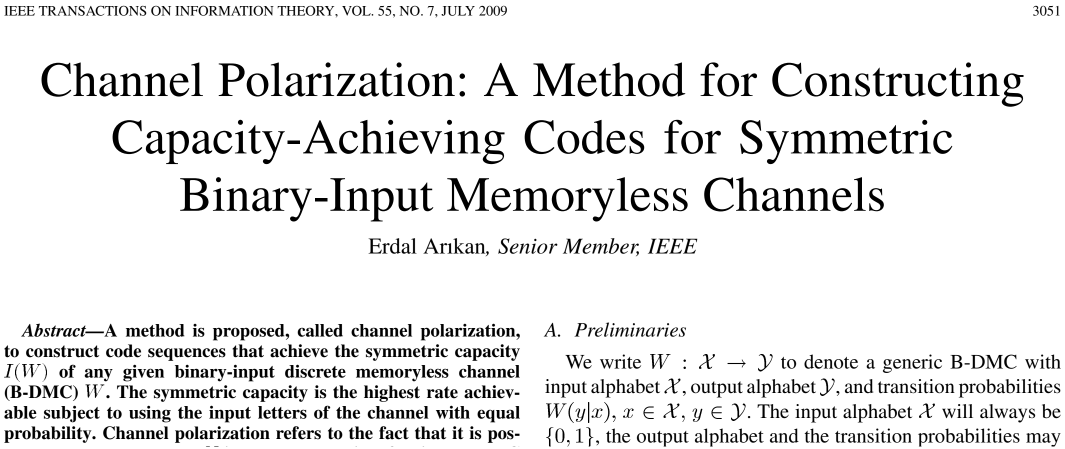
Polar Code Design for SCL Decoding: An Information Theoretic Perspective

Mustafa Cemil Coşkun, mustafa.coskun@tum.de
Technical University of Munich (TUM)

35th Meeting of the ITG Professional Group
"Applied Information Theory"
May 27, 2021

Polar Codes



IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 55, NO. 7, JULY 2009 3051

Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels

Erdal Arıkan, *Senior Member, IEEE*

***Abstract*—A method is proposed, called channel polarization, to construct code sequences that achieve the symmetric capacity $I(W)$ of any given binary-input discrete memoryless channel (B-DMC) W. The symmetric capacity is the highest rate achievable subject to using the input letters of the channel with equal probability. Channel polarization refers to the fact that it is pos-**

A. Preliminaries

We write $W : \mathcal{X} \to \mathcal{Y}$ to denote a generic B-DMC with input alphabet $\mathcal{X}$, output alphabet $\mathcal{Y}$, and transition probabilities $W(y|x)$, $x \in \mathcal{X}$, $y \in \mathcal{Y}$. The input alphabet $\mathcal{X}$ will always be $\{0, 1\}$, the output alphabet and the transition probabilities may

- They are capacity-achieving on binary memoryless symmetric (BMS) channels with low encoding/decoding complexity [Arı09].

Polar Codes

IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 55, NO. 7, JULY 2009 3051

Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels

Erdal Arıkan, *Senior Member, IEEE*

***Abstract*—A method is proposed, called channel polarization, to construct code sequences that achieve the symmetric capacity $I(W)$ of any given binary-input discrete memoryless channel (B-DMC) W. The symmetric capacity is the highest rate achievable subject to using the input letters of the channel with equal probability. Channel polarization refers to the fact that it is pos-**

A. Preliminaries

We write $W : \mathcal{X} \to \mathcal{Y}$ to denote a generic B-DMC with input alphabet $\mathcal{X}$, output alphabet $\mathcal{Y}$, and transition probabilities $W(y|x)$, $x \in \mathcal{X}$, $y \in \mathcal{Y}$. The input alphabet $\mathcal{X}$ will always be $\{0,1\}$, the output alphabet and the transition probabilities may

- They are capacity-achieving on binary memoryless symmetric (BMS) channels with low encoding/decoding complexity [Arı09].
- But successive cancellation (SC) decoding performs poorly for small blocks.

Successive List Cancellation Decoding

IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 61, NO. 5, MAY 2015 2213

List Decoding of Polar Codes

Ido Tal, *Member, IEEE* and Alexander Vardy, *Fellow, IEEE*

***Abstract*— We describe a successive-cancellation list decoder for polar codes, which is a generalization of the classic successive-cancellation decoder of Arıkan. In the proposed list decoder, L decoding paths are considered concurrently at each decoding stage, where L is an integer parameter. At the end of the decoding process, the most likely among the L paths is selected as the single codeword at the decoder output. Simulations show that the resulting performance is very close to that of maximum-likelihood decoding, even for moderate values of L. Alternatively, if a genie is allowed to pick the transmitted codeword from the list, the results are comparable with the performance of current state-of-the-art LDPC codes. We show that such a genie can be easily implemented using simple CRC precoding. The specific list-decoding algorithm that achieves this performance doubles the number of decoding paths for each information bit, and then uses a pruning procedure to discard all but the L most likely paths. However, straightforward implementation of this**

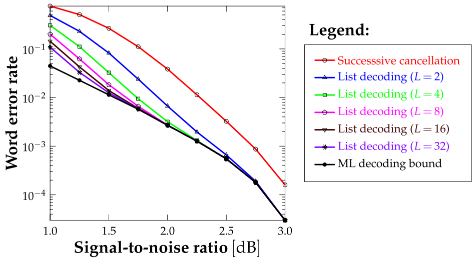


Fig. 1. List-decoding performance for a polar code of length $n = 2048$ and rate $R = 0.5$ on the BPSK-modulated Gaussian channel. The code was constructed using the methods of [15], with optimization for $E_b/N_0 = 2\,\text{dB}$.

- SC list (SCL) decoding with CRC and large list-size performs very well and matches maximum-likelihood (ML) [TV15].

Successive List Cancellation Decoding

IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 61, NO. 5, MAY 2015 2213

List Decoding of Polar Codes

Ido Tal, *Member, IEEE* and Alexander Vardy, *Fellow, IEEE*

***Abstract*— We describe a successive-cancellation list decoder for polar codes, which is a generalization of the classic successive-cancellation decoder of Arıkan. In the proposed list decoder, L decoding paths are considered concurrently at each decoding stage, where L is an integer parameter. At the end of the decoding process, the most likely among the L paths is selected as the single codeword at the decoder output. Simulations show that the resulting performance is very close to that of maximum-likelihood decoding, even for moderate values of L. Alternatively, if a genie is allowed to pick the transmitted codeword from the list, the results are comparable with the performance of current state-of-the-art LDPC codes. We show that such a genie can be easily implemented using simple CRC precoding. The specific list-decoding algorithm that achieves this performance doubles the number of decoding paths for each information bit, and then uses a pruning procedure to discard all but the L most likely paths. However, straightforward implementation of this**

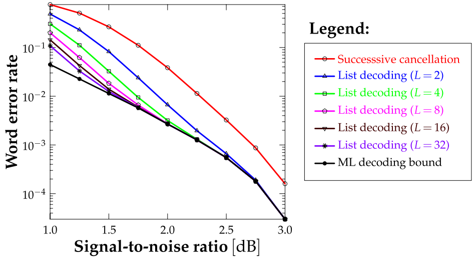


Fig. 1. List-decoding performance for a polar code of length $n = 2048$ and rate $R = 0.5$ on the BPSK-modulated Gaussian channel. The code was constructed using the methods of [15], with optimization for $E_b/N_0 = 2\,\text{dB}$.

- SC list (SCL) decoding with CRC and large list-size performs very well and matches maximum-likelihood (ML) [TV15].
- It can also be used to decode other codes (e.g., Reed–Muller codes).

Polar Codes with Dynamic Frozen Bits

254 IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 34, NO. 2, FEBRUARY 2016

Polar Subcodes

Peter Trifonov, *Member, IEEE*, and Vera Miloslavskaya, *Member, IEEE*

***Abstract*—An extension of polar codes is proposed, which allows some of the frozen symbols, called dynamic frozen symbols, to be data-dependent. A construction of polar codes with dynamic frozen symbols, being subcodes of extended BCH codes, is proposed. The proposed codes have higher minimum distance than classical polar codes, but still can be efficiently decoded using the successive cancellation algorithm and its extensions. The codes with Arikan, extended BCH and Reed-Solomon kernel are considered. The proposed codes are shown to outperform LDPC and turbo codes, as well as polar codes with CRC.**

RM codes, and are therefore likely to provide better finite length performance. However, there are still no efficient MAP decoding algorithms for these codes.

It was suggested in [17] to construct subcodes of RM codes, which can be efficiently decoded by a recursive list decoding algorithm. In this paper we generalize this approach, and propose a code construction "in between" polar codes and EBCH codes. The proposed codes can be efficiently decoded using the techniques developed in the area of polar coding, but provide

- Later, polar codes were extended with the concept of dynamic frozen bits, which enabled state-of-art designs.

Polar Codes with Dynamic Frozen Bits

254 IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 34, NO. 2, FEBRUARY 2016

Polar Subcodes

Peter Trifonov, *Member, IEEE*, and Vera Miloslavskaya, *Member, IEEE*

***Abstract*—An extension of polar codes is proposed, which allows some of the frozen symbols, called dynamic frozen symbols, to be data-dependent. A construction of polar codes with dynamic frozen symbols, being subcodes of extended BCH codes, is proposed. The proposed codes have higher minimum distance than classical polar codes, but still can be efficiently decoded using the successive cancellation algorithm and its extensions. The codes with Arikan, extended BCH and Reed-Solomon kernel are considered. The proposed codes are shown to outperform LDPC and turbo codes, as well as polar codes with CRC.**

RM codes, and are therefore likely to provide better finite length performance. However, there are still no efficient MAP decoding algorithms for these codes.

It was suggested in [17] to construct subcodes of RM codes, which can be efficiently decoded by a recursive list decoding algorithm. In this paper we generalize this approach, and propose a code construction "in between" polar codes and EBCH codes. The proposed codes can be efficiently decoded using the techniques developed in the area of polar coding, but provide

- Later, polar codes were extended with the concept of dynamic frozen bits, which enabled state-of-art designs.
- It is also shown that any code can be decoded using SCL decoding, but some require very large complexity for a good performance.

Motivating Question

- What list size is sufficient to approach ML decoding performance for a given polar code and channel?

Motivating Question

- What list size is sufficient to approach ML decoding performance for a given polar code and channel?
  - To avoid losing true codeword, its rank must not be larger than list size.

Motivating Question

- What list size is sufficient to approach ML decoding performance for a given polar code and channel?
  - To avoid losing true codeword, its rank must not be larger than list size.
  - The expected log-rank of correct codeword is upper bounded by an entropy.

Motivating Question

- What list size is sufficient to approach ML decoding performance for a given polar code and channel?
  - To avoid losing true codeword, its rank must not be larger than list size.
  - The expected log-rank of correct codeword is upper bounded by an entropy.
- For the BEC,
  - This entropy equals the dimension of an affine subspace.

Motivating Question

- What list size is sufficient to approach ML decoding performance for a given polar code and channel?
  - To avoid losing true codeword, its rank must not be larger than list size.
  - The expected log-rank of correct codeword is upper bounded by an entropy.
- For the BEC,
  - This entropy equals the dimension of an affine subspace.
  - The average dimension can be well modeled by a simple Markov chain (only numerical results shown here!).

Motivating Question

- What list size is sufficient to approach ML decoding performance for a given polar code and channel?
  - To avoid losing true codeword, its rank must not be larger than list size.
  - The expected log-rank of correct codeword is upper bounded by an entropy.
- For the BEC,
  - This entropy equals the dimension of an affine subspace.
  - The average dimension can be well modeled by a simple Markov chain (only numerical results shown here!).

