Polar Codes

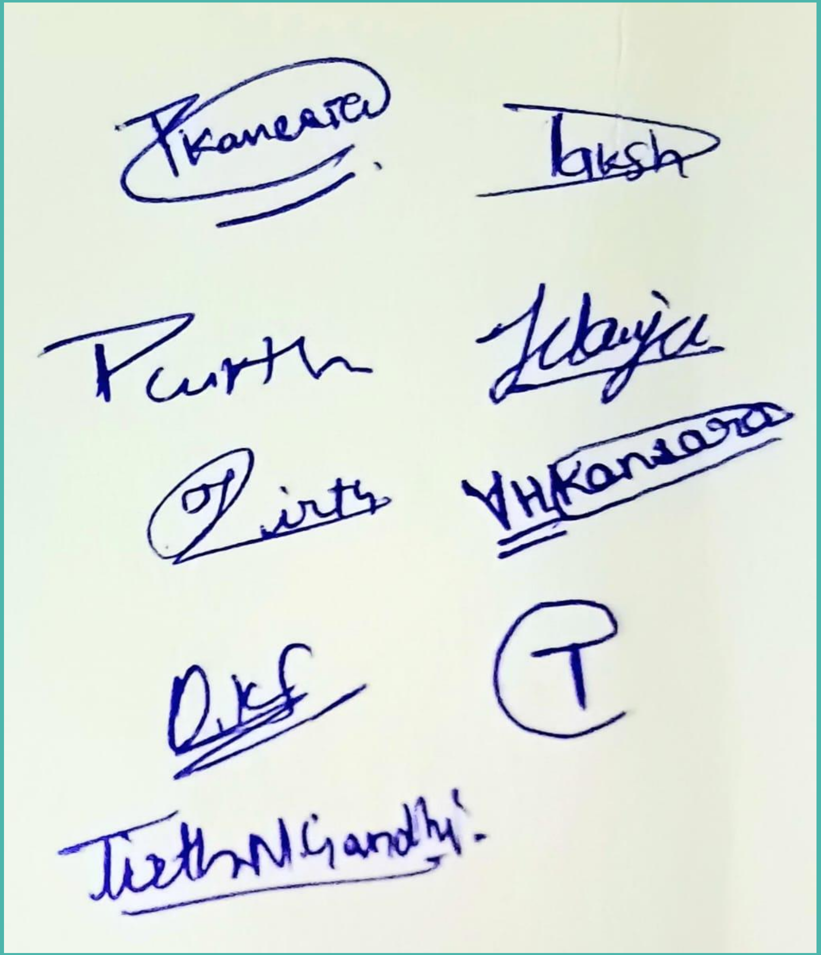
Group - 2

Prof. Yash Vasavada

Honor code

We declare that

- The work that we are presenting is our own work.
- We have not copied the work (the code, the results, etc.) that someone else has done.
- Concepts, understanding and insights we will be describing are our own.
- We make this pledge truthfully. We know that violation of this solemn pledge can carry grave consequences.



PKaneria

Daksh

Parth

VHKansara

Tirth N Gandhi.

Overview

1. Introduction to Polar Codes
2. Polarization : A simple Analogy for everyone
3. The Bhattacharyya Parameter
4. Binary Erasure Channel
5. Polarization in Action
6. A theorem for Polarization
7. Polar Transform
8. Encoding
9. BPSK and AWGN
10. Decoding
11. Plots
12. Proof of Achieving Shannon’s Capacity Bound
13. Comparison with other codes
14. References

Introduction to Polar Codes

- Introduced by Erdal Arikan in 2009.

- First codes to achieve channel capacity for Binary Discrete Memoryless

  Symmetric Channels (B-DMC).

- Using clever technique, channels are transformed into either very good channels or very bad ones, i.e. **channels are polarized**.

- Polar codes have low encoding and decoding complexity.

- Due to its exceptional performance, polar codes are adopted by 3rd Generation Partnership Project (3GPP) for 5G-NR (New Radio) control channels.

Polarization: A simple analogy for everyone

- Coaching Institutes often separate students based on their performance in mock tests.
    - Students who perform well, get more attention and help from teachers.
    - Struggling and mediocre students are given less attention (or **ignored**).
- In a similar manner, Polar Codes separate channels in good channels or bad channels,and focuses only on the good channels.
    - The polar codes classifies channels into good or bad using **Bhattacharyya Parameter ($\mathscr{Z}$).**

The Bhattacharyya Parameter ($\mathscr{Z}$)

- It gives a limit to the maximum probability of error during decision making using Maximum Likelihood (ML) rule.

- For Binary Input Channels it is given by:

$$\mathscr{Z} = \Sigma_{y\ \varepsilon\ Y}\ (P(y|0)P(y|1))^{½}$$

Y = Set of possible output values at the receiver.

- Higher the value of $\mathscr{Z}$, less reliable the channel is.

Binary Erasure Channel (BEC)

- It is a discrete memoryless channel with input alphabets $\mathcal{X}$ = {0 , 1} and output alphabets **Y** = {0 , 1 , e}.

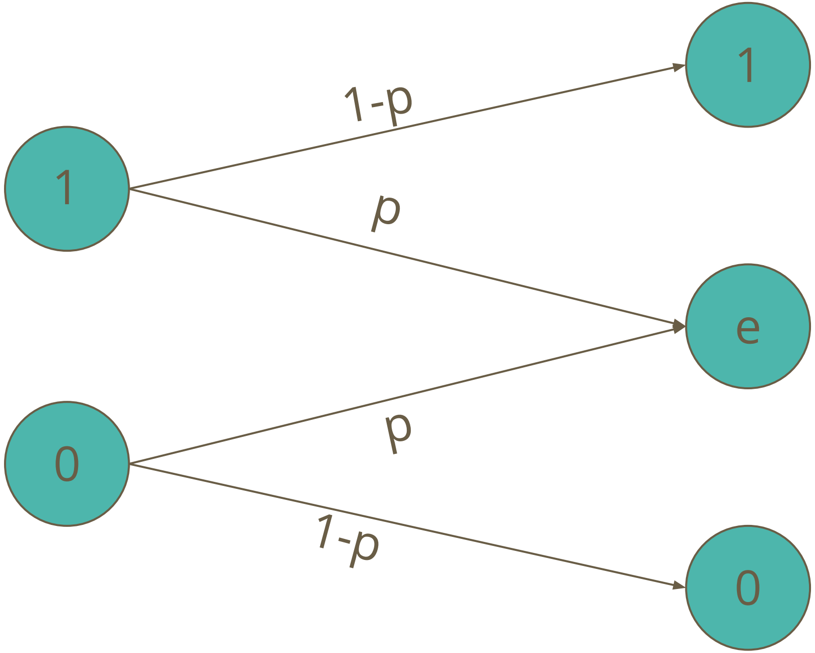
- The channel is characterized by parameter p , called ***erasure probability.***

- With probability of 1-p, the output is same as input, and with probability of p, the output is replaced by erasure symbol.

- Denoted by BEC(p).

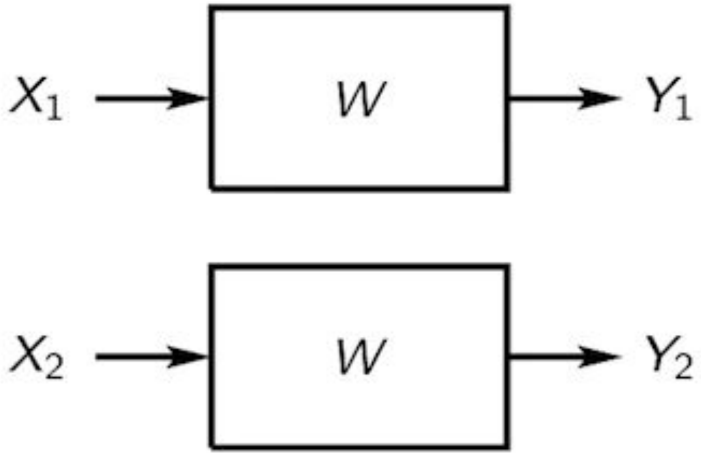
- The Bhattacharyya Parameter $\mathcal{Z}$ for BEC(p) is : $\boldsymbol{\mathcal{Z} = p}$

Binary Erasure Channel (BEC)



Polarization in Action : 2-bit Polar Code Example

- Consider a BEC W with some erasure probability p used to transmit two bits X_1 and X_2.
- Output is Y_1 and Y_2 respectively.
- The erasure probability in both cases is p.

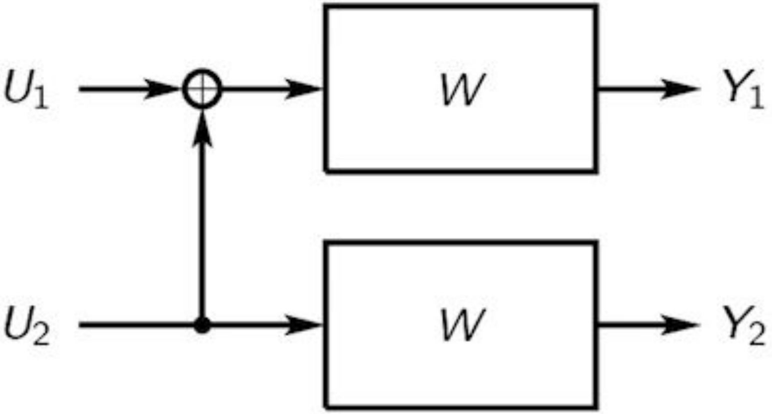


Polarization in Action : 2-bit Polar Code Example

- Consider two bits U_1 and U_2 which are then passed as inputs to X_1 and X_2 in following way:

$$[X_1 , X_2] = [U_1 \oplus U_2 , U_2]$$

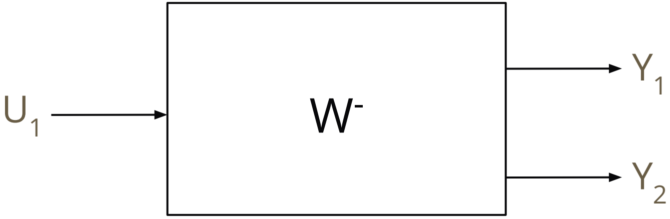
- Consider Two Channels , W^{-} and W^{+} .



$W^- : U_1 \longrightarrow (Y_1, Y_2)$

- W^- tries to decode/reconstruct U_1 given output Y_1 and Y_2.
- The output possibilities are as follows: -

$$(Y_1, \quad Y_2) = \begin{cases} (U_1 \oplus U_2, U_2) & \text{w.p. } (1-p)^2 \\ (e, U_2) & \text{w.p. } p(1-p) \\ (U_1 \oplus U_2, e) & \text{w.p. } (1-p)p \\ (e, e) & \text{w.p. } p^2 \end{cases}$$

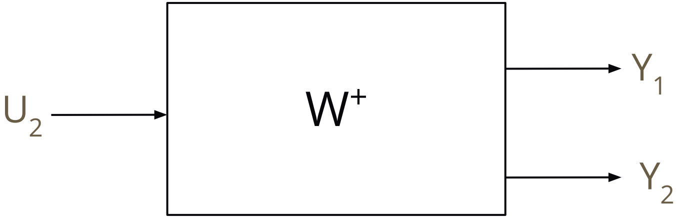


- Probability of decoding failure $= 2p - p^2$
- Probability of decoding success $= (1 - p)^2$

$W^+ : U_2 \longrightarrow (Y_1, Y_2, U_1)$

- W^+ tries to decode/reconstruct U_2 given output Y_1, Y_2 and decoded U_1 .
- The output possibilities are as follows: -

$$(Y_1, Y_2, U_1) = \begin{cases} (U_1 \oplus U_2\ , U_2\ , U_1) & \text{w.p. } (1-p)^2 \\ (e\ , U_2\ , U_1) & \text{w.p. } p(1-p) \\ (U_1 \oplus U_2\ , e\ , U_1) & \text{w.p. } (1-p)p \\ (e\ , e\ , U_1) & \text{w.p. } p^2 \end{cases}$$



- Probability of decoding failure $= p^2$
- Probability of decoding success = $2p - p^2$

Polarization in Action : 2-bit Polar Code Example

- Since $0 < p < 1$:

$$p^2 < 2p - p^2$$

- $\mathscr{Z}(W^+) = p^2$ and $\mathscr{Z}(W^-) = 2p - p^2$
- Thus , W^- is a bad channel compared to W^+.
- Thus, we effectively **polarized** the channel W into a good and a bad channel.

