Dhirubhai Ambani University

Introduction to Communication Systems
(CT216)

Polar codes

Dhirubhai Ambani
University
Formerly known as
DA-IICT

Group - 23

Mentor: Darpan Lunagariya
Prof: Yash Vasavada

April 22, 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.

Student Information:

Roll Number	Name
202301245	ISHAN THAKKAR
202301246	SHREYAS DUTTA
202301248	MOHAMMAD FARZAN BHALARA
202301249	JENIL SHAH
202301250	VIVEK DODIYA
202301251	DEVARSH VASANI
202301253	DHYEY RAVAL
202301254	JAYADITYA SHAH
202301255	RUDRA DESAI
202301256	FARHAN ANSARI
202301257	KANU BHADARAKA

Contents

Chapter 1

Introduction

$3GPP$ has selected polar codes as the error correcting code on the $5G$ NR control channels. Polar codes are unique in the way they split the channel into good and bad bit-channels.
What do we mean by good and bad bit-channels? Consider a polar code where K information bits are being sent in a block of N bits. Polar code encoding will polarize the channel into reliable and unreliable bit-channels. The information bits will be transmitted on the most reliable K bit-channels. The remaining N-K channels are unreliable are usually set to 0 as they are not reliable for data transmission.

1.1 Gerrymandering

What is this process of polarization? Consider the US practice of gerrymandering illustrated in the following image:

- The rectangle on left depicts a population with 50% blue voters and 50% red voters.
- The middle rectangle represents an electoral district split that still contain close to 50% blue and red voters in each district.
- The right rectangle shows a polarizing split that draws the electoral districts so as to make the bluer and redder.

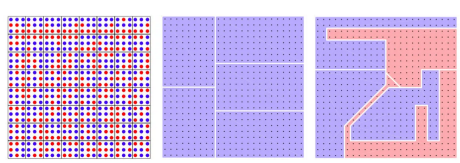


Figure 1.1: Polarizing with Gerrymandering

1.2 Channel Polarization

How can we polarize a data channel into extremal good and bad cha Now let's define W before we proceed further. Let's model W to the Binary Erasure Channel (BEC). BEC is defined as:Let us look at two channels shown below. Both channels implement the transform W that converts the input X to the output Y.

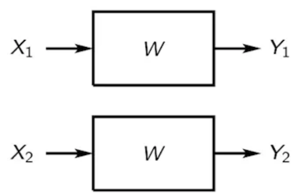


Figure 1.2: Two independent channels

Now consider two inputs U_1 and U_2 that feed into the inputs X_1 and X_2. The connections are as made as shown below.

$$X_1 = U_1 \oplus U_2$$
$$X_2 = U_2$$

This results in the network shown below. We will see later that this simple network polarizes the output.

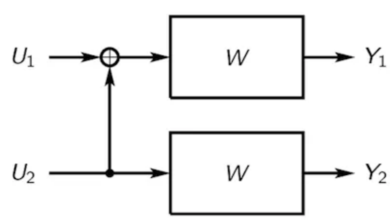


Figure 1.3: Connecting the two-channel as shown above polarizes the output

1.2.1 Binary erasure channel

Now let us define W before we proceed further. Let's model W to the Binary Erasure Channel (BEC). BEC is defined as:

- X is unsuccessfully decoded with a probability p. In the BEC, the bit is erased, so the receiver is able to detect that a bit has been lost.

- X is successfully decoded into Y with a probability 1-p.

Now let us analyze the network we discussed above in terms of two bit-channels:

- W^{-}:$U_1 \rightarrow (Y_1, Y_2)$
- W^{+}:$U_2 \rightarrow (Y_1, Y_2)$

1.2.2 W^{-}: U_1 gives (Y_1, Y_2)

The W^{-} channel tries to reconstruct the input received on the U_1 channel. The following possibilities exist when the W channel is mo

- Both channels are successfully decoded — probability $(1-p)^2$
- The first channel is erased but the second one is decoded successfully — probability $p(1-p)$
- The first channel is decoded successfully but the second channel is erased probability $(1-p)p$
- Both channels are erased — probability p^2

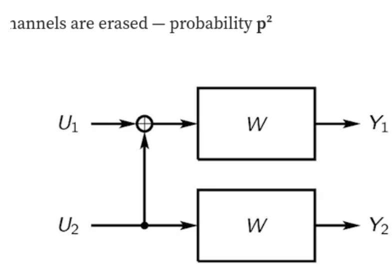


Figure 1.4: Polarizing channel set

These possibilities are summarized below: Here U_1 can be successfully decoded

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

Figure 1.5: Connecting the two-channel as shown above polarizes the output

only in the first case as U_1 can be extracted from $u_1 \oplus u_2$ as U_2 was also decoded successfully. All other cases represent decode failure as it is not possible to extract U_1.

We have created a W^{-} with the following characteristics..

