Dhirubhai Ambani University

Introduction to Communication Systems (CT216)

Polar Codes

Group 2(1)

Mentor: Aditya Iyer
Professor: Yash Vasavada

6th May 2025

Honor Code:

- We declare that the work presented is our own work.
- We have not copied the work that someone else has done.
- Concepts, understanding, and insights we will be describing are our own.
- Wherever we have relied on existing work that is not our own, we have provided a proper reference citation. We make this pledge truthfully.
- We know that violation of this solemn pledge can carry grave consequences.

Group Numbers:

1. Daksh Ubhadia (ID:202301014)
2. Vrund Kansara (ID: 202301015)
3. Tirth Gandhi (ID: 202301016)
4. Jeet Daiya (ID: 202301017)
5. Tirth Koradiya (ID: 202301018)
6. Om Santoki (ID: 202301019)
7. Manu Pradhan (ID: 202301020)
8. Pal Kaneria (ID: 202301021)
9. Parth Bhatt (ID: 202301022)
10. Tirth Patel (ID: 202301023)

1 Introduction

Polar codes, introduced by **Erdal Arikan**, are the first class of codes to achieve **channel capacity** for **Binary Discrete Memoryless Symmetric Channels (B-DMC)** with low encoding and decoding complexity. The principle behind polar codes is known as **channel polarization**. That is, using channel splitting and grouping, a set of identical channels can be transformed into **highly reliable channels** (where capacity approaches 1) or **highly unreliable channels** (where capacity approaches 0). As the code length increases, this polarization effect becomes more pronounced, resulting in channels that are either very good or very bad.

By transmitting information only through the good channels and fixing the inputs to the bad ones, polar codes effectively exploit this polarization to approach the theoretical limits of channel capacity.

Due to their exceptional performance, polar codes have been selected as the standard for control channels in **5G New Radio (5G-NR)** by the **3rd Generation Partnership Project (3GPP)**.

2 Channel Polarization

Polarization is a phenomenon in channel coding where, by combining and transforming multiple independent copies of a binary-input discrete memoryless channel (B-DMC) W, one can create a new set of synthesized channels $\{W_N^{(i)}\}$. As the number of copies $N = 2^n$ increases, these channels **polarize** — meaning that their capacities tend toward either 0 or 1. Specifically, a fraction $I(W)$ of the channels become nearly perfect (capacity ≈ 1), and the remaining $1 - I(W)$ become nearly useless (capacity ≈ 0).This the reason behind the extremely efficient nature of the polar codes, where data is sent only through the reliable channels, achieving reliable communication rates up to the symmetric capacity $I(W)$ of the original channel.

Polarization includes **(i) Channel Combining** and **Channel Splitting**.

2.1 Channel Combining

Channel Combining is a process by which N independent copies of a discrete memoryless input binary channel (B-DMC) W are recursively combined into a single channel.

The basis step involves taking two consecutive channels and combining them to form a single channel. This is done by following the transformation:

$$(x_1, x_2) \mapsto (x_1 \oplus x_2, x_2)^1$$

This process is performed until all the W_N channels are combined.

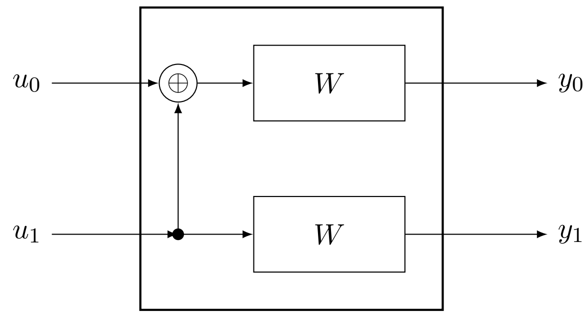

Figure 1: Constructing W_2 channel

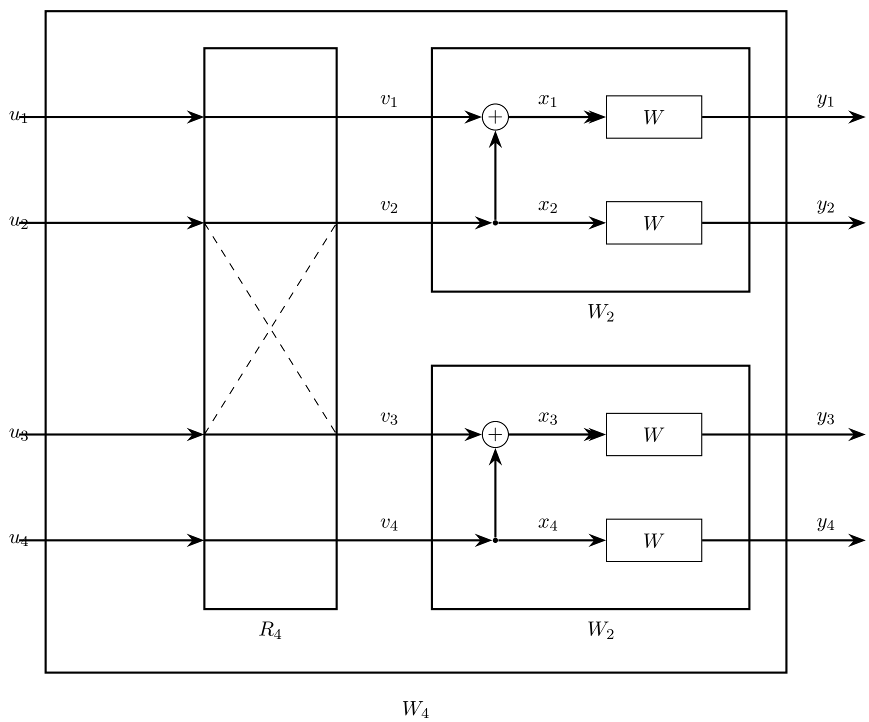

Figure 2: The channel W_4 and its relation to W_2 and W.

[1] If x_1 and x_2 are more than one bit, element-wise XOR should be performed.

2.2 Channel Splitting

After combining, *Channel Splitting* breaks this larger channel into smaller, individual channels. Some of these new channels become very good at sending data (reliable), while others become poor (unreliable).

