. Initialise list with empty path
2. Compute the path metric
3. Sort candidate path
4. Reserve L candidates and delete others
5. Check length L = N, if YES goto step 6 else goto step 3
6. Output K candidates in the order of decreasing metric
7. If L > 1goto step 9 else goto step 8
8. Output the sequence with least path metric
9. Check CRC bits
10. Output the path with least path metric

DM is even assigned to frozen bits, which produces conflicts during decoding of estimated bits. These conflicts can be resolved by evaluating path metric. Path metric (PM) of a path is the sum of the decision metric (DM) of all bits of that path L. This PM value is used in decoding of estimated bits.

Only that path codeword which passes the CRC is considered as a valid codeword among the list of codewords. If multiple path codewords passes the CRC, then only that codeword which has highest path metric (PM) is treated as a valid codeword.

D. Adaptive Decoder

Fig 4 shows only receiver section of digital communication system which involves adaptive decoder of polar codes. At the receiver end first decoding procedure is starts with SC decoder. The resultant decoded information bits is transferred to a CRC block. If the code vector clears CRC then it is considered as valid decoder output. Else the same decoded vector of SC decoder which doesn't clear CRC is passed to a SCL decoder. Now the SCL decoder is activated and L information vector candidates are generated [5]. Among these candidates, the CRC decoder selects a code vector that clears CRC. If multiple code vectorsclears CRC, then the most probable one among these candidates is selected. Lastly, if none of the candidates clears CRC vector, the CRC decoder selects the most probable decision estimation vector to reduce BER [7].

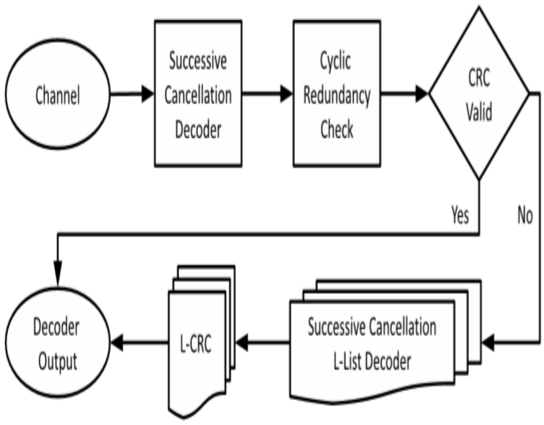

Figure 4: Adaptive decoder with SC, SCL and CRC blocks

Algorithm: Adaptive Decoder

Input: received codeword y^N_1, the code block length N, frozen bit vector u_{Ac} , list size L and the information set A

Output: estimated bits $\hat{u}_A$

Variable: j valid CRC vector SCD

Variable: k valid CRC vector of SCLD

```
Begin
    ûA = SC Decoding (yN1, N, uAc and A)
    j = CRC(ûA)
    if j is true
        return ûA
    else
        ûAL = SC Decoding (yN1, N, uAc , L and A)
    for l = 1 to L do
        k = CRC(ûAl)
        if k is true then
            ûA = ûAL
            return ûA
        ûA = ûAL
        return ûA
```

E. Results and Conclusions

In this paper, I concentrated on encoding and decoding of polar code using MATLAB.BER vs E_b/N_0 plot is used to analyse the performance of each decoding algorithm [13]. These graphical analysis prove that BER drops to zero for small value of energy density (E_b/N_0). Thus small amount of energy is enough to transmit information between sender and receiver unlike Turbo and LDPC codes.

Polar encoder and decoders are explained successfully, SCD decoder, SCL decoder and adaptive decoder is successfully explained and implemented using MATLAB compiler. All the decoders are implemented for different number of iterations 2000 and 5000 respectively, for N = 1024 and K = 512 at the rate of R = ½. The waveforms obtained for these above-mentioned values is plotted respectively for all the decoders explained. From all of these observations I can say that SC decoder is very fast decoder and has a very good efficiency for longer message bits but inefficient for shorter message and medium length bits. SCL decoder is not as fast as SC decoder due to its high computation and complexity in the algorithm, but it is very efficient for shorter and medium length message blocks. Adaptive decoder combines results a better efficiency than SC decoder and is faster than SCL decoder