- U_1 is successfully decoded with the probability $(1-p)^2$
- U_1 decode fails with the probability of $p(1-p)+(1-p)p+p^2$ $=2p-p^2$

Note here that we have managed to create a worse channel than the BEC. If the BEC had a 30% chance or erasure:

- BEC Success probability is 0.7
- BEC Erasure probability is 0.3

For the W^- bit-channel:

- W^- success probability is $(1.0-0.3)^2 = 0.7^2 = 0.49$
- W^- erasure probability is $(2*0.3–0.3)^2 = 0.6–0.09 = 0.51$

1.2.3 W^+: U_2 gives (Y_1, Y_2)

Now let us turn our attention to the W^+ channel. The channel reconstructs the input received on the U_2 channel. With a BEC channel, we have the following possibilities:

- Both channels are successfully decoded — probability $(1-p)^2$
- The first channel is erased but the second one is decoded successfully — probability p(1-p)
- The first channel is decoded successfully but the second channel is erased — probability (1-p)p
- Both channels are erased — probability p^2

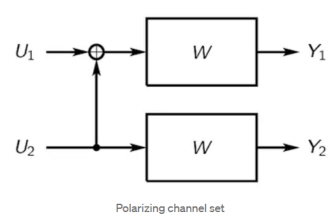


Figure 1.6: Polarizing channel set

These possibilities are summarized below:

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

Figure 1.7: Connecting the two-channel as shown above polarizes the output

Note: From the previous section we know that W^- is a worse channel that the BEC. If we send a known value, this can help in improving the decode for the W^+ channel. This is part of our plan to polarize an OK channel into a good and bad bit-channels. We will send our data bits on the good bit-channel and clamp the bad bit-channel to a known value.

Here U_2 can be successfully decoded in the first and second cases as U_2 was received successfully. In the third case, U_2 can be extracted from $U_1 \oplus U_2$ as we have advance knowledge of U_1. The last case represents a decode failure as it is not possible to extract U_2.

We have created a W^+ with the following characteristics:

- U_1 is successfully decoded with the probability $(1-p)^2+p(1-p)+(1-p)p = 1-p^2$
- U_1 decode fails with the probability of p^2

Note here that we have managed to create a better channel than the BEC. If the BEC had a 30% chance or erasure:

- BEC Success probability is 0.7
- BEC Erasure probability is 0.3

For the W^+ bit-channel:

- W^+ Success probability is 1.0–0.3^2 = 1–0.09= 0.91
- W^+ Erasure probability is $0.3^2 = 0.09$

1.2.4 Experimenting with Polarization

We saw above that when the polarizing transform was applied to a Binary Erasure Channel (BEC) with an erasure probability of p, we ended up with two bit-channels:

- W^+ with an erasure probability of p^2

– W^- with an erasure probability of $2p - p^2 = p(2-p)$

The results above clearly show that W^+ has a lower erasure probability than W^-. Since $0 \leq p \leq 1$, it follows that:

$$p^2 \leq p(2-p)$$

1.3 Polarize Once

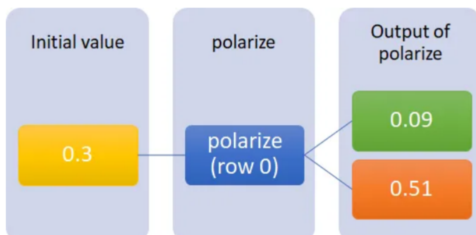


Figure 1.8: Polarization applied once to a channel

1.4 Polarize Twice

A polarizing building block that takes two OK bit-channels and produces a good bit-channel W^+ and a bad bit-channel W^-.Cascading this building block as shown below.

The cascade results in the following channels:

$$\begin{aligned}
W^{--}: &\quad V_1 \to Y_1Y_2Y_3Y_4 \\
W^{-+}: &\quad V_2 \to Y_1Y_2Y_3Y_4V_1 \\
W^{+-}: &\quad V_3 \to Y_1Y_2U_1Y_3Y_4U_3 \\
W^{++}: &\quad V_4 \to Y_1Y_2U_1Y_3Y_4U_3V_3
\end{aligned}$$

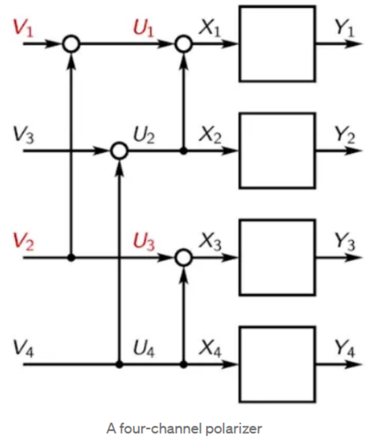


Figure 1.9: Cascading the polarizing building block

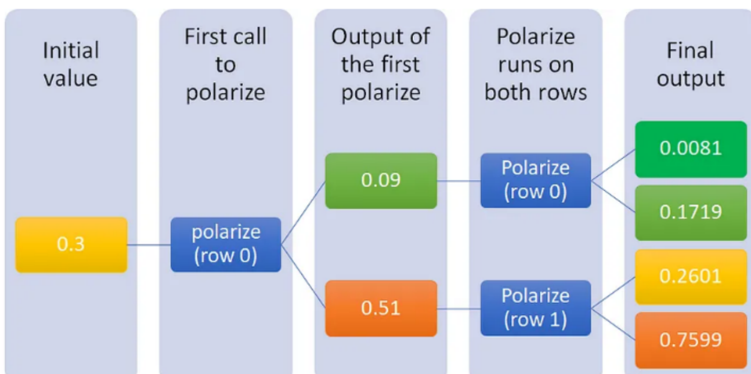


Figure 1.10: Visualization of a cascade of two "polarize" scripts. We see in the final output the polarization into good (green) and bad channels (orange and red)

So we can say that when the number of channels increases, the error probability decreases.