Based on joint works with Henry D. Pfister [CP20, CP21]

Outline

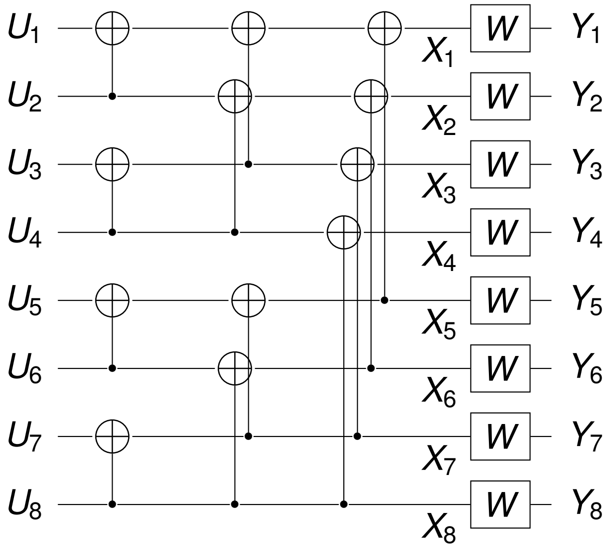
Outline

Polar Codes and Density Evolution

$$x_1^N = u_1^N \mathbf{G}_2^{\otimes n} \quad \text{where} \quad \mathbf{G}_2 \triangleq \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad \text{and} \quad N = 2^n$$

Polar Codes and Density Evolution

$$x_1^N = u_1^N \mathbf{G}_2^{\otimes n} \quad \text{where} \quad \mathbf{G}_2 \triangleq \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad \text{and} \quad N = 2^n$$

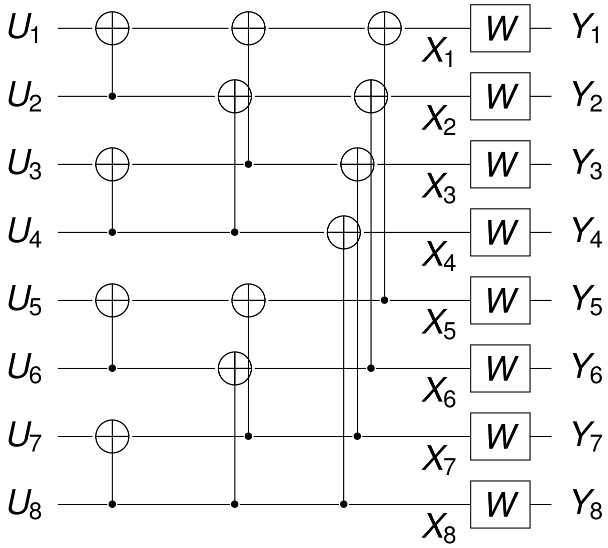


Polar Codes and Density Evolution

$$x_1^N = u_1^N \mathbf{G}_2^{\otimes n} \quad \text{where} \quad \mathbf{G}_2 \triangleq \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad \text{and} \quad N = 2^n$$

Synthesized Channel i

$$W_N^{(i)}\left(y_1^N, u_1^{i-1} | u_i\right)$$

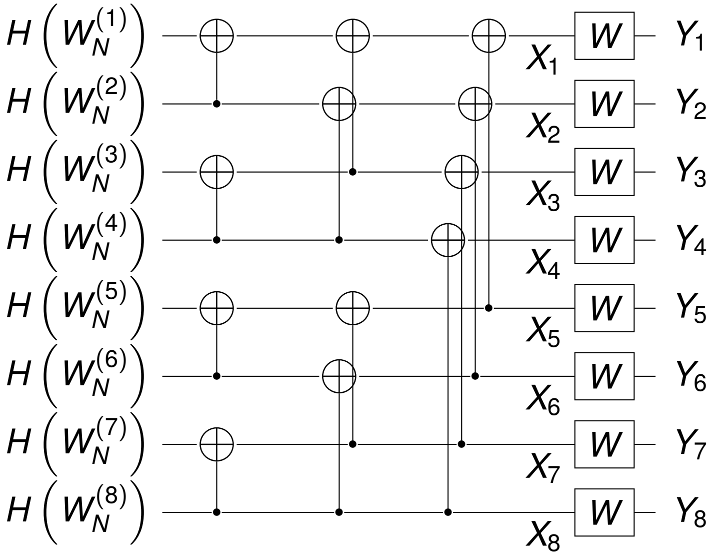


Polar Codes and Density Evolution

$$x_1^N = u_1^N \mathbf{G}_2^{\otimes n} \quad \text{where} \quad \mathbf{G}_2 \triangleq \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad \text{and} \quad N = 2^n$$

Synthesized Channel i

$$W_N^{(i)}\left(y_1^N, u_1^{i-1} | u_i\right)$$



Polar Codes and Density Evolution

$$x_1^N = u_1^N \mathbf{G}_2^{\otimes n} \quad \text{where} \quad \mathbf{G}_2 \triangleq \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad \text{and} \quad N = 2^n$$

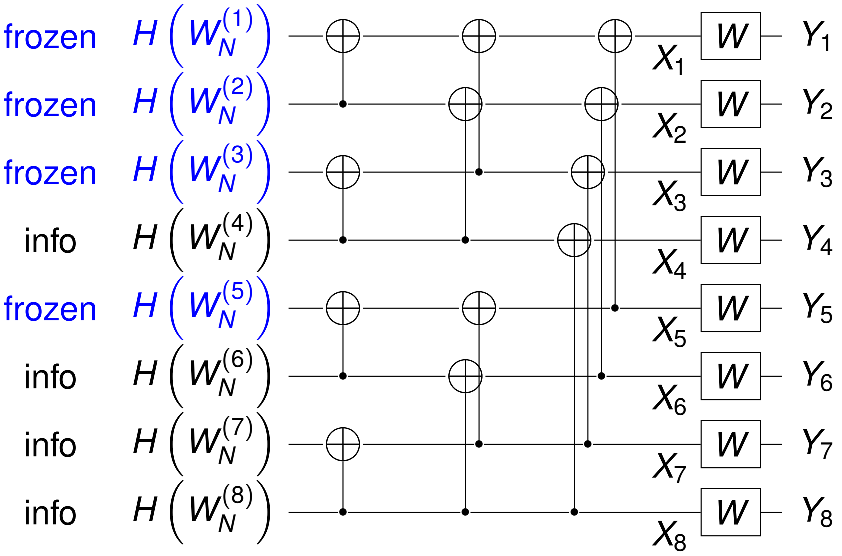
Synthesized Channel i

$$W_N^{(i)}\left(y_1^N, u_1^{i-1} | u_i\right)$$

frozen/info sets

$$\mathcal{F} = \{1, 2, 3, 5\}$$

$$\mathcal{A} = \{4, 6, 7, 8\}$$



Polar Codes and Density Evolution

$$x_1^N = u_1^N \mathbf{G}_2^{\otimes n} \quad \text{where} \quad \mathbf{G}_2 \triangleq \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad \text{and} \quad N = 2^n$$

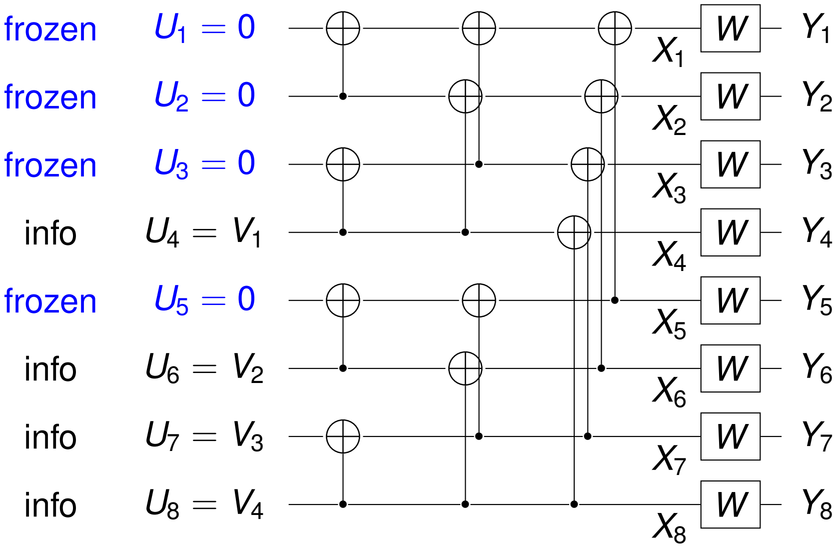
Synthesized Channel i

$$W_N^{(i)}\left(y_1^N, u_1^{i-1} | u_i\right)$$

frozen/info sets

$\mathcal{F} = \{1, 2, 3, 5\}$

$\mathcal{A} = \{4, 6, 7, 8\}$



Dynamic Frozen Bits

- The value of a frozen bit can also be set to a linear combination of previous information bits (rather than a fixed 0 or 1 value) [TM16].

Dynamic Frozen Bits

- The value of a frozen bit can also be set to a linear combination of previous information bits (rather than a fixed 0 or 1 value) [TM16].
- A frozen bit whose value depends on past inputs is called dynamic.

Dynamic Frozen Bits

- The value of a frozen bit can also be set to a linear combination of previous information bits (rather than a fixed 0 or 1 value) [TM16].
- A frozen bit whose value depends on past inputs is called dynamic.
- SC/SCL decoding easily modified for polar codes with dynamic frozen bits.

Dynamic Frozen Bits

- The value of a frozen bit can also be set to a linear combination of previous information bits (rather than a fixed 0 or 1 value) [TM16].
- A frozen bit whose value depends on past inputs is called dynamic.
- SC/SCL decoding easily modified for polar codes with dynamic frozen bits.
- Any binary linear block code can be represented as a polar code with dynamic frozen bits!

Successive Cancellation List Decoding

- Idea of SCL decoding of y_1^N is to recursively $(i = 1, 2, \ldots)$ compute

$$Q_i(\tilde{u}_1^i) \propto \mathbb{P}\left(U_1^i = \tilde{u}_1^i, Y_1^N = y_1^N\right)$$

Successive Cancellation List Decoding

- Idea of SCL decoding of y_1^N is to recursively $(i = 1, 2, \ldots)$ compute

$$Q_i(\tilde{u}_1^i) \propto \mathbb{P}\big(U_1^i = \tilde{u}_1^i, Y_1^N = y_1^N\big)$$

- If $Q_{i-1}(\tilde{u}_1^{i-1})$ known for all $\tilde{u}_1^{i-1} \in \mathcal{U}_{i-1} \subseteq \{0,1\}^{i-1}$, then for $\tilde{u}_i \in \{0,1\}$

$$\begin{aligned} Q_i(\tilde{u}_1^i) &\propto \mathbb{P}\left(U_1^i = \tilde{u}_1^i, Y_1^N = y_1^N\right) \\ &\propto Q_{i-1}(\tilde{u}_1^{i-1})\, \mathbb{P}\big(U_i = \tilde{u}_i \big| Y_1^N = y_1^N, U_1^{i-1} = \tilde{u}_1^{i-1}\big), \end{aligned}$$

  - blue term is computed efficiently by the standard SC decoder
  - gives $Q_i(\tilde{u}_1^i)$ values for $2|\mathcal{U}_{i-1}|$ partial input sequences
  - let $\mathcal{U}_i$ be set of paths after pruning to subset of most likely paths

Successive Cancellation List Decoding

- Idea of SCL decoding of y_1^N is to recursively $(i = 1, 2, \ldots)$ compute

$$Q_i(\tilde{u}_1^i) \propto \mathbb{P}\big(U_1^i = \tilde{u}_1^i, Y_1^N = y_1^N\big)$$

- If $Q_{i-1}(\tilde{u}_1^{i-1})$ known for all $\tilde{u}_1^{i-1} \in \mathcal{U}_{i-1} \subseteq \{0,1\}^{i-1}$, then for $\tilde{u}_i \in \{0,1\}$

$$\begin{aligned} Q_i(\tilde{u}_1^i) &\propto \mathbb{P}\left(U_1^i = \tilde{u}_1^i, Y_1^N = y_1^N\right) \\ &\propto Q_{i-1}(\tilde{u}_1^{i-1})\, \mathbb{P}\big(U_i = \tilde{u}_i \big| Y_1^N = y_1^N, U_1^{i-1} = \tilde{u}_1^{i-1}\big), \end{aligned}$$

  - blue term is computed efficiently by the standard SC decoder
  - gives $Q_i(\tilde{u}_1^i)$ values for $2|\mathcal{U}_{i-1}|$ partial input sequences
  - let $\mathcal{U}_i$ be set of paths after pruning to subset of most likely paths

- After N-th stage, estimate $\hat{u}_1^N \in \mathcal{U}_N$ chosen to maximize $Q_N(\tilde{u}_1^N)$.

