COCO 2020
July 16, 2020

# Bounds on the List Size of Successive Cancellation List Decoding

Mustafa Cemil Coşkun[1,2]

Joint work with H. D. Pfister[3]

[1]German Aerospace Center

[2]Technical University of Munich

[3]Duke University

COCO 2020
July 16, 2020

# Average List Size of Successive Cancellation Inactivation Decoding

Mustafa Cemil Coşkun[1,2]

Joint work with H. D. Pfister[3]

[1]German Aerospace Center

[2]Technical University of Munich

[3]Duke University

# Outline

- Introduction
- Preliminaries
- The Binary Erasure Channel
- Numerical Results
- Conclusions

# 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

- Capacity-achieving on binary memoryless symmetric (BMS) channels with low encoding/decoding complexity[1]

[1] E. Arıkan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," T-IT, Jul. 2009.

# 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

- Capacity-achieving on binary memoryless symmetric (BMS) channels with low encoding/decoding complexity[1]
- But successive cancellation (SC) decoding performs poorly for small blocks

[1] E. Arıkan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," T-IT, Jul. 2009.

# 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**

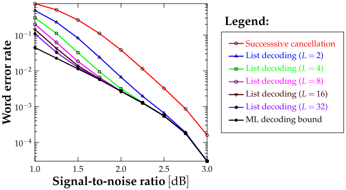



Fig. 1. List-decoding performance for a polar code of length $n = 2048$ and rate $R = 0.5$ on the BPSK-modulated Gaussian channel. The code was constructed using the methods of [15], with optimization for $E_b/N_0 = 2$ dB.

- SC list (SCL) decoding with CRC and large list-size performs very well and matches maximum-likelihood (ML)[2]

[2] I. Tal, A. Vardy, "List decoding of polar codes," T-IT, May 2015.

# 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**

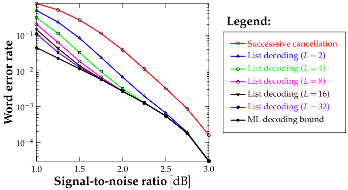



Fig. 1. List-decoding performance for a polar code of length $n = 2048$ and rate $R = 0.5$ on the BPSK-modulated Gaussian channel. The code was constructed using the methods of [15], with optimization for $E_b/N_0 = 2$ dB.

- SC list (SCL) decoding with CRC and large list-size performs very well and matches maximum-likelihood (ML)[2]
- Can also be used to decode other codes (e.g., Reed–Muller codes)

[2] I. Tal, A. Vardy, "List decoding of polar codes," T-IT, May 2015.

# 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?
  - Can be attacked via simulation but quite complex for long codes and lists
  - But, simulation unlikely to provide insight into the question
  - A theoretical answer might enable better code designs for SCL decoding

# Summary of Results

- In this talk, we focus on the binary erasure channel (BEC)

[3] M. C. Coșkun, J. Neu, H. D. Pfister, "Successive Cancellation Inactivation Decoding for Modified Reed-Muller and eBCH Codes," ISIT, Jun. 2020.

# Summary of Results

- In this talk, we focus on the binary erasure channel (BEC)
  - We consider the SC inactivation (SCI) decoder[3], which stores a basis for the subspace of all valid partial sequences

[3] M. C. Coșkun, J. Neu, H. D. Pfister, "Successive Cancellation Inactivation Decoding for Modified Reed-Muller and eBCH Codes," ISIT, Jun. 2020.

# Summary of Results

- In this talk, we focus on the binary erasure channel (BEC)
  - We consider the SC inactivation (SCI) decoder[3], which stores a basis for the subspace of all valid partial sequences
  - The random dimension sequence of this subspace can be approximated by a Markov chain

[3] M. C. Coşkun, J. Neu, H. D. Pfister, "Successive Cancellation Inactivation Decoding for Modified Reed-Muller and eBCH Codes," ISIT, Jun. 2020.