Polarization in Action : 4-bit Polar Code Example

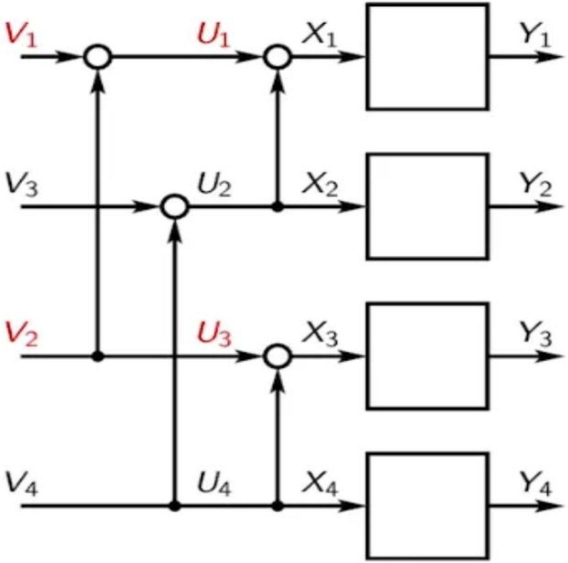
- To make 4 bit encoder, we can use 2 , 2 bit encoder as follows:
    - And obtain

$W^{--} : V_1 \longrightarrow Y_1Y_2Y_3Y_4$

$W^{-+} : V_2 \longrightarrow Y_1Y_2Y_3Y_4V_1$

$W^{+-} : V_3 \longrightarrow Y_1Y_2Y_3Y_4V_1V_2$

$W^{++} : V_4 \longrightarrow Y_1Y_2Y_3Y_4V_1V_2V_3$



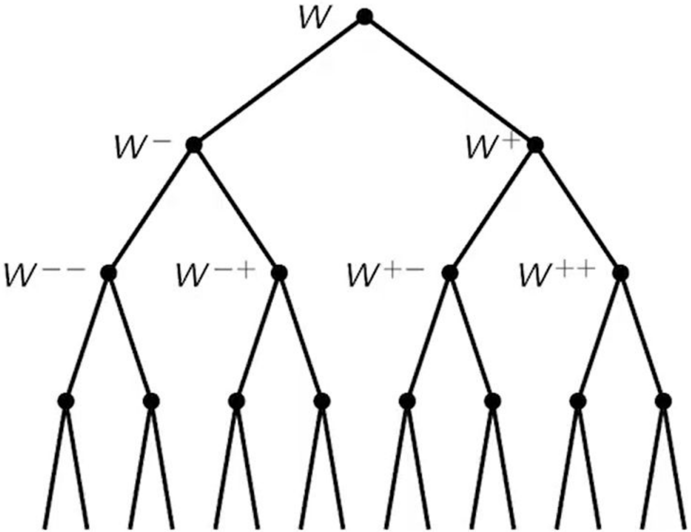
Family Tree of Channels

t = 0

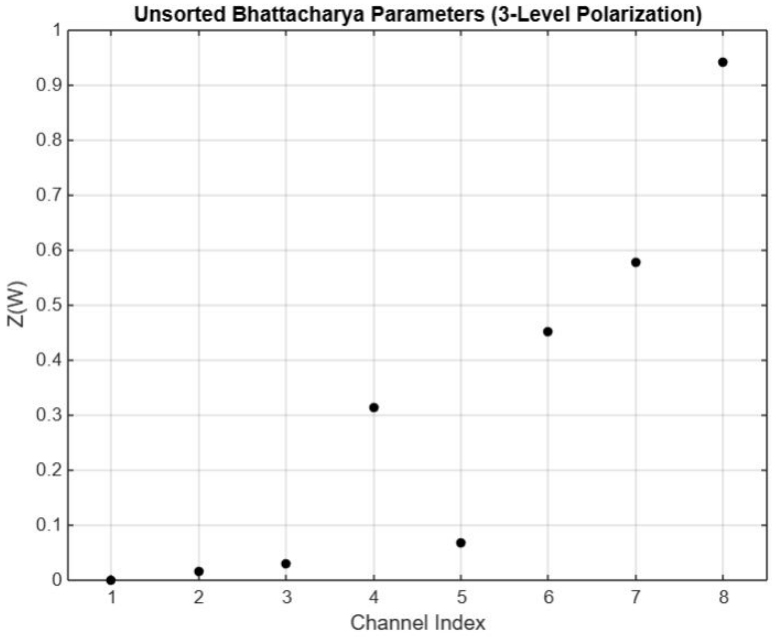
t = 1

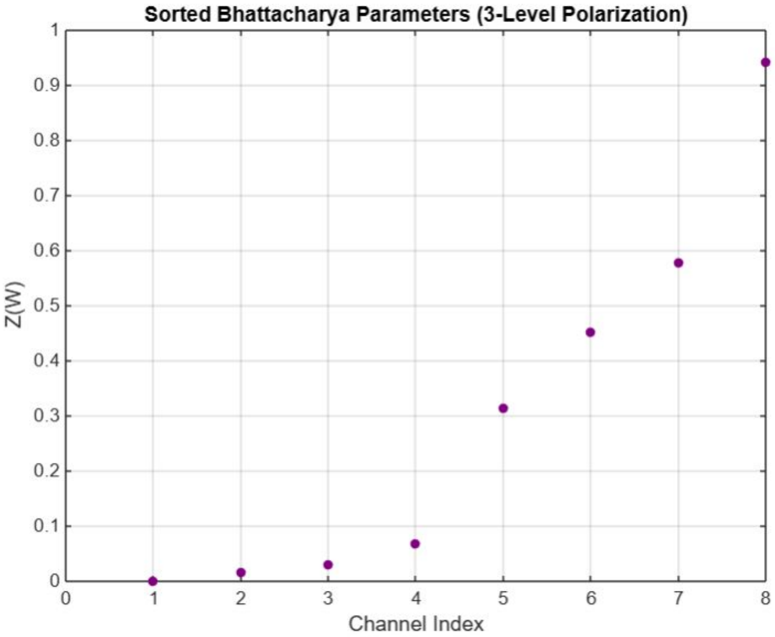
t = 2

t = 3

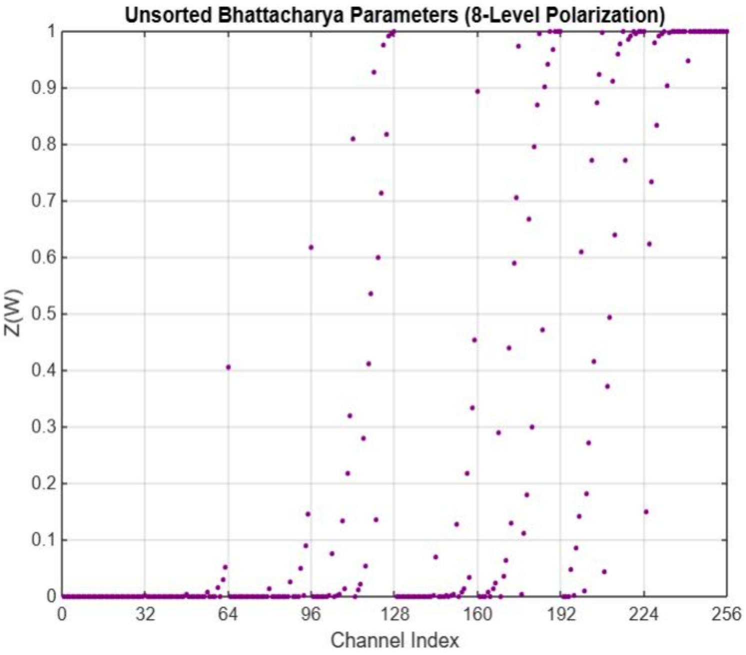


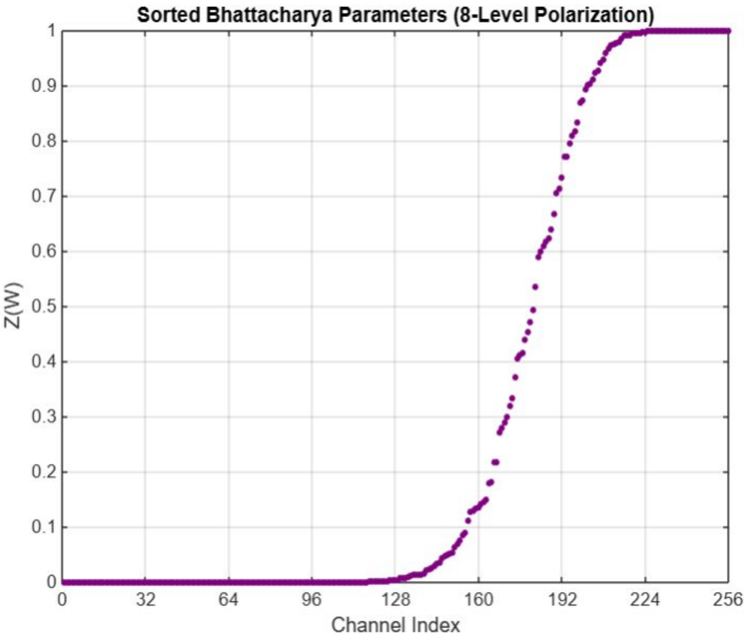
Polarization for t = 3 and $p = 0.3$



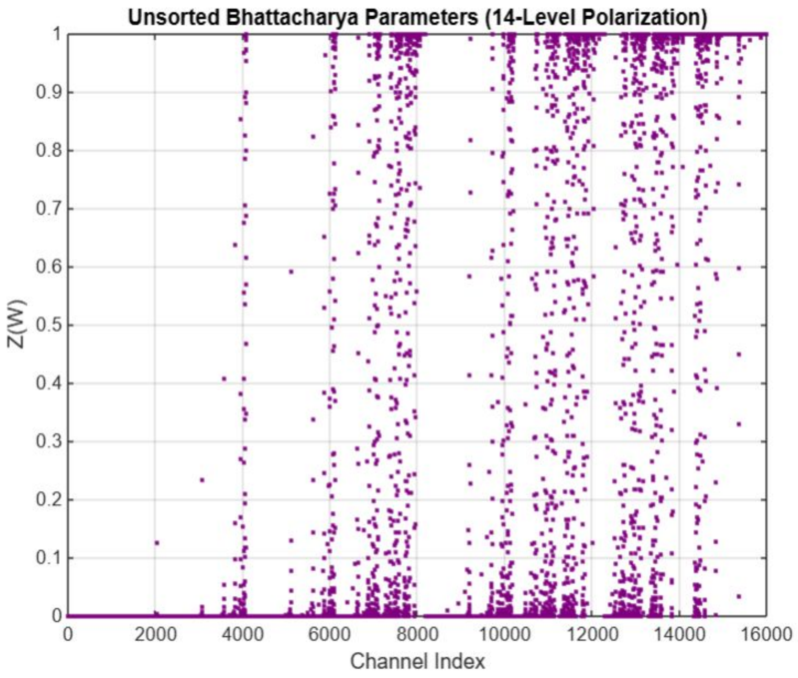


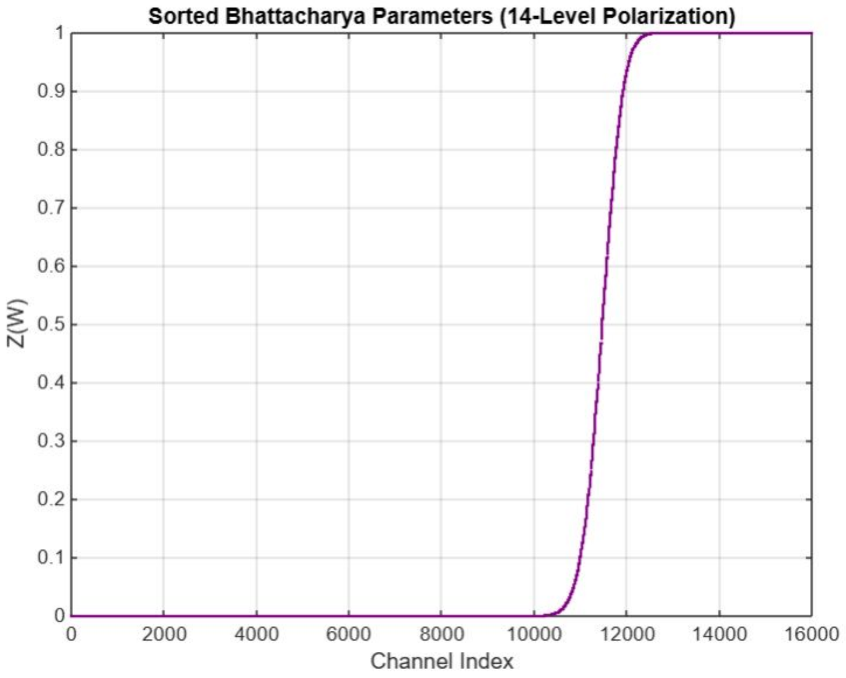
Polarization for t = 8 and p = 0.3





Polarization for t = 14 and $p = 0.3$





A Theorem for Polarization

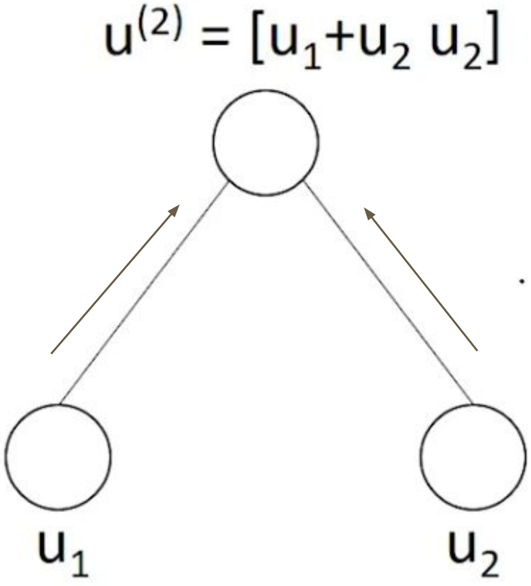
For any ϵ $>$ 0 , and any channel *W*, the fraction of ϵ - mediocre channels vanishes when we repeatedly apply transform.

$$\mu_t(\epsilon) := \frac{1}{2^t} \sum_{s^t \in \{+,-\}^t} \mathbf{1}\left\{I(W^s) \in (\epsilon, 1-\epsilon)\right\}, \quad \lim_{t\to\infty} \mu_t(\epsilon) = 0$$

- **Note for a mediocre channel : 0< $\mathcal{Z}(W^s)$ < 1**

Polar Transform : 2 bits

- Binary Tree Representation of Polar Transform of 2 bits is given as:
  - **G** = $\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$
- **$[u_1 \ , u_2]$ G = $[u_1 \oplus u_2 \ , u_2]$**