Outline

1. Preliminaries

2. Information-Theoretic Perspective on the SCL Decoding

    - Binary Erasure Channel

3. Conclusions

An Information-Theoretic Perspective (1)

Basic Idea: After m steps, consider the conditional entropy $H\left(U_1^m \middle| Y_1^N\right)$.

An Information-Theoretic Perspective (1)

Basic Idea: After m steps, consider the conditional entropy $H\left(U_1^m \middle| Y_1^N\right)$.

The chain rule of entropy implies:

$$H\left(U_1^m \middle| Y_1^N\right) = \sum_{i=1}^{m} H\left(U_i \middle| U_1^{i-1}, Y_1^N\right)$$
$$= \sum_{i=1}^{m} H\left(W_N^{(i)}\right)$$

An Information-Theoretic Perspective (1)

Basic Idea: After m steps, consider the conditional entropy $H\left(U_1^m | Y_1^N\right)$.

The chain rule of entropy implies:

$$H\left(U_1^m | Y_1^N\right) = \sum_{i=1}^{m} H\left(U_i \middle| U_1^{i-1}, Y_1^N\right)$$
$$= \sum_{i=1}^{m} H\left(W_N^{(i)}\right)$$

Note: this ignores frozen bits and will be modified soon!

An Information-Theoretic Perspective (2)

- For the first m input bits, the information/frozen sets are denoted as

$$\mathcal{A}^{(m)} \triangleq \mathcal{A} \cap [m] \quad \text{and} \quad \mathcal{F}^{(m)} \triangleq \mathcal{F} \cap [m]$$

An Information-Theoretic Perspective (2)

- For the first m input bits, the information/frozen sets are denoted as

$$\mathcal{A}^{(m)} \triangleq \mathcal{A} \cap [m] \text{ and } \mathcal{F}^{(m)} \triangleq \mathcal{F} \cap [m]$$

- **Key Idea:** information entropy given frozen bits and difference sequence

$$\bar{D}_m \triangleq H\left(U_{\mathcal{A}^{(m)}} \middle| Y_1^N, U_{\mathcal{F}^{(m)}}\right) \quad \text{and} \quad \Delta_m \triangleq \bar{D}_m - \bar{D}_{m-1}$$

An Information-Theoretic Perspective (2)

- For the first m input bits, the information/frozen sets are denoted as

$$\mathcal{A}^{(m)} \triangleq \mathcal{A} \cap [m] \text{ and } \mathcal{F}^{(m)} \triangleq \mathcal{F} \cap [m]$$

- **Key Idea:** information entropy given frozen bits and difference sequence

$$\bar{D}_m \triangleq H\left(U_{\mathcal{A}^{(m)}} \middle| Y_1^N, U_{\mathcal{F}^{(m)}}\right) \quad \text{and} \quad \Delta_m \triangleq \bar{D}_m - \bar{D}_{m-1}$$

- Experiment: assume U_1^N is uniform and Rx learns frozen bits causally.

An Information-Theoretic Perspective (2)

- For the first m input bits, the information/frozen sets are denoted as

$$\mathcal{A}^{(m)} \triangleq \mathcal{A} \cap [m] \text{ and } \mathcal{F}^{(m)} \triangleq \mathcal{F} \cap [m]$$

- **Key Idea:** information entropy given frozen bits and difference sequence

$$\bar{D}_m \triangleq H\left(U_{\mathcal{A}^{(m)}} \middle| Y_1^N, U_{\mathcal{F}^{(m)}}\right) \quad \text{and} \quad \Delta_m \triangleq \bar{D}_m - \bar{D}_{m-1}$$

- Experiment: assume U_1^N is uniform and Rx learns frozen bits causally.
- If U_m is an information bit, then

  $\Delta_m = H(U_m | Y_1^N, U_1^{m-1})$

An Information-Theoretic Perspective (2)

- For the first m input bits, the information/frozen sets are denoted as

$$\mathcal{A}^{(m)} \triangleq \mathcal{A} \cap [m] \text{ and } \mathcal{F}^{(m)} \triangleq \mathcal{F} \cap [m]$$

- **Key Idea:** information entropy given frozen bits and difference sequence

$$\bar{D}_m \triangleq H\left(U_{\mathcal{A}^{(m)}} \middle| Y_1^N, U_{\mathcal{F}^{(m)}}\right) \quad \text{and} \quad \Delta_m \triangleq \bar{D}_m - \bar{D}_{m-1}$$

- Experiment: assume U_1^N is uniform and Rx learns frozen bits causally.

- If U_m is an information bit, then

$$\Delta_m = H(U_m | Y_1^N, U_1^{m-1})$$

- If U_m is a frozen bit, then

$$0 \geq \Delta_m \geq H\left(U_m \middle| Y_1^N, U_1^{m-1}\right) - 1$$

An Information-Theoretic Perspective (2)

- For the first m input bits, the information/frozen sets are denoted as

$$\mathcal{A}^{(m)} \triangleq \mathcal{A} \cap [m] \text{ and } \mathcal{F}^{(m)} \triangleq \mathcal{F} \cap [m]$$

- **Key Idea:** information entropy given frozen bits and difference sequence

$$\bar{D}_m \triangleq H\left(U_{\mathcal{A}^{(m)}} \middle| Y_1^N, U_{\mathcal{F}^{(m)}}\right) \quad \text{and} \quad \Delta_m \triangleq \bar{D}_m - \bar{D}_{m-1}$$

- Experiment: assume U_1^N is uniform and Rx learns frozen bits causally.

- If U_m is an information bit, then
  $\Delta_m = H(U_m | Y_1^N, U_1^{m-1})$

- If U_m is a frozen bit, then
  $0 \geq \Delta_m \geq H\left(U_m | Y_1^N, U_1^{m-1}\right) - 1$

$$\sum_{i \in \mathcal{A}^{(m)}} H\left(U_i | Y_1^N, U_1^{i-1}\right) - \sum_{i \in \mathcal{F}^{(m)}} \left(1 - H\left(U_i | Y_1^N, U_1^{i-1}\right)\right) \leq \bar{D}_m \leq \sum_{i \in \mathcal{A}^{(m)}} H\left(U_i | Y_1^N, U_1^{i-1}\right)$$

Bounding the List Size

Theorem

Upon observing y_1^N when u_1^N is sent, we define the set (for $\alpha \in (0,1]$)
$\mathcal{S}_\alpha^{(m)}\left(u_1^m, y_1^N\right) \triangleq \left\{\tilde{u}_1^m : \mathbb{P}\left(\tilde{u}_{\mathcal{A}^{(m)}} | y_1^N, \tilde{u}_{\mathcal{F}^{(m)}}\right) \geq \alpha \mathbb{P}\left(u_{\mathcal{A}^{(m)}} | y_1^N, u_{\mathcal{F}^{(m)}}\right)\right\}$. *Then,*

$$\mathrm{E}\left[\log_2 \left|\mathcal{S}_\alpha^{(m)}\right|\right] \leq \bar{D}_m + \log_2 \frac{1}{\alpha} = H\left(U_{\mathcal{A}^{(m)}} \middle| Y_1^N, U_{\mathcal{F}^{(m)}}\right) + \log_2 \frac{1}{\alpha}$$

Bounding the List Size

Theorem

Upon observing y_1^N when u_1^N is sent, we define the set (for $\alpha \in (0,1]$)
$\mathcal{S}_\alpha^{(m)}\left(u_1^m, y_1^N\right) \triangleq \left\{\tilde{u}_1^m : \mathbb{P}\left(\tilde{u}_{\mathcal{A}^{(m)}} | y_1^N, \tilde{u}_{\mathcal{F}^{(m)}}\right) \geq \alpha \mathbb{P}\left(u_{\mathcal{A}^{(m)}} | y_1^N, u_{\mathcal{F}^{(m)}}\right)\right\}$. *Then,*

$$\mathrm{E}\left[\log_2 \left|\mathcal{S}_\alpha^{(m)}\right|\right] \leq \bar{D}_m + \log_2 \frac{1}{\alpha} = H\left(U_{\mathcal{A}^{(m)}} \middle| Y_1^N, U_{\mathcal{F}^{(m)}}\right) + \log_2 \frac{1}{\alpha}$$

Proof.

$$\log_2 \left|\mathcal{S}_\alpha^{(m)}\right| = \log_2 \overbrace{\sum_{\tilde{u}_1^m} \mathbb{1}\left\{\mathbb{P}\left(\tilde{u}_{\mathcal{A}^{(m)}} | y_1^N, \tilde{u}_{\mathcal{F}^{(m)}}\right) \geq \underbrace{\alpha \mathbb{P}\left(u_{\mathcal{A}^{(m)}} \middle| y_1^N, u_{\mathcal{F}^{(m)}}\right)}_{q}\right\}}^{L}$$

$$\leq \log_2 1 / \left(\alpha \mathbb{P}\left(u_{\mathcal{A}^{(m)}} \middle| y_1^N, u_{\mathcal{F}^{(m)}}\right)\right)$$

Valid for all u_1^N and y_1^N; thus, we take expectation over all u_1^m and y_1^N □

Bounding the List Size

Theorem

Upon observing y_1^N when u_1^N is sent, we define the set (for $\alpha \in (0,1]$)
$\mathcal{S}_\alpha^{(m)}\left(u_1^m, y_1^N\right) \triangleq \left\{\tilde{u}_1^m : \mathbb{P}\left(\tilde{u}_{\mathcal{A}^{(m)}} | y_1^N, \tilde{u}_{\mathcal{F}^{(m)}}\right) \geq \alpha \mathbb{P}\left(u_{\mathcal{A}^{(m)}} | y_1^N, u_{\mathcal{F}^{(m)}}\right)\right\}$. *Then,*

$$\mathrm{E}\left[\log_2 |\mathcal{S}_\alpha^{(m)}|\right] \leq \bar{D}_m + \log_2 \frac{1}{\alpha} = H\big(U_{\mathcal{A}^{(m)}} \big| Y_1^N, U_{\mathcal{F}^{(m)}}\big) + \log_2 \frac{1}{\alpha}$$

- For an SCL decoder with max list size L_m during the m-th decoding step,
  - the decoder needs $L_m \geq |\mathcal{S}_1^{(m)}|$ for the true u_1^m to stay on the list
  - Choosing $\alpha < 1$ (say 0.94) captures near misses and matches entropy better.

A Few Remarks

- Our approach currently has two weaknesses:

A Few Remarks

- Our approach currently has two weaknesses:
    - Entropy mainly characterizes typical events but we care about rare events.