Chapter 2

Bhattacharyya Parameter

Let W be a binary-input AWGN channel with input $X \in \{0, 1\}$ mapped using BPSK:

$$0 \mapsto +1, \quad 1 \mapsto -1$$

The channel output is:

$$Y = X + N, \quad N \sim \mathcal{N}(0, \sigma^2)$$

The Bhattacharyya parameter is defined as:

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

For continuous-output channels like AWGN, this becomes:

$$Z(W) = \int_{-\infty}^{\infty} \sqrt{W(y|0) \cdot W(y|1)}\, dy$$

Given the conditional PDFs:

$$W(y|0) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y-1)^2}{2\sigma^2}\right), \quad W(y|1) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y+1)^2}{2\sigma^2}\right)$$

Now compute the square root of the product:

$$\sqrt{W(y|0) \cdot W(y|1)} = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(-\frac{(y-1)^2 + (y+1)^2}{4\sigma^2}\right)$$

Simplify the exponent:

$$(y-1)^2 + (y+1)^2 = y^2 - 2y + 1 + y^2 + 2y + 1 = 2y^2 + 2$$

So the expression becomes:

$$\sqrt{W(y|0) \cdot W(y|1)} = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{2y^2+2}{4\sigma^2}\right) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{1}{2\sigma^2}\right) \cdot \exp\left(-\frac{y^2}{2\sigma^2}\right)$$

Therefore,

$$Z(W) = \int_{-\infty}^{\infty} \frac{1}{2\pi\sigma^2} \exp\left(-\frac{1}{2\sigma^2}\right) \cdot \exp\left(-\frac{y^2}{2\sigma^2}\right) dy = \exp\left(-\frac{1}{2\sigma^2}\right) \cdot \int_{-\infty}^{\infty} \frac{1}{2\pi\sigma^2} \exp\left(-\frac{y^2}{2\sigma^2}\right) dy$$

Note that:

$$\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{y^2}{2\sigma^2}\right) dy = 1$$

But our integrand is scaled by another $\frac{1}{\sqrt{2\pi\sigma^2}}$, so:

$$\int_{-\infty}^{\infty} \frac{1}{2\pi\sigma^2} \exp\left(-\frac{y^2}{2\sigma^2}\right) dy = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot 1 = \frac{1}{\sqrt{2\pi\sigma^2}}$$

Thus, the Bhattacharyya parameter is:

$$Z(W) = \exp\left(-\frac{1}{2\sigma^2}\right) \cdot \frac{1}{\sqrt{2\pi\sigma^2}}$$

Often, the exponential term is emphasized in polar code analysis:

$$Z(W) \propto \exp\left(-\frac{1}{2\sigma^2}\right)$$

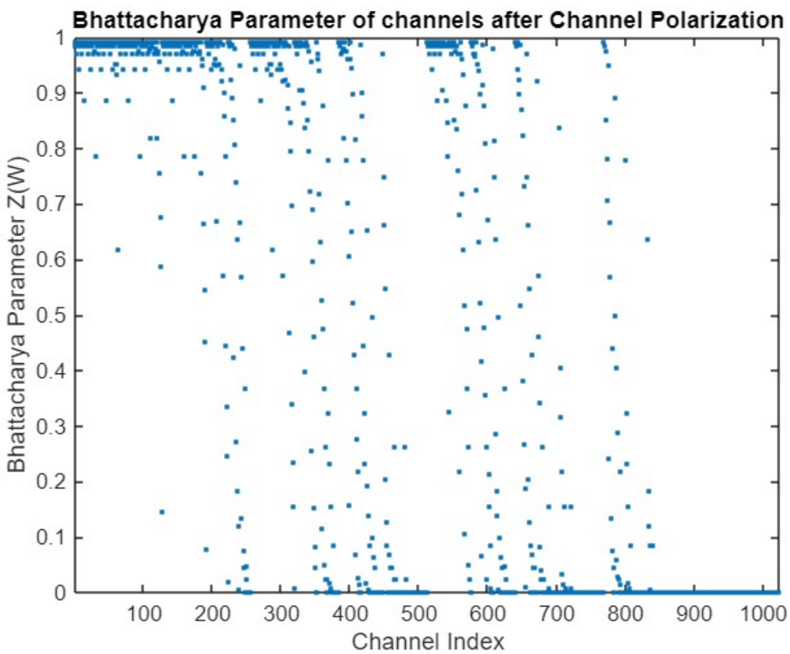


Figure 2.1: Bhattacharya Parameter

Chapter 3

Encoding in Polar Codes

To understand the process of encoding in polar codes, we first need to understand the **Kronecker product**, which is used to construct the generator matrix G_N.

We begin with the base matrix:

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

The **Kronecker product** is computed by replacing each element $G_2(i,j)$ with $G_2(i,j) \cdot G_2$. This recursively expands the matrix to form a larger structured matrix.

To construct G_4, we take the Kronecker product:

$$G_4 = G_2 \otimes G_2$$

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

In general, for a codeword of length N, the generator matrix G_N is given by:

$$G_N = G_2^{\otimes n}, \quad \text{where } N = 2^n$$

For an (N, K) polar code:

- $N = 2^n$ is the block length.
- The message m is a K-bit vector.