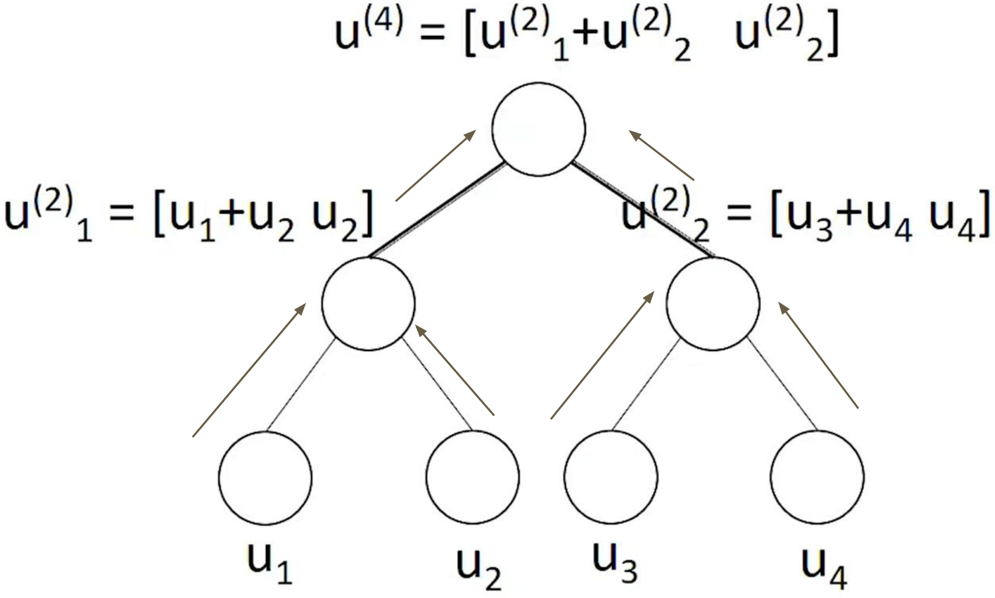
Polar Transform : 4 bits

- Similarly for 4 bits the polar transform can be given as :

$$G^4 = G^2 \otimes G^2$$

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

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



$$u^{(4)} = [u^{(2)}_1+u^{(2)}_2 \quad u^{(2)}_2]$$

$$u^{(2)}_1 = [u_1+u_2 \ u_2]$$

$$u^{(2)}_2 = [u_3+u_4 \ u_4]$$

- $[u_1 , u_2 , u_3 , u_4]G^4 = [u_1 \oplus u_2 \oplus u_3 \oplus u_4 , u_2 \oplus u_4 , u_3 \oplus u_{4,}, u_4]$

Encoding

- K message bits are encoded in N = 2^n vector bit.
- Transformation Matrix is :

  $\mathbf{G}^{\mathbf{n}} = \mathbf{G}^{\square \mathbf{n}}$

  - $\mathbf{G} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$
  - $\mathbf{G}^{\square \mathbf{n}}$ is the kronecker product of **G**

- For u vector of length N :
  - Find N- K least reliable channel and freeze them to zero.
  - Remaining K positions have K message bits.
- Codeword $\mathbf{c} = \mathbf{u}\mathbf{G}^{\mathbf{n}}$

Reliability Sequence

- Reliability Sequence is determined by Bhattacharyya Parameter.

  Eg : For 5G standard:

  N = 8 : 1 2 3 5 4 6 7 8

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

  N = 32 : 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

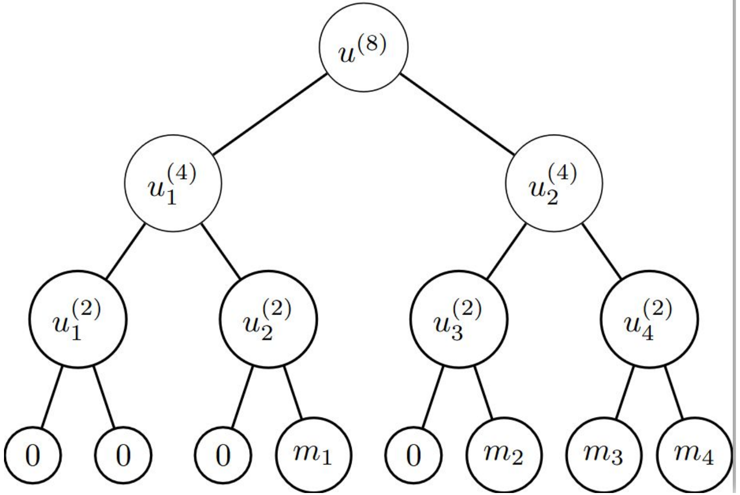
- Earlier a number appears , less reliable it is.

Polar code example: (8, 4)

Reliability Sequence for N = 8 : 1 2 3 5 4 6 7 8

Frozen : 1 2 3 5
Message : 4 6 7 8

$$\left[u_1^{(4)} + u_2^{(4)} \quad u_2^{(4)}\right]$$



Encoding Complexity

- It can be seen that the transformation matrix can be made recursively.

- Using method like butterfly network (a divide and conquer approach), the kronecker product can be obtained in ***O(nlogn)*** instead of ***O(n^2).***

- Multiplying by input vector takes ***O(n)***.

- Hence overall time complexity is ***O(nlogn)***.

BPSK MODULATION (Binary Phase Shift Keying)

- As polar codes are binary codes (0 & 1), so we need modulation to convert bits to analog waveforms.
- BPSK maps:

Bit	Amplitude
0	+1
1	-1

- Mathematical Representation: **$s = 1 - 2x$**, for x belongs to { 0,1 }.

AWGN (Additive White Gaussian Noise)

- After BPSK modulation, next step is to pass the symbols to AWGN channel.
- The AWGN channel adds Gaussian noise to the transmitted signal, characterized by normal distribution: **$N(0,\sigma^2)$**
- The received signal (r) will be of the form:

  $r = x + n$, where x is the BPSK modulated bit and n is $N(0,\sigma^2)$
- The error probability for BPSK is $Q(1 / \sigma)$.

Decoding

- We used *two* methods to decode the received codeword:

1. **Successive Cancellation Decoder (SC)** which decodes one bit at a time, by using the bits already decoded. It is simple but not always accurate.

2. **Successive Cancellation List Decoder (SCL)** which maintains multiple decoding paths and uses CRC to select the most reliable one.

- The **idea** we used here for decoding both the algorithms is that the *beliefs* [(**L**) gives confidence value 0 or 1, based on received signals] obtained from received bits to estimate the codeword bits and then we successively estimate the next codeword bits using the received beliefs and previously estimated codeword bits.

Decoding

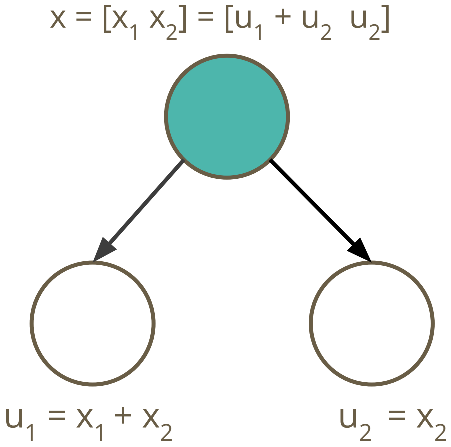
Calculation of Beliefs:

1. We calculate belief for first bit using SISO decoder for SPC **$u_1 = x_1 \oplus x_2$** The belief can be calculated using function f:

   $$L = f(r_1, r_2) = sgn(r_1) \cdot sgn(r_2) \cdot min(r_1, r_2)$$

2. Successively we use r_1, r_2, estimated u_1 to estimate u_2 using SISO decoder for RPC using function g:

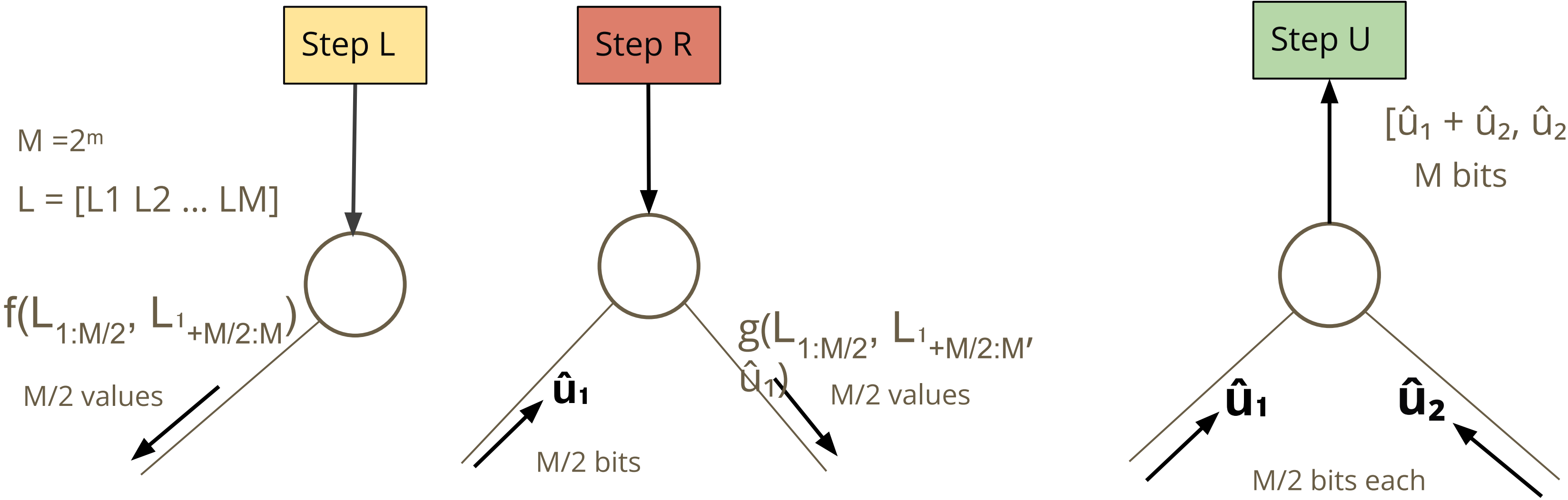
   $$L = g(r_1, r_2, u_1) = r_2 + (1 - 2 \cdot u_1) \cdot r_1$$



Decoding

Successive Cancellation Decoder:

We start from the root node and keep on transmitting beliefs to left and right child of node using f(r1, r2) and g(r1, r2, u1) functions till we reach the leaf node.



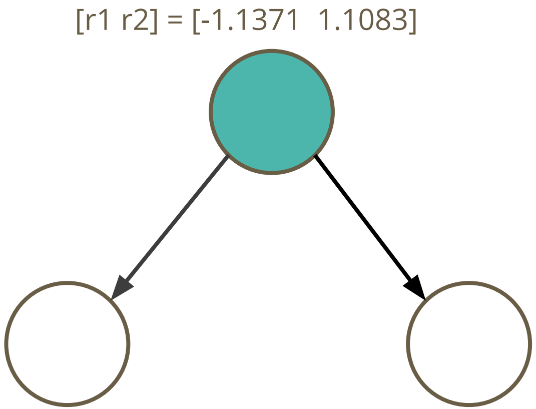
Once we reach the leaf node, we estimate the value of codeword depending on:

1. If frozen bit, the value of the bit is 0.
2. Else, we estimate the value of bit depending on sign of belief — we assign 0 if positive, else 1.

