Algebra Remains a Cradle for Improving Codes

Li Chen

Sun Yat-sen University

Sydney, Australia

Oct. 3, 2025

Outline

- Review of Channel Codes
- Algebraic Decoding vs. Probabilistic Decoding
- Short-to-Medium Length Codes
- U-UV Codes
- GII Codes
- End / Beginning ...

Review of Channel Codes

C bits/sym.

Channel Capacity & Coding Theorem
Shannon
1948

$C(N,\ K,\ d)$

randomness

N being large

polarization

$R = K/N \to C$

<u>Classic Codes</u>

<u>Modern Codes</u>

Hamming codes
Hamming
1947

RM codes
Muller
Reed
1954

Conv. codes
P. Elias
1955

BCH codes
Bose
Chaudhuri
Hocquenghem
1959

RS codes
Reed
Solomon
1960

LDPC codes
Gallager
1960

Conct. codes
Forney
1965

AG codes
Goppa
1981

TCM
Ungerböck
1982

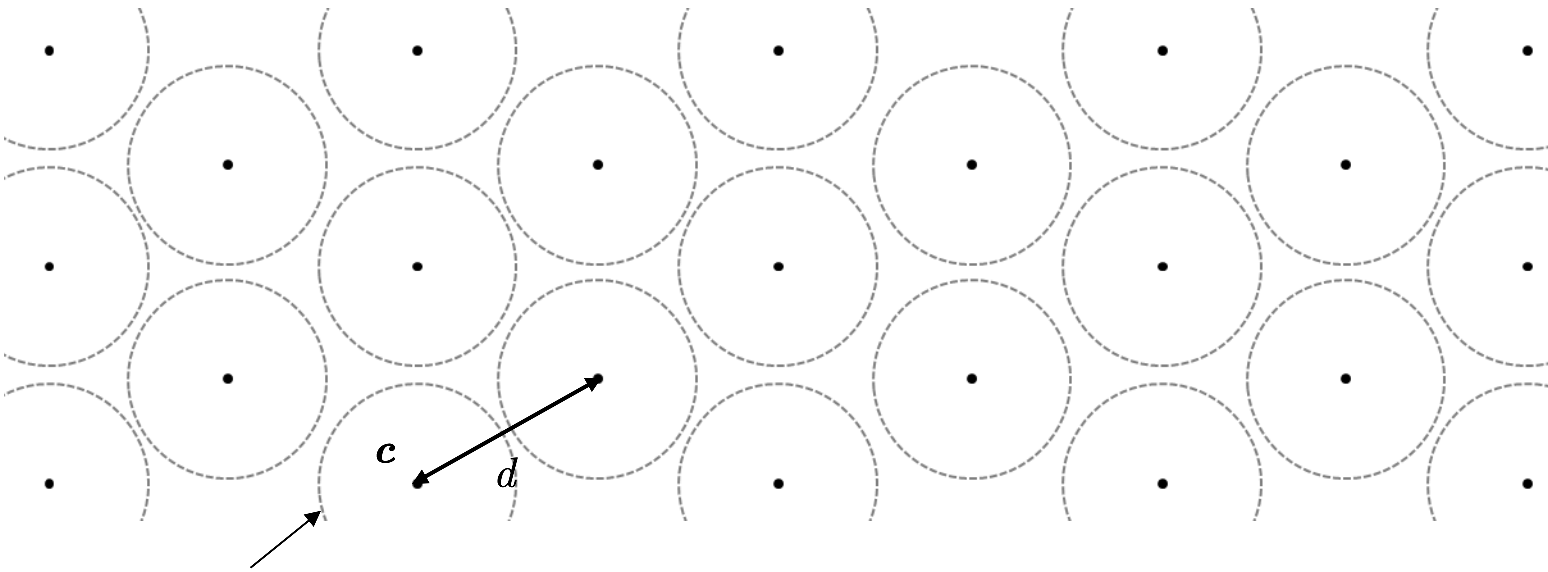
Turbo codes
Berrou
Glavieux
Thitimajshima
1993

LDPC codes (rediscovery)
Wiberg
MacKay
Neal
1995, 1996

Polar codes
Arıkan
2008

Review of Channel Codes

- Encoding



Hamming sphere

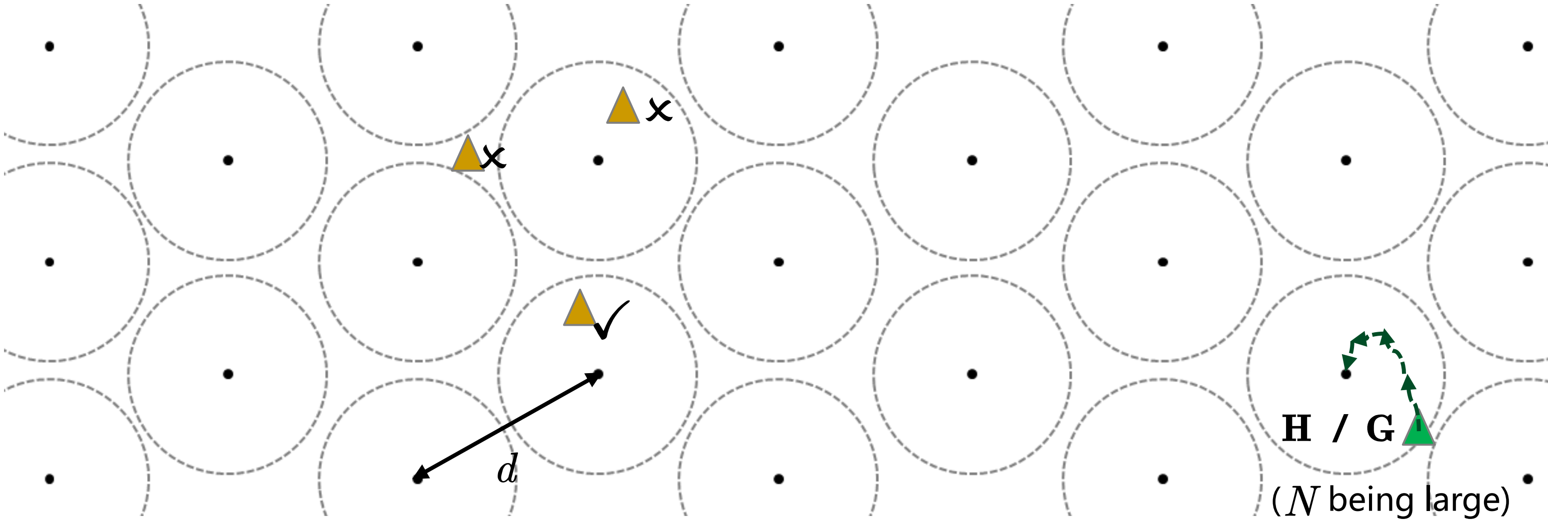
Encoding (Φ): $\boldsymbol{m} \overset{\mathbf{G}}{\mapsto} \boldsymbol{c},\ \mathbf{dim}(\boldsymbol{m}) = K \ \&\ \mathbf{dim}(\boldsymbol{c}) = N$

$$\{\mathbb{F}_q^N\} = \underbrace{\{\boldsymbol{c}\}}_{\text{codebook } \mathcal{C}} \cup \{\boldsymbol{c}+\boldsymbol{e}\}$$

● : codewords
Φ : encoding func.
$\mathbf{G}$: generator mtx.

Review of Channel Codes

- Decoding



Decoding (Φ^{-1}):

$\boldsymbol{r} = \boldsymbol{c} + \boldsymbol{e}$

in Hamming domain

$$\Phi^{-1}(\boldsymbol{r}) \mapsto \hat{\boldsymbol{c}} \;/\; \hat{\boldsymbol{m}}$$

$$\hat{\boldsymbol{c}} = \boldsymbol{r} - \hat{\boldsymbol{e}}$$

$\boldsymbol{y} = \mathcal{M}(\boldsymbol{c}) + \boldsymbol{n}$

in probability domain

$$\Phi^{-1}(\boldsymbol{y}) \mapsto \hat{\boldsymbol{c}} \;/\; \hat{\boldsymbol{m}}$$

$$\begin{array}{cccc} p(c_0), & p(c_1), & \ldots, & p(c_{N-1}) \\ \downarrow & \downarrow & & \downarrow \\ \hat{c}_0 & \hat{c}_1 & \cdots & \hat{c}_{N-1} \end{array}$$

Φ^{-1}	: decoding func.
$\mathcal{M}$	: modulation
$\mathbf{H}$	: parity-check mtx.
$\boldsymbol{e}$	: error vector
$\boldsymbol{n}$	: noise vector

Review of Channel Codes

- Decoding

$\boldsymbol{r} = \boldsymbol{c} + \boldsymbol{e}$

in Hamming domain

$\Phi^{-1}(\boldsymbol{r}) \mapsto \hat{\boldsymbol{c}} \,/\, \hat{\boldsymbol{m}}$

$\hat{\boldsymbol{c}} = \boldsymbol{r} - \hat{\boldsymbol{e}}$

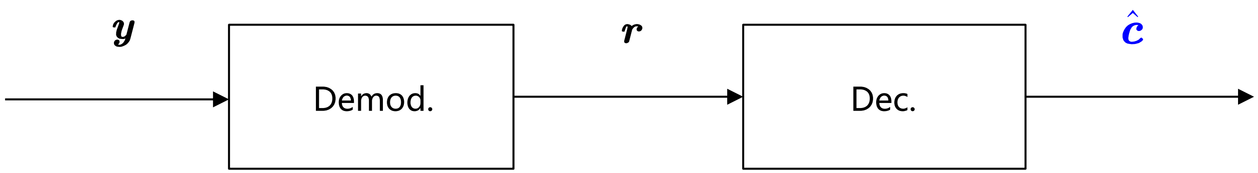
$\boldsymbol{y} = \mathcal{M}(\boldsymbol{c}) + \boldsymbol{n}$

in probability domain

$\Phi^{-1}(\boldsymbol{y}) \mapsto \hat{\boldsymbol{c}} \,/\, \hat{\boldsymbol{m}}$

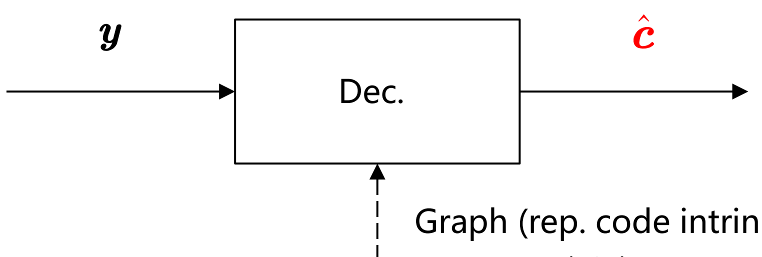
$$\begin{array}{cccc} p(c_0), & p(c_1), & \ldots, & p(c_{N-1}) \\ \downarrow & \downarrow & & \downarrow \\ \hat{c}_0 & \hat{c}_1 & \cdots & \hat{c}_{N-1} \end{array}$$

Algebraic decoding of classic codes



Probabilistic decoding of modern codes

$$\boxed{I(\boldsymbol{y};\hat{\boldsymbol{c}}) \geqslant I(\boldsymbol{y};\hat{\boldsymbol{c}})}$$



Graph (rep. code intrinsic structure)

$\boldsymbol{I}_{\text{apriori}} \quad p(y|c)$

$d;\ \boldsymbol{I}_{\text{apriori}}$

Algebraic Decoding vs. Probabilistic Decoding

- Algebraic Decoding

Encoding of a (7, 5, 3) Reed-Solomon (RS) code

Finite field: $\mathbb{F}_8 = \{0, 1, \sigma, \sigma^2, \ldots, \sigma^6\}$, where σ is a primitive element

Message $\boldsymbol{m} = (m_0, m_1, \ldots, m_4)$

$$m(x) = m_0 + m_1 x + \cdots + m_4 x^4$$

Encoding (Φ):

Codeword $\boldsymbol{c} = (m(1), m(\sigma), m(\sigma^2), \ldots, m(\sigma^6))$

$$= (c_0, c_1, c_2, \ldots, c_6)$$

$(1, \sigma, \sigma^2, \ldots, \sigma^6) \in \mathbb{F}_8 \backslash \{0\}$ are code locators.
Their evaluation order can be arbitrary.

Algebraic Decoding vs. Probabilistic Decoding

- Algebraic Decoding

Decoding of the RS code

Received word $\boldsymbol{r} = (r_0, r_1, \ldots, r_6)$

$$r(x) = r_0 + r_1 x + \cdots + r_6 x^6$$

Syndrome based decoding (Φ_{syn}^{-1}):

Syndromes $S_1 = r(\sigma), S_2 = r(\sigma^2)$

$$S(x) = S_1 + S_2 x$$

Key equation $\Omega(x) = \Lambda(x) S(x) \mod x^2$

$\xrightarrow{\text{BM algorithm}} \hat{\boldsymbol{e}} \to \hat{\boldsymbol{c}}$

Interpolation based decoding (Φ_{int}^{-1}):

Points $(1, r_0), (\sigma, r_1), \ldots, (\sigma^6, r_6)$

Int. polynomial $Q(x,y),\ y - \hat{m}(x) \mid Q(x,y)$

$\xrightarrow{\text{GS algorithm}} \hat{\boldsymbol{m}} / \hat{\boldsymbol{c}}$

BM: Berlekamp-Massey
GS: Guruswami-Sudan

Algebraic Decoding vs. Probabilistic Decoding

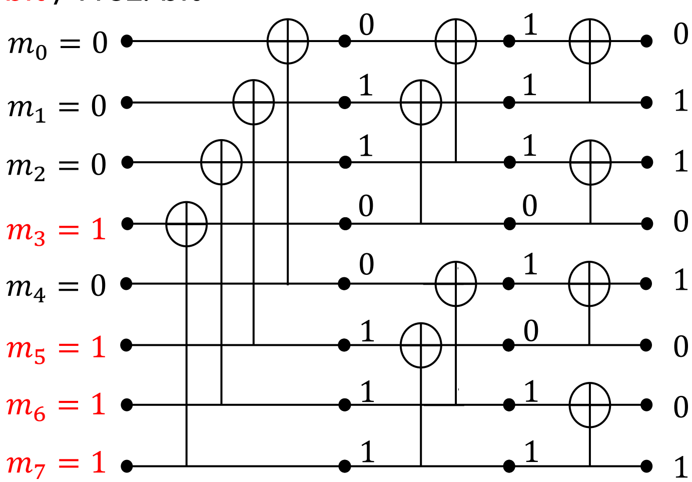
- Probabilistic Decoding

<u>Encoding of an (8, 4, 4) polar code</u>

Arıkan kernel

$$\mathbf{F} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \longrightarrow \mathbf{G} = \mathbf{F}^{\otimes 3} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{g}_0 \\ \boldsymbol{g}_1 \\ \boldsymbol{g}_2 \\ \boldsymbol{g}_3 \\ \boldsymbol{g}_4 \\ \boldsymbol{g}_5 \\ \boldsymbol{g}_6 \\ \boldsymbol{g}_7 \end{bmatrix}$$

Info. bit / Froz. bit



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

Info. bits $\boldsymbol{m}_{\mathcal{A}} = (1,1,1,1)$

Froz. bits $\boldsymbol{m}_{\mathcal{A}^c} = (0,0,0,0)$

$$\boldsymbol{c} = \boldsymbol{m}\mathbf{G} = \boldsymbol{m}_{\mathcal{A}}\mathbf{G}_{\mathcal{A}} = \boldsymbol{m}_{\mathcal{A}} \begin{bmatrix} \boldsymbol{g}_3 \\ \boldsymbol{g}_5 \\ \boldsymbol{g}_6 \\ \boldsymbol{g}_7 \end{bmatrix}$$

$$\downarrow$$

$$(\boldsymbol{c}\,\mathbf{G}^{-1})_{\mathcal{A}} = (\boldsymbol{c}\,\mathbf{G})_{\mathcal{A}} = \boldsymbol{m}_{\mathcal{A}}$$

Parity-check constraint

$$(\boldsymbol{c}\,\mathbf{G})_{\mathcal{A}^C} = \boldsymbol{m}_{\mathcal{A}^C} = (0,0,0,0)$$

Algebraic Decoding vs. Probabilistic Decoding

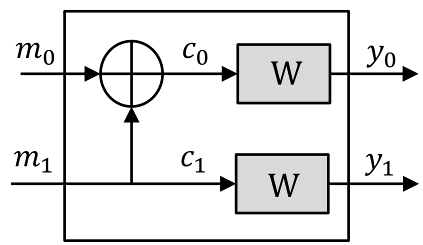
- Probabilistic Decoding

<u>Decoding of the polar code</u>

Decision LLRs ← - - - - - - - - - - - - - - - - - Channel observations

f function: $L_0^{(1)} = \ln \frac{1 + e^{L_0^{(0)} + L_1^{(0)}}}{e^{L_0^{(0)}} + e^{L_1^{(0)}}}$

g function: $L_1^{(1)} = (-1)^{\hat{m}_0} L_0^{(0)} + L_1^{(0)}$



$L_0^{(0)} = \ln \frac{P(y_0|0)}{P(y_0|1)}$

$L_1^{(0)} = \ln \frac{P(y_1|0)}{P(y_1|1)}$

Parity-check constraint

$(\boldsymbol{c}\,\mathbf{G})_{\mathcal{A}^C} = (0,0,0,0)$ ⟶

Info. bit: $\hat{m}_i = \begin{cases} 0, & \text{if } L_i^{(0)} \geq 0; \\ 1, & \text{if } L_i^{(0)} < 0. \end{cases}$

Froz. bit: $\hat{m}_i = 0$

Algebraic Decoding vs. Probabilistic Decoding

Algebraic decoding of Classic codes

minimum distance d

weight distributions ($A_0, \ A_d, \ A_{d+1}, \ldots$)

- structure / algebra

error-correction efficiency: e.g., $2N/d$

decoding computation in $\mathbb{F}_q$

Probabilistic decoding of Modern codes

minimum distance d

code intrinsic structure $\mathbf{H} / \mathbf{G}$.

$N \geqslant 10\ 000$ bits

- randomness
- polarization

capacity approaching / achieving: $C - R < \epsilon$

decoding computation in $\mathbb{R}$.

ϵ : arbitrarily small positive
$\mathbb{R}$: real number field

Algebraic Decoding vs. Probabilistic Decoding

- Decoding: solving equation group implied by $\mathbf{H} \in$ "Algebra"

  a) "الجبر" (al-Khwarizmi, 9C AD) → "Algebra" (16C AD) → "代数" (19C AD)

  b) Algebraic decoding: solving the equation group in $\mathbb{F}_q$
  Probabilistic decoding: solving the equation group in $\mathbb{R}$

  $$\xrightarrow{(N \geqslant 10\ 000 \text{ bits})} C - R < \epsilon$$

- Is the conventional algebra in coding practice still important?

  a) How if $N \ll 10\ 000$ bits?

  b) How if $p(y|c)$ is absence?

  c) How if latency / decoding power consumption is critical?

