KO codes

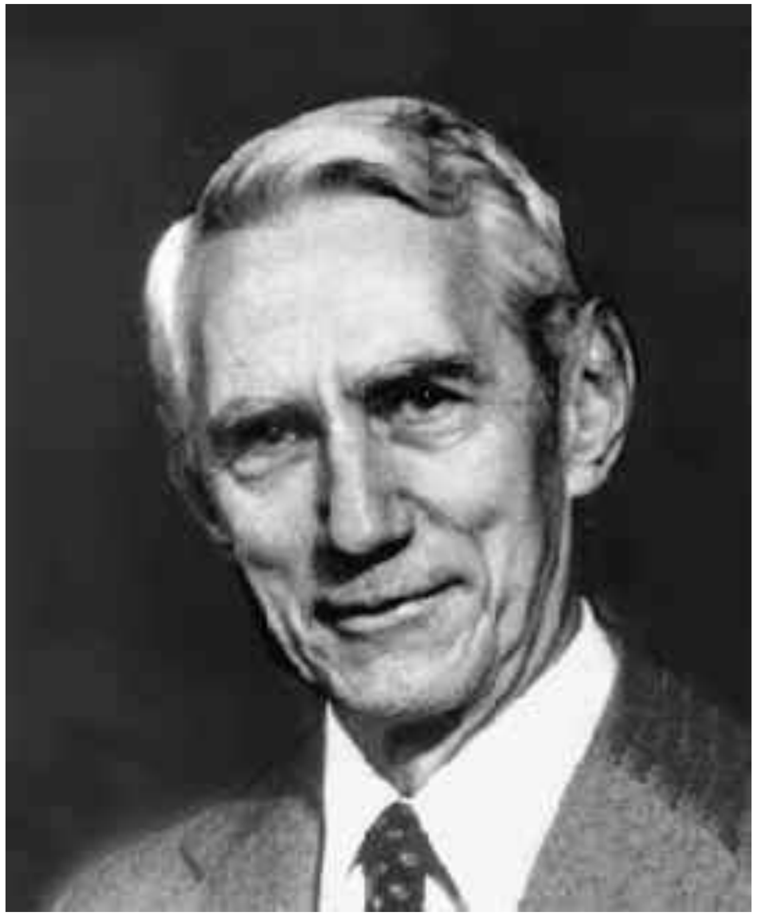
Ashok Vardhan Makkuva (UIUC)

Outline

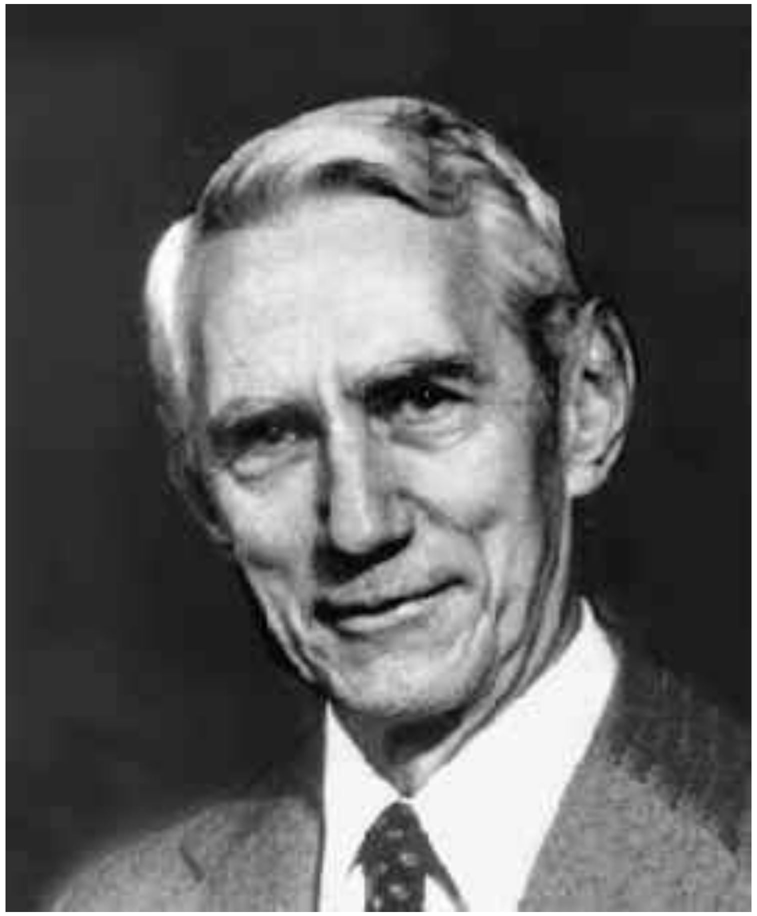
- Motivation
- Learning codes
- KO codes
- Future directions

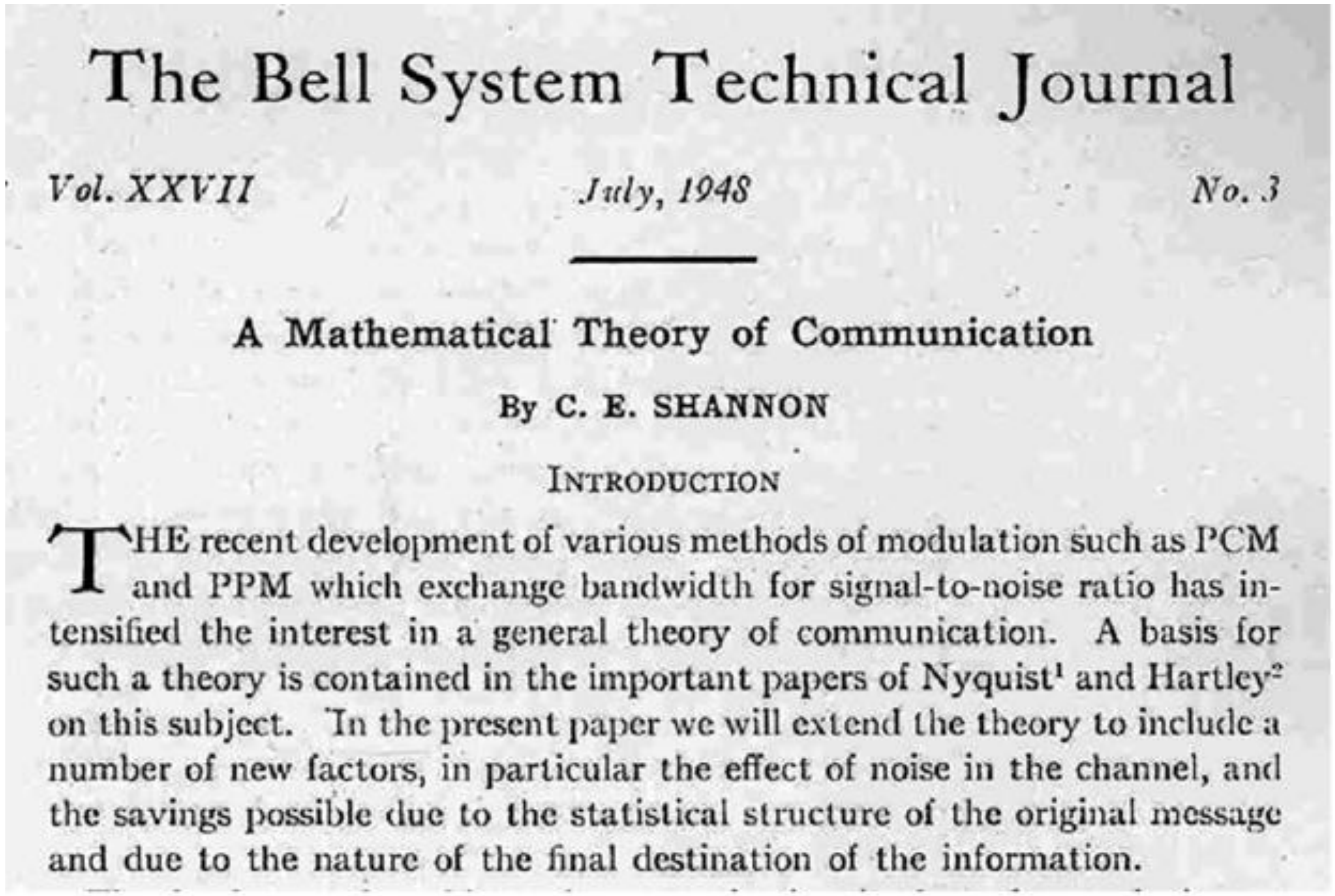
Age of Information

How did it start?



How did it start?





The Bell System Technical Journal

Vol. XXVII *July, 1948* *No. 3*

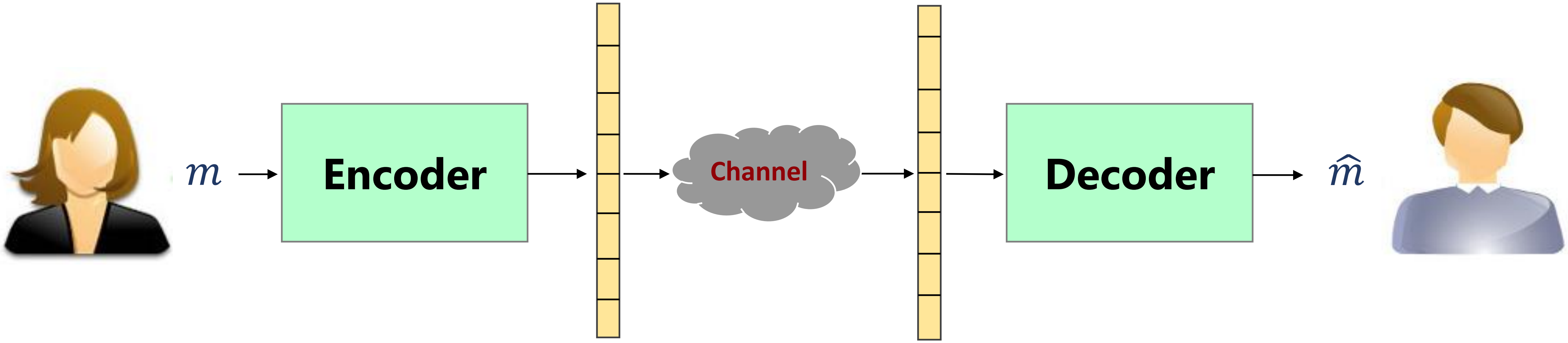
A Mathematical Theory of Communication

By C. E. SHANNON

INTRODUCTION

THE recent development of various methods of modulation such as PCM and PPM which exchange bandwidth for signal-to-noise ratio has intensified the interest in a general theory of communication. A basis for such a theory is contained in the important papers of Nyquist[1] and Hartley[2] on this subject. In the present paper we will extend the theory to include a number of new factors, in particular the effect of noise in the channel, and the savings possible due to the statistical structure of the original message and due to the nature of the final destination of the information.

Mathematical model of communication



Shannon, 1948

Communication codes

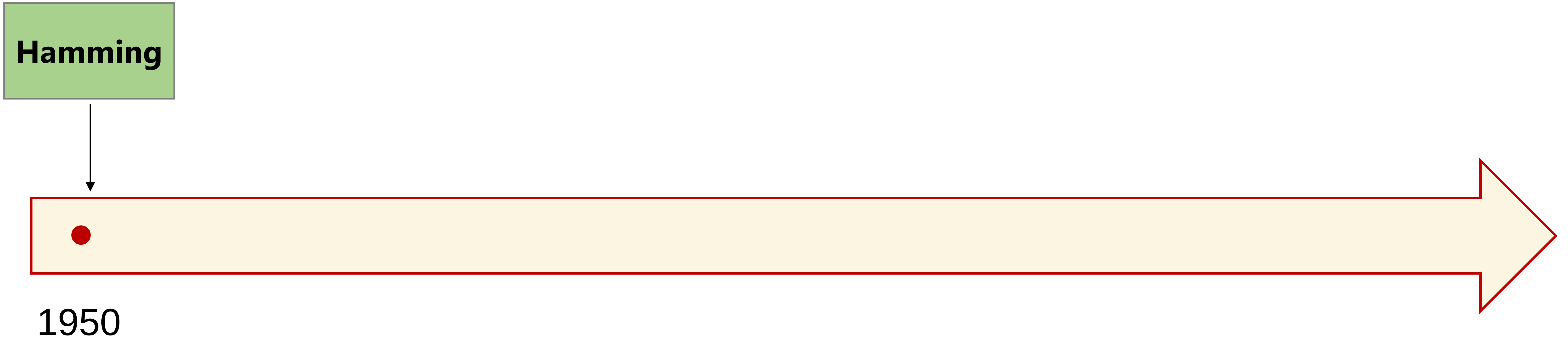
- Simple models: **AWGN channel**
  - Precise performance metrics

- Challenge: Space of (encoders, decoders) very large

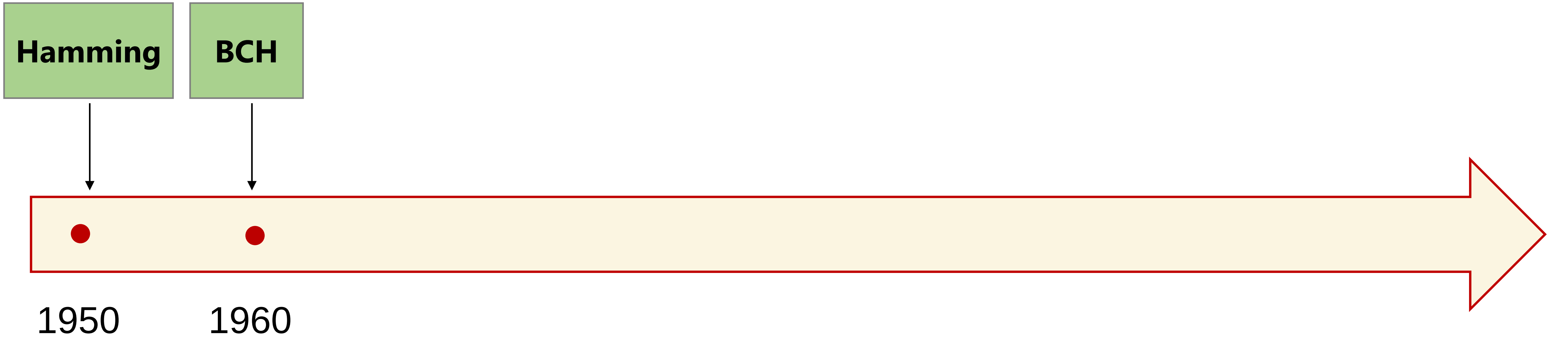
- Information theory, Communication theory, Coding theory

