Polar Decoding on Faulty Hardware

Alexios Balatsoukas-Stimming

Telecommunications Circuits Laboratory
EPFL

October 20, 2016

Polar Codes

Polar Decoding on Faulty Hardware

Polar codes are a new kind of codes discovered by Arıkan in 2009.

Polar Codes

Polar Decoding on Faulty Hardware

Polar codes are a new kind of codes discovered by Arıkan in 2009.

- Few breakthroughs in coding theory since Turbo and LDPC codes.
- Currently under consideration for **adoption in 5G**.

Polar Codes

Polar Decoding on Faulty Hardware

Polar codes are a new kind of codes discovered by Arıkan in 2009.

- Few breakthroughs in coding theory since Turbo and LDPC codes.
- Currently under consideration for **adoption in 5G**.

Advantages

- Explicit construction $\rightarrow$ **no random ensemble** to pick from.
- **Provably capacity achieving**.
- **Fine-grained rate-adaptation** in a natural way.

Polar Codes

Polar Decoding on Faulty Hardware

Polar codes are a new kind of codes discovered by Arıkan in 2009.

- Few breakthroughs in coding theory since Turbo and LDPC codes.
- Currently under consideration for **adoption in 5G**.

Advantages

- Explicit construction $\rightarrow$ **no random ensemble** to pick from.
- **Provably capacity achieving**.
- **Fine-grained rate-adaptation** in a natural way.

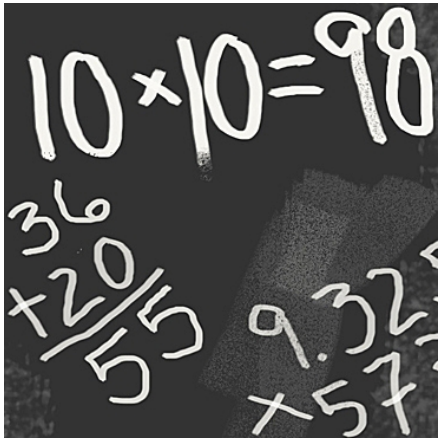
Disadvantages

- **Lower parallelism** than LDPC codes $\rightarrow$ high decoding latency.
- **Worse error-correcting performance** than LDPC codes.

Approximate Computing

Polar Decoding on **Faulty Hardware**

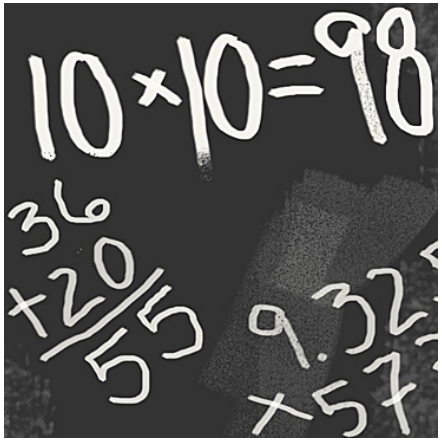
- Computations do not always have to be **exact**.
- **Small** degradation in quality $\rightarrow$ **large** complexity savings.



Approximate Computing

Polar Decoding on **Faulty Hardware**

- Computations do not always have to be **exact**.
- **Small** degradation in quality $\rightarrow$ **large** complexity savings.

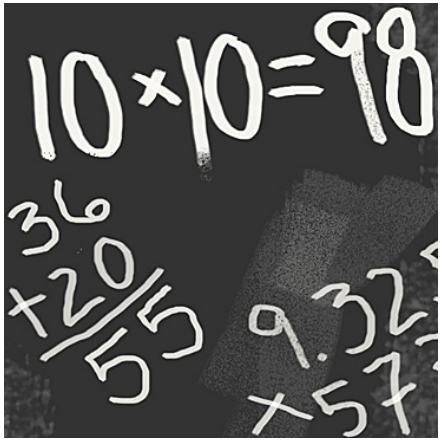


Unintentionally faulty **Intentionally approximate**

Approximate Computing

Polar Decoding on **Faulty Hardware**

- Computations do not always have to be **exact**.
- **Small** degradation in quality → **large** complexity savings.

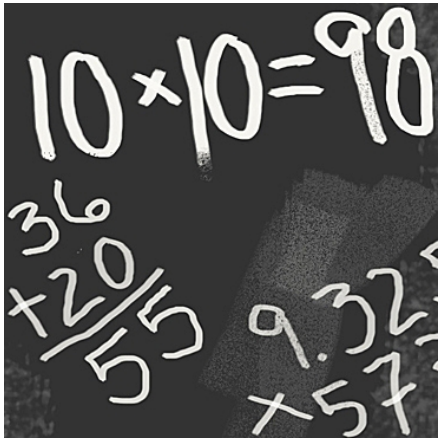


Unintentionally faulty	**Intentionally approximate**
Voltage scaling	
Parameter variations	
Radiation	

Approximate Computing

Polar Decoding on **Faulty Hardware**

- Computations do not always have to be **exact**.
- **Small** degradation in quality → **large** complexity savings.

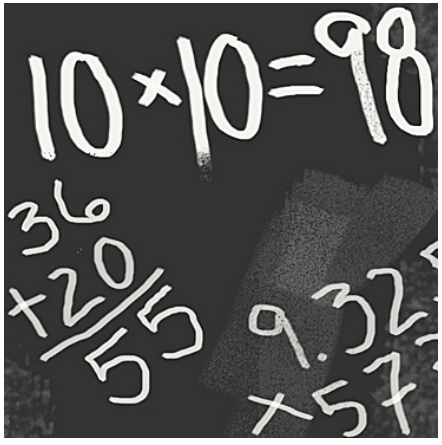


Unintentionally faulty	**Intentionally approximate**
Voltage scaling	Approximate adders/multipliers
Parameter variations	Quantization
Radiation	Algorithmic approximations

Approximate Computing

Polar Decoding on **Faulty Hardware**

- Computations do not always have to be **exact**.
- **Small** degradation in quality → **large** complexity savings.



Unintentionally faulty	**Intentionally approximate**
Voltage scaling	Approximate adders/multipliers
Parameter variations	Quantization
Radiation	Algorithmic approximations

Overview

1. Polar coding preliminaries
   - Polarizing transformation
   - Encoding
   - Successive cancellation decoding

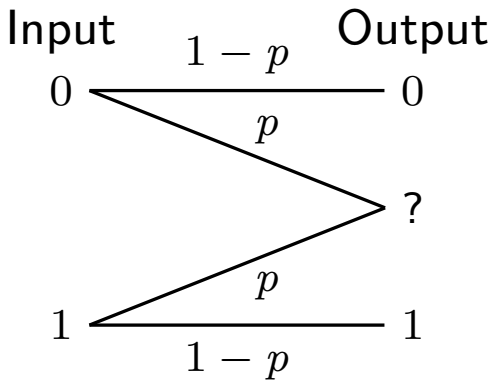
Overview

1. Polar coding preliminaries
   - Polarizing transformation
   - Encoding
   - Successive cancellation decoding

2. Polar decoding on faulty hardware
   - Hardware fault model
   - Negative analytical results
   - Efficient fault tolerance mechanisms

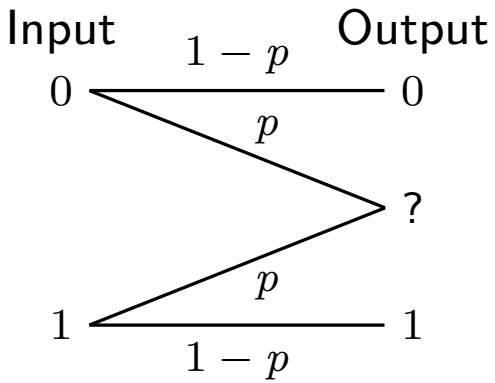
Preliminaries

- **Binary erasure channel (BEC):** a binary-input and ternary-output channel.



Preliminaries

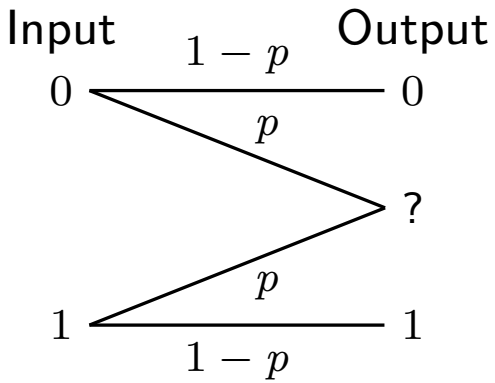
- **Binary erasure channel (BEC):** a binary-input and ternary-output channel.



- **Channel capacity:** the maximum rate at which information can be reliably transmitted over a communications channel.

Preliminaries

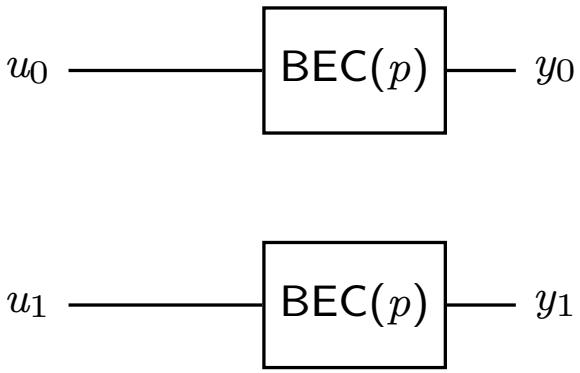
- **Binary erasure channel (BEC):** a binary-input and ternary-output channel.



- **Channel capacity:** the maximum rate at which information can be reliably transmitted over a communications channel.
  - **BEC capacity:** $1-p$ bits per channel use.

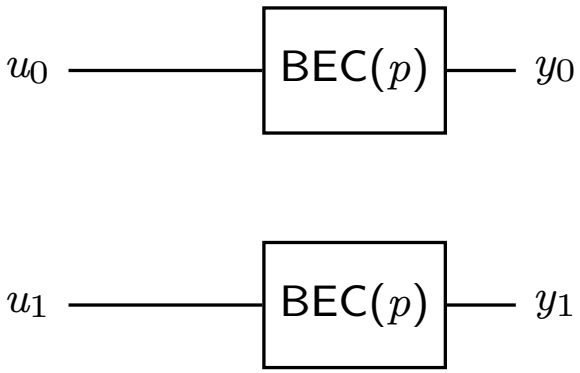
Basic Polarizing Transformation

- **Idea:** Transform N independent copies of a channel W into new channels.



Basic Polarizing Transformation

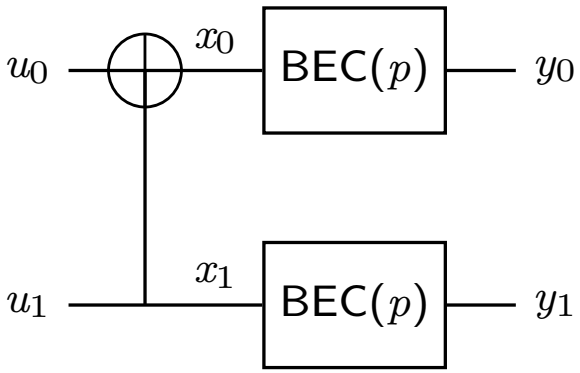
- **Idea:** Transform N independent copies of a channel W into new channels.



Bit erasure rate: $P_b = p$

Basic Polarizing Transformation

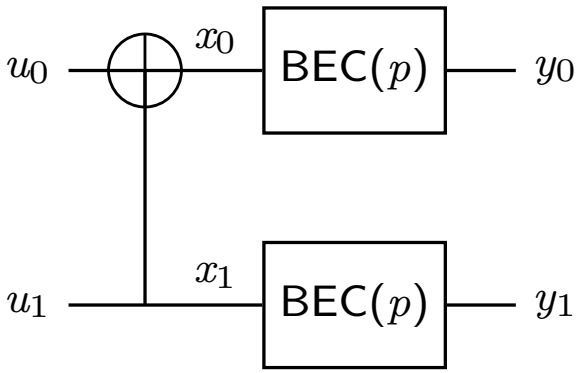
- **Idea:** Transform N independent copies of a channel W into new channels.



Bit erasure rate: $P_b =$???

Basic Polarizing Transformation

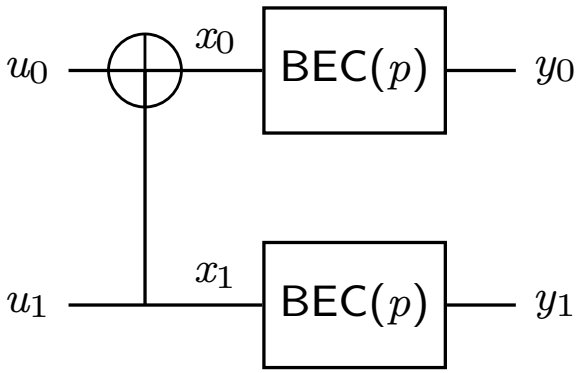
- **Idea:** Transform N independent copies of a channel W into new channels.



- Construct "synthetic" channels: $W_2^{(-)}(y_0, y_1|u_0)$ and $W_2^{(+)}(y_0, y_1, u_0|u_1)$

Basic Polarizing Transformation

- **Idea:** Transform N independent copies of a channel W into new channels.



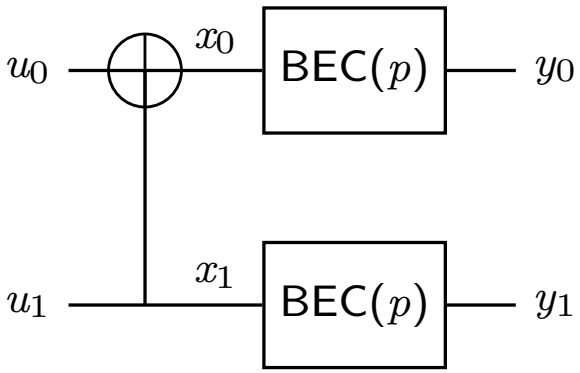
W^-:

y_0	y_1	$\hat{u}_0$
?	b	?
b	?	?
?	?	?

- Construct "synthetic" channels: $W_2^{(-)}(y_0, y_1|u_0)$ and $W_2^{(+)}(y_0, y_1, u_0|u_1)$
- **Bit erasure rate:**
  1. W^-:

Basic Polarizing Transformation

- **Idea:** Transform N independent copies of a channel W into new channels.



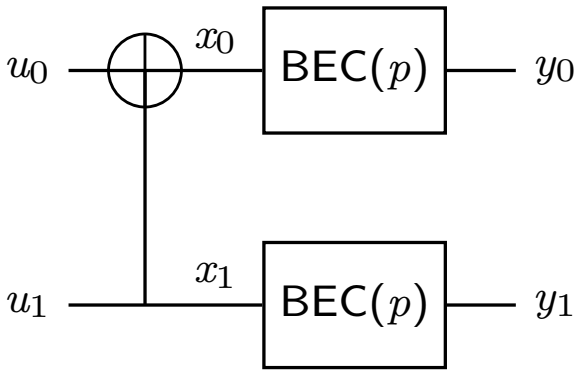
W^-:

y_0	y_1	$\hat{u}_0$
?	b	?
b	?	?
?	?	?

- Construct "synthetic" channels: $W_2^{(-)}(y_0, y_1|u_0)$ and $W_2^{(+)}(y_0, y_1, u_0|u_1)$
- **Bit erasure rate:**
  1. W^-: $P_b = 1 - (1-p)^2 = 2p - p^2 \geq p$

Basic Polarizing Transformation

- **Idea:** Transform N independent copies of a channel W into new channels.



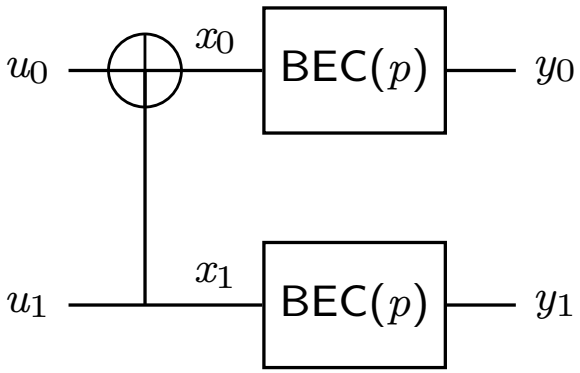
W^+:

y_0	y_1	$\hat{u}_1$
?	b	u_1
b	?	u_1
?	?	?

- Construct "synthetic" channels: $W_2^{(-)}(y_0, y_1|u_0)$ and $W_2^{(+)}(y_0, y_1, u_0|u_1)$
- **Bit erasure rate:**
  1. W^-: $P_b = 1 - (1-p)^2 = 2p - p^2 \geq p$
  2. W^+:

Basic Polarizing Transformation

- **Idea:** Transform N independent copies of a channel W into new channels.



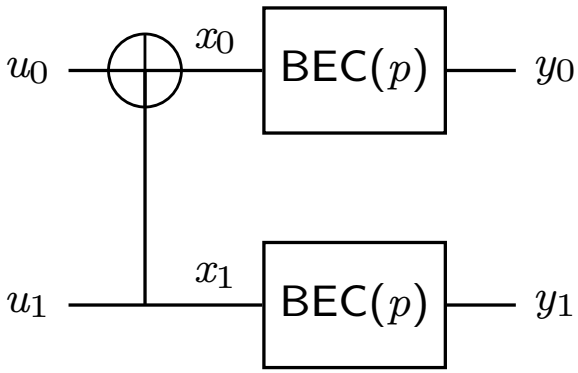
W^+:

y_0	y_1	$\hat{u}_1$
?	b	u_1
b	?	u_1
?	?	?

- Construct "synthetic" channels: $W_2^{(-)}(y_0, y_1|u_0)$ and $W_2^{(+)}(y_0, y_1, u_0|u_1)$
- **Bit erasure rate:**
  1. W^-: $P_b = 1 - (1-p)^2 = 2p - p^2 \geq p$
  2. W^+: $P_b = p^2 \leq p$

Basic Polarizing Transformation

- **Idea:** Transform N independent copies of a channel W into new channels.