Decoding

Successive Cancellation Decoder

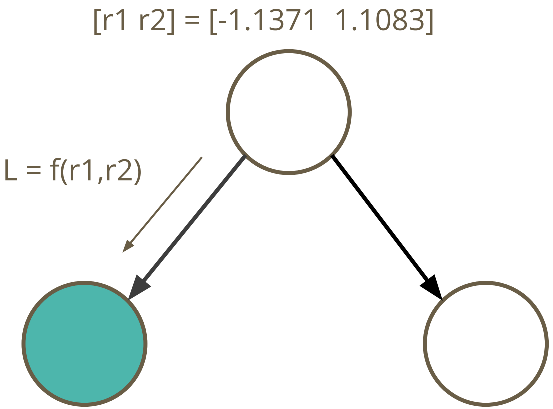
For 2 message bits,



Decoding

Successive Cancellation Decoder

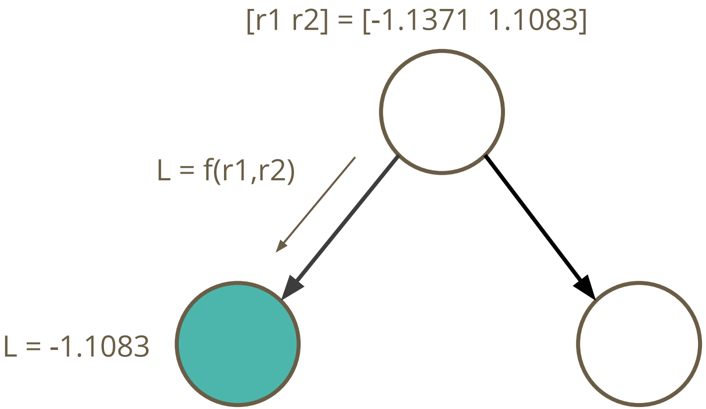
For 2 message bits,



Decoding

Successive Cancellation Decoder

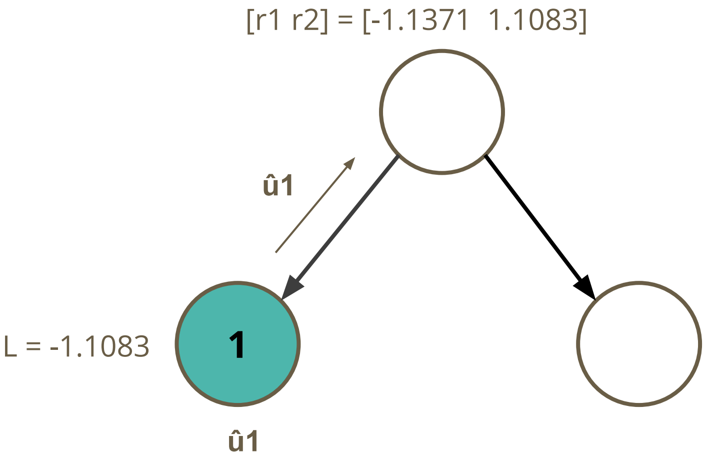
For 2 message bits,



Decoding

Successive Cancellation Decoder

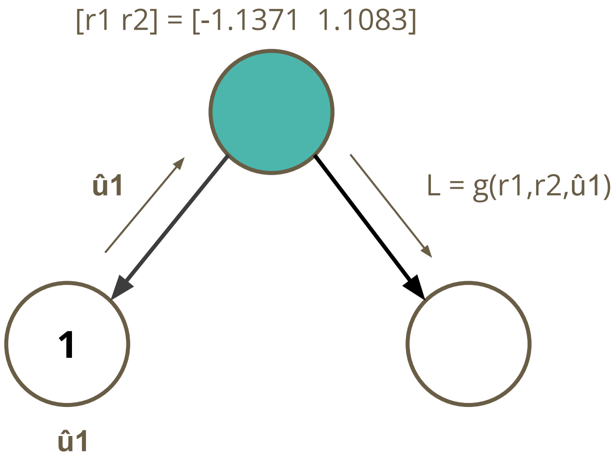
For 2 message bits,



Decoding

Successive Cancellation Decoder

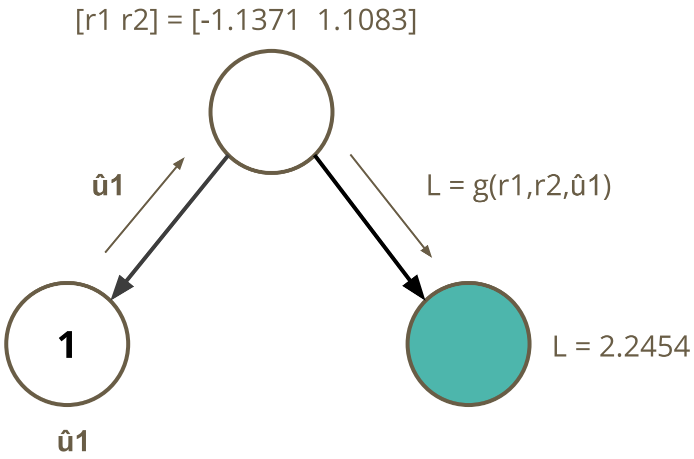
For 2 message bits,



Decoding

Successive Cancellation Decoder

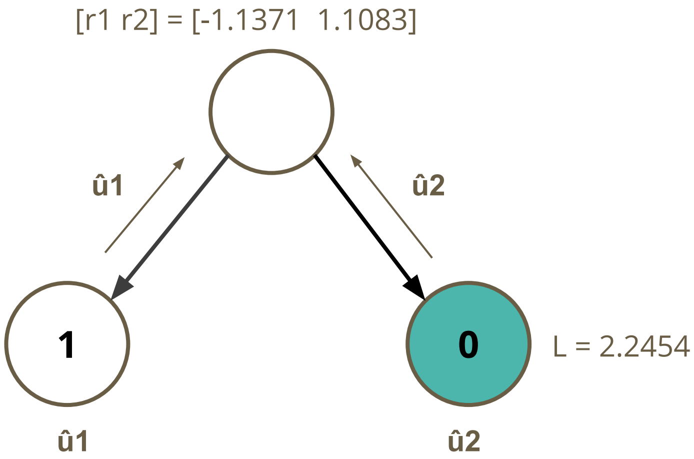
For 2 message bits,



Decoding

Successive Cancellation Decoder

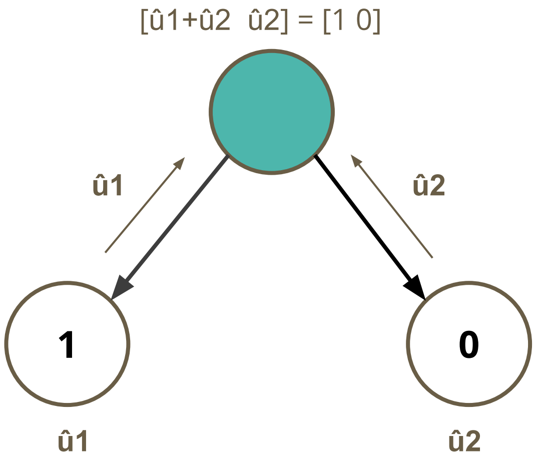
For 2 message bits,



Decoding

Successive Cancellation Decoder

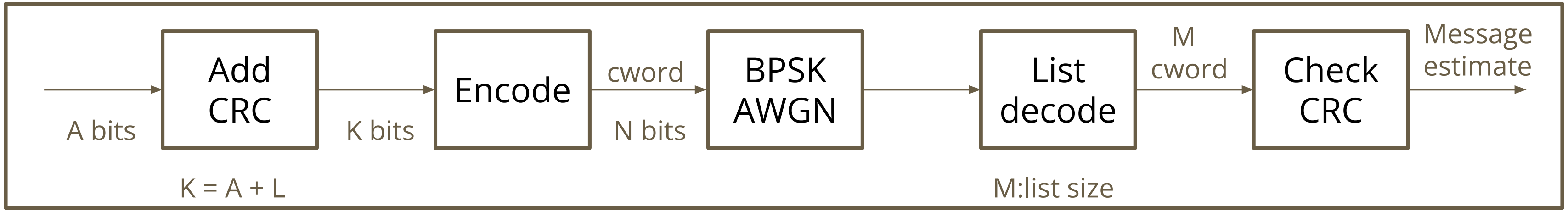
For 2 message bits,



Decoding

Successive Cancellation list Decoder

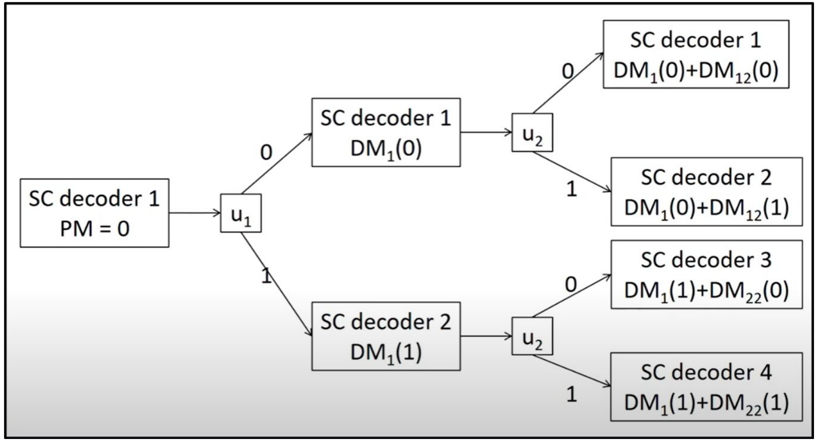
- The SCL decoding algorithm is quite similar to the SC decoder, but unlike SC decoding—where *u* bits are estimated based on the sign of the belief (L)—SCL considers both possible bit values. A penalty, equal to the absolute value of the belief (|L|), is added to the path metric when the non-estimated bit is chosen.
- We continue adding sequences to a list until a specified threshold is reached, after which the sequences with higher penalties are discarded.
- We also append CRC (Cyclic Redundancy Check) bits, which are traditionally used to identify the correct codeword from the list of candidates.

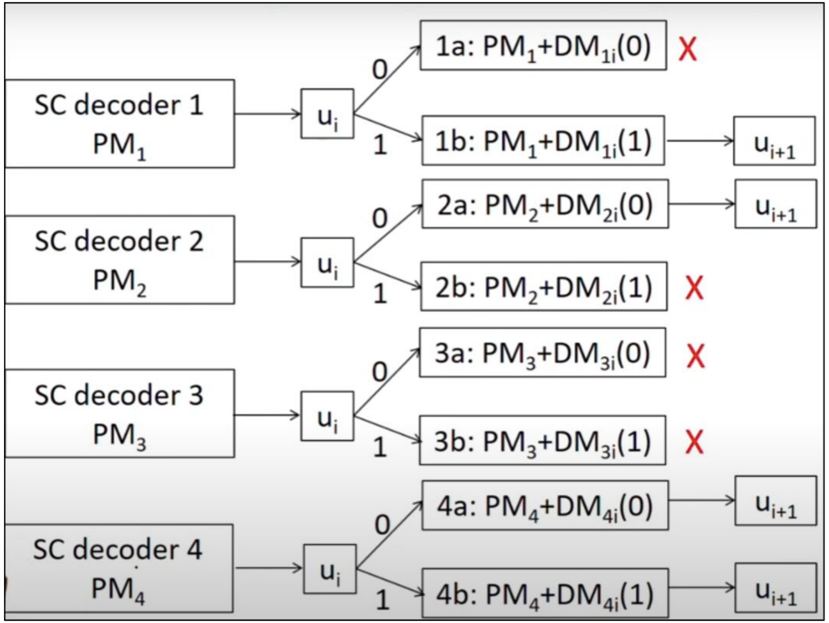


Decoding

Successive Cancellation list Decoder

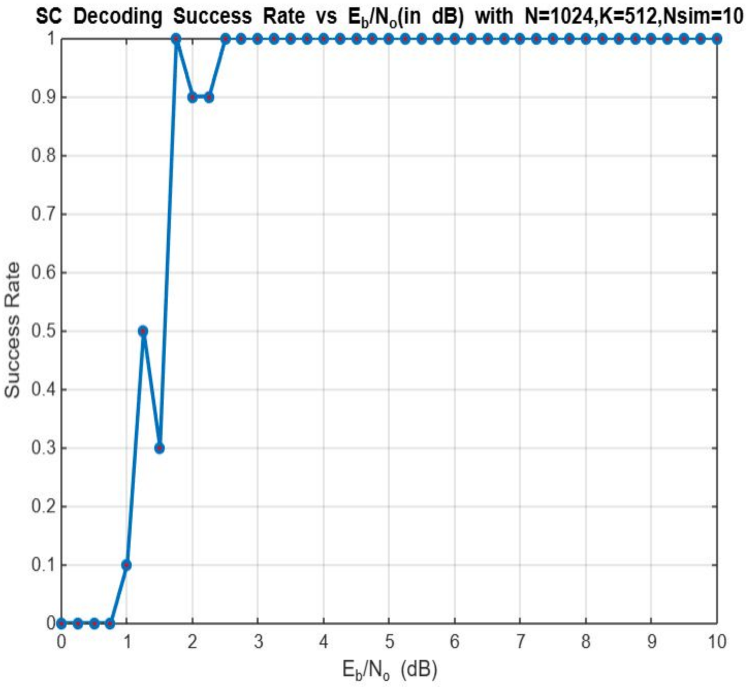
Diagramatic representation of SCL decoding process



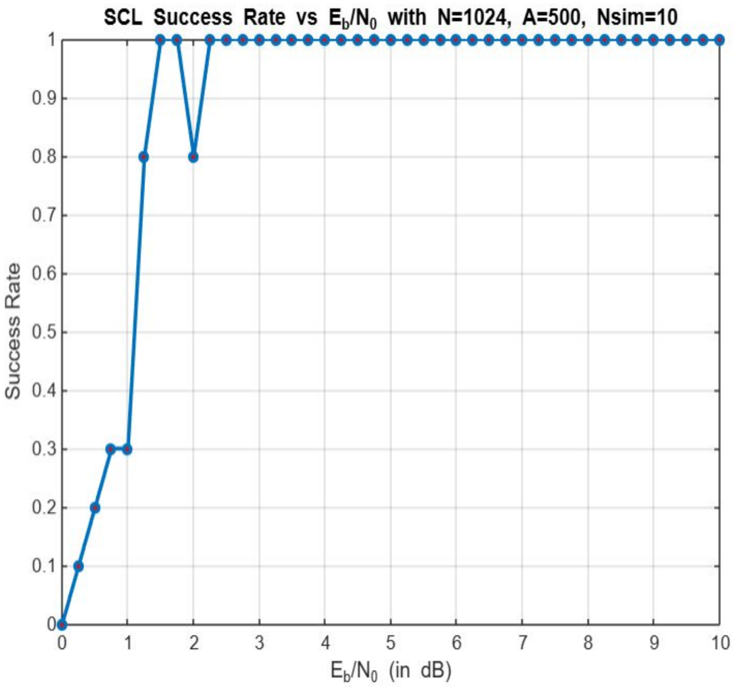


Plots(N=1024, K=512, Nsim = 10)