Landmark codes: AWGN

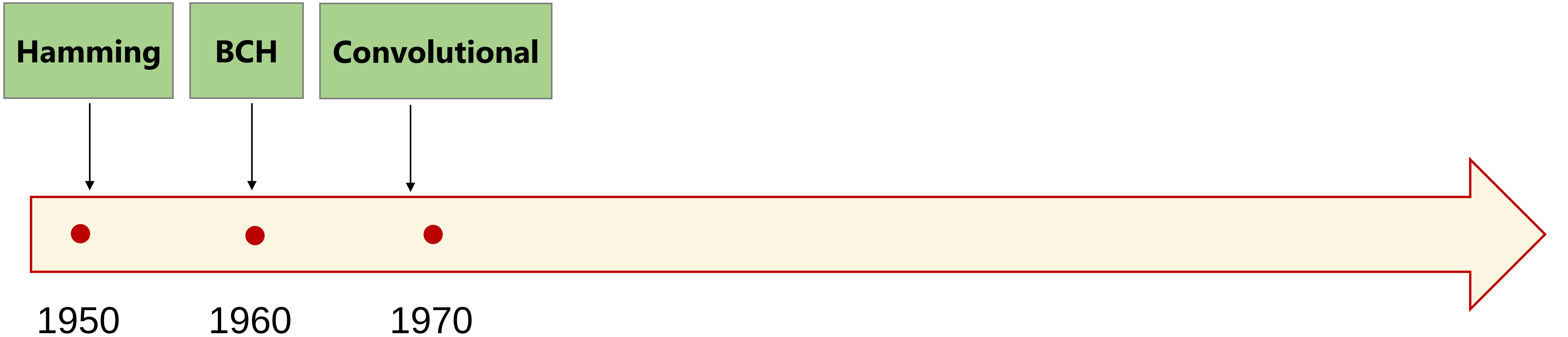
Landmark codes: AWGN



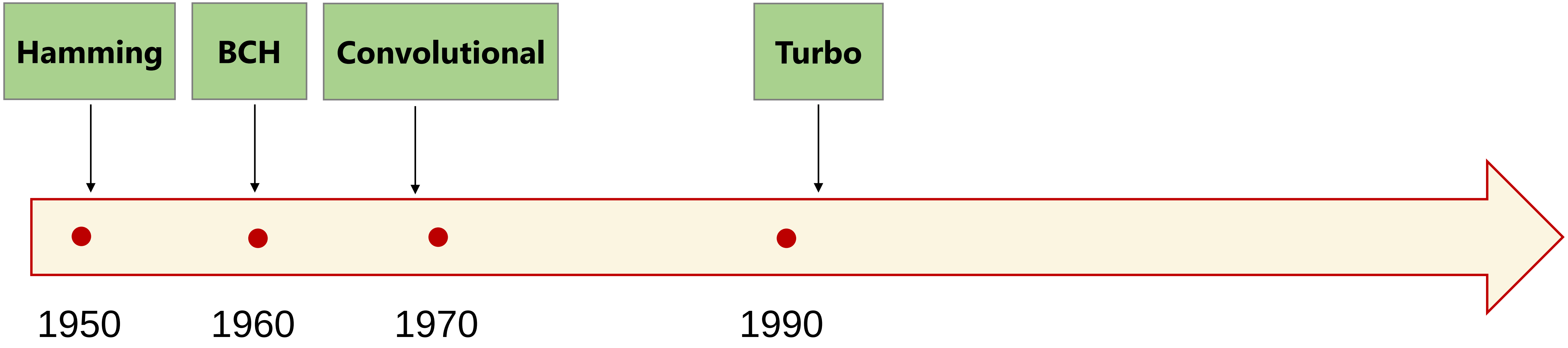
Landmark codes: AWGN



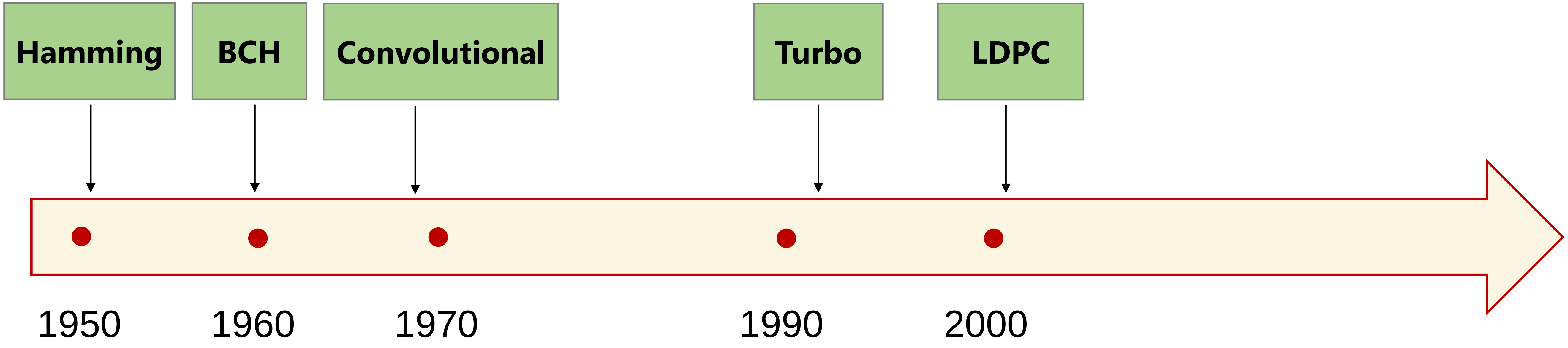
Landmark codes: AWGN



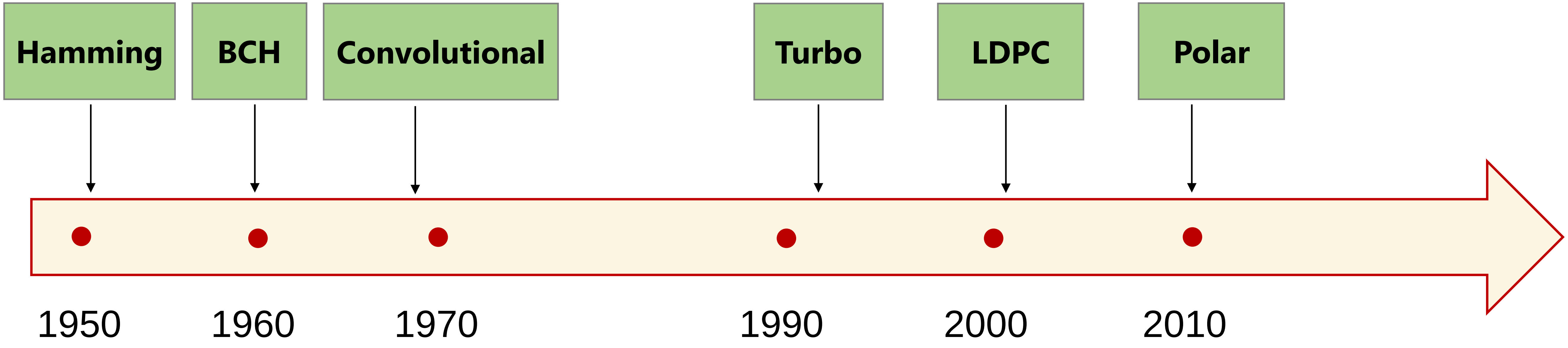
Landmark codes: AWGN



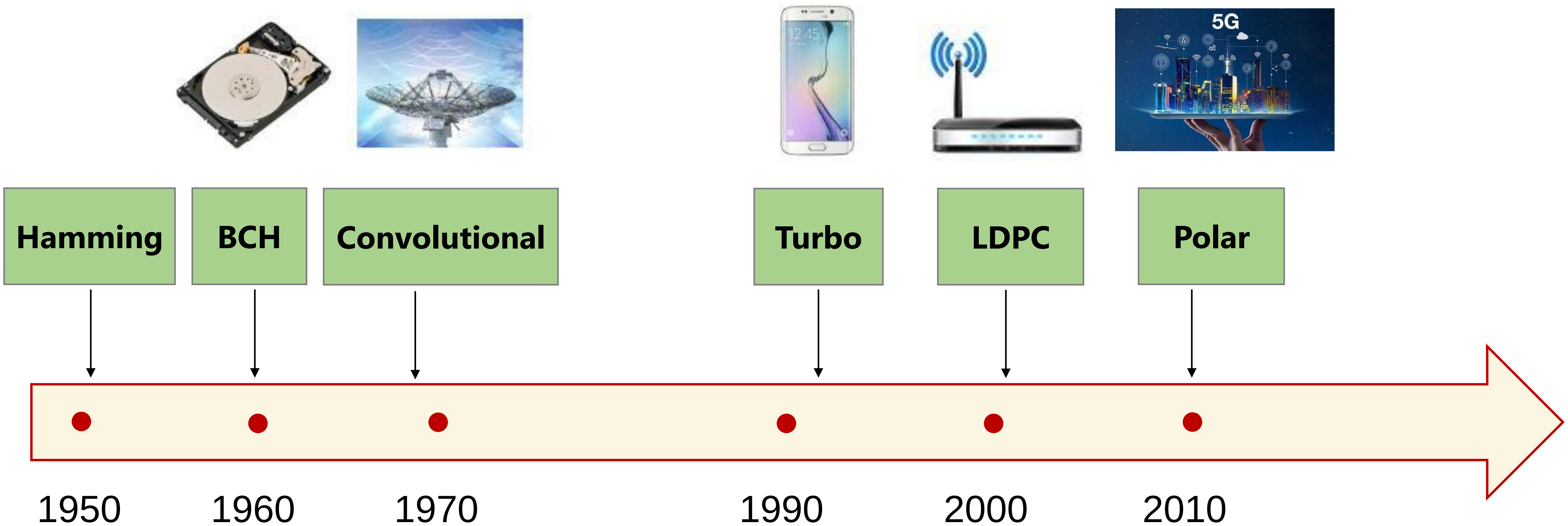
Landmark codes: AWGN



Landmark codes: AWGN



Huge practical impact

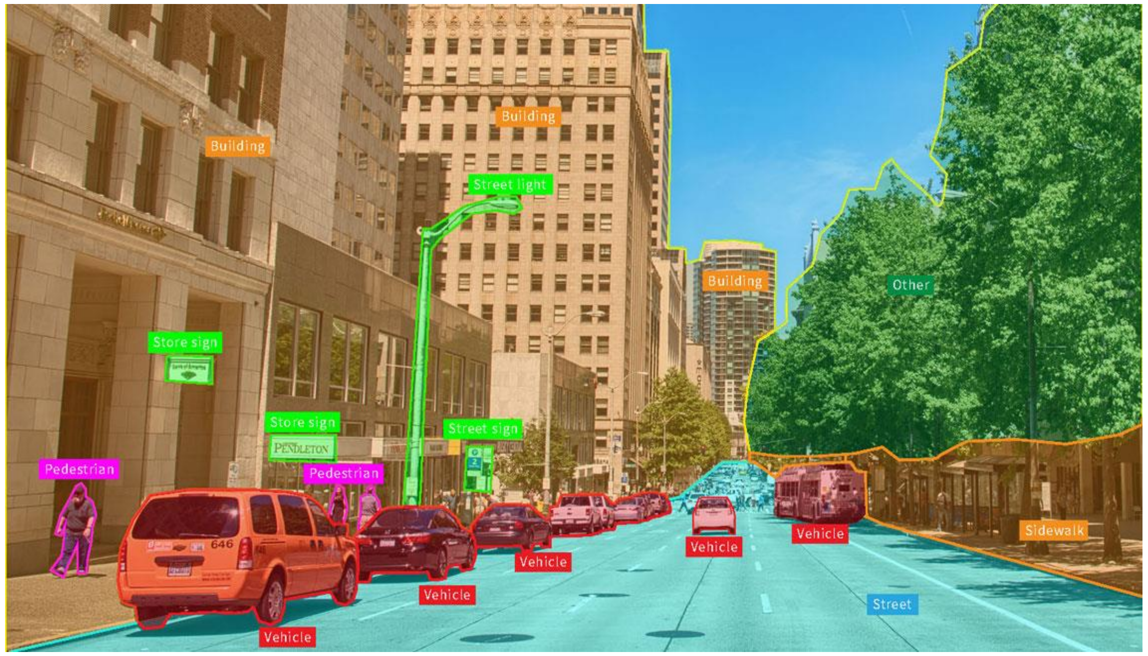


Vision

- Discovery of codes
  - ➢ Human eureka moments

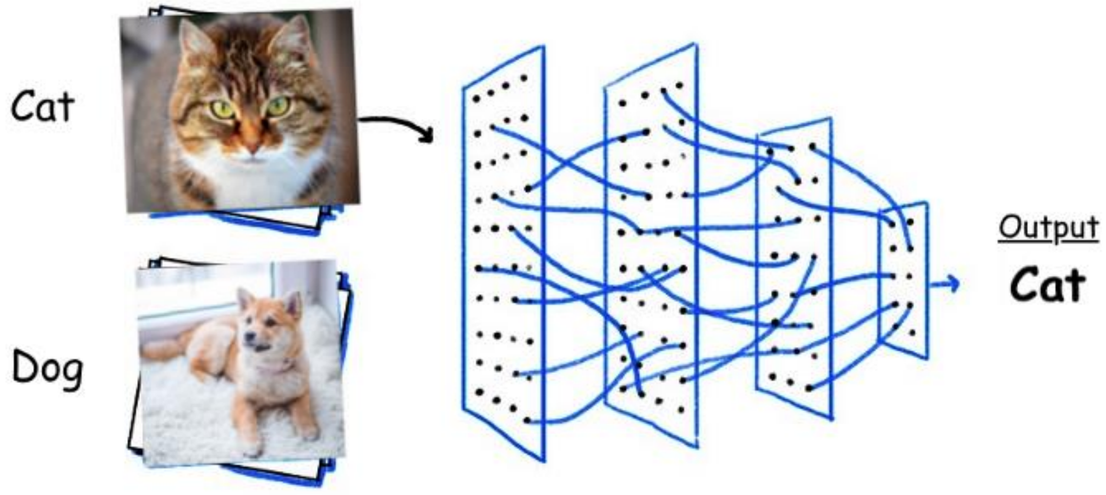
- **Goal:** Automate the discovery

Deep learning (DL)

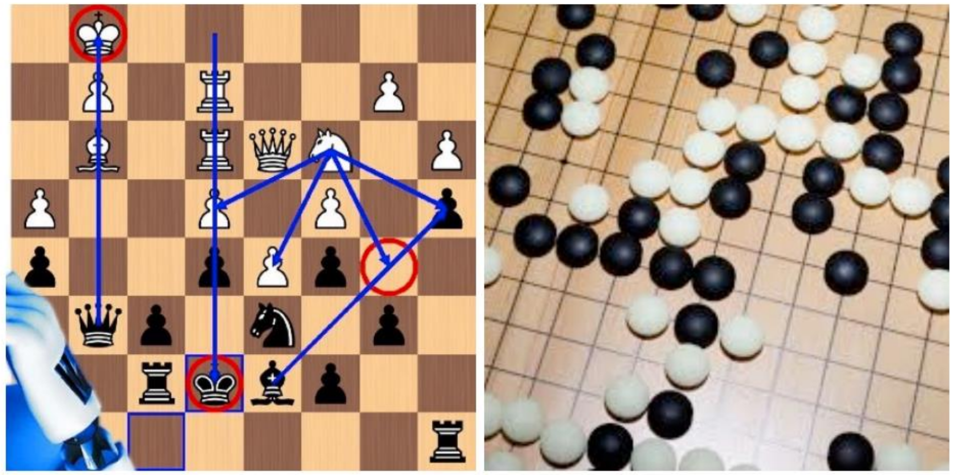


DL success story

- Model deficiency
  - ➢ No analytical model
  - ➢ **AlexNet**



- Algorithm deficiency
  - ➢ Clear model
  - ➢ Space of algorithms large
  - ➢ **AlphaGo (Zero)**