Short-to-Medium Length Codes

- SML Codes: e.g., $N = 50 \sim 1000$ bits

Long codes $N \geq 10000$ bits

- Good decoding performance
- High latency / power consumption

SML codes

$N \leq 1000$ bits

$d \leq N(1-R)+1$

code intrinsic structure $\mathbf{H} / \mathbf{G}$

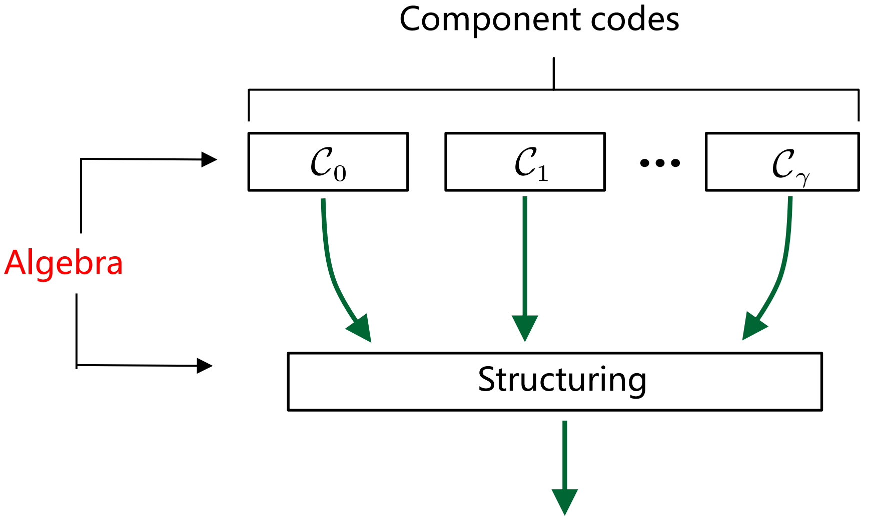
I_{apriori}

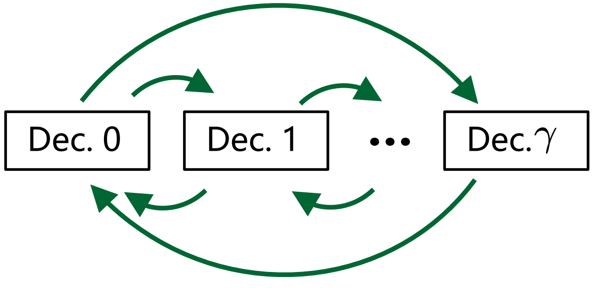
Algebra

- Decoding performance degrades
- Low latency / power consumption

Short-to-Medium Length Codes

- Structuring Multiple Component Codes

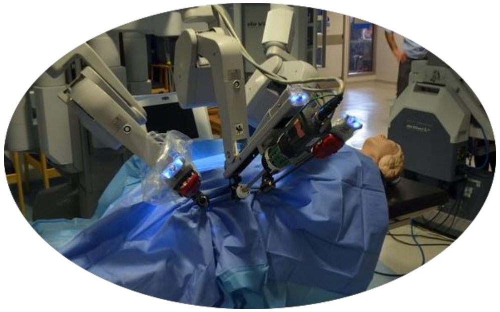




Decoding of component codes can be empowered by I_{apriori}

U-UV Codes

unmanned driving



wise information technology of medicine

factory automation

- Structure: Plotkin construction / Boolean sequences
- Component codes: BCH / eBCH codes

U-UV codes: (U | U + V) codes

U-UV Codes

- Boolean Sequences

$\mathbf{1}$	1	1	1	1	1	1	1	1
$\boldsymbol{v_0}$	0	0	0	0	1	1	1	1
$\boldsymbol{v_1}$	0	0	1	1	0	0	1	1
$\boldsymbol{v_2}$	0	1	0	1	0	1	0	1
$\boldsymbol{v_0 v_1}$	0	0	0	0	0	0	1	1
$\boldsymbol{v_0 v_2}$	0	0	0	0	0	1	0	1
$\boldsymbol{v_1 v_2}$	0	0	0	1	0	0	0	1
$\boldsymbol{v_0 v_1 v_2}$	0	0	0	0	0	0	0	1

The first order RM code

$$m_0\mathbf{1} + m_1\boldsymbol{v_0} + m_2\boldsymbol{v_1} + m_3\boldsymbol{v_2}$$

The second order RM code

$$m_0\mathbf{1} + m_1\boldsymbol{v_0} + m_2\boldsymbol{v_1} + m_3\boldsymbol{v_2} + m_4\boldsymbol{v_0}\boldsymbol{v_1} + m_5\boldsymbol{v_0}\boldsymbol{v_2} + m_6\boldsymbol{v_1}\boldsymbol{v_2}$$

The third order RM code

$$m_0\mathbf{1} + m_1\boldsymbol{v_0} + m_2\boldsymbol{v_1} + m_3\boldsymbol{v_2} + m_4\boldsymbol{v_0}\boldsymbol{v_1} + m_5\boldsymbol{v_0}\boldsymbol{v_2} + m_6\boldsymbol{v_1}\boldsymbol{v_2} + m_7\boldsymbol{v}_0\boldsymbol{v_1}\boldsymbol{v_2}$$

message symbols $m_i \in \mathbb{F}_2$

U-UV Codes

- Construction Initiatives

<u>Rediscovery of Plotkin construction</u>

$\mathbf{1}$	1	1	1	1	1	1	1	1
v_0	0	0	0	0	1	1	1	1
v_1	0	0	1	1	0	0	1	1
v_2	0	1	0	1	0	1	0	1
$v_0 v_1$	0	0	0	0	0	0	1	1
$v_0 v_2$	0	0	0	0	0	1	0	1
$v_1 v_2$	0	0	0	1	0	0	0	1
$v_0 v_1 v_2$	0	0	0	0	0	0	0	1

Arıkan kernel

$$\mathbf{F} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \longrightarrow \mathbf{F}^{\otimes 3} = \begin{array}{c} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \end{array} \begin{array}{c} v_0 v_1 v_2 \\ v_1 v_2 \\ v_0 v_2 \\ v_2 \\ v_0 v_1 \\ v_1 \\ v_0 \\ \mathbf{1} \end{array}$$

Arıkan kernel $\mathbf{F} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \longrightarrow \boxed{\mathbf{F}^{\otimes 3}}$

RM codes: $\mathbf{G}_{\mathrm{RM}}$ is defined based on row weight of $\mathbf{F}^{\otimes 3}$

Polar codes: subchannel capacity

$\longrightarrow \mathbf{F}^{\otimes 3} = \mathbf{G}_{\mathcal{A}} \cup \mathbf{G}_{\mathcal{A}^c}$

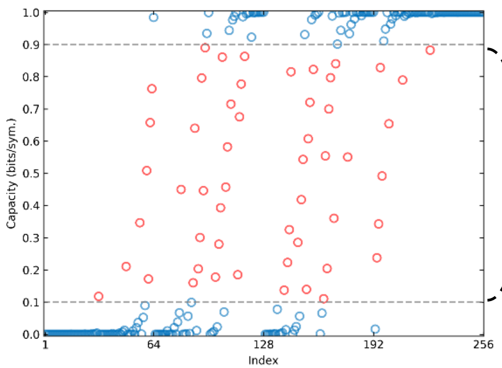
Plotkin kernel $\mathbf{F}' = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \longrightarrow \boxed{(\mathbf{F}')^{\otimes 3} = \left(\mathbf{F}^{\otimes 3}\right)^{\mathrm{T}}}$

$\longrightarrow$ (U | U + V) code

U-UV Codes

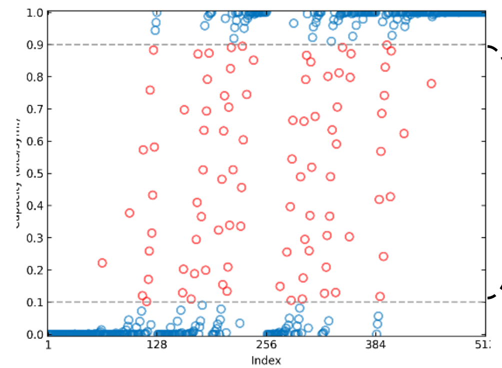
- Construction Initiatives

<u>Polarization theorem ($N \to \infty$) and polar coding practice (N = 256)</u>



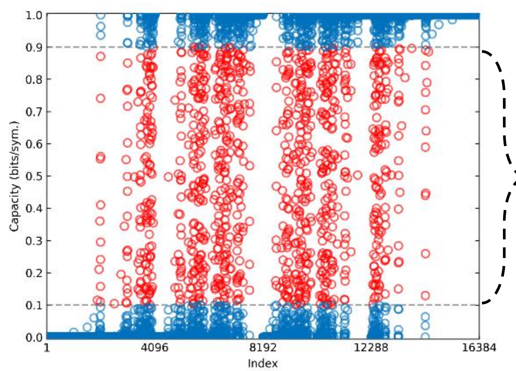
N = 256

20.7%



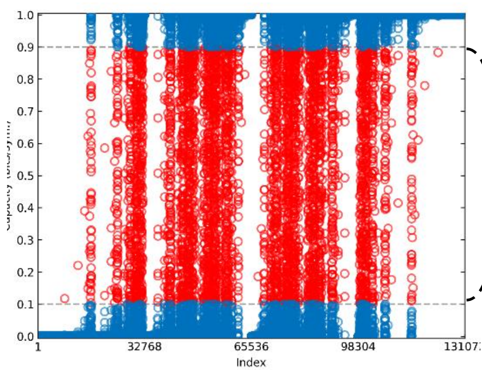
N = 512

18.6%



N = 16384

6.31%



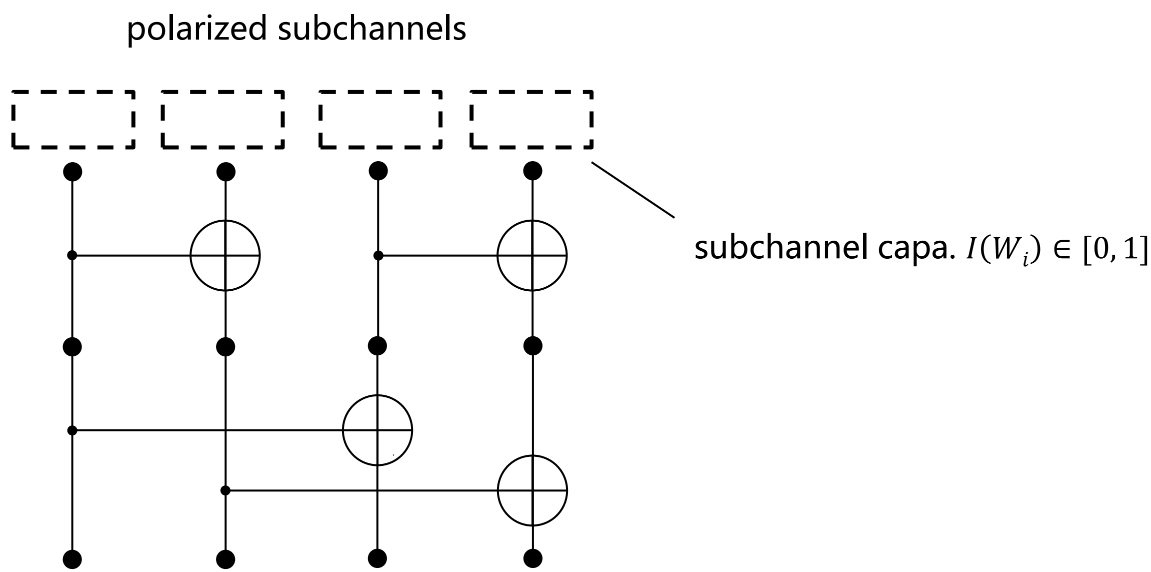
N = 131072

3.60%

AWGN with $E_s/N_0 = 0.0$ dB, Gaussian Approximation (GA)

U-UV Codes

- Construction Initiatives



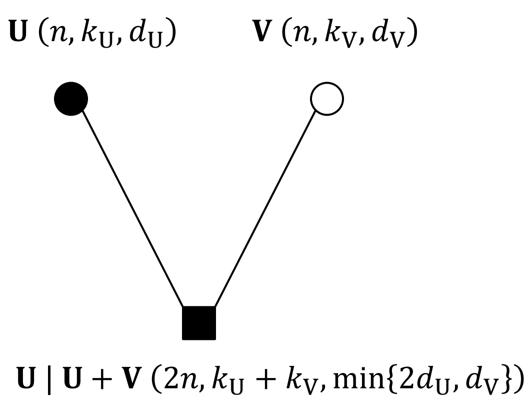
Polar codes
- Subchannels convey symbols
- Rate $R_i \in \{0, 1\}$

U-UV codes
- Subchannels convey codewords
- Rate $R_i \in [0, 1]$

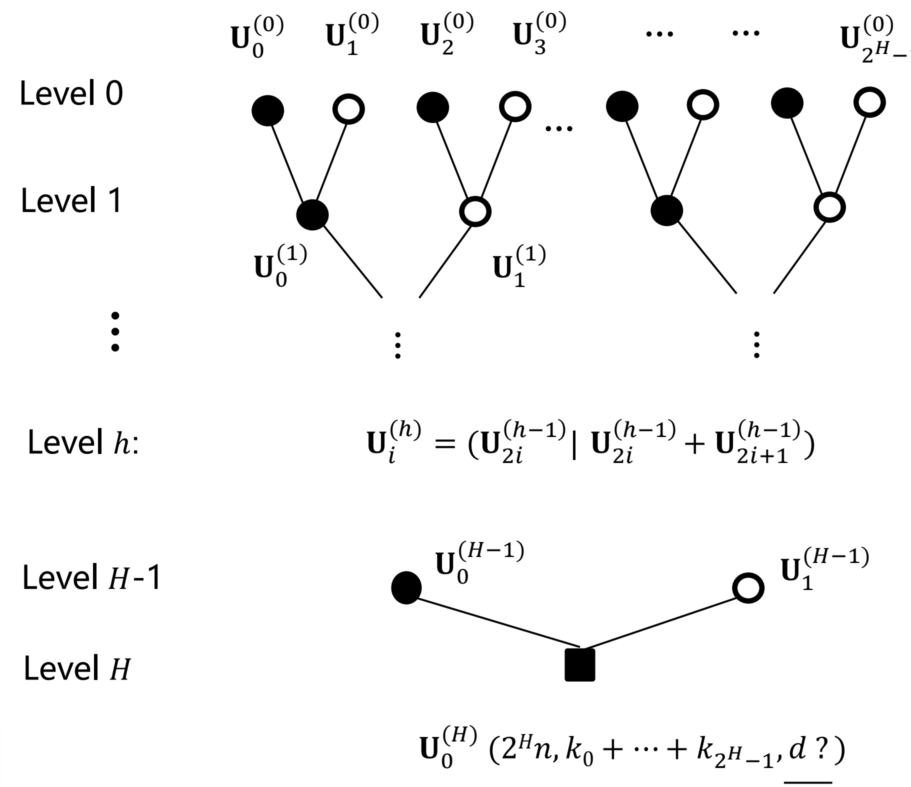
U-UV Codes

- Encoding

Single level construction



Multi-level construction



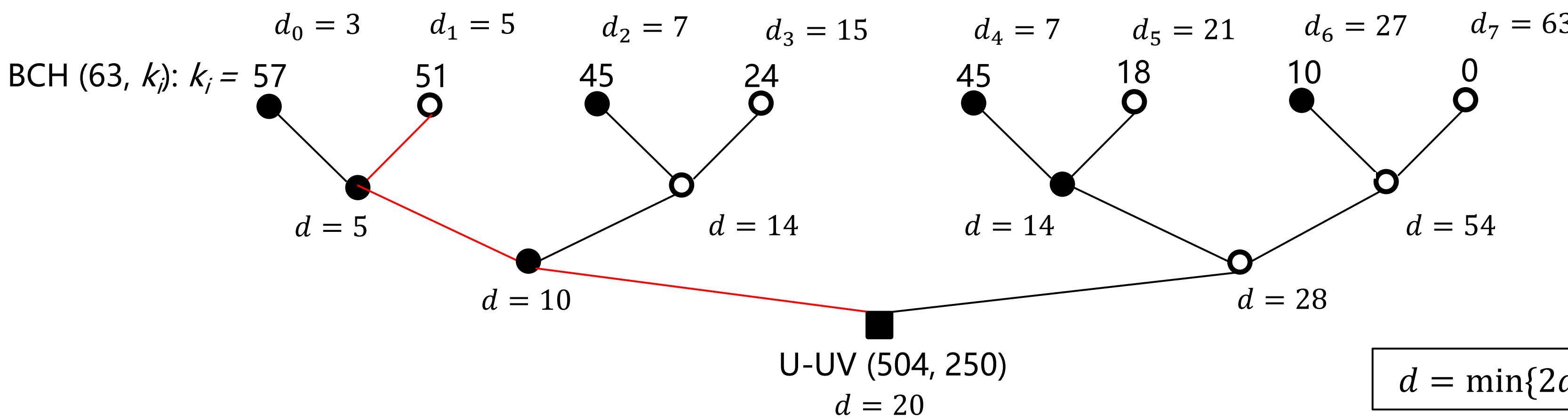
● - U codeword
○ - V codeword
■ - U-UV codeword

U-UV Codes

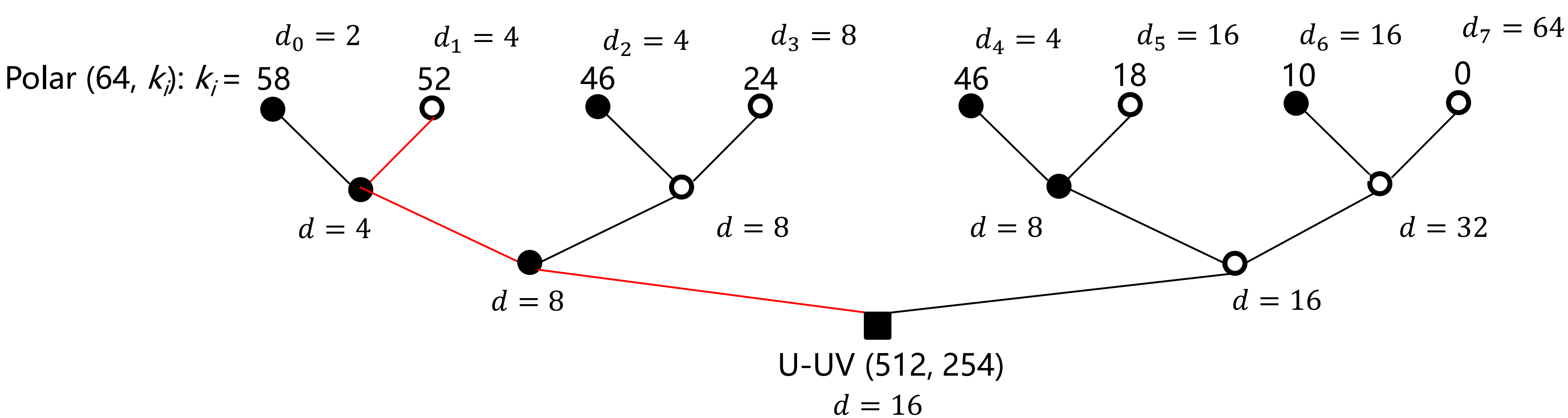
- Encoding

BCH (511, 250): $d = 63$

CRC-polar (512, 254): $d = 16$



$d = \min\{2d_{\mathrm{U}}, d_{\mathrm{V}}\}$

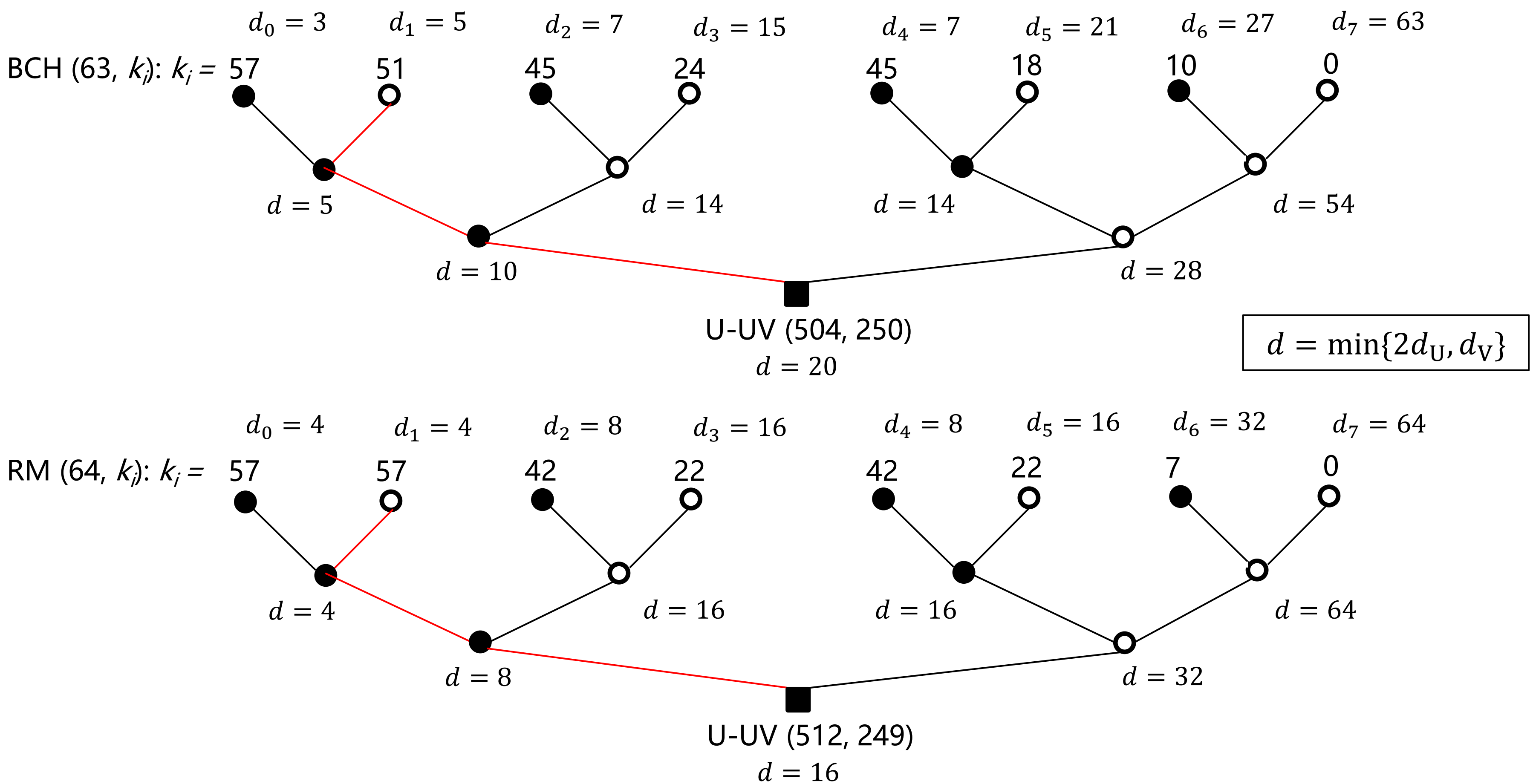


U-UV Codes

- Encoding

BCH (511, 250): $d = 63$

CRC-polar (512, 254): $d = 16$



U-UV Codes

- Encoding

Encoding in linear algebra $\boldsymbol{c}_0^{(H)} = \boldsymbol{m}\,\mathbf{G}_0^{(H)}$

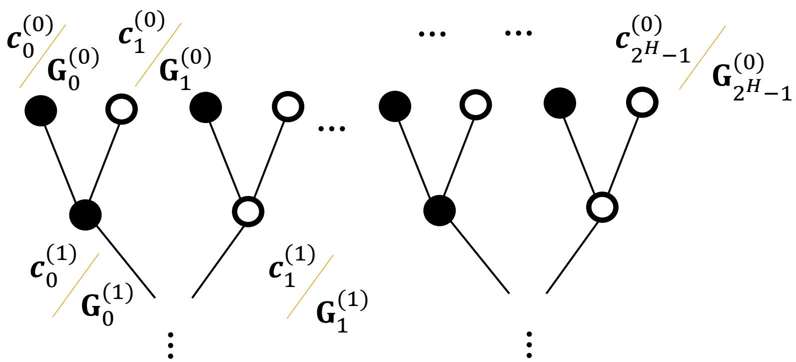
Plotkin kernel $\mathbf{F}' = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$

$\mathbf{U}_i = \{\mathbf{c}_i^{(0)}\}$

Level 0

Level 1

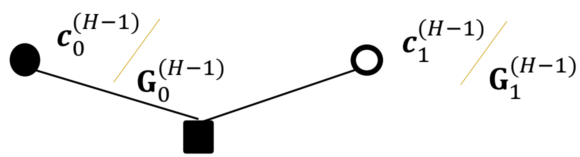
⋮



Level h: $\boldsymbol{c}_i^{(h)} = (\boldsymbol{c}_{2i}^{(h-1)} \mid \boldsymbol{c}_{2i}^{(h-1)} + \boldsymbol{c}_{2i+1}^{(h-1)})$

Level H-1

Level H



$\boldsymbol{c}_0^{(H)}\ (2^H n, k_0 + \cdots + k_{2^H-1}, d)$ $\mathbf{G}_0^{(H)}$

$\mathbf{G}_0^{(0)}$ $\mathbf{G}_1^{(0)}$ $\mathbf{G}_2^{(0)}$ $\mathbf{G}_3^{(0)}$ …

$$\mathbf{G}_0^{(1)} = \begin{bmatrix} \mathbf{G}_0^{(0)} & \mathbf{G}_0^{(0)} \\ \mathbf{0} & \mathbf{G}_1^{(0)} \end{bmatrix} \qquad \mathbf{G}_1^{(1)} = \begin{bmatrix} \mathbf{G}_2^{(0)} & \mathbf{G}_2^{(0)} \\ \mathbf{0} & \mathbf{G}_3^{(0)} \end{bmatrix} \quad \cdots$$

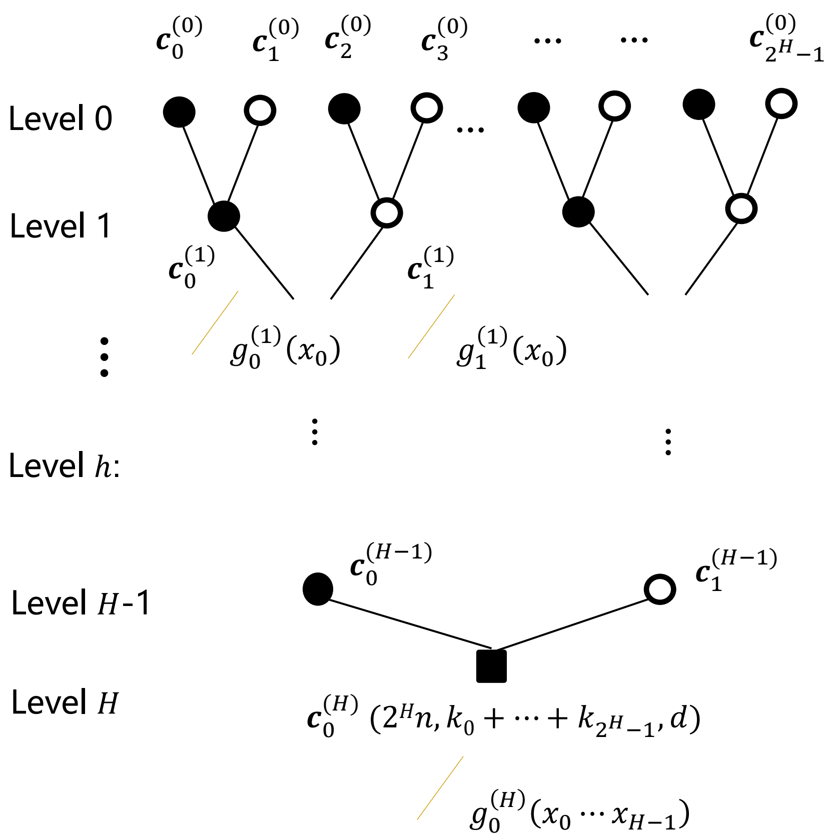
$$\mathbf{G}_0^{(2)} = \begin{bmatrix} \mathbf{G}_0^{(0)} & \mathbf{G}_0^{(0)} & \mathbf{G}_0^{(0)} & \mathbf{G}_0^{(0)} \\ \mathbf{0} & \mathbf{G}_1^{(0)} & \mathbf{0} & \mathbf{G}_1^{(0)} \\ \mathbf{0} & \mathbf{0} & \mathbf{G}_2^{(0)} & \mathbf{G}_2^{(0)} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{G}_3^{(0)} \end{bmatrix} \quad \cdots$$

$$\mathbf{G}_0^{(H)} = \begin{bmatrix} \mathbf{G}_0^{(H-1)} & \mathbf{G}_0^{(H-1)} \\ \mathbf{0} & \mathbf{G}_1^{(H-1)} \end{bmatrix}$$

U-UV Codes

- Encoding

<u>Boolean sequences in encoding</u>



$$g_0^{(1)}(x_0) = \boldsymbol{c}_0^{(0)} + \boldsymbol{c}_1^{(0)} x_0 \qquad g_1^{(1)}(x_0) = \boldsymbol{c}_2^{(0)} + \boldsymbol{c}_3^{(0)} x_0 \quad \cdots$$

$$\boldsymbol{c}_0^{(1)} = \left(\boldsymbol{c}_0^{(0)}, \boldsymbol{c}_0^{(0)} + \boldsymbol{c}_1^{(0)}\right) = \left(g_0^{(1)}(0),\ g_0^{(1)}(1)\right) \qquad \boldsymbol{c}_1^{(1)} = \left(\boldsymbol{c}_2^{(0)}, \boldsymbol{c}_2^{(0)} + \boldsymbol{c}_3^{(0)}\right) = \left(g_1^{(1)}(0),\ g_1^{(1)}(1)\right) \quad \cdots$$

$$g_0^{(2)}(x_0\, x_1) = \boldsymbol{c}_0^{(1)} + \boldsymbol{c}_1^{(1)} x_1 = g_0^{(1)}(x_0) + g_1^{(1)}(x_0) x_1 \quad \cdots$$

$$\boldsymbol{c}_0^{(2)} = \left(\boldsymbol{c}_0^{(0)}, \boldsymbol{c}_0^{(0)} + \boldsymbol{c}_1^{(0)}, \boldsymbol{c}_0^{(0)} + \boldsymbol{c}_2^{(0)}, \boldsymbol{c}_0^{(0)} + \boldsymbol{c}_1^{(0)} + \boldsymbol{c}_2^{(0)} + \boldsymbol{c}_3^{(0)}\right) = \left(g_0^{(2)}(0,0), g_0^{(2)}(1,0), g_0^{(2)}(0,1), g_0^{(2)}(1,1)\right)$$

$$g_0^{(H)}(x_0 \cdots x_{H-1}) = \sum_{i=0}^{2^H-1} c_i x_0^{i_0} \cdots x_{H-1}^{i_{H-1}}$$

$$\boldsymbol{c}_0^{(H)} = \left(g_0^{(H)}(0,0,\ldots,0), g_0^{(H)}(1,0,\ldots,0), \ldots, g_0^{(H)}(1,1,\ldots,1)\right)$$

U-UV Codes

- Decoding

<u>SC decoding: in light of a 2-level U-UV code</u>

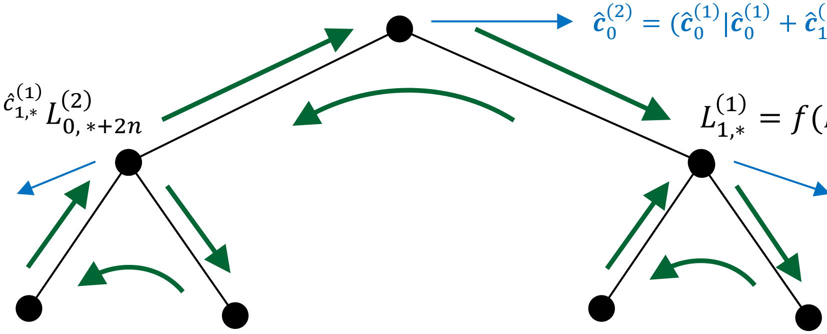
Received sequence: $\boldsymbol{y} = \mathcal{M}\left(\boldsymbol{c}_0^{(2)}\right) + \boldsymbol{w}$; $\mathcal{M}$ - binary modulation, $\boldsymbol{w}$ - AWGN

$$L_{0,*}^{(2)} = \ln \frac{P(y_* |\, c_{0,*}^{(2)} = 0)}{P(y_* |\, c_{0,*}^{(2)} = 1)}$$

$\hat{\boldsymbol{c}}_0^{(2)} = (\hat{\boldsymbol{c}}_0^{(1)} | \hat{\boldsymbol{c}}_0^{(1)} + \hat{\boldsymbol{c}}_1^{(1)})$

$L_{0,*}^{(1)} = L_{0,*}^{(2)} + (-1)^{\hat{c}_{1,*}^{(1)}} L_{0,*+2n}^{(2)}$

$L_{1,*}^{(1)} = f(L_{0,*}^{(2)}, L_{0,*+2n}^{(2)})$



$\hat{\boldsymbol{c}}_0^{(1)} = (\hat{\boldsymbol{c}}_0^{(0)} | \hat{\boldsymbol{c}}_0^{(0)} + \hat{\boldsymbol{c}}_1^{(0)})$

$\hat{\boldsymbol{c}}_1^{(1)} = (\hat{\boldsymbol{c}}_2^{(0)} | \hat{\boldsymbol{c}}_2^{(0)} + \hat{\boldsymbol{c}}_3^{(0)})$

$L_{0,*}^{(0)} = L_{0,*}^{(1)} + (-1)^{\hat{c}_{1,*}^{(0)}} L_{0,*+n}^{(1)}$ $\quad L_{1,*}^{(0)} = f(L_{0,*}^{(1)}, L_{0,*+n}^{(1)})$ $\quad L_{2,*}^{(0)} = L_{1,*}^{(1)} + (-1)^{\hat{c}_{3,*}^{(0)}} L_{1,*+n}^{(1)}$ $\quad L_{3,*}^{(0)} = f(L_{1,*}^{(1)}, L_{1,*+n}^{(1)})$

$\Downarrow$ $\quad \Downarrow$ $\quad \Downarrow$ $\quad \Downarrow$

$\hat{\boldsymbol{c}}_0^{(0)}$ $\quad \hat{\boldsymbol{c}}_1^{(0)}$ $\quad \hat{\boldsymbol{c}}_2^{(0)}$ $\quad \hat{\boldsymbol{c}}_3^{(0)}$

<u>I_{apriori} comes from the Plotkin structure</u> ★

U-UV Codes

- Decoding

<u>SC-list (SCL) decoding: in light of a 2-level U-UV code</u>

Decoding of component codes can provide plural outputs, e.g., ordered statistics decoding (OSD).

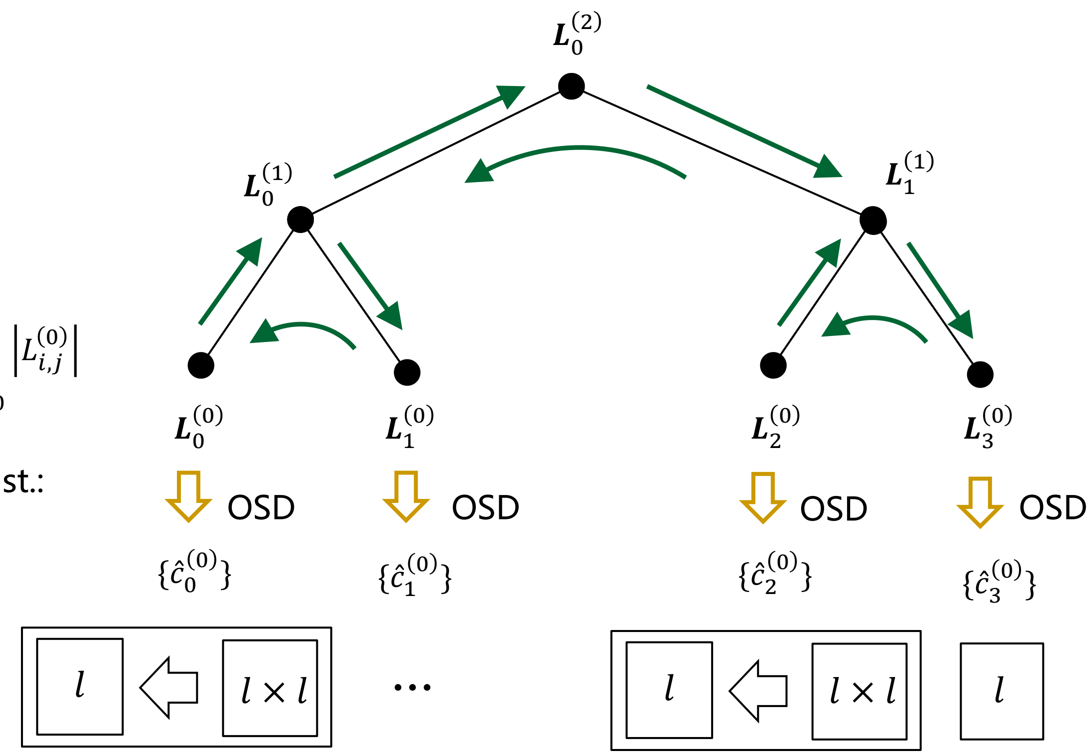
Path metrics

Correlation dist.:

$$\lambda\left(\hat{\boldsymbol{c}}_i^{(0)}, \boldsymbol{L}_i^{(0)}\right) = \sum_{j:\, L_{i,j}^{(0)}\left(1-2\hat{c}_{i,j}^{(0)}\right)<0} \left|L_{i,j}^{(0)}\right|$$

Accumulated correlation dist.:

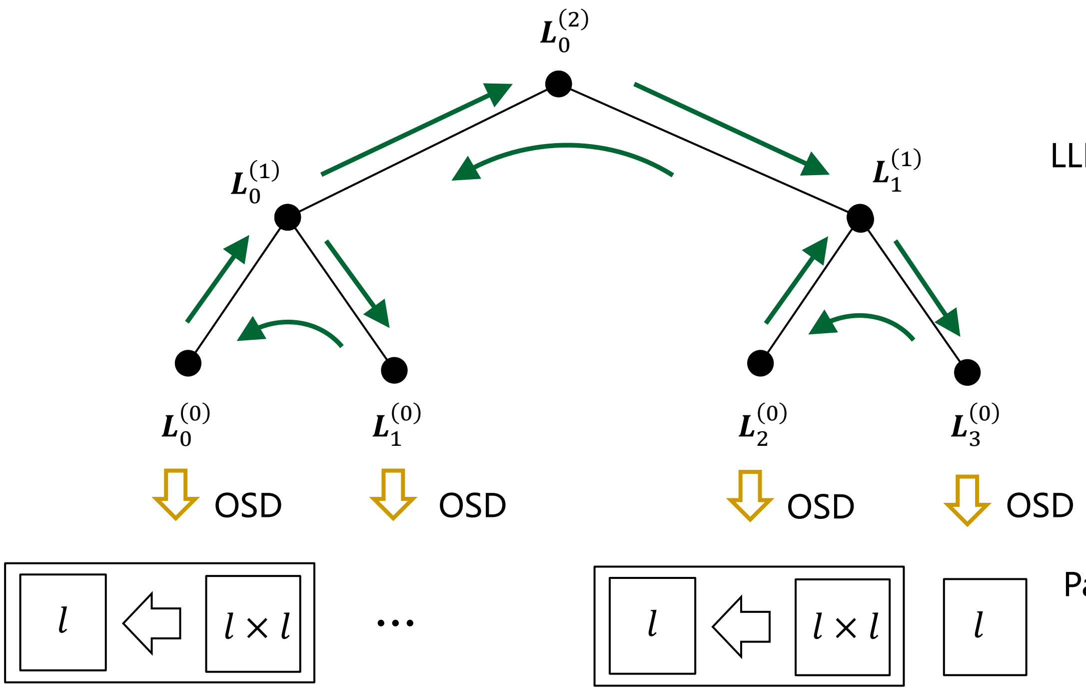
$$D_{i'} = \sum_{i=i'}^{2^H-1} \lambda(\hat{\boldsymbol{c}}_i^{(0)}, \boldsymbol{L}_i^{(0)})$$



U-UV Codes

- Decoding

<u>Decoding complexity & latency</u>



	U-UV, SCL	Polar, SCL
LLR update (FLOP)	$O(lnH2^H)$	$O(ln2^H \log_2(n2^H))$
OSD (BOP)	$O\left(l2^H k_{i^*}^{\tau_{i^*}}\right)$	
Path sorting (FLOP)	$O\left(l2^H(\Gamma + l)\log_2 l\right)$	$O(2kl\log_2 l)$
Latency	$2 \cdot 2^H - 2 + 2^H \cdot T_{\text{OSD}}$	$2 \cdot n2^H - 2$

τ_i: the OSD order for $\mathbf{U}_i^{(0)}$,

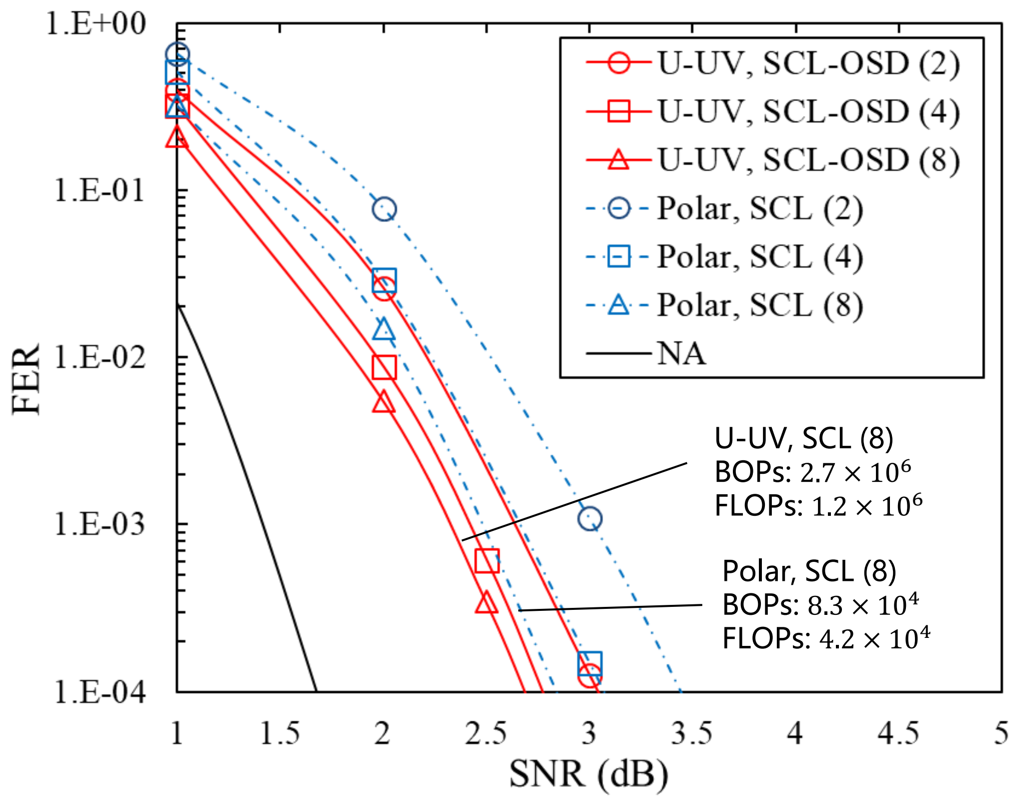
$i^* = \arg\max\{k_i^{\tau_i} | i = 0, 1, \dots, 2^H - 1\}$,

Γ: the maximum value of Γ_i, $\Gamma_i = \sum_{j=0}^{\tau_i} \binom{k_i}{j}$.

U-UV Codes

- Decoding Performance

(504, 250) U-UV code over the AWGN channel using BPSK



(504, 250) U-UV
(63, 57) BCH
(63, 51) BCH
(63, 45) BCH
(63, 24) BCH
(63, 45) BCH
(63, 18) BCH
(63, 10) BCH
(63, 0) BCH

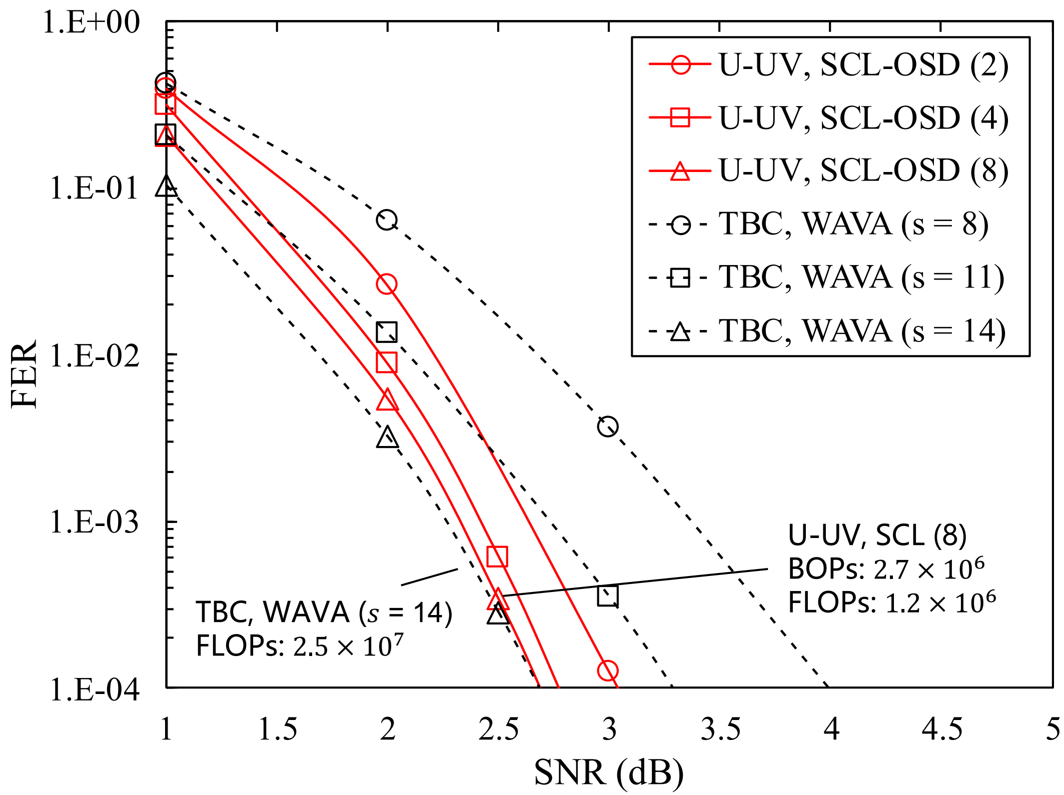
(512, 254) Polar
5G NR, CRC-8

	U-UV	CRC-polar
d	20	16

U-UV Codes

- Decoding Performance

(504, 250) U-UV code over the AWGN channel using BPSK



(504, 250) U-UV
(63, 57) BCH
(63, 51) BCH
(63, 45) BCH
(63, 24) BCH
(63, 45) BCH
(63, 18) BCH
(63, 10) BCH
(63, 0) BCH

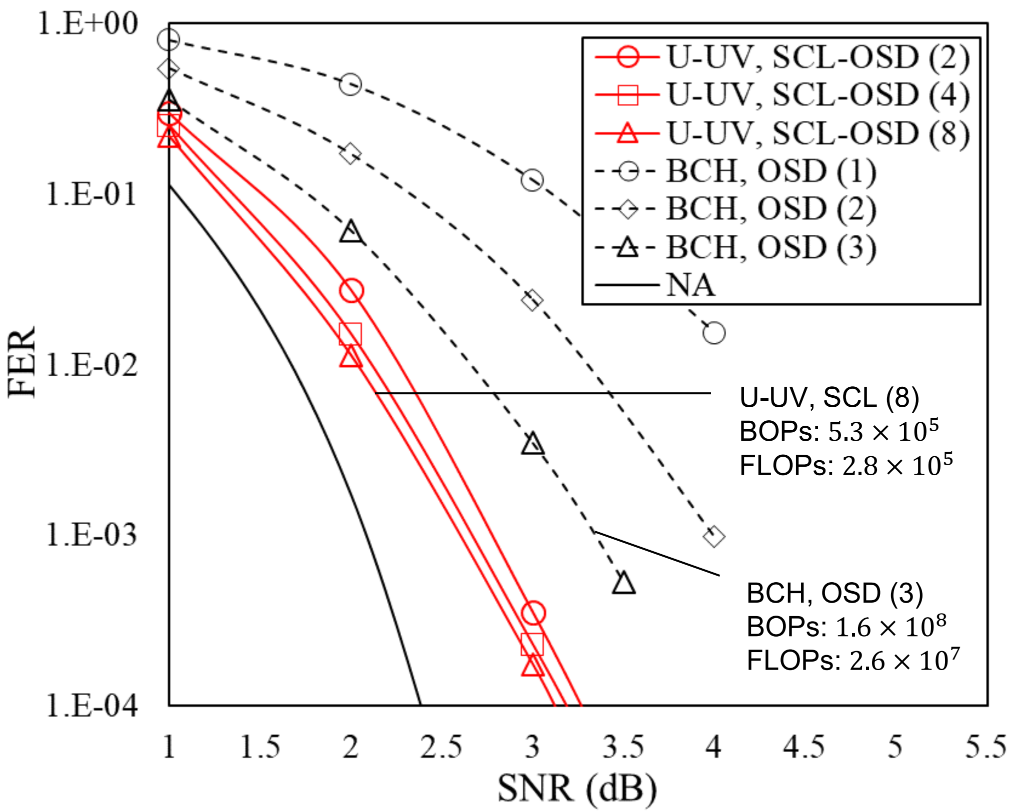
(512, 256) TBC
s = 8 / 11 / 14

TBC code: Tail-Biting Convolutional code
WAVA: Wrap-around Viterbi Algorithm
s : memory order

U-UV Codes

- Decoding Performance

(252, 139) U-UV code over the AWGN channel using BPSK



(252, 139) U-UV
(63, 57) BCH
(63, 39) BCH
(63, 36) BCH
(63, 7) BCH

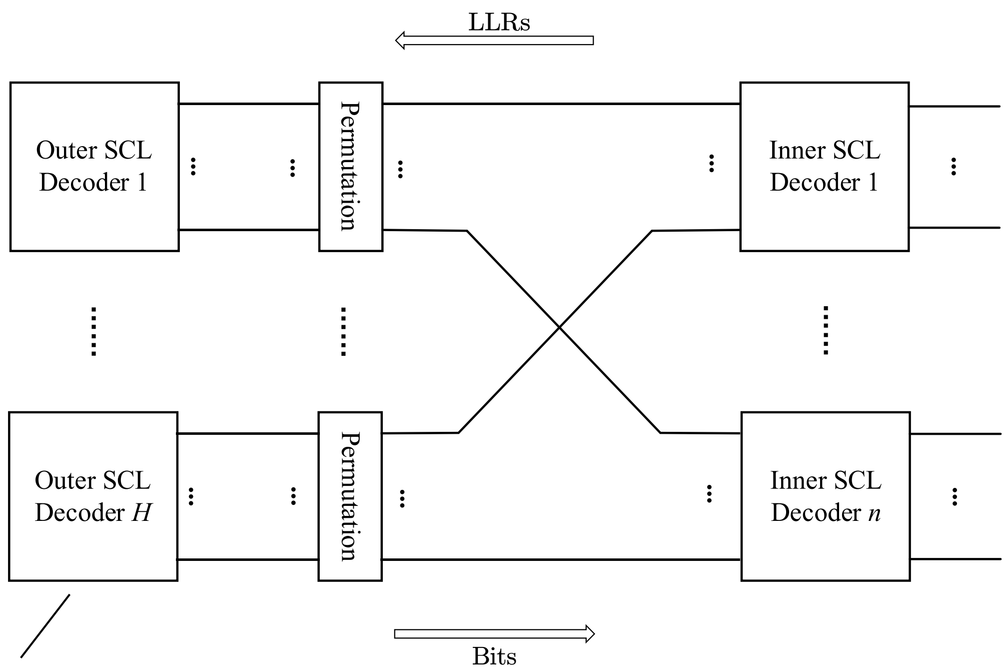
(255, 139) BCH

	U-UV	BCH
d	12	31

U-UV Codes

- Improved Decoding

SCL-SCL (SCL^2) decoding (in the paradigm of generalized concatenated codes)



LLR update (FLOP): $O(\sum_{i=0}^{2^{H-1}} l_i n \log_2 n)$
Outer path sorting: $O(\sum_{i=0}^{2^{H-1}} k_i l_i \log_2 l_i)$
Latency: $2 \cdot 2^H - 2 + 2^H \cdot (2n - 2)$

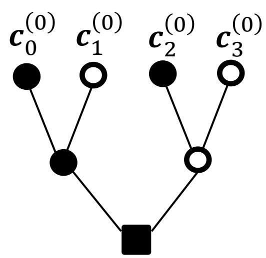
SCL-OSD

OSD (BOP): $O\left(l 2^H k_{i^*}^{\tau_{i^*}}\right)$
Outer path sorting: $O\left(l 2^H \Gamma \log_2 l\right)$
Latency: $2 \cdot 2^H - 2 + 2^H \cdot T_{\mathrm{OSD}}$

U-UV Codes

- Improved Decoding

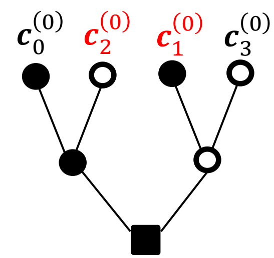
<u>Permutation SC (PSC) decoding</u>



$$g^{(2)}(x_0, x_1) = \boldsymbol{c}_0^{(0)} + \boldsymbol{c}_1^{(0)} x_0 + \boldsymbol{c}_2^{(0)} x_1 + \boldsymbol{c}_3^{(0)} x_0 x_1$$

$\boldsymbol{c}_0^{(0)}$	$\boldsymbol{c}_0^{(0)} + \boldsymbol{c}_1^{(0)}$	$\boldsymbol{c}_0^{(0)} + \boldsymbol{c}_2^{(0)}$	$\boldsymbol{c}_0^{(0)} + \boldsymbol{c}_1^{(0)} + \boldsymbol{c}_2^{(0)} + \boldsymbol{c}_3^{(0)}$

$\Rightarrow$



$$g^{(2)}(x_0, x_1) = \boldsymbol{c}_0^{(0)} + \boldsymbol{c}_2^{(0)} x_0 + \boldsymbol{c}_1^{(0)} x_1 + \boldsymbol{c}_3^{(0)} x_0 x_1$$

$\boldsymbol{c}_0^{(0)}$	$\boldsymbol{c}_0^{(0)} + \boldsymbol{c}_2^{(0)}$	$\boldsymbol{c}_0^{(0)} + \boldsymbol{c}_1^{(0)}$	$\boldsymbol{c}_0^{(0)} + \boldsymbol{c}_1^{(0)} + \boldsymbol{c}_2^{(0)} + \boldsymbol{c}_3^{(0)}$

$$g^{(H)}(x_0, x_1, \ldots, x_{H-1}) = \sum_{i=0}^{2^H-1} \boldsymbol{c}_i^{(0)} x_0^{i_0} \ldots x_{H-1}^{i_{H-1}} \Rightarrow g^{(H)}(x_0, x_1, \ldots, x_{H-1}) = \sum_{i=0}^{2^H-1} \boldsymbol{c}_{\Pi(i)}^{(0)} x_0^{i_0} \ldots x_{H-1}^{i_{H-1}}$$

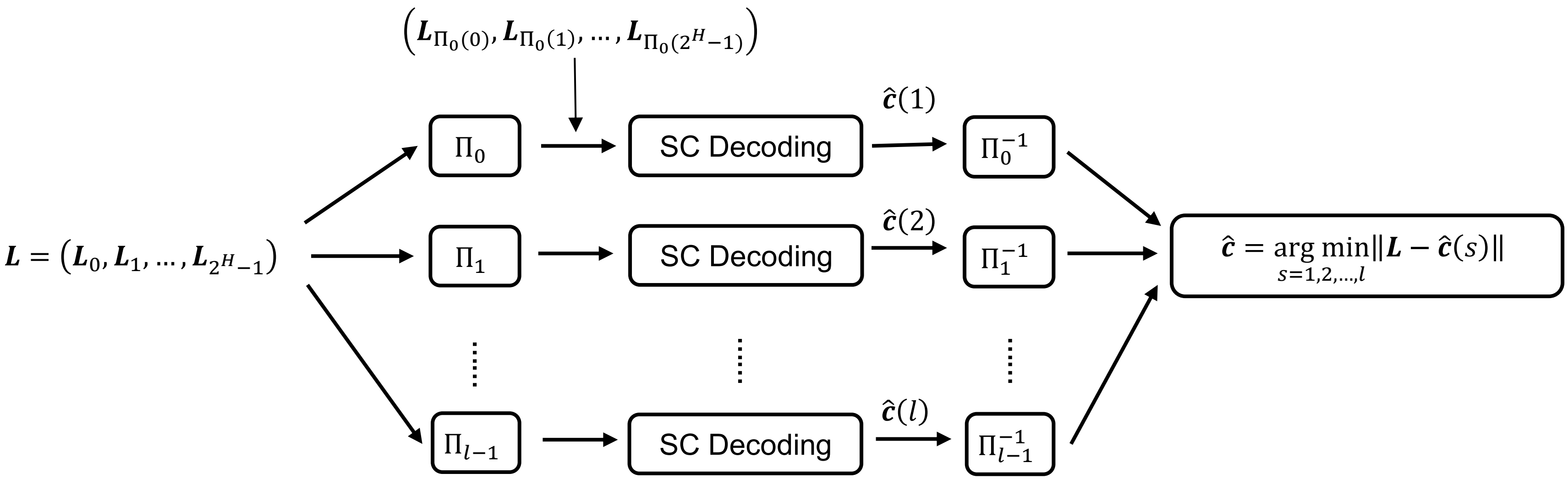
$$\downarrow \qquad\qquad \downarrow$$

$$\boldsymbol{c} = (\boldsymbol{c}_0, \boldsymbol{c}_1, \ldots, \boldsymbol{c}_{2^H-1}) \qquad \boldsymbol{c} = \left(\boldsymbol{c}_{\Pi(0)}, \boldsymbol{c}_{\Pi(1)}, \ldots, \boldsymbol{c}_{\Pi(2^H-1)}\right)$$

U-UV Codes

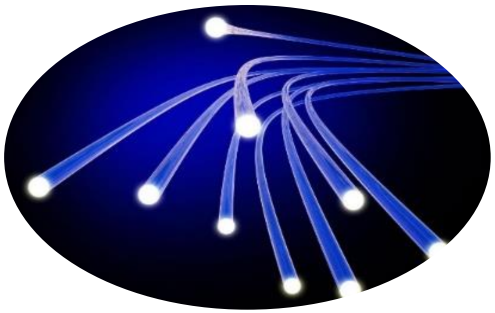
- Improved Decoding

<u>Permutation SC (PSC) decoding</u>

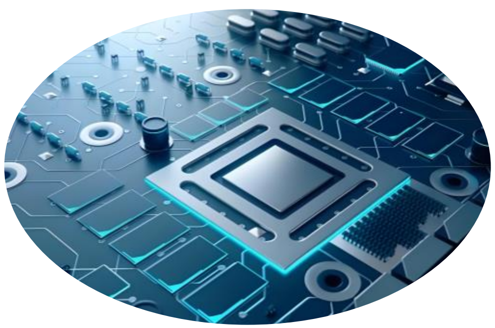


Advantage over SCL decoding: no path management

GII Codes


optical communications


data storage


chip communications

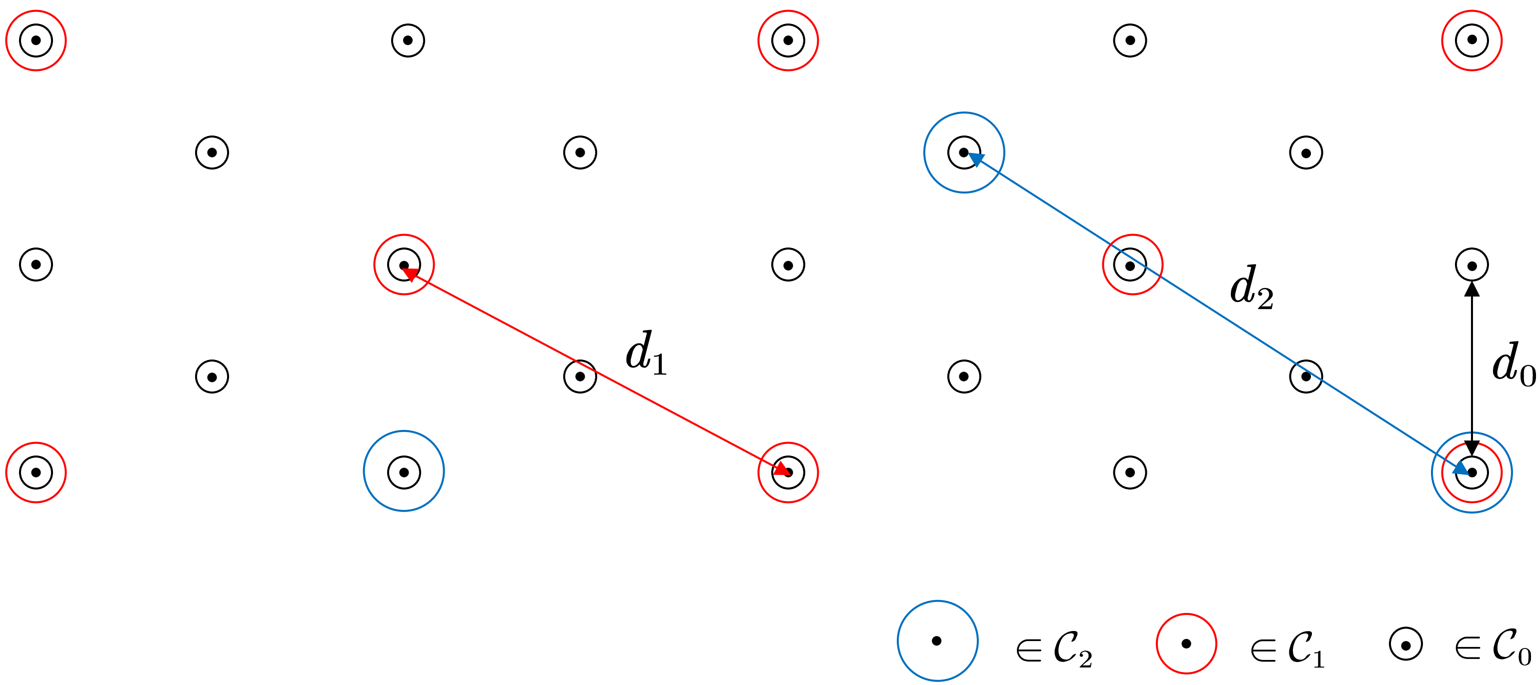
- Structure: Nested codebooks
- Component codes: RS / BCH codes

GII codes: Generalized Integrated Interleaved codes

GII Codes

- Construction Initiatives

Codebook ($\mathcal{C}_0$) and its nested ones ($\mathcal{C}_1$, $\mathcal{C}_2$) $\mathcal{C}_2 \subseteq \mathcal{C}_1 \subseteq \mathcal{C}_0$



Minimum distances: $d_2 \geq d_1 \geq d_0$

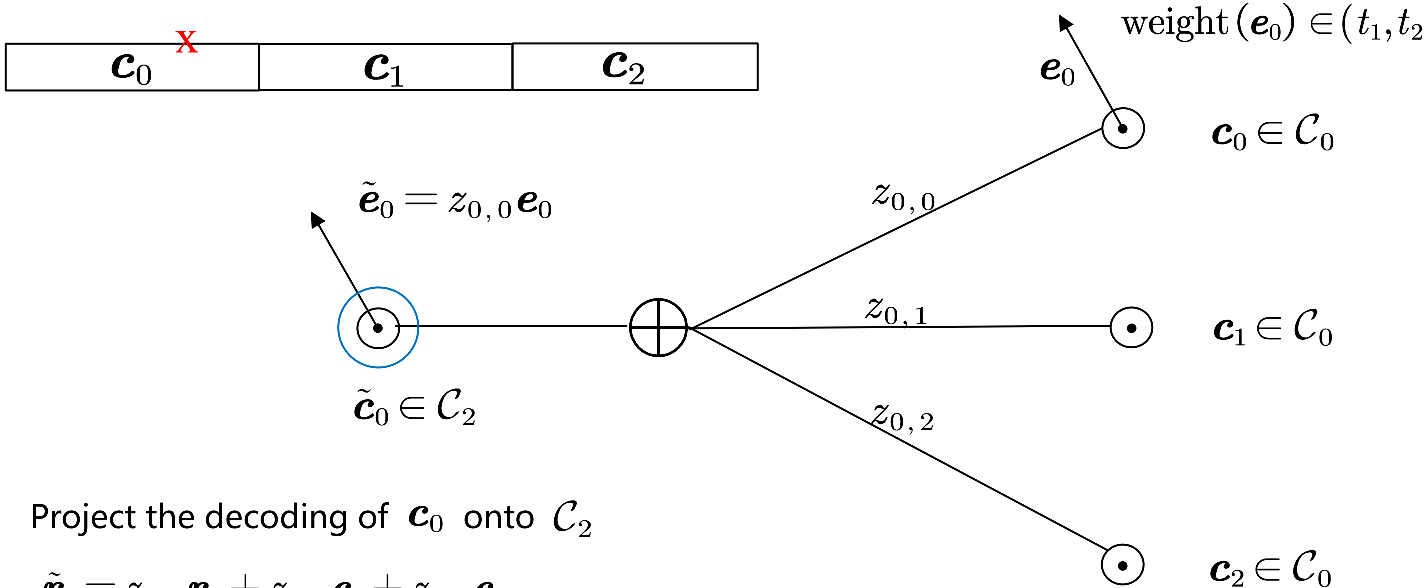
Error-correction capabilities: $t_2 \geq t_1 \geq t_0$

GII Codes

- Construction Initiatives

Codebook ($\mathcal{C}_0$) and its nested ones ($\mathcal{C}_1,\ \mathcal{C}_2$) $\quad \mathcal{C}_2 \subseteq \mathcal{C}_1 \subseteq \mathcal{C}_0$

$$z_{0,0}\boldsymbol{c}_0 + z_{0,1}\boldsymbol{c}_1 + z_{0,2}\boldsymbol{c}_2 = \tilde{\boldsymbol{c}}_0 \in \mathcal{C}_2$$



Project the decoding of $\boldsymbol{c}_0$ onto $\mathcal{C}_2$

$$\begin{aligned}\tilde{\boldsymbol{r}}_0 &= z_{0,0}\boldsymbol{r}_0 + z_{0,1}\boldsymbol{c}_1 + z_{0,2}\boldsymbol{c}_2 \\ &= z_{0,0}(\boldsymbol{c}_0 + \boldsymbol{e}_0) + z_{0,1}\boldsymbol{c}_1 + z_{0,2}\boldsymbol{c}_2 \\ &= \tilde{\boldsymbol{c}}_0 + \underbrace{z_{0,0}\boldsymbol{e}_0}_{\tilde{\boldsymbol{e}}_0}\end{aligned}$$

$$\boxed{z_{0,0},\ z_{0,1},\ z_{0,1} \in \mathbb{F}_q}$$

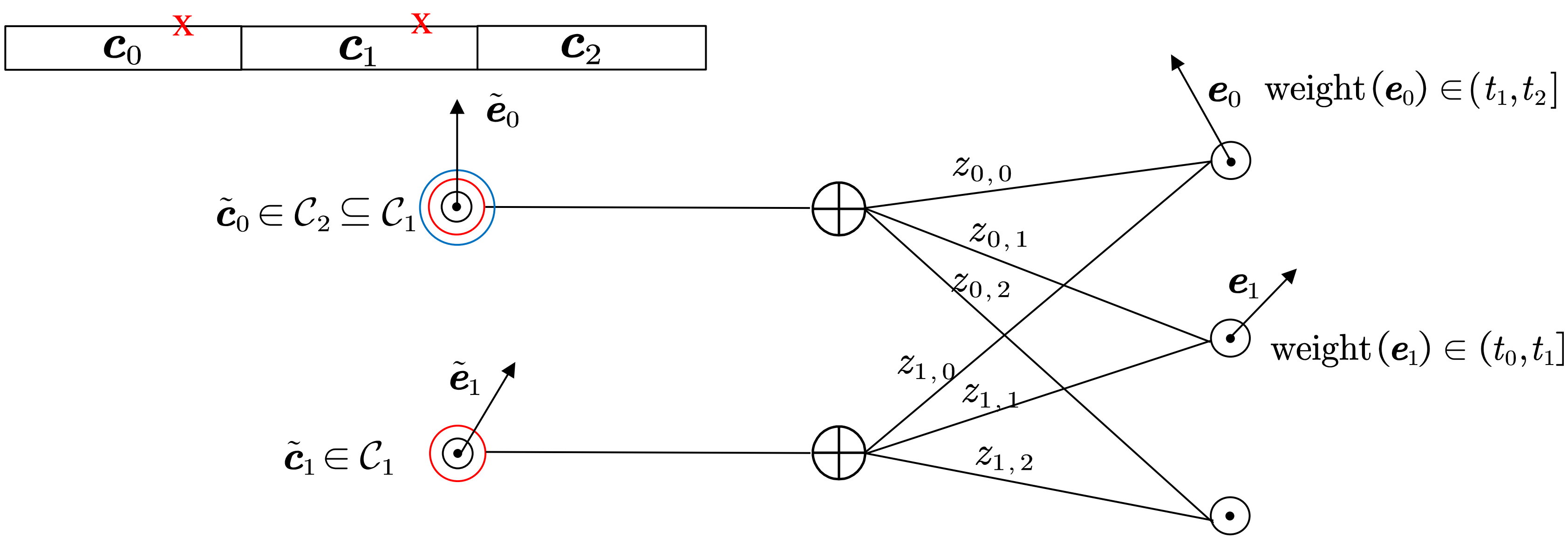
GII Codes

- Construction Initiatives

Codebook ($\mathcal{C}_0$) and its nested ones ($\mathcal{C}_1, \mathcal{C}_2$) $\quad \mathcal{C}_2 \subseteq \mathcal{C}_1 \subseteq \mathcal{C}_0$

$$z_{0,0}\boldsymbol{c}_0 + z_{0,1}\boldsymbol{c}_1 + z_{0,2}\boldsymbol{c}_2 = \tilde{\boldsymbol{c}}_0 \in \mathcal{C}_2$$

$$z_{1,0}\boldsymbol{c}_0 + z_{1,1}\boldsymbol{c}_1 + z_{1,2}\boldsymbol{c}_2 = \tilde{\boldsymbol{c}}_1 \in \mathcal{C}_1$$



Project the decoding of $\boldsymbol{c}_0$ and $\boldsymbol{c}_1$ onto $\mathcal{C}_2$ and $\mathcal{C}_1$

$$\tilde{\boldsymbol{r}}_0 = \tilde{\boldsymbol{c}}_0 + \underbrace{z_{0,0}\boldsymbol{e}_0 + z_{0,1}\boldsymbol{e}_1}_{\tilde{\boldsymbol{e}}_0}$$

$$\tilde{\boldsymbol{r}}_1 = \tilde{\boldsymbol{c}}_1 + \underbrace{z_{1,0}\boldsymbol{e}_0 + z_{1,1}\boldsymbol{e}_1}_{\tilde{\boldsymbol{e}}_1}$$

$$\begin{bmatrix} \tilde{\boldsymbol{e}}_0 \\ \tilde{\boldsymbol{e}}_1 \end{bmatrix} = \begin{bmatrix} z_{0,0} & z_{0,1} \\ z_{1,0} & z_{1,1} \end{bmatrix} \begin{bmatrix} \boldsymbol{e}_0 \\ \boldsymbol{e}_1 \end{bmatrix}$$

$$\boxed{z_{0,0}, z_{0,1}, z_{0,2}, z_{1,0}, z_{1,1}, z_{1,2} \in \mathbb{F}_q}$$

GII Codes

- Construction Initiatives

Codebook ($\mathcal{C}_0$) and its nested ones ($\mathcal{C}_1$, $\mathcal{C}_2$) $\mathcal{C}_2 \subseteq \mathcal{C}_1 \subseteq \mathcal{C}_0$

$\mathcal{C}_0$: $(15, 11)$ BCH code $d_0 = 3$

$\mathcal{C}_1$: $(15, 7)$ BCH code $d_1 = 5$

$\mathcal{C}_2$: $(15, 5)$ BCH code $d_2 = 7$

In light of (15, 5) – (15, 7) – (15, 11) BCH codes

	$\boldsymbol{e}_0$	$\boldsymbol{e}_1$	$\boldsymbol{e}_2$
weight($\boldsymbol{e}_i$)	3	2	1

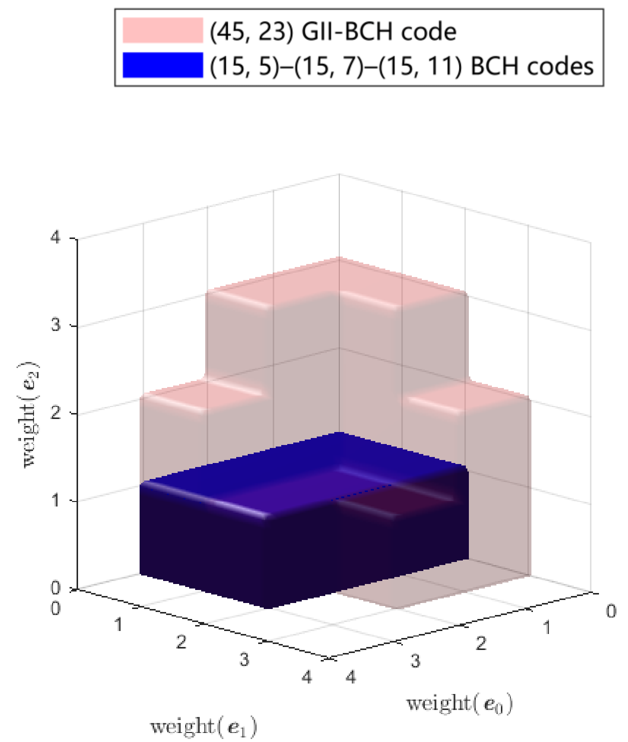
$$d = \min\{d_0, d_1, d_2\} = 3$$

(45, 23) GII-BCH code $(d = 7)$

	$\boldsymbol{e}_0$	$\boldsymbol{e}_1$	$\boldsymbol{e}_2$
weight($\boldsymbol{e}_i$)	3	2	1
weight($\boldsymbol{e}_i$)	1	2	3
weight($\boldsymbol{e}_i$)	2	3	1

or ……

$$d = \min\{3d_0, 2d_1, d_2\} = 7$$



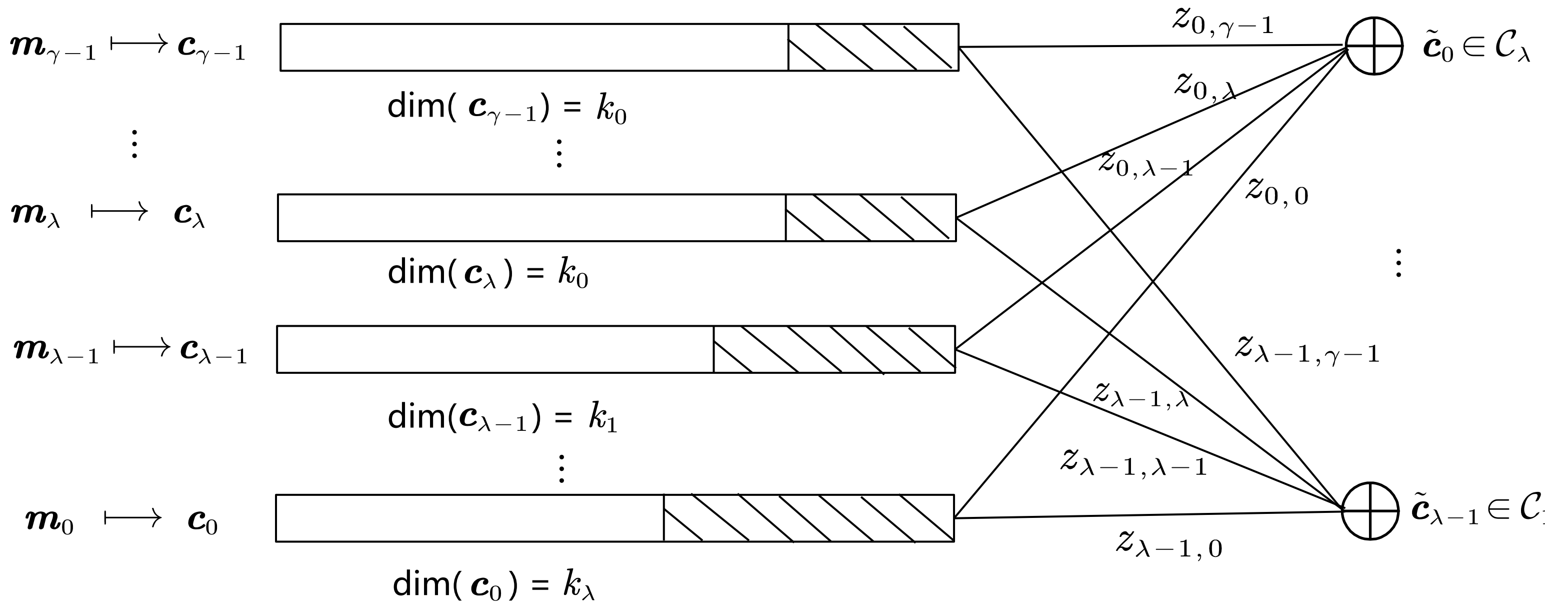
GII Codes

- Encoding

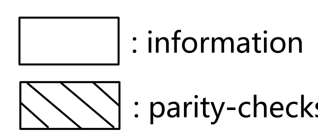
$\mathcal{C}_\lambda \subseteq \mathcal{C}_{\lambda-1} \subseteq \cdots \subseteq \mathcal{C}_1 \subseteq \mathcal{C}_0$

γ component codes $\boldsymbol{c}_0, \cdots, \boldsymbol{c}_{\gamma-1} \in \mathcal{C}_0$

λ nested codes : $\tilde{\boldsymbol{c}}_0, \cdots, \tilde{\boldsymbol{c}}_{\lambda-1}$



$$\begin{bmatrix} z_{0,0} & z_{0,1} & \cdots & z_{0,\gamma-1} \\ z_{1,0} & z_{1,1} & \cdots & z_{1,\gamma-1} \\ \vdots & \vdots & \ddots & \vdots \\ z_{\lambda-1,0} & z_{\lambda-1,1} & \cdots & z_{\lambda-1,\gamma-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{c}_0 \\ \boldsymbol{c}_1 \\ \vdots \\ \boldsymbol{c}_{\gamma-1} \end{bmatrix} = \begin{bmatrix} \tilde{\boldsymbol{c}}_0 \\ \tilde{\boldsymbol{c}}_1 \\ \vdots \\ \tilde{\boldsymbol{c}}_{\lambda-1} \end{bmatrix}$$



GII Codes

- Encoding

$$
\begin{array}{lccccccccc}
\text{Codebooks} & \mathcal{C}_\lambda & \subseteq & \mathcal{C}_{\lambda-1} & \subseteq & \cdots & \subseteq & \mathcal{C}_1 & \subseteq & \mathcal{C}_0 \\
 & \uparrow & & \uparrow & & & & \uparrow & & \uparrow \\
\text{Generator matrices} & \mathbf{G}_\lambda & & \mathbf{G}_{\lambda-1} & & & & \mathbf{G}_1 & & \mathbf{G}_0 \\
 & & & & & & & & & \\
\text{Parity-check matrices} & \mathbf{H}_\lambda & & \mathbf{H}_{\lambda-1} & & & & \mathbf{H}_1 & & \mathbf{H}_0 \\
 & \downarrow & & \downarrow & & & & \downarrow & & \downarrow \\
\text{Dual codebooks} & \mathcal{C}_\lambda{}^\perp & \supseteq & \mathcal{C}_{\lambda-1}{}^\perp & \supseteq & \cdots & \supseteq & \mathcal{C}_1{}^\perp & \supseteq & \mathcal{C}_0{}^\perp
\end{array}
$$

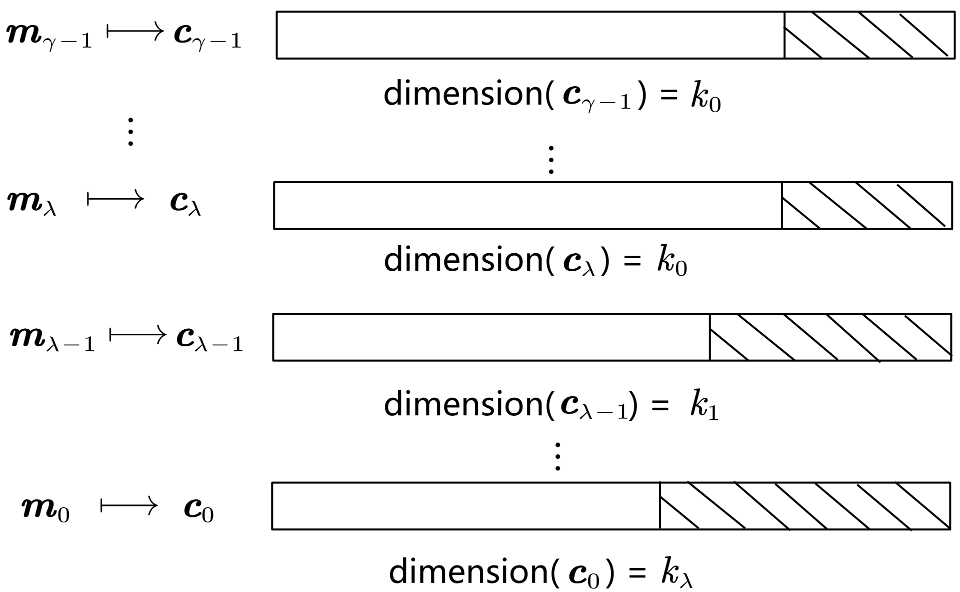
$$
\mathbf{G}_0 = \begin{bmatrix} \boldsymbol{g}_0 \\ \boldsymbol{g}_1 \\ \vdots \\ \boldsymbol{g}_{k_0-1} \end{bmatrix} = \begin{bmatrix} \mathbf{G}_1 \\ \underbrace{\begin{matrix} \boldsymbol{g}_{k_1} \\ \vdots \\ \boldsymbol{g}_{k_0-1} \end{matrix}}_{\mathbf{G}_{0\setminus 1}} \end{bmatrix} = \cdots = \begin{bmatrix} \mathbf{G}_\lambda \\ \underbrace{\begin{matrix} \boldsymbol{g}_{k_\lambda} \\ \vdots \\ \boldsymbol{g}_{k_0-1} \end{matrix}}_{\mathbf{G}_{0\setminus \lambda}} \end{bmatrix}
\qquad
\mathbf{H}_\lambda = \begin{bmatrix} \boldsymbol{h}_0 \\ \boldsymbol{h}_1 \\ \vdots \\ \boldsymbol{h}_{n-k_\lambda-1} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_{\lambda-1} \\ \underbrace{\begin{matrix} \boldsymbol{h}_{n-k_{\lambda-1}} \\ \vdots \\ \boldsymbol{h}_{n-k_\lambda-1} \end{matrix}}_{\mathbf{H}_{\lambda\setminus\lambda-1}} \end{bmatrix} = \cdots = \begin{bmatrix} \mathbf{H}_0 \\ \underbrace{\begin{matrix} \boldsymbol{h}_{n-k_0} \\ \vdots \\ \boldsymbol{h}_{n-k_\lambda-1} \end{matrix}}_{\mathbf{H}_{\lambda\setminus 0}} \end{bmatrix}
$$

GII Codes

- Encoding

$$\mathcal{C}_{\lambda} \subseteq \mathcal{C}_{\lambda-1} \subseteq \cdots \subseteq \mathcal{C}_1 \subseteq \mathcal{C}_0$$

γ component codes $\boldsymbol{c}_0, \cdots, \boldsymbol{c}_{\gamma-1} \in \mathcal{C}_0$



$$\boldsymbol{c}_{\gamma-1} = \boldsymbol{u}_{\gamma-1}\mathbf{G}_0$$

$$\vdots$$

$$\boldsymbol{c}_{\lambda} = \boldsymbol{u}_{\lambda}\mathbf{G}_0$$

$\boldsymbol{c}_{\lambda-1}$ // $\mathbf{H}_1$

$\boldsymbol{c}_0$ // $\mathbf{H}_{\lambda}$

GII Codes

- Encoding

$$\mathcal{C}_\lambda \subseteq \cdots \subseteq \mathcal{C}_2 \subseteq \boxed{\mathcal{C}_1} \subseteq \mathcal{C}_0 \qquad\qquad \mathbf{H}_1$$

Generate $\boldsymbol{c}_{\lambda-1}$

$$\begin{bmatrix} z_{0,0} & z_{0,1} & \cdots & z_{0,\gamma-1} \\ z_{1,0} & z_{1,1} & \cdots & z_{1,\gamma-1} \\ \vdots & \vdots & \ddots & \vdots \\ z_{\lambda-1,0} & z_{\lambda-1,1} & \cdots & z_{\lambda-1,\gamma-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{c}_0 \\ \boldsymbol{c}_1 \\ \vdots \\ \boldsymbol{c}_{\gamma-1} \end{bmatrix} = \left. \begin{bmatrix} \tilde{\boldsymbol{c}}_0 \\ \tilde{\boldsymbol{c}}_1 \\ \vdots \\ \tilde{\boldsymbol{c}}_{\lambda-1} \end{bmatrix} \right\} \in \mathcal{C}_1$$

$$\begin{bmatrix} z_{0,0} & z_{0,1} & \cdots & z_{0,\gamma-1} \\ z_{1,0} & z_{1,1} & \cdots & z_{1,\gamma-1} \\ \vdots & \vdots & \ddots & \vdots \\ z_{\lambda-1,0} & z_{\lambda-1,1} & \cdots & z_{\lambda-1,\gamma-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{c}_0 \mathbf{H}_1^T \\ \boldsymbol{c}_1 \mathbf{H}_1^T \\ \vdots \\ \boldsymbol{c}_{\gamma-1} \mathbf{H}_1^T \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}$$

$$\begin{bmatrix} \boldsymbol{c}_0 \mathbf{H}_1^T \\ \boldsymbol{c}_1 \mathbf{H}_1^T \\ \vdots \\ \boldsymbol{c}_{\lambda-1} \mathbf{H}_1^T \end{bmatrix} = - \begin{bmatrix} z_{0,0} & z_{0,1} & \cdots & z_{0,\lambda-1} \\ z_{1,0} & z_{1,1} & \cdots & z_{1,\lambda-1} \\ \vdots & \vdots & \ddots & \vdots \\ z_{\lambda-1,0} & z_{\lambda-1,1} & \cdots & z_{\lambda-1,\lambda-1} \end{bmatrix}^{-1} \begin{bmatrix} z_{0,\lambda} & z_{0,\lambda+1} & \cdots & z_{0,\gamma-1} \\ z_{1,\lambda} & z_{1,\lambda+1} & \cdots & z_{1,\gamma-1} \\ \vdots & \vdots & \ddots & \vdots \\ z_{\lambda-1,\lambda} & z_{\lambda-1,\lambda+1} & \cdots & z_{\lambda-1,\gamma-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{c}_\lambda \mathbf{H}_1^T \\ \boldsymbol{c}_{\lambda+1} \mathbf{H}_1^T \\ \vdots \\ \boldsymbol{c}_{\gamma-1} \mathbf{H}_1^T \end{bmatrix}$$

$$\pi^{(\lambda-1)}$$

$$\boldsymbol{c}_{\lambda-1}\mathbf{H}_1^T = \pi^{(\lambda-1)} [\boldsymbol{c}_\lambda \mathbf{H}_1^T, \boldsymbol{c}_{\lambda+1}\mathbf{H}_1^T, \cdots, \boldsymbol{c}_{\gamma-1}\mathbf{H}_1^T]^T$$

GII Codes

- Encoding

$$\boxed{\mathcal{C}_\lambda} \subseteq \cdots \subseteq \boxed{\mathcal{C}_2} \subseteq \mathcal{C}_1 \subseteq \mathcal{C}_0 \qquad \mathbf{H}_2 \quad \cdots \quad \mathbf{H}_\lambda$$

Generate $\boldsymbol{c}_{\lambda-2}$

$$\begin{bmatrix} z_{0,0} & z_{0,1} & \cdots & z_{0,\gamma-1} \\ z_{1,0} & z_{1,1} & \cdots & z_{1,\gamma-1} \\ \vdots & \vdots & \ddots & \vdots \\ \cancel{z_{\lambda-1,0}} & \cancel{z_{\lambda-1,1}} & \cancel{\cdots} & \cancel{z_{\lambda-1,\gamma-1}} \end{bmatrix} \begin{bmatrix} \boldsymbol{c}_0 \\ \boldsymbol{c}_1 \\ \vdots \\ \boldsymbol{c}_{\gamma-1} \end{bmatrix} = \left.\begin{bmatrix} \tilde{\boldsymbol{c}}_0 \\ \tilde{\boldsymbol{c}}_1 \\ \vdots \\ \cancel{\tilde{\boldsymbol{c}}_{\lambda-1}} \end{bmatrix}\right\} \in \mathcal{C}_2 \quad \begin{bmatrix} z_{0,0} & z_{0,1} & \cdots & z_{0,\gamma-1} \\ z_{1,0} & z_{1,1} & \cdots & z_{1,\gamma-1} \\ \vdots & \vdots & \ddots & \vdots \\ z_{\lambda-2,0} & z_{\lambda-2,1} & \cdots & z_{\lambda-2,\gamma-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{c}_0\mathbf{H}_2^T \\ \boldsymbol{c}_1\mathbf{H}_2^T \\ \vdots \\ \boldsymbol{c}_{\gamma-1}\mathbf{H}_2^T \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}$$

$$\begin{bmatrix} \boldsymbol{c}_0\mathbf{H}_2^T \\ \boldsymbol{c}_1\mathbf{H}_2^T \\ \vdots \\ \boldsymbol{c}_{\lambda-2}\mathbf{H}_2^T \end{bmatrix} = -\begin{bmatrix} z_{0,0} & z_{0,1} & \cdots & z_{0,\lambda-2} \\ z_{1,0} & z_{1,1} & \cdots & z_{1,\lambda-2} \\ \vdots & \vdots & \ddots & \vdots \\ z_{\lambda-2,0} & z_{\lambda-2,1} & \cdots & z_{\lambda-2,\lambda-2} \end{bmatrix}^{-1} \begin{bmatrix} z_{0,\lambda-1} & z_{0,\lambda} & \cdots & z_{0,\gamma-1} \\ z_{1,\lambda-1} & z_{1,\lambda} & \cdots & z_{1,\gamma-1} \\ \vdots & \vdots & \ddots & \vdots \\ z_{\lambda-2,\lambda-1} & z_{\lambda-2,\lambda} & \cdots & z_{\lambda-2,\gamma-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{c}_{\lambda-1}\mathbf{H}_2^T \\ \boldsymbol{c}_\lambda\mathbf{H}_2^T \\ \vdots \\ \boldsymbol{c}_{\gamma-1}\mathbf{H}_2^T \end{bmatrix}$$

$$\pi^{(\lambda-2)}$$

$$\boldsymbol{c}_{\lambda-2}\mathbf{H}_2^T = \pi^{(\lambda-2)}[\boldsymbol{c}_{\lambda-1}\mathbf{H}_2^T, \boldsymbol{c}_\lambda\mathbf{H}_2^T, \cdots, \boldsymbol{c}_{\gamma-1}\mathbf{H}_2^T]^T$$

Generate $\boldsymbol{c}_0$

$$\boldsymbol{c}_0\mathbf{H}_\lambda^T = \pi^{(0)}[\boldsymbol{c}_1\mathbf{H}_\lambda^T, \boldsymbol{c}_2\mathbf{H}_\lambda^T, \cdots, \boldsymbol{c}_{\gamma-1}\mathbf{H}_\lambda^T]^T$$

GII Codes

- Encoding

$$\mathcal{C}_\lambda \subseteq \cdots \subseteq \mathcal{C}_2 \subseteq \mathcal{C}_1 \subseteq \mathcal{C}_0$$

$$\boldsymbol{c}_{\lambda-1}$$

$$\vdots$$

$$\boldsymbol{c}_{\lambda-i+1}$$

$$\boldsymbol{c}_{\lambda-i}\mathbf{H}_i^T = \pi^{(\lambda-i)}[\boldsymbol{c}_{\lambda-i+1}\mathbf{H}_i^T, \boldsymbol{c}_{\lambda-i+2}\mathbf{H}_i^T, \cdots, \boldsymbol{c}_{\gamma-1}\mathbf{H}_i^T]^T, \ 1 \leq i \leq \lambda$$

<u>Non-systematic encoding</u>

$$\boldsymbol{c}_{\lambda-i} = [\boldsymbol{m}_{\lambda-i}, \underline{\boldsymbol{m}'_{\lambda-i}}]\mathbf{G}_0$$

<u>Systematic encoding</u>

$$\boldsymbol{c}_{\lambda-i} = [\boldsymbol{m}_{\lambda-i}, \ \underline{\boldsymbol{\zeta}_{\lambda-i}}]$$

dim($\boldsymbol{m}_{\lambda-i}$) = $k_i < k_0$

$$\mathbf{G}_i = [\mathbf{I}_{k_i} \ \mathbf{P}_i]$$

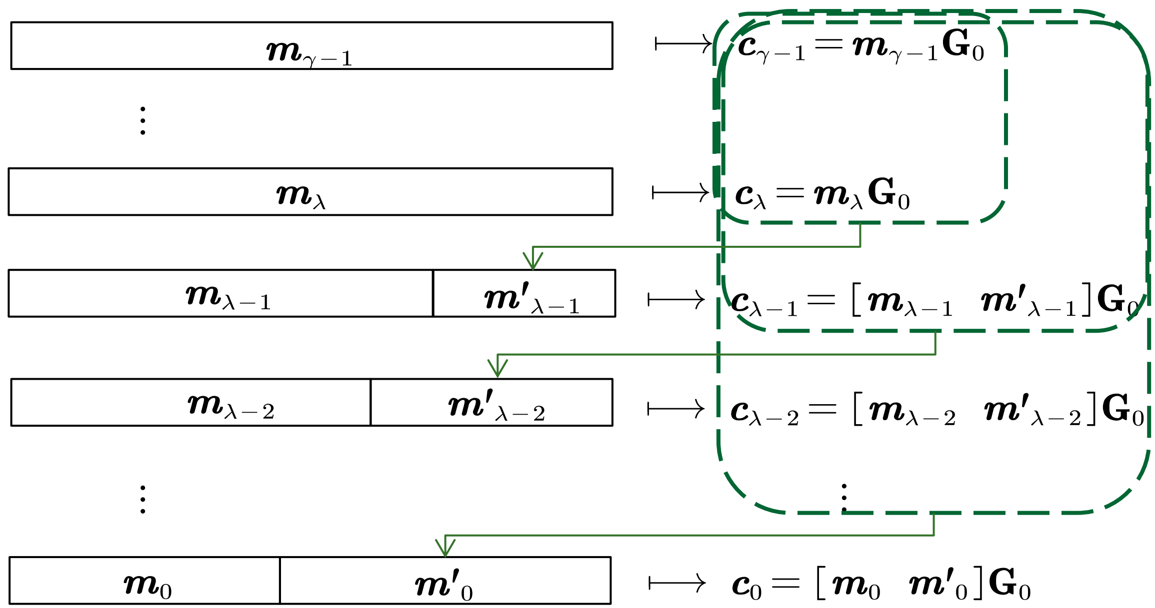
$$\mathbf{H}_i^T = \begin{bmatrix} \mathbf{P}_i \\ \mathbf{I}_{n-k_i} \end{bmatrix}$$

GII Codes

- Encoding

<u>Non-systematic encoding</u> $\mathcal{C}_\lambda \subseteq \cdots \subseteq \mathcal{C}_2 \subseteq \mathcal{C}_1 \subseteq \mathcal{C}_0$

$$\boldsymbol{m'}_{\lambda-i} = \pi^{(\lambda-i)}[\boldsymbol{c}_{\lambda-i+1}\mathbf{H}_{i\backslash 0}^T, \boldsymbol{c}_{\lambda-i+2}\mathbf{H}_{i\backslash 0}^T, \cdots, \boldsymbol{c}_{\gamma-1}\mathbf{H}_{i\backslash 0}^T]^T(\mathbf{G}_{0\backslash i}\mathbf{H}_{i\backslash 0}^T)^{-1}$$

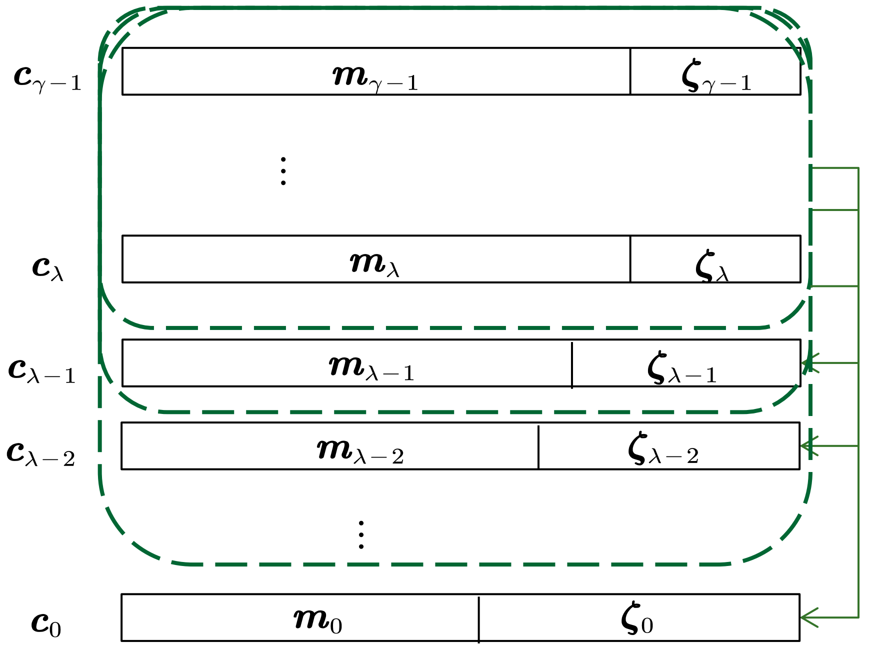


GII Codes

- Encoding

Systematic encoding $\mathcal{C}_\lambda \subseteq \cdots \subseteq \mathcal{C}_2 \subseteq \mathcal{C}_1 \subseteq \mathcal{C}_0$

$$\boldsymbol{\zeta}_{\lambda-i} = \boldsymbol{m}_{\lambda-i}\mathbf{P}_i + \pi^{(\lambda-i)}[\boldsymbol{c}_{\lambda-i+1}\mathbf{H}_i^T, \boldsymbol{c}_{\lambda-i+2}\mathbf{H}_i^T, \cdots, \boldsymbol{c}_{\gamma-1}\mathbf{H}_i^T]^T$$



GII Codes

- Decoding

<u>Successive Decoding in light of $\gamma = 3, \lambda = 2$.</u> $\mathcal{C}_2 \subseteq \mathcal{C}_1 \subseteq \mathcal{C}_0$

Decoding round-0 in $\mathcal{C}_0$

$\text{weight}(\boldsymbol{e}_0) \in [0, t_0]$

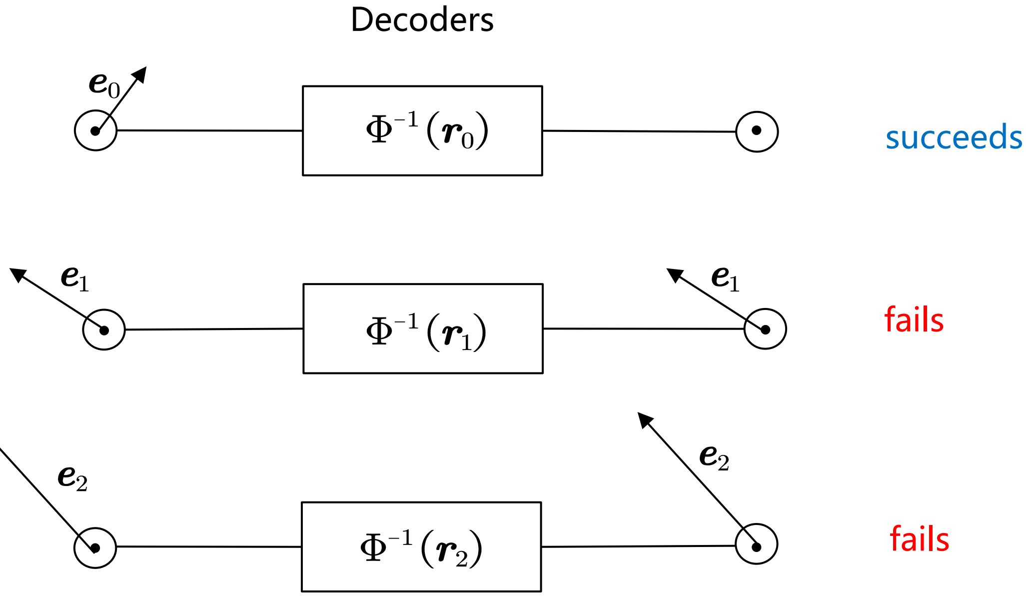
$\boldsymbol{c}_0 \in \mathcal{C}_0$

$\text{weight}(\boldsymbol{e}_1) \in (t_0, t_1]$

$\boldsymbol{c}_1 \in \mathcal{C}_0$

$\text{weight}(\boldsymbol{e}_2) \in (t_1, t_2]$

$\boldsymbol{c}_2 \in \mathcal{C}_0$

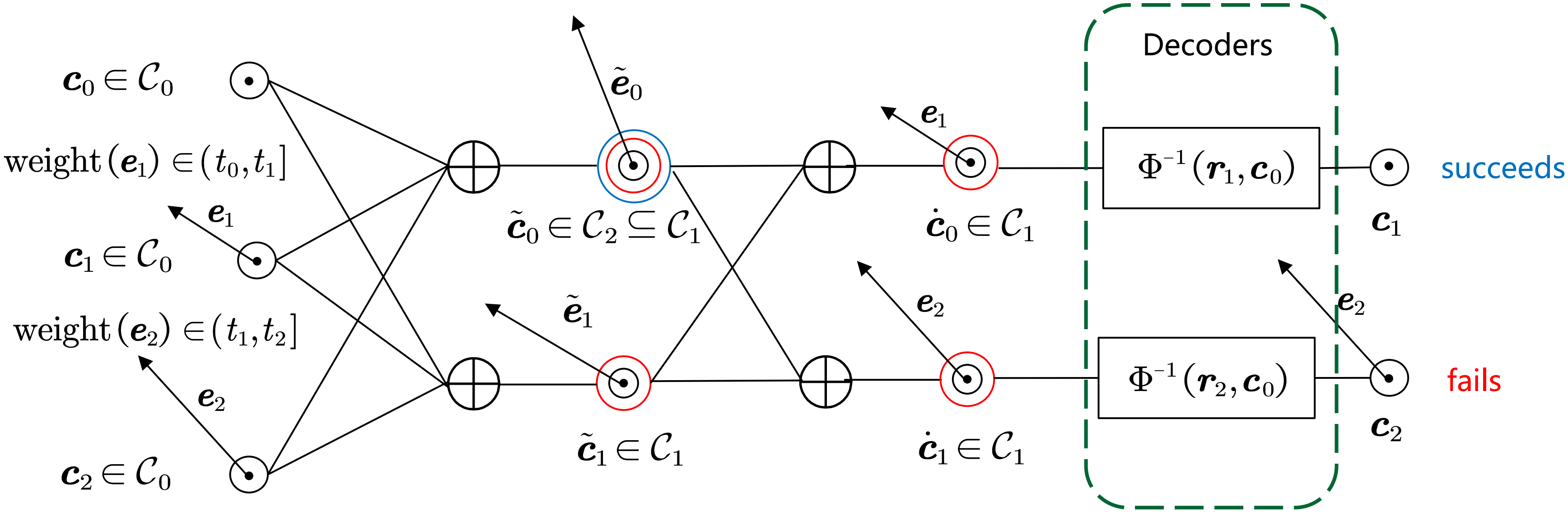


GII Codes

- Decoding

<u>Successive Decoding in light of $\gamma = 3, \lambda = 2.$</u> $\mathcal{C}_2 \subseteq \mathcal{C}_1 \subseteq \mathcal{C}_0$

Decoding round-1 in $\mathcal{C}_1$



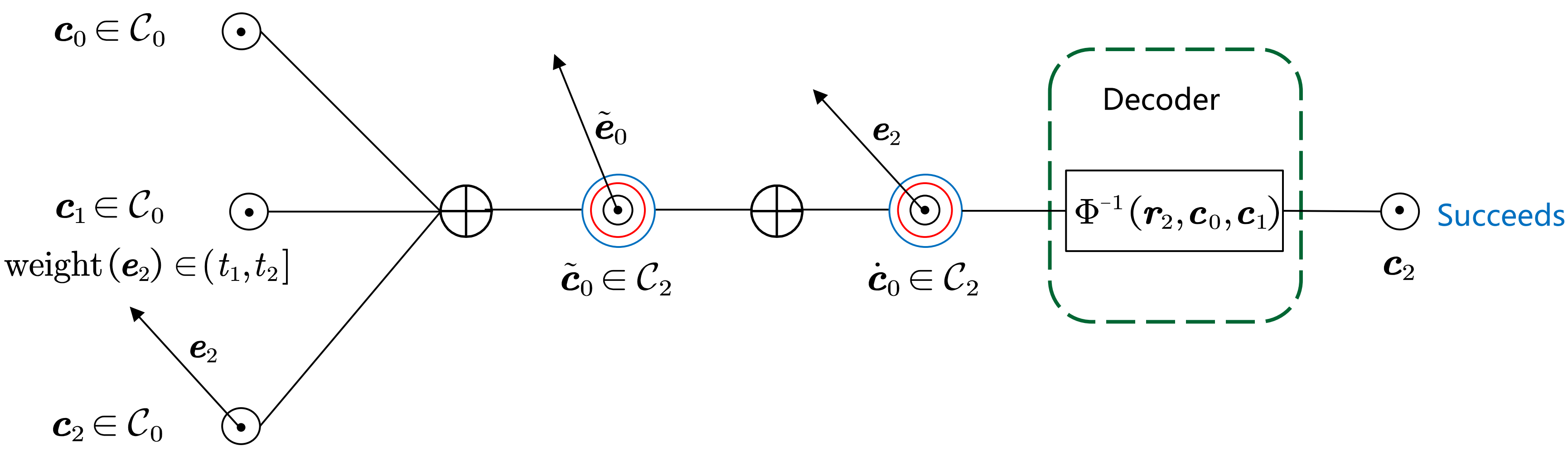
$$\begin{bmatrix} z_{0,0} & z_{0,1} & z_{0,2} \\ z_{1,0} & z_{1,1} & z_{1,2} \end{bmatrix} \begin{bmatrix} \boldsymbol{c}_0 \\ \boldsymbol{c}_1 \\ \boldsymbol{c}_2 \end{bmatrix} = \begin{bmatrix} \tilde{\boldsymbol{c}}_0 \\ \tilde{\boldsymbol{c}}_1 \end{bmatrix} \dashrightarrow \begin{bmatrix} z_{0,1} & z_{0,2} \\ z_{1,1} & z_{1,2} \end{bmatrix}^{-1} \begin{bmatrix} \tilde{\boldsymbol{e}}_0 \\ \tilde{\boldsymbol{e}}_1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{e}_1 \\ \boldsymbol{e}_2 \end{bmatrix} \dashrightarrow$$

GII Codes

- Decoding

Successive Decoding in light of $\gamma=3, \lambda=2$. $\mathcal{C}_2 \subseteq \mathcal{C}_1 \subseteq \mathcal{C}_0$

Decoding round-2 in $\mathcal{C}_2$



$$[z_{0,0} \quad z_{0,1} \quad z_{0,2}] \begin{bmatrix} \boldsymbol{c}_0 \\ \boldsymbol{c}_1 \\ \boldsymbol{c}_2 \end{bmatrix} = \tilde{\boldsymbol{c}}_0 \dashrightarrow z_{0,2}^{-1} \tilde{\boldsymbol{e}}_0 = \boldsymbol{e}_2 \dashrightarrow$$

I_{apriori} comes from the nested codebook structure ☆

GII Codes

- Decoding

Successive Decoding depends on

$$\begin{bmatrix} z_{0,0} & z_{0,1} & z_{0,2} \\ z_{1,0} & z_{1,1} & z_{1,2} \end{bmatrix} \begin{bmatrix} \boldsymbol{c}_0 \\ \boldsymbol{c}_1 \\ \boldsymbol{c}_2 \end{bmatrix} = \begin{bmatrix} \tilde{\boldsymbol{c}}_0 \\ \tilde{\boldsymbol{c}}_1 \end{bmatrix} \dashrightarrow \begin{bmatrix} z_{0,1} & z_{0,2} \\ z_{1,1} & z_{1,2} \end{bmatrix}^{-1}$$

$[\ z\ ]$

\# nested codes

\# remaining components after round-0

a) Submatrix of $[\ z\ ]$ being invertible

a1) # remaining components after round-0 $\leqslant$ # nested codes
a2) full rank

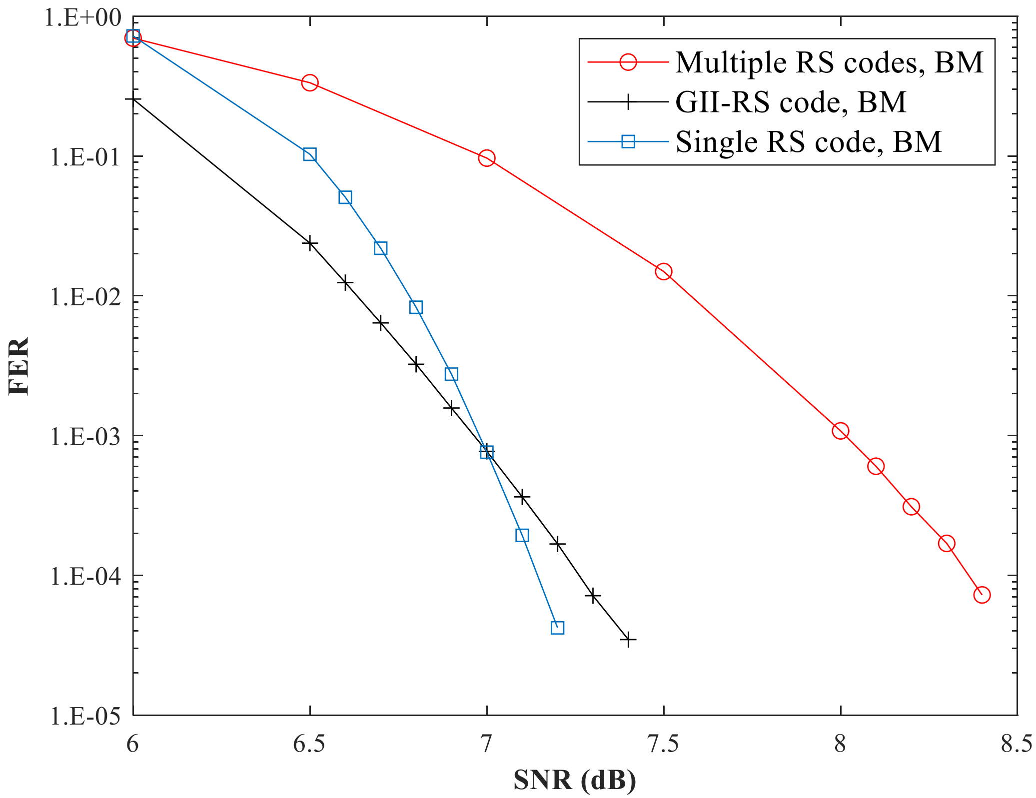
b) Decoding being progressive

$$\Phi^{-1}(\boldsymbol{r}_0) \dashrightarrow \Phi^{-1}(\boldsymbol{r}_1, \boldsymbol{c}_0) \dashrightarrow \Phi^{-1}(\boldsymbol{r}_2, \boldsymbol{c}_0, \boldsymbol{c}_1)$$

GII Codes

- Decoding Performance

Decoding performance over the AWGN channel using BPSK



Multiple RS
(63, 31) RS
(63, 19) RS
(63, 17) RS
(63, 15) RS

(252, 82) GII-RS
$\mathcal{C}_0$ (63, 31) RS
$\mathcal{C}_1$ (63, 19) RS
$\mathcal{C}_2$ (63, 17) RS
$\mathcal{C}_3$ (63, 15) RS

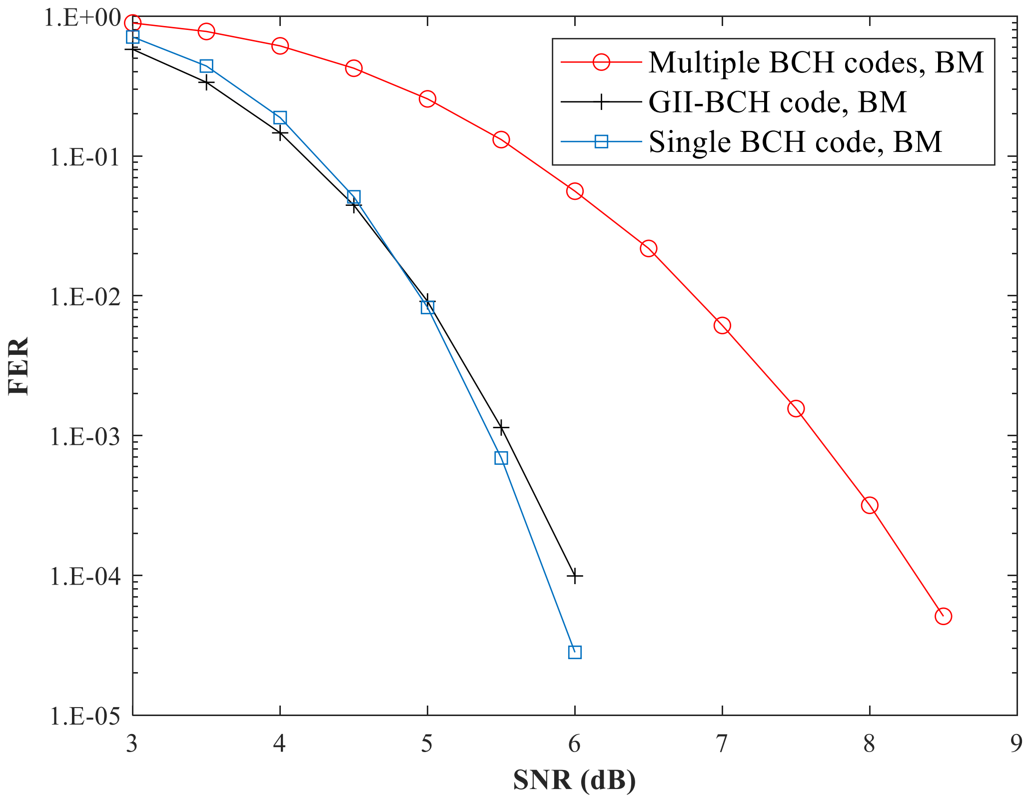
Single RS
(255, 83) RS

	Multiple RS	GII-RS	Single RS
d	33	49	173

GII Codes

- Decoding Performance

Decoding performance over the AWGN channel using BPSK



Multiple BCH
(63, 45) BCH
(63, 45) BCH
(63, 30) BCH
(63, 18) BCH

(252,138) GII-BCH	
$\mathcal{C}_0$	(63, 45) BCH
$\mathcal{C}_1$	(63, 30) BCH
$\mathcal{C}_2$	(63, 18) BCH

Single BCH
(255, 139) BCH

	Multiple BCH	GII-BCH	Single BCH
d	7	21	31

GII Codes

- Improved Decoding

<u>Component code decoding: BM → GS</u>

BM algorithm $\Omega(x) = \Lambda(x)S(x) \mod x^2$

$\hat{\boldsymbol{e}} \rightarrow \hat{\boldsymbol{c}}$ If $\text{weight}(\boldsymbol{e}) \leqslant \left\lfloor \frac{d}{2} \right\rfloor$,

$\hat{\boldsymbol{e}} = \boldsymbol{e}$ / $\hat{\boldsymbol{c}} = \boldsymbol{c}$

GS algorithm $Q(x,y),\ y - \hat{m}(x) \mid Q(x,y)$

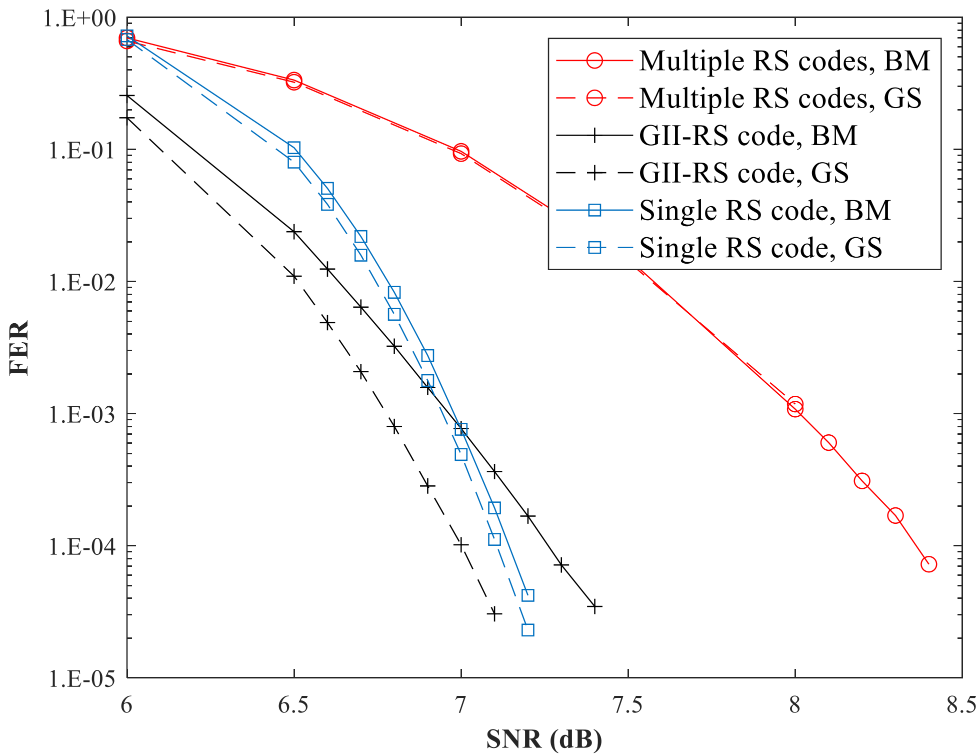
$\hat{\boldsymbol{m}}/\hat{\boldsymbol{c}}$ If $\text{weight}(\boldsymbol{e}) \leqslant n - \left\lfloor \sqrt{n(n-d)} \right\rfloor - 1$,

$\hat{\boldsymbol{m}} = \boldsymbol{m}$ / $\hat{\boldsymbol{c}} = \boldsymbol{c}$

GII Codes

- Improved Decoding Performance

Decoding performance over the AWGN channel using BPSK



Multiple RS
(63, 31) RS
(63, 19) RS
(63, 17) RS
(63, 15) RS

(252, 82) GII-RS
$\mathcal{C}_0$ (63, 31) RS
$\mathcal{C}_1$ (63, 19) RS
$\mathcal{C}_2$ (63, 17) RS
$\mathcal{C}_3$ (63, 15) RS

Single RS
(255, 83) RS

	Multiple RS	GII-RS	Single RS
d	33	49	173

End / Beginning ...

- Classic Algebraic Structures and Codes
- Hamming Domain and Codebooks
- Finite Fields

are the Cradles for Improving Code,

especially when decoding latency & / power consumption become critical, and when soft information is absence.

Acknowledgements







Huazi Zhang   Yingquan Wu   Jinjun Cheng   Changyu Wu   Wenhao Chen   Jingyu Lin   Xinzheng He

Tyne Bridge of Newcastle, opened in 1928

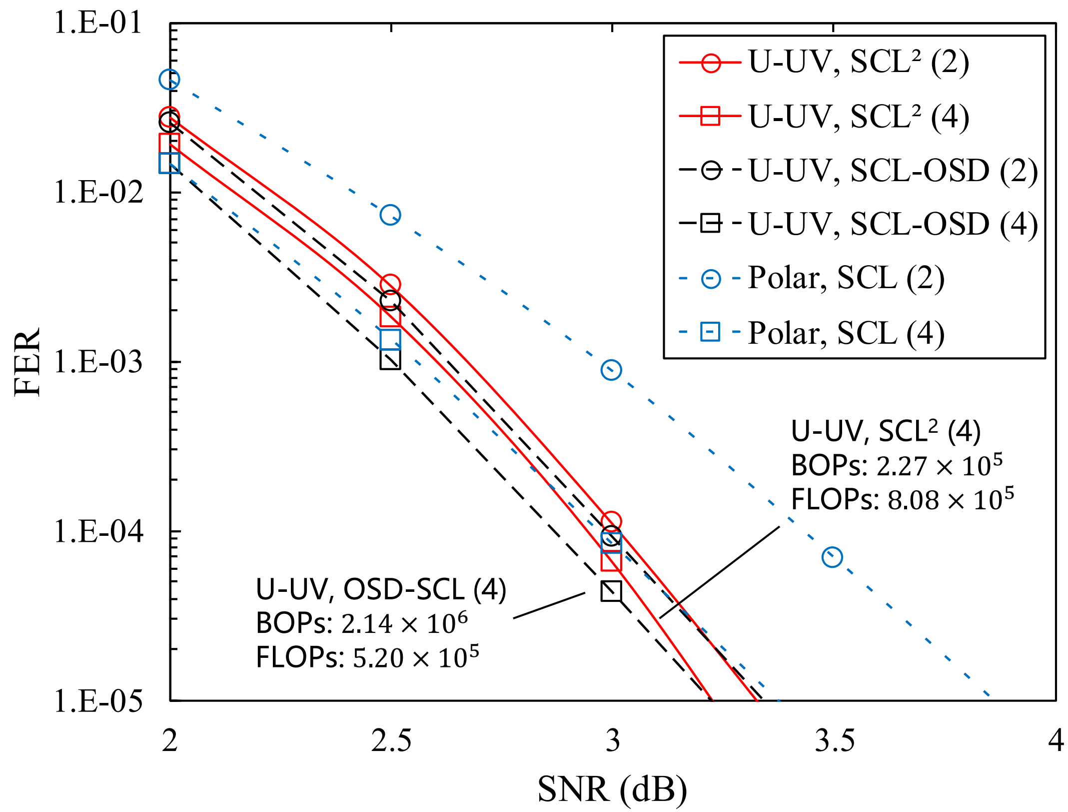
Sydney Harbour Bridge, opened in 1932

谢谢！

U-UV Codes

- Improved Decoding

SCL-SCL (SCL^2) decoding performance over the AWGN channel using BPSK



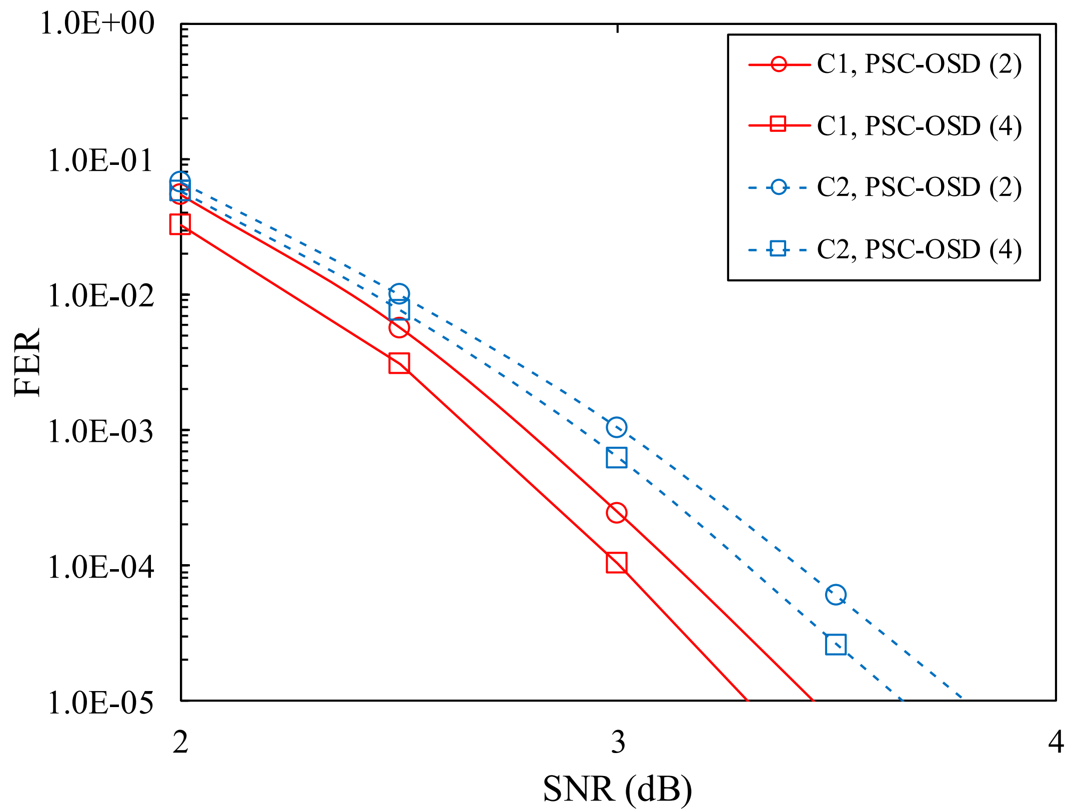
(512, 250) U-UV
(64, 57) eBCH
(64, 51) eBCH
(64, 45) eBCH
(64, 24) eBCH
(64, 45) eBCH
(64, 18) eBCH
(64, 10) eBCH
(64, 0) eBCH

(512, 250) polar
5G NR, CRC-8

U-UV Codes

- Improved Decoding

Permutation SC (PSC) decoding performance over the AWGN channel using BPSK



New design

C_1 (504, 270)
(63, 57) BCH
(63, 51) BCH
(63, 51) BCH
(63, 24) BCH
(63, 51) BCH
(63, 18) BCH
(63, 18) BCH
(63, 0) BCH

Original design

C_2 (504, 268)
(63, 57) BCH
(63, 51) BCH
(63, 51) BCH
(63, 30) BCH
(63, 45) BCH
(63, 18) BCH
(63, 16) BCH
(63, 0) BCH

GII Codes

- Encoding

Non-systematic encoding $\mathcal{C}_\lambda \subseteq \cdots \subseteq \mathcal{C}_2 \subseteq \mathcal{C}_1 \subseteq \mathcal{C}_0$

Generate $\boldsymbol{c}_{\lambda-i},\ (0 \leqslant i \leqslant \lambda)$

$$\boldsymbol{c}_{\lambda-i} = [\boldsymbol{m}_{\lambda-i}, \boldsymbol{m}'_{\lambda-i}]\mathbf{G}_0$$

$\boldsymbol{m}_{\lambda-i}$	$\boldsymbol{m}'_{\lambda-i}$
dim: k_i	dim: $k_0 - k_i$

$$\boldsymbol{c}_{\lambda-i}\mathbf{H}_i^T = \pi^{(\lambda-i)}[\boldsymbol{c}_{\lambda-i+1}\mathbf{H}_i^T, \boldsymbol{c}_{\lambda-i+2}\mathbf{H}_i^T, \cdots, \boldsymbol{c}_{\gamma-1}\mathbf{H}_i^T]^T,\ 1 \leq i \leq \lambda$$

$$\begin{aligned}\boldsymbol{c}_{\lambda-i}\mathbf{H}_i^T &= [\boldsymbol{m}_{\lambda-i}\ \boldsymbol{m}'_{\lambda-i}]\mathbf{G}_0\mathbf{H}_i^T \\ &= [\boldsymbol{m}_{\lambda-i}\ \boldsymbol{m}'_{\lambda-i}]\begin{bmatrix}\mathbf{G}_i \\ \mathbf{G}_{0\backslash i}\end{bmatrix}\mathbf{H}_i^T\end{aligned}$$

$$\boldsymbol{m}'_{\lambda-i} = \pi^{(\lambda-i)}[\boldsymbol{c}_{\lambda-i+1}\mathbf{H}_{i\backslash 0}^T, \boldsymbol{c}_{\lambda-i+2}\mathbf{H}_{i\backslash 0}^T, \cdots, \boldsymbol{c}_{\gamma-1}\mathbf{H}_{i\backslash 0}^T]^T(\mathbf{G}_{0\backslash i}\mathbf{H}_{i\backslash 0}^T)^{-1}$$

GII Codes

- Encoding

<u>Systematic encoding</u> $\mathcal{C}_\lambda \subseteq \cdots \subseteq \mathcal{C}_2 \subseteq \mathcal{C}_1 \subseteq \mathcal{C}_0$

Generate $\boldsymbol{c}_{\lambda-i},\ (0 \leqslant i \leqslant \lambda)$

$$\boldsymbol{c}_{\lambda-i} = [\boldsymbol{m}_{\lambda-i},\ \boldsymbol{\zeta}_{\lambda-i}]$$

$\boldsymbol{c}_{\lambda-i}$	$\boldsymbol{m}_{\lambda-i}$	$\boldsymbol{\zeta}_{\lambda-i}$
	dim: k_i	dim: $n-k_i$

$$\mathbf{G}_i = [\mathbf{I}_{k_i}\ \ \mathbf{P}_i]$$

$$\mathbf{H}_i^T = \begin{bmatrix} \mathbf{P}_i \\ \mathbf{I}_{n-k_i} \end{bmatrix}$$

$$\boldsymbol{c}_{\lambda-i}\mathbf{H}_i^T = \pi^{(\lambda-i)}[\boldsymbol{c}_{\lambda-i+1}\mathbf{H}_i^T, \boldsymbol{c}_{\lambda-i+2}\mathbf{H}_i^T, \cdots, \boldsymbol{c}_{\gamma-1}\mathbf{H}_i^T]^T$$

$$[\boldsymbol{m}_i\ \boldsymbol{\zeta}_i]\begin{bmatrix} \mathbf{P}_i \\ \mathbf{I}_{n-k_i} \end{bmatrix} = \pi^{(\lambda-i)}[\boldsymbol{c}_{\lambda-i+1}\mathbf{H}_i^T, \boldsymbol{c}_{\lambda-i+2}\mathbf{H}_i^T, \cdots, \boldsymbol{c}_{\gamma-1}\mathbf{H}_i^T]^T$$

$$\boldsymbol{\zeta}_{\lambda-i} = \boldsymbol{m}_{\lambda-i}\mathbf{P}_i + \pi^{(\lambda-i)}[\boldsymbol{c}_{\lambda-i+1}\mathbf{H}_i^T, \boldsymbol{c}_{\lambda-i+2}\mathbf{H}_i^T, \cdots, \boldsymbol{c}_{\gamma-1}\mathbf{H}_i^T]^T$$

GII Codes

- Enhanced Chase Decoding

<u>Component code decoding: hard-decision → Chase</u>

Round 0	Round 1	Round 2
Decoders	Decoders	Decoder
$\Phi^{-1}(\boldsymbol{r}_0)$	$\Phi^{-1}(\boldsymbol{r}_1, \boldsymbol{c}_0)$	$\Phi^{-1}(\boldsymbol{r}_2, \boldsymbol{c}_0, \boldsymbol{c}_1)$
$\Phi^{-1}(\boldsymbol{r}_1)$	$\Phi^{-1}(\boldsymbol{r}_2, \boldsymbol{c}_0)$	
$\Phi^{-1}(\boldsymbol{r}_2)$		

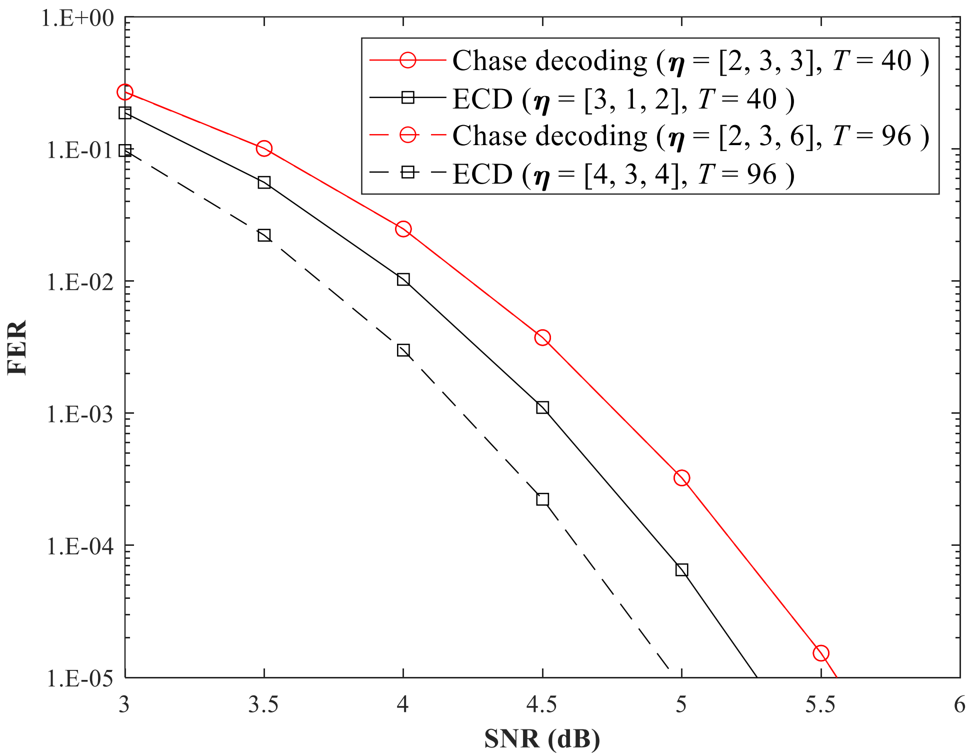
Error probability: P_{f_0} P_{f_1} P_{f_2}

$$P_{f_b} = \max\{P_{f_0}, P_{f_1}, P_{f_2}\} \rightleftarrows \eta_b = \eta_b + 1$$

GII Codes

- Enhanced Chase Decoding Performance

Enhanced Chase Decoding performance over the AWGN channel using BPSK



(252,138) GII-BCH	
$\mathcal{C}_0$	(63, 45) BCH
$\mathcal{C}_1$	(63, 30) BCH
$\mathcal{C}_2$	(63, 18) BCH

Maximum number of test vector : T

Flipping vector : $\boldsymbol{\eta}$