3.0.1 Steps to Form the Encoded Message

1. **Form vector u of length N:**
    - Select the $N - K$ least reliable bit positions using a *reliability sequence.*

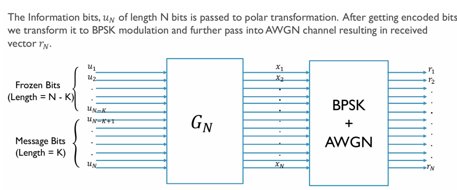
- Set the bits $u_i = 0$ at these positions — these are called **frozen bits**.
- Insert the K message bits into the remaining positions — these are called **message bits**.

2. **Encoding:**

$$x = u \cdot G_N$$

where x is the encoded codeword of length N.

This encoded sequence is then transmitted over a **BPSK (Binary Phase Shift Keying)** modulation scheme followed by an **AWGN (Additive White Gaussian Noise)** channel.



Graphical/tree representation of polar encoding

3.0.2 Graph Structure of Polar Encoding

Polar encoding can be represented as a **binary tree structure** with $\log_2 N$ stages and N bits flowing through each stage. This graph captures the recursive bit-combination process defined by the generator matrix G_N.

Each non-leaf node in the graph is computed using two child nodes: the XOR of both children forms the left output, while the right output passes the second (right) child unmodified. This is equivalet to the mathematical operation:

$$(u_0, u_1) \rightarrow (u_0 \oplus u_1, u_1)$$

3.0.3 Recursive Construction:

At each level of the graph:

- Nodes take two input bits and output their XOR and the second bit itself.
- This operation propagates upward in a binary tree of XORs, combining bits in stages of increasing scope.

Properties of the Encoding Tree:

1. **Depth:** $\log_2 N$ stages, corresponding to the levels of Kronecker expansion of G_2.
2. **Width:** Each level has N signal lines (1 per bit).

3. **Recursiveness:** The structure follows the recursive identity:

$$G_{2N} = \begin{bmatrix} G_N & 0 \\ G_N & G_N \end{bmatrix}$$

4. **Parallelism:** All XOR operations in a stage are independent and can be executed in parallel, enabling hardware acceleration.

5. **Time Complexity:** The Time Complexity for encoding is O(NlogN).

Chapter 4

Decoding in Polar Codes

4.1 Successive Cancellation Decoder (SC)

Due to the recursive structure of channel polarization, we can use a simple decoding scheme called **Successive Cancellation (SC)** for the decoding purpose.

The Successive Cancellation (SC) Decoder is the basic decoding algorithm for polar codes, which works sequentially. Due to the sequential nature of the SC decoder, the decoding latency is relatively high.

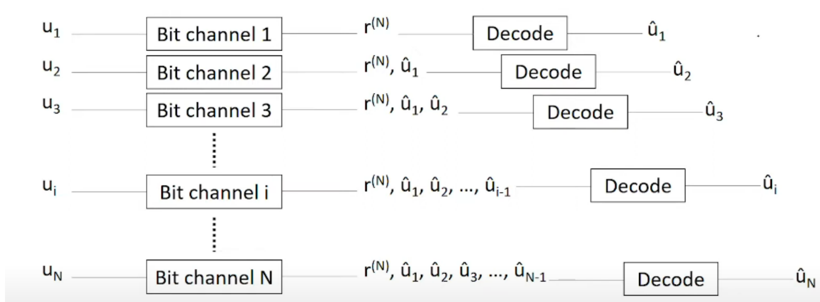


Figure 4.1: If bit i is frozen, then $\hat{u}_i = 0$.

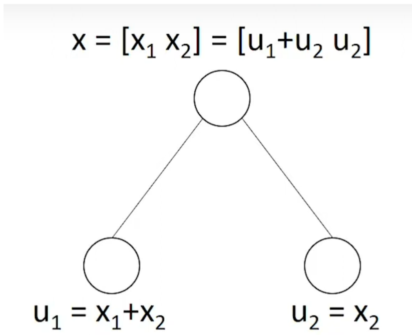
Basic Building Block of SC Decoder: $N = 2$

We start with a 2-bit input vector:

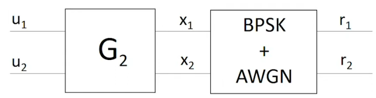
$$x = [x_1 \ x_2] = [u_1 \oplus u_2 \quad u_2]$$

From the received symbols:

$$u_1 = x_1 \oplus x_2, \quad u_2 = x_2$$



$u_1 = x_1 \oplus x_2, \quad u_2 = x_2$



Transformation

- Soft-In Soft-Out (SISO) decode u_1 first (using a Single Parity Check (SPC)).
- Given $\hat{u}_1$, decode u_2 (as a Repetition (Rep) code).