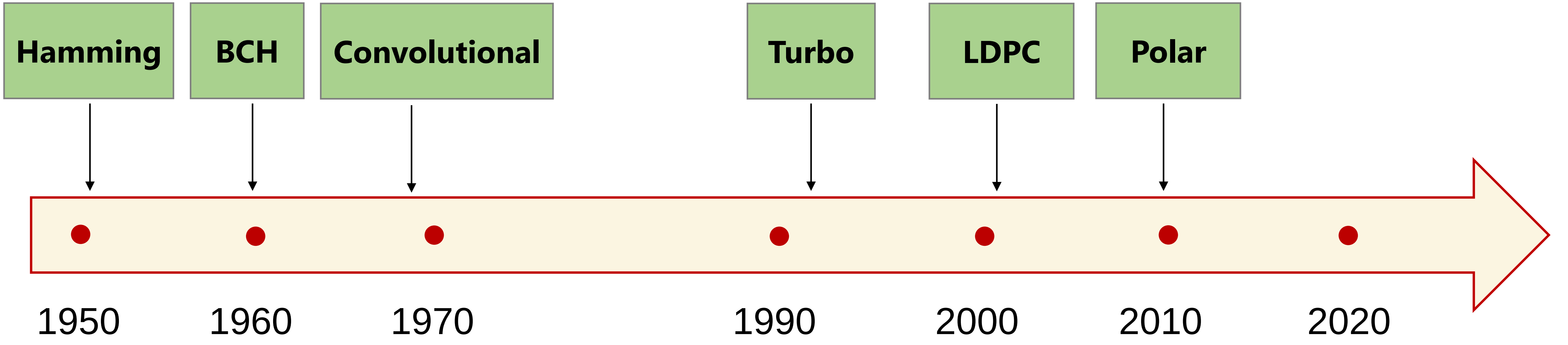
Breakthroughs of DL

AlphaGo

Main goal

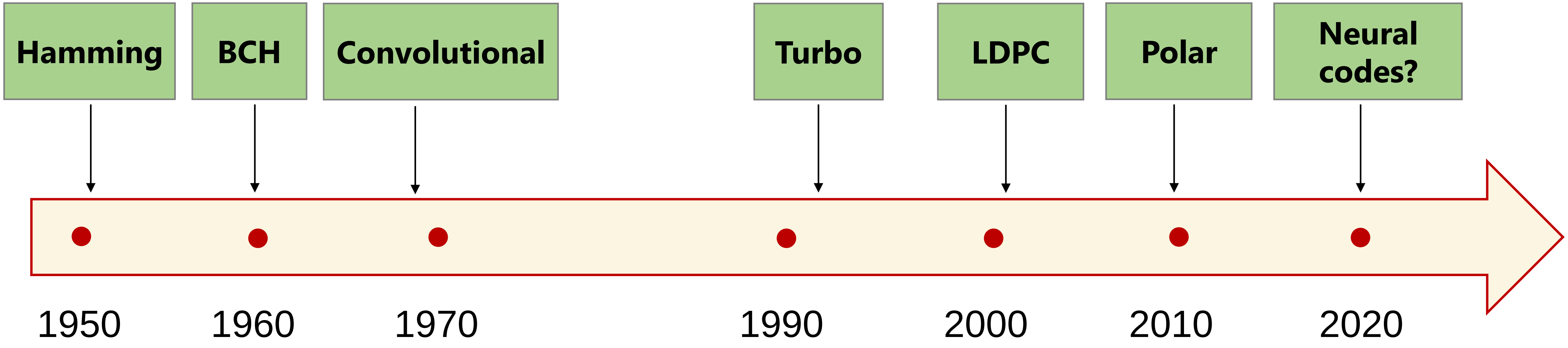
Can we automate the search for codes via DL?

Main goal



Hamming
BCH
Convolutional
Turbo
LDPC
Polar
1950
1960
1970
1990
2000
2010
2020

Main goal



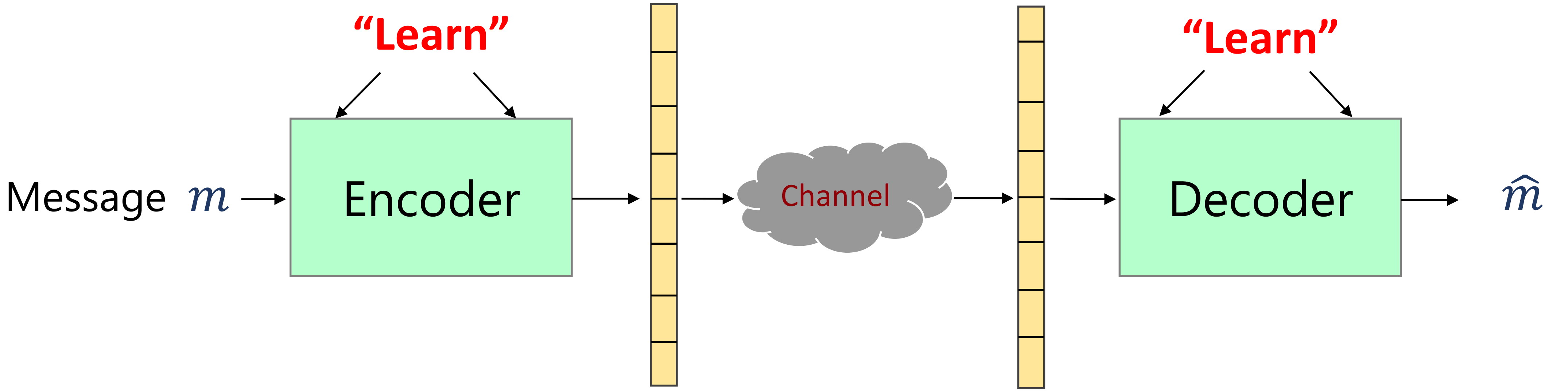
Hamming
BCH
Convolutional
Turbo
LDPC
Polar
Neural codes?
1950
1960
1970
1990
2000
2010
2020

Agenda

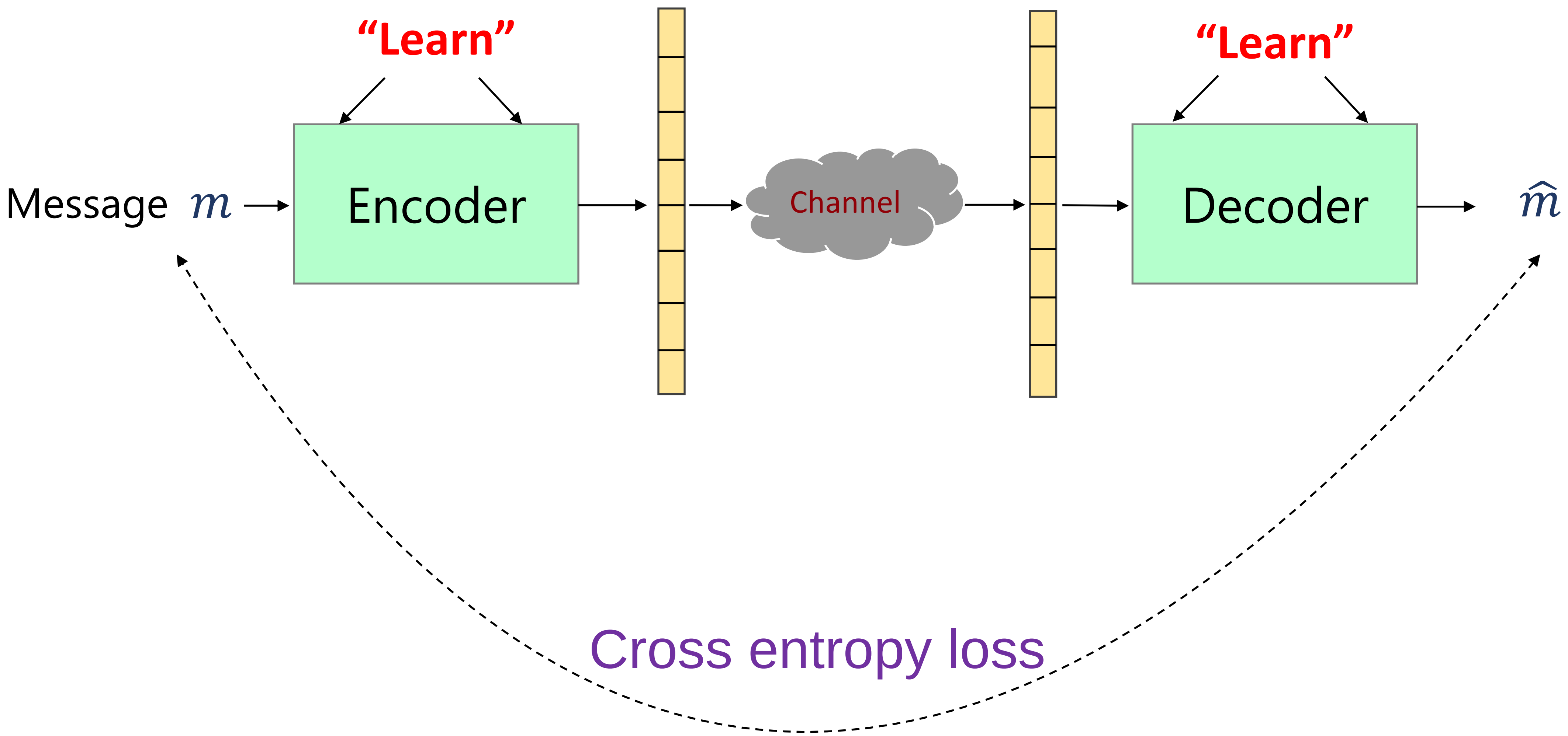
- New (deep learning) tools for classical problems
  - New state-of-the-art codes
  - Inherent practical value

- Insight into deep learning methods
  - Communication framework as a lens

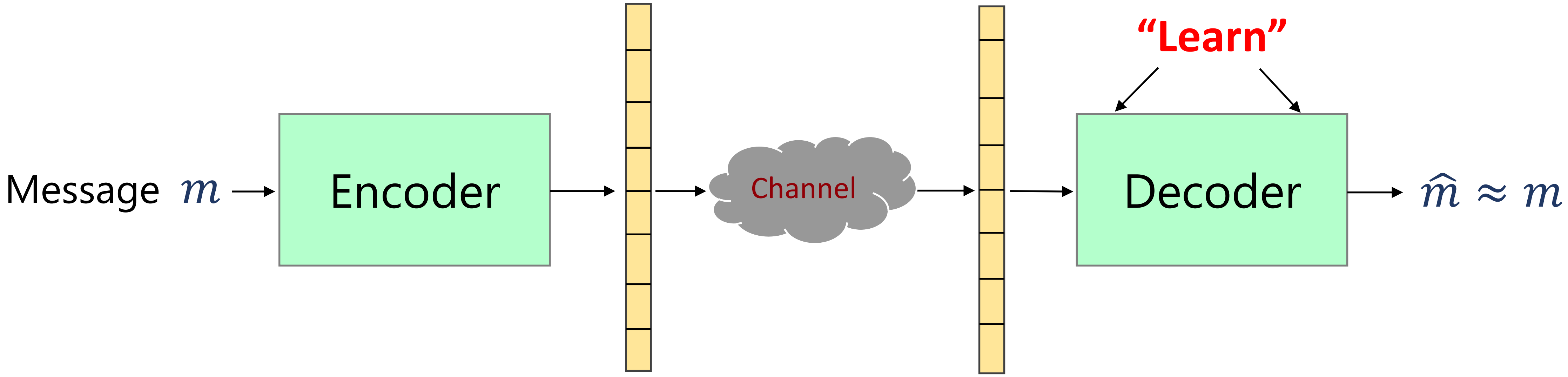
Learning a new code



Learning a new code



Learning to decode



Vast literature

- Supervised learning
  - Nachmani et al., 2016
  - Gruber et al. 2017
  - Cammerer et al., 2017
  - Nachmani et al., 2018
  - Kim et al., 2018a;b
  - Vasic et al., 2018
  - Teng et al., 2019
  - Nachmani & Wolf, 2019
  - Buchberger et al., 2020
  - Chen & Ye, 2021

- Reinforcement learning
  - Carpi et al., 2019
  - Habib et al., 2020
  - Doan et al., 2020

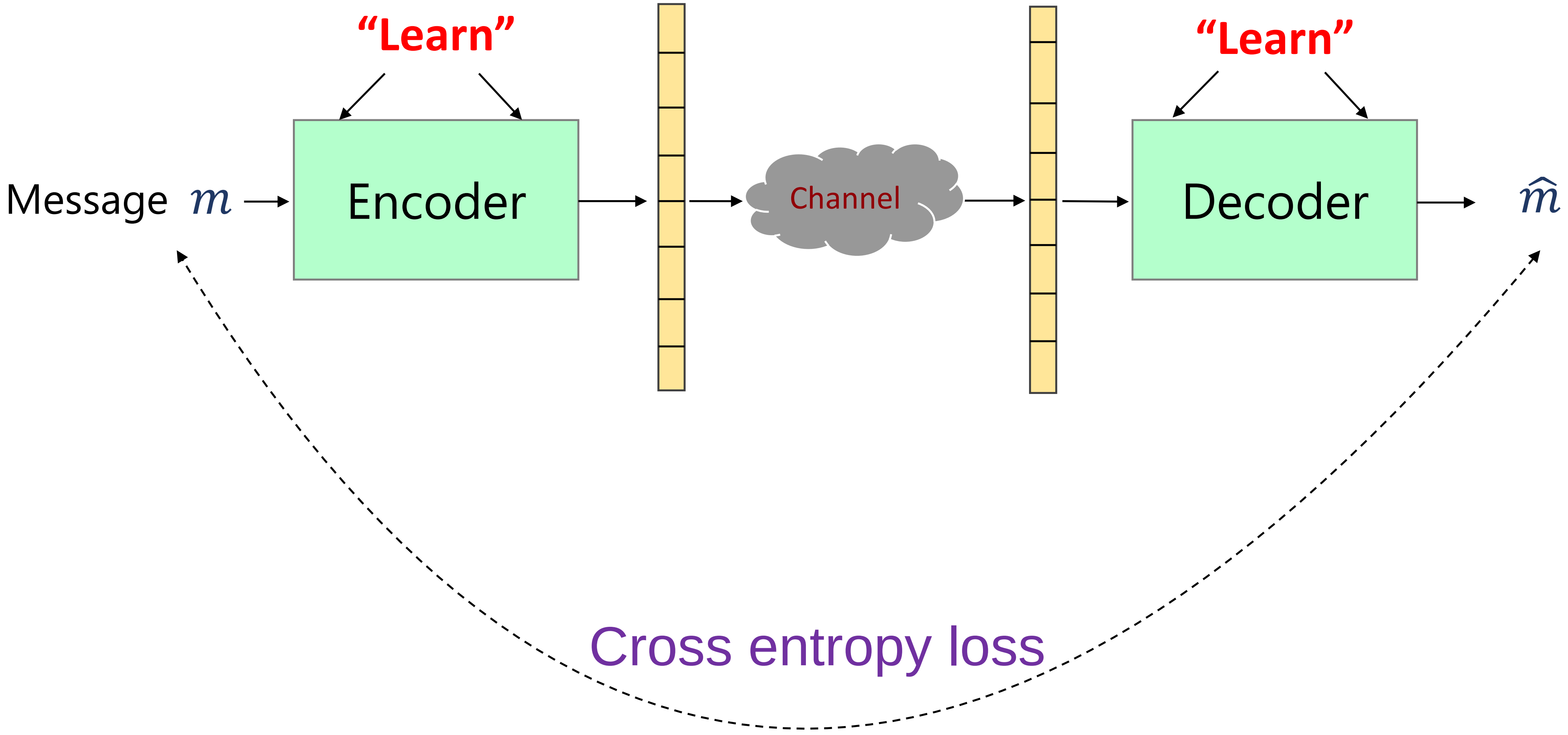
Learning to decode: summary

- Fix the encoding

- DL decoders learn state-of-the-art decoders
  - Convolutional codes: Viterbi, BCJR, dynamic programming
  - Turbo codes: BCJR
  - RM & Polar codes: Successive Cancellation

- Clever architectural choices
  - Recurrent neural networks $\longleftrightarrow$ dynamic programming

Learning a new code



Code structure

- Linear and binary: Classical codes

- Non-linear and real valued: **Neural networks (NNs)**
  - Fully connected NNs worse than repetition codes (Jiang et. al ‘19)
  - Still need a structure

Imparting structure

- Capitalize on state-of-the-art codes

- **What class of codes?**

Taxonomy of codes

Sequential codes
Eg. Convolutional and Turbo codes.

Graphical codes
Eg. LDPC codes.

Algebraic codes
Eg. Reed-Solomon, BCH, Reed-Muller and Polar codes.

Taxonomy of codes

Sequential codes
Eg. Convolutional and Turbo codes.

Graphical codes
Eg. LDPC codes.