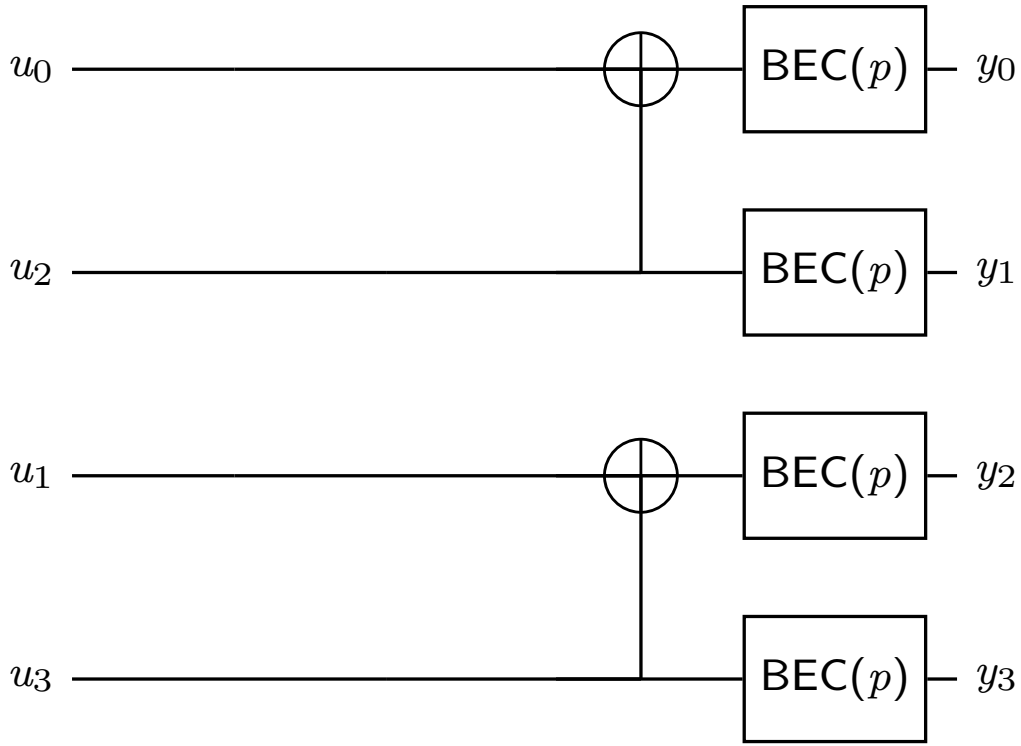


W^+:

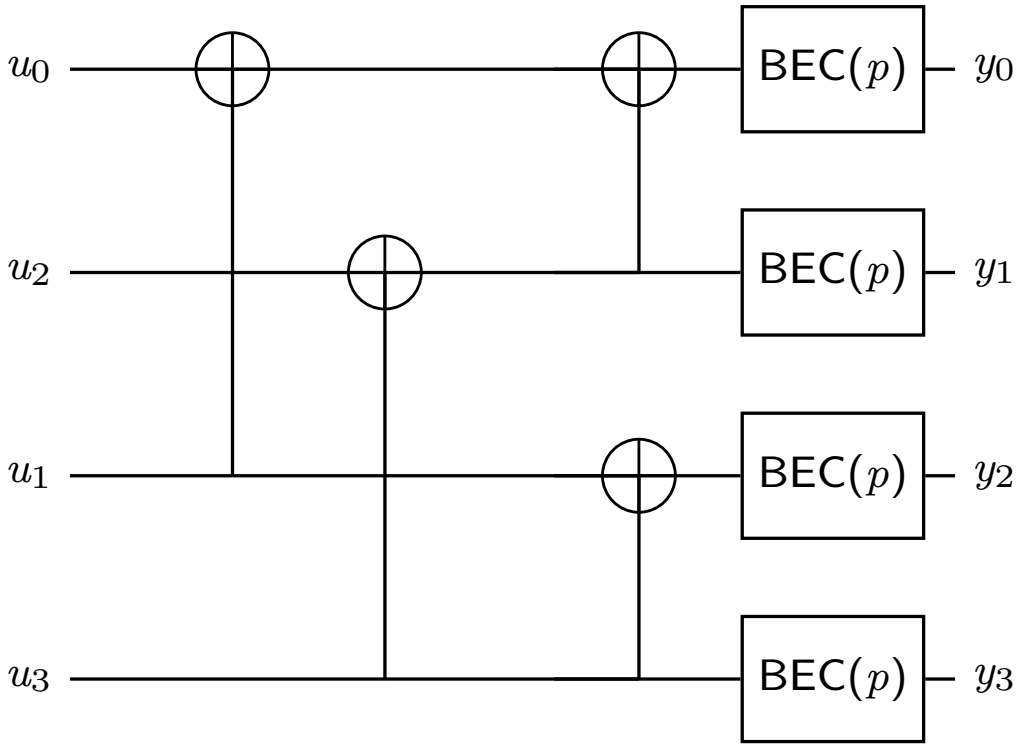
y_0	y_1	$\hat{u}_1$
?	b	u_1
b	?	u_1
?	?	?

- Construct "synthetic" channels: $W_2^{(-)}(y_0, y_1|u_0)$ and $W_2^{(+)}(y_0, y_1, u_0|u_1)$
- **Bit erasure rate:**
  1. W^-: $P_b = 1 - (1-p)^2 = 2p - p^2 \geq p$
  2. W^+: $P_b = p^2 \leq p$
- **Average bit erasure rate:** $\frac{1}{2}\left(2p - p^2 + p^2\right) = p.$

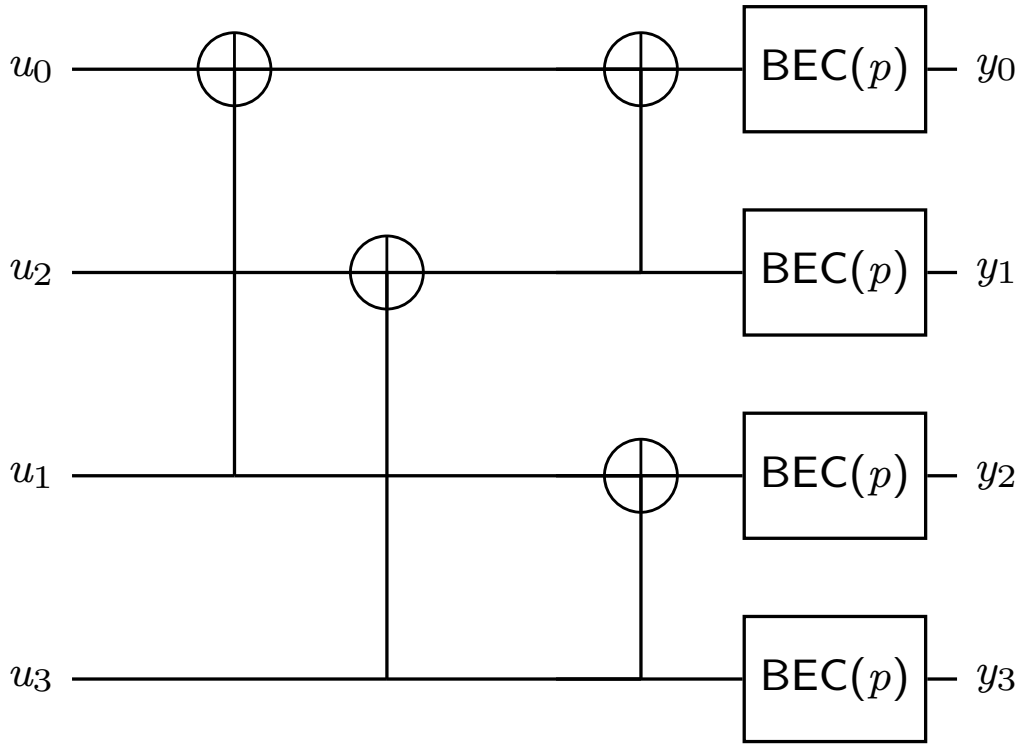
General Polarizing Transformation



General Polarizing Transformation

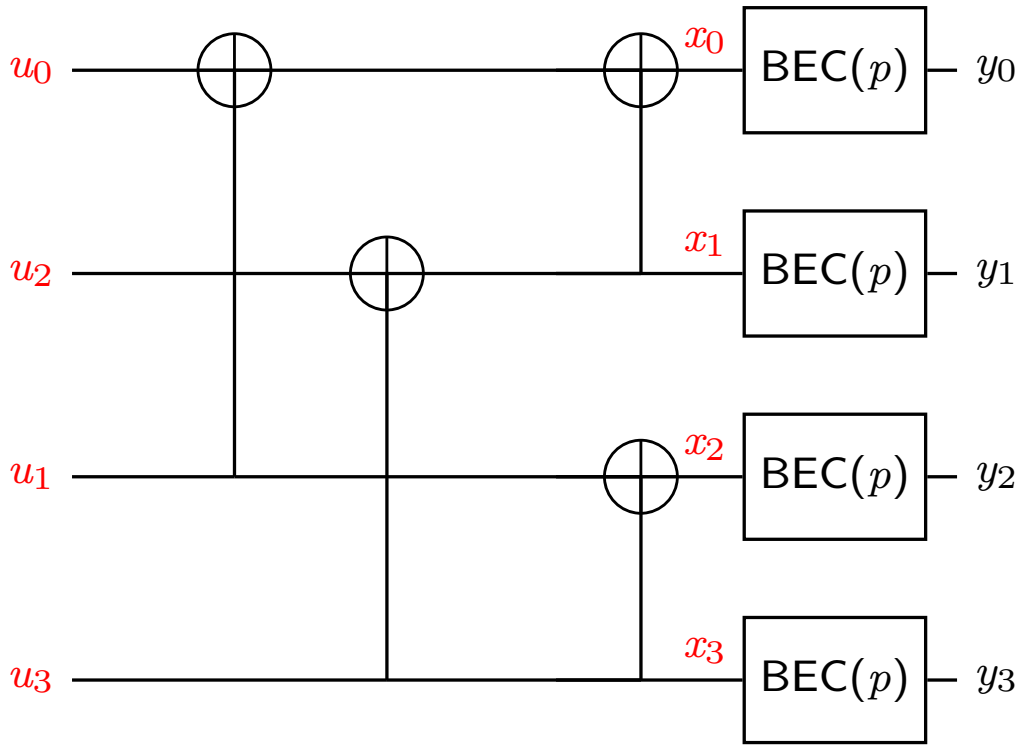


General Polarizing Transformation



- Construct four synthetic channels:
  1. $W_4^{(--)}(y_0^3|u_0)$ (obtained through $-$ transformation on $W_2^{(-)}$)
  2. $W_4^{(-+)}(y_0^3, u_0|u_1)$ (obtained through $+$ transformation on $W_2^{(-)}$)
  3. $W_4^{(+-)}(y_0^3, u_0^1|u_2)$ (obtained through $-$ transformation on $W_2^{(+)}$)
  4. $W_4^{(++)}(y_0^3, u_0^2|u_3)$ (obtained through $+$ transformation on $W_2^{(+)}$)

General Polarizing Transformation



- Transformation from $\mathbf{u} = [\, u_0 \;\; u_1 \;\; u_2 \;\; u_3 \,]$ to $\mathbf{x} = [\, x_0 \;\; x_1 \;\; x_2 \;\; x_3 \,]$ is:

$$\mathbf{x} = \mathbf{u} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

General Polarizing Transformation: Encoding

- For a polar code of length N the transformation from $\mathbf{u}$ to $\mathbf{x}$ is:

$$\mathbf{x} = \mathbf{u}\mathbf{G}_N, \quad \text{where} \quad \mathbf{G}_N = \mathbf{R}_N\mathbf{F}^{\otimes n}.$$

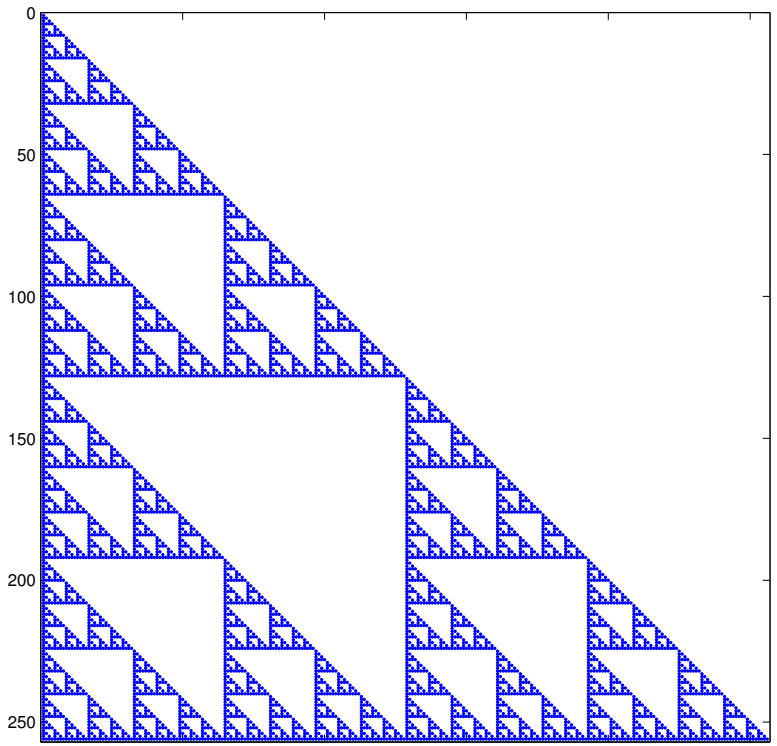
- $\mathbf{R}_N$ is a bit-reversal permutation matrix, $\mathbf{F} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$, and $n = \log_2 N$.

General Polarizing Transformation: Encoding

- For a polar code of length N the transformation from $\mathbf{u}$ to $\mathbf{x}$ is:

$$\mathbf{x} = \mathbf{u}\mathbf{G}_N, \quad \text{where} \quad \mathbf{G}_N = \mathbf{R}_N \mathbf{F}^{\otimes n}.$$

- $\mathbf{R}_N$ is a bit-reversal permutation matrix, $\mathbf{F} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$, and $n = \log_2 N$.



General Polarizing Transformation: Erasure rate

- The erasure probability of a synthetic channel $W_N^{(\mathbf{s})}$ is given by

$$T^{s_n}\left(\cdots T^{s_2}\left(T^{s_1}(p)\right)\right) \quad \text{where} \quad \mathbf{s} = [\, s_1 \ \ldots \ s_n \,] \in \{-,+\}$$

General Polarizing Transformation: Erasure rate

- The erasure probability of a synthetic channel $W_N^{(\mathbf{s})}$ is given by

$$T^{s_n}\left(\cdots T^{s_2}\left(T^{s_1}(p)\right)\right) \quad \text{where} \quad \mathbf{s} = [\, s_1 \ \ldots \ s_n \,] \in \{-,+\}$$

where

$$T^{+}(\epsilon) \triangleq \epsilon^2 \quad \text{and} \quad T^{-}(\epsilon) \triangleq 2\epsilon - \epsilon^2$$

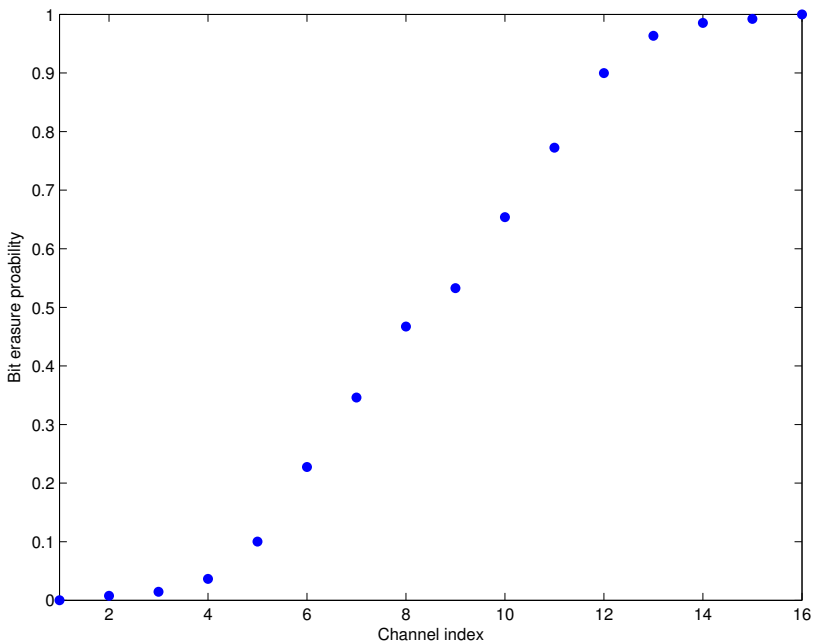
General Polarizing Transformation: Erasure rate

- The erasure probability of a synthetic channel $W_N^{(\mathbf{s})}$ is given by

$$T^{s_n}\left(\cdots T^{s_2}\left(T^{s_1}(p)\right)\right) \quad \text{where} \quad \mathbf{s} = [\, s_1 \ \ldots \ s_n \,] \in \{-,+\}$$

where

$$T^{+}(\epsilon) \triangleq \epsilon^2 \quad \text{and} \quad T^{-}(\epsilon) \triangleq 2\epsilon - \epsilon^2$$



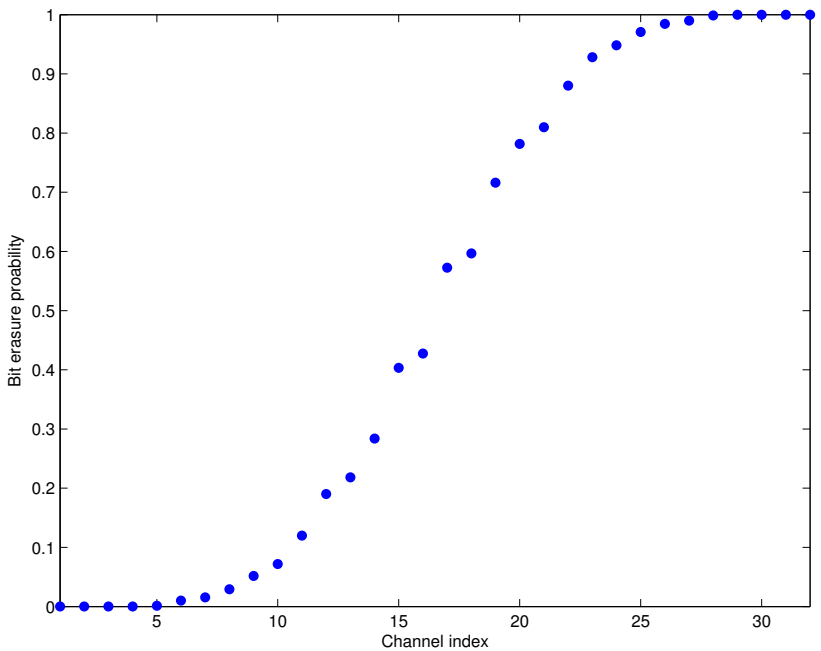
General Polarizing Transformation: Erasure rate

- The erasure probability of a synthetic channel $W_N^{(\mathbf{s})}$ is given by

$$T^{s_n}\left(\cdots T^{s_2}\left(T^{s_1}(p)\right)\right) \quad \text{where} \quad \mathbf{s} = [\, s_1 \ \ldots \ s_n \,] \in \{-,+\}$$

where

$$T^{+}(\epsilon) \triangleq \epsilon^2 \quad \text{and} \quad T^{-}(\epsilon) \triangleq 2\epsilon - \epsilon^2$$



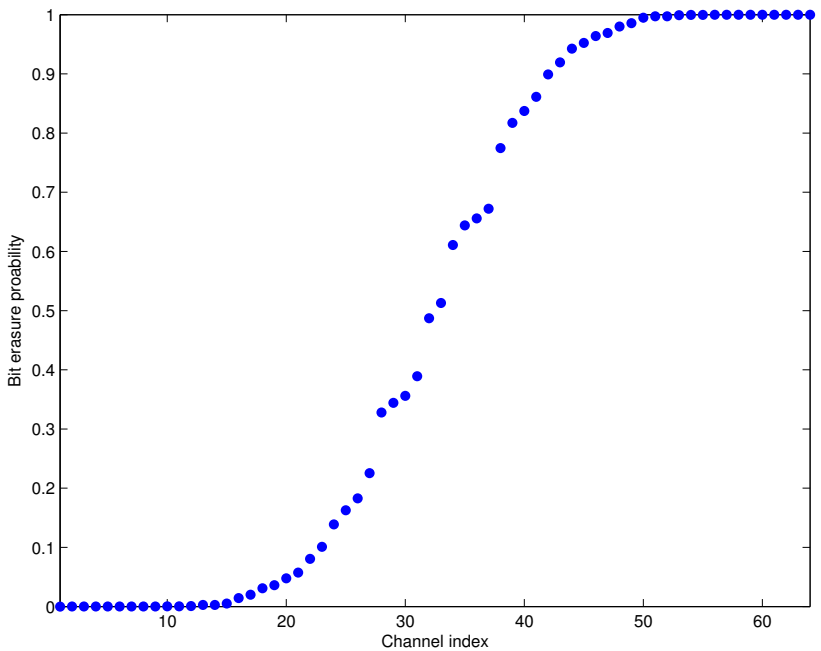
General Polarizing Transformation: Erasure rate

- The erasure probability of a synthetic channel $W_N^{(\mathbf{s})}$ is given by

$$T^{s_n}\left(\cdots T^{s_2}\left(T^{s_1}(p)\right)\right) \quad \text{where} \quad \mathbf{s} = [\, s_1 \ \ldots \ s_n \,] \in \{-,+\}$$

where

$$T^{+}(\epsilon) \triangleq \epsilon^2 \quad \text{and} \quad T^{-}(\epsilon) \triangleq 2\epsilon - \epsilon^2$$