# Summary of Results

- In this talk, we focus on the binary erasure channel (BEC)
  - We consider the SC inactivation (SCI) decoder[3], which stores a basis for the subspace of all valid partial sequences
  - The random dimension sequence of this subspace can be approximated by a Markov chain
  - For a fixed number of erasures, the approximation is reasonably accurate

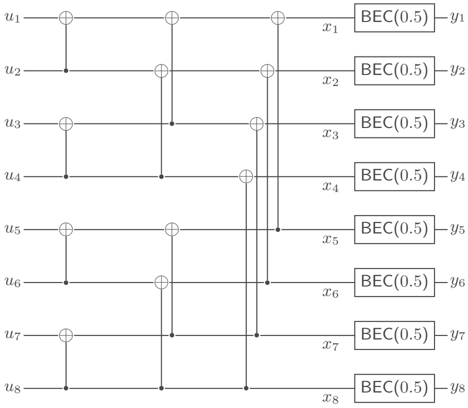
[3] M. C. Coşkun, J. Neu, H. D. Pfister, "Successive Cancellation Inactivation Decoding for Modified Reed-Muller and eBCH Codes," ISIT, Jun. 2020.

# Polar Codes and Density Evolution

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

# Polar Codes and Density Evolution

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



# Polar Codes and Density Evolution

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

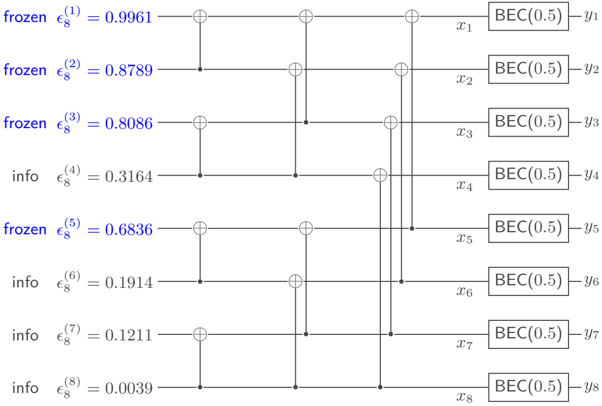
Effective Channel $i$

$$\mathrm{W}_N^{(i)}(y_1^n, u_1^{i-1} | u_i)$$

$$H(\mathrm{W}_N^{(i)}) \triangleq H(U_i | Y_1^N, U_1^{i-1}) = \epsilon_N^{(i)}$$

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

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



# 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)[4]

[4] P. Trifonov, V. Miloslavskaya, "Polar subcodes," J-SAC, Feb. 2016.

# 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)[4]
- A frozen bit whose value depends on past inputs is called dynamic

[4] P. Trifonov, V. Miloslavskaya, "Polar subcodes," J-SAC, Feb. 2016.

DLR

# 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)[4]
- A frozen bit whose value depends on past inputs is called dynamic
- SC/SCL decoding easily modified for polar codes with dynamic frozen bits

[4] P. Trifonov, V. Miloslavskaya, "Polar subcodes," J-SAC, Feb. 2016.

# 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)[4]
- 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!

[4] P. Trifonov, V. Miloslavskaya, "Polar subcodes," J-SAC, Feb. 2016.

# The Binary Erasure Channel

- For linear codes on the BEC, uncertainty in the information bits can always be represented by an affine subspace

# The Binary Erasure Channel

- For linear codes on the BEC, uncertainty in the information bits can always be represented by an affine subspace
- Full SCL decoder for the BEC lists all valid sequences in this subspace

# The Binary Erasure Channel

- For linear codes on the BEC, uncertainty in the information bits can always be represented by an affine subspace
- Full SCL decoder for the BEC lists all valid sequences in this subspace
- The SCI decoder instead stores a basis:

# The Binary Erasure Channel

- For linear codes on the BEC, uncertainty in the information bits can always be represented by an affine subspace
- Full SCL decoder for the BEC lists all valid sequences in this subspace
- The SCI decoder instead stores a basis:
  - If SC decoding step outputs erasure, inactivate the bit and add basis vector

# The Binary Erasure Channel

- For linear codes on the BEC, uncertainty in the information bits can always be represented by an affine subspace
- Full SCL decoder for the BEC lists all valid sequences in this subspace
- The SCI decoder instead stores a basis:
  - If SC decoding step outputs erasure, inactivate the bit and add basis vector
  - Later messages in decoder are functions of inactivated bits (i.e., basis vectors)