Algebraic codes
Eg. Reed-Solomon, BCH, Reed-Muller and Polar codes.

Jiang et al, 2019

Algebraic codes

Sequential codes
Eg. Convolutional and Turbo codes.

Graphical codes
Eg. LDPC codes.

Algebraic codes
Eg. Reed-Solomon, BCH, Reed-Muller and Polar codes.

Reed-Muller and Polar codes

Sequential codes
Eg. Convolutional and Turbo codes.

Graphical codes
Eg. LDPC codes.

Algebraic codes
Eg. Reed-Solomon, BCH, Reed-Muller and Polar codes.

Reed-Muller codes (RM)

- **Classical**
  - Muller, 1954
  - Efficient decoder by Reed, 1954
- **Recent Interest**
  - Polar codes
  - RM codes are capacity achieving (very recent!)

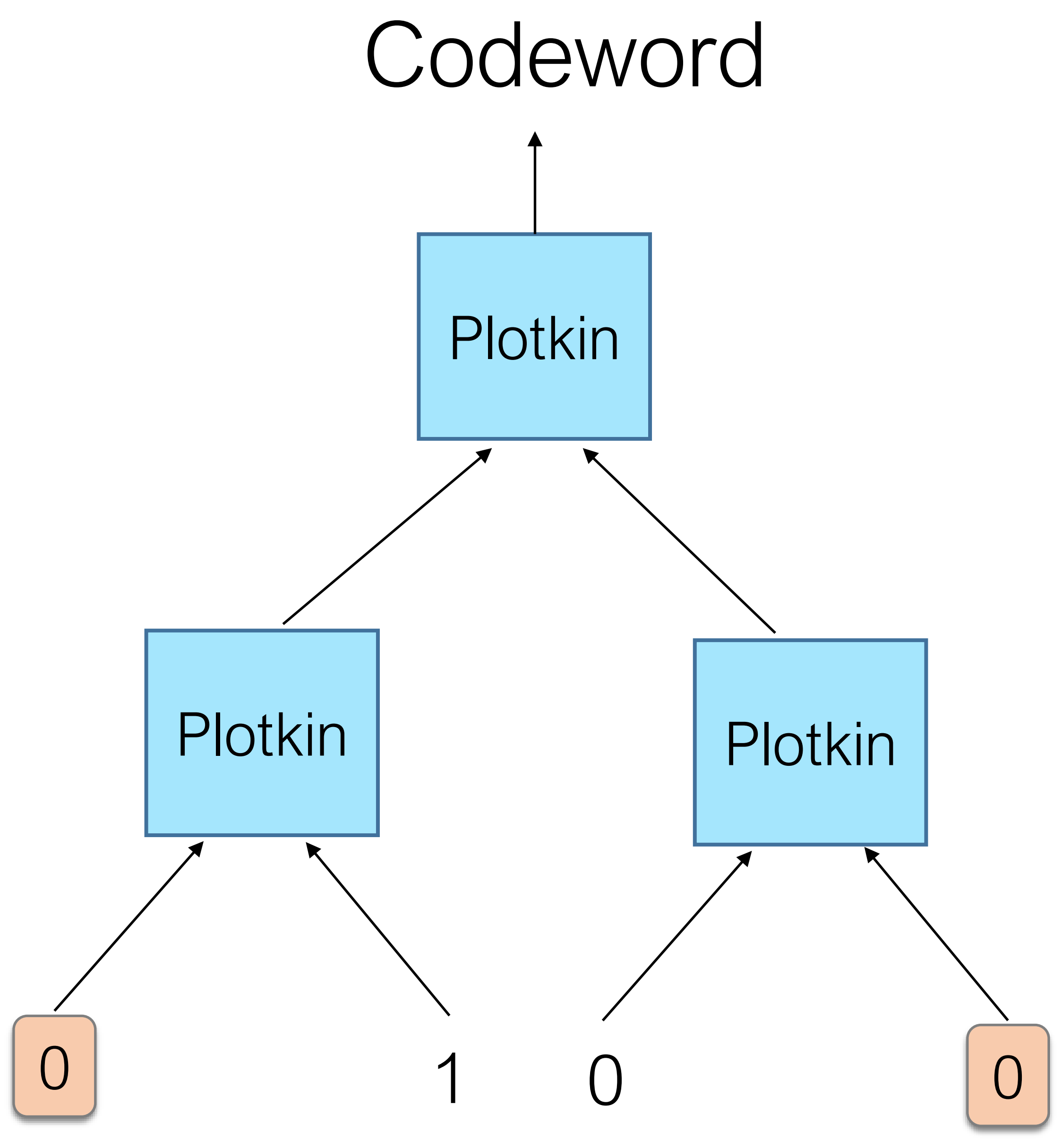
Polar codes

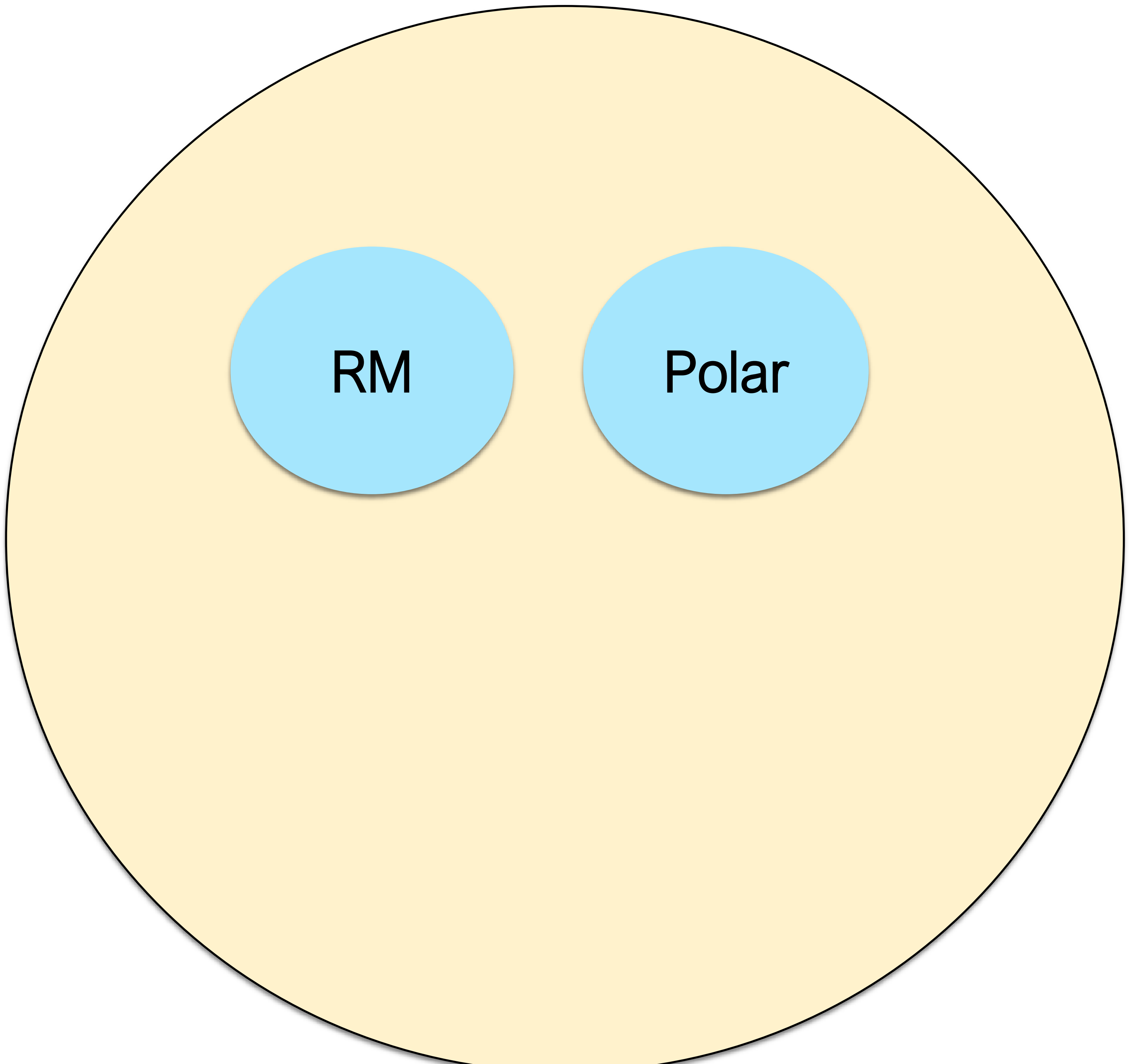
- Arikan, 2009
- **First codes proven to achieve capacity**
- Recent interest: 5G

RM and Polar

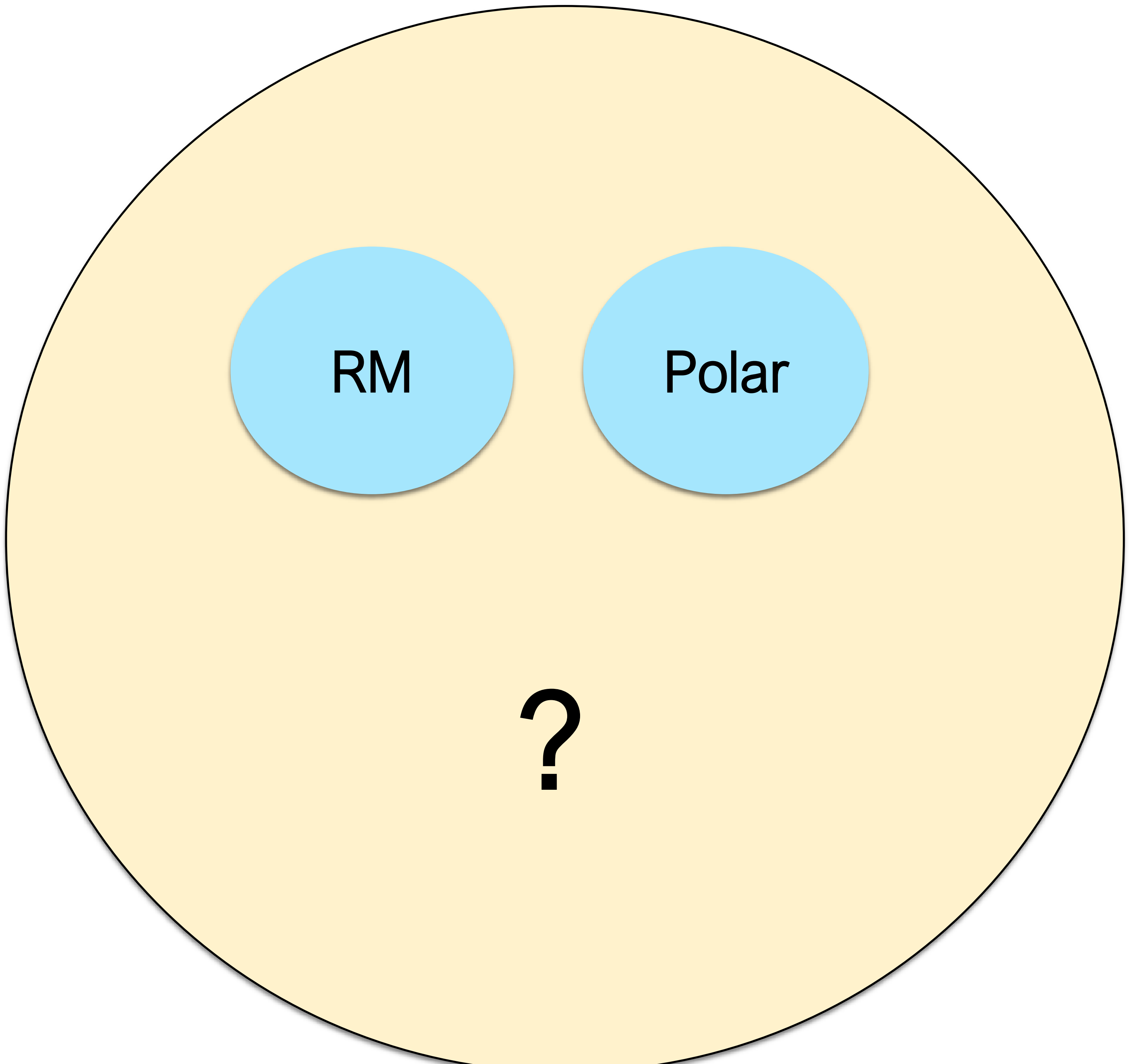
- Common structure
- **Kronecker Operation on the Plotkin transform**
- Freezing the leaves

Structure: Kronecker Operation (KO)