Success Rate for SC Decoder

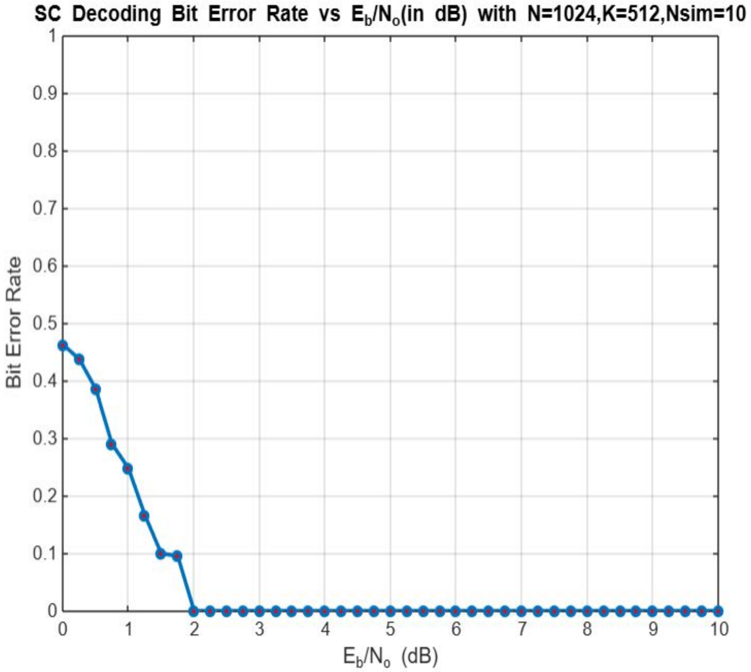


Success Rate for SCL Decoder

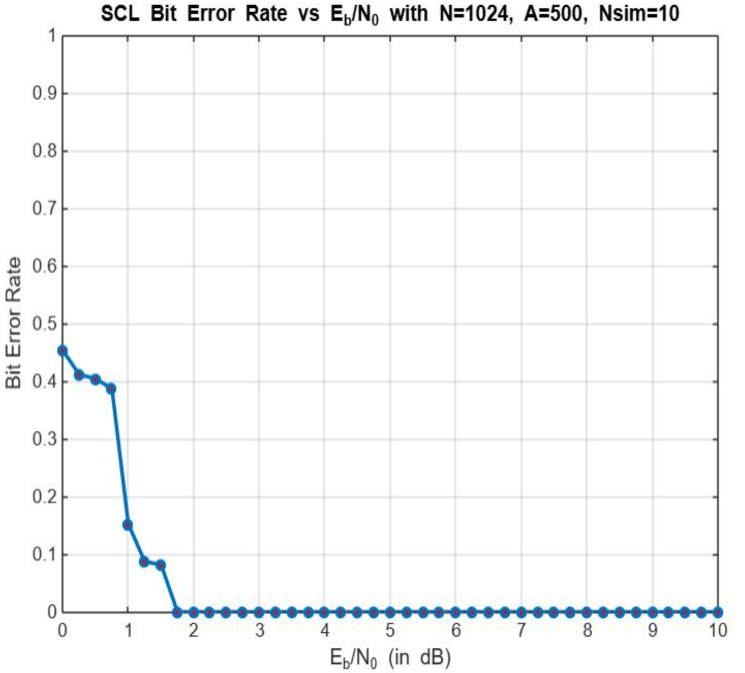


Plots(N=1024, K=512, Nsim = 10)

Bit Error Rate for SC Decoder

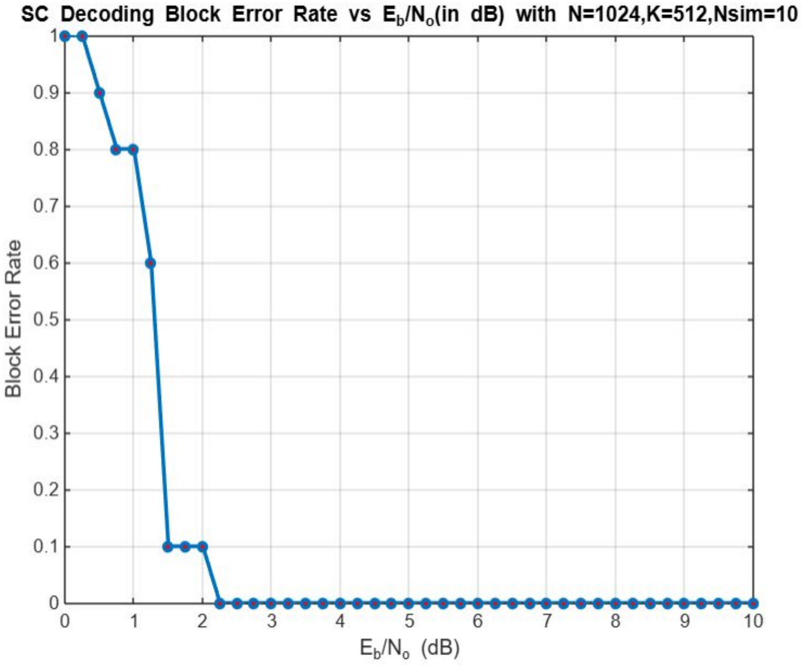


Bit Error Rate for SCL Decoder

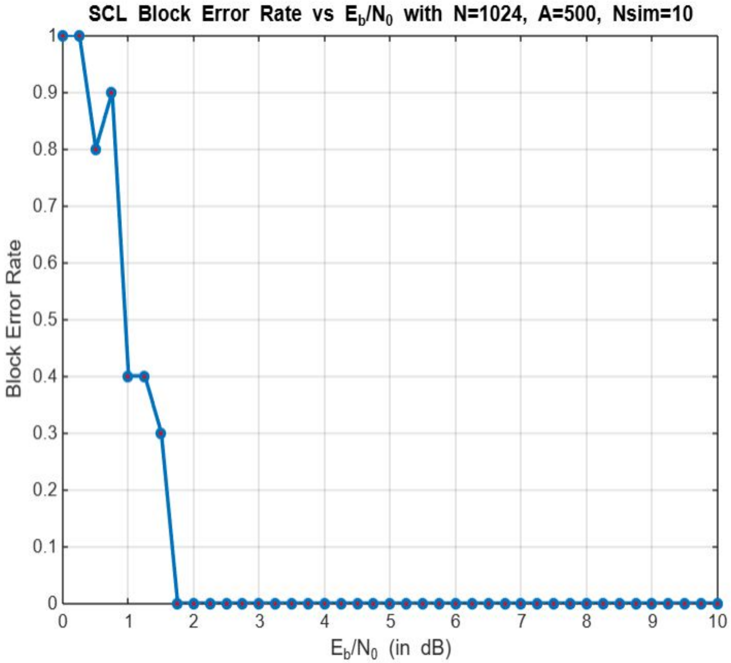


Plots(N=1024, K=512, Nsim = 10)

Block Error Rate for SC Decoder

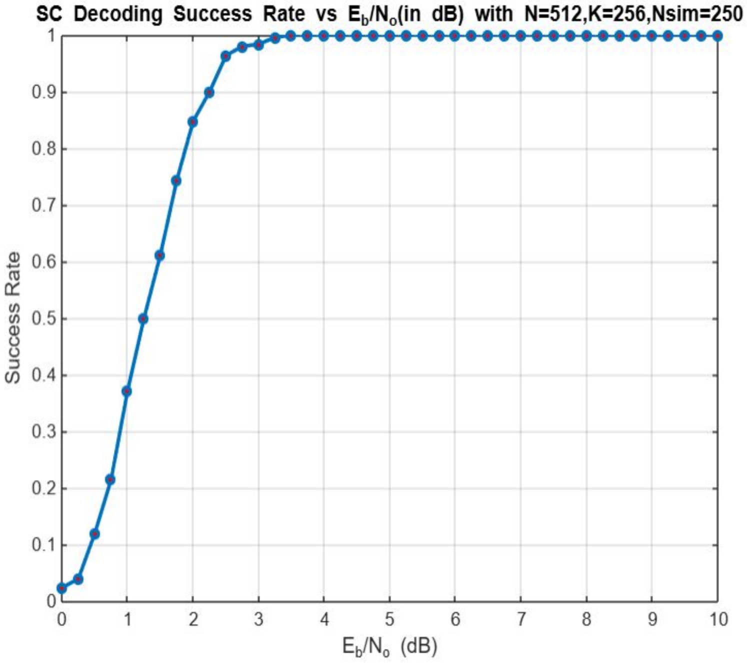


Block Error Rate for SCL Decoder

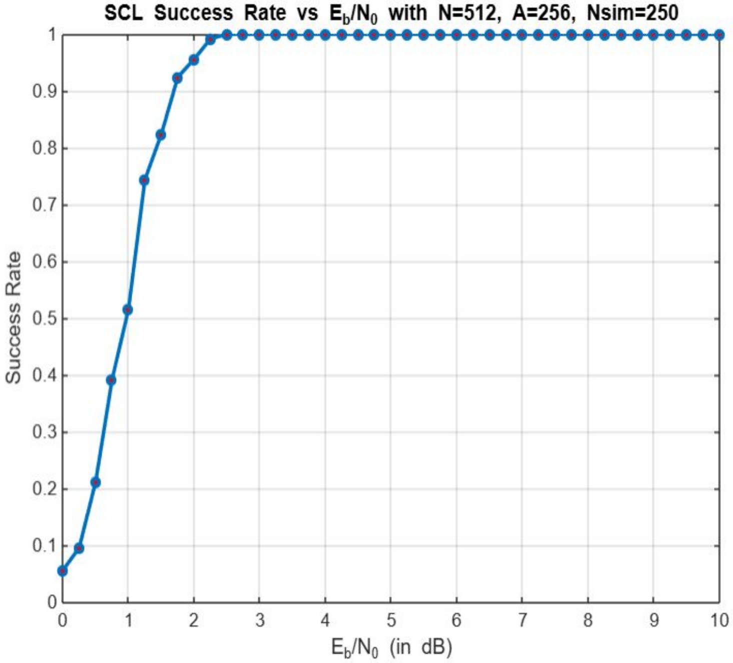


Plots(N=512, K=256, Nsim = 250)

Success Rate for SC Decoder

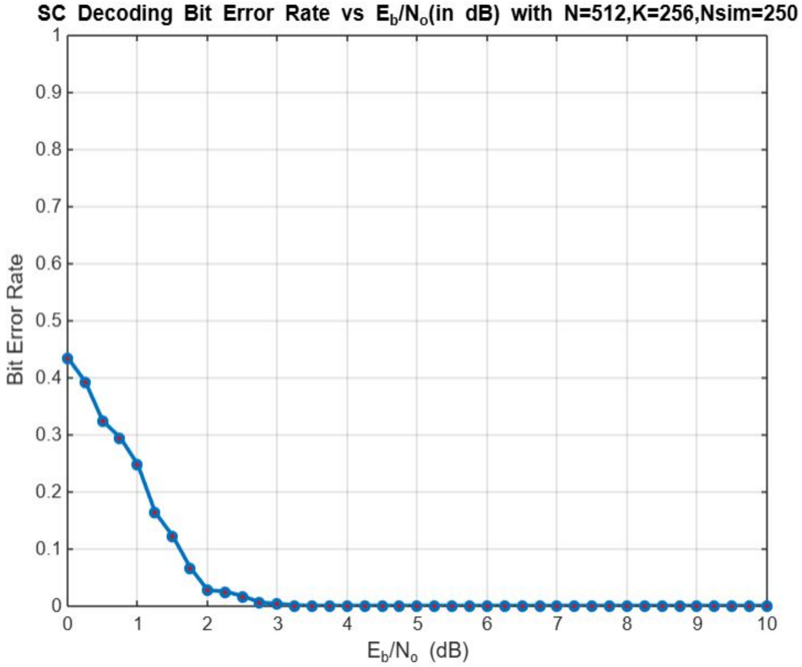


Success Rate for SCL Decoder

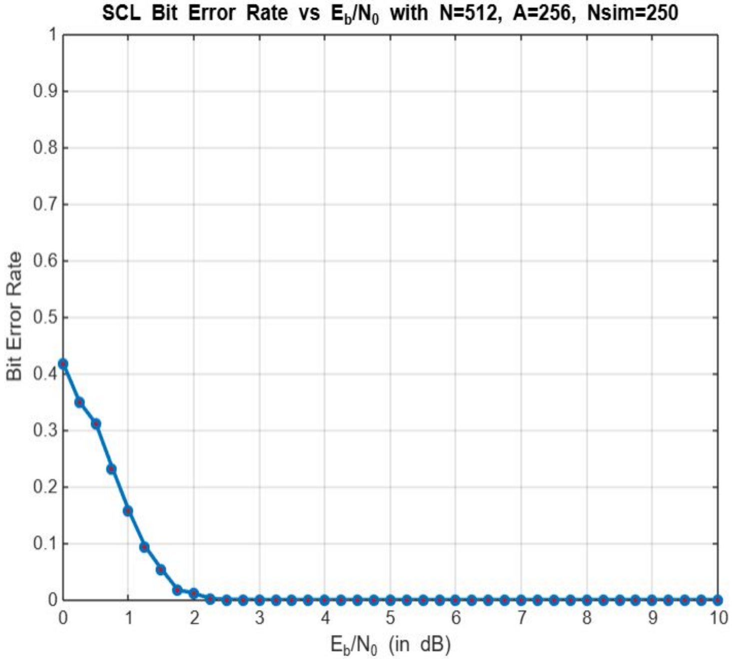


Plots(N=512, K=256, Nsim = 250)