# The Binary Erasure Channel

- For linear codes on the BEC, uncertainty in the information bits can always be represented by an affine subspace
- Full SCL decoder for the BEC lists all valid sequences in this subspace
- The SCI decoder instead stores a basis:
  - If SC decoding step outputs erasure, inactivate the bit and add basis vector
  - Later messages in decoder are functions of inactivated bits (i.e., basis vectors)
  - If SC decoding of frozen bit is an unerased message, then resulting equation may allow one to consolidate the basis (i.e., remove a basis vector)

# The Uncertainty Dimension

- For a fixed $y_1^N$, the subspace dimension is $d_m(y_1^N)$

---

$d_m(y_1^N) = H\left(U_{\mathcal{A}^{(m)}} \middle| Y_1^N = y_1^N, U_{\mathcal{F}^{(m)}}\right)$ where $\mathcal{A}^{(m)} \triangleq \mathcal{A} \cap [m]$ and $\mathcal{F}^{(m)} \triangleq \mathcal{F} \cap [m]$

# The Uncertainty Dimension

- For a fixed $y_1^N$, the subspace dimension is $d_m(y_1^N)$
- Let $D_m = d_m(Y_1^N)$ denote corresponding random value at step $m$

$d_m(y_1^N) = H\left(U_{\mathcal{A}^{(m)}} \middle| Y_1^N = y_1^N, U_{\mathcal{F}^{(m)}}\right)$ where $\mathcal{A}^{(m)} \triangleq \mathcal{A} \cap [m]$ and $\mathcal{F}^{(m)} \triangleq \mathcal{F} \cap [m]$

# The Uncertainty Dimension

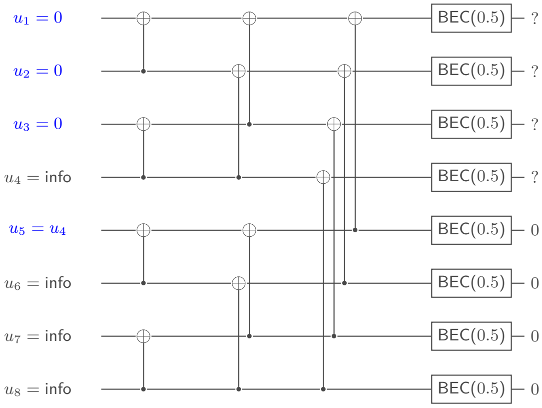
- For a fixed $y_1^N$, the subspace dimension is $d_m(y_1^N)$
- Let $D_m = d_m(Y_1^N)$ denote corresponding random value at step $m$

Next goal is to understand the evolution of the random sequence $D_m$

$d_m(y_1^N) = H\left(U_{\mathcal{A}^{(m)}} \middle| Y_1^N = y_1^N, U_{\mathcal{F}^{(m)}}\right)$ where $\mathcal{A}^{(m)} \triangleq \mathcal{A} \cap [m]$ and $\mathcal{F}^{(m)} \triangleq \mathcal{F} \cap [m]$

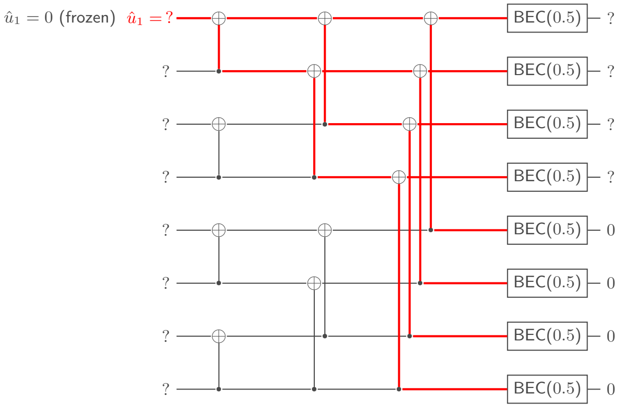
# Succesive Cancellation Inactivation Decoding

Example: $u_1 = u_2 = u_3 = 0$, $u_5 = u_4$ (frozen bits)