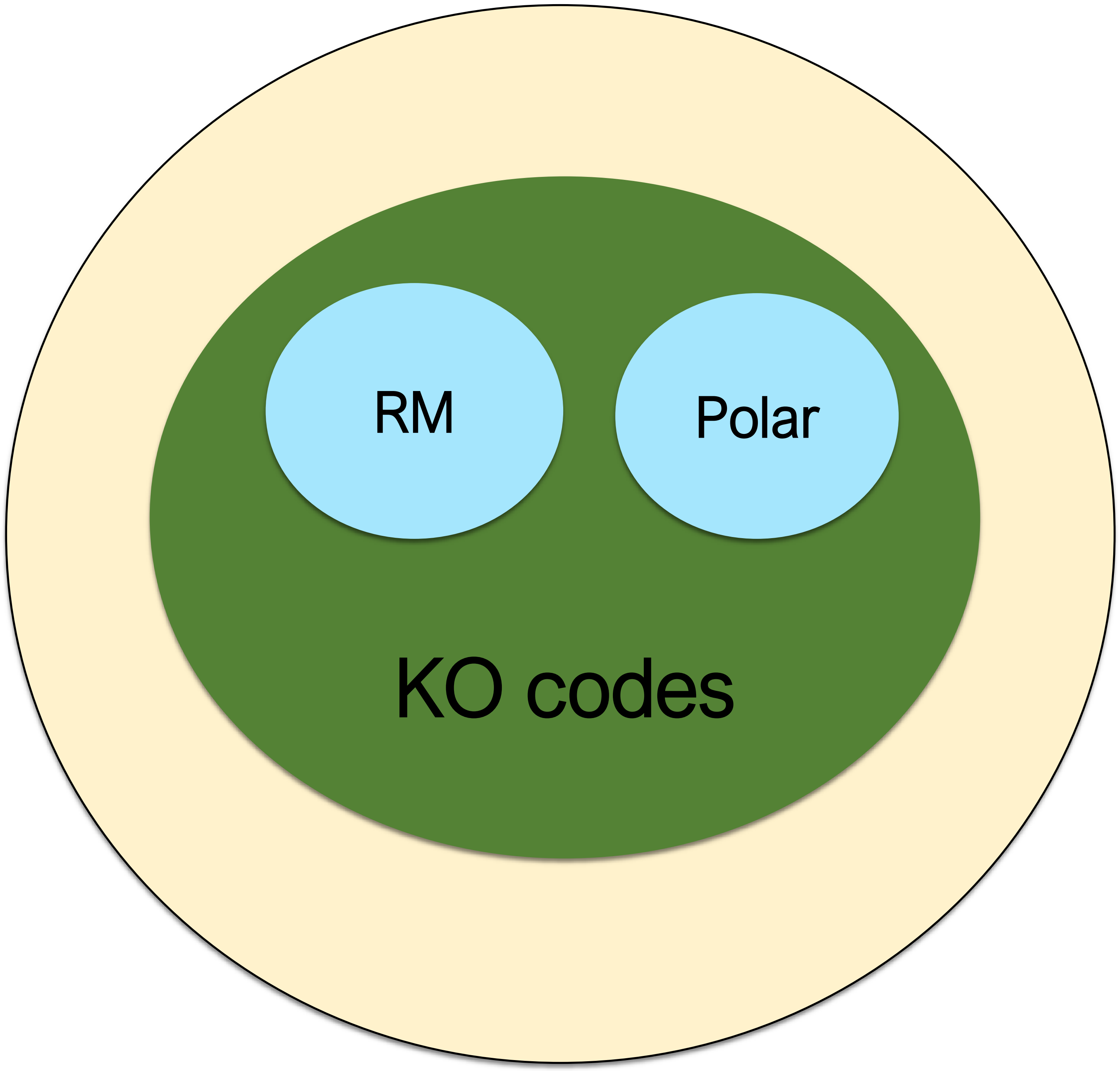



RM
Polar

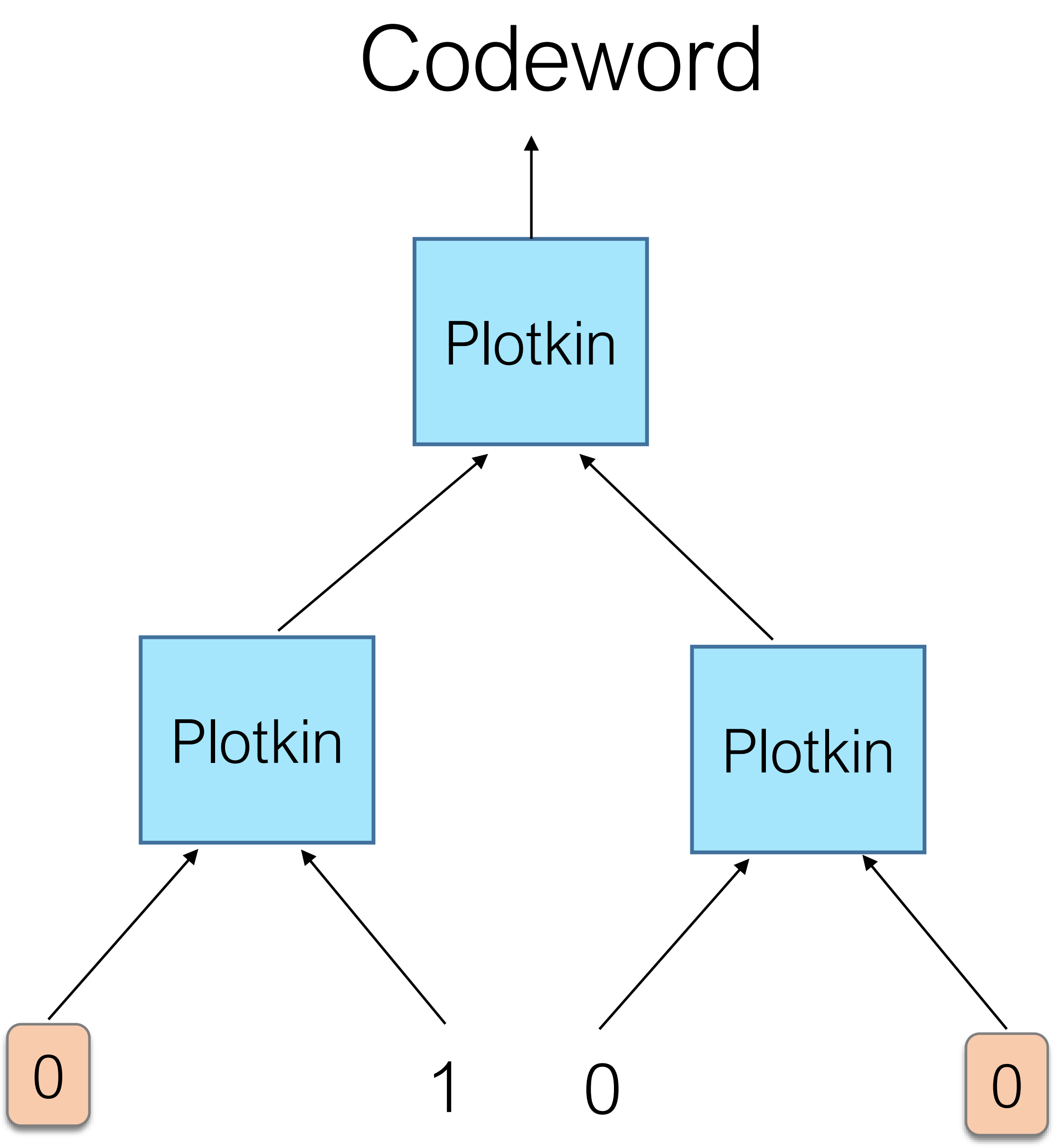
More codes?



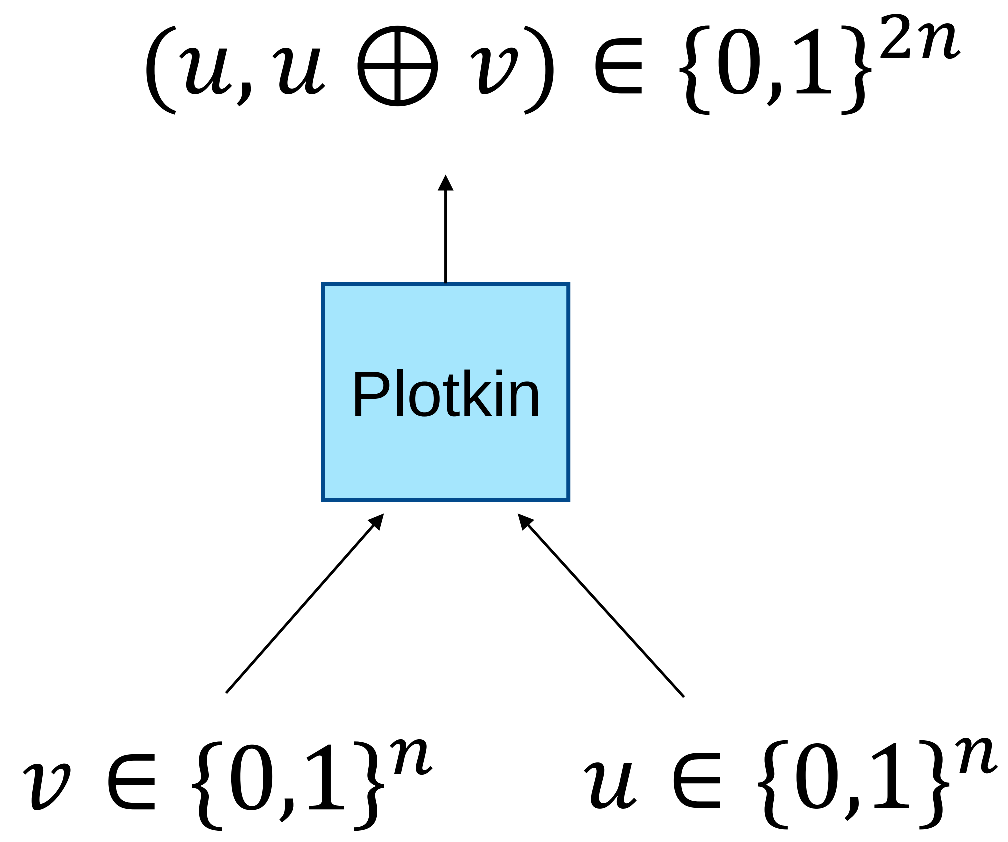
KO Codes



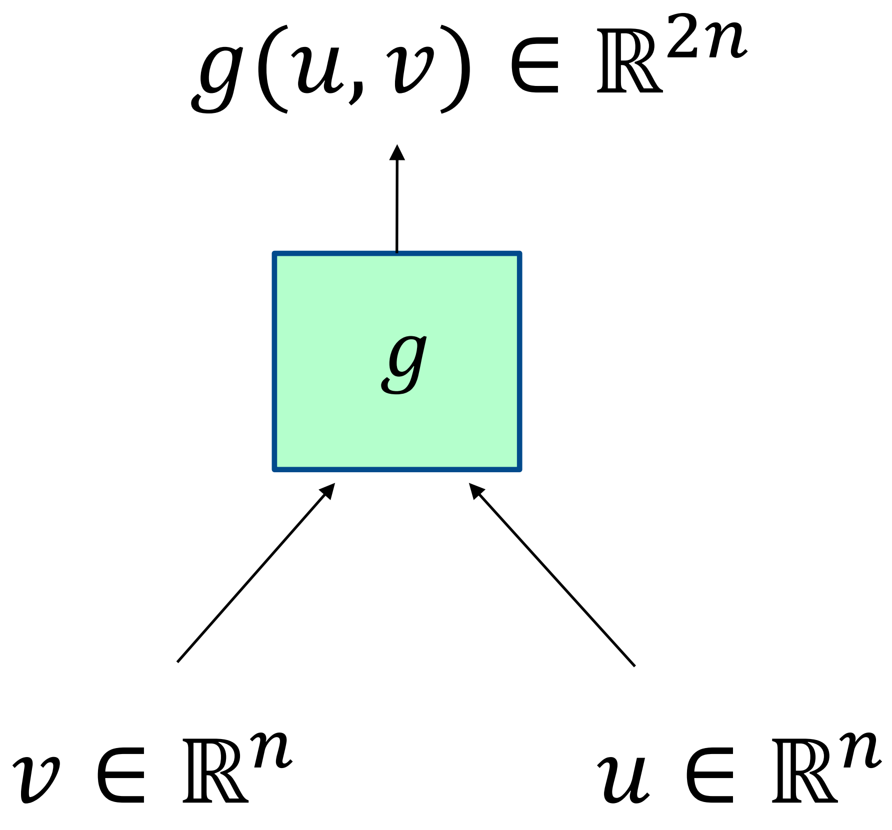
Encoding: RM and Polar



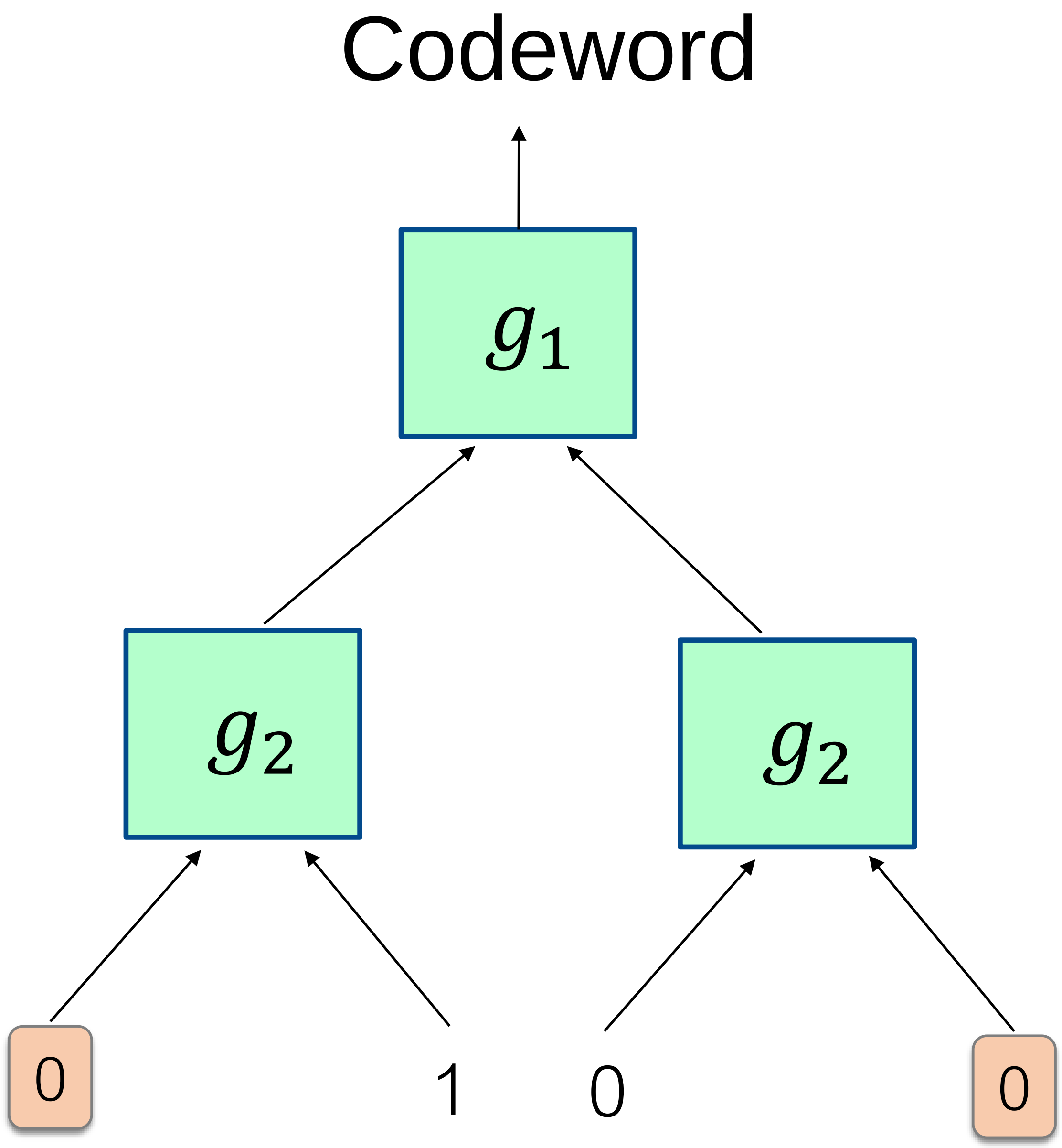
Plotkin mapping



KO neural network



KO encoder



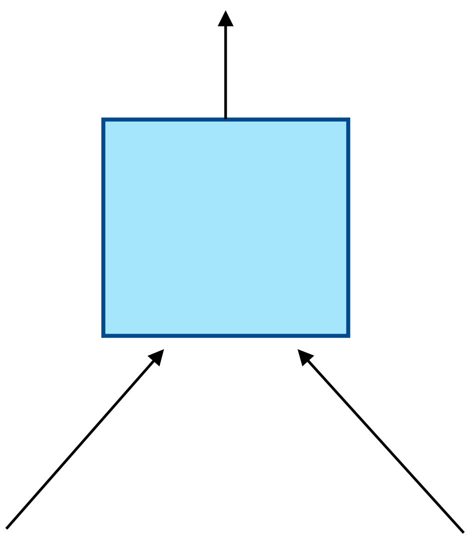
Decoder

- Matching decoder for KO encoder?