A Few Remarks

- Our approach currently has two weaknesses:
  - Entropy mainly characterizes typical events but we care about rare events.
  - The sequence $\bar{D}_m$ is averaged over Y_1^N, i.e., $\bar{D}_m = \sum_{y_1^N} \mathbb{P}(y_1^N) H\left(U_{\mathcal{A}(m)} \middle| Y_1^N = y_1^N, U_{\mathcal{F}(m)}\right)$.

A Few Remarks

- Our approach currently has two weaknesses:
  - Entropy mainly characterizes typical events but we care about rare events.
  - The sequence $\bar{D}_m$ is averaged over Y_1^N, i.e., $\bar{D}_m = \sum_{y_1^N} \mathbb{P}(y_1^N) H\left(U_{\mathcal{A}(m)} \middle| Y_1^N = y_1^N, U_{\mathcal{F}(m)}\right)$.
    But the actual decoder sees a realization $d_m(y_1^N) \triangleq H\left(U_{\mathcal{A}(m)} \middle| Y_1^N = y_1^N, U_{\mathcal{F}(m)}\right)$.

A Few Remarks

- Our approach currently has two weaknesses:
  - Entropy mainly characterizes typical events but we care about rare events.
  - The sequence $\bar{D}_m$ is averaged over Y_1^N, i.e., $\bar{D}_m = \sum_{y_1^N} \mathbb{P}(y_1^N) H\left(U_{\mathcal{A}(m)} | Y_1^N = y_1^N, U_{\mathcal{F}(m)}\right)$.
    But the actual decoder sees a realization $d_m(y_1^N) \triangleq H\left(U_{\mathcal{A}(m)} | Y_1^N = y_1^N, U_{\mathcal{F}(m)}\right)$.
- Significance for code design:

A Few Remarks

- Our approach currently has two weaknesses:
  - Entropy mainly characterizes typical events but we care about rare events.
  - The sequence $\bar{D}_m$ is averaged over Y_1^N, i.e., $\bar{D}_m = \sum_{y_1^N} \mathbb{P}(y_1^N) H\left(U_{\mathcal{A}^{(m)}} | Y_1^N = y_1^N, U_{\mathcal{F}^{(m)}}\right)$.
    But the actual decoder sees a realization $d_m(y_1^N) \triangleq H\left(U_{\mathcal{A}^{(m)}} | Y_1^N = y_1^N, U_{\mathcal{F}^{(m)}}\right)$.
- Significance for code design:
  - A first-order code design criterion can be seen as $\log_2 L_m \geq d_m$.

A Few Remarks

- Our approach currently has two weaknesses:
  - Entropy mainly characterizes typical events but we care about rare events.
  - The sequence $\bar{D}_m$ is averaged over Y_1^N, i.e., $\bar{D}_m = \sum_{y_1^N} \mathbb{P}(y_1^N) H\left(U_{\mathcal{A}^{(m)}} | Y_1^N = y_1^N, U_{\mathcal{F}^{(m)}}\right)$.
    But the actual decoder sees a realization $d_m(y_1^N) \triangleq H\left(U_{\mathcal{A}^{(m)}} | Y_1^N = y_1^N, U_{\mathcal{F}^{(m)}}\right)$.
- Significance for code design:
  - A first-order code design criterion can be seen as $\log_2 L_m \geq d_m$.
  - Based on this, the improved code designs will be reported.

Dynamic Reed-Muller Codes

- dRM code ensemble [CNP20, CP21]:
  - Let $\mathcal{A}$ be the information indices of an RM code.
  - u_i is an information bit if $i \in \mathcal{A}$.
  - $u_i = \sum_{j \in \mathcal{A}^{(i)}} A_{ij} u_j$ if $i \in \mathcal{F}$ where A_{ij} iid $\sim$ Bernoulli(0.5).

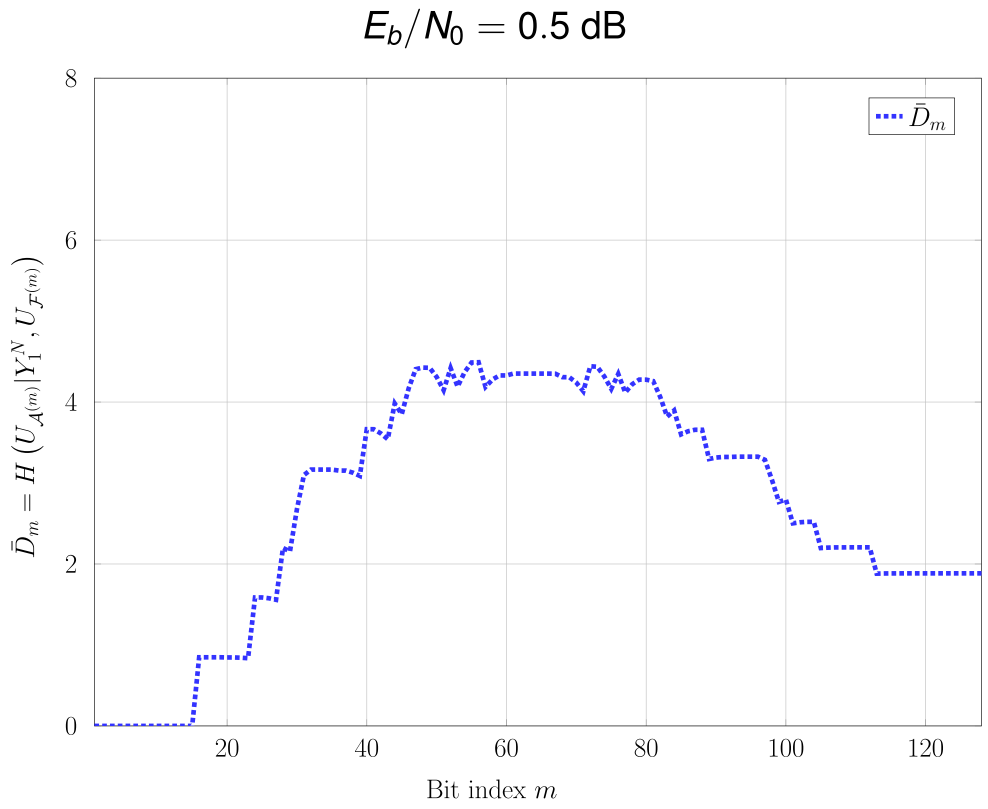
Dynamic Reed-Muller Codes

- dRM code ensemble [CNP20, CP21]:
  - Let $\mathcal{A}$ be the information indices of an RM code.
  - u_i is an information bit if $i \in \mathcal{A}$.
  - $u_i = \sum_{j \in \mathcal{A}^{(i)}} A_{ij} u_j$ if $i \in \mathcal{F}$ where A_{ij} iid $\sim$ Bernoulli(0.5).
- Closely related to polarization-adjusted convolutional (PAC) codes [Arı19].

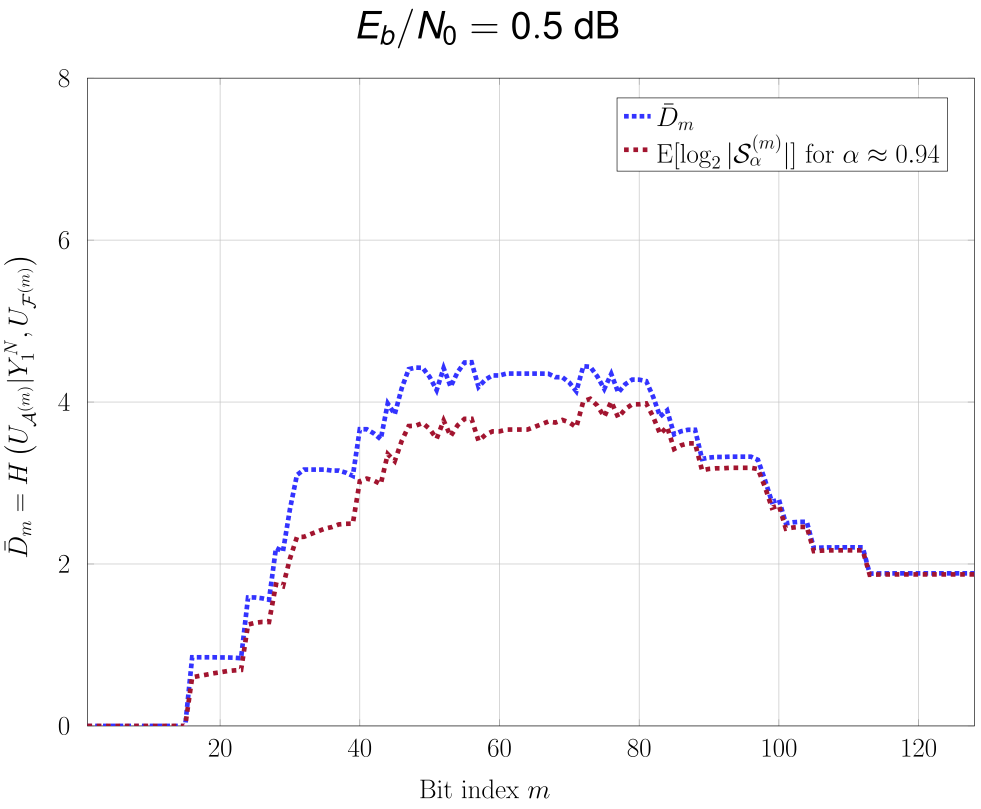
Dynamic Reed-Muller Codes

- dRM code ensemble [CNP20, CP21]:
  - Let $\mathcal{A}$ be the information indices of an RM code.
  - u_i is an information bit if $i \in \mathcal{A}$.
  - $u_i = \sum_{j \in \mathcal{A}^{(i)}} A_{ij} u_j$ if $i \in \mathcal{F}$ where A_{ij} iid $\sim$ Bernoulli(0.5).
- Closely related to polarization-adjusted convolutional (PAC) codes [Arı19].
- PAC is outperformed slightly by (random instances of) dRM code under SCL decoding with the same list sizes.

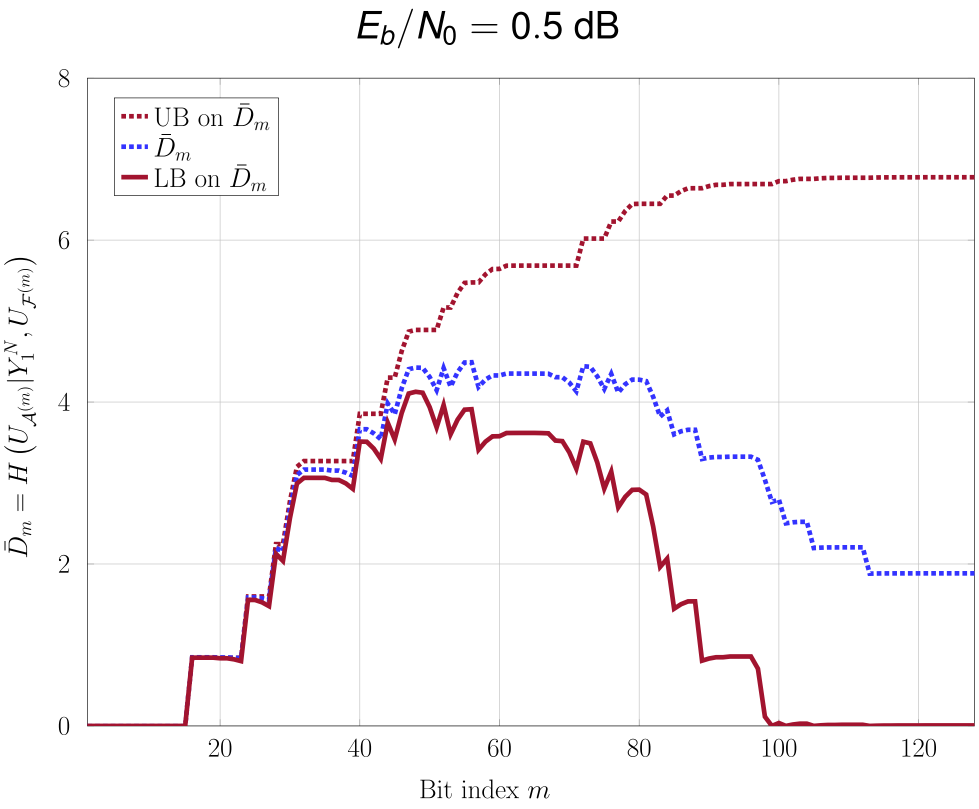
$(128, 64)$ dRM Code over the AWGN Channel