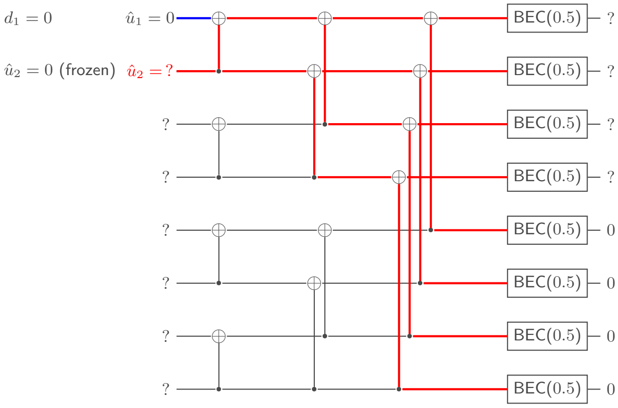
# Succesive Cancellation Inactivation Decoding

Example: $u_1 = u_2 = u_3 = 0$, $u_5 = u_4$ (frozen bits)



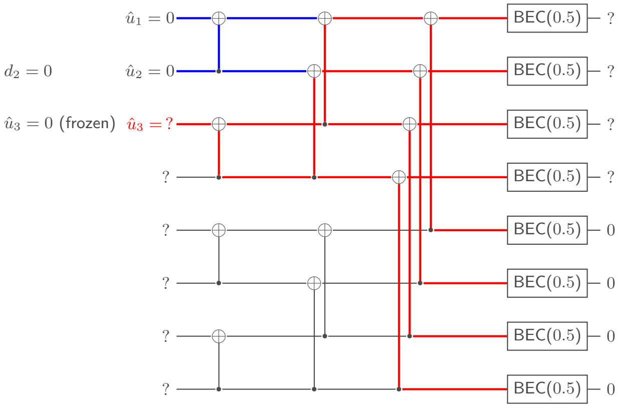
# Succesive Cancellation Inactivation Decoding

Example: $u_1 = u_2 = u_3 = 0$, $u_5 = u_4$ (frozen bits)



# Succesive Cancellation Inactivation Decoding

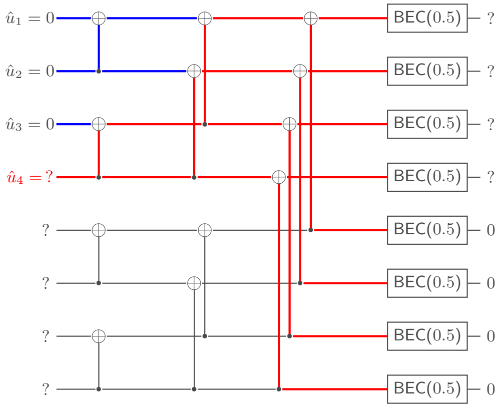
Example: $u_1 = u_2 = u_3 = 0$, $u_5 = u_4$ (frozen bits)



# Succesive Cancellation Inactivation Decoding

Example: $u_1 = u_2 = u_3 = 0$, $u_5 = u_4$ (frozen bits)

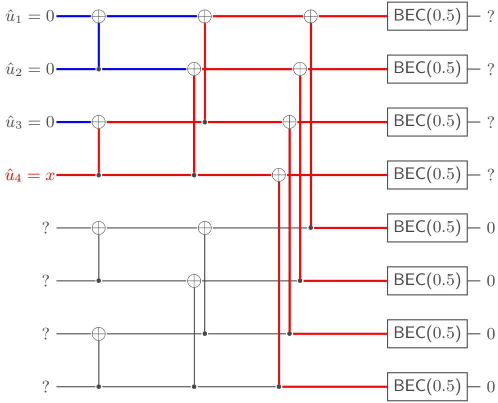
$d_3 = 0$



# Succesive Cancellation Inactivation Decoding

Example: $u_1 = u_2 = u_3 = 0$, $u_5 = u_4$ (frozen bits)