Log-Likelihood Ratio (LLR) for u_1:

$$L(u_1) = f(r_1, r_2) = \text{sgn}(r_1) \cdot \text{sgn}(r_2) \cdot \min(|r_1|, |r_2|)$$

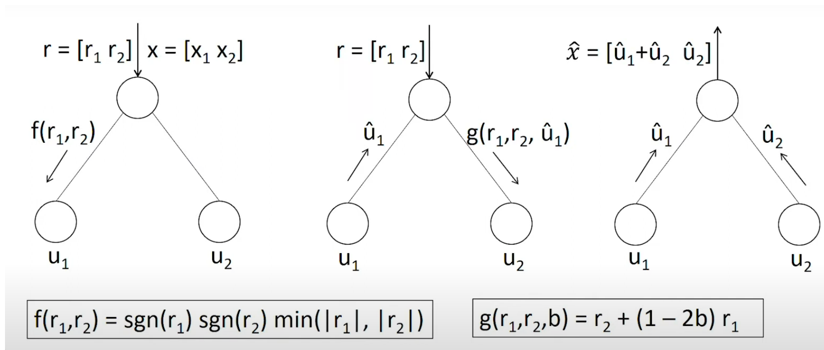
Estimated bit:

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

Then, based on $\hat{u}_1$, compute $L(u_2)$ as:

$$L(u_2) = \begin{cases} r_2 + r_1, & \text{if } \hat{u}_1 = 0 \quad (\text{since } x = [u_2 \; u_2]) \\ r_2 - r_1, & \text{if } \hat{u}_1 = 1 \quad (\text{since } x = [\bar{u}_2 \; u_2]) \end{cases}$$

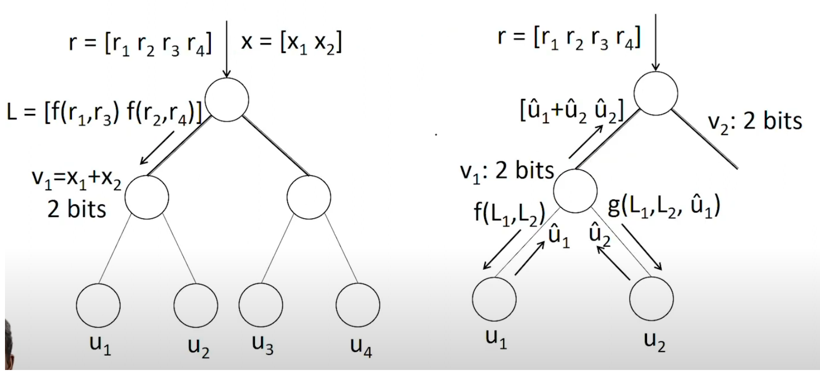
First, the received vector r_N is decoded using a soft-decision decoding algorithm. In soft-decision decoding, the reliability of each received bit is assessed using the Log-Likelihood Ratio (LLR). These soft bits are then used to compute the estimated hard bits with the help of the F-function, G-function, and partial sums as shown below:



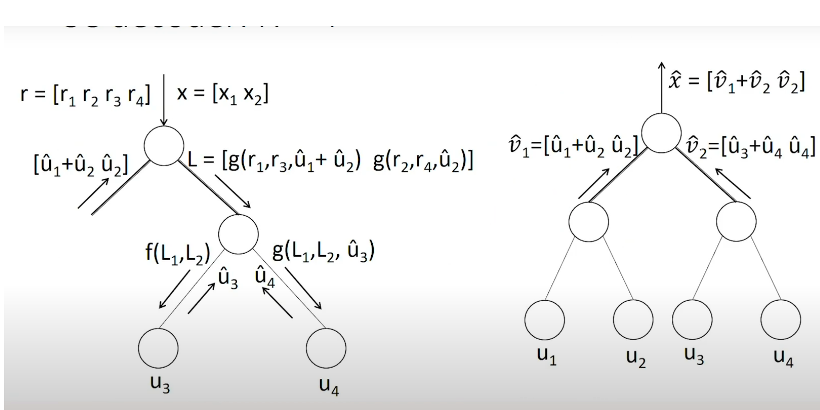
Working of decoding

SC Decoder: $N = 4$

As polar codes have recursive structure , we can apply this algorithm for decoding any length of codeword. Let us take an example of length 4 SC decoder and understand its behaviour.



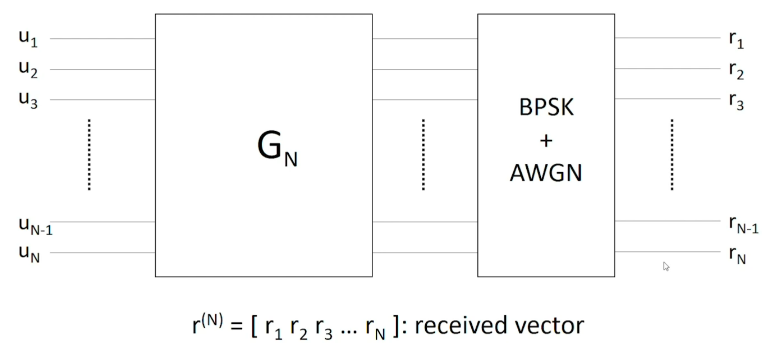
Working of decoding for N=4



Working of decoding for N=4

SC Decoder: General Polar Code

Now let us see hoe decoding works for a general polar code.