$(128, 64)$ dRM Code over the AWGN Channel



$(128, 64)$ dRM Code over the AWGN Channel

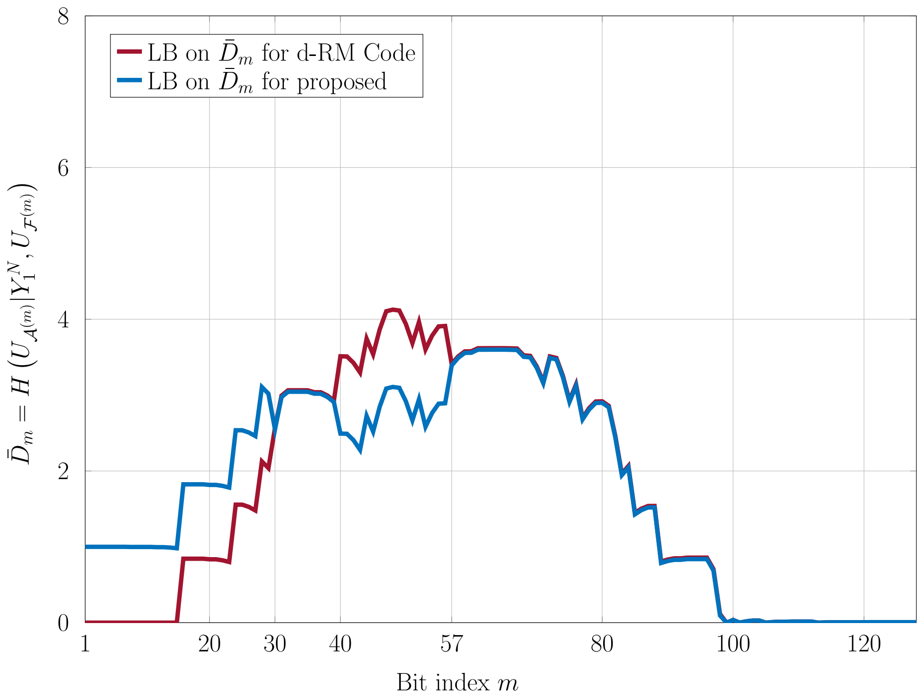


$(128, 64)$ Proposed vs dRM Code over the AWGN Channel

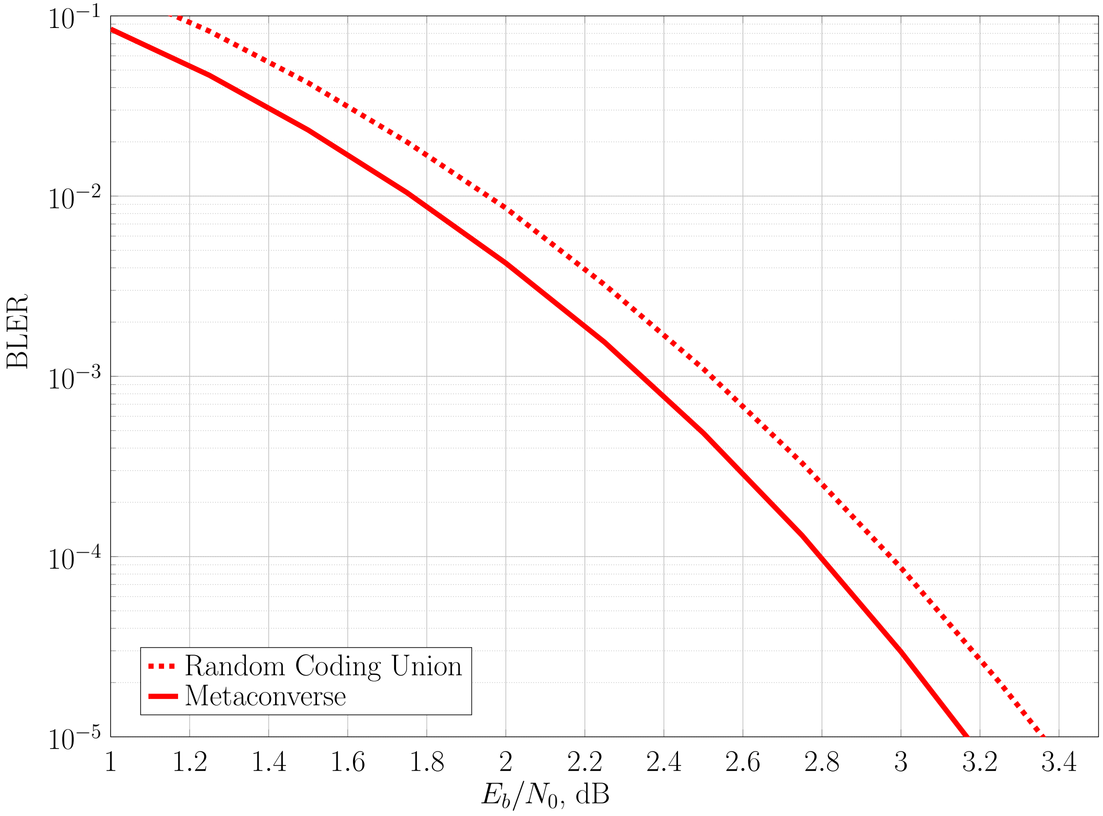
Proposed Code

- $u_{\{30,40\}}$ dynamic frozen bits
- $u_{\{1,57\}}$ info. bits

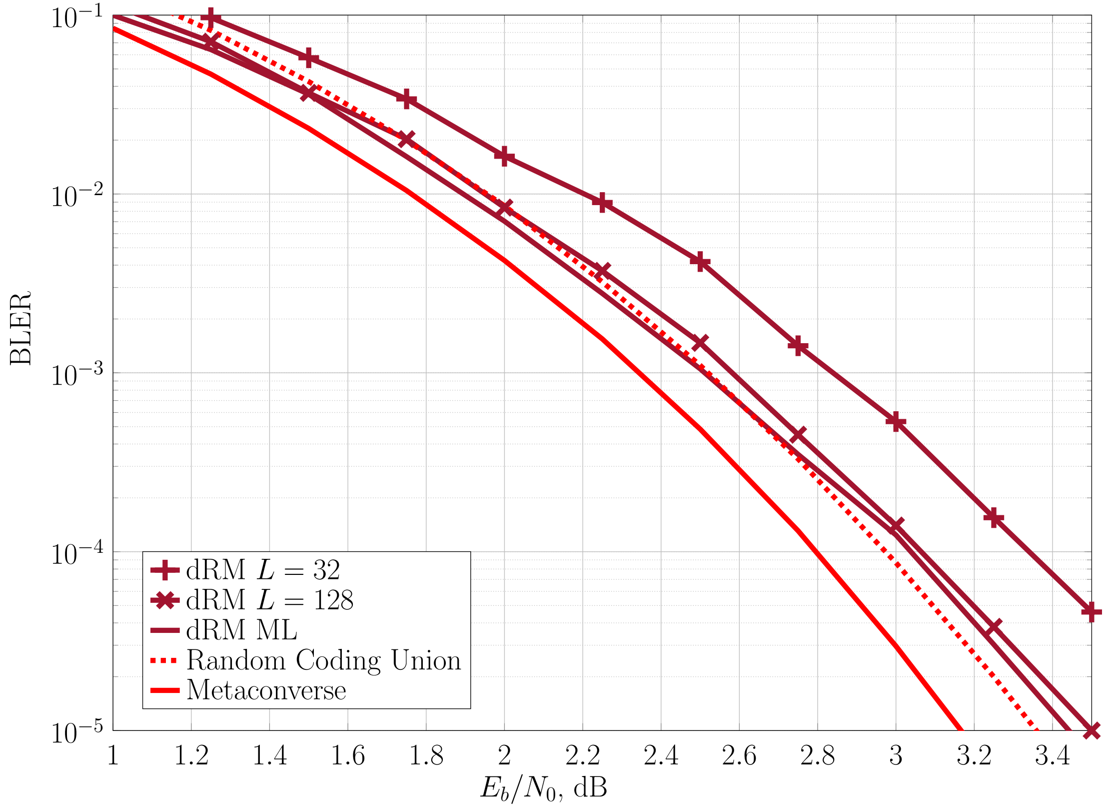
$E_b/N_0 = 0.5$ dB



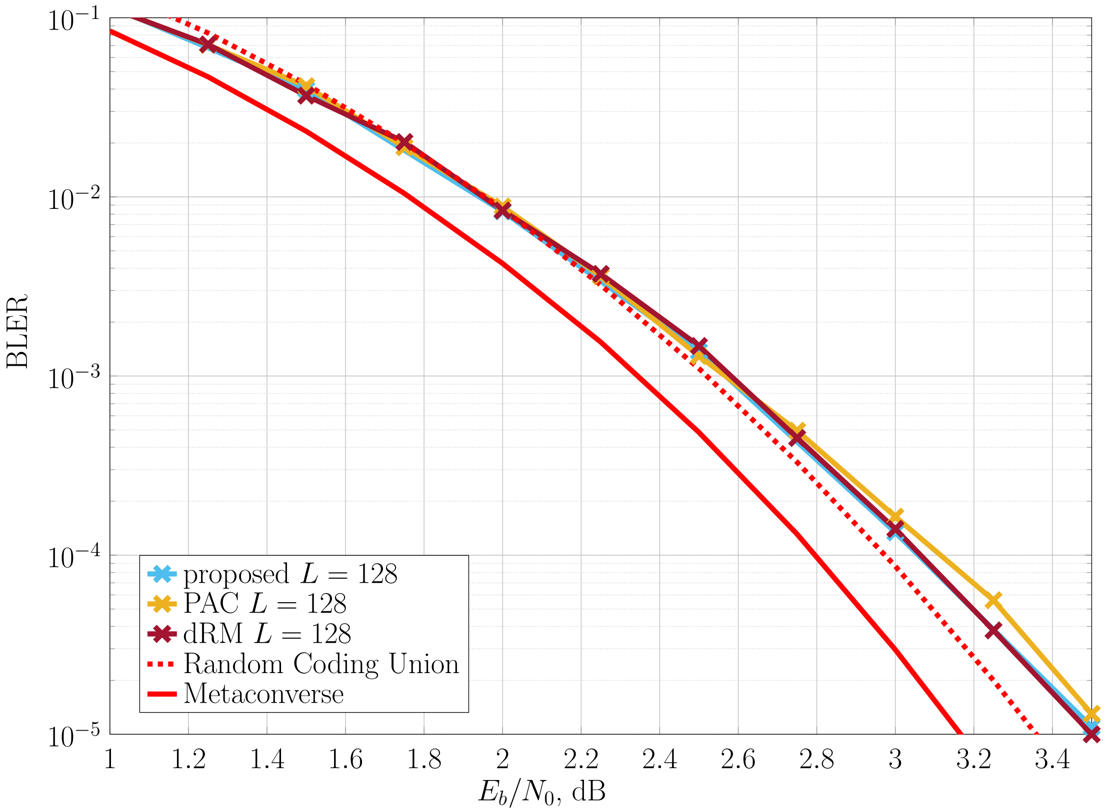
$(128, 64)$ Codes over the AWGN Channel



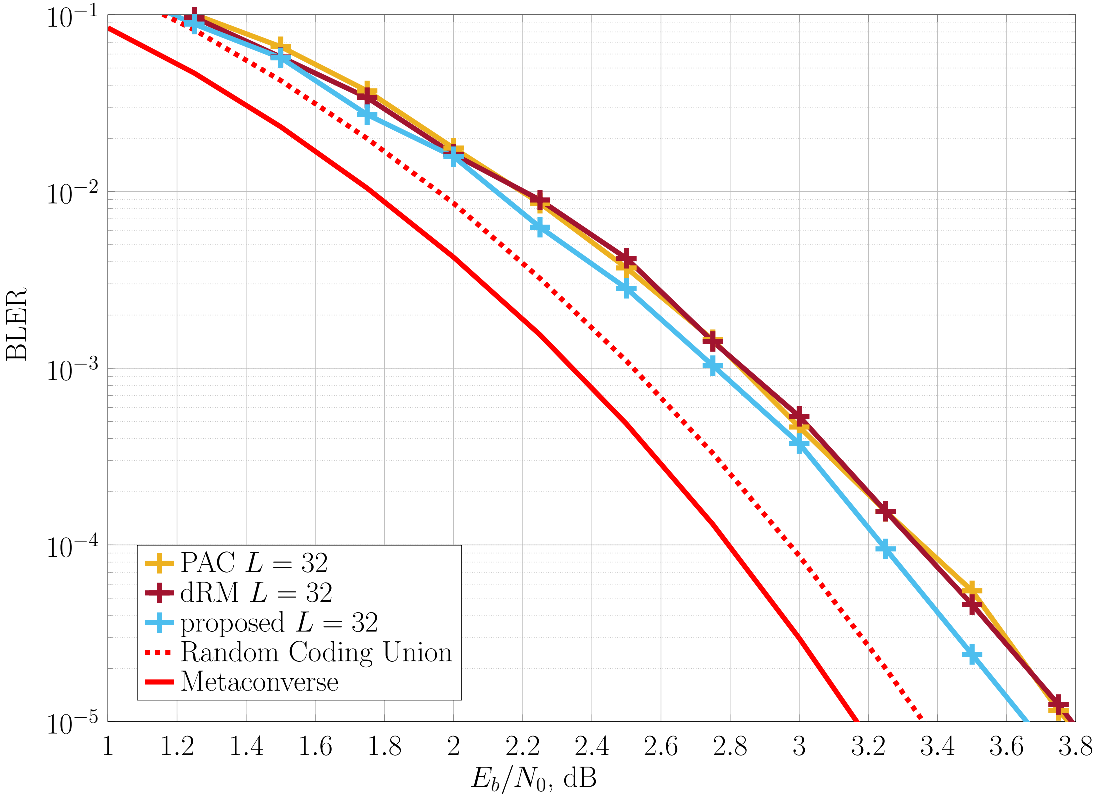
(128, 64) Codes over the AWGN Channel



$(128, 64)$ Codes over the AWGN Channel

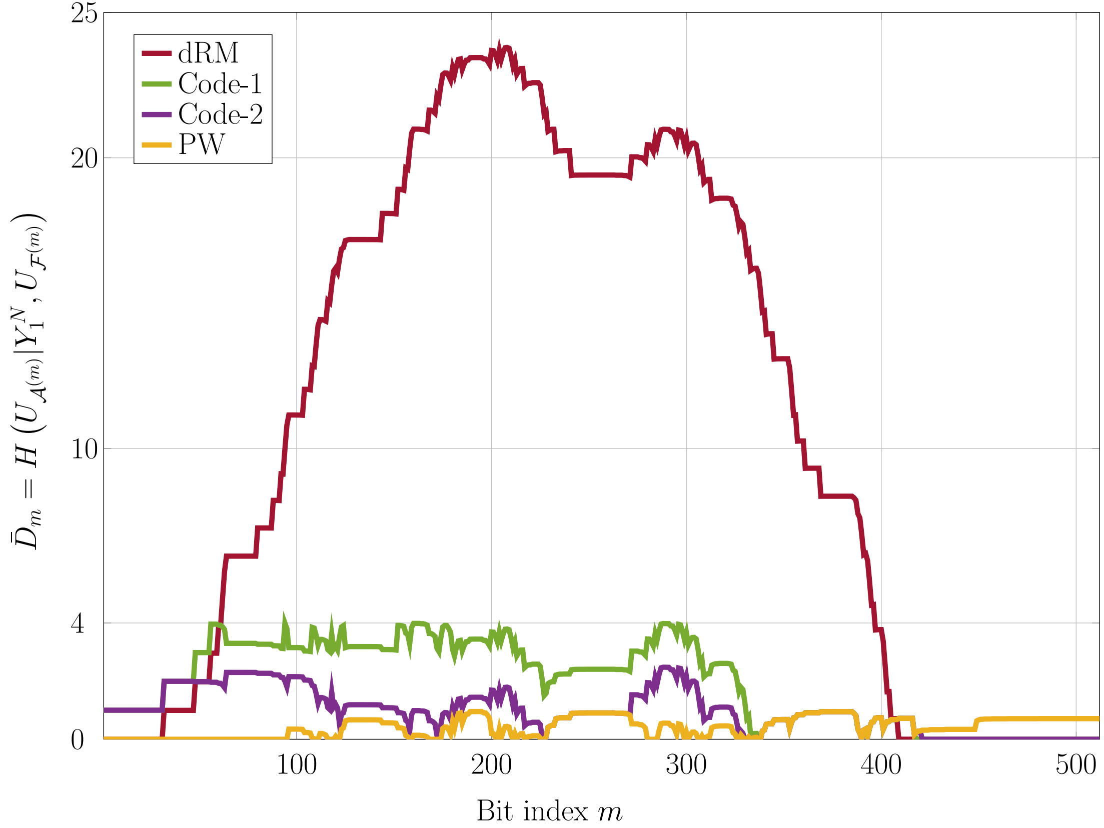


$(128, 64)$ Codes over the AWGN Channel

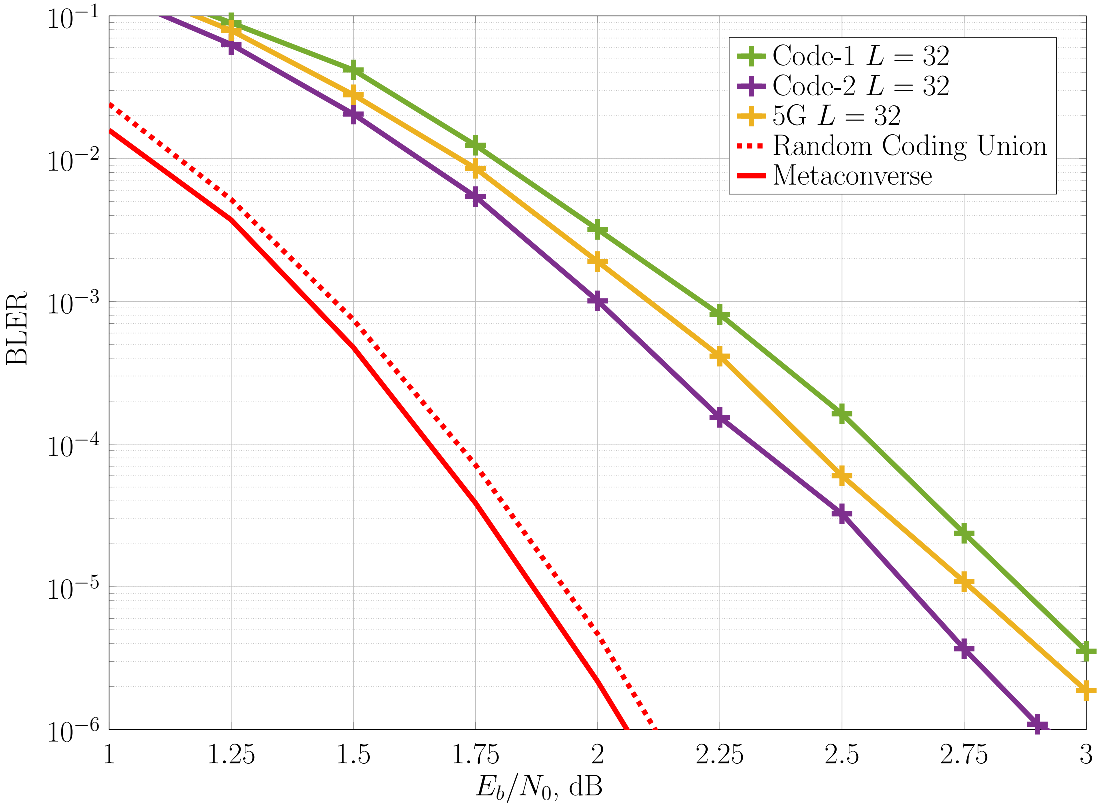


(512, 256) Proposed vs dRM Codes over the AWGN Channel

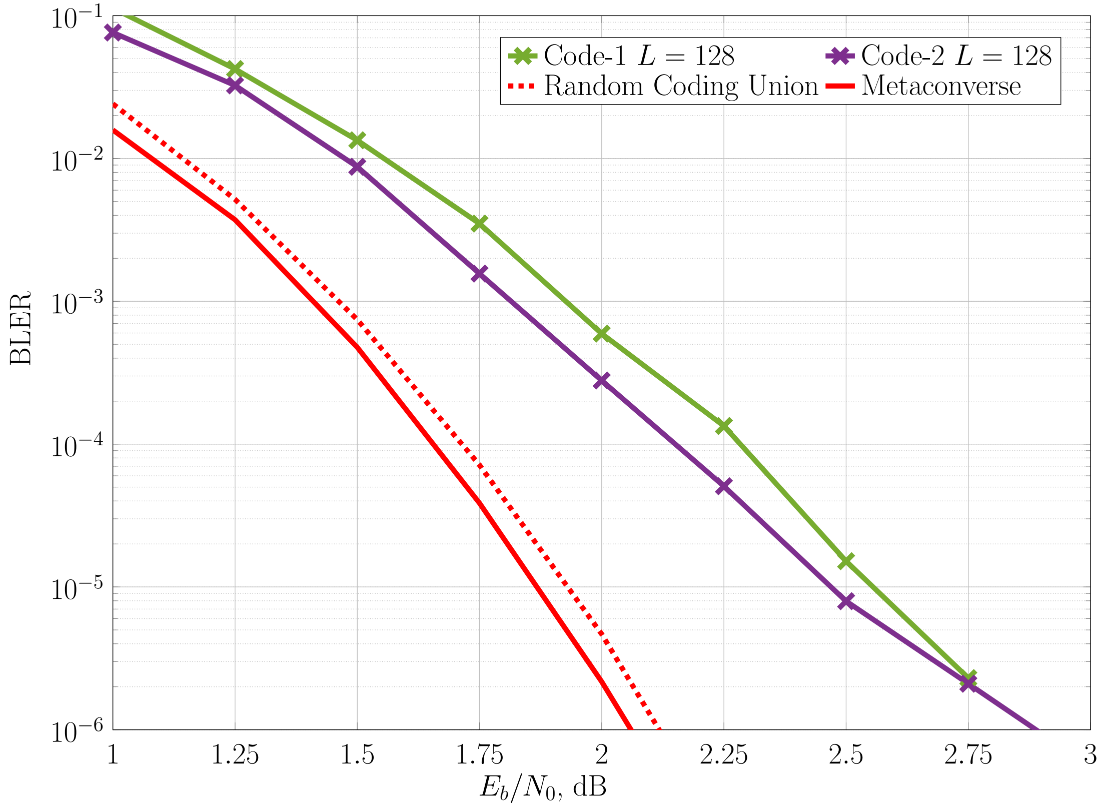
$E_b/N_0 = 0.5$ dB



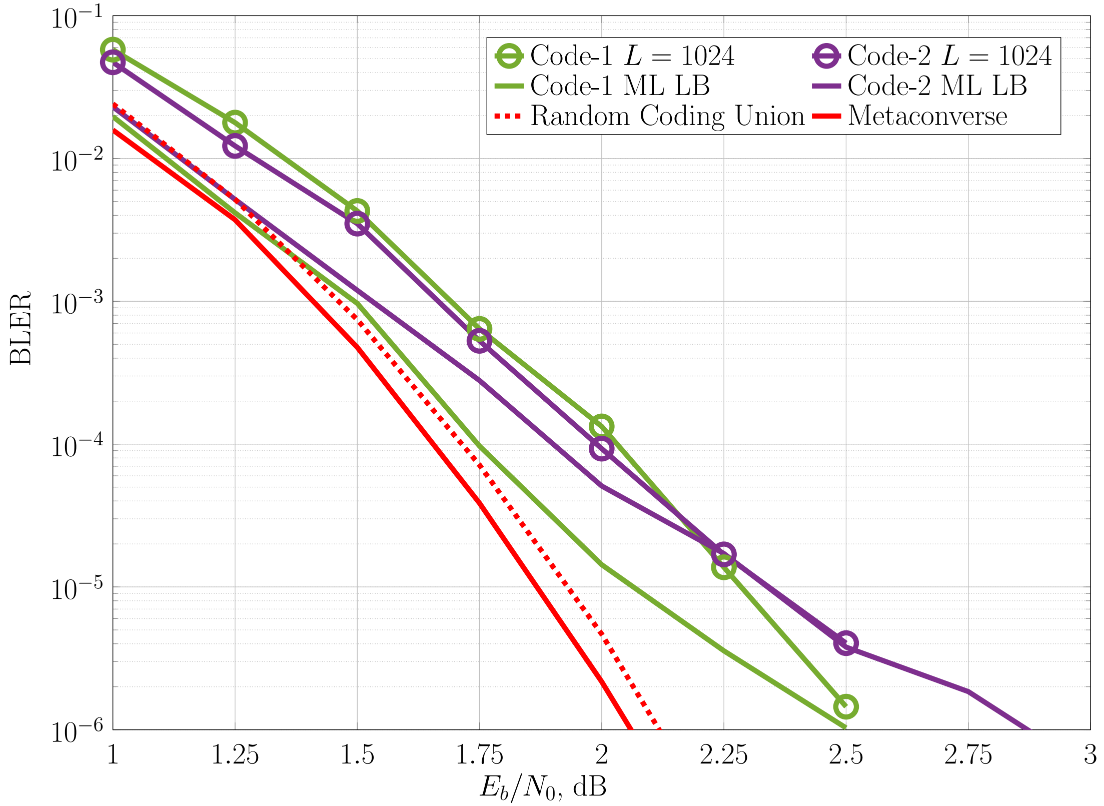
(512, 256) Codes over the AWGN Channel