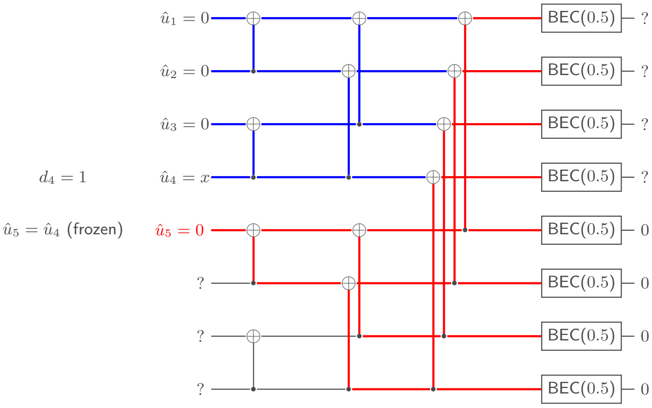
$d_3 = 0$




DLR

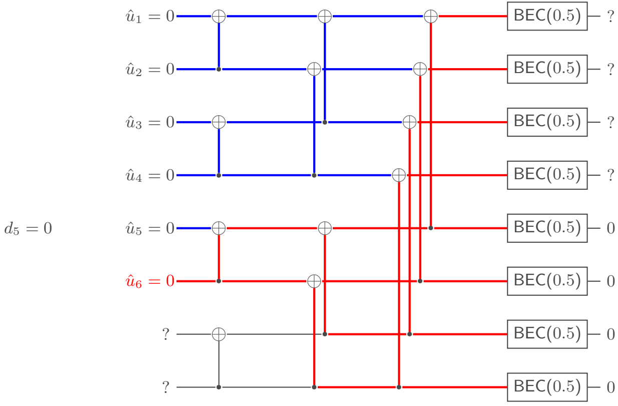
# Succesive Cancellation Inactivation Decoding

Example: $u_1 = u_2 = u_3 = 0$, $u_5 = u_4$ (frozen bits)



# Succesive Cancellation Inactivation Decoding

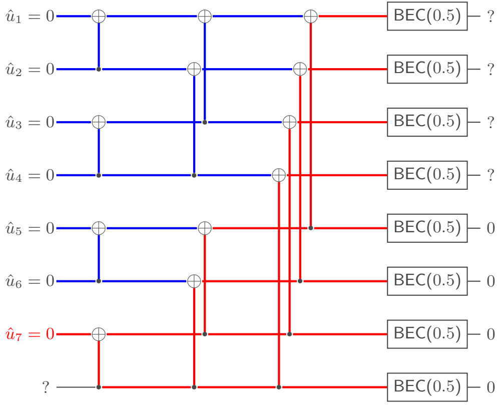
Example: $u_1 = u_2 = u_3 = 0$, $u_5 = u_4$ (frozen bits)



# Succesive Cancellation Inactivation Decoding

Example: $u_1 = u_2 = u_3 = 0$, $u_5 = u_4$ (frozen bits)

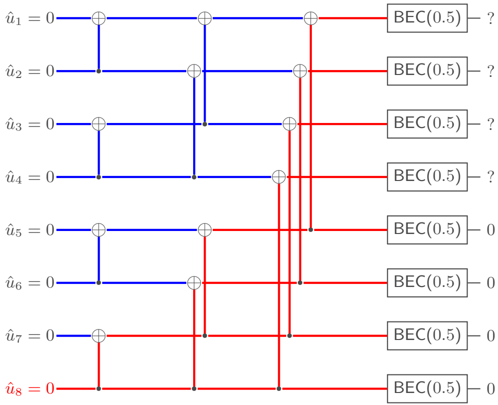
$d_6 = 0$



# Succesive Cancellation Inactivation Decoding

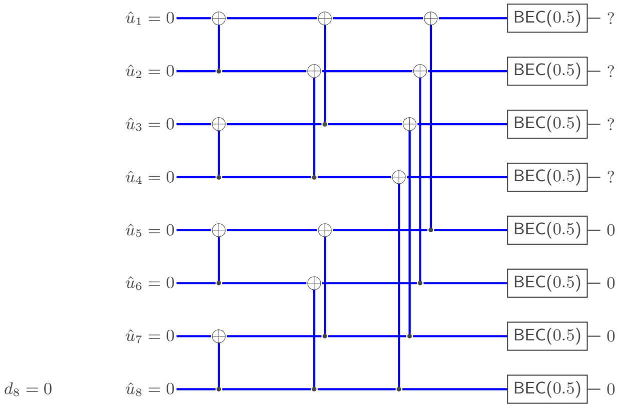
Example: $u_1 = u_2 = u_3 = 0$, $u_5 = u_4$ (frozen bits)

$d_7 = 0$



# Succesive Cancellation Inactivation Decoding

Example: $u_1 = u_2 = u_3 = 0$, $u_5 = u_4$ (frozen bits)



# Evolution of the Uncertainty Dimension

- If $U_m$ is an information bit, then
  - ✓ If decoder outputs an erasure, then $d_m(y_1^N) = d_{m-1}(y_1^N) + 1$
  - ✓ Else, it outputs affine function and $d_m(y_1^N) = d_{m-1}(y_1^N)$