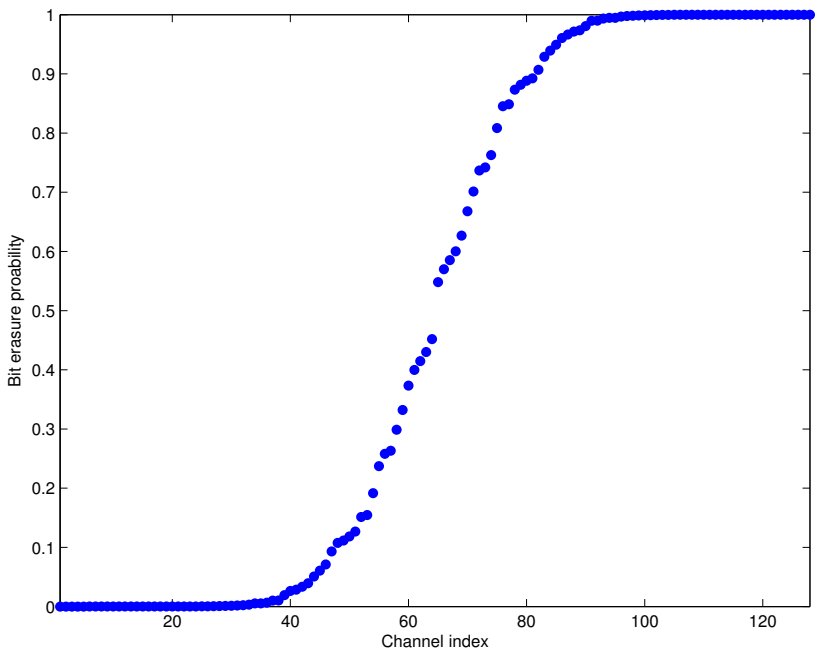
General Polarizing Transformation: Erasure rate

- The erasure probability of a synthetic channel $W_N^{(\mathbf{s})}$ is given by

$$T^{s_n}\left(\cdots T^{s_2}\left(T^{s_1}(p)\right)\right) \quad \text{where} \quad \mathbf{s} = [\, s_1 \ \ldots \ s_n \,] \in \{-,+\}$$

where

$$T^{+}(\epsilon) \triangleq \epsilon^2 \quad \text{and} \quad T^{-}(\epsilon) \triangleq 2\epsilon - \epsilon^2$$



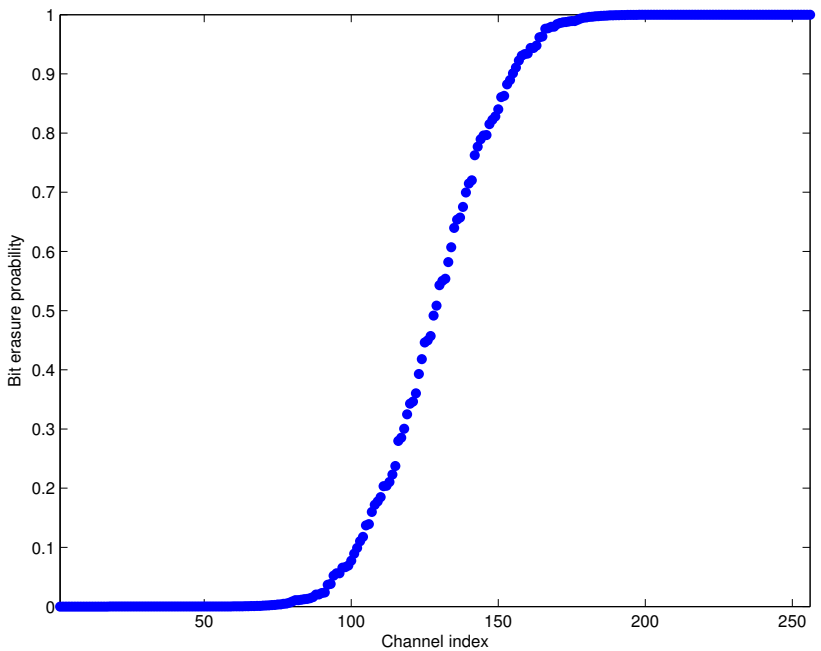
General Polarizing Transformation: Erasure rate

- The erasure probability of a synthetic channel $W_N^{(\mathbf{s})}$ is given by

$$T^{s_n}\left(\cdots T^{s_2}\left(T^{s_1}(p)\right)\right) \quad \text{where} \quad \mathbf{s} = [\, s_1 \ \ldots \ s_n \,] \in \{-,+\}$$

where

$$T^{+}(\epsilon) \triangleq \epsilon^2 \quad \text{and} \quad T^{-}(\epsilon) \triangleq 2\epsilon - \epsilon^2$$



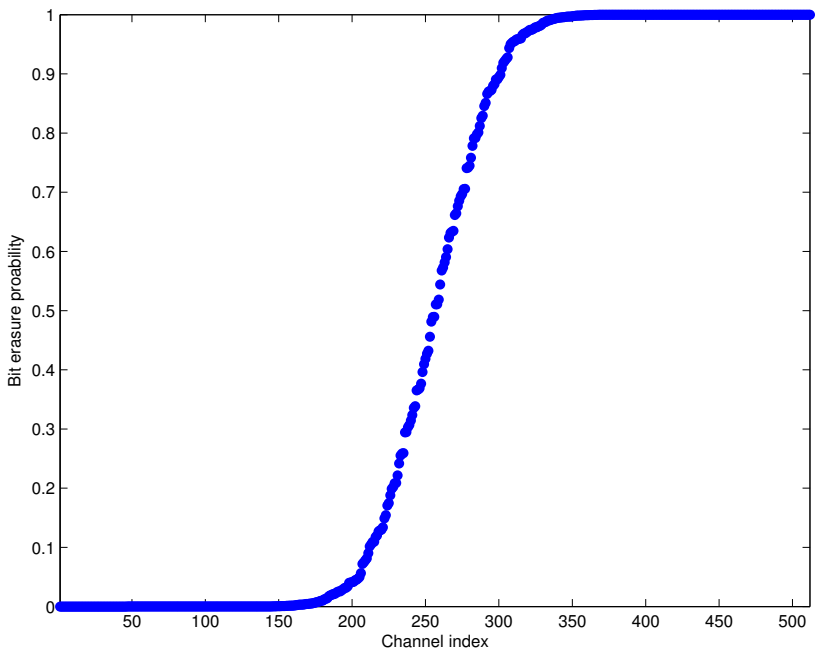
General Polarizing Transformation: Erasure rate

- The erasure probability of a synthetic channel $W_N^{(\mathbf{s})}$ is given by

$$T^{s_n}\left(\cdots T^{s_2}\left(T^{s_1}(p)\right)\right) \quad \text{where} \quad \mathbf{s} = [\, s_1 \ \ldots \ s_n \,] \in \{-,+\}$$

where

$$T^{+}(\epsilon) \triangleq \epsilon^2 \quad \text{and} \quad T^{-}(\epsilon) \triangleq 2\epsilon - \epsilon^2$$



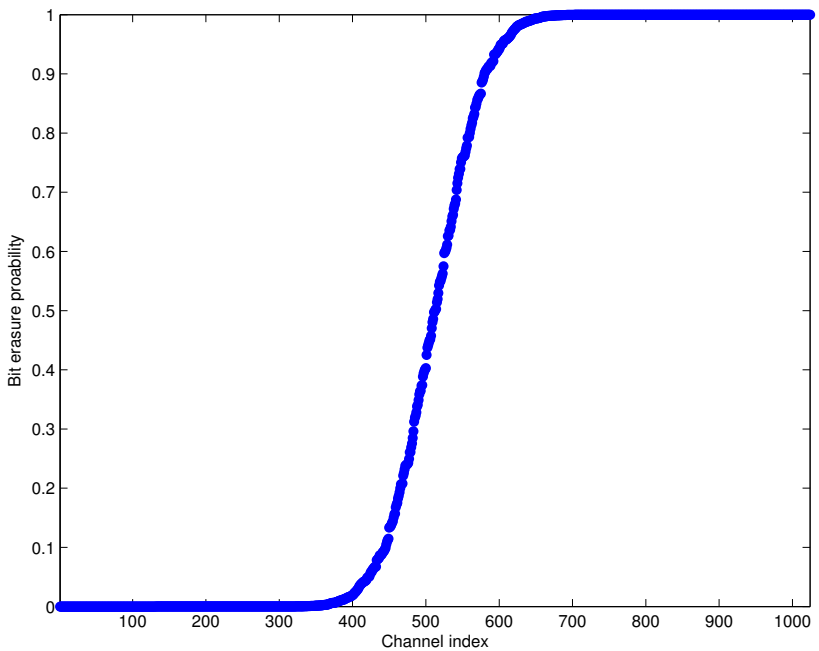
General Polarizing Transformation: Erasure rate

- The erasure probability of a synthetic channel $W_N^{(\mathbf{s})}$ is given by

$$T^{s_n}\left(\cdots T^{s_2}\left(T^{s_1}(p)\right)\right) \quad \text{where} \quad \mathbf{s} = [\, s_1 \ \ldots \ s_n \,] \in \{-,+\}$$

where

$$T^{+}(\epsilon) \triangleq \epsilon^2 \quad \text{and} \quad T^{-}(\epsilon) \triangleq 2\epsilon - \epsilon^2$$



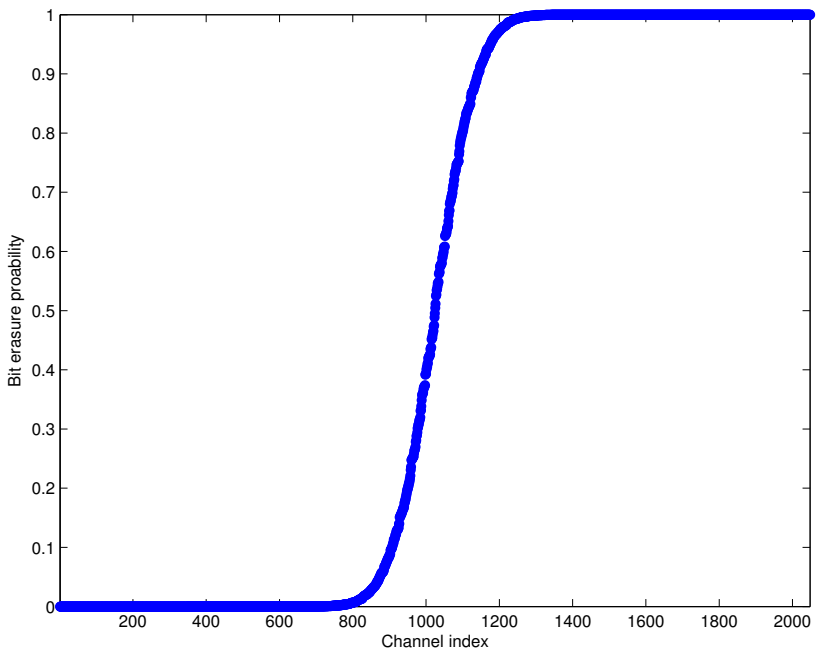
General Polarizing Transformation: Erasure rate

- The erasure probability of a synthetic channel $W_N^{(\mathbf{s})}$ is given by

$$T^{s_n}\left(\cdots T^{s_2}\left(T^{s_1}(p)\right)\right) \quad \text{where} \quad \mathbf{s} = [\, s_1 \ \ldots \ s_n \,] \in \{-,+\}$$

where

$$T^{+}(\epsilon) \triangleq \epsilon^2 \quad \text{and} \quad T^{-}(\epsilon) \triangleq 2\epsilon - \epsilon^2$$



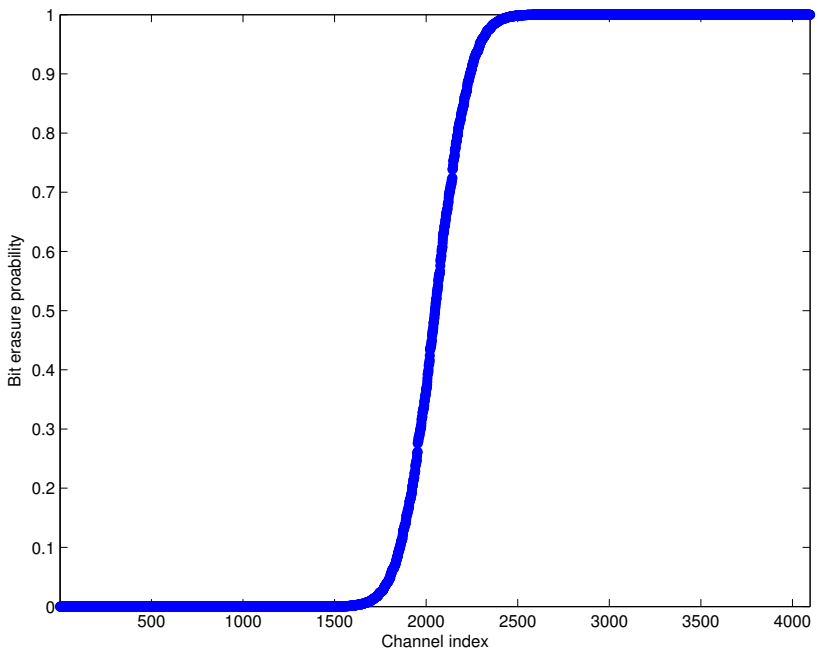
General Polarizing Transformation: Erasure rate

- The erasure probability of a synthetic channel $W_N^{(\mathbf{s})}$ is given by

$$T^{s_n}\left(\cdots T^{s_2}\left(T^{s_1}(p)\right)\right) \quad \text{where} \quad \mathbf{s} = [\, s_1 \ \ldots \ s_n \,] \in \{-,+\}$$

where

$$T^{+}(\epsilon) \triangleq \epsilon^2 \quad \text{and} \quad T^{-}(\epsilon) \triangleq 2\epsilon - \epsilon^2$$



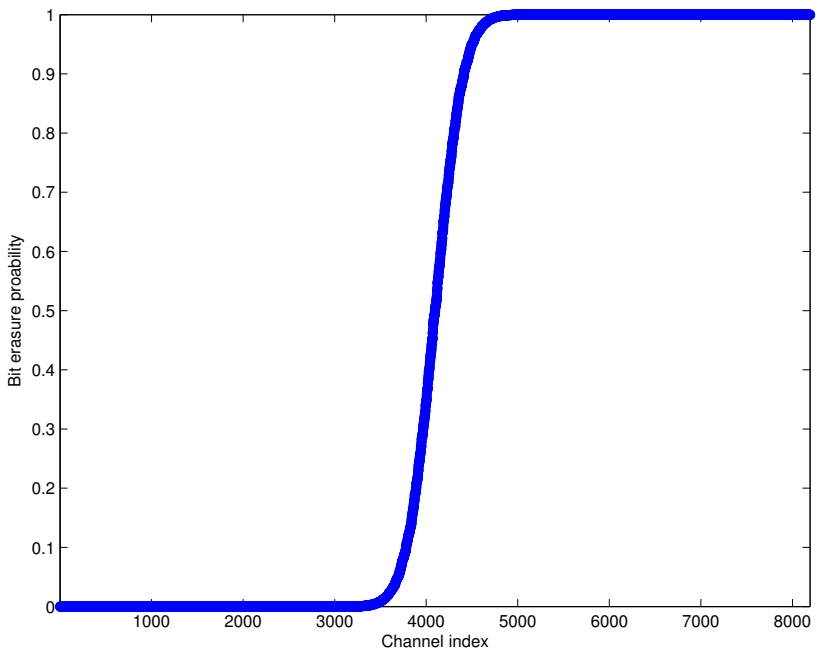
General Polarizing Transformation: Erasure rate

- The erasure probability of a synthetic channel $W_N^{(\mathbf{s})}$ is given by

$$T^{s_n}\left(\cdots T^{s_2}\left(T^{s_1}(p)\right)\right) \quad \text{where} \quad \mathbf{s} = [\, s_1 \ \ldots \ s_n \,] \in \{-,+\}$$

where

$$T^{+}(\epsilon) \triangleq \epsilon^2 \quad \text{and} \quad T^{-}(\epsilon) \triangleq 2\epsilon - \epsilon^2$$



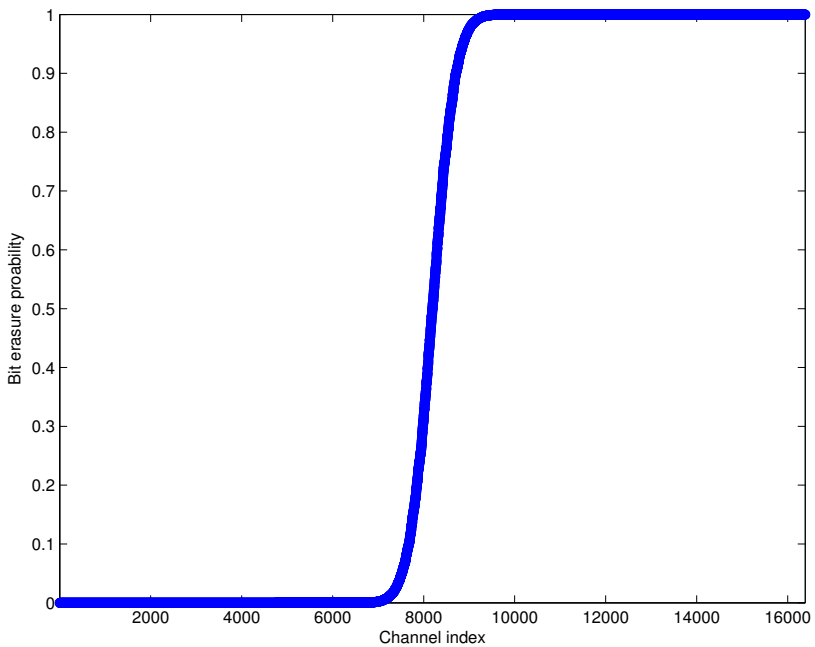
General Polarizing Transformation: Erasure rate

- The erasure probability of a synthetic channel $W_N^{(\mathbf{s})}$ is given by

$$T^{s_n}\left(\cdots T^{s_2}\left(T^{s_1}(p)\right)\right) \quad \text{where} \quad \mathbf{s} = [\, s_1 \ \ldots \ s_n \,] \in \{-,+\}$$

where

$$T^{+}(\epsilon) \triangleq \epsilon^2 \quad \text{and} \quad T^{-}(\epsilon) \triangleq 2\epsilon - \epsilon^2$$



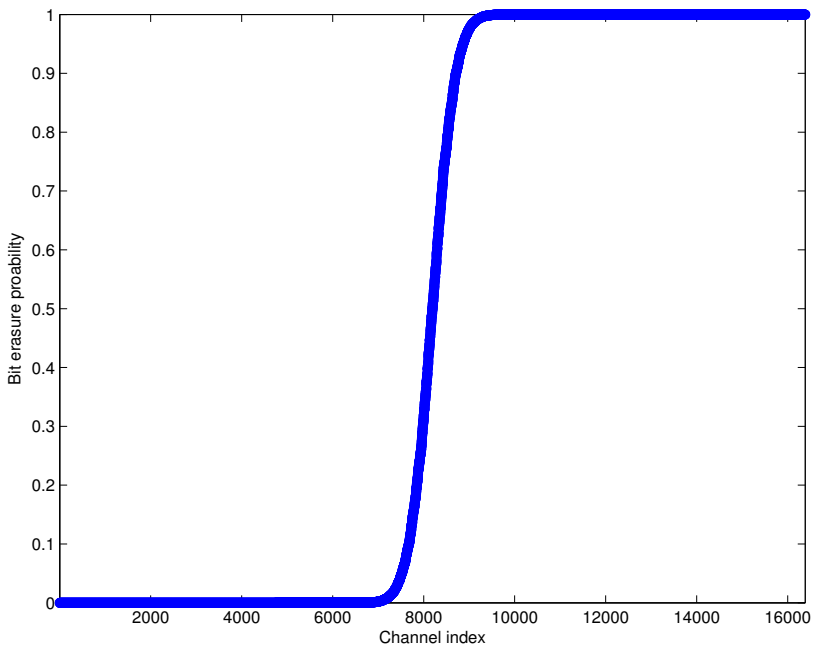
General Polarizing Transformation: Erasure rate

- The erasure probability of a synthetic channel $W_N^{(\mathbf{s})}$ is given by

$$T^{s_n}\left(\cdots T^{s_2}\left(T^{s_1}(p)\right)\right) \quad \text{where} \quad \mathbf{s} = [\, s_1 \ \ldots \ s_n \,] \in \{-,+\}$$

where

$$T^{+}(\epsilon) \triangleq \epsilon^2 \quad \text{and} \quad T^{-}(\epsilon) \triangleq 2\epsilon - \epsilon^2$$



Synthetic channels **polarize** $\rightarrow$ **Polar codes**

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Arıkan's Polar Coding Paradigm

$\mathcal{A} = \{3, 5, 6, 7\}$

0
1
2
3
4
5
6
7

Construction:

- Information indices: $\mathcal{A} \subset \{0, 1, \ldots, N-1\}$, $|\mathcal{A}| = NR$, $N = 2^n$.