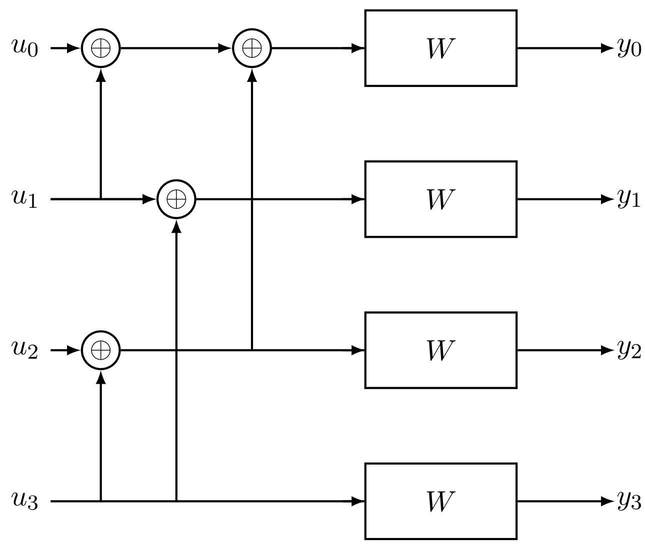

Figure 3: Splitting the W_4 channel

2.3 Example with BEC(p)

A *Binary Erasure Channel (BEC)* is a binary discrete memoryless channel with input alphabet $\chi = \{0, 1\}$ and output alphabet $\Upsilon = \{0, e, 1\}$.The probability of getting an erasure (e) is p while getting the same symbol is $1 - p$

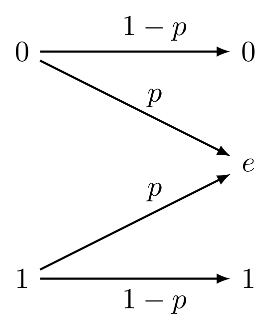

Figure 4: BEC(p) channel

If two bits are transmitted, it means that the channel is used twice. In each case the probability of erasure would be $1 - p$

Now, instead of sending bits through the above channels directly, if the

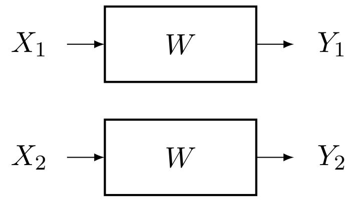

Figure 5: Two independent uses of channel W

channels undergo the process of combining and splitting, two new channels are obtained: W^- and W^+.

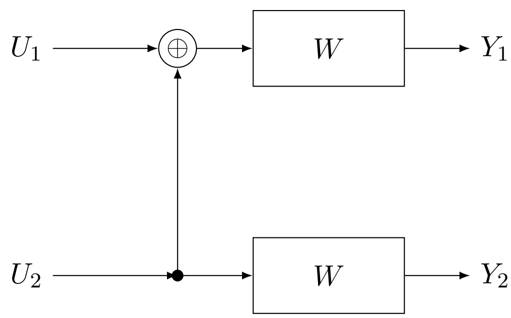

Figure 6: Channel polarization of W. Where $Y_1 = (U_1 \oplus U_2)$ and $Y_2 = (U_2)$.

U_1 is decoded using channel W^- and U_2 is decoded using channel W^+.

Since channel W is used twice to send two different bits, mutual information is additive.

$$2I(W) = I(U_1, U_2; Y_1, Y_2)$$
$$2I(W) = I(U_1; Y_1, Y_2) + (U_2; Y_1, Y_2/U_1) \quad Chain\ Rule$$
$$2I(W) = I(U_1; Y_1, Y_2) + I(U_2; Y_1, Y_2, U_1)$$
$$2I(W) = I(W^-) + I(W^+)$$

Thus, $W^- : (U_1 : Y_1, Y_2)$ and $W^+ : (U_2 : Y_1, Y_2, U_1)$.

2.3.1 W^- channel

W^- tries to decode U_1 using Y_1 and Y_2. For U_1 to be successfully decoded, both Y_1 and Y_2 should be obtained non erased. Else its decoding will fail. The probability of successfully decoding U_1 is $(1-p)^2$ and it probability of decoding failure is $2p - p^2$.

$$W^{+-} = \begin{cases} U_1 \rightarrow (Y_1, Y_2) & \Pr = (1-p)^2 \\ U_1 \rightarrow (?, Y_2) & \Pr = (1-p)p \\ U_1 \rightarrow (Y_1, ?) & \Pr = (1-p)p \\ U_1 \rightarrow (?, ?) & \Pr = p^2 \end{cases}$$

2.3.2 W^+ channel

W+ channel tries to decoding U_2 using Y_1, Y_2 and U_1. It is assumed that U_1 is known at the time of decoding of U_2. For successful decoding of U_2, only Y_1 or Y_2 are needed at the receiver. The probability of successful decoding is $1 - p^2$ and probability of decoding error is (p^2)

$$W^{+-} = \begin{cases} U_2 \rightarrow (Y_1, Y_2, U_1) & \Pr = (1-p)^2 \\ U_2 \rightarrow (?, Y_2, U_1) & \Pr = (1-p)p \\ U_2 \rightarrow (Y_1, ?, U_1) & \Pr = (1-p)p \\ U_2 \rightarrow (?, ?, U_1) & \Pr = p^2 \end{cases}$$

2.3.3 The result

From above two expression we effectively obtain two channels with W^- begin $BEC(2p - p^2)$ and W^+ begin $BEC(p^2)$.Since 0 ¡ p ¡ 1 , channel W^+ is a better channel with less erasure probability compared to W^-. Thus, using this simple technique, a channel can be polarized into good and bad channel.

2.4 Polarization for large number of bits

Since channel combining and splitting are a recursive, the effect of polarization is also recursive.

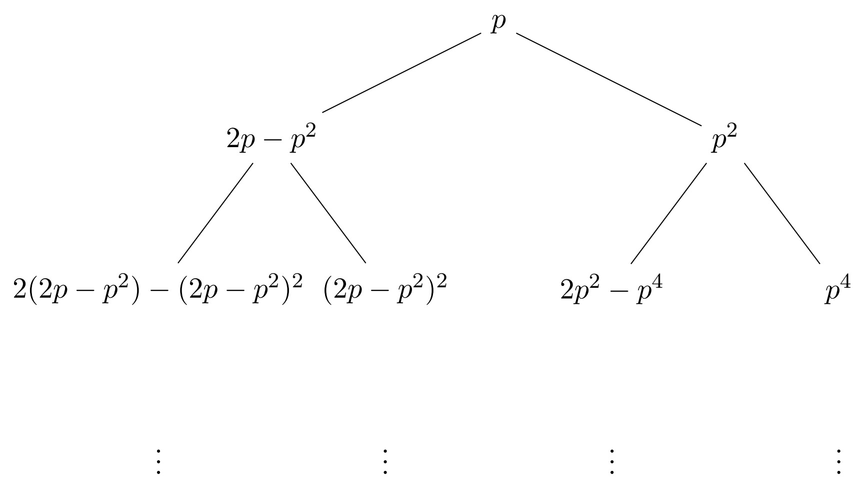

Figure 7: error probability tree of channel polarization

3 Modulation and AWGN Channel

In digital communication systems, modulation and channel modeling are essential components that determine the efficiency and reliability of data transmission. Modulation is the process of converting digital bits into analog waveforms suitable for transmission over physical channels. By mapping bits to signal waveforms, modulation allows better utilization of bandwidth and enables the signal to withstand noise and other impairments encountered during transmission.