# Evolution of the Uncertainty Dimension

- If $U_m$ is an information bit, then
  - ✓ If decoder outputs an erasure, then $d_m(y_1^N) = d_{m-1}(y_1^N) + 1$
  - ✓ Else, it outputs affine function and $d_m(y_1^N) = d_{m-1}(y_1^N)$

- If $U_m$ is a frozen bit, then
  - ✓ If decoder outputs an erasure, then $d_m(y_1^N) = d_{m-1}(y_1^N)$
  - ✓ Else, it outputs affine function:

# Evolution of the Uncertainty Dimension

- If $U_m$ is an information bit, then
  - ✓ If decoder outputs an erasure, then $d_m(y_1^N) = d_{m-1}(y_1^N) + 1$
  - ✓ Else, it outputs affine function and $d_m(y_1^N) = d_{m-1}(y_1^N)$

- If $U_m$ is a frozen bit, then
  - ✓ If decoder outputs an erasure, then $d_m(y_1^N) = d_{m-1}(y_1^N)$
  - ✓ Else, it outputs affine function:
    - × If consolidation: $d_m(y_1^N) = d_{m-1}(y_1^N) - 1$
    - × Else, no consolidation: $d_m(y_1^N) = d_{m-1}(y_1^N)$

# The Markov Chain Approximation

- The random sequence $D_1, \ldots, D_N$ can be approximated by an inhomogeneous Markov chain with transition probabilities $P_{i,j}^{(m)} \approx \Pr\left(D_m = j \mid D_{m-1} = i\right)$ where

$$P_{i,j}^{(m)} = \begin{cases} \epsilon_N^{(m)} & \text{if } m \in \mathcal{A},\ j = i+1 \\ 1-\epsilon_N^{(m)} & \text{if } m \in \mathcal{A},\ j = i \\ \epsilon_N^{(m)} + \left(1-\epsilon_N^{(m)}\right) 2^{-D_{m-1}} & \text{if } m \in \mathcal{F},\ j = i \\ \left(1-\epsilon_N^{(m)}\right)\left(1-2^{-D_{m-1}}\right) & \text{if } m \in \mathcal{F},\ j = i-1 \end{cases}$$

# The Markov Chain Approximation

- The random sequence $D_1, \ldots, D_N$ can be approximated by an inhomogeneous Markov chain with transition probabilities $P_{i,j}^{(m)} \approx \Pr\left(D_m = j \mid D_{m-1} = i\right)$ where

$$P_{i,j}^{(m)} = \begin{cases} \epsilon_N^{(m)} & \text{if } m \in \mathcal{A},\ j = i+1 \\ 1 - \epsilon_N^{(m)} & \text{if } m \in \mathcal{A},\ j = i \\ \epsilon_N^{(m)} + \left(1 - \epsilon_N^{(m)}\right) 2^{-D_{m-1}} & \text{if } m \in \mathcal{F},\ j = i \\ \left(1 - \epsilon_N^{(m)}\right) \left(1 - 2^{-D_{m-1}}\right) & \text{if } m \in \mathcal{F},\ j = i-1 \end{cases}$$

- $\epsilon_N^{(m)}$ is the DE erasure probability of $m$-th effective channel

# The Markov Chain Approximation

- The random sequence $D_1, \dots, D_N$ can be approximated by an inhomogeneous Markov chain with transition probabilities $P_{i,j}^{(m)} \approx \Pr\left(D_m = j \,|\, D_{m-1} = i\right)$ where

$$P_{i,j}^{(m)} = \begin{cases} \epsilon_N^{(m)} & \text{if } m \in \mathcal{A},\ j = i+1 \\ 1 - \epsilon_N^{(m)} & \text{if } m \in \mathcal{A},\ j = i \\ \epsilon_N^{(m)} + \left(1 - \epsilon_N^{(m)}\right) 2^{-D_{m-1}} & \text{if } m \in \mathcal{F},\ j = i \\ \left(1 - \epsilon_N^{(m)}\right)\left(1 - 2^{-D_{m-1}}\right) & \text{if } m \in \mathcal{F},\ j = i-1 \end{cases}$$

- $\epsilon_N^{(m)}$ is the DE erasure probability of $m$-th effective channel
- $2^{-D}$ is probability a random $D$-variable equation has all zero coefficients

# d-RM and PAC Codes

- d-RM code ensemble[3]:
  - 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$(0.5)$

[3] M. C. Coşkun, J. Neu, H. D. Pfister, "Successive Cancellation Inactivation Decoding for Modified Reed-Muller and eBCH Codes," ISIT, Jun. 2020.