Arıkan's Polar Coding Paradigm

$$\mathbf{u} = \begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \end{bmatrix}$$

Construction:

- Information indices: $\mathcal{A} \subset \{0, 1, \ldots, N-1\}$, $|\mathcal{A}| = NR$, $N = 2^n$.

Encoding:

- $\mathbf{u}_{\mathcal{A}} \triangleq [u_i, i \in \mathcal{A}]^T \leftarrow$ data bits

Arıkan's Polar Coding Paradigm

$$\mathbf{u} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ u_3 \\ 0 \\ u_5 \\ u_6 \\ u_7 \end{bmatrix}$$

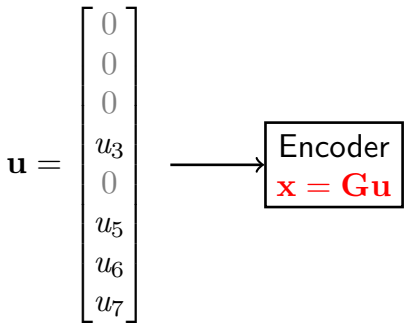
Construction:

- Information indices: $\mathcal{A} \subset \{0, 1, \ldots, N-1\}$, $|\mathcal{A}| = NR$, $N = 2^n$.

Encoding:

- $\mathbf{u}_{\mathcal{A}} \triangleq [u_i, i \in \mathcal{A}]^T \leftarrow$ data bits
- $\mathbf{u}_{\mathcal{A}^C} \leftarrow$ known-to-receiver frozen bits (say all-zero)

Arıkan's Polar Coding Paradigm



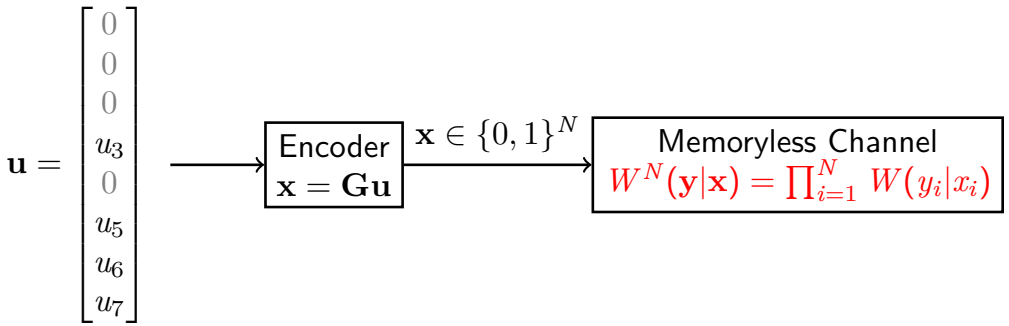
Construction:

- Information indices: $\mathcal{A} \subset \{0, 1, \ldots, N-1\}$, $|\mathcal{A}| = NR$, $N = 2^n$.

Encoding:

- $\mathbf{u}_{\mathcal{A}} \triangleq [u_i, i \in \mathcal{A}]^T \leftarrow$ data bits
- $\mathbf{u}_{\mathcal{A}^C} \leftarrow$ known-to-receiver frozen bits (say all-zero)
- $\mathbf{x} \leftarrow \mathbf{Gu}$

Arıkan's Polar Coding Paradigm



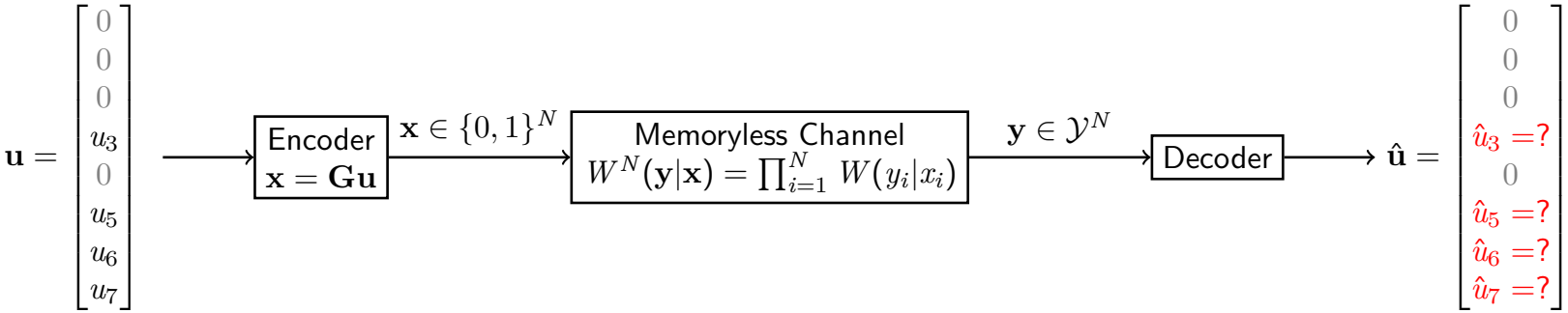
Construction:

- Information indices: $\mathcal{A} \subset \{0, 1, \ldots, N-1\}$, $|\mathcal{A}| = NR$, $N = 2^n$.

Encoding:

- $\mathbf{u}_{\mathcal{A}} \triangleq [u_i, i \in \mathcal{A}]^T \leftarrow$ data bits
- $\mathbf{u}_{\mathcal{A}^C} \leftarrow$ known-to-receiver frozen bits (say all-zero)
- $\mathbf{x} \leftarrow \mathbf{G}\mathbf{u}$

Arıkan's Polar Coding Paradigm



Construction:

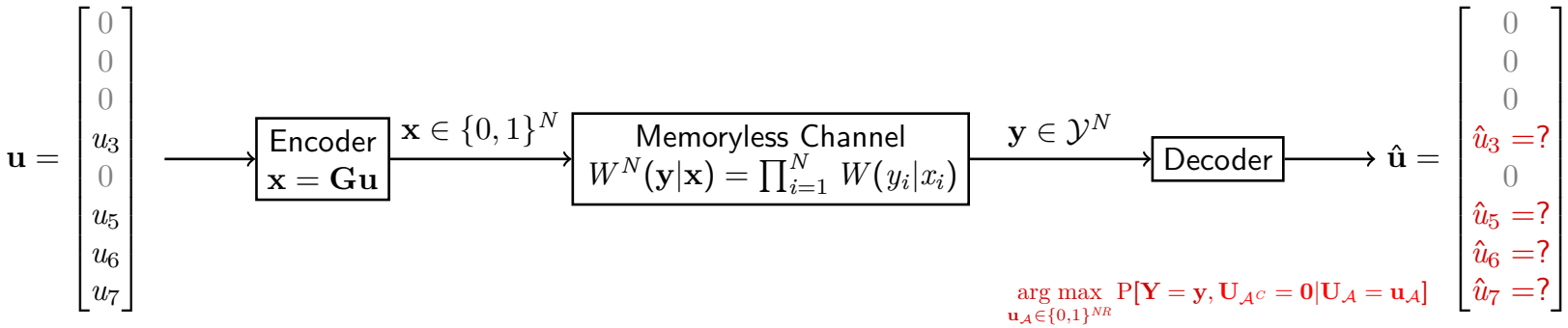
- Information indices: $\mathcal{A} \subset \{0, 1, \ldots, N-1\}$, $|\mathcal{A}| = NR$, $N = 2^n$.

Encoding:

- $\mathbf{u}_{\mathcal{A}} \triangleq [u_i, i \in \mathcal{A}]^T \leftarrow$ data bits
- $\mathbf{u}_{\mathcal{A}^C} \leftarrow$ known-to-receiver frozen bits (say all-zero)
- $\mathbf{x} \leftarrow \mathbf{Gu}$

Decoding: Estimate the information bits $\hat{\mathbf{u}}_{\mathcal{A}}$.

Arıkan's Polar Coding Paradigm



Construction:

- Information indices: $\mathcal{A} \subset \{0, 1, \ldots, N-1\}$, $|\mathcal{A}| = NR$, $N = 2^n$.

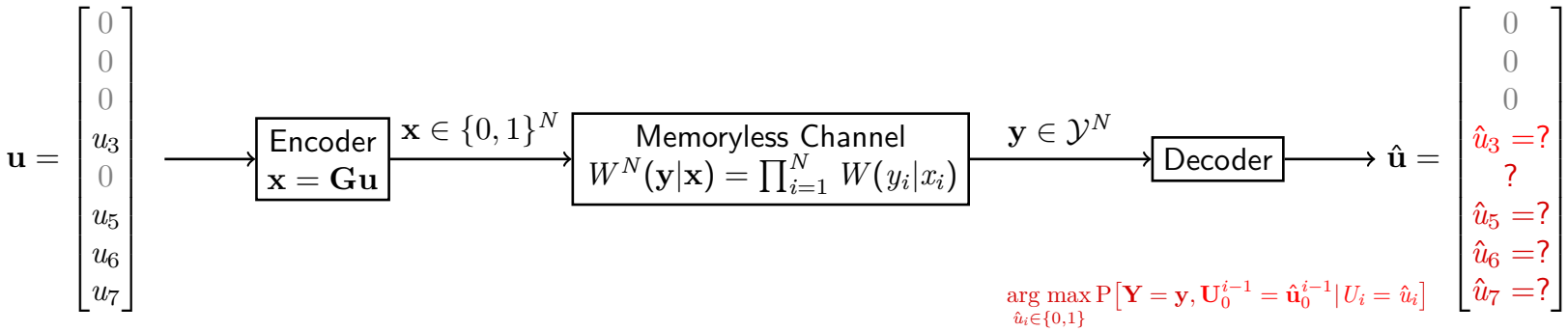
Encoding:

- $\mathbf{u}_\mathcal{A} \triangleq [u_i, i \in \mathcal{A}]^T \leftarrow$ data bits
- $\mathbf{u}_{\mathcal{A}^C} \leftarrow$ known-to-receiver frozen bits (say all-zero)
- $\mathbf{x} \leftarrow \mathbf{Gu}$

Decoding: Estimate the information bits $\hat{\mathbf{u}}_\mathcal{A}$.

- ML Decoding: Consider all 2^{NR} codewords and choose the best $\rightarrow$ **Too complex!**

Arıkan's Polar Coding Paradigm



Construction:

- Information indices: $\mathcal{A} \subset \{0, 1, \ldots, N-1\}$, $|\mathcal{A}| = NR$, $N = 2^n$.

Encoding:

- $\mathbf{u}_{\mathcal{A}} \triangleq [u_i, i \in \mathcal{A}]^T \leftarrow$ data bits
- $\mathbf{u}_{\mathcal{A}^C} \leftarrow$ known-to-receiver frozen bits (say all-zero)
- $\mathbf{x} \leftarrow \mathbf{Gu}$

Decoding: Estimate the information bits $\hat{\mathbf{u}}_{\mathcal{A}}$.

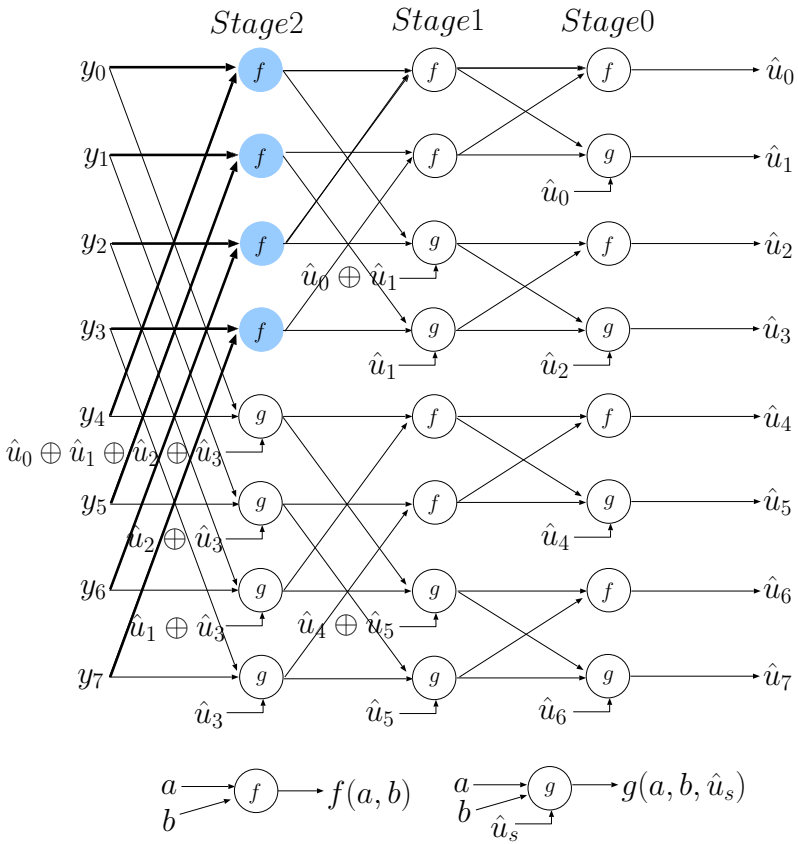
- ML Decoding: Consider all 2^{NR} codewords and choose the best $\rightarrow$ **Too complex!**
- (Suboptimal) Successive Cancellation (SC) Decoding:

$$\hat{u}_i \leftarrow \begin{cases} 0 & , i \notin \mathcal{A} \\ \arg\max_{u_i \in \{0,1\}} W_n^{(i)}(\mathbf{y}, \hat{\mathbf{u}}^{i-1} | u_i) & , i \in \mathcal{A} \end{cases}$$

Successive Cancellation Decoding

Data dependency graph for computing $W_n^{(i)}(\mathbf{y}, \hat{\mathbf{u}}^{i-1}|u_i)$

- Two types of nodes: f nodes and g nodes
- Processing starts from the left and proceeds in **topological order**

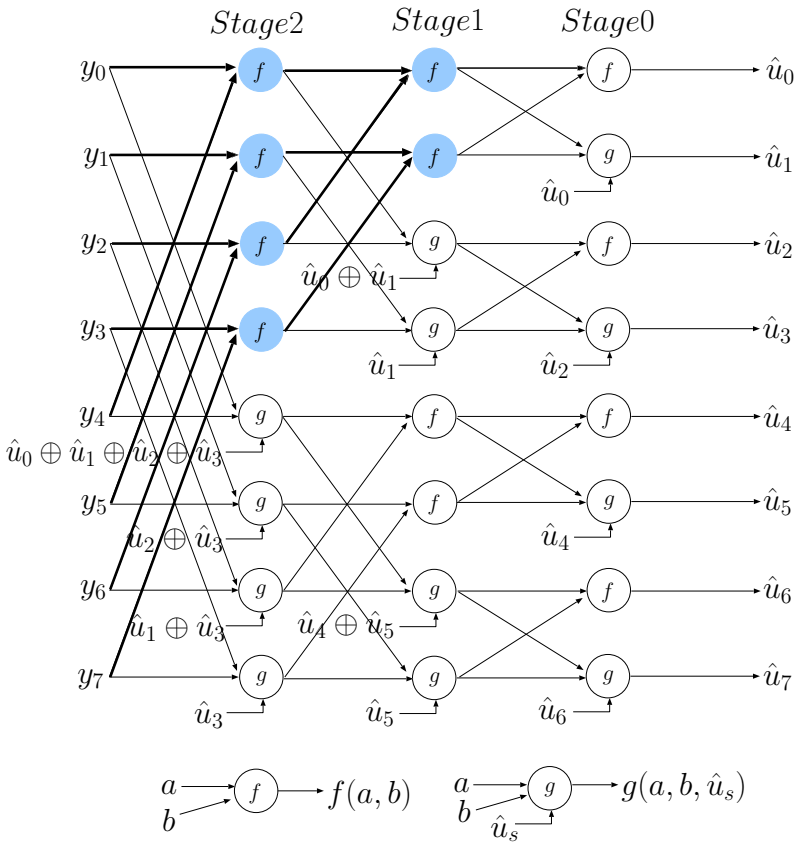


- No partial sums are needed to decode $\hat{u}_0$.

Successive Cancellation Decoding

Data dependency graph for computing $W_n^{(i)}(\mathbf{y}, \hat{\mathbf{u}}^{i-1}|u_i)$

- Two types of nodes: f nodes and g nodes
- Processing starts from the left and proceeds in **topological order**

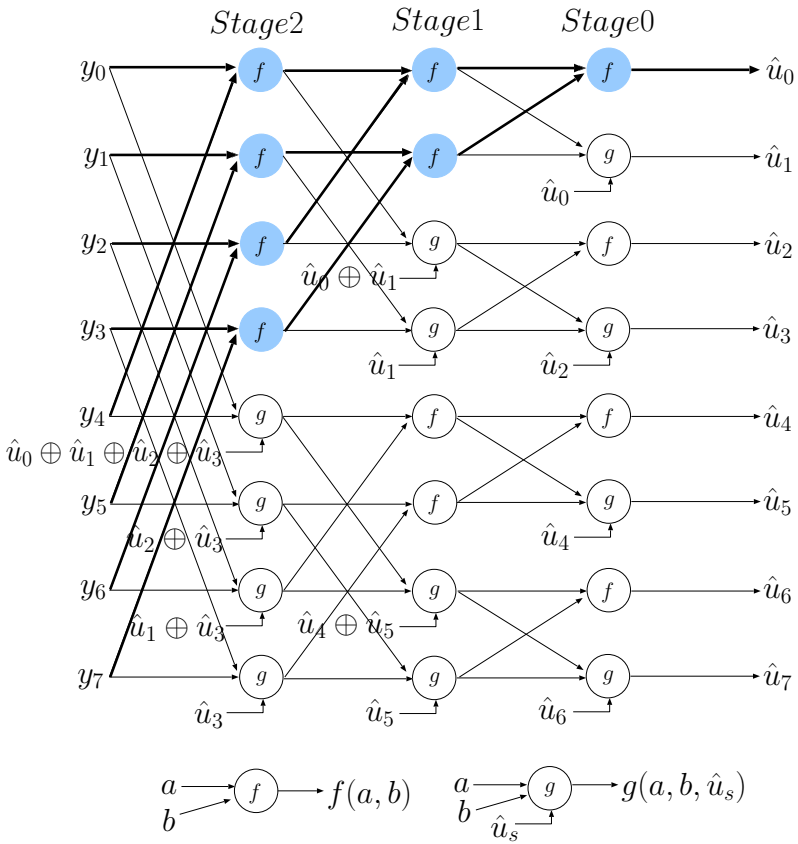


- No partial sums are needed to decode $\hat{u}_0$.