Common digital modulation techniques include Binary Phase Shift Keying (BPSK), Quadrature Phase Shift Keying (QPSK), and Quadrature Amplitude Modulation (QAM).

In this section, we present a detailed derivation of the key equations governing Binary Phase Shift Keying (BPSK) modulation over the Additive White Gaussian Noise (AWGN) channel. We derive the noise variance in terms of E_b/N_0.

3.1 BPSK Symbol Mapping

Each information bit $b \in \{0, 1\}$ is mapped to an antipodal signal level:

$$x = \begin{cases} +\sqrt{E_s}, b = 0 \\ -\sqrt{E_s}, b = 1 \end{cases}$$

where E_s is the energy per symbol. Since BPSK carries one bit per symbol, the bit energy E_b equals the symbol energy E_s.

3.2 AWGN Channel and Noise Variance

The received signal on AWGN is

$$y = x + n,$$

where $n \sim \mathcal{N}(0, \sigma^2)$. The spectral density of the two-sided noise power is $N_0/2$, so the variance of n is $\sigma^2 = \frac{N_0}{2}$.

3.3 Decision Rule and BER Integral

A coherent receiver compares y to zero:

$$b = \begin{cases} 0, & y \geq 0 \\ 1, & y < 0 \end{cases}$$

When $b = 0$ (hence $x = +\sqrt{E_b}$) is sent, a bit error occurs if $y < 0$. Thus

$$P_e = P\big(y < 0 \mid x = +\sqrt{E_b}\big) = P\big(n < -\sqrt{E_b}\big).$$

Since $n \sim \mathcal{N}(0, \sigma^2)$, its probability density function (pdf) is

$$p_n(u) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\tfrac{u^2}{2\sigma^2}\Big).$$

Therefore,

$$P_e = \int_{-\infty}^{-\sqrt{E_b}} p_n(u)\, du = \int_{-\infty}^{-\sqrt{E_b}} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\tfrac{u^2}{2\sigma^2}\Big)\, du. \tag{1}$$

3.4 Change of Variable to the Q-Function

Define the standard normal variable

$$t = \frac{u}{\sigma} \quad \Longrightarrow \quad du = \sigma\, dt.$$

When $u = -\sqrt{E_b}$, we have

$$t = \frac{-\sqrt{E_b}}{\sigma} = -\sqrt{\frac{E_b}{\sigma^2}} = -\sqrt{\frac{E_b}{\frac{E_b}{2(E_b/N_0)}}} = -\sqrt{2\,\frac{E_b}{N_0}}.$$

Thus the integral becomes

$$P_e = \int_{-\infty}^{-\sqrt{2\,E_b/N_0}} \frac{1}{\sqrt{2\pi}}\, e^{-t^2/2}\, dt = Q\big(\sqrt{2\,E_b/N_0}\big),$$

where the Q-function is defined by

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-t^2/2}\, dt = \tfrac{1}{2}\,\mathrm{erfc}\Big(\tfrac{x}{\sqrt{2}}\Big).$$

3.5 Final BER Expression

Putting it all together, the bit-error rate for uncoded BPSK over an AWGN channel is

$$\boxed{P_e = Q\big(\sqrt{2\,E_b/N_0}\big).} \tag{2}$$

3.6 Discussion

- The factor $\sqrt{2\,E_b/N_0}$ arises because BPSK symbols are antipodal and each symbol energy equals bit energy.
- The Q-function captures the tail probability of the standard normal distribution.
- All intermediate steps—mapping, noise variance, decision threshold, integral transformation—are now explicitly shown.

This detailed derivation provides a solid foundation for extending to coded systems, where similar integrals appear in the analysis of soft-decision decoding under AWGN.

4 Encoding

Encoding is performed in Polar Codes in a relatively easy manner. It is as simple as matrix multiplication of information/message bits with the generator matrix, which gives us the transmitted codeword. The generator matrix defined as G is shown below.

$$\mathbf{c} = \mathbf{u_{1\times N}}.\mathbf{G_{N\times N}}$$

where $\mathbf{c}$ is the transmitted codeword, $\mathbf{u}$ is the message matrix $1 \times N$ and $\mathbf{G_2}$ is the generator matrix.

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

This generator matrix is valid for the 2-bit message codewords but for lengths greater than 2, let's take 4-bits here and after we will write a general result for N bits, so for 4-bits we need a generator matrix in order to get the codeword that we need to transmit through the channel. So, the codeword can be obtained by Kronecker product of two Generator matrices of $\mathbf{2} \times \mathbf{2}$ dimensions which is being shown below.

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

The generator matrix obtained can be subsequently multiplied by the matrix of message bits suppose let it be $[m_1, m_2, m_3, m_4]$ and then we obtain the encoded codeword in the form of $[m_1+m_2+m_3+m_4, m_2+m_4, m_3+m_4, m_4]$.

Similarly to make things more clear I am also defining $\mathbf{G_8}$ in the same way so follow me.

$$\mathbf{G_8} = \mathbf{G_2} \otimes \mathbf{G_4} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Encoded codeword obtained as result by multiplication with the Generator matrix is given as $[m_1 + m_2 + m_3 + m_4 + m_5 + m_6 + m_7 + m_8, m_2 + m_4 + m_6 + m_8, m_3 + m_4 + m_7 + m_8, m_4 + m_8, m_5 + m_6 + m_7 + m_8, m_6 + m_8, m_7 + m_8, m_8]$ Similarly, for N message bit codeword we can generator matrix as

$$\mathbf{G_N} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}^{\otimes \mathbf{k}}$$

where k is $log_2(N)$.

4.1 Binary Tree Representation for Encoding

The same result can be obtained using binary tree analysis where we start from the leaf nodes which contains message bits and backtrack up to root node which contain the encoded message bits. Other nodes depict intermediate message generated. Say example of representing the binary tree for N=8 message bit codeword, leaf nodes of the tree will contain message bits $\begin{bmatrix} m_1 & m_2 & m_3 & \dots & m_8 \end{bmatrix}$ and the root node will have the final codeword. The following figure depicts the same.