[5] E. Arıkan, "From sequential decoding to Channel Polarization and Back Again," Shannon Lecture, Jun. 2019.

# d-RM and PAC Codes

- d-RM code ensemble[3]:
  - 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$(0.5)$

- PAC code[5]:
  - Given set $\mathcal{A}$ and convolutional code with rate $1$ and memory $\nu$
  - $u_i$ is an information bit if $i \in \mathcal{A}$
  - $u_i = g_i(u_{i-\nu}^{i-1})$ if $i \in \mathcal{F}$ where $g_i(\cdot)$ defined by CC

[3] M. C. Coşkun, J. Neu, H. D. Pfister, "Successive Cancellation Inactivation Decoding for Modified Reed-Muller and eBCH Codes," ISIT, Jun. 2020.

[5] E. Arıkan, "From sequential decoding to Channel Polarization and Back Again," Shannon Lecture, Jun. 2019.

# d-RM and PAC Codes

- d-RM code ensemble[3]:
  - 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$(0.5)$

- PAC code[5]:
  - Given set $\mathcal{A}$ and convolutional code with rate $1$ and memory $\nu$
  - $u_i$ is an information bit if $i \in \mathcal{A}$
  - $u_i = g_i(u_{i-\nu}^{i-1})$ if $i \in \mathcal{F}$ where $g_i(\cdot)$ defined by CC

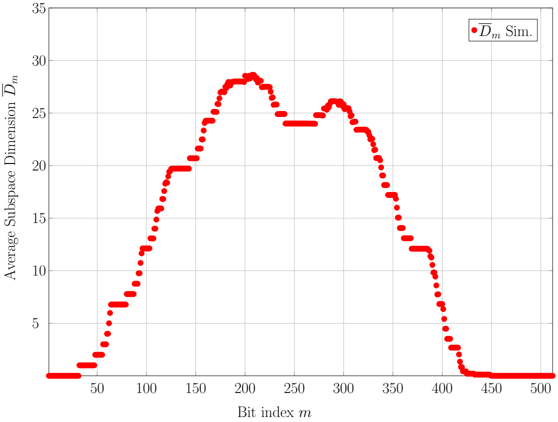
Choosing $\mathcal{A}$ like an RM code for the PAC code (as Arıkan did) makes two codes differ only in the dynamic frozen bit constraints!

[3] M. C. Coşkun, J. Neu, H. D. Pfister, "Successive Cancellation Inactivation Decoding for Modified Reed-Muller and eBCH Codes," ISIT, Jun. 2020.

[5] E. Arıkan, "From sequential decoding to Channel Polarization and Back Again," Shannon Lecture, Jun. 2019.