Successive Cancellation Decoding

Data dependency graph for computing $W_n^{(i)}(\mathbf{y}, \hat{\mathbf{u}}^{i-1}|u_i)$

- Two types of nodes: f nodes and g nodes
- Processing starts from the left and proceeds in **topological order**

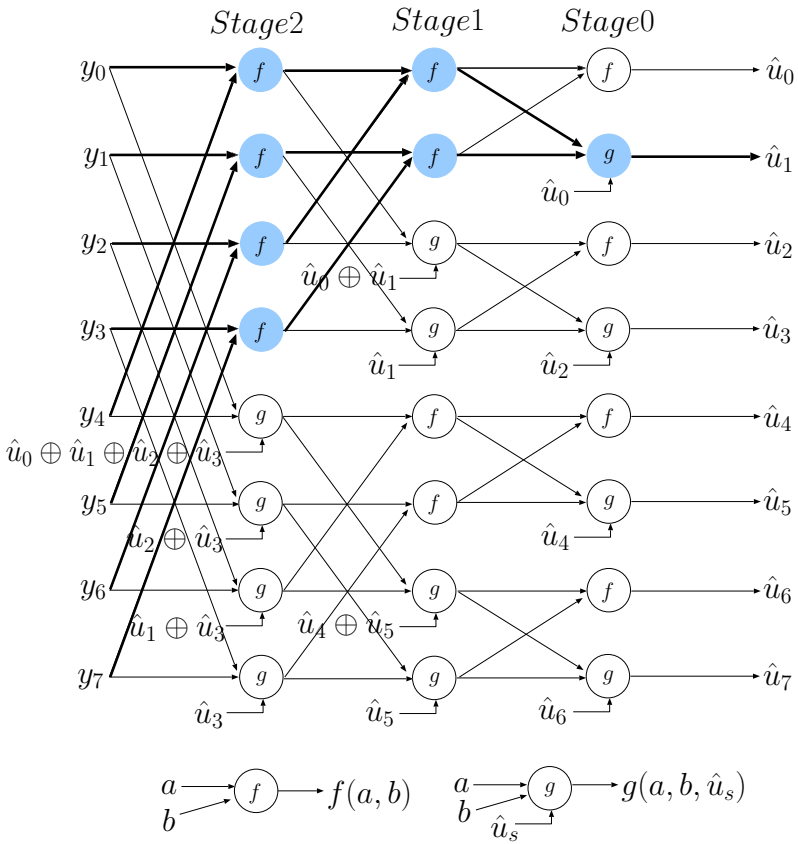


- No partial sums are needed to decode $\hat{u}_0$.

Successive Cancellation Decoding

Data dependency graph for computing $W_n^{(i)}(\mathbf{y}, \hat{\mathbf{u}}^{i-1}|u_i)$

- Two types of nodes: f nodes and g nodes
- Processing starts from the left and proceeds in **topological order**

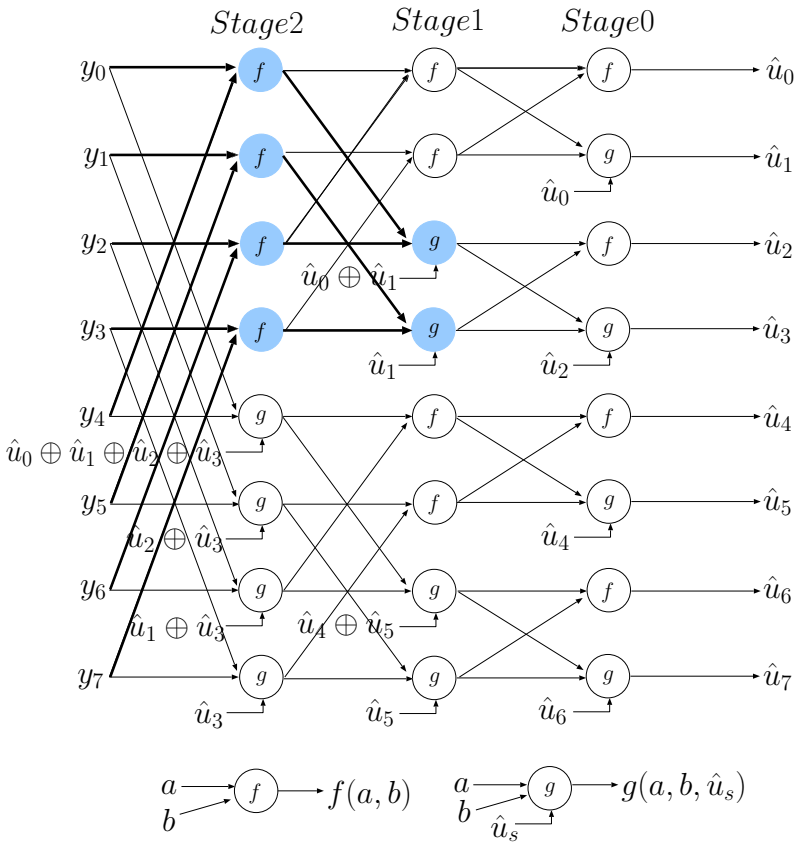


- Partial sum $\hat{u}_0$ is know, proceed decoding $\hat{u}_1$.

Successive Cancellation Decoding

Data dependency graph for computing $W_n^{(i)}(\mathbf{y}, \hat{\mathbf{u}}^{i-1}|u_i)$

- Two types of nodes: f nodes and g nodes
- Processing starts from the left and proceeds in **topological order**

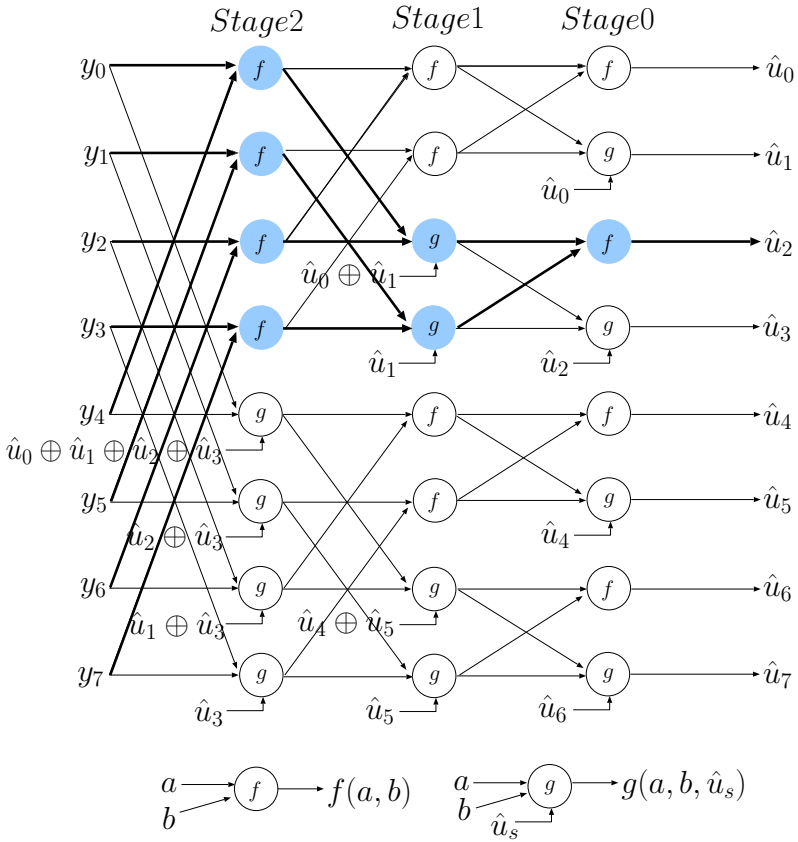


- Partial sums $\hat{u}_0 \oplus \hat{u}_1$ and $\hat{u}_1$ are known, proceed decoding $\hat{u}_2$.

Successive Cancellation Decoding

Data dependency graph for computing $W_n^{(i)}(\mathbf{y}, \hat{\mathbf{u}}^{i-1}|u_i)$

- Two types of nodes: f nodes and g nodes
- Processing starts from the left and proceeds in **topological order**

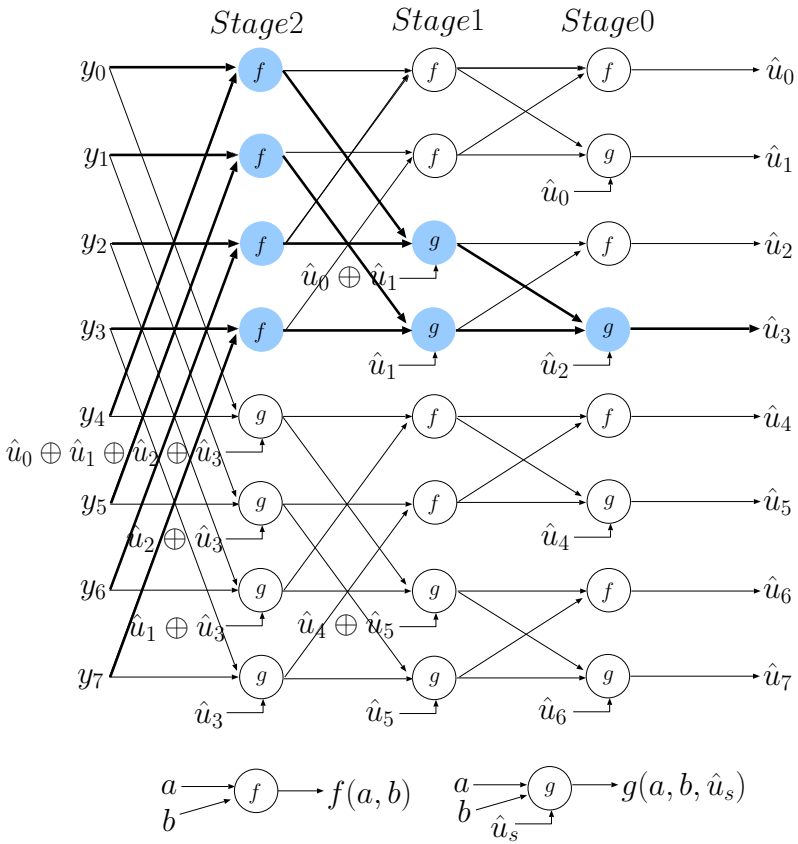


- Partial sums $\hat{u}_0 \oplus \hat{u}_1$ and $\hat{u}_1$ are known, proceed decoding $\hat{u}_2$.

Successive Cancellation Decoding

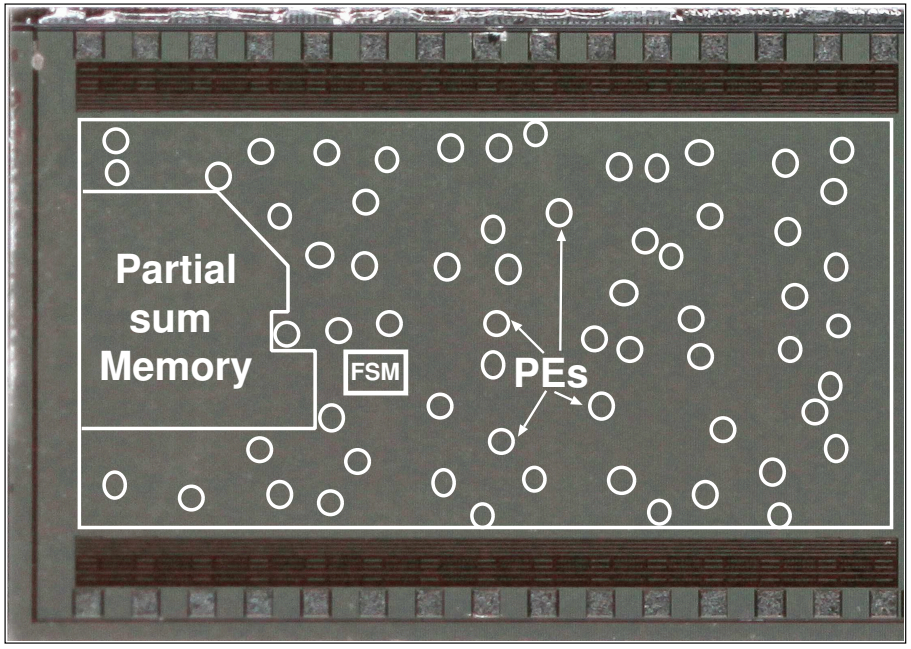
Data dependency graph for computing $W_n^{(i)}(\mathbf{y}, \hat{\mathbf{u}}^{i-1}|u_i)$

- Two types of nodes: f nodes and g nodes
- Processing starts from the left and proceeds in **topological order**



- Partial sum $\hat{u}_2$ is known, proceed decoding $\hat{u}_3$.

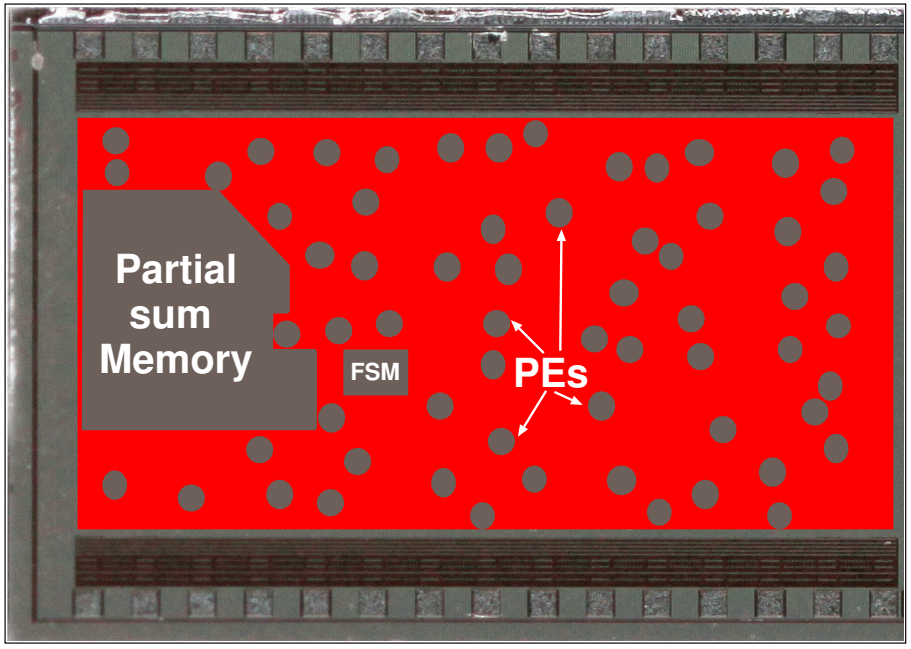
Successive Cancellation Decoder in 180nm CMOS



Successive Cancellation Decoder in 180nm CMOS

LLR memory area:

- Area: 1.35 mm^2
- Gate count: 144k
- 78% of the chip area



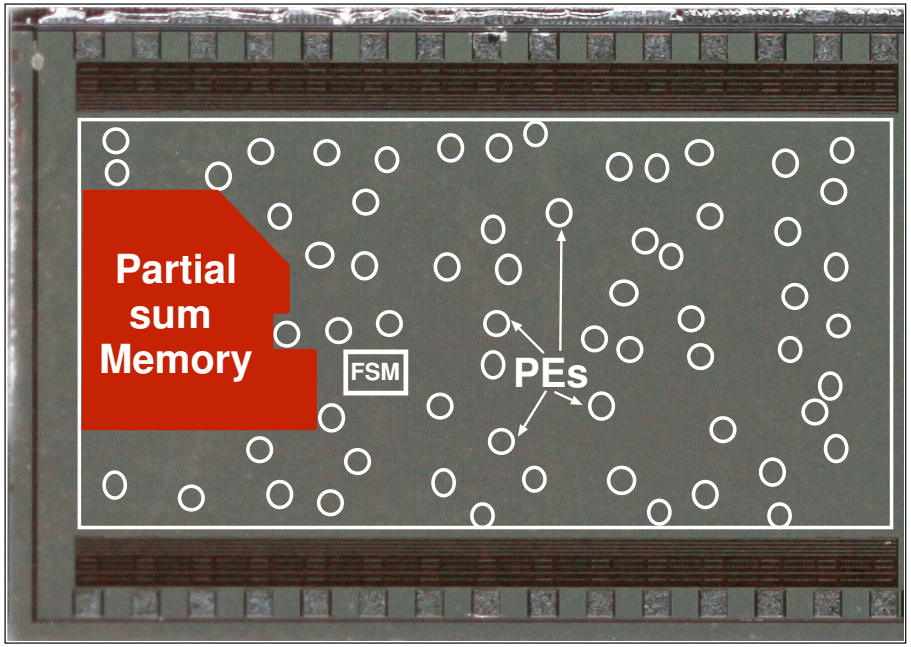
Successive Cancellation Decoder in 180nm CMOS

LLR memory area:

- Area: 1.35 mm^2
- Gate count: 144k
- 78% of the chip area

Partial sum memory area:

- Area: 0.22 mm^2
- Gate count: 24k
- 13% of the chip area



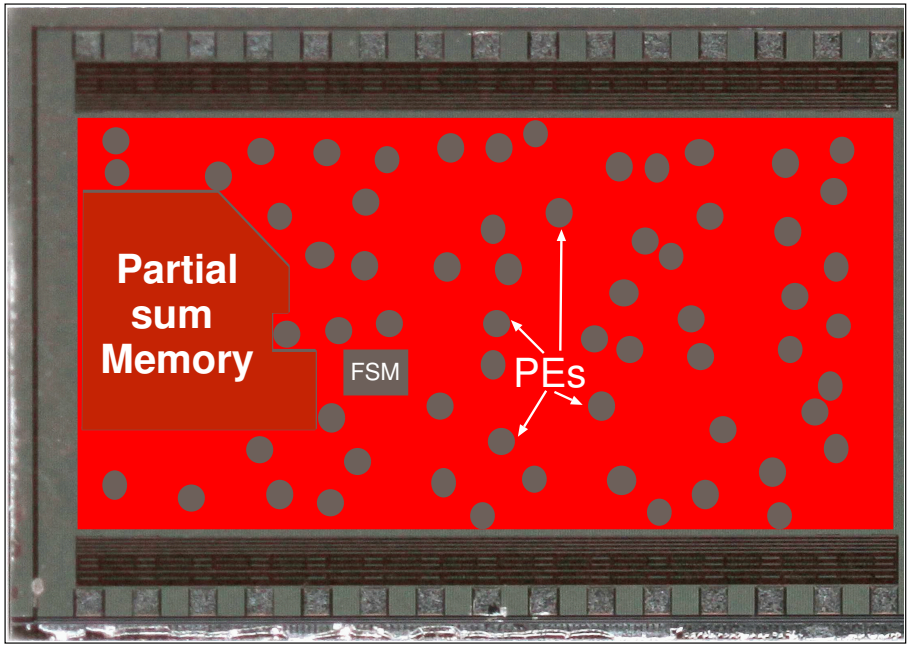
Successive Cancellation Decoder in 180nm CMOS

LLR memory area:

- Area: 1.35 mm^2
- Gate count: 144k
- 78% of the chip area

Partial sum memory area:

- Area: 0.22 mm^2
- Gate count: 24k
- 13% of the chip area



Memory is **more than 91%** of the total area.

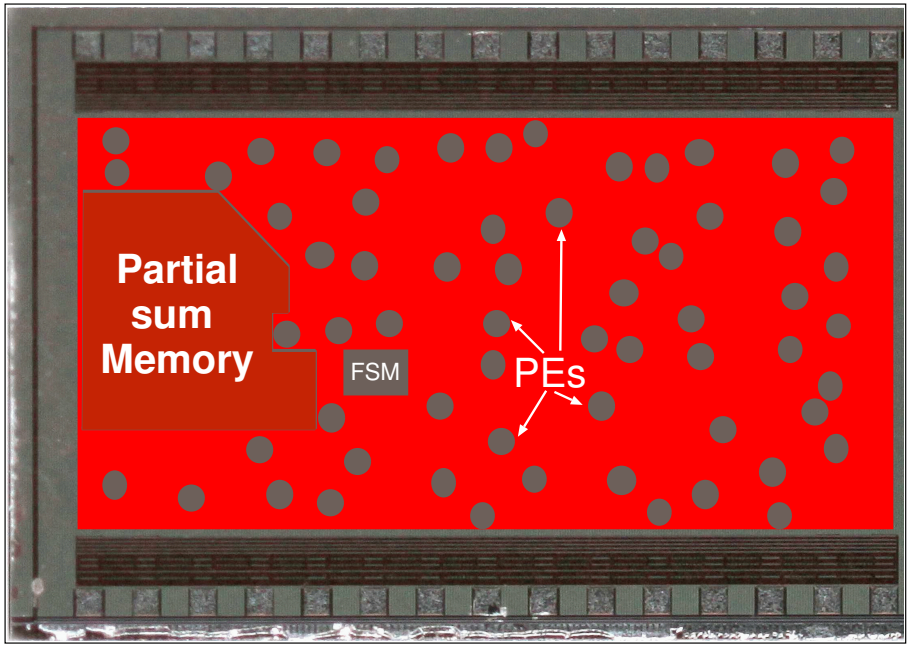
Successive Cancellation Decoder in 180nm CMOS

LLR memory area:

- Area: 1.35 mm^2
- Gate count: 144k
- 78% of the chip area

Partial sum memory area:

- Area: 0.22 mm^2
- Gate count: 24k
- 13% of the chip area



Memory is **more than 91%** of the total area.