5 Decoding

5.1 Successive Cancellation Decoding Algorithm

As we have seen in the previous section encoded codeword gets modulated and transmitted to a channel but due to channel noise the original message gets corrupted due to effect to Additive White Gaussian Noise. So, in order to understand the basic successive cancellation decoding let's take basic example

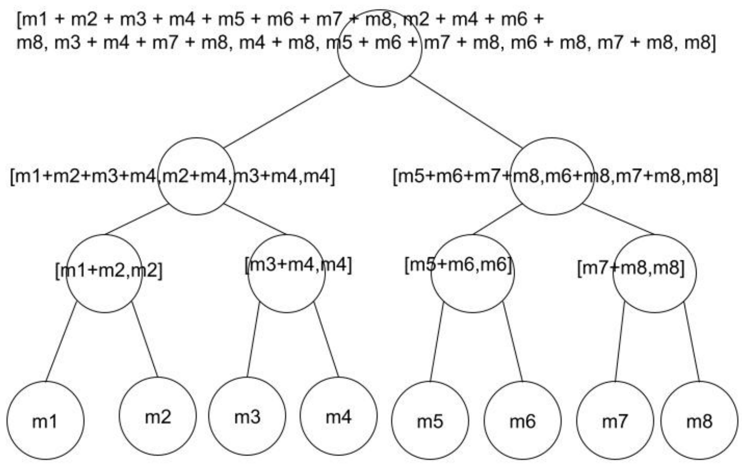

Figure 8: Encoding using Binary tree

for N=2 where received codeword is $\begin{bmatrix}\mathbf{x_1} & \mathbf{x_2}\end{bmatrix} = \begin{bmatrix}\mathbf{u_1} + \mathbf{u_2} & \mathbf{u_2}\end{bmatrix}$, in order to decode the original code we just need to reverse the encoding step, so in order to get the message codeword $\mathbf{u_1} = \mathbf{x_1} + \mathbf{x_2}$ and $\mathbf{u_2} = \mathbf{x_2}$. We have already learnt polarization our first attempt is to decode $\mathbf{u_1}$ using SPC decoding and then using the decoded $\mathbf{u_1}$ we can further decode $\mathbf{u_2}$ which will be done using repetition coding.

$$\mathbf{L(u_1) = sgn(x_1).sgn(x_2).min(|x_1|, |x_2|)}$$

When $\mathbf{L(u_1)} \geq \mathbf{0}$ then $\hat{\mathbf{u}}_\mathbf{1} = \mathbf{0}$ otherwise $\hat{\mathbf{u}}_\mathbf{1} = \mathbf{0}$ when $\mathbf{L(u_1)} < \mathbf{0}$ The proof of this formula which works as an Single Parity Check code is as follows. So, $p(x_1 = 1) = p_1$ and $p(x_2 = 1) = p_2$ and similarly $p(x_1 = 0) = 1 - p_1$ and $p(x_2 = 0) = 1 - p_2$. Likelihood ratio for the odds in favor of 1 can be defined as

$$\lambda_1 = \frac{p(x_1 = 1)}{p(x_1 = 0)} = \frac{p_1}{1 - p_1}$$

Similarly for the log-likelihood ratio is defined as following and the same can be written for x_2 and in the same way by only replacing the probabilities from p_1 to p_2.

$$L_1 = log\left(\frac{p(x_1 = 1)}{p(x_1 = 0)}\right) = log\left(\frac{p_1}{1 - p_1}\right)$$
$$L_2 = log\left(\frac{p(x_2 = 1)}{p(x_2 = 0)}\right) = log\left(\frac{p_2}{1 - p_2}\right)$$

We can employ this information to use this into deriving the likelihood ratio for the case of Single Parity Check code. In Single Parity Check code for two bits we know the following information. We know the value for $\mathbf{x_1}$ is 1 when $\mathbf{u_1 = 0}$ and $\mathbf{u_2 = 1}$ or when $\mathbf{u_1 = 1}$ and $\mathbf{u_2 = 0}$. Both of these can be expressed mathematically that for $\mathbf{x_1 = 1}$ we need either of two cases.

$$p(u_1 = 1) = p_1(1 - p_2) + p_2(1 - p_1)$$

$$p(u_1 = 0) = (1 - p_1)(1 - p_2) + p_1 p_2$$

Likelihood ratio for the following can be written as following

$$\lambda_1 = \frac{p(u_1 = 1)}{p(u_1 = 0)} = \frac{p_1(1 - p_2) + (1 - p_1)p_2}{(1 - p_1)(1 - p_2) + p_1 p_2}$$

In order to get the exact formula we need to follow few steps

$$\Rightarrow p_{u_1} - (1 - p_{u_1}) = p_1(1 - p_2) + p_2(1 - p_1) - (1 - p_1)(1 - p_2) - p_1 p_2$$

$$\Rightarrow p_{u_1} - (1 - p_{u_1}) = (p_1 - (1 - p_1))(p_2 - (1 - p_2))$$

$$\Rightarrow p_{u_1} + (1 - p_{u_1}) = (p_1 + (1 - p_1))(p_2 + (1 - p_2))$$

Dividing the expression will lead to a new expression which together can lead to obtain L_{u_1}

$$\Rightarrow \frac{p_{u_1} - (1 - p_{u_1})}{p_{u_1} + (1 - p_{u_1})} = \frac{(p_1 - (1 - p_1))(p_2 - (1 - p_2))}{(p_1 + (1 - p_1))(p_2 + (1 - p_2))}$$

Dividing p_{u_1} on the right side and p_1 and p_2 on the left side both in numerator and denominator to simplify the expression.

$$\Rightarrow \frac{1 - \frac{(1-p_{u_1})}{p_{u_1}}}{1 + \frac{(1-p_{u_1})}{p_{u_1}}} = \left(\frac{1 - \frac{(1-p_1)}{p_1}}{1 + \frac{(1-p_1)}{p_1}}\right)\left(\frac{1 - \frac{(1-p_2)}{p_2}}{1 + \frac{(1-p_2)}{p_2}}\right)$$

The above can be written in a similar form

$$\Rightarrow \frac{1 - e^{-L_{u_1}}}{1 + e^{-L_{u_1}}} = \left(\frac{1 - e^{-L_1}}{1 + e^{-L_1}}\right)\left(\frac{1 - e^{-L_2}}{1 + e^{-L_2}}\right)$$

The above expression can be converted to the trigonometric form

$$\Rightarrow tanh\left(\frac{L_{u_1}}{2}\right) = tanh\left(\frac{L_1}{2}\right) tanh\left(\frac{L_2}{2}\right)$$

Further simplication of the expression can yield the value for L_{u_1} as following

$$\Rightarrow L_{u_1} = 2tanh^{-1}\left(tanh\left(\frac{L_1}{2}\right)tanh\left(\frac{L_2}{2}\right)\right)$$

Above function for L_{u_1} can be approximated to the equation stated above

$$\Rightarrow L_{u_1} = sgn(L_1).sgn(L_2).min(|L_1|,|L_2|)$$

The result of L_{u_1} can be further used to estimate the value of $\hat{u}_1$ which is stated as follows

$$\hat{u}_1 = \begin{cases} 0, & L(u_1) \geq 0 \\ 1, & L(u_1) < 0 \end{cases}$$

The above function will evaluate the belief for u_1 and that result can be used to evaluate further to obtain u_2 which will based on repetition coding. For repetition coding we can write u_2 as functions of probabilities for x_1 and x_2

$$p(u_2 = 1) = p_1 p_2$$

And we can write the the same expression for $p(u_2 = 0)$

$$p(u_2 = 0) = (1 - p_1)(1 - p_2)$$

$$\Rightarrow \frac{p_{u_2}}{1 - p_{u_2}} = \left(\frac{p_1}{1 - p_1}\right)\left(\frac{p_2}{1 - p_2}\right)$$