$(512, 256)$ Codes over the AWGN Channel



$(512, 256)$ Codes over the AWGN Channel



Outline

The Uncertainty Dimension

- For a fixed y_1^N, the subspace dimension is

$$d_m(y_1^N) = H\left(U_{\mathcal{A}^{(m)}} \middle| Y_1^N = y_1^N, U_{\mathcal{F}^{(m)}}\right)$$

The Uncertainty Dimension

- For a fixed y_1^N, the subspace dimension is

$$d_m(y_1^N) = H\left(U_{\mathcal{A}^{(m)}} \middle| Y_1^N = y_1^N, U_{\mathcal{F}^{(m)}}\right)$$

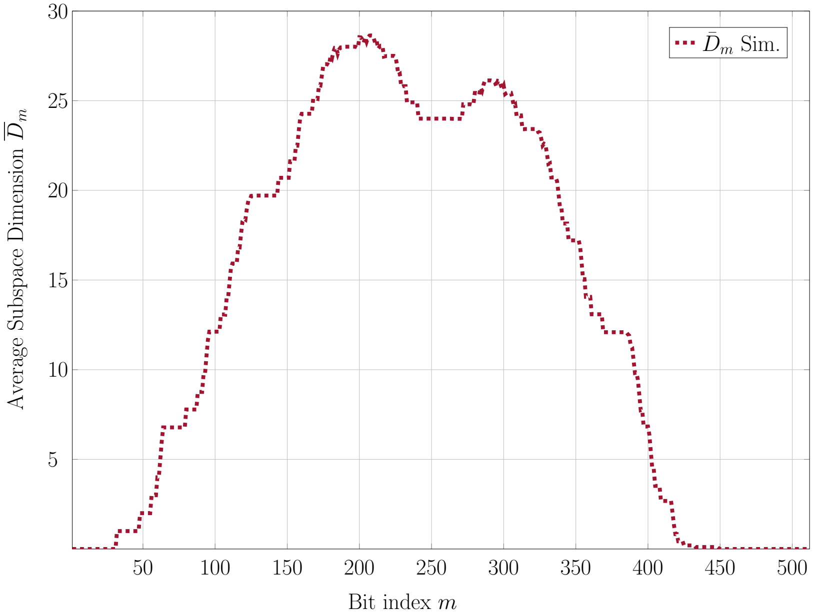
- Let $D_m = d_m(Y_1^N)$ denote corresponding random value at step m.

The Uncertainty Dimension

- For a fixed y_1^N, the subspace dimension is

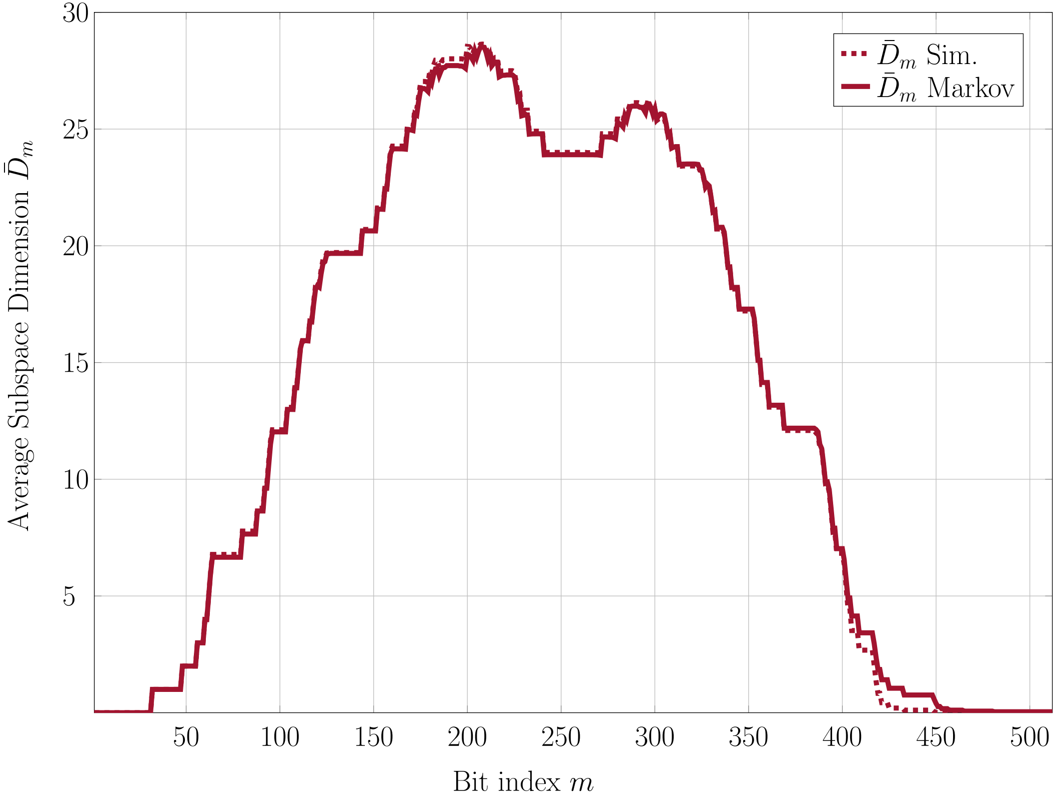
$$d_m(y_1^N) = H\left(U_{\mathcal{A}^{(m)}} \middle| Y_1^N = y_1^N, U_{\mathcal{F}^{(m)}}\right)$$

- Let $D_m = d_m(Y_1^N)$ denote corresponding random value at step m.



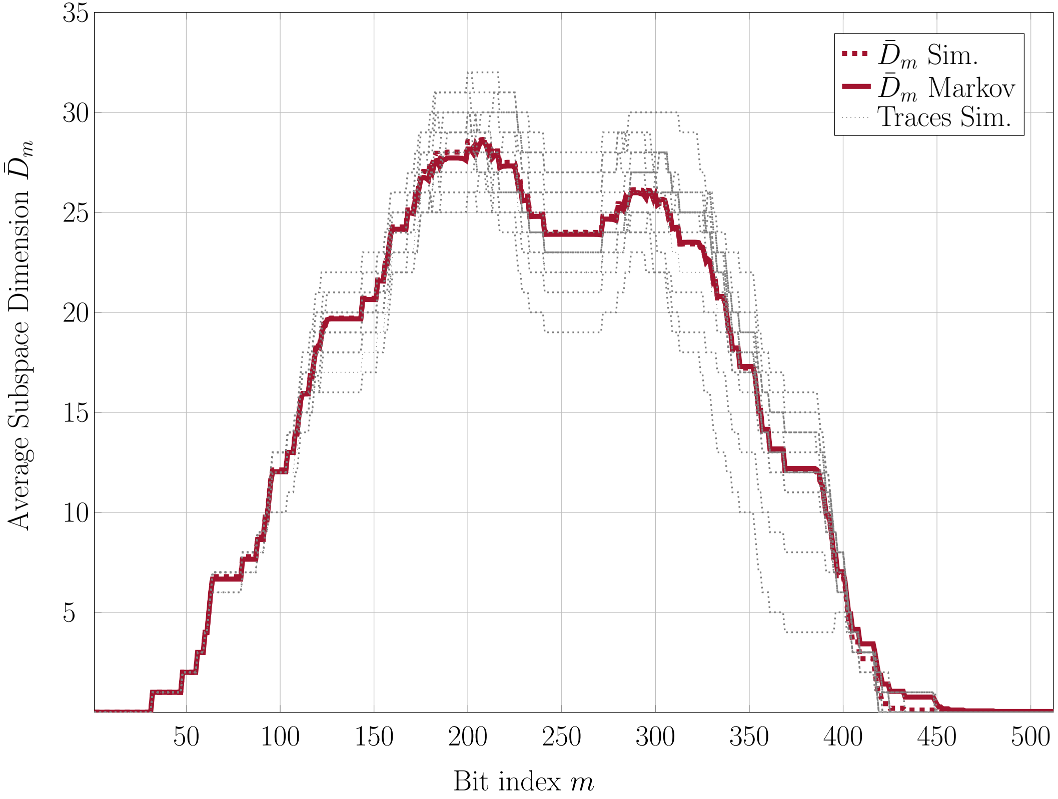
$(512, 256)$ dRM Code

A fixed-weight BEC with exactly $\texttt{round}(512 \times 0.48) = 246$ erasures



$(512, 256)$ dRM Code

A fixed-weight BEC with exactly $\texttt{round}(512 \times 0.48) = 246$ erasures



Concentration of the Subspace Dimension

Theorem

The subspace dimension D_m for a particular random realization Y_1^N concentrates around the mean $\bar{D}_m$ for sufficiently large block lengths [CP21], i.e., for any $\beta > 0$, we have

$$\mathbb{P}\left\{\frac{1}{N}|D_m - \bar{D}_m| > \beta\right\} \leq 2\exp\left(-\frac{\beta^2}{2}N\right). \tag{1}$$

Concentration of the Subspace Dimension

Theorem

The subspace dimension D_m for a particular random realization Y_1^N concentrates around the mean $\bar{D}_m$ for sufficiently large block lengths [CP21], i.e., for any $\beta > 0$, we have

$$\mathbb{P}\left\{\frac{1}{N}|D_m - \bar{D}_m| > \beta\right\} \leq 2\exp\left(-\frac{\beta^2}{2}N\right). \quad (1)$$

Proof.