Our focus: reliability of FEC decoders

1. Polar codes under **successive cancellation decoding** with faulty memory.
2. LDPC codes under **quantized min-sum decoding** with faulty memory.

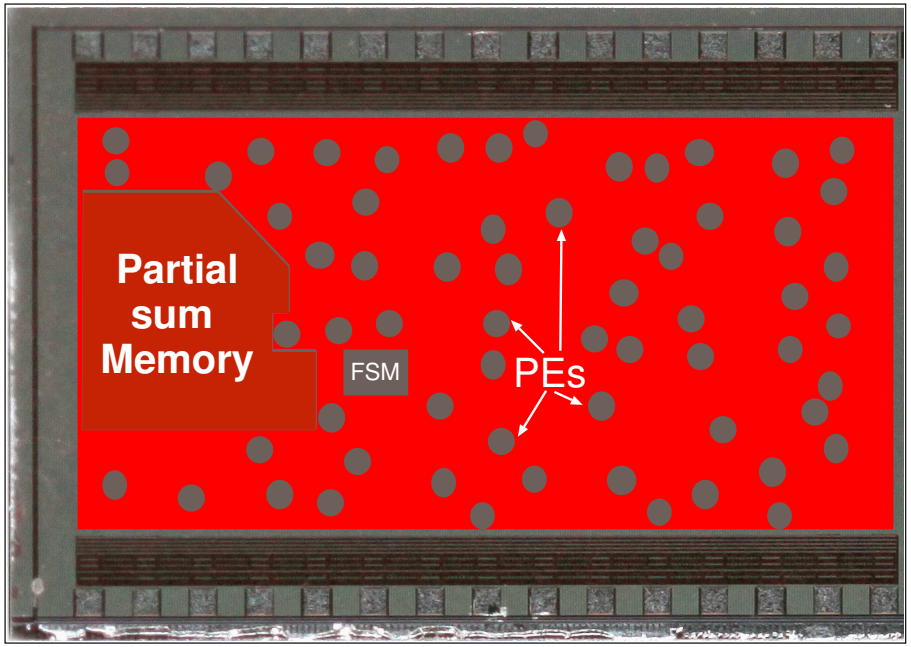
Successive Cancellation Decoder in 180nm CMOS

LLR memory area:

- Area: 1.35 mm^2
- Gate count: 144k
- 78% of the chip area

Partial sum memory area:

- Area: 0.22 mm^2
- Gate count: 24k
- 13% of the chip area



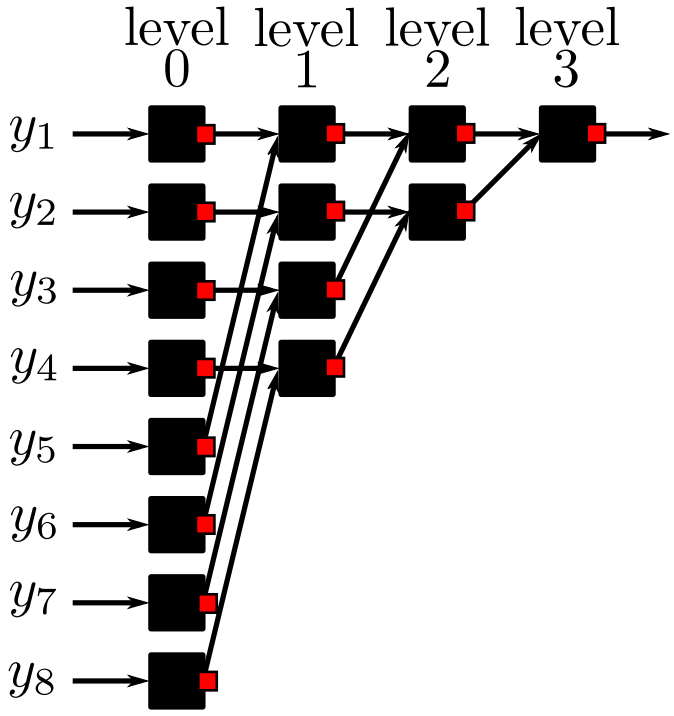
Memory is **more than 91%** of the total area.

Our focus: reliability of FEC decoders

1. Polar codes under **successive cancellation decoding** with faulty memory.
2. LDPC codes under **quantized min-sum decoding** with faulty memory.

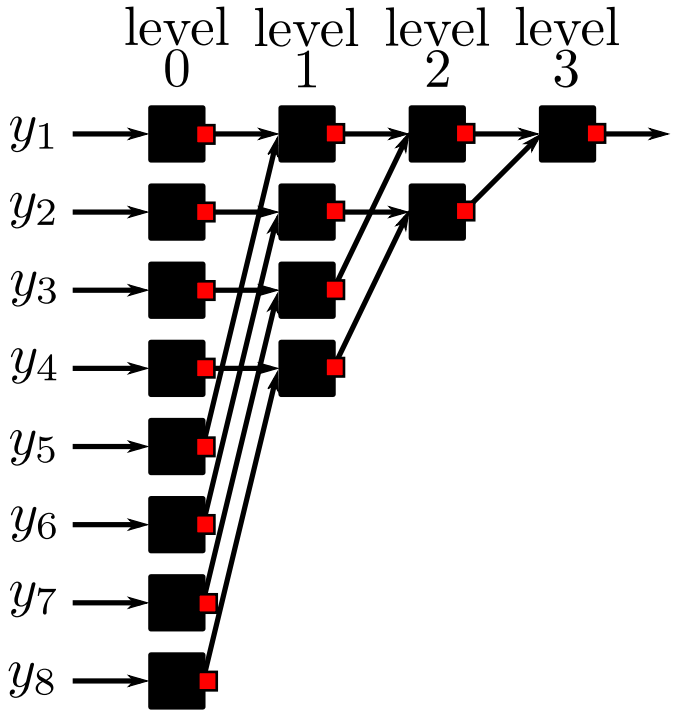
Hardware Implementation of SC Decoding

- For each synthetic channel, **tree structure** with $n+1$ stages (2^n **unique trees**)



Hardware Implementation of SC Decoding

- For each synthetic channel, **tree structure** with $n+1$ stages (2^n **unique trees**)



Implementation: nodes implemented as processing elements (PEs)

- Combinational logic (reliable) can be shared between PEs.
- **Dedicated memory cell** is required **for the output of each node**.

Density evolution for faulty SC decoding

Describe failures in a memory cell as unreliable variable/check node computation

Fault model

Additional erasures appear in **non-erased** outputs with probability $0 < \delta < 1$

Density evolution for faulty SC decoding

Describe failures in a memory cell as unreliable variable/check node computation

Fault model

Additional erasures appear in **non-erased** outputs with probability $0 < \delta < 1$

- Erasure probability at g **nodes**: $T_\delta^+(\epsilon) \triangleq \epsilon^2 + (1 - \epsilon^2)\delta$

Density evolution for faulty SC decoding

Describe failures in a memory cell as unreliable variable/check node computation

Fault model

Additional erasures appear in **non-erased** outputs with probability $0 < \delta < 1$

- Erasure probability at g **nodes**: $T_\delta^+(\epsilon) \triangleq \epsilon^2 + (1-\epsilon^2)\delta$
- Erasure probability at f **nodes**: $T_\delta^-(\epsilon) \triangleq 2\epsilon - \epsilon^2 + (1-2\epsilon+\epsilon^2)\delta$

Density evolution for faulty SC decoding

Describe failures in a memory cell as unreliable variable/check node computation

Fault model

Additional erasures appear in **non-erased** outputs with probability $0 < \delta < 1$

- Erasure probability at g **nodes**: $T_\delta^+(\epsilon) \triangleq \epsilon^2 + (1-\epsilon^2)\delta$
- Erasure probability at f **nodes**: $T_\delta^-(\epsilon) \triangleq 2\epsilon - \epsilon^2 + (1-2\epsilon+\epsilon^2)\delta$

- Polarization process:

$$\epsilon_{j+1,\delta} = \begin{cases} T_\delta^+(\epsilon_{j,\delta}) & \text{w.p. } 1/2, \\ T_\delta^-(\epsilon_{j,\delta}) & \text{w.p. } 1/2, \end{cases} \qquad \epsilon_{0,\delta} = p.$$

A few negative results

Theorem (No channel becomes arbitrarily good)

Let $\mathcal{S}$ denote the sample space of $\epsilon_{j,\delta}$. Then, $\nexists s \in \mathcal{S}$ such that $\epsilon_{j,\delta}(s) \overset{j\to\infty}{\longrightarrow} 0$.

A few negative results

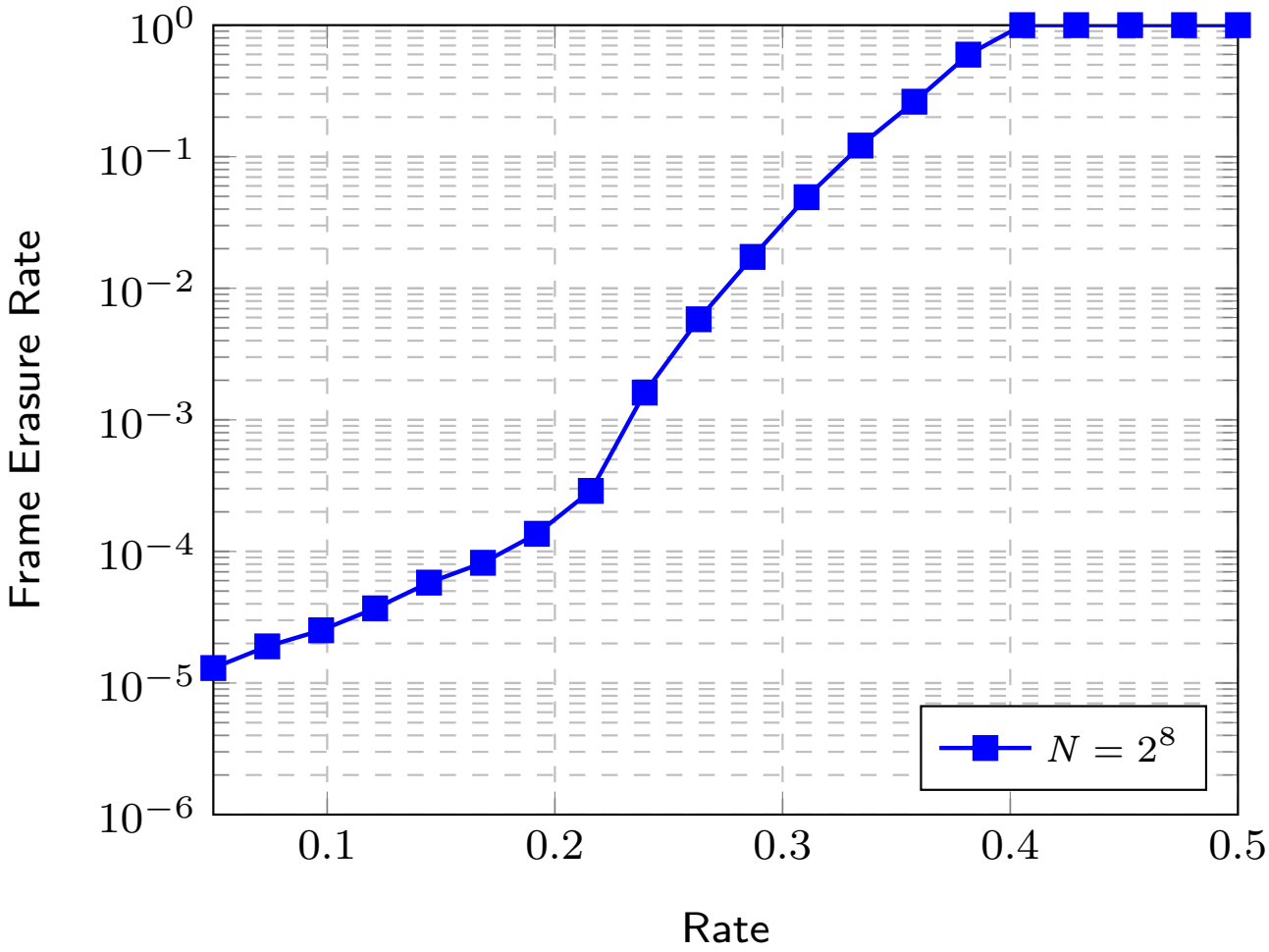
Theorem (No channel becomes arbitrarily good)

Let $\mathcal{S}$ denote the sample space of $\epsilon_{j,\delta}$. Then, $\nexists s \in \mathcal{S}$ such that $\epsilon_{j,\delta}(s) \overset{j\to\infty}{\longrightarrow} 0$.

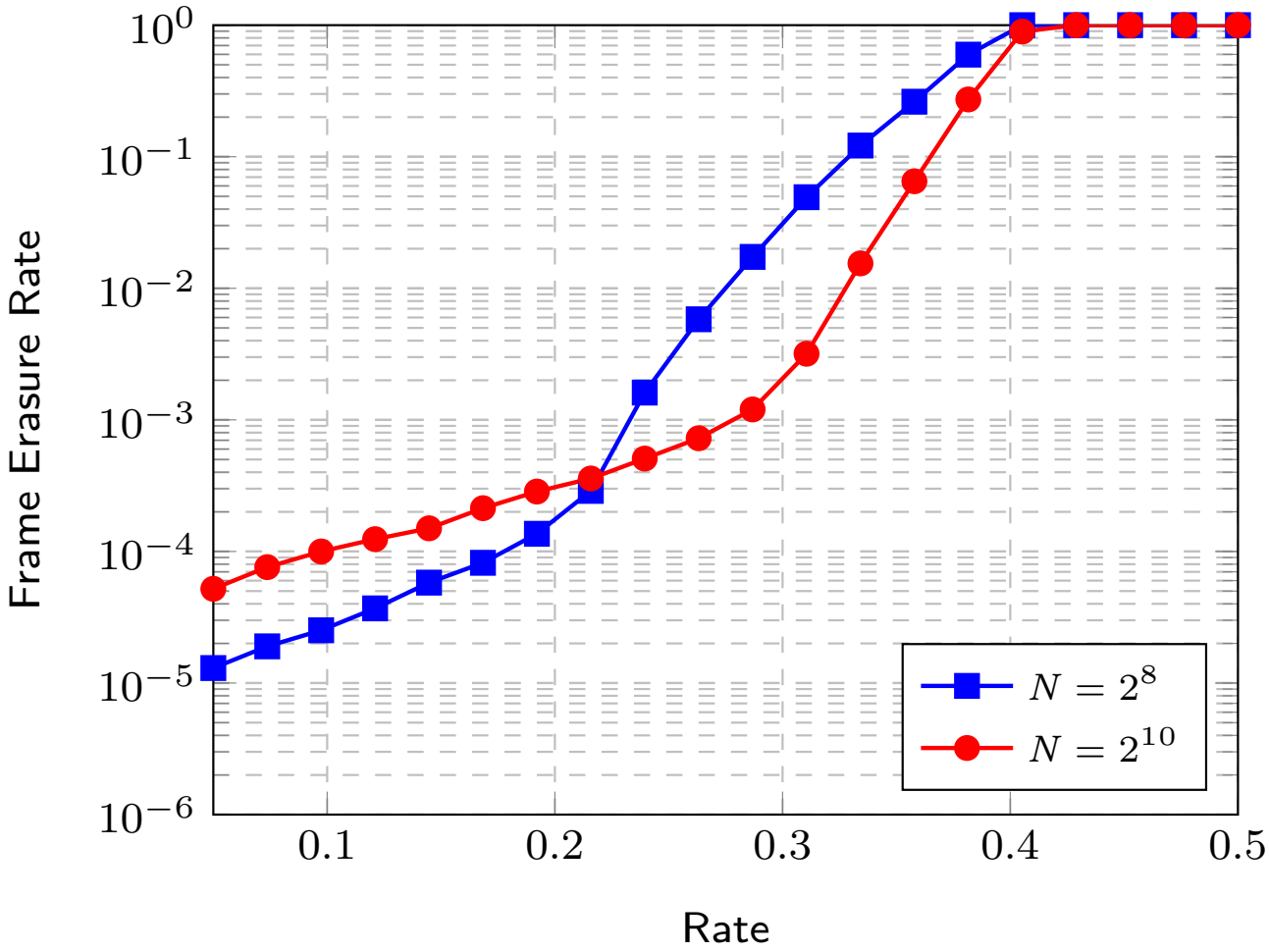
Theorem (All channels become asymptotically useless)

Under faulty SC decoding over the BEC, $\epsilon_{j,\delta} \overset{\text{a.s.}}{\longrightarrow} 1$.

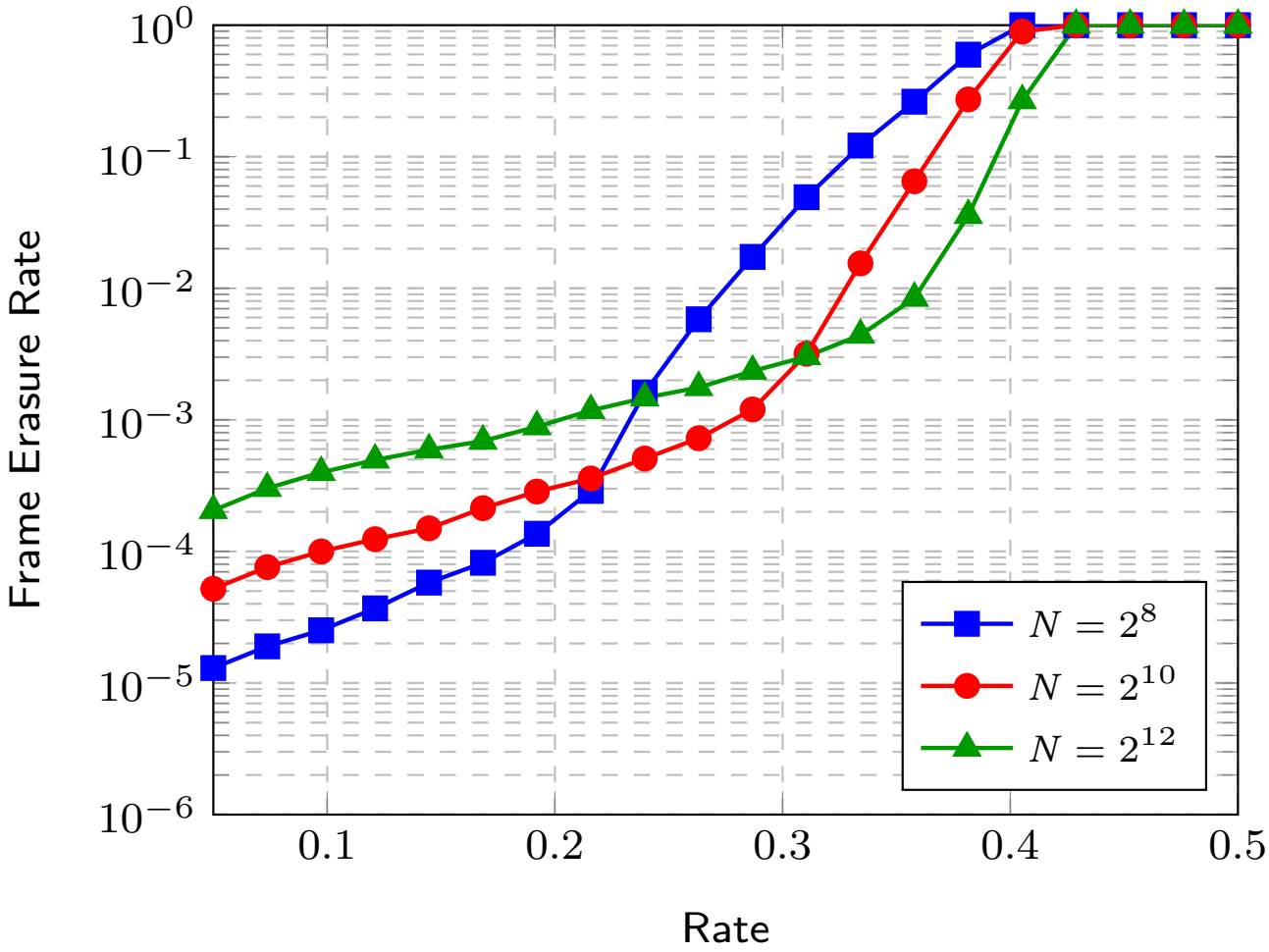
Numerical results: Frame erasure rate



Numerical results: Frame erasure rate



Numerical results: Frame erasure rate



Improving robustness: Optimal blocklength

- Two conflicting processes:
  1. **Polarization** tends to decrease the FER
  2. **Internal erasures** tend to increase the FER

Improving robustness: Optimal blocklength

- Two conflicting processes:
  1. **Polarization** tends to decrease the FER
  2. **Internal erasures** tend to increase the FER

- Asymptotically the **internal erasures** dominate (recall that $\epsilon_{j,\delta} \xrightarrow{\text{a.s.}} 1$)

Improving robustness: Optimal blocklength

- Two conflicting processes:
  1. **Polarization** tends to decrease the FER
  2. **Internal erasures** tend to increase the FER

- Asymptotically the **internal erasures** dominate (recall that $\epsilon_{j,\delta} \xrightarrow{\text{a.s.}} 1$)

- There exists a **FER-optimal** finite blocklength.