Taking the logarithm on both sides and this will become log-likelihood for $\hat{u}_2 = 1$

$$\Rightarrow L(u_2 = 1) = L_2 - L_1$$

We can derive log-likelihood function for $\hat{u}_2 = 0$

$$\Rightarrow L(u_2) = L_1 + L_2$$

We can generalize the expression for $\hat{u}_2$ as

$$\Rightarrow L(u_2 = 0) = L_2 + (-1)^{u_0} L(u_1)$$

Now, we can generalize the expression for deriving $\hat{u_2}$

$$\hat{u_2} = \begin{cases} 0, & L(u_2) \geq 0 \\ 1, & L(u_2) < 0 \end{cases}$$

The derived expression can be used to find out the original transmitted codeword given the values of $\hat{u}_1$ and $\hat{u}_2$ the original codeword is $\begin{bmatrix}\hat{u}_1 & \hat{u}_2\end{bmatrix}$.

The algorithm is similar to traversal of a tree, first traversal to the left node till it has reached the leaf node, if the leaf node is frozen then simply zero is returned without any further evaluation if the node is not frozen then it is estimated using LLR(also known as minsum method) and after estimation it goes towards right node traversal which further determines its $\hat{u}_2$ value it can be vector too, as we have values both $\hat{u}_1$ and $\hat{u}_2$ we can get the estimated codeword of the parent node which is subsequently returned to its parent till the root node is not reached. The root node contains the final estimated message. This all can be understood with the help of schematic diagram displayed below.

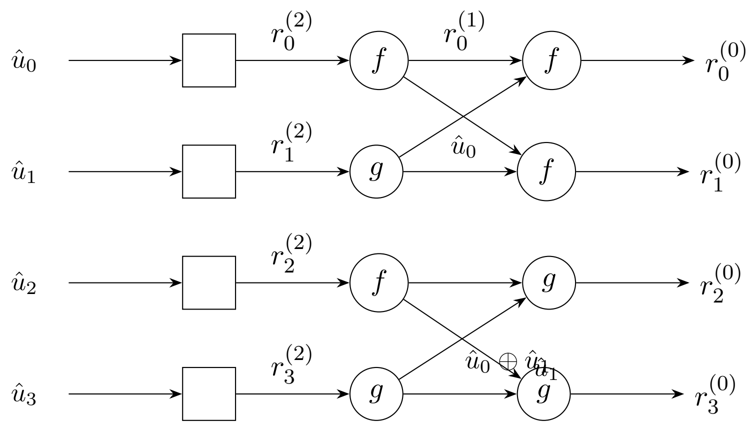

Figure 9: SC decoder

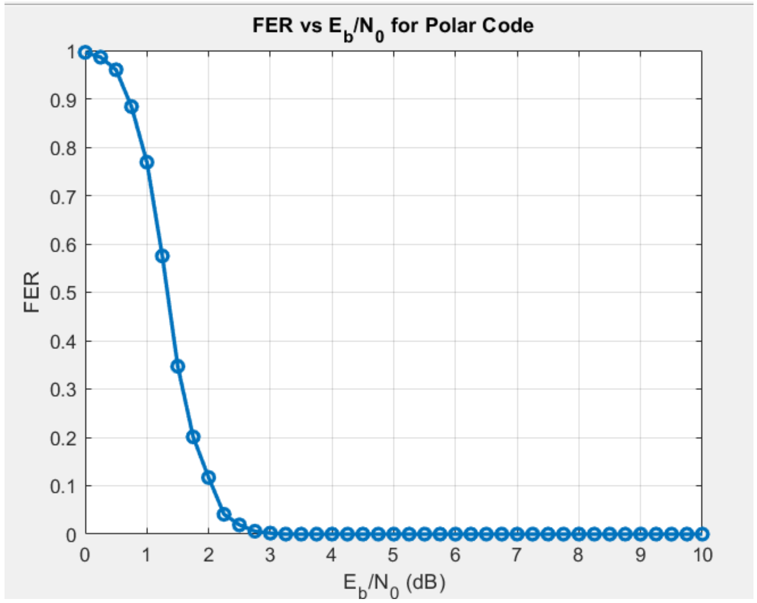

Figure 10: Block Error Rate in SC Decoding

5.2 Successive Cancellation List Decoder

In successive cancellation list decoding we have seen that we made a decision based on value of $L(u_1)$. If $L(u_1) \geq 0$ then we made decision that $\hat{u}_1 = 0$ or $\hat{u}_1 = 1$. Put there are still chances that $\hat{u}_1 = 1$ when $L(u_1) \geq 0$. So, in order to address this particular issue we use successive cancellation list decoding. This will help us to improve the error probability by approximately 1 dB than successive cancellation decoding. So, now we take chances of $\hat{u}_1 = 1$ when $L(u_1) \geq 0$ so when we consider both the probabilites then we add a penalty namely path metric which we impose to the choice when opting $u_1 = 1$ when $L(u_1) \geq 0$ or select $u_1 = 0$ when $L(u_1) < 0$ but taking these choices impose penalty of $|L(u_1)|$. After iterating all the paths we calculate total penalty require to select a choice and select that path which has least penalty and CRCs(Cyclic Redundancy Check) will select the one that is most likely message from the list of candidate codewords. It will make the code more efficient. We can compare efficiency of SC and SCL by the following figures.