- **Dumer's decoder / Successive Cancellation (SC)**

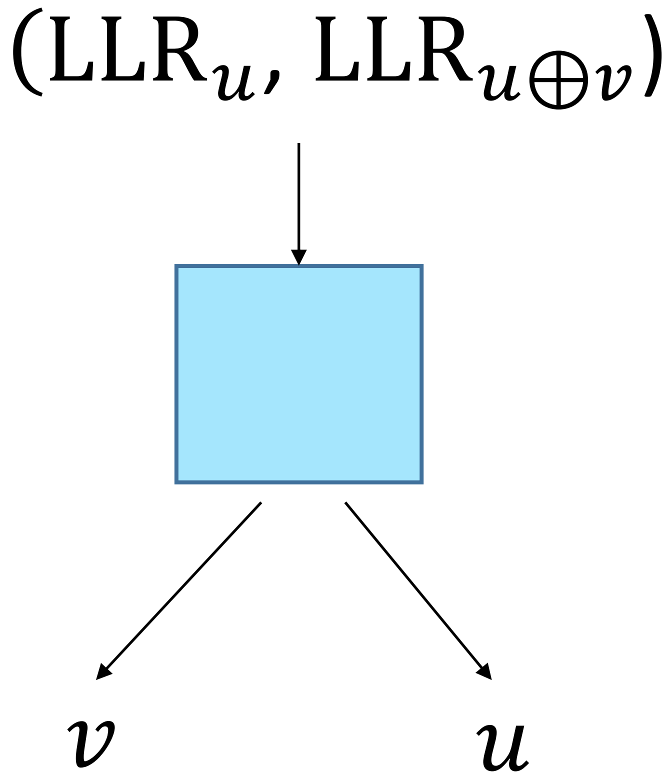
Plotkin revisited

$$(u, u \oplus v) \in \{0,1\}^{2n}$$



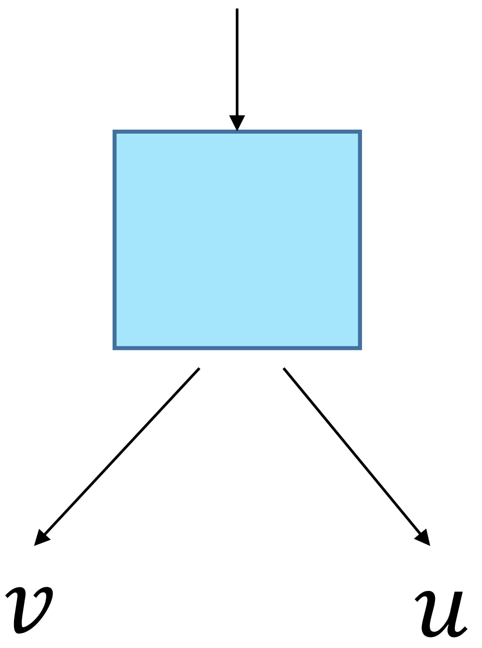
$$v \in \{0,1\}^n \qquad u \in \{0,1\}^n$$

Decoding



SC decoder

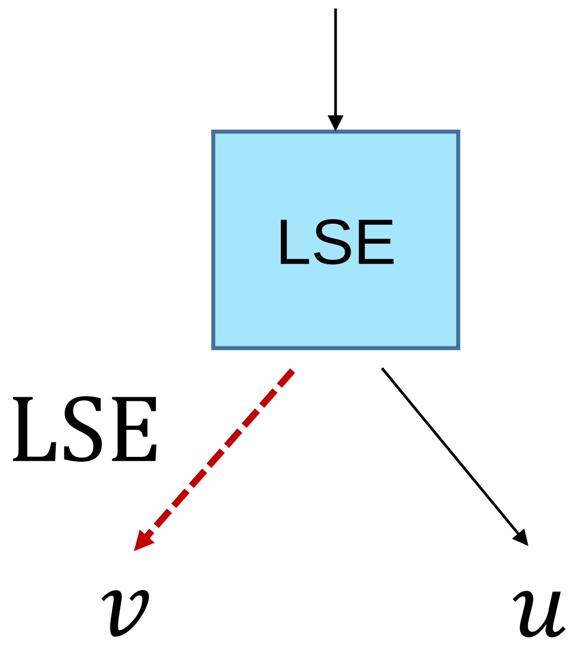
$$(\mathrm{LLR}_u,\ \mathrm{LLR}_{u\oplus v})$$



Dumer, 2004-06
Arikan, 2009

SC decoder

$$(\mathrm{LLR}_u, \mathrm{LLR}_{u \oplus v})$$

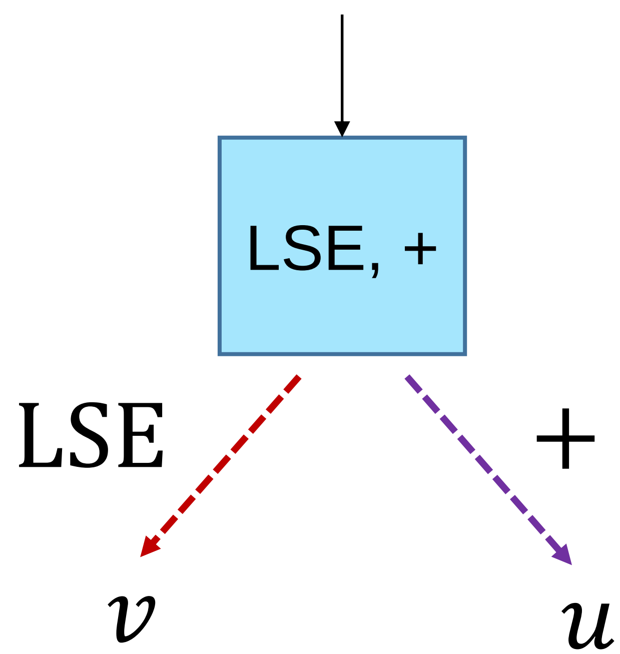


First Decode v

Dumer, 2004-06
Arikan, 2009

SC decoder

$$(\mathrm{LLR}_u, \mathrm{LLR}_{u \oplus v})$$

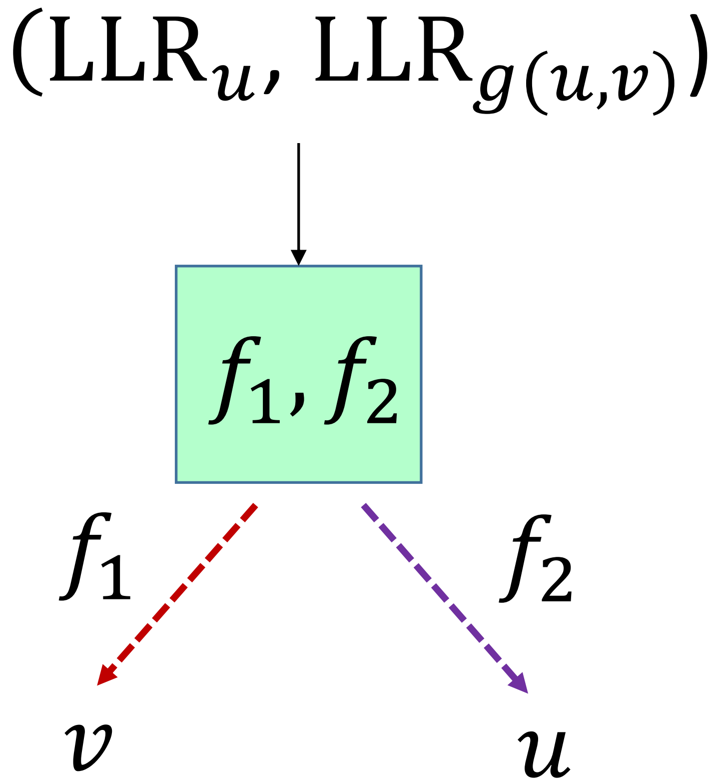


First Decode v

Next Decode u

Dumer, 2004-06
Arikan, 2009

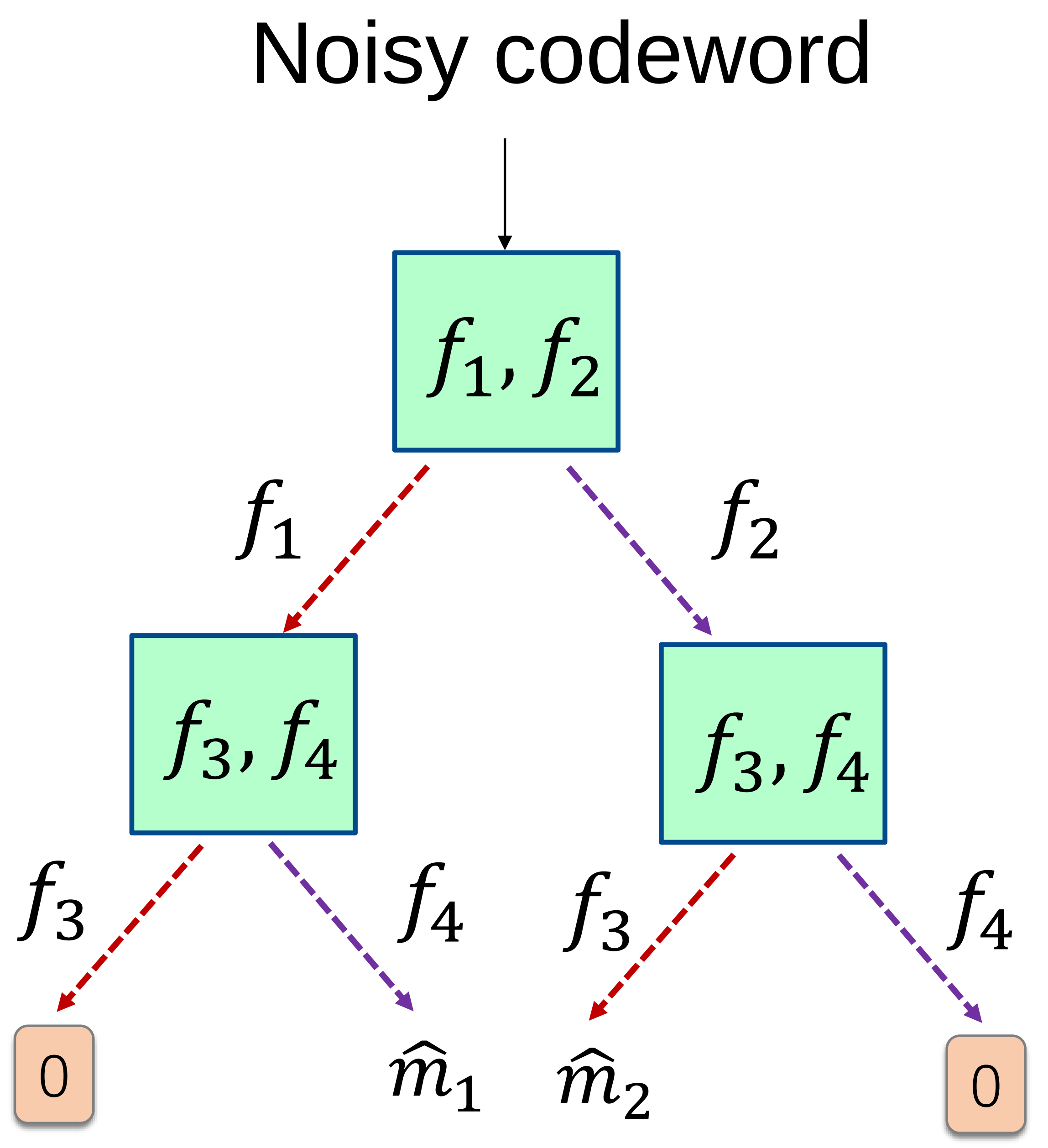
KO decoder



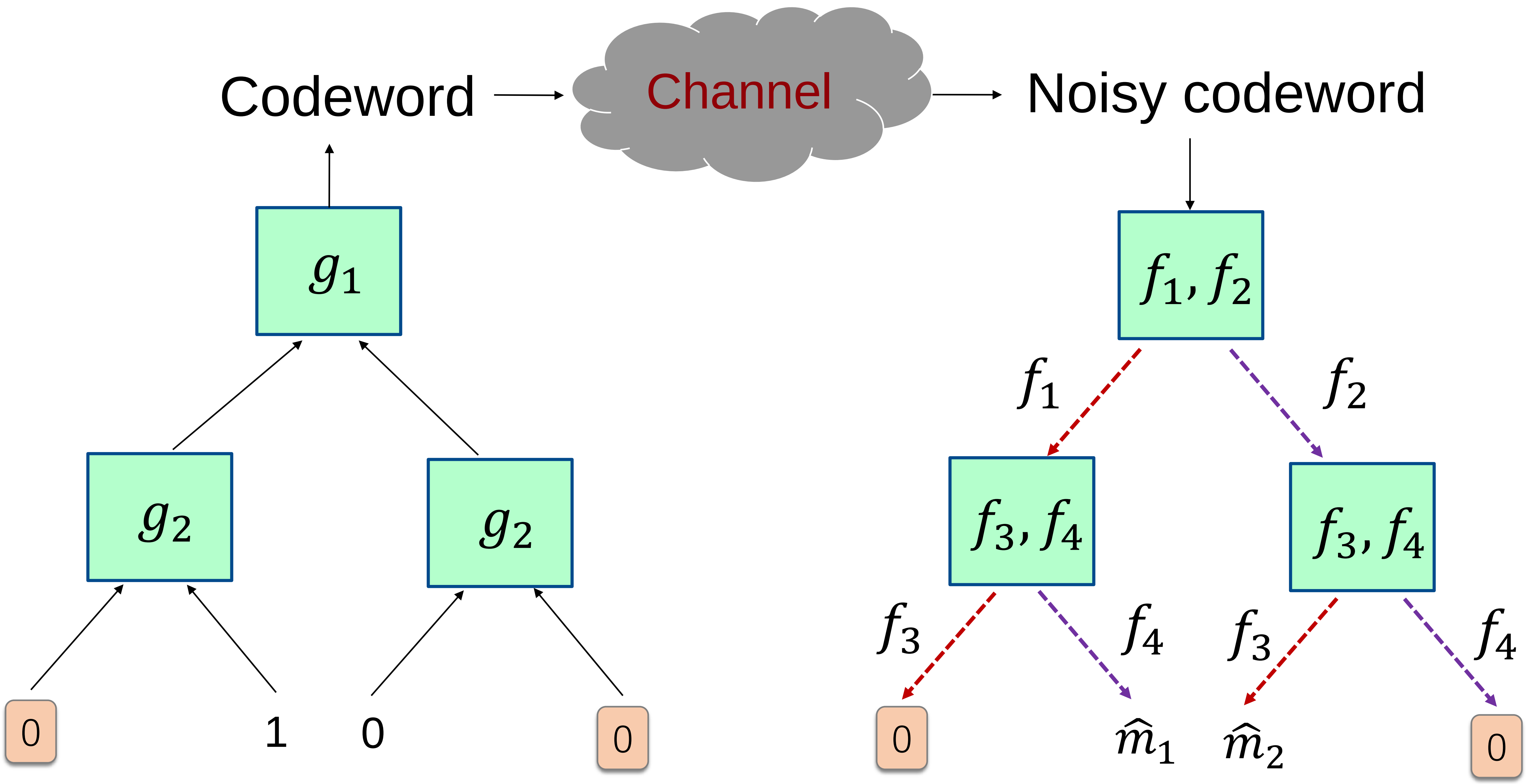
First Decode v

Next Decode u

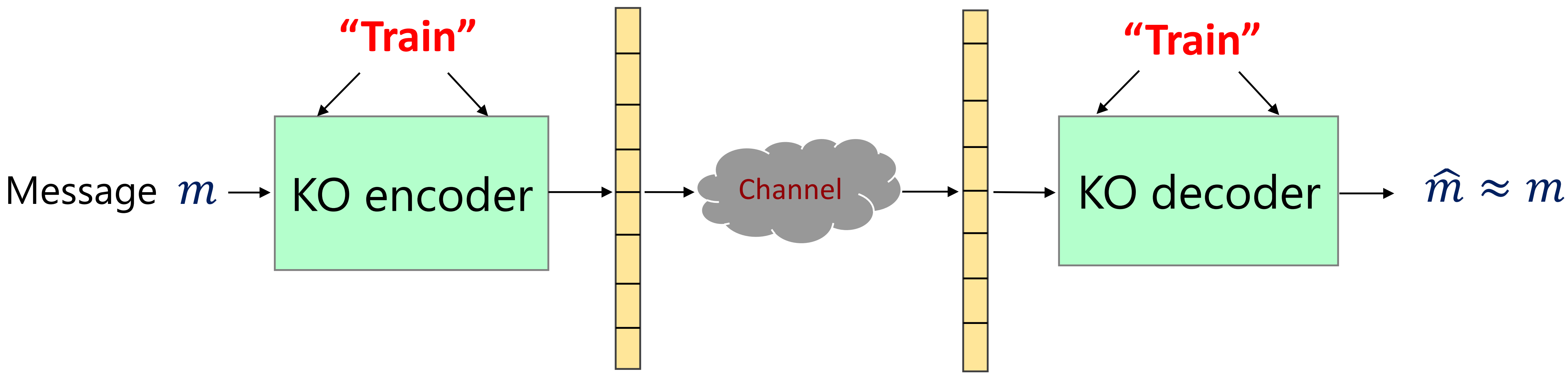
KO decoder


Noisy codeword
f_1, f_2
f_1
f_2
f_3, f_4
f_3, f_4
f_3
f_4
f_3
f_4
0
$\widehat{m}_1$
$\widehat{m}_2$
0