Improving robustness: Optimal blocklength

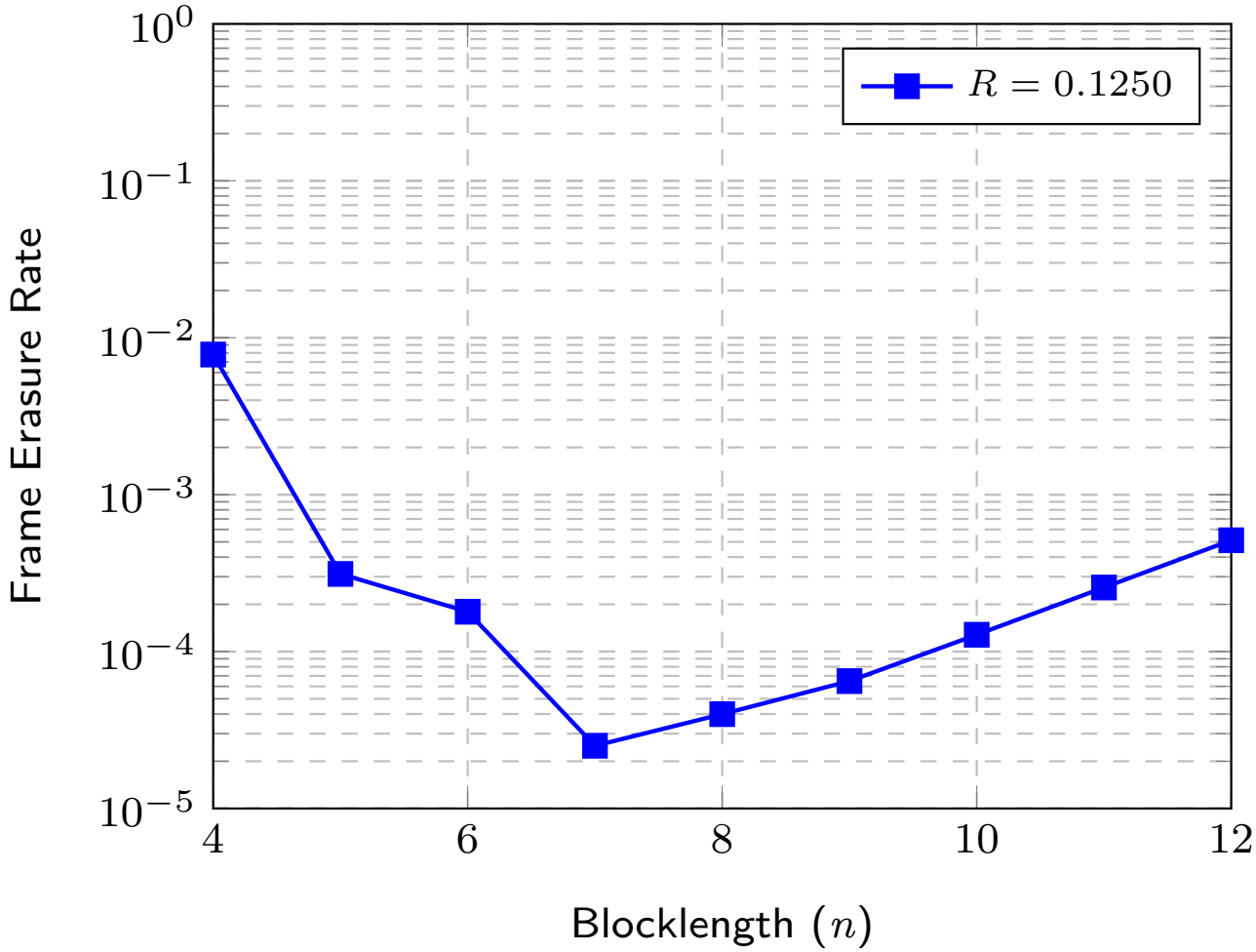
- Two conflicting processes:
  1. **Polarization** tends to decrease the FER
  2. **Internal erasures** tend to increase the FER

- Asymptotically the **internal erasures** dominate (recall that $\epsilon_{j,\delta} \xrightarrow{\text{a.s.}} 1$)

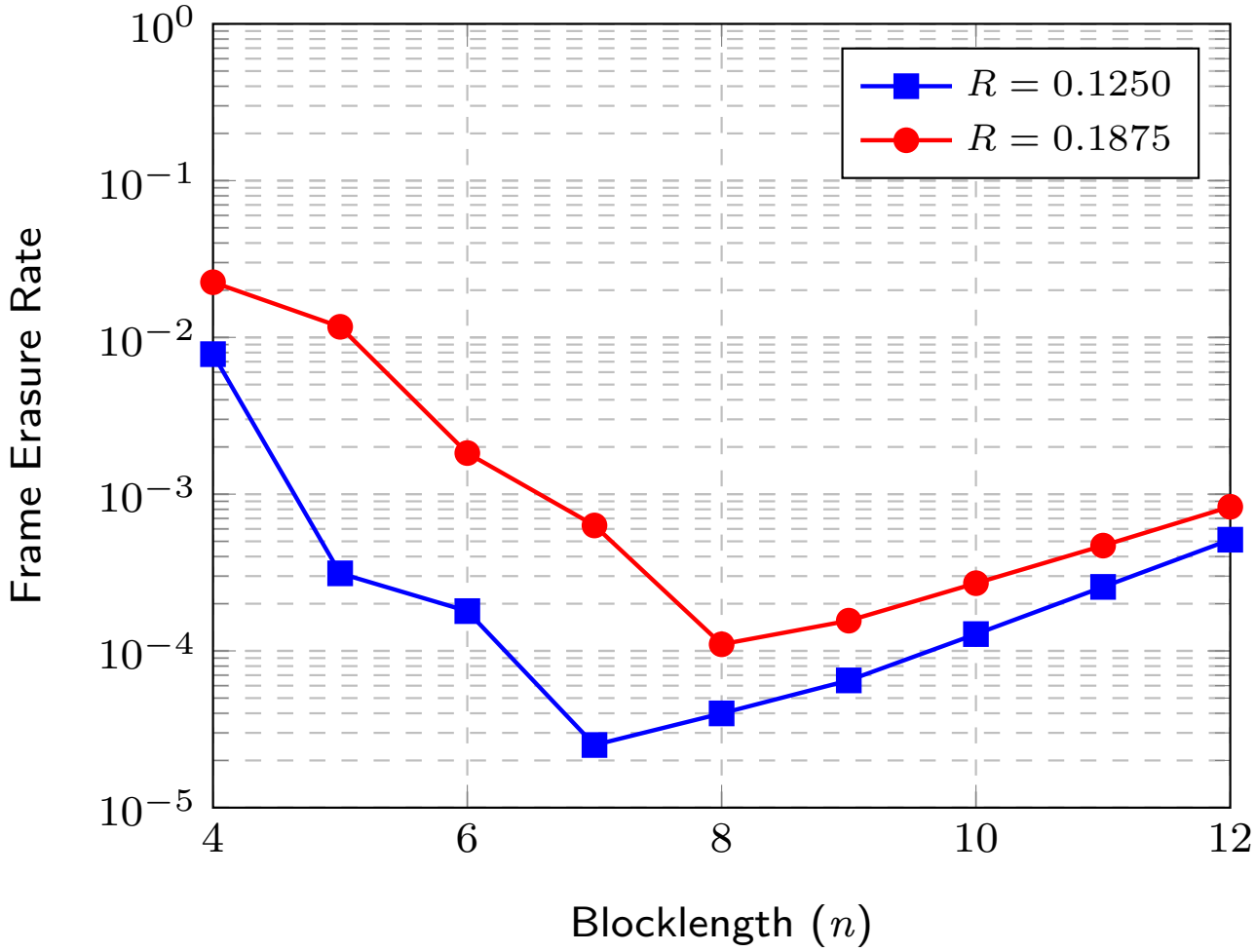
- There exists a **FER-optimal** finite blocklength.
- For a given code rate R, optimal blocklength N^* is defined as

$$N^* = \arg \min_{N=2^n,\ n\in\mathbb{N}} P_f(N, R)$$

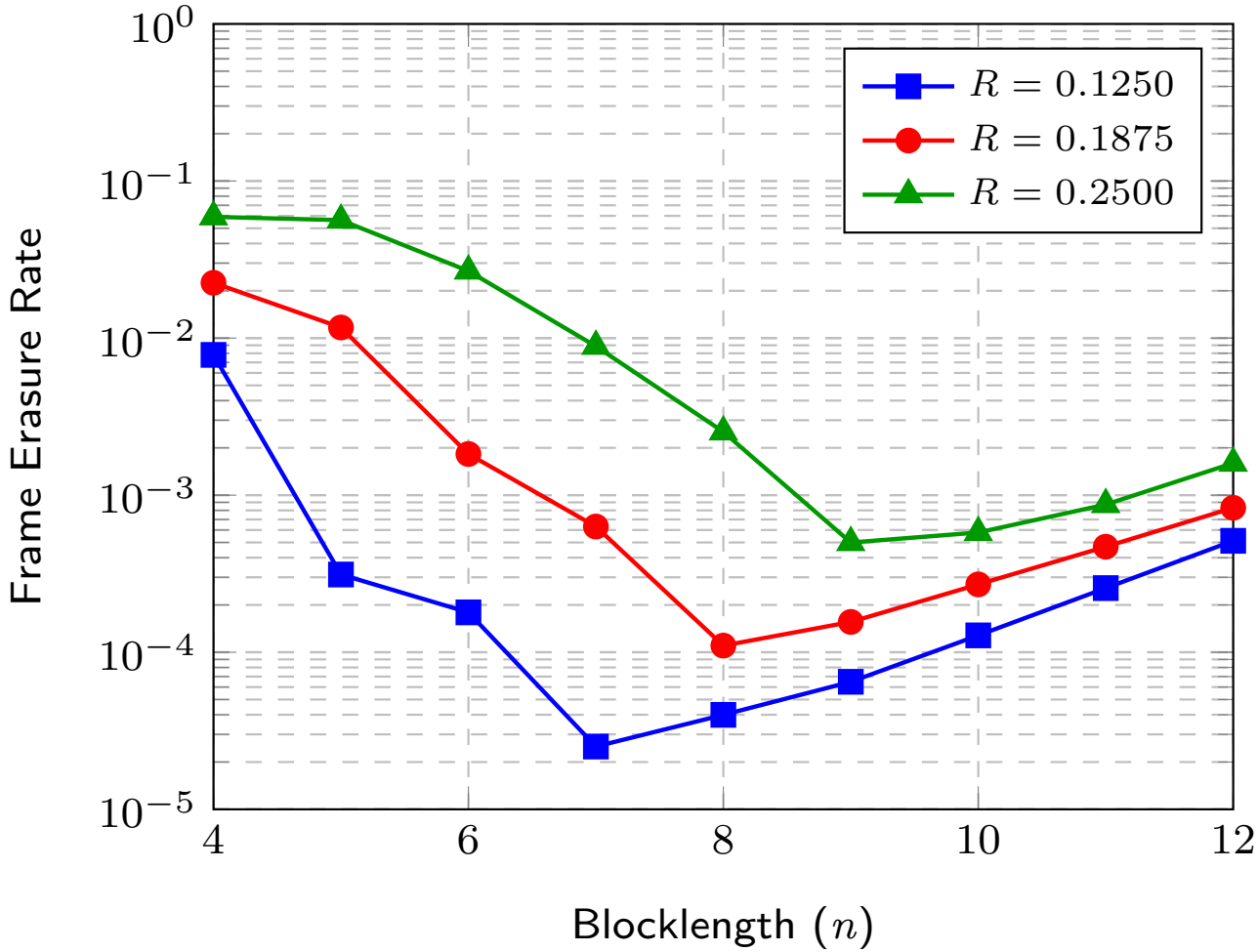
Numerical results: Optimal blocklength



Numerical results: Optimal blocklength



Numerical results: Optimal blocklength

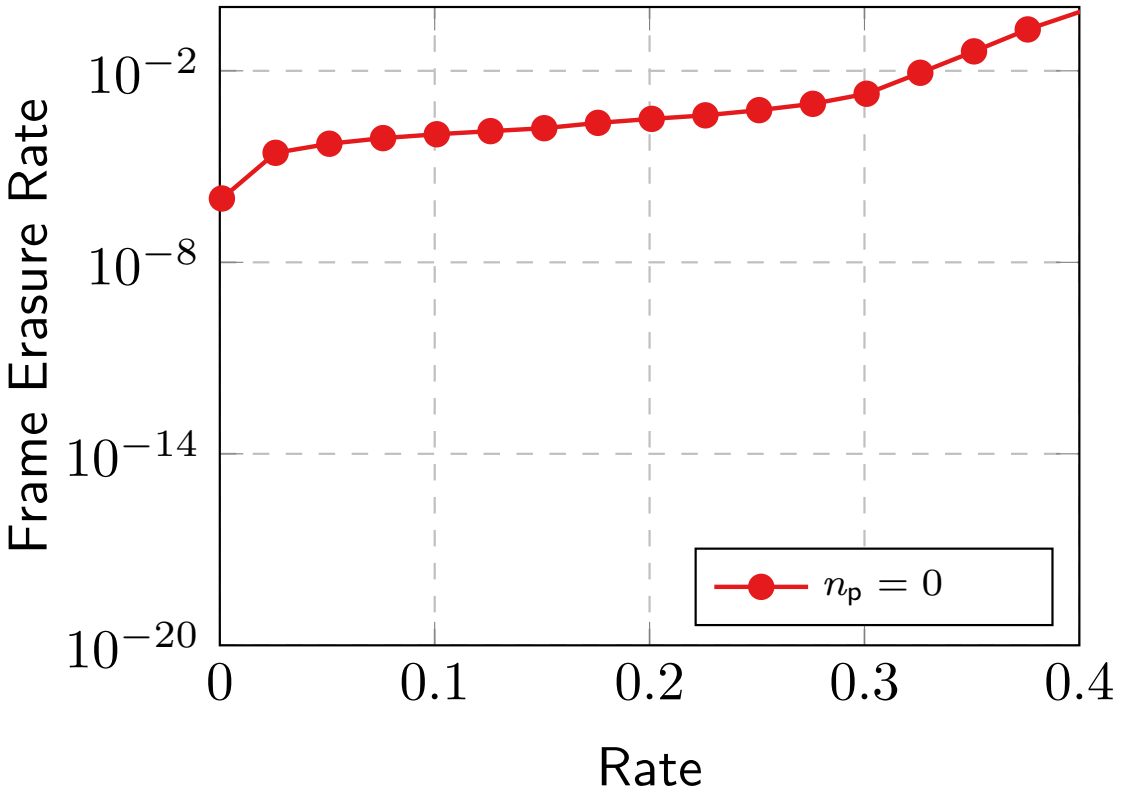


Improving robustness: Unequal error protection

- **Concept:** not all parts of the decoder are equally important.

Improving robustness: Unequal error protection

- **Concept:** not all parts of the decoder are equally important.



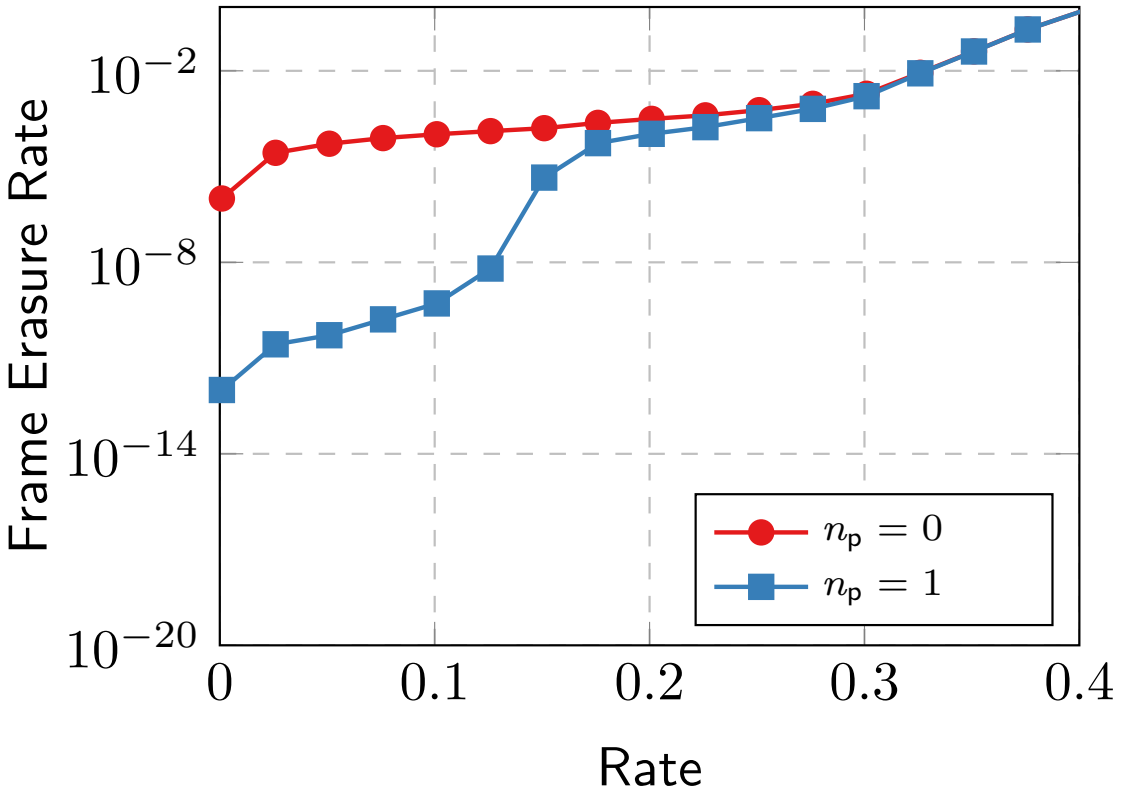
% of protected MEs ($N_\mathsf{p}/N_\mathsf{ME}$):

- $n_\mathsf{p} = 0$: 0.00%

$N = 1024$, BEC(0.5), $\delta = 10^{-6}$

Improving robustness: Unequal error protection

- **Concept:** not all parts of the decoder are equally important.