Bit Error Rate for SC Decoder

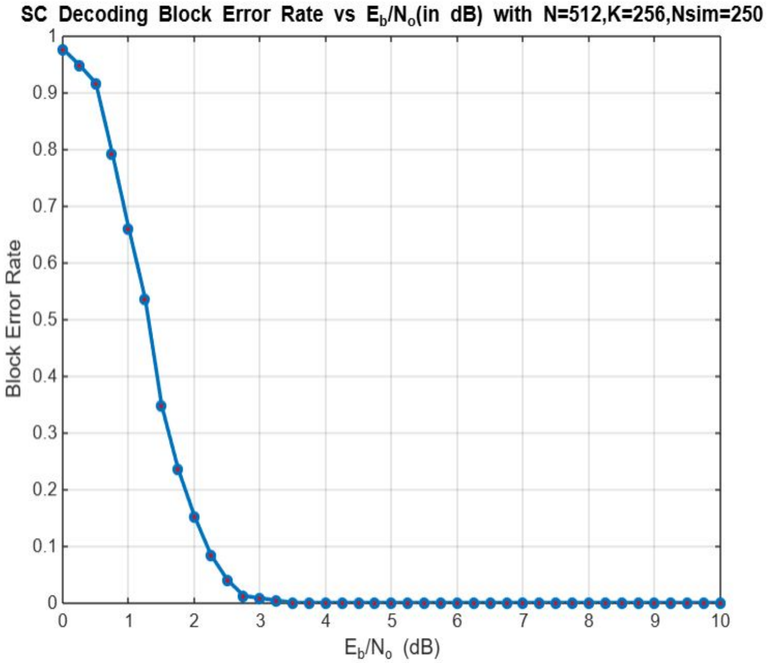


Bit Error Rate for SCL Decoder

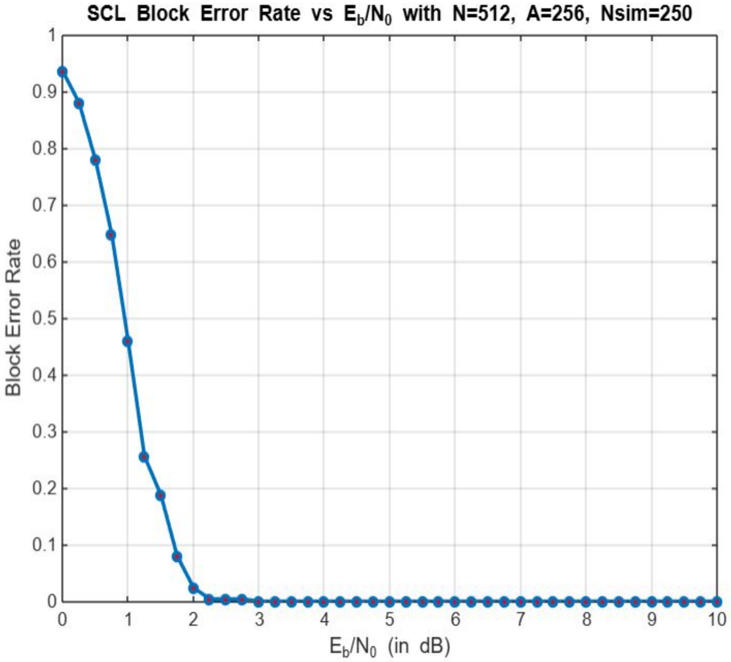


Plots(N=512, K=256, Nsim = 250)

Block Error Rate for SC Decoder

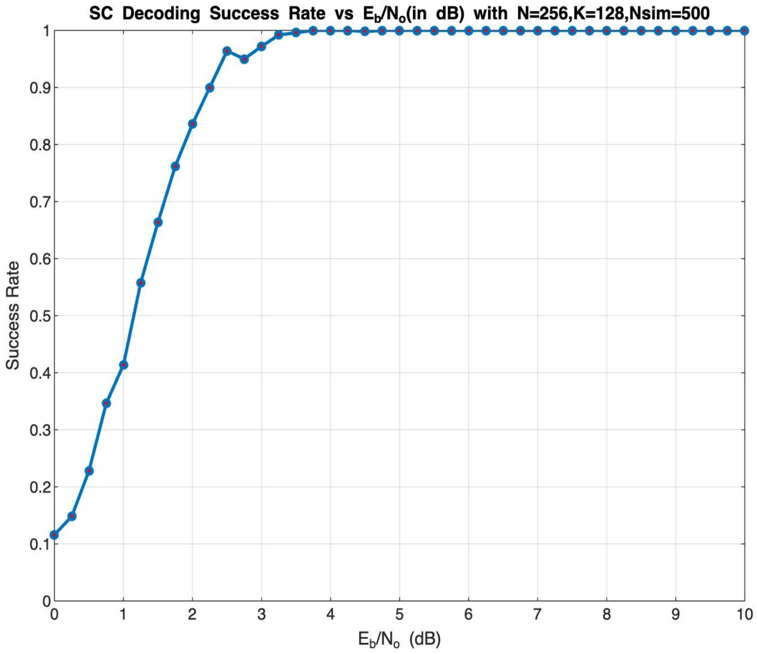


Block Error Rate for SCL Decoder

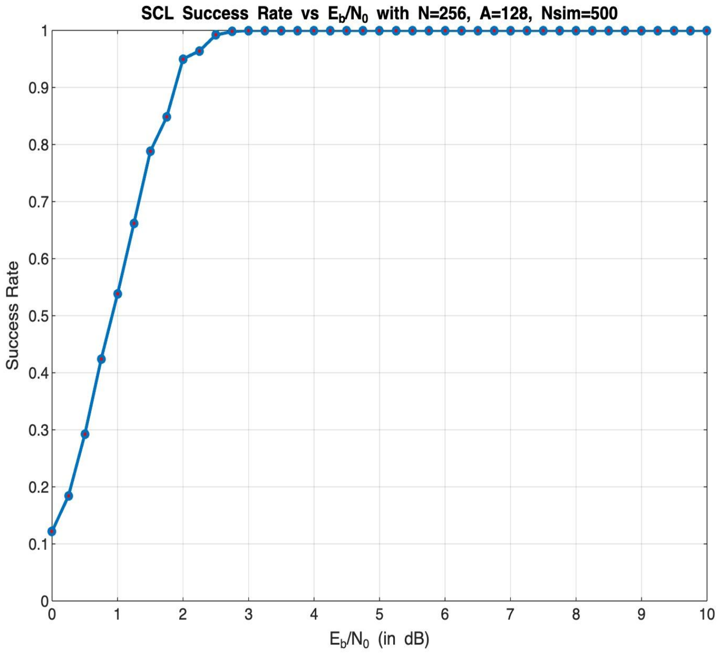


Plots(N=256, K=128, Nsim = 500)

Success Rate for SC Decoder

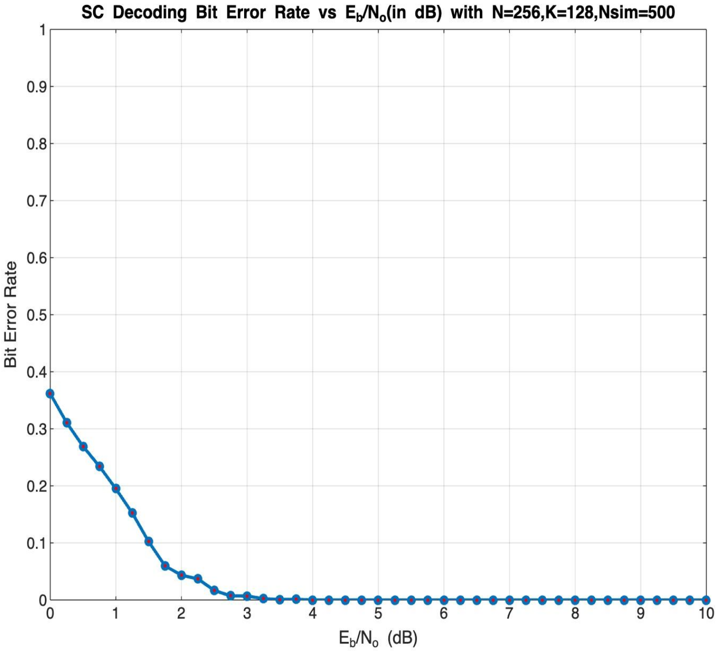


Success Rate for SCL Decoder

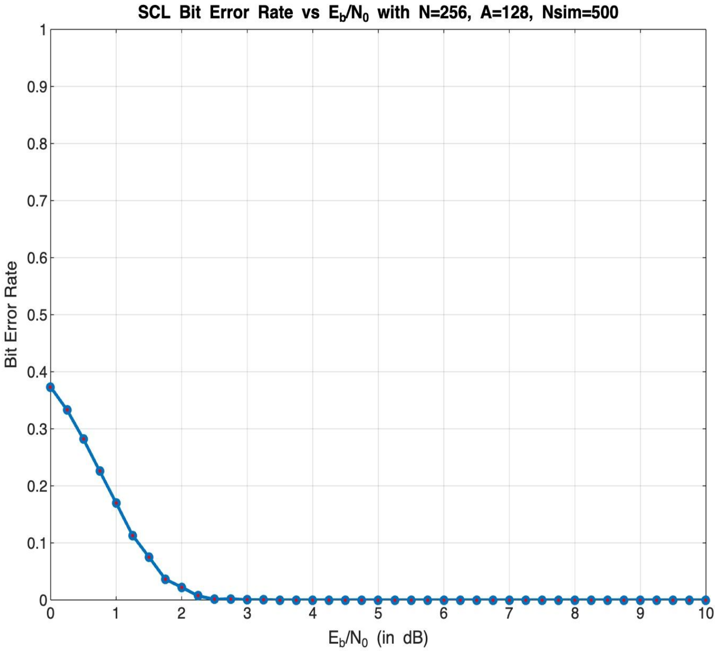


Plots(N=256, K=128, Nsim = 500)

Bit Error Rate for SC Decoder

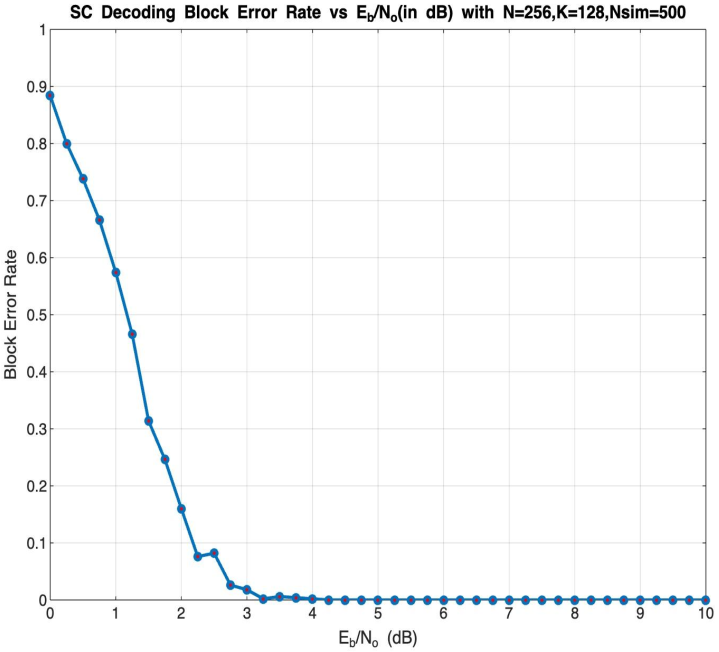


Bit Error Rate for SCL Decoder

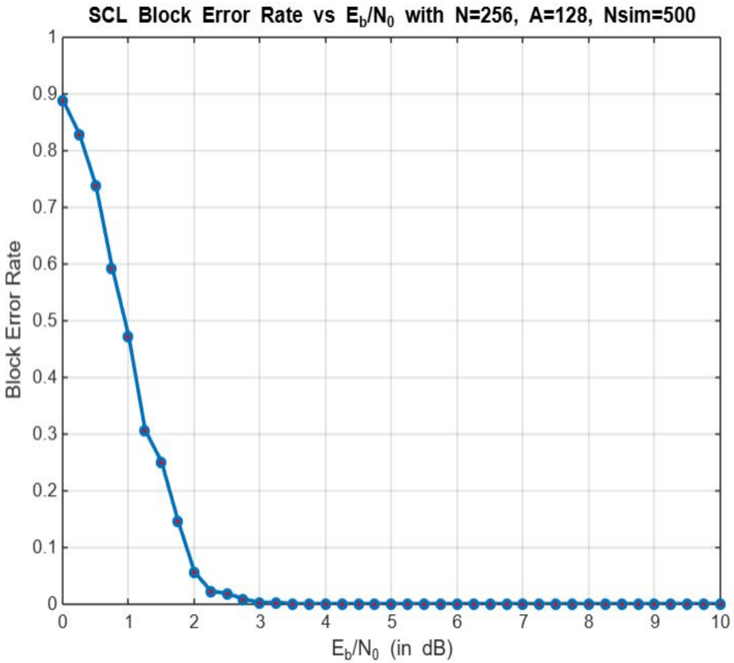


Plots(N=256, K=128, Nsim = 500)