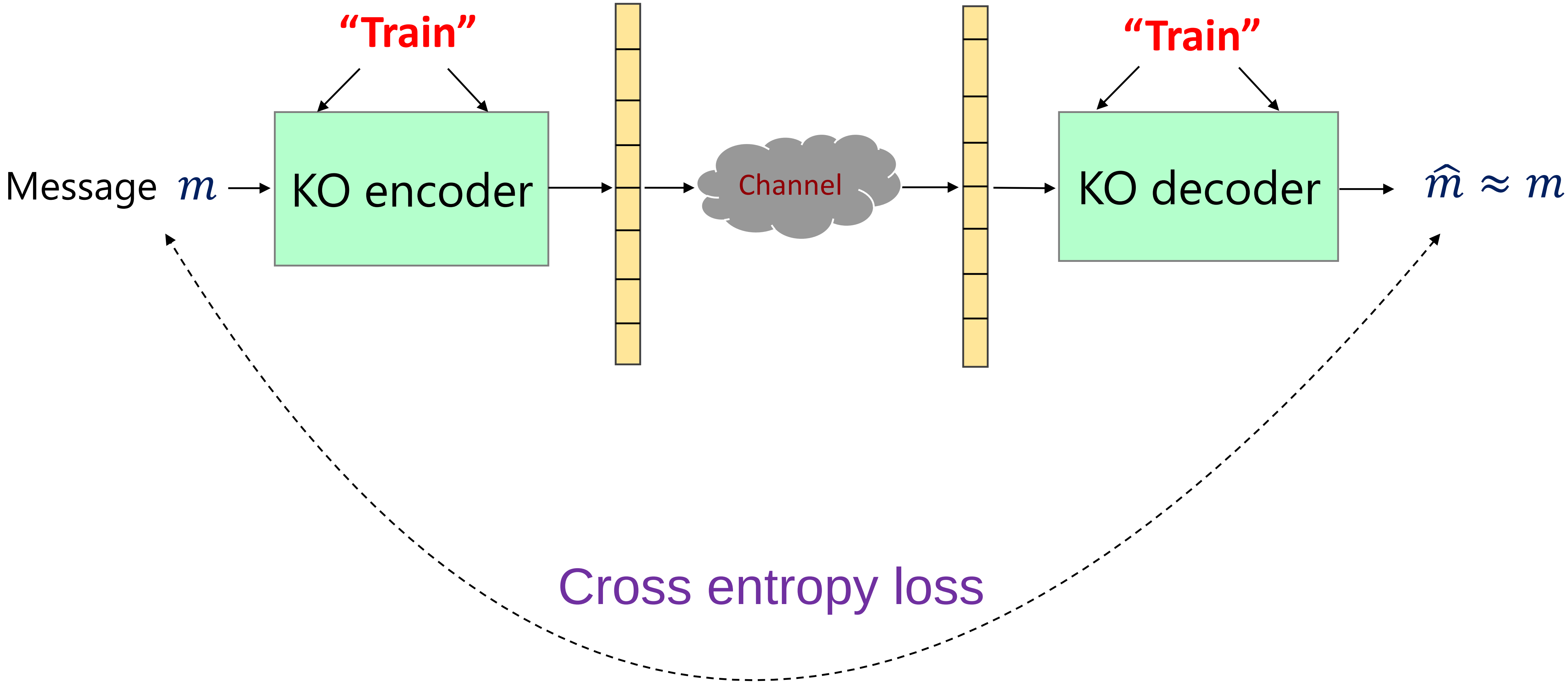
KO (encoder, decoder)



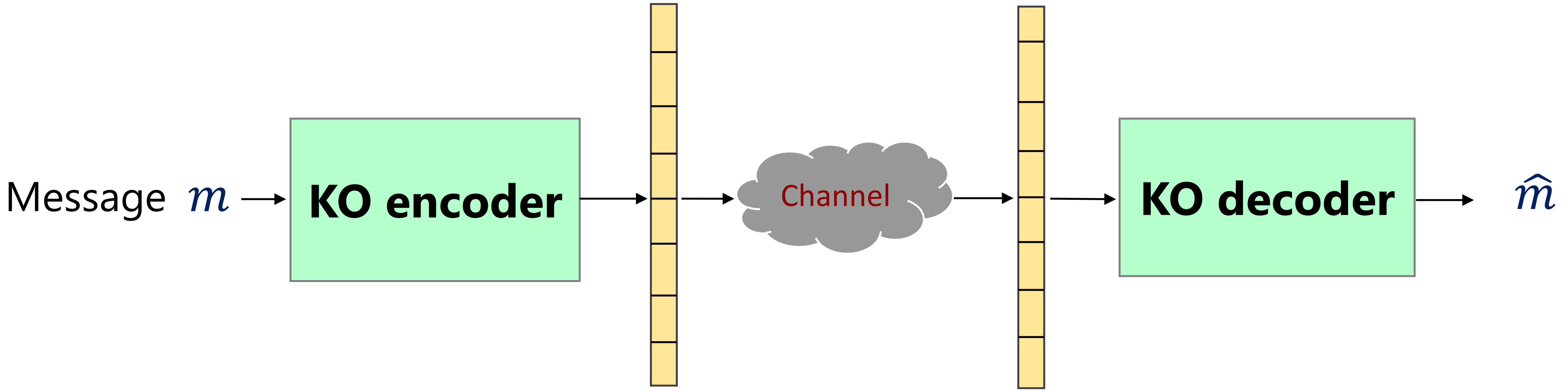
Training KO Codes



Training KO Codes



Testing



Performance metrics

- Reliability

- Computational complexity

Baselines

- KO codes vs. RM codes

- KO codes vs. Polar codes

Setup

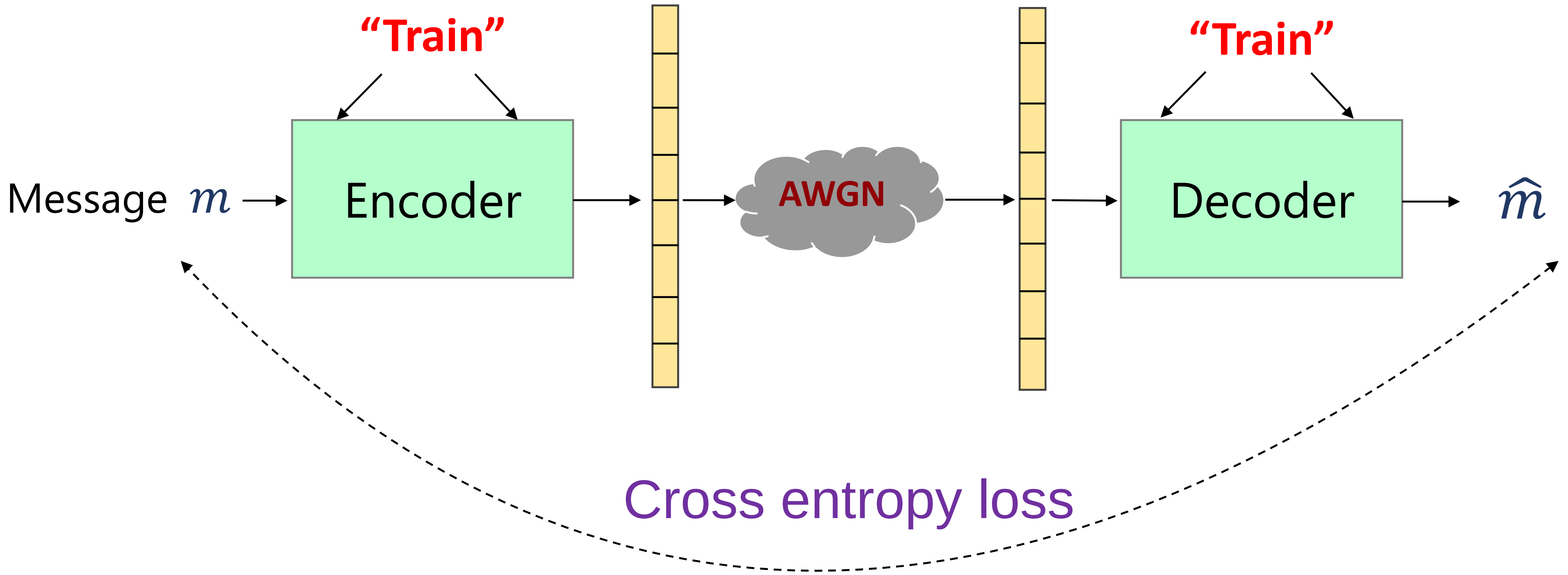
- Train and test on the same channel
  - AWGN
- Robustness: Train and test on different channels
  - Rayleigh fading

Setup

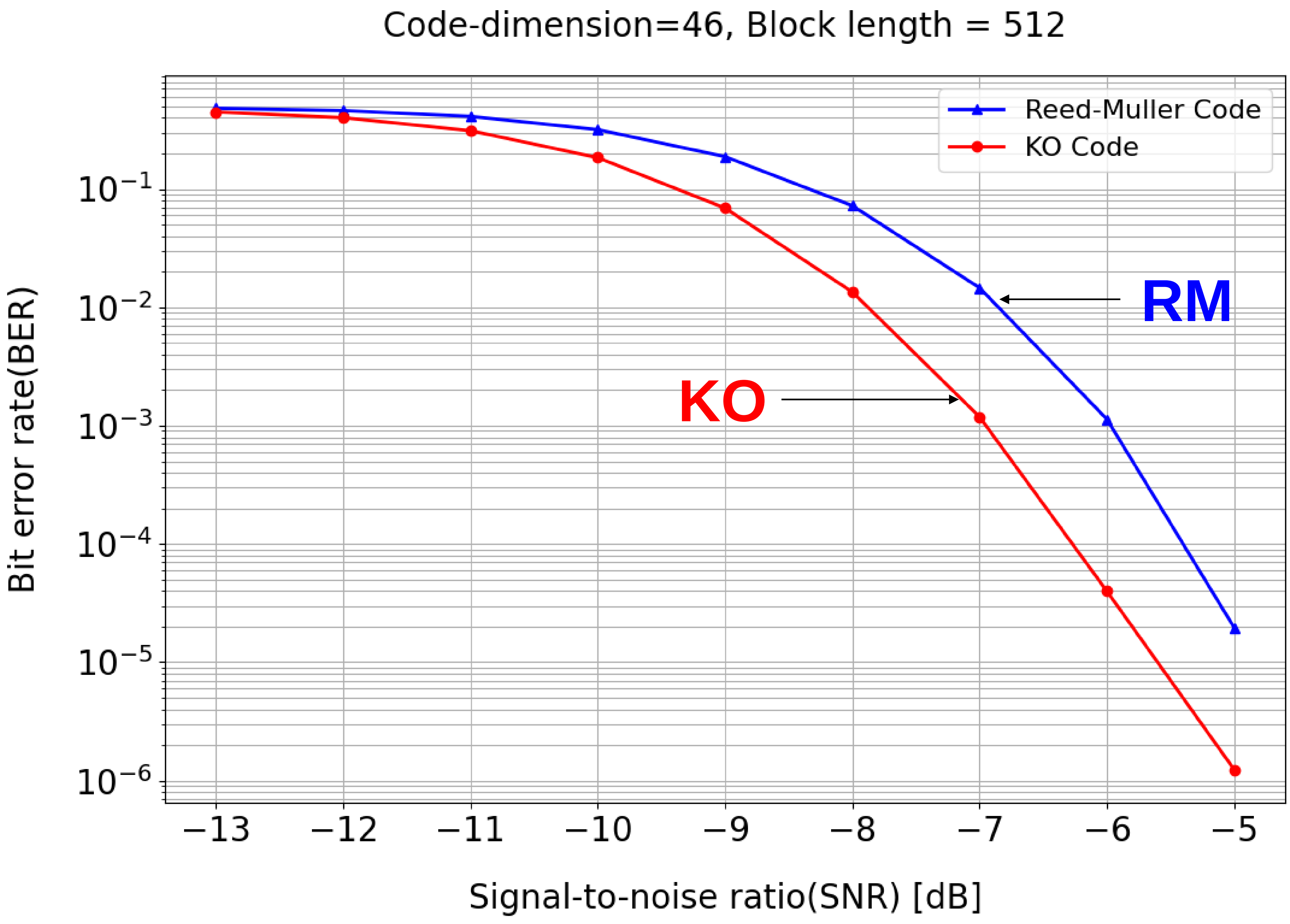
- Train and test on the same channel
  - AWGN
- Robustness: Train and test on different channels
  - Rayleigh fading

Setup #1: AWGN

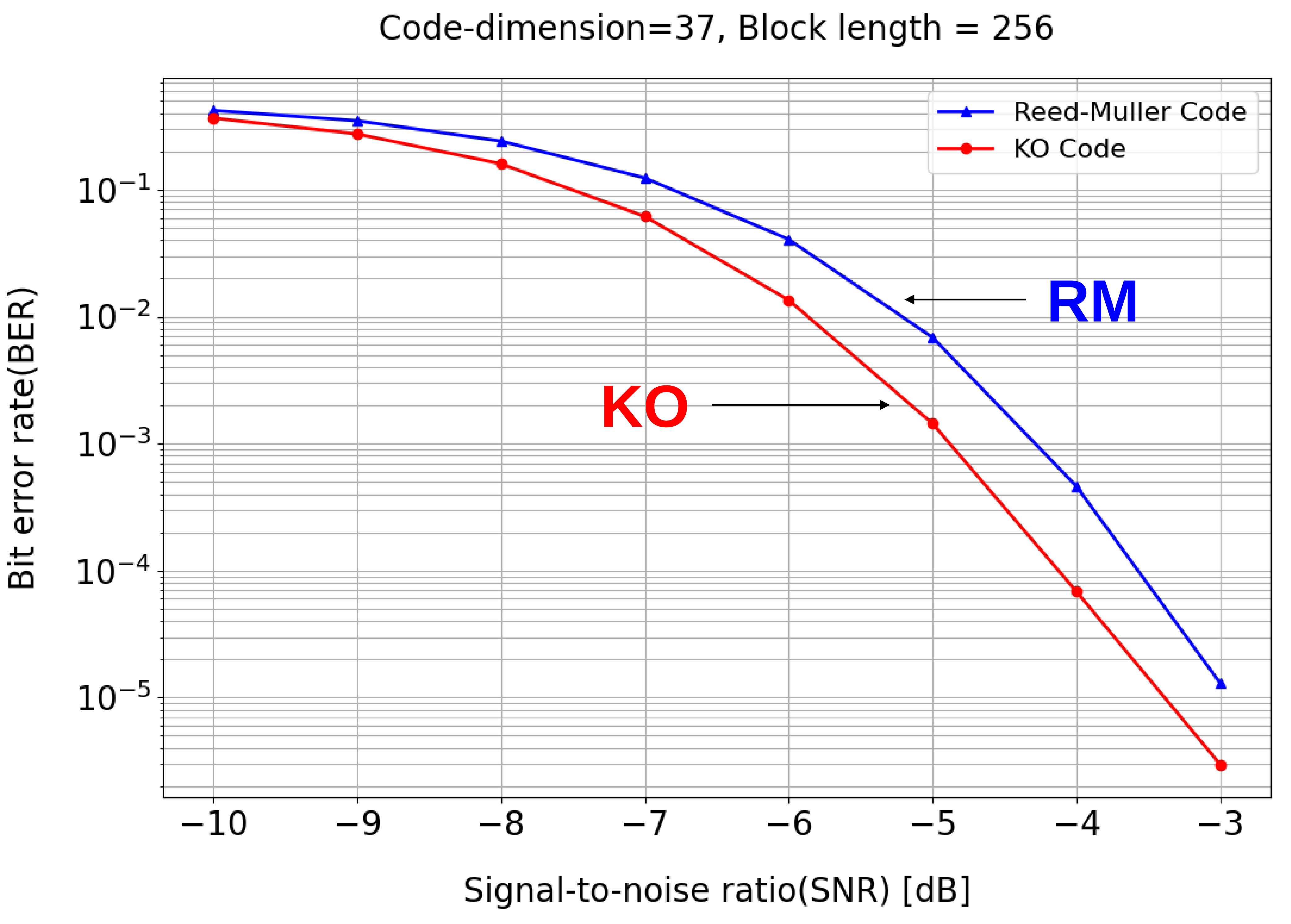
- Train and test on AWGN



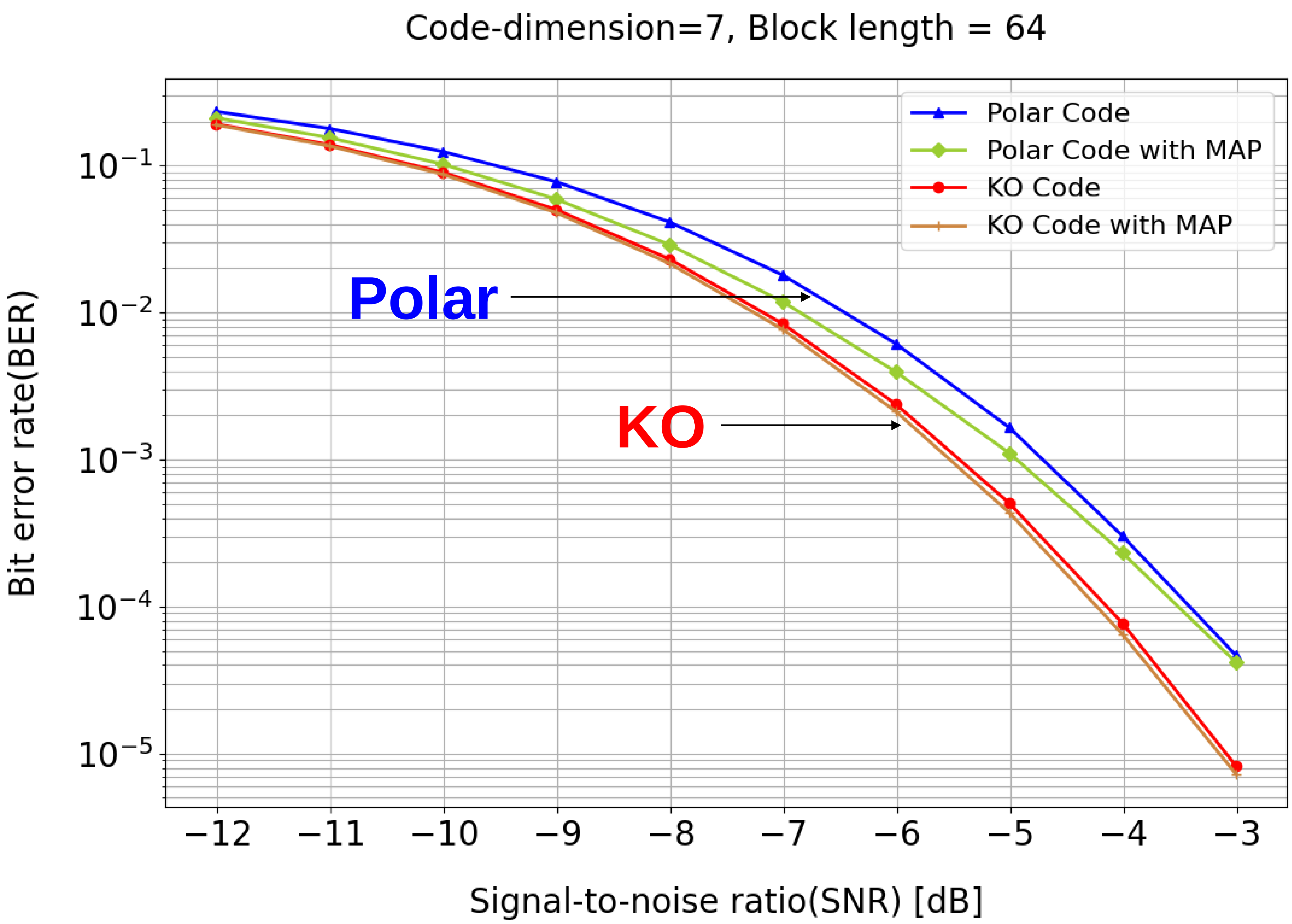
KO codes beat RM



KO codes beat RM



KO beats Polar



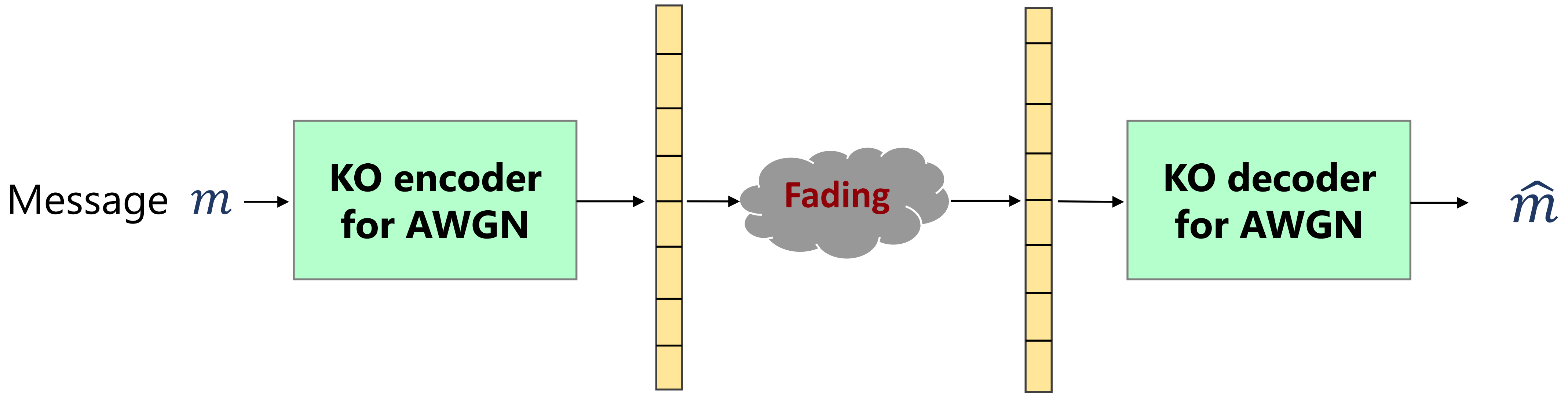
Setup

- Train and test on the same channel
  - AWGN

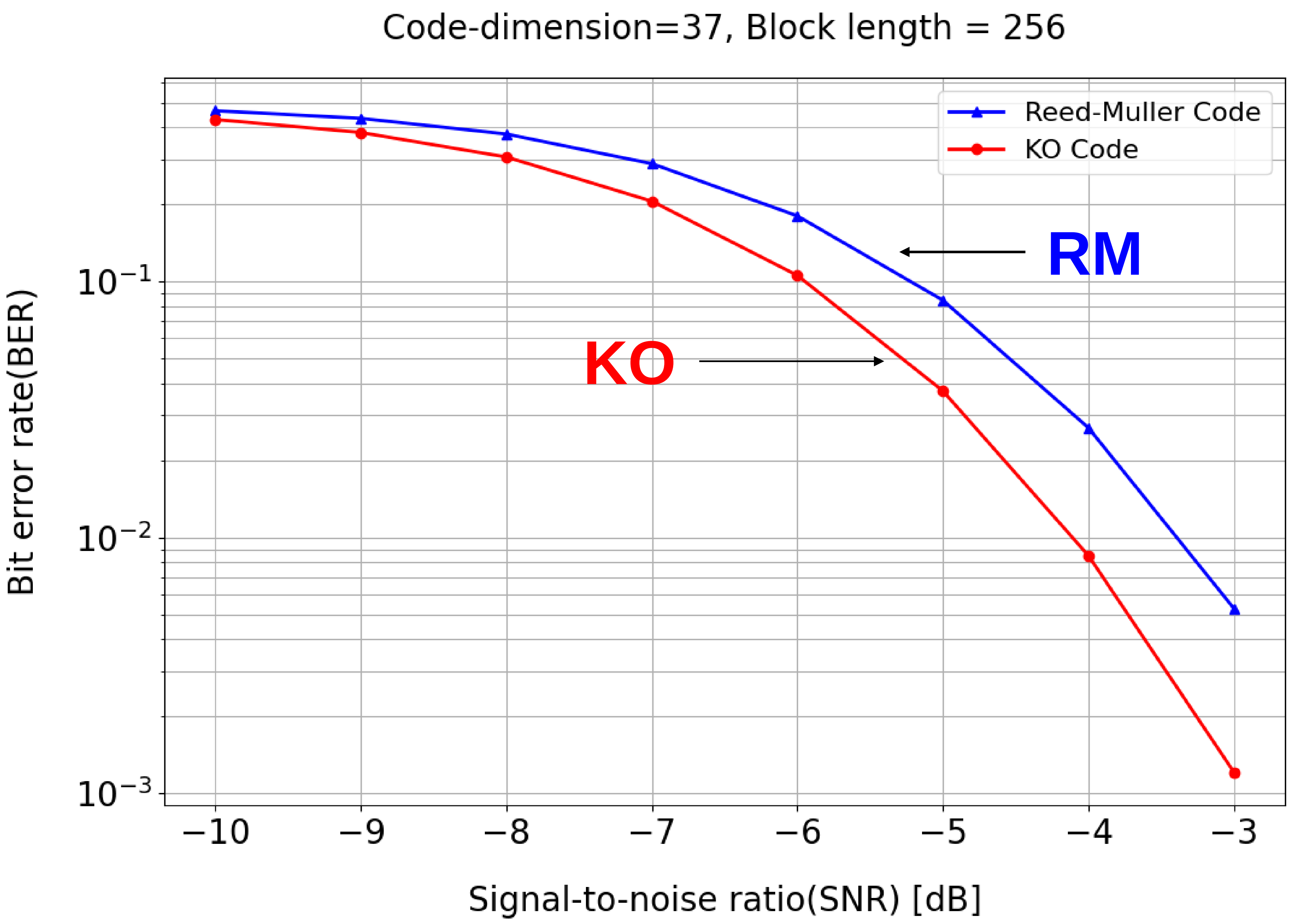
- Robustness: Train and test on different channels
  - Rayleigh fading

Setup #2: Robustness

- Train on AWGN → Test under Rayleigh fading



Robustness: Fading channel



Setup

- Train and test on the same channel
  - AWGN

- Robustness: Train and test on different channels
  - Rayleigh fading

Complexity

- Computational complexity: O(n log n)
  - KO codes ≈ RM codes

- Number of operations
  - RM codes (11k) ≪ KO codes (550k)

Complexity

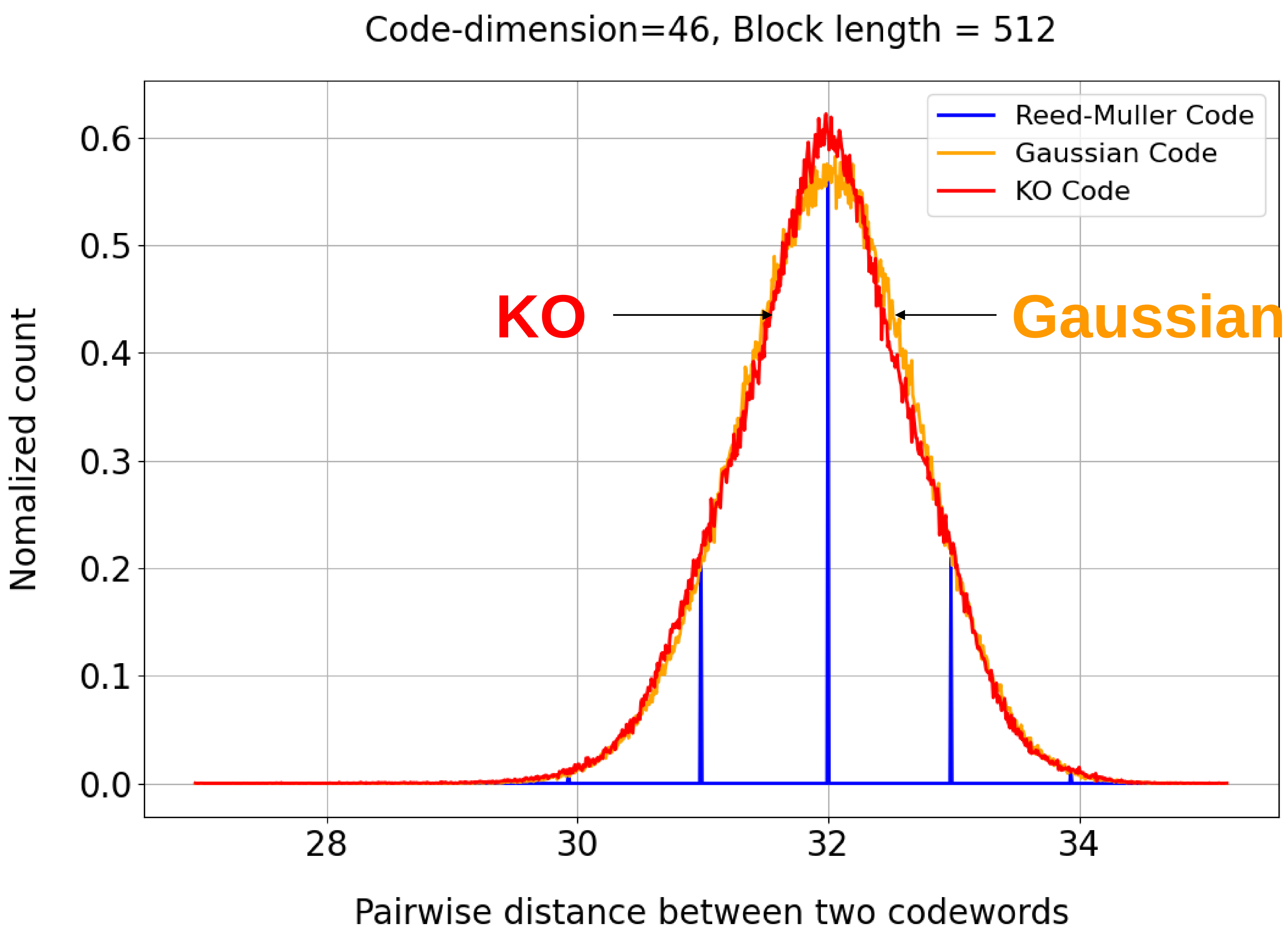
- Computational complexity: $O(n \log n)$
  - KO codes $\approx$ RM codes
- Number of operations
  - RM codes (11k) $\approx$ Tiny KO (44k) $\ll$ KO codes (550k)

Why are KO codes good

Why are KO codes good

- Surprising resemblance to Gaussian codes!

Gaussian like!



Future directions

- Training with complex decoding algorithms
  - Recursive Projection Aggregation (RPA)
  - SC + list decoder

Future directions

- Training with complex decoding algorithms
  - Recursive Projection Aggregation (RPA)
  - SC + list decoder

- Learning the frozen bits: Liao et al, 2020

Future directions

- Training with complex decoding algorithms
  - Recursive Projection Aggregation (RPA)
  - SC + list decoder

- Learning the frozen bits: Liao et al, 2020

- Commercialization
  - Hardware implementation
  - Standardization (6G?)

Future directions

- Discover new coding structures
  - Recursive: this work
  - Graph: LDPC
  - Sequential: convolutional
  - What is the best structure?
- Theoretical analysis
- Beyond point-to-point: Network coding

Collaborators





La Fin

Thank you!