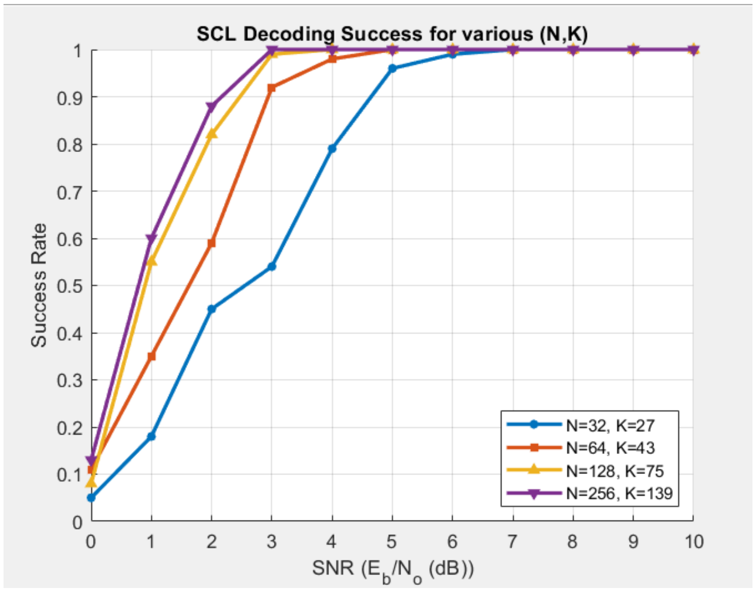

Figure 11: Comparsion between different code lengths using SCL Decoding

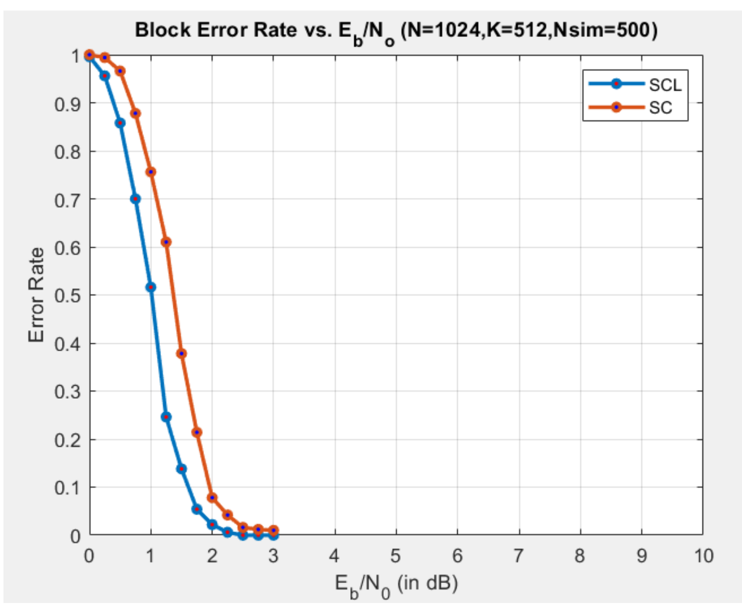

Figure 12: Comparison between SC and SCL decoding

6 Construction of Polar codes via Gaussian Approximation for LLR distribution

6.1 Introduction

Polar codes represent a significant advancement in coding theory, achieving channel capacity for symmetric binary-input discrete memoryless channels. These codes employ channel polarization to transform N identical channel instances into extremal channels - either perfect noiseless channels or completely noisy channels. The construction process utilizes Gaussian approximation of log-likelihood ratio (LLR) distributions to estimate channel reliabilities, enabling optimal information bit placement.

6.2 LLR Distribution for AWGN Channels

For a binary-input additive white Gaussian noise (AWGN) channel, the received signal y is given by:

$$y = x + n$$

where:

- $x \in \{-1, +1\}$ represents the BPSK-modulated transmitted symbol
- $n \sim \mathcal{N}(0, \sigma^2)$ denotes the AWGN component

The log-likelihood ratio (LLR) provides a soft decision metric:

$$\text{LLR} = \log\left(\frac{P(y|x=1)}{P(y|x=-1)}\right) = \frac{2y}{\sigma^2}$$

[LLR Distribution] For the AWGN channel with BPSK modulation, the LLR follows a Gaussian distribution:

$$L(y)|x \sim \mathcal{N}\left(\frac{2x}{\sigma^2}, \frac{4}{\sigma^2}\right) \tag{3}$$

The conditional distributions are:

$$P(y|x=+1) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y-1)^2}{2\sigma^2}\right)$$
$$P(y|x=-1) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y+1)^2}{2\sigma^2}\right)$$

The LLR becomes:

$$
\begin{aligned}
L(y) &= \log\left(\frac{P(y|x=+1)}{P(y|x=-1)}\right) \\
&= \frac{1}{2\sigma^2}\left[-(y-1)^2 + (y+1)^2\right] \\
&= \frac{2y}{\sigma^2}
\end{aligned}
$$

Since y is Gaussian, $L(y)$ is also Gaussian with:

$$
\begin{aligned}
E[L(y)|x] &= \frac{2E[y|x]}{\sigma^2} = \frac{2x}{\sigma^2} \\
\mathrm{Var}[L(y)|x] &= \left(\frac{2}{\sigma^2}\right)^2 \mathrm{Var}[y|x] = \frac{4}{\sigma^2}
\end{aligned}
$$

6.2.1 Properties of LLR Distribution

The LLR distribution has several important characteristics:

- **Symmetry Property**:

$$
P(L=l|x=+1) = e^l P(L=-l|x=-1) \tag{4}
$$

- **Conditional Distributions**:

$$
L|x=+1 \sim \mathcal{N}(\mu, 2\mu) \tag{5}
$$

$$
L|x=-1 \sim \mathcal{N}(-\mu, 2\mu) \tag{6}
$$

  where $\mu = \frac{2}{\sigma^2}$.

- **Decoding Rule**:

$$
\hat{x} = \begin{cases} +1 & \text{if } L(y) \geq 0 \\ -1 & \text{if } L(y) < 0 \end{cases} \tag{7}
$$

- **Variance Relationship**: The variance of the LLR distribution affects the polarization process, with higher variance leading to more pronounced polarization effects during recursive channel transformations.

6.3 Recursive Mean Calculation for Polarized Channels

The channel polarization transform creates N synthesized channels $W_N^{(i)}$ through recursive application of Arıkan's transform:

$$W_N^{(i)}(y_1^N, u_1^{i-1}|u_i) = \sum_{u_{i+1}^N \in \{0,1\}^{N-i}} \frac{1}{2^{N-1}} W_N(y_1^N | u_1^N G_N) \tag{8}$$

where G_N is the generator matrix and u_1^N are the input bits.

6.3.1 Recursive Mean LLR Relations

The mean LLR for polarized channels follows these recursive relations:

[H] Recursive Mean LLR Calculation [1] Initial mean $m = \frac{2}{\sigma^2}$, code length $N = 2^n$ Array of mean LLRs $m_N^{(1)}, \ldots, m_N^{(N)}$ Initialize $m_1^{(1)} \leftarrow m$ $s \leftarrow 1$ to n $j \leftarrow 1$ to 2^{s-1} $m_{2^s}^{(2j-1)} \leftarrow f^{-1}\left(1 - (1 - f(m_{2^{s-1}}^{(j)}))^2\right)$ $m_{2^s}^{(2j)} \leftarrow 2m_{2^{s-1}}^{(j)}$

The function $f(x)$ and its inverse are approximated as:

$$f(x) \approx \begin{cases} \exp(-0.4527x^{0.86} + 0.0218) & 0 < x < 10 \\ \sqrt{\frac{\pi}{x}}\left(1 - \frac{7}{10x}\right)\exp(-\frac{x}{4}) & x \geq 10 \end{cases} \tag{9}$$

6.3.2 Polarization Dynamics

The recursive relations demonstrate two fundamental polarization effects:

- **Improvement Channel (W^+):** The mean LLR doubles ($m_N^{(2j)} = 2m_{N/2}^{(j)}$), indicating increased reliability
- **Degradation Channel (W^-):** The mean LLR follows a complex non-linear transformation, generally decreasing reliability