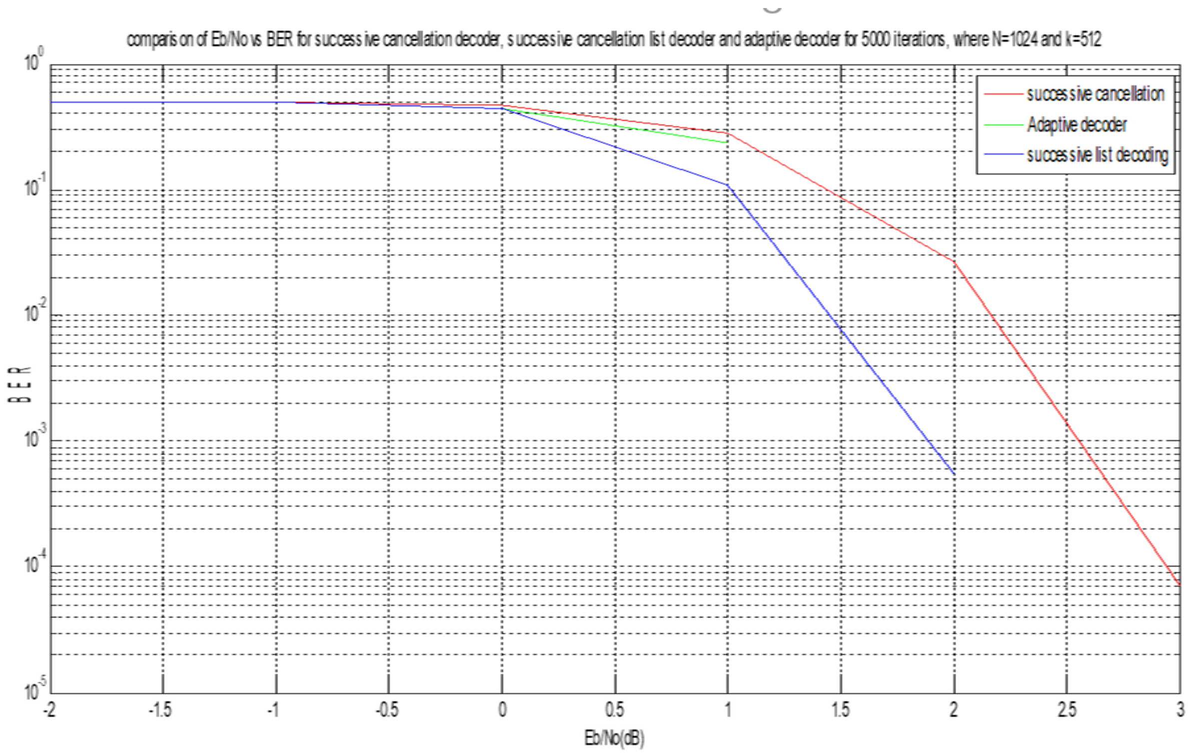

Figure 5: Comparison of SC, SCL and Adaptive decoder for 5000 iterations, where N = 1024 and A = 512