Block Error Rate for SC Decoder

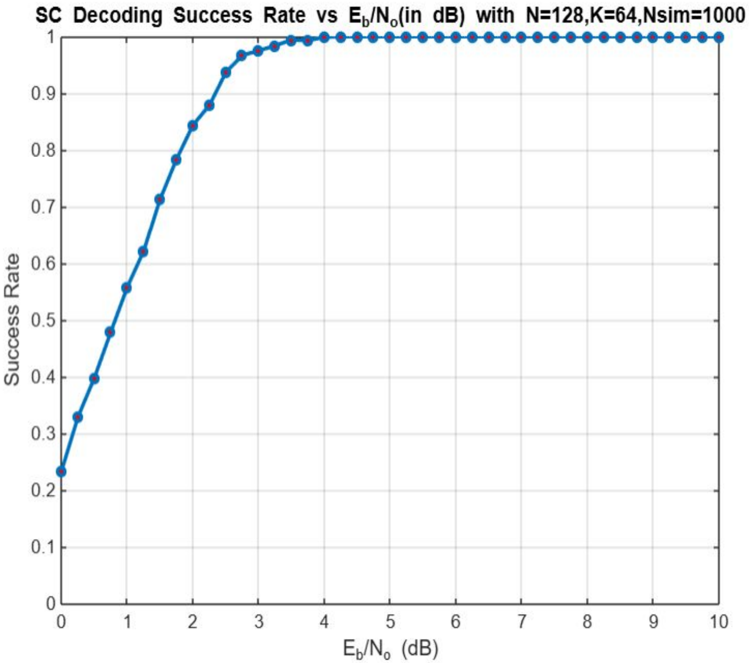


Block Error Rate for SCL Decoder

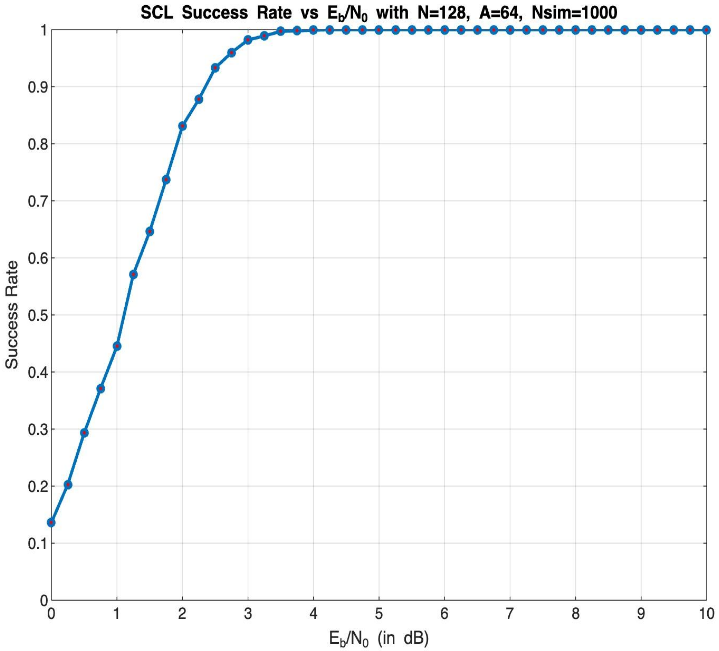


Plots(N=128, K=64, Nsim = 1000)

Success Rate for SC Decoder

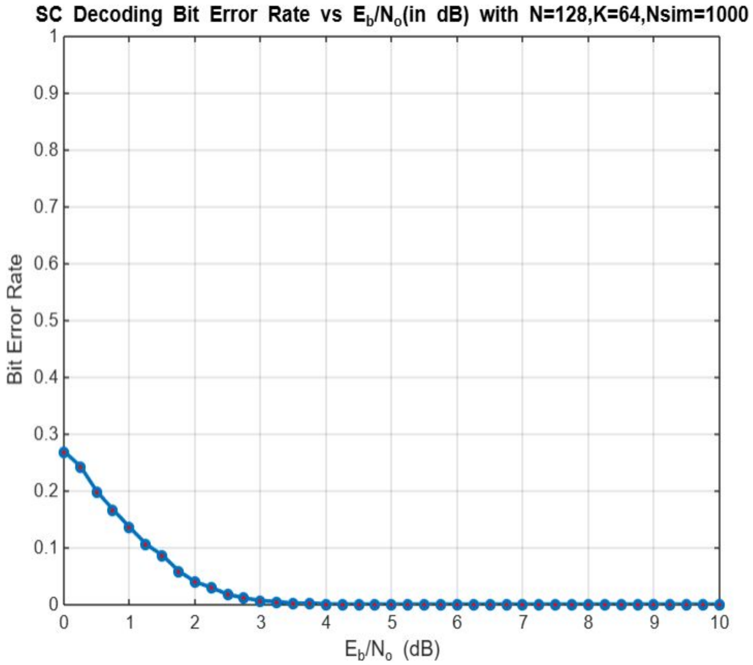


Success Rate for SCL Decoder

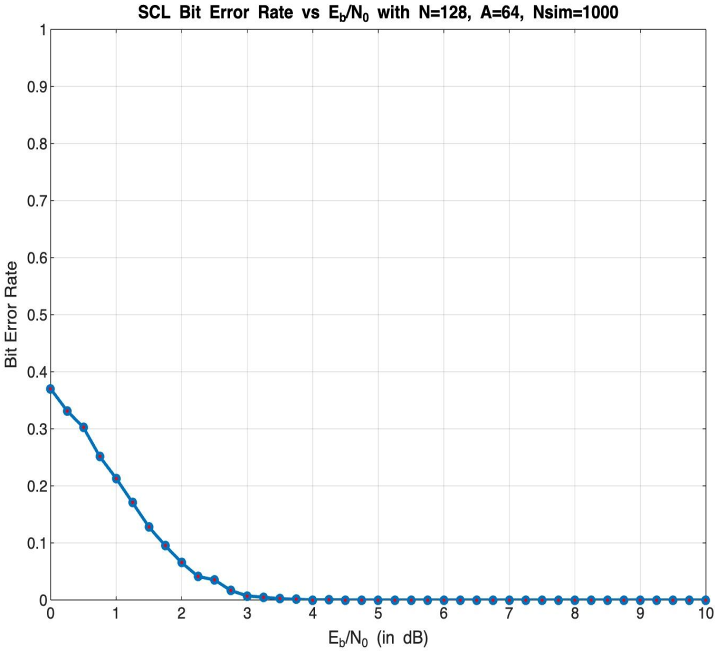


Plots(N=128, K=64, Nsim = 1000)

Bit Error Rate for SC Decoder

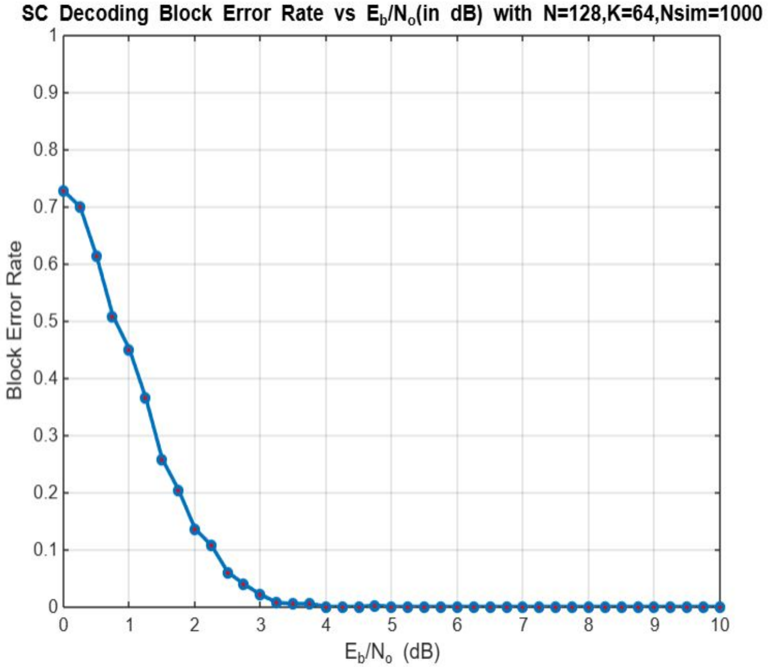


Bit Error Rate for SCL Decoder

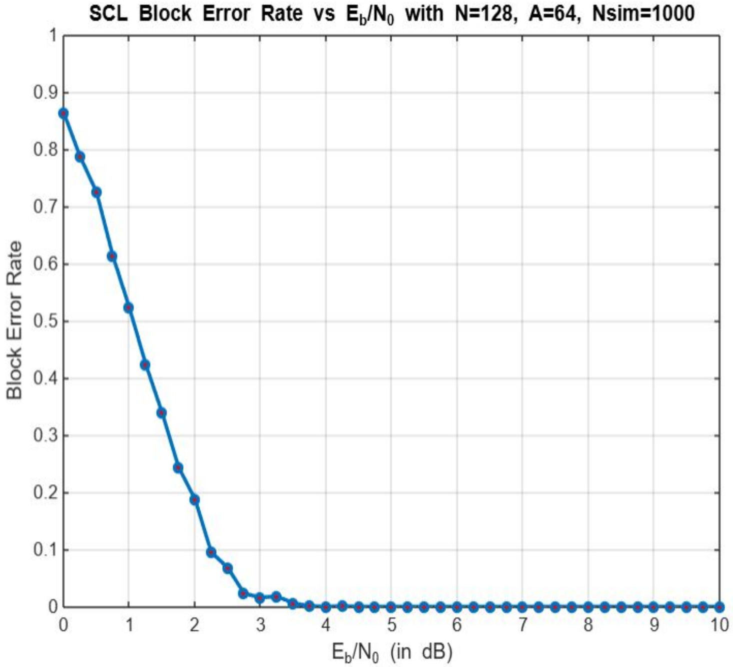


Plots(N=128, K=64, Nsim = 1000)

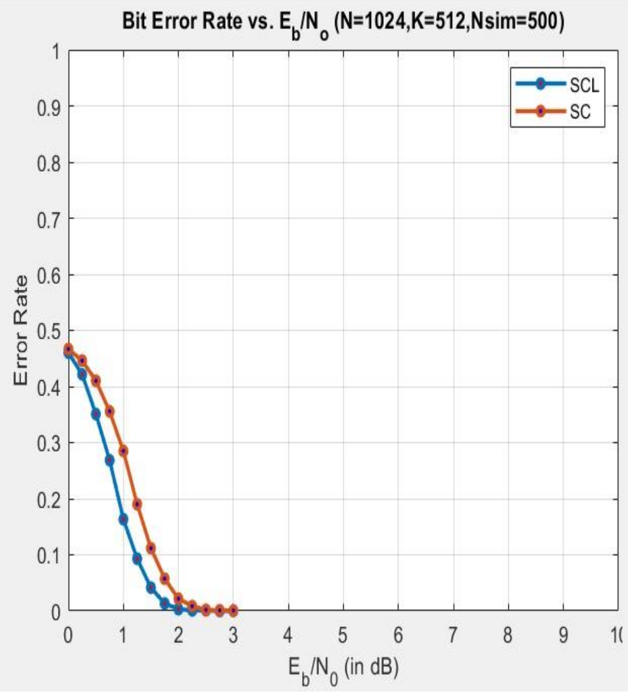
Block Error Rate for SC Decoder

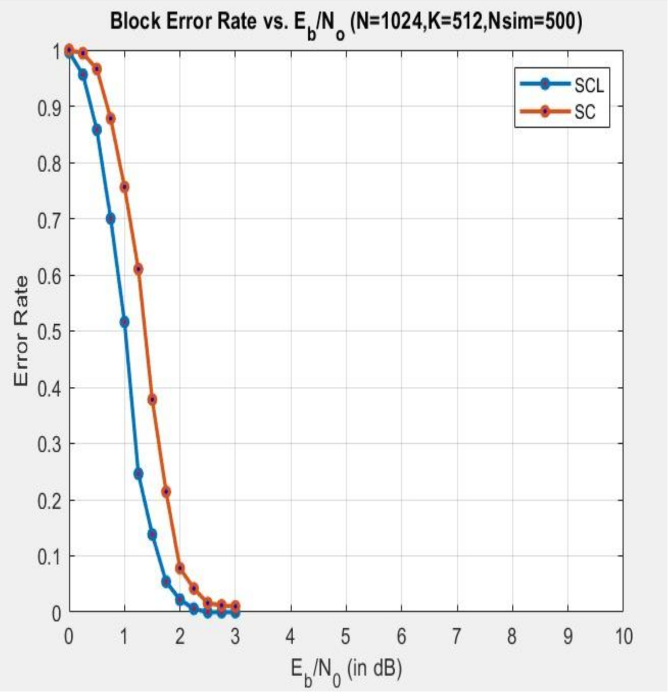


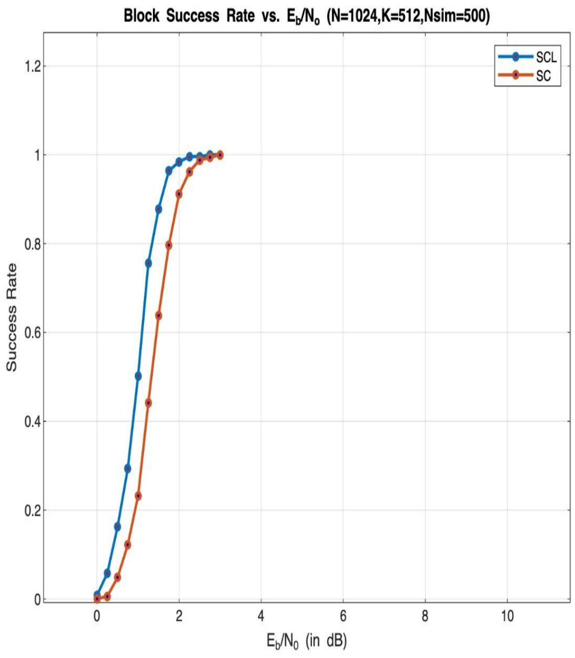
Block Error Rate for SCL Decoder



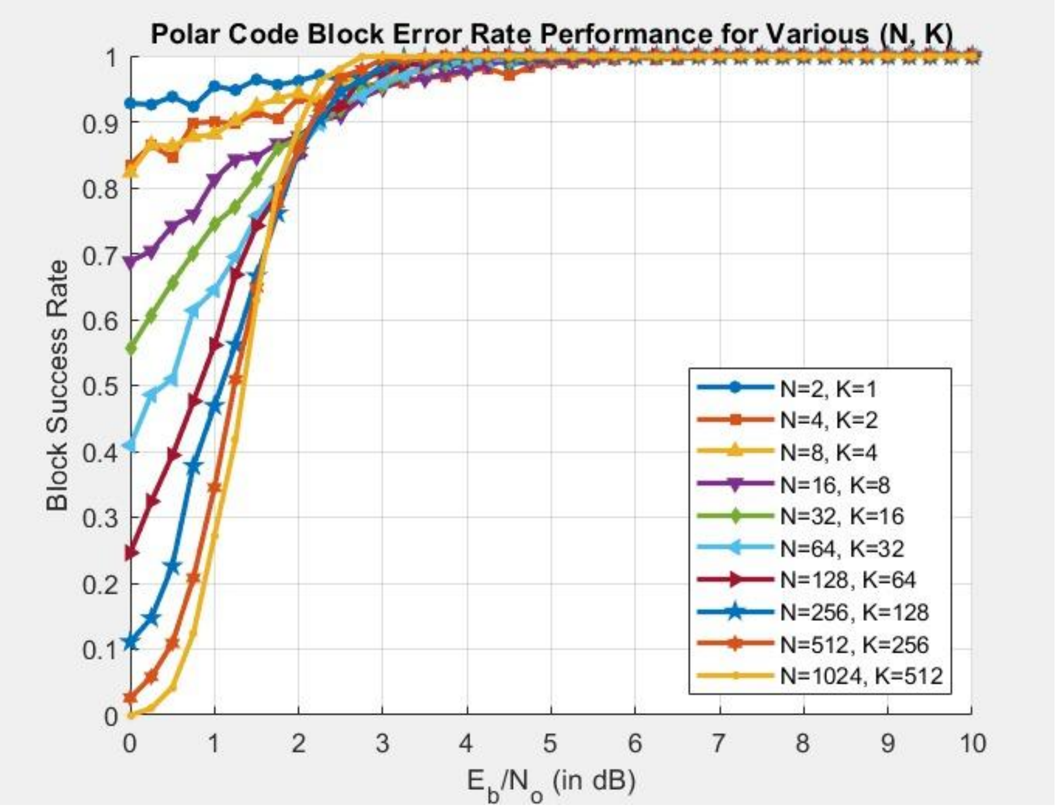
Combined Plots(N=1024, K=512, Nsim=500)



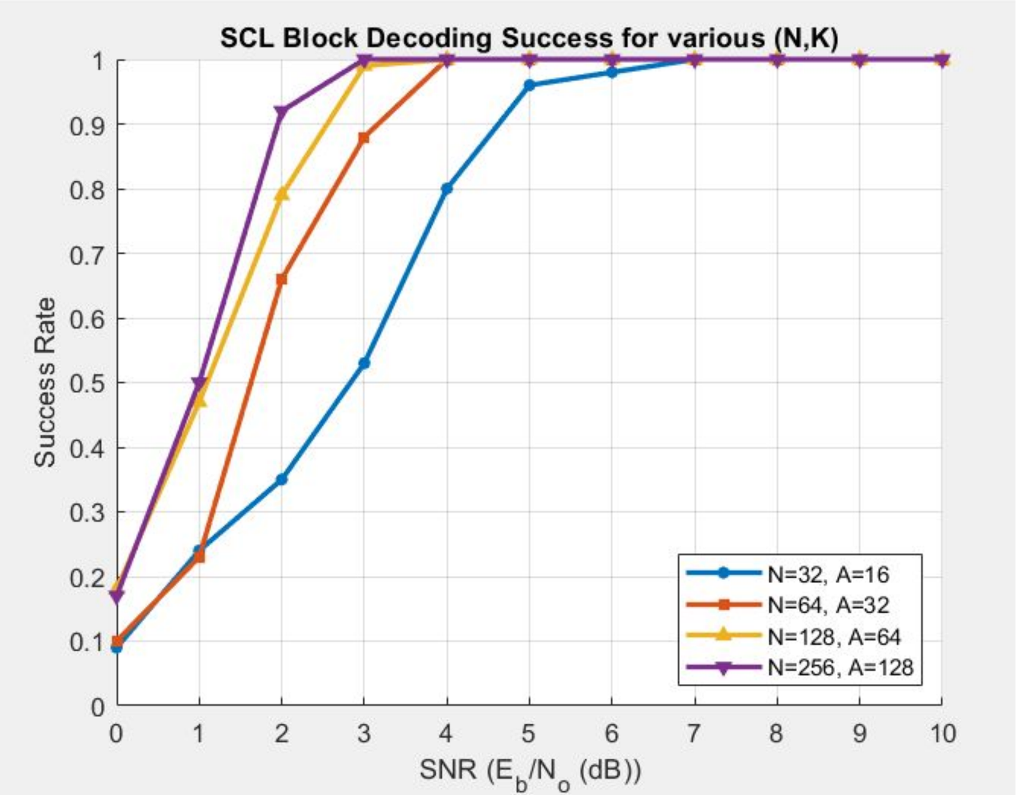




Success Rate for SC Decoding



Success Rate for SCL Decoding using CRC bits



PROOF OF ACHIEVING SHANNON'S CHANNEL CAPACITY BOUND

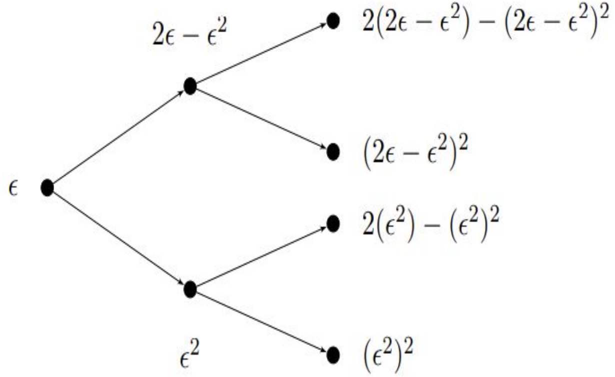
- **Polar codes** achieve channel capacity by assigning information bits to highly reliable synthesized channels and freezing bits on unreliable ones.This method contrasts with traditional codes that aim to maximize Hamming distance. Through **channel polarization**, a physical channel is transformed into virtual channels that become either nearly perfect or completely noisy.

- For a **Binary Erasure Channel (BEC)** with erasure probability ε, polarization produces:

  W⁻: $2ε−ε^2$

  W⁺: $ε^2$

- With each level of recursion, the number of channels increases and their reliability polarizes further—approaching either 0 or 1 error probability, as visualized in the diagram.


ϵ
$2\epsilon - \epsilon^2$
ϵ^2
$2(2\epsilon - \epsilon^2) - (2\epsilon - \epsilon^2)^2$
$(2\epsilon - \epsilon^2)^2$
$2(\epsilon^2) - (\epsilon^2)^2$
$(\epsilon^2)^2$

PROOF OF ACHIEVING SHANNON’S CHANNEL CAPACITY BOUND

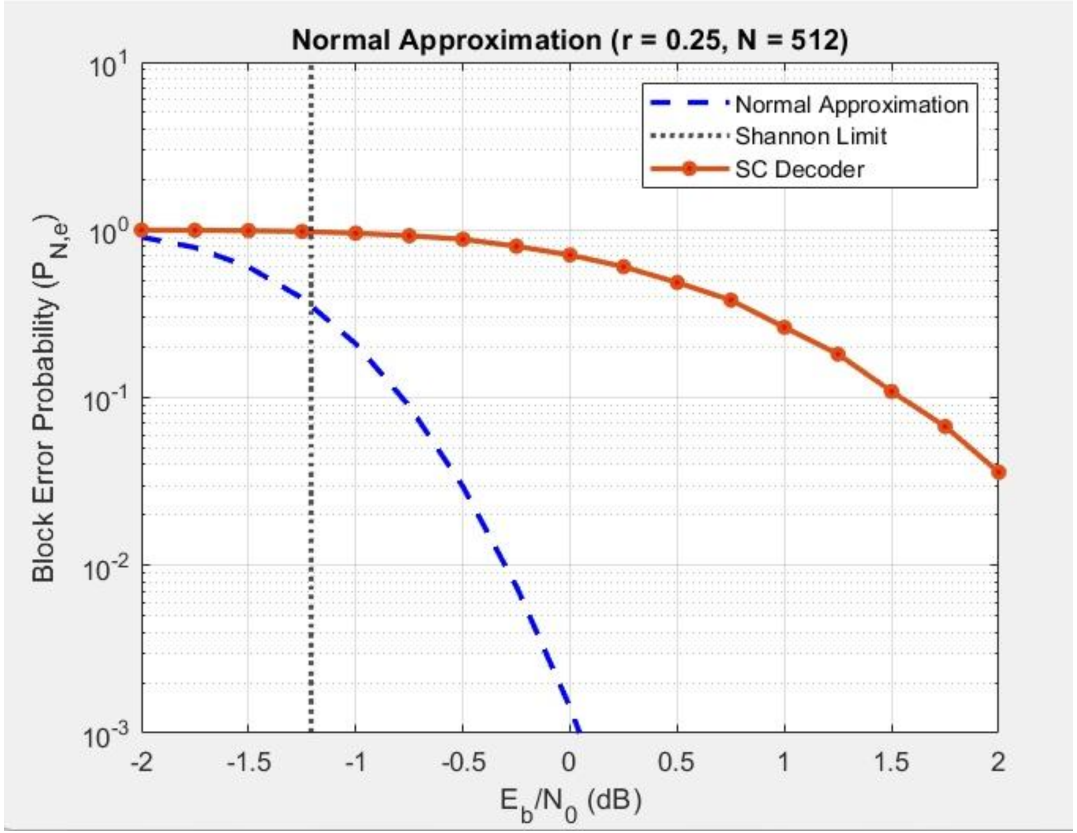
- As the number of synthesized channels N increases, each channel becomes either highly reliable (error probability → 0) or highly unreliable (error probability → 1).
- We transmit message bits through reliable channels and assign fixed (frozen) bits to unreliable ones, ensuring overall reliability.
- For a Binary Erasure Channel (BEC) with erasure probability p, the channel capacity is: $I=1-p$.
- With polarization, the proportion of reliable channels approaches capacity I, while the rest become unusable.

PROOF OF ACHIEVING SHANNON’S CHANNEL CAPACITY BOUND

- As the number of channels increases, the capacities polarize — tending toward either 0 (reliable) or 1 (unreliable), while intermediate channels vanish.

- This polarization effect is supported by **Martingale’s Convergence Theorem**.

- Hence, using polar codes, we can effectively achieve **Shannon’s Channel Capacity Bound**.


Normal Approximation (r = 0.25, N = 512)
Block Error Probability ($P_{N,e}$)
10^1
10^0
10^{-1}
10^{-2}
10^{-3}
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
E_b/N_0 (dB)
Normal Approximation
Shannon Limit
SC Decoder

COMPARISON WITH OTHER CODES

	Encoding		Design & Construction	
	Structure	Complexity	Methods	Complexity
Polar	Recursive encoder	O(NlogN) medium	DE	High
			Tal & Vardy	Medium
			GA	Low
Turbo	Convolutional encoder	O(mN) low	Interleaver optimization	High
LDPC	Matrix multiplication	O(N^2)	Degree distribution optimization	High

Decoding			
	Algorithm	Complexity	Performance
Polar	SC	O(NlogN) low	Suboptimal
	SCL	O(LNlogN) medium	Approach ML
	BP	$O(I_{max}N\log N)$ high	Suboptimal
	CA-SCL	O(LNlogN) medium	Outperform ML
Turbo	Iterative BCJR	$O(I_{max}(4N2^m))$ high	Approach ML
LDPC	BP	$O(I_{max}(N\bar{d}_v+M\bar{d}_c))$ high	Approach ML

REFERENCE

- EventHelix. (2019, April 22). Polar codes: Develop an intuitive understanding of polar codes. 5G NR. https://medium.com/5g-nr/polar-codes-703336e9f26b

- Telatar, E. (2017, June 29). The flesh of polar codes [Conference presentation]. 2017 IEEE International Symposium on Information Theory (ISIT), Aachen, Germany. https://www.youtube.com/watch?v=VhyoZSB9g0w

- Thangraj, A. (2019). Introduction to polar codes: Polar transform [Video lecture]. https://www.youtube.com/watch?v=rB0rhQKyV34&t=81s

- E. Arikan, "Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels," in IEEE Transactions on Information Theory, vol. 55, no. 7, pp. 3051-3073, July 2009, doi: 10.1109/TIT.2009.2021379.

- J. Xiong and L. Zhang, "Simplified Calculation of Bhattacharyya Parameters in Polar Codes," 2020 IEEE 14th International Conference on Anti-counterfeiting, Security, and Identification (ASID), Xiamen, China, 2020, pp. 169-173, doi: 10.1109/ASID50160.2020.9271700. keywords: {Polar codes;Reliability;Encoding;Complexity theory;Channel capacity;Memoryless systems;Error probability;Reliable channel;Bhattacharyya parameter;binary erasure channel},

THANK YOU

“Just like polar codes, ignore all the noise and focus only on the good — and your capacity to enjoy life will be maximized”

-Group2

202301014	DAKSH UBHADIA
202301015	VRUND KANSARA
202301016	TIRTH GANDHI
202301017	JEET DAIYA
202301018	TIRTH KORADIYA
202301019	OM SANTOKI
202301021	PAL KANERIA
202301022	PARTH BHATT
202301023	TIRTH PATEL