The polarization phenomenon becomes more pronounced with each recursion level:

Recursion Level	Best Channel $m_N^{(i)}$	Worst Channel $m_N^{(i)}$
1	$2/\sigma^2$	$2/\sigma^2$
2	$4/\sigma^2$	$f^{-1}(1-(1-f(2/\sigma^2))^2)$
3	$8/\sigma^2$	$f^{-1}(1-(1-f(f^{-1}(1-(1-f(2/\sigma^2))^2))^2)$

6.3.3 Numerical Stability Considerations

For practical implementations:

- Small mean values ($m < 0.01$) are clamped to prevent underflow
- The inverse function f^{-1} is implemented via lookup tables with interpolation
- Recursion depth is limited to prevent excessive computation

6.4 Bhattacharya Parameter and Channel Selection

The Bhattacharya parameter serves as a channel reliability metric:

$$Z(W) = \sum_{y \in \mathcal{Y}} \sqrt{W(y|0)W(y|1)} \tag{10}$$

For the AWGN channel with Gaussian approximation:

$$Z_N^{(i)} \approx \exp\left(-\frac{m_N^{(i)}}{4}\right) \tag{11}$$

For any B-DMC W, the Bhattacharya parameter satisfies:

1. $0 \leq Z(W) \leq 1$
2. $Z(W) = 0$ iff W is perfect
3. $Z(W) = 1$ iff W is completely noisy
4. $P_e(W) \leq Z(W) \leq \sqrt{2P_e(W)}$

6.4.1 Properties of Bhattacharya Parameter

- **Monotonicity**: $Z(W)$ strictly decreases as $m_N^{(i)}$ increases
- **Bounds**: $0 \leq Z(W) \leq 1$, where:
  - $Z(W) \to 0$ indicates a perfect channel
  - $Z(W) \to 1$ indicates a completely noisy channel
- **Additivity**: For independent channels W_1 and W_2:

$$Z(W_1 W_2) \leq Z(W_1) + Z(W_2) - Z(W_1)Z(W_2)$$

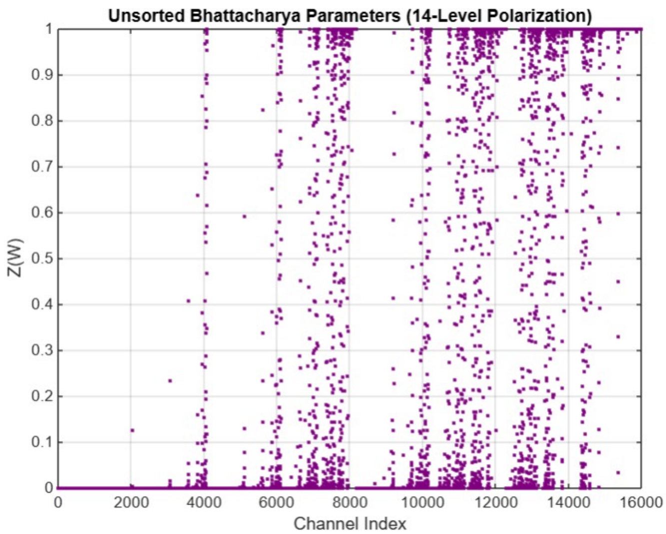

Figure 13: Encoding using Binary tree

6.4.2 Channel Selection Algorithm

The complete channel selection process is formalized in Algorithm 6.4.2.

[t] Polar Code Construction via Gaussian Approximation [1] SNR γ, code length $N = 2^n$, dimension K Information set $\mathcal{A} \subseteq \{1, \ldots, N\}$ Compute initial mean $m \leftarrow 2\gamma$ Initialize $m_1^{(1)} \leftarrow m$ $s \leftarrow 1$ to n Recursive polarization $j \leftarrow 1$ to 2^{s-1} Compute $m_{2^s}^{(2j-1)}$ and $m_{2^s}^{(2j)}$ using (10) $i \leftarrow 1$ to N $Z_N^{(i)} \leftarrow \exp(-m_N^{(i)}/4)$ Sort channels by $Z_N^{(i)}$ in ascending order Select K channels with smallest $Z_N^{(i)}$ as $\mathcal{A}$

6.4.3 Optimization Techniques

Practical implementations often employ:

- **Early Termination**: Stop recursion when $Z_N^{(i)} < \epsilon$ or $Z_N^{(i)} > 1 - \epsilon$
- **Parallel Computation**: Calculate different recursion paths independently
- **Quantization**: Use fixed-point arithmetic for hardware efficiency

6.5 Error Probability Analysis

The error probability for uncoded BPSK provides a performance baseline:

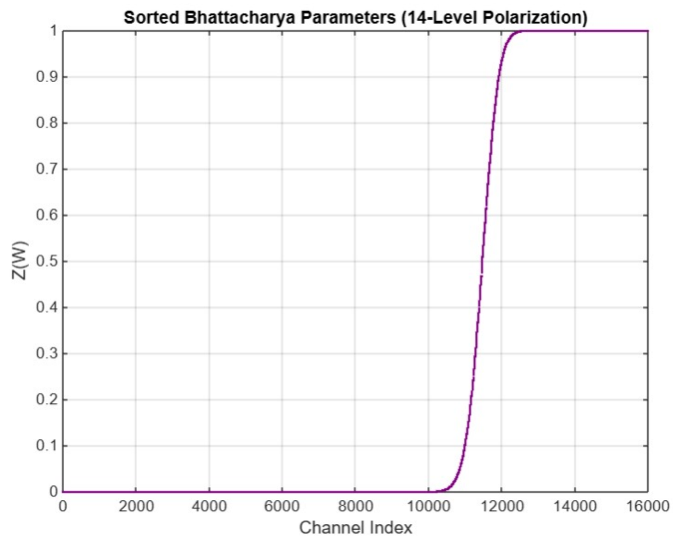

Figure 14: Encoding using Binary tree

$$P_e = Q\left(\sqrt{2\gamma}\right) = \frac{1}{2}\text{erfc}\left(\sqrt{\gamma}\right) \tag{12}$$

where $\gamma = E_b/N_0$ is the SNR per bit.

6.5.1 Decision Regions

The optimal detector decides:

$$\hat{x} = \begin{cases} +1 & \text{if } y \geq 0 \\ -1 & \text{if } y < 0 \end{cases}$$

6.5.2 Error Events

Two error events can occur:

1. When $x = +1$ is transmitted but $y < 0$:

$$P(e|x = +1) = P(n < -1) = \int_{-\infty}^{-1} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{t^2}{2\sigma^2}} dt$$

2. When $x = -1$ is transmitted but $y \geq 0$:

$$P(e|x = -1) = P(n > 1) = \int_{1}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{t^2}{2\sigma^2}} dt$$

6.5.3 SNR Relationship

The signal-to-noise ratio (SNR) per bit is:

$$\text{SNR} = \frac{E_b}{N_0} = \frac{1}{2\sigma^2}$$

6.5.4 Approximations and Asymptotics

For high SNR (SNR $\gg 1$), the error probability can be approximated:

$$P_{\text{error}} \approx \frac{1}{\sqrt{4\pi \cdot \text{SNR}}} e^{-\text{SNR}}$$

This shows the characteristic exponential decay of error probability with increasing SNR.