Key observation: at any decoding stage, the subspace dimension satisfies Lipschitz$-$1 condition: For all $i \in [N]$ and all values y_1^N and $\tilde{y}_i$, we have

$$|d_m(y_1^N) - d_m(y_1^{i-1}, \tilde{y}_i, y_{i+1}^N)| \leq 1.$$

Then, use Azuma-Hoeffding inequality by forming a Doob's Martingale. □

Concentration of the Subspace Dimension

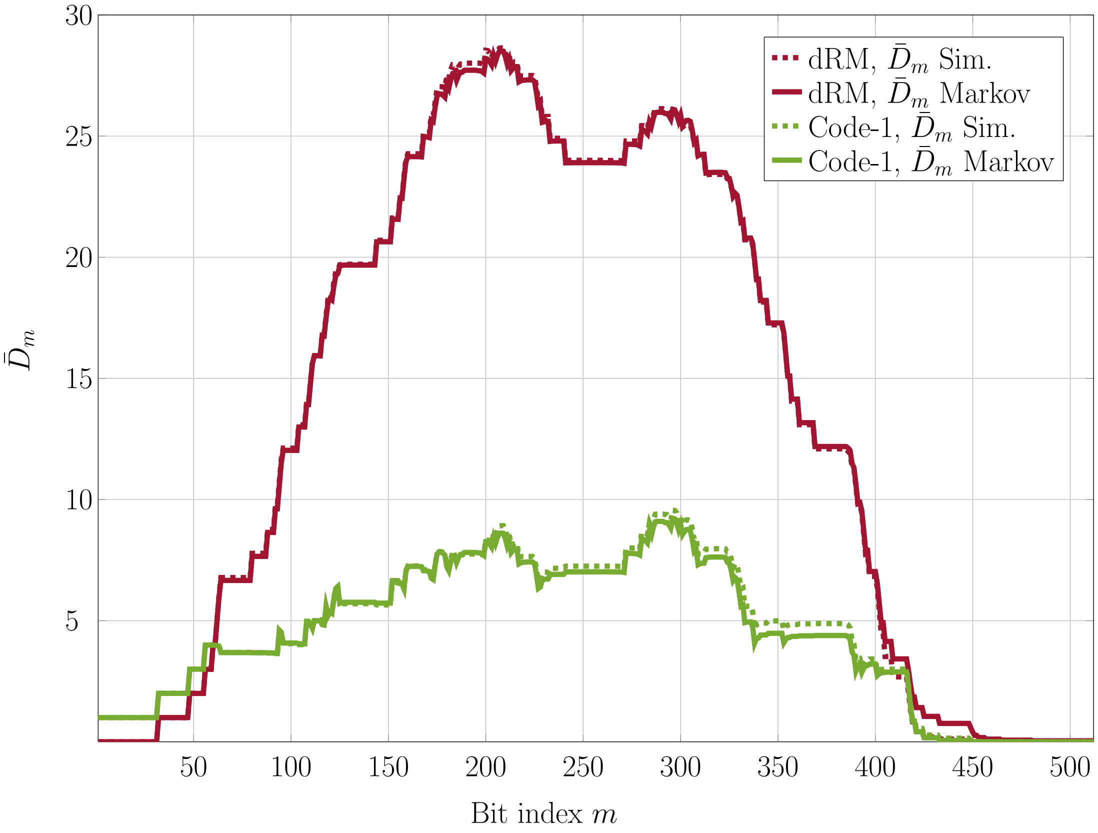
Theorem

The subspace dimension D_m for a particular random realization Y_1^N concentrates around the mean $\bar{D}_m$ for sufficiently large block lengths [CP21], i.e., for any $\beta > 0$, we have

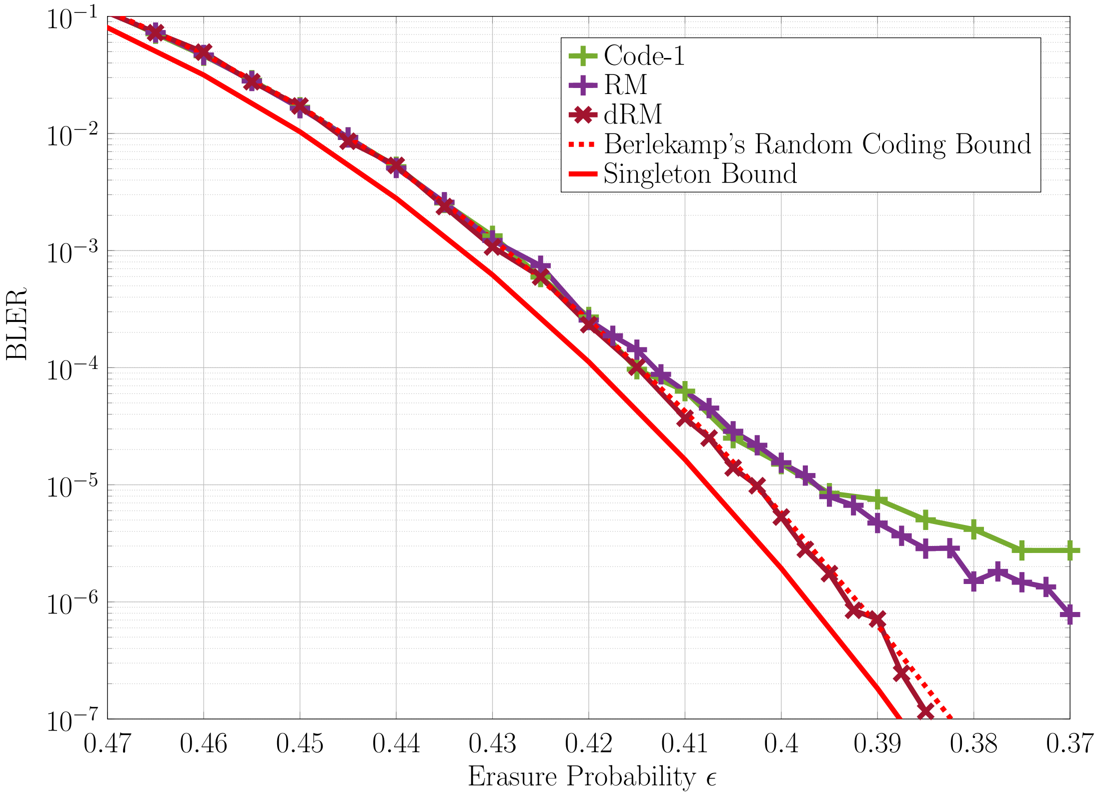
$$\mathbb{P}\left\{\frac{1}{N}|D_m - \bar{D}_m| > \beta\right\} \leq 2\exp\left(-\frac{\beta^2}{2}N\right). \quad (1)$$

- We may use the theorem above to give bounds on the average complexity of ML decoding of a given code implemented via SCI decoding.
- Extension to more general class of channels follows [CP21].

(512, 256) Codes over the BEC



(512, 256) Codes over the BEC



Outline

Summary (1)

AWGN Channel

- "What list size is sufficient to approach maximum-likelihood (ML) decoding performance under an SCL decoder?"
  - Information theory provides some useful measures.

Summary (1)

AWGN Channel

- “What list size is sufficient to approach maximum-likelihood (ML) decoding performance under an SCL decoder?”
  - Information theory provides some useful measures.
- The analysis leads to improved code designs (in comparison to the existing good codes, e.g., the PAC code [Arı19] or 5G codes) especially in under SCL decoding with practical list sizes, e.g., 32 or 128.
  - Not only for $N = 128$ but also for $N = 512$.

Summary (1)

AWGN Channel

- "What list size is sufficient to approach maximum-likelihood (ML) decoding performance under an SCL decoder?"
  - Information theory provides some useful measures.
- The analysis leads to improved code designs (in comparison to the existing good codes, e.g., the PAC code [Arı19] or 5G codes) especially in under SCL decoding with practical list sizes, e.g., 32 or 128.
  - Not only for $N = 128$ but also for $N = 512$.
- Recent advances (dynamic frozen bits + SCL) in polar codes allow performance near random coding union bound for $(128, 64)$ with moderate complexity.
  - Moderate-length regime, e.g., $2048 \geq N \geq 256$, has more margin to improve.

Summary (2)

BEC

- The logarithm of the random required list size is shown to concentrate around the mean and a simple but accurate approximation of this mean is provided for the BEC.
  - Holds for more general class of channels (see [CP21] for details).

Summary (2)

BEC

- The logarithm of the random required list size is shown to concentrate around the mean and a simple but accurate approximation of this mean is provided for the BEC.
  - Holds for more general class of channels (see [CP21] for details).
- Designed code performs as well as the RM code for frame error rates equal to 10^{-5} or higher with much lower average list size (or, inactivations as it is BEC).

Summary (2)

BEC

- The logarithm of the random required list size is shown to concentrate around the mean and a simple but accurate approximation of this mean is provided for the BEC.
  - Holds for more general class of channels (see [CP21] for details).
- Designed code performs as well as the RM code for frame error rates equal to 10^{-5} or higher with much lower average list size (or, inactivations as it is BEC).

Outlook and Future Work

- The concentration of the random subspace dimension makes the average analysis meaningful; hence, we may upper bound the average complexity of SC inactivation decoding (for the BEC).

Summary (2)

BEC

- The logarithm of the random required list size is shown to concentrate around the mean and a simple but accurate approximation of this mean is provided for the BEC.
  - Holds for more general class of channels (see [CP21] for details).
- Designed code performs as well as the RM code for frame error rates equal to 10^{-5} or higher with much lower average list size (or, inactivations as it is BEC).

Outlook and Future Work

- The concentration of the random subspace dimension makes the average analysis meaningful; hence, we may upper bound the average complexity of SC inactivation decoding (for the BEC).
- Constructive algorithm to design longer codes with good performance vs. complexity trade-off under SCL decoding.

Thanks

References I

[Arı09] E. Arıkan.
Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels.
IEEE Trans. Inf. Theory, 55(7):3051–3073, July 2009.

[Arı19] E. Arıkan.
From sequential decoding to channel polarization and back again.
CoRR, abs/1908.09594, 2019.

[CNP20] M. C. Coşkun, J. Neu, and H. D. Pfister.
Successive cancellation inactivation decoding for modified Reed-Muller and eBCH codes.
In *IEEE Int. Symp. Inf. Theory*, 2020.

[CP20] M. C. Coşkun and H. D. Pfister.
Bounds on the list size of successive cancellation list decoding.
In *2020 Int. Conf. on Signal Process. and Commun. (SPCOM)*, pages 1–5, 2020.

[CP21] Mustafa Cemil Coşkun and Henry D. Pfister.
An information-theoretic perspective on successive cancellation list decoding and polar code design.
IEEE Trans. Inf. Theory, submitted, 2021.

[TM16] P. Trifonov and V. Miloslavskaya.
Polar subcodes.
IEEE J. Sel. Areas Commun., 34(2):254–266, Feb. 2016.

[TV15] I. Tal and A. Vardy.
List decoding of polar codes.
IEEE Trans. Inf. Theory, 61(5):2213–2226, May 2015.