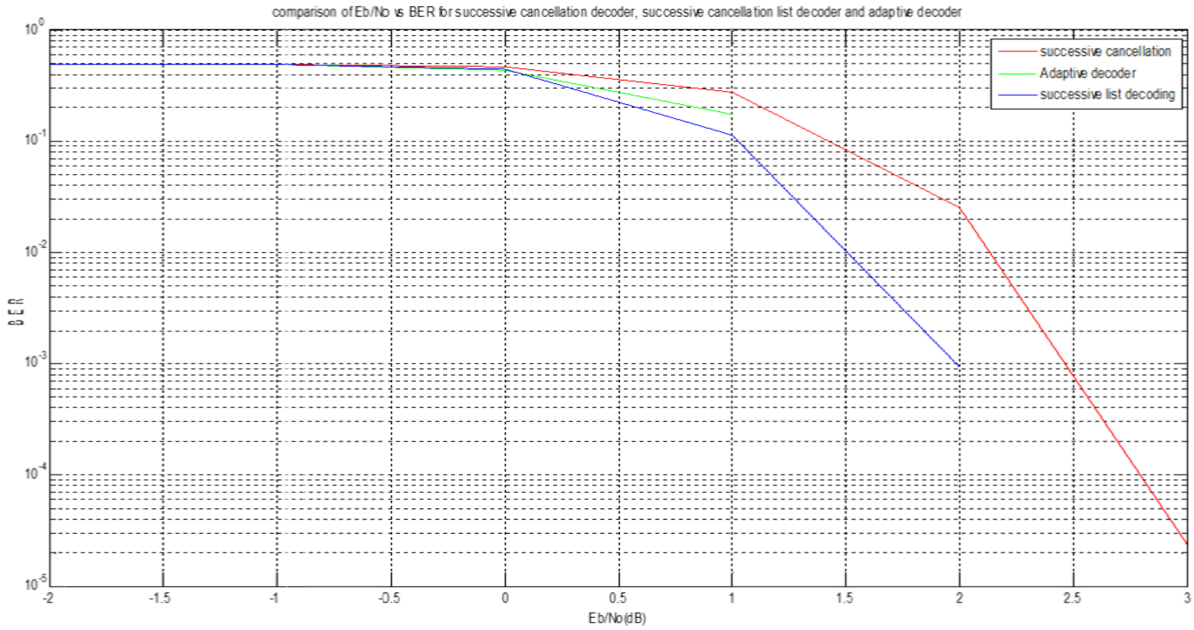

Figure 6: Comparison of SC, SCL and Adaptive decoder for 2000 iterations, where N = 1024 and A = 512

References

[1] Wenqing Song, Chuan Zhang, Shunqing Zhang, and Xiaohu You "Efficient Adaptive Successive Cancellation List Decoders for Polar Codes", 978-1-5090-4165-7/16 IEEE, 2016

[2] Aldrin Claytus Vaz, C Gurudas Nayak and Dayananda Nayak "Performance Comparison between Turbo and Polar Codes", 978-1-7281-0167-5/19 IEEE, 2019

[3] Man Hu, JianpingLi, Yansong Lv "A Comparative Study of Polar Code Decoding Algorithms", 978-1-5090-5363-6/17 IEEE 2017

[4] Ajith Cyriac, Gayathri Narayanan "Polar Code Encoder and Decoder Implementation", 978-1-5386-4765-3/18 IEEE 2018

[5] Erdal Arikan "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels", IEEE 2009

[6] Aarti Sharma, Mohammad Salim "Polar Code: The Channel Code Contender for 5G Scenarios", 978-1-5090-4708-6/17/$31.00 ©2017 IEEE

[7] Fang Yuan, Bin Tian "Double-Parity-Check CA-SCL Encoding and Decoding for Polar Codes", 978-1-5386-4673-1/18/$31.00 ©2018 IEEE

[8] Marwan DHUHEIR, Sıtkı ÖZTÜRK "Polar Codes Analysis of 5G System", 978-1-5386-7641-7/18/$31.00 ©2018 IEEE

[9] Mahdi Bersali, Hocine Ait-Saadi and Messaoud Bensebti "Performance Analysis of Polar Codes vs Turbo Codes over AWGN channel", 978-1-5386-0686-5/17/$31.00 ©2017 IEEE

[10] Jun Lin "A High Throughput List Decoder Architecture for Polar Codes", 1063-8210 © 2015 IEEE

[11] Ido Tal "List Decoding of Polar Codes", 0018-9448 © 2015 IEEE

[12] Wenqing Song, Chuan Zhang, Shunqing Zhang, and Xiaohu You "Efficient Adaptive Successive Cancellation List Decoders for Polar Codes", 978 1 5090 4165 7/00 ©2016 IEEE

[13] Bashar Tahir, Stefan Schwarz, and Markus Rupp "BER Comparison Between Convolutional, Turbo, LDPC, and Polar Codes", 978-1-5386-0643-8/17 IEEE 2017

[14] Alexios Balatsoukas-Stimming, Pascal Giard, and Andreas Burg "Comparison of Polar Decoders with Existing Low-Density Parity-Check and Turbo Decoders", IEEE 2017