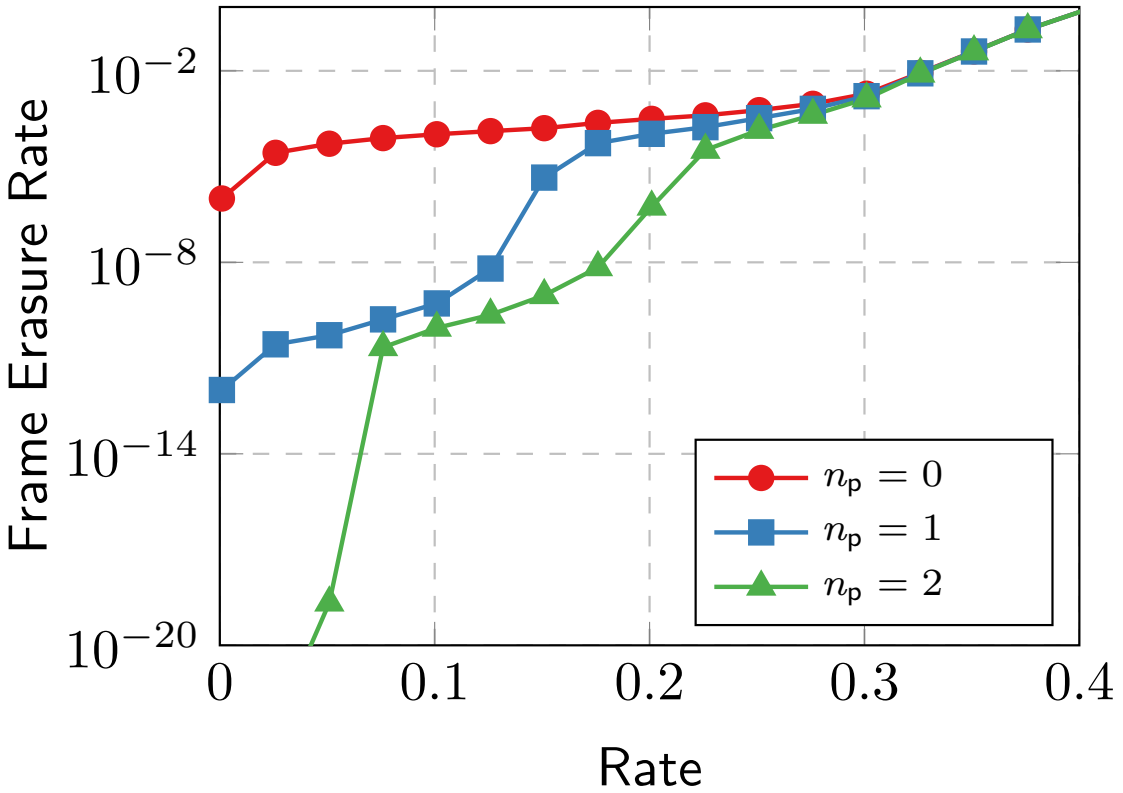
% of protected MEs (N_p/N_{ME}):

- $n_p = 0$: 0.00%
- $n_p = 1$: 0.05%

$N = 1024$, BEC(0.5), $\delta = 10^{-6}$

Improving robustness: Unequal error protection

- **Concept:** not all parts of the decoder are equally important.



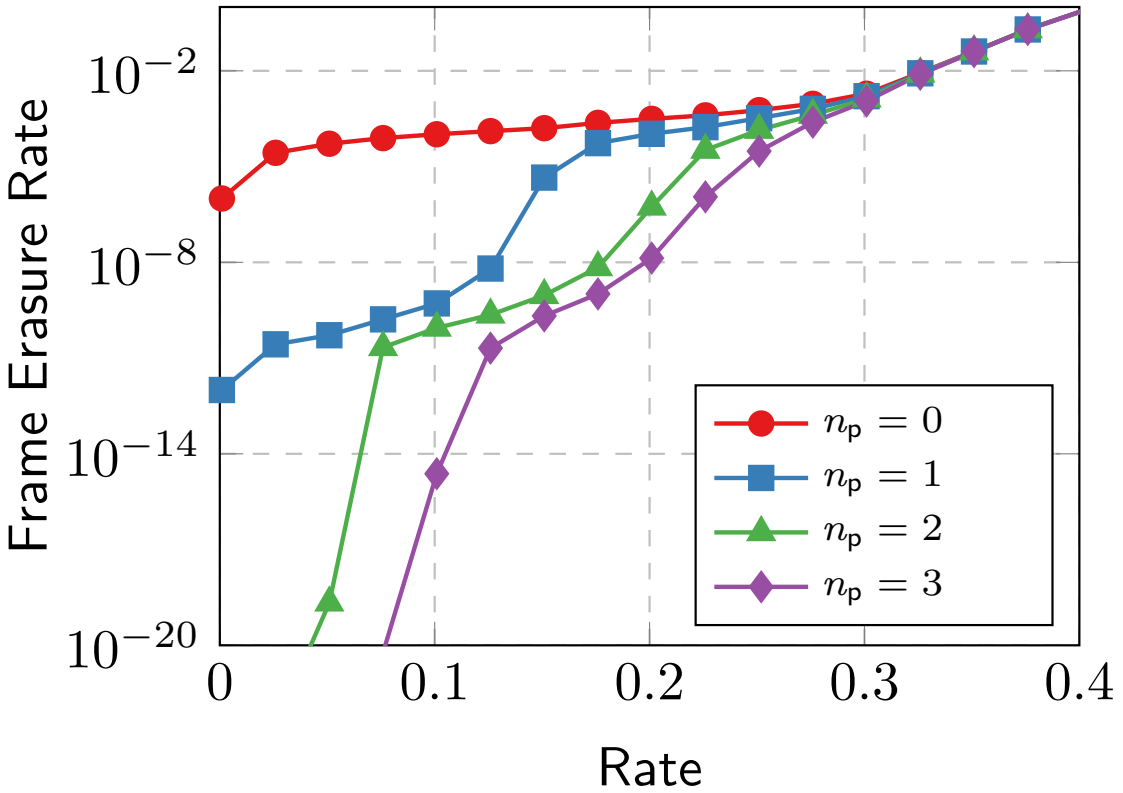
% of protected MEs (N_p/N_{ME}):

- $n_p = 0$: 0.00%
- $n_p = 1$: 0.05%
- $n_p = 2$: 0.15%

$N = 1024$, BEC(0.5), $\delta = 10^{-6}$

Improving robustness: Unequal error protection

- **Concept:** not all parts of the decoder are equally important.



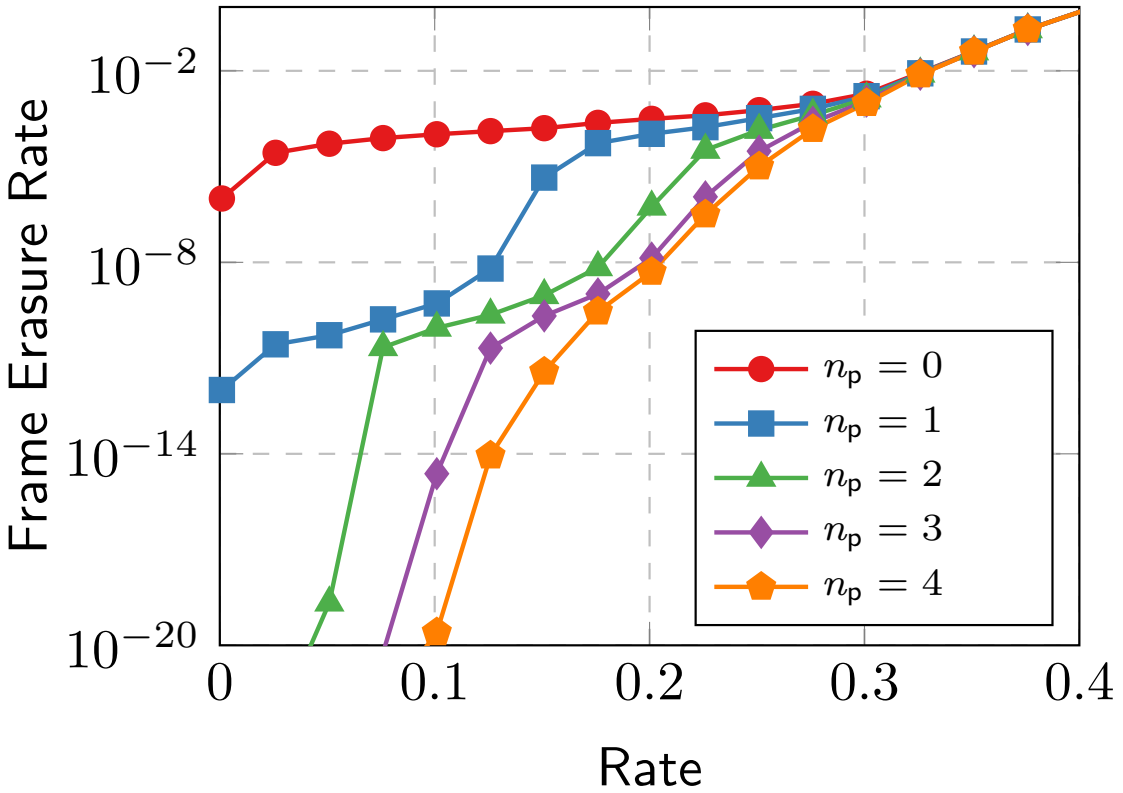
$N = 1024$, BEC(0.5), $\delta = 10^{-6}$

% of protected MEs (N_p/N_{ME}):

- $n_p = 0$: 0.00%
- $n_p = 1$: 0.05%
- $n_p = 2$: 0.15%
- $n_p = 3$: 0.34%

Improving robustness: Unequal error protection

- **Concept:** not all parts of the decoder are equally important.



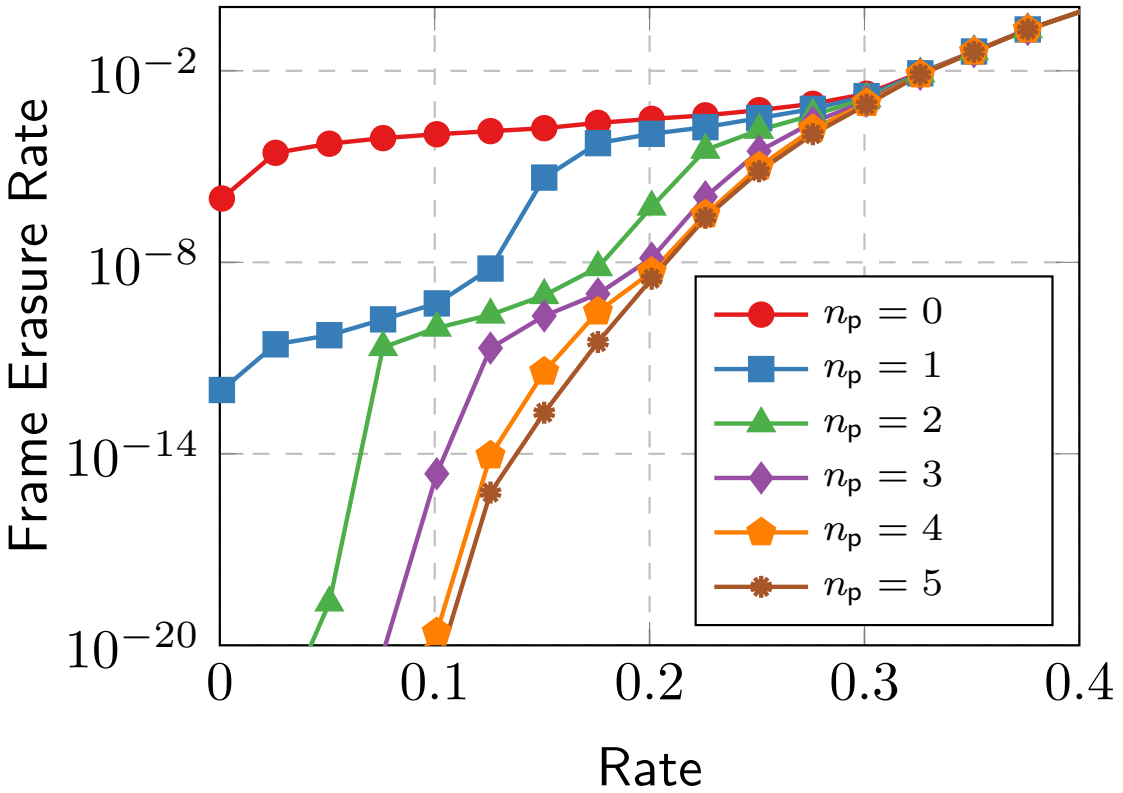
% of protected MEs (N_p/N_{ME}):

- $n_p = 0$: 0.00%
- $n_p = 1$: 0.05%
- $n_p = 2$: 0.15%
- $n_p = 3$: 0.34%
- $n_p = 4$: 0.73%

$N = 1024$, BEC(0.5), $\delta = 10^{-6}$

Improving robustness: Unequal error protection

- **Concept:** not all parts of the decoder are equally important.



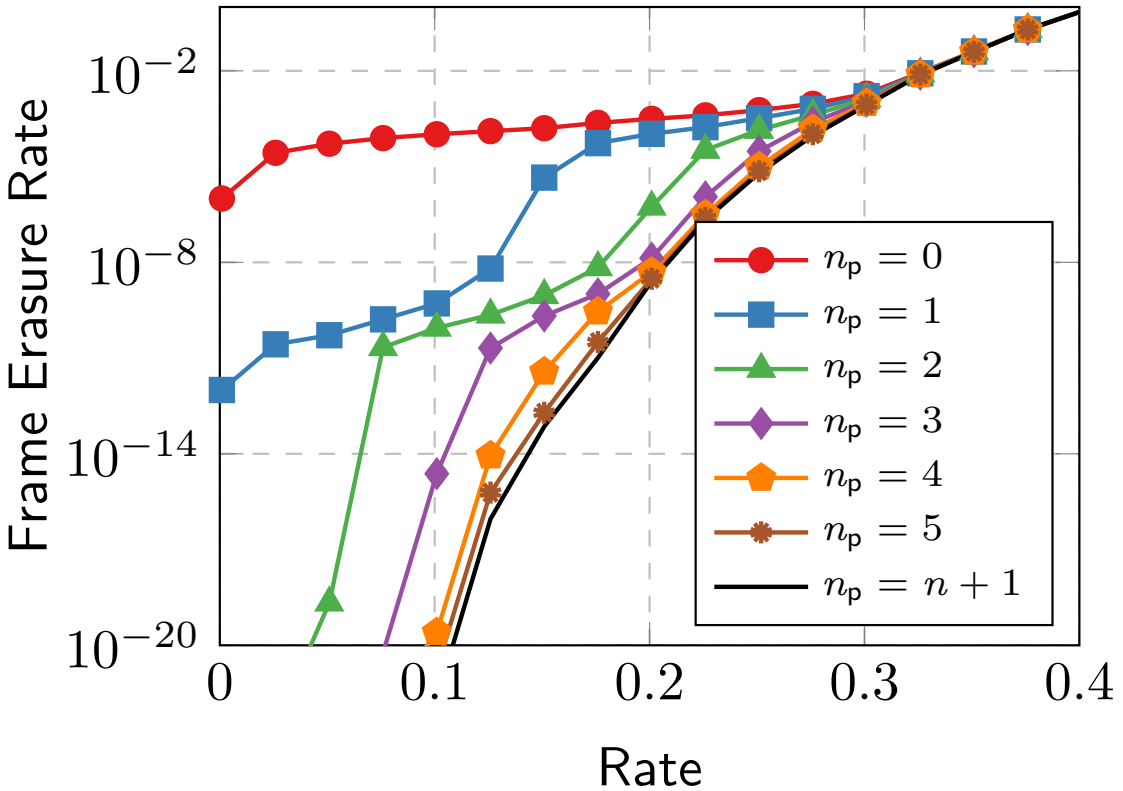
% of protected MEs (N_p/N_{ME}):

- $n_p = 0$: 0.00%
- $n_p = 1$: 0.05%
- $n_p = 2$: 0.15%
- $n_p = 3$: 0.34%
- $n_p = 4$: 0.73%
- $n_p = 5$: 1.51%

$N = 1024$, BEC(0.5), $\delta = 10^{-6}$

Improving robustness: Unequal error protection

- **Concept:** not all parts of the decoder are equally important.



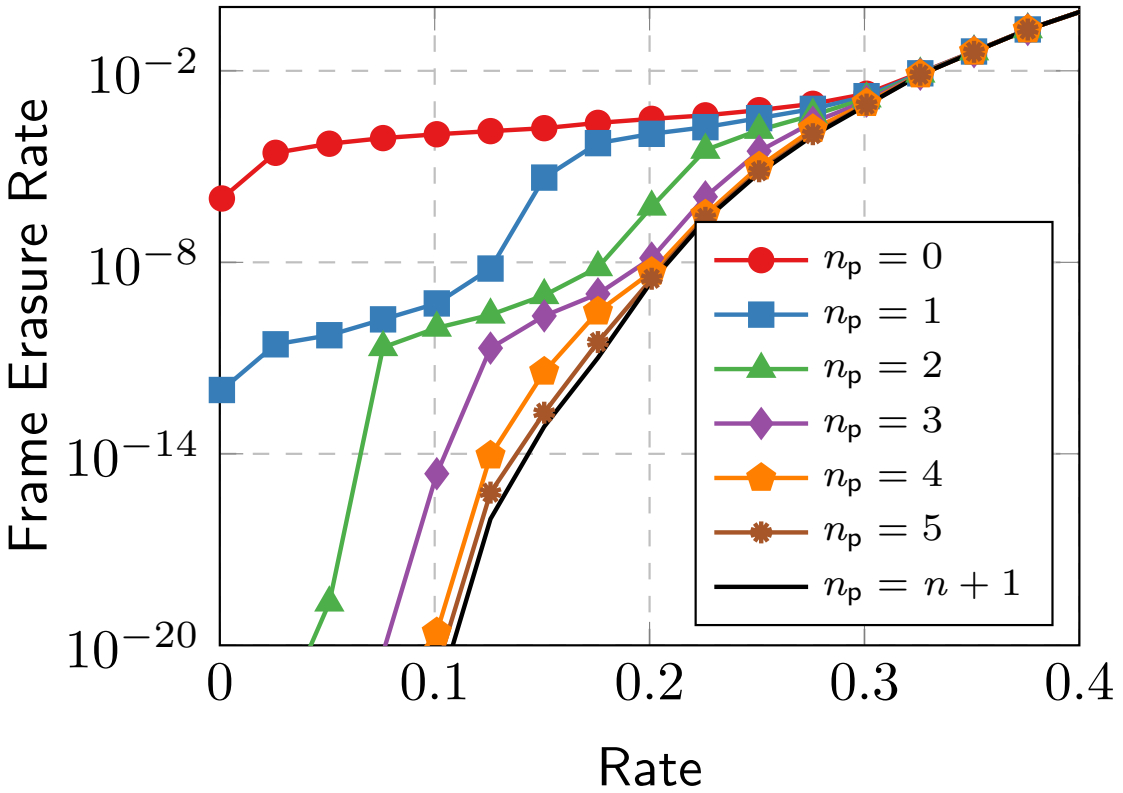
$N = 1024$, BEC(0.5), $\delta = 10^{-6}$

% of protected MEs (N_p/N_{ME}):

- $n_p = 0$: 0.00%
- $n_p = 1$: 0.05%
- $n_p = 2$: 0.15%
- $n_p = 3$: 0.34%
- $n_p = 4$: 0.73%
- $n_p = 5$: 1.51%
- $n_p = n + 1$: 100.00%

Improving robustness: Unequal error protection

- **Concept:** not all parts of the decoder are equally important.



% of protected MEs (N_p/N_{ME}):

- $n_p = 0$: 0.00%
- $n_p = 1$: 0.05%
- $n_p = 2$: 0.15%
- $n_p = 3$: 0.34%
- $n_p = 4$: 0.73%
- $n_p = 5$: 1.51%
- $n_p = n + 1$: 100.00%

$N = 1024$, BEC(0.5), $\delta = 10^{-6}$

Asymptotically **error-free** transmission with **very little overhead**

Summary

Background on polar codes.

- Polarizing transformation, encoding, decoding.

Summary

Background on polar codes.

- Polarizing transformation, encoding, decoding.

Polar Decoding with Faulty Hardware

- Introduced a memory fault model.
- Asymptotic analysis shows that SC decoder is **very sensitive** to internal erasures.
- **Efficient** robustness enhancement methods:
  1. Optimal blocklength.
  2. Unequal error protection.