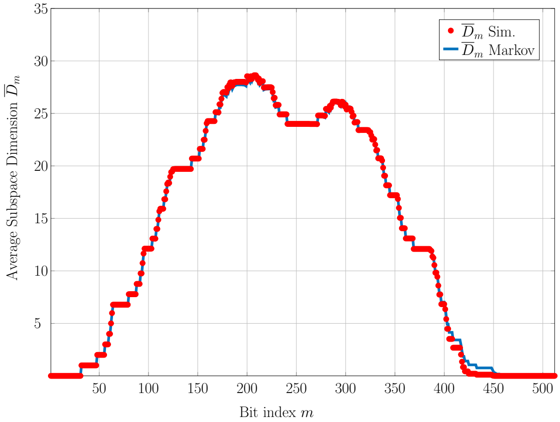
# $(512, 256)$ d-RM Code

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



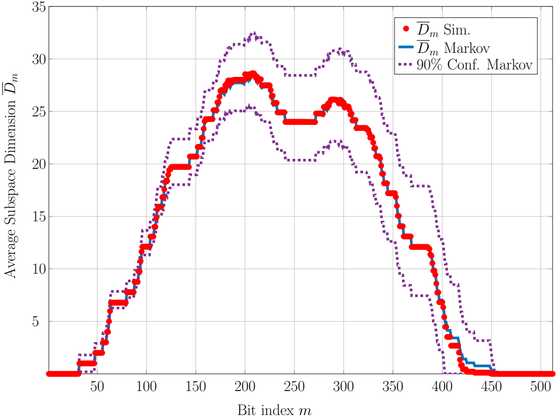
# $(512, 256)$ d-RM Code

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



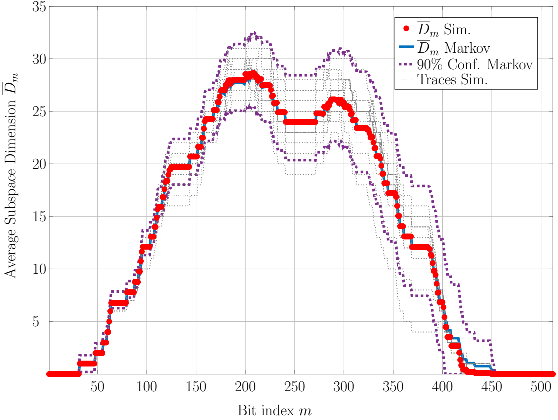
# $(512, 256)$ d-RM Code

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



# $(512, 256)$ d-RM Code

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



# Summary

- "What list size is sufficient to approach ML decoding performance under an SCL decoder?"
  - ✓ Good approximation proposed for the BEC that can be computed efficiently

[6] M. C. Coșkun, H. D. Pfister, "Bounds on the list size of successive cancellation list decoding," SPCOM, Jul. 2020.

# Summary

- "What list size is sufficient to approach ML decoding performance under an SCL decoder?"
  - ✓ Good approximation proposed for the BEC that can be computed efficiently
- What is not covered in this talk?[6]
  - ✓ The quantity $d_m$ (a conditional entropy) can be used as proxy for uncertainty in SCL decoding for general BMS channels

[6] M. C. Coşkun, H. D. Pfister, "Bounds on the list size of successive cancellation list decoding," SPCOM, Jul. 2020.

DLR

# Summary

- "What list size is sufficient to approach ML decoding performance under an SCL decoder?"
  - ✓ Good approximation proposed for the BEC that can be computed efficiently
- What is not covered in this talk?[6]
  - ✓ The quantity $d_m$ (a conditional entropy) can be used as proxy for uncertainty in SCL decoding for general BMS channels
  - ✓ The analysis leads to an improved code design (in comparison with the PAC code) under SCL decoding with list size $32$

[6] M. C. Coşkun, H. D. Pfister, "Bounds on the list size of successive cancellation list decoding," SPCOM, Jul. 2020.

# Summary

- "What list size is sufficient to approach ML decoding performance under an SCL decoder?"
  - ✓ Good approximation proposed for the BEC that can be computed efficiently
- What is not covered in this talk?[6]
  - ✓ The quantity $d_m$ (a conditional entropy) can be used as proxy for uncertainty in SCL decoding for general BMS channels
  - ✓ The analysis leads to an improved code design (in comparison with the PAC code) under SCL decoding with list size $32$
- Outlook and Future Work
  - ✓ Show concentration of $D_m(Y_1^N)$ so that its average is meaningful

[6] M. C. Coşkun, H. D. Pfister, "Bounds on the list size of successive cancellation list decoding," SPCOM, Jul. 2020.

# Summary

- "What list size is sufficient to approach ML decoding performance under an SCL decoder?"
  - ✓ Good approximation proposed for the BEC that can be computed efficiently
- What is not covered in this talk?[6]
  - ✓ The quantity $d_m$ (a conditional entropy) can be used as proxy for uncertainty in SCL decoding for general BMS channels
  - ✓ The analysis leads to an improved code design (in comparison with the PAC code) under SCL decoding with list size $32$
- Outlook and Future Work
  - ✓ Show concentration of $D_m(Y_1^N)$ so that its average is meaningful
  - ⚠ Apply this technique to design longer codes with good SCL performance

[6] M. C. Coşkun, H. D. Pfister, "Bounds on the list size of successive cancellation list decoding," SPCOM, Jul. 2020.

Thank you! Questions?

# $(512, 256)$ Codes - Performance