6.6 Algorithm for Reliability Sequence Generation

The reliability sequence generation algorithm is a recursive process that determines the relative reliability of polarized channels. This algorithm is fundamental to polar code construction and operates as follows:

6.6.1 Initialization

The algorithm begins by computing the initial error probability for the mother channel:

$$p = Q\left(\sqrt{\frac{\text{SNR}}{2}}\right)$$

where:

- $\text{SNR} = 1/(2\sigma^2)$ is the signal-to-noise ratio
- $Q(\cdot)$ is the Q-function representing the tail probability

6.6.2 Recursive Channel Transformation

For each polarization level, the algorithm generates two new channel types:

- W^- **channels (combining operation)**:

  $$p^- = 2p - p^2$$

  This represents the error probability after channel combining, where the reliability decreases (higher error probability).

- W^+ **channels (splitting operation)**:

  $$p^+ = p^2$$

  This represents the error probability after channel splitting, where the reliability improves (lower error probability).

6.6.3 Recursive Application

The algorithm applies these transformations recursively:

(a) Start with the mother channel W with error probability p

(b) For each polarization level l from 1 to $n = \log_2 N$:

  - Generate 2^l channels by applying W^- and W^+ transformations to all channels from previous level
  - Maintain a binary tree structure tracking the transformation history

(c) After n levels, obtain N polarized channels with distinct reliabilities

6.6.4 Example

For $N = 4$ and initial $p = 0.1$:

(a) Level 1: $W^- = 0.19$, $W^+ = 0.01$

(b) Level 2:

- $W^{--} = 0.3439$, $W^{-+} = 0.0361$
- $W^{+-} = 0.0199$, $W^{++} = 0.0001$

(c) Sorted sequence: $W^{++}, W^{+-}, W^{-+}, W^{--}$

6.7 Numerical Results

We evaluate the construction method for $N = 1024$ at various SNRs:

Table 1: Information Set Sizes for Different Rates (N=1024)

SNR (dB)	Rate 1/2	Rate 3/4	Rate 1/4
0	512	768	256
2	512	768	256
4	512	768	256
6	512	768	256

6.8 Conclusion

This paper presented a complete framework for polar code construction using Gaussian approximation of LLR distributions. The method provides:

- Efficient channel reliability estimation through recursive mean LLR calculation
- Optimal bit allocation via Bhattacharya parameter analysis
- Practical implementation with complexity $\mathcal{O}(N \log N)$

Future research directions include:

- Extension to fading channels
- Adaptive construction for time-varying conditions
- Improved approximations for higher accuracy

7 Achieving Shannon's Channel Capacity Using Polar Codes

Polar codes, introduced by Erdal Arıkan in 2008, represent a landmark achievement in coding theory by being the first class of codes that are both provably capacity-achieving and efficiently decodable. They exploit a unique concept known as **channel polarization** to approach Shannon's channel capacity limit.

7.1 Channel Polarization

Channel polarization is a recursive transformation process where $N = 2^n$ identical binary-input channels are combined and split in a way that results in N synthesized channels with extreme reliabilities. Specifically:

- A subset of channels become **highly reliable** (error probability $\to 0$),
- The remaining channels become **highly unreliable** (error probability $\to 1$).

This means that intermediate-quality channels effectively vanish as N increases. Information bits are sent only through the reliable channels, while fixed (frozen) bits are assigned to the unreliable ones, ensuring overall reliable transmission.

7.2 Binary Erasure Channel (BEC) Case Study

For a Binary Erasure Channel (BEC) with erasure probability p, the capacity is given by:

$$I = 1 - p$$

In each polarization step:

$$\begin{aligned} \text{Good Channel:} \quad & \varepsilon^{+} = p^2 \\ \text{Bad Channel:} \quad & \varepsilon^{-} = 2p - p^2 \end{aligned}$$

Over successive transformations, the proportion of reliable channels approaches the channel capacity I, and the rest tend to have capacity 0.

7.3 Theoretical Foundation: Martingale Convergence

The polarization effect is supported by the **Martingale Convergence Theorem**, which guarantees that as $N \to \infty$, the synthetic channels' reliabilities converge to either 0 or 1 with probability 1. Thus, the fraction of reliable channels converges to the channel capacity.

7.4 Capacity-Achieving Coding Strategy

After polarization:

- **Information bits** are assigned to the most reliable synthesized channels.
- **Frozen bits** (e.g., zeros) are assigned to the unreliable ones.

Because the number of reliable channels asymptotically approaches $N \cdot I$, polar codes effectively achieve the Shannon capacity of the original channel.

7.5 Applications and Impact

Due to their low encoding and decoding complexity ($\mathcal{O}(N \log N)$) and provable capacity-achieving performance, polar codes have been adopted in the **5G New Radio (NR)** standard for control channels.

They offer:

- Provably optimal performance under successive cancellation decoding,
- Excellent scalability for large block lengths,
- Improved finite-length performance when combined with CRC and list decoding.

7.6 Conclusion

Through channel polarization, polar codes provide a structured, efficient, and theoretically sound approach to achieve reliable communication at rates approaching Shannon's limit. Their incorporation into modern wireless standards marks a significant advancement in the practical application of information theory.

References

[1] E. Arıkan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Transactions on Information Theory*, vol. 55, pp. 3051–3073, July 2009.

[2] A. Thangraj, "Introduction to polar codes: Polar transform." https://www.youtube.com/watch?v=rB0rhQKyV34t=81s, 2019. Video lecture.

[3] E. Telatar, "The flesh of polar codes." Conference presentation, June 2017. 2017 IEEE International Symposium on Information Theory (ISIT), Aachen, Germany.

[4] EventHelix, "Polar codes: Develop an intuitive understanding of polar codes," Apr. 2019. Accessed: 2025-05-06.

[5] K. Niu, P. Zhang, J. Dai, Z. Si, and C. Dong, "A golden decade of polar codes: From basic principle to 5g applications," *China Communications*, vol. 20, pp. 94–121, Feb. 2023.

[6] R. M. Oliveira and R. C. D. Lamare, "Construction of polar codes based on piecewise gaussian approximation," in *2021 17th International Symposium on Wireless Communication Systems (ISWCS)*, pp. 1–5, IEEE, 2021.

[7] S. Mude and B. Naik, "Minimization of bit error rate in polar codes for achieving channel capacity," *International Journal of Engineering and Advanced Technology*, vol. 9, pp. 1433–1436, October 2019.

[1] [2] [3] [4] [5] [6] [7]