Duality of Channel Encoding and Decoding for Rate-1 Convolutional Codes

Yonghui Li

Electrical and Information Engineering
The University of Sydney

Background

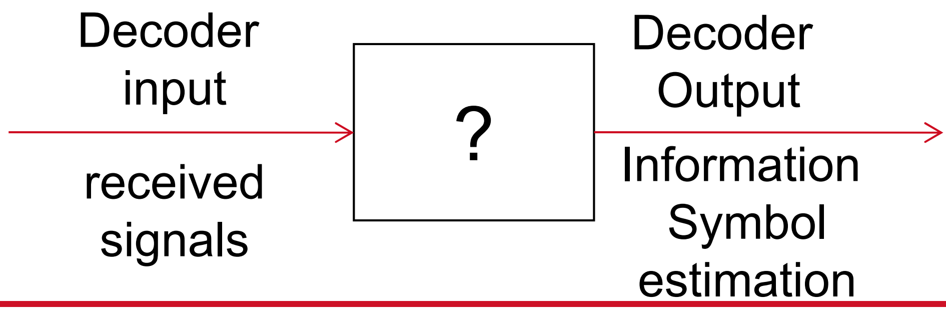
- Channel coding provides reliable transmission by introducing structured redundancy into the message
- Coding theorist has been trying to find the capacity-approaching codes
- Turbo and LDPC codes approach the capacity limit within 1dB on AWGN channels

Complex BCJR MAP Decoding

- Performance and complexity are usually contradicting metrics
- Strong codes usually have high decoding complexity and high latency
- Turbo decoder consists of iterative decoding of two convolutional decoders
- BCJR maximum a posteriori (MAP) probability decoder is the optimal convolutional decoder
- It has an exponential complexity in terms of the memory order
- Little progress so far in reducing decoding complexity without losing optimality

MAP Decoding

- The key hurdle is due to the fundamental lack of theoretical understanding of decoding process
- BCJR MAP decoding of a convolutional code is complex dynamic process
- Decoding depends on all received signals and all historical decoding information
- No explicit decoder input-output transfer function is known
- Thus difficult to simplify the decoder



Motivation

- (1) Convolutional encoder is simple and can be implemented by shift registers
- (2) Decoding is a reverse process of encoding process

This implies that there should exist some explicit relationship between encoder and decoder

(1) What is the explicit relationship between encoding and decoding?

(2) Can the decoder be represented by just using simple shift register encoder structures as well ?

Contributions in this initial work

- Revisit the BCJR soft-input soft-output (SISO) MAP decoding process of rate-1 convolutional codes.
- Establish some interesting duality properties between encoding and decoding of rate-1 convolutional codes.
- The BCJR SISO MAP decoders can be represented by their dual SISO channel encoders using shift registers in the complex number field.
- The dual encoder structures for various recursive and non-recursive rate-1 convolutional codes are derived.