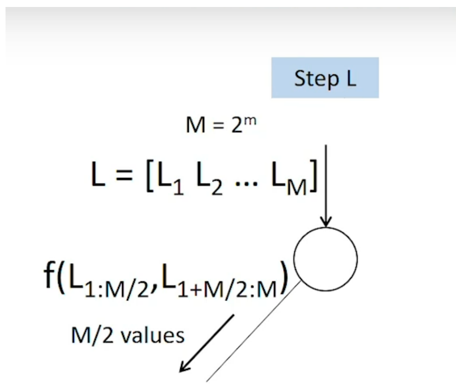


General Setup

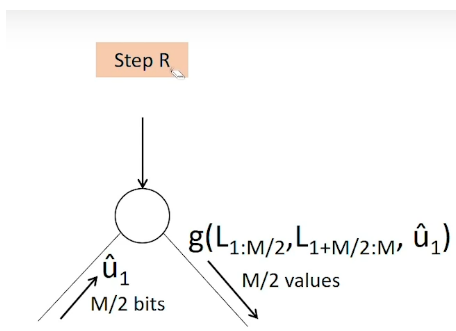
4.1.1 Operations at Each Interior Node

For any interior node present in the tree-like presentation, the following three operations take place:

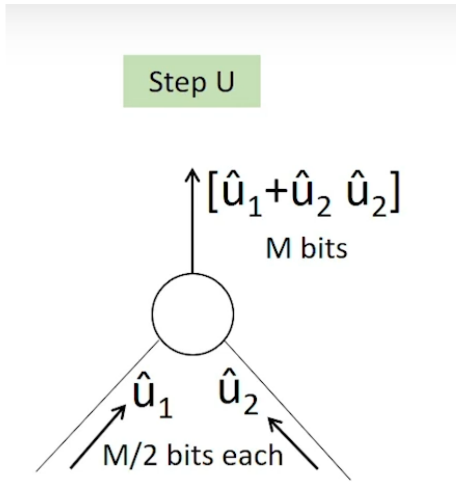
1. **Step L**



2. **Step R**



3. **Step U**

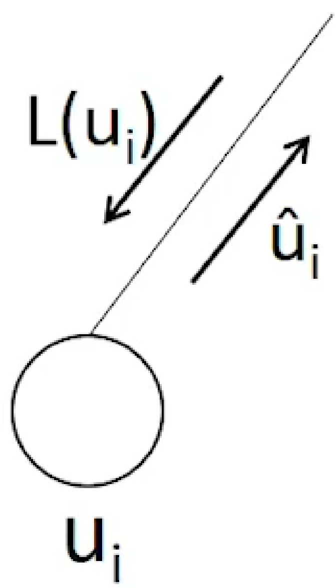


Here:

$$f(a_1{:}p,\ b_1{:}p) = [f(a_1, b_1),\ f(a_2, b_2),\ \ldots,\ f(a_p, b_p)]$$
$$g(a_1{:}P,\ b_1{:}P,\ c) = [g(a_1, b_1, c),\ g(a_2, b_2, c),\ \ldots,\ g(a_P, b_P, c)]$$

$$g(r_1, r_2, b) = r_2 + (1 - 2b) \cdot r_1$$
$$f(r_1, r_2) = \mathrm{sgn}(r_1) \cdot \mathrm{sgn}(r_2) \cdot \min\left(|r_1|,\ |r_2|\right)$$

4.1.2 Operation at Leaf Node



Leaf Node

$$\hat{u}_i = \begin{cases} 0, & \text{if } i \text{ is a frozen position} \\ 0, & \text{if } i \text{ is a message position and } L(u_i) \geq 0 \\ 1, & \text{if } i \text{ is a message position and } L(u_i) < 0 \end{cases}$$

Polar Code Decoding – Node-wise Summary: At every node:

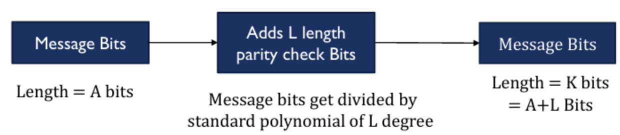
- **If not a leaf**, perform the following steps in sequence:
  - Perform **Step L** and move to the left child.
  - Once a decision is received from the left child, perform **Step R** and move to the right child.
  - Once a decision is received from the right child, perform **Step U** and return to the parent.
- **If a leaf node**, make the decision and return to the parent.

4.2 Successive Cancellation List Decoder (SCL)

SCL(Successive Cancellation List Decoder) somewhat is similar to SC decoding,given the fact that now it maintains a list to store more than one possible codewords that are most likely to be transmitted . SCL decoding typically results in better error correction performance compared to SC decoding, especially in situations with high noise or challenging channel conditions.In SCL path represents a hypothesis about the transmitted codeword, and the decoder explores the multiple possible hypothesis simultaneously.

CRC-Aided SCL Decoding

- CRC is used to select a single codeword from the list of most likely possible codewords.
- During polar encoding adding bits for CRC:



- The nearest (most likely) correct codeword is selected using the CRC check.
- If no codeword in the list passes the CRC check, it is declared a decoding failure.

List Decoding, Decision Metric, and Path Metric

- Now we will proceed to see how we produce multiple codewords at decoder?
- In Polar SCL decoding, both decisions (0 and 1) for each bit are considered.
- Assign a **Decision Metric (DM)** for each possible bit decision:

$$\text{If } L(u_i) \geq 0 : \begin{cases} \hat{u}_i = 0 \Rightarrow \text{DM}_i = 0 \\ \hat{u}_i = 1 \Rightarrow \text{DM}_i = |L(u_i)| \end{cases}$$

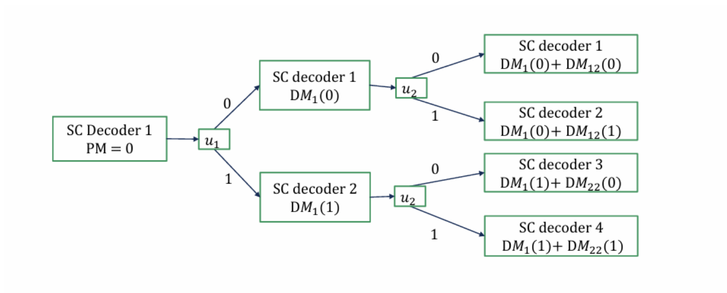
$$\text{If } L(u_i) < 0 : \begin{cases} \hat{u}_i = 1 \Rightarrow \text{DM}_i = 0 \\ \hat{u}_i = 0 \Rightarrow \text{DM}_i = |L(u_i)| \end{cases}$$

- **Note:** The Decision Metric (DM) is assigned even if index i corresponds to a frozen bit, that is, $\hat{u}_i = 0$ is the only decision taken.
  - If $L(u_i) \geq 0$, then $\text{DM}_i = 0$
  - If $L(u_i) < 0$, then $\text{DM}_i = |L(u_i)|$

- **Path Metric (PM)**: sum of all decision metrics along a decoding path.

$$\text{PM} = \sum_i \text{DM}_i$$

Successive Cancellation List (for $N = 4$)



Chapter 5

Shannon Channel Capacity and how Polar Codes Achieve it

5.1 Error Probability Calculation for AWGN Channel

Consider the probability:

$$P(X = 0 \text{ and } n < -1) + P(X = 1 \text{ and } n > 1)$$

Since X and n are independent, we can write:

$$P(X = 0)P(n < -1) + P(X = 1)P(n > 1)$$

Given that $P(X = 0) = P(X = 1) = 0.5$ and $P(n < -1) = P(n > 1)$, we obtain:

$$\text{Error} = 2 \cdot P(X = 0) \cdot P(n < -1) = 2 \cdot 0.5 \cdot P(n < -1)$$

Assume $n \sim \mathcal{N}(0, \sigma^2)$ is a normal random variable with mean 0 and variance σ^2. Then:

$$P(n < -1) = \int_{-\infty}^{-1} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}}\, dx$$

Changing variable:

$$\frac{x^2}{2\sigma^2} = t^2 \Rightarrow x = t\sqrt{2}\sigma, \quad dx = \sqrt{2}\sigma\, dt$$

The integral becomes:

$$\int_{\frac{1}{\sqrt{2}\sigma}}^{\infty} \frac{1}{\sqrt{\pi}} e^{-t^2} dt$$

Therefore, the total error probability is:

$$\text{Error} = 2 \cdot \int_{\frac{1}{\sqrt{2\sigma}}}^{\infty} \frac{1}{\sqrt{\pi}} e^{-t^2} dt = \text{erfc}\left(\frac{1}{\sqrt{2\sigma}}\right)$$

Where the complementary error function is defined as:

$$\text{erfc}(x) = 1 - \text{erf}(x)$$

and the error function is:

$$\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$$

Thus, the error probability becomes:

$$\text{Error} = 0.5 \cdot \text{erfc}\left(\frac{1}{\sqrt{2\sigma}}\right)$$

Q-Function Representation

In terms of the Q-function (tail probability of the standard normal distribution):

$$Q(x) = 0.5 \cdot \text{erfc}(x)$$

The error probability can be expressed as:

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

5.2 Proof that Polar Codes Achieve Shannon Channel Capacity

Whether we use any channel model for adding noise, the behavior of the error probability p is similar:

- For the Binary Erasure Channel (BEC) or Binary Symmetric Channel (BSC), we define p directly as the error probability.
- For the Additive White Gaussian Noise (AWGN) channel, the error probability is given by:

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

In the process of channel polarization, the channels split recursively into two synthesized channels:

$$W^- : \quad p^- = 2p - p^2, \qquad W^+ : \quad p^+ = p^2$$

This recursive splitting continues with each polarization level.

As the block length $N \to \infty$, the following behavior emerges:

- The channel W^- becomes more noisy, with high entropy, and the mutual information $I(W^-)$ tends to 0.
- The channel W^+ becomes nearly noiseless, and $I(W^+)$ tends to $I(W)$.

In the limit, out of N total channels, approximately:

$$N(1 - I(W)) \text{ are noisy (useless)} \quad \text{and} \quad NI(W) \text{ are noiseless (useful)}$$

We send the message bits over the noiseless synthesized channels. Hence, the number of message bits is:

$$K = NI(W)$$

The rate of the polar code is:

$$\text{Rate} = \frac{K}{N} = \frac{NI(W)}{N} = I(W)$$

The Shannon capacity of the AWGN channel is:

$$C = \frac{1}{2}\log_2(1 + \text{SNR})$$

Since reliable communication is possible when:

$$\text{Rate} \leq C$$

and we know that:

$$I(W) \leq C$$

it follows that:

$$\text{Rate} \leq C$$

Hence, polar codes asymptotically achieve the Shannon channel capacity.

Theorem (Arıkan, 2009)

Let W be a binary-input discrete memoryless channel (B-DMC) with capacity $I(W)$. For any $\epsilon > 0$ and sufficiently large block length $N = 2^n$, there exists a subset $\mathcal{A} \subseteq \{1, 2, \ldots, N\}$ of indices with $|\mathcal{A}| \geq N(I(W) - \epsilon)$ such that the synthesized channels $W_N^{(i)}$, $i \in \mathcal{A}$, satisfy:

$$Z(W_N^{(i)}) < \epsilon,$$

where $Z(\cdot)$ is the Bhattacharyya parameter. Furthermore, there exists a polar code of rate $R \geq I(W) - \epsilon$ with vanishing block error probability under successive cancellation (SC) decoding.

Proof Outline

1. Channel Polarization

Given a binary-input memoryless symmetric channel W, polar codes transform $N = 2^n$ independent copies of W into N synthesized subchannels $\{W_N^{(i)}\}_{i=1}^N$ through a recursive channel combining and splitting process defined by:

$$W^-(y_1, y_2|u_1) = \sum_{u_2 \in \{0,1\}} \frac{1}{2} W(y_1|u_1 \oplus u_2) W(y_2|u_2),$$
$$W^+(y_1, y_2, u_1|u_2) = \frac{1}{2} W(y_1|u_1 \oplus u_2) W(y_2|u_2).$$

This recursive process leads to the polarization phenomenon:

$$\lim_{n\to\infty} \frac{1}{2^n} \left| \left\{ i : I(W_N^{(i)}) \in (1-\delta, 1] \right\} \right| = I(W),$$

$$\lim_{n\to\infty} \frac{1}{2^n} \left| \left\{ i : I(W_N^{(i)}) \in [0, \delta) \right\} \right| = 1 - I(W),$$

for any $0 < \delta < 1$.

2. Bhattacharyya Parameter and Reliability

The Bhattacharyya parameter $Z(W)$ is a measure of the reliability of a channel:

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

Under the polarization process, the Bhattacharyya parameters of the synthesized channels satisfy:

$$Z(W_N^{(2i-1)}) \le 2Z(W_i) - Z(W_i)^2, \quad Z(W_N^{(2i)}) = Z(W_i)^2.$$

This recursion ensures that the fraction of channels with $Z(W_N^{(i)}) \to 0$ approaches $I(W)$, and the fraction with $Z(W_N^{(i)}) \to 1$ approaches $1 - I(W)$.

3. Code Construction

Choose the index set $\mathcal{A} = \{i : Z(W_N^{(i)}) < \epsilon\}$ to carry information bits and set the remaining bits (frozen bits) to fixed values known at both transmitter and receiver.

4. Decoding and Error Probability

Under successive cancellation (SC) decoding, the block error probability is bounded by:

$$P_e \le \sum_{i \in \mathcal{A}} Z(W_N^{(i)}).$$

Since $|\mathcal{A}| \approx NI(W)$ and each $Z(W_N^{(i)}) < \epsilon$, the total error probability vanishes as $N \to \infty$.

5. Conclusion

The constructed polar code has:

- Rate $R = \frac{|\mathcal{A}|}{N} \to I(W)$,
- Block error probability under SC decoding tending to 0 as $N \to \infty$.

Hence, polar codes achieve the capacity $C = I(W)$ of any binary-input symmetric memoryless channel.

Chapter 6

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, no. 7, pp. 3051–3073, Jul. 2009. [Online]. Available: `https://ieeexplore.ieee.org/document/5075875`

2. "Polar Codes - 5G NR," *Medium*, [Online]. Available: `https://medium.com/5g-nr/polar-codes-703336e9f26b`

3. NPTEL Course: Error Control Coding by Prof. Andrew Thangaraj, *IIT Madras*. [Online]. Available: `https://archive.nptel.ac.in/noc/courses/noc19/SEM1/noc19-ee29/`

4. P. Kumawat and M. Vijayvargiya, "Survey Paper on Polar Code for 5G," *International Journal of Engineering and Advanced Technology (IJEAT)*, vol. 9, no. 1, Oct. 2019. [Online]. Available: `https://www.ijeat.org/wp-content/uploads/papers/v9i1/A1233109119.pdf`

5. E. Arıkan, "Systematic Polar Coding," *IEEE Communications Letters*, vol. 15, no. 8, pp. 860–862, Aug. 2011. [Online]. Available: `https://ieeexplore.ieee.org/document/6823688`

6. "Implementation of Polar Codes in 5G," *StudyLib.net*. [Online]. Available: `https://studylib.net/doc/27190911/implementation-ofplor-codes-in-5g`

7. Darpan Lunagariya (ID: 202201462), "Polar Codes - Project Report," Internal Project Report, DA-IICT, 2023.

8. Video Lectures by Prof. Andrew Thangaraj on Polar Codes, *NPTEL*. [Online].

9. E. Arıkan, "Slides on Polar Codes," *Simons Institute*, [Online]. Available: `https://simons.berkeley.edu/sites/default/files/docs/2689/slidesarikan.pdf`

10. Prof. Erdal Arıkan, "Lecture Slides on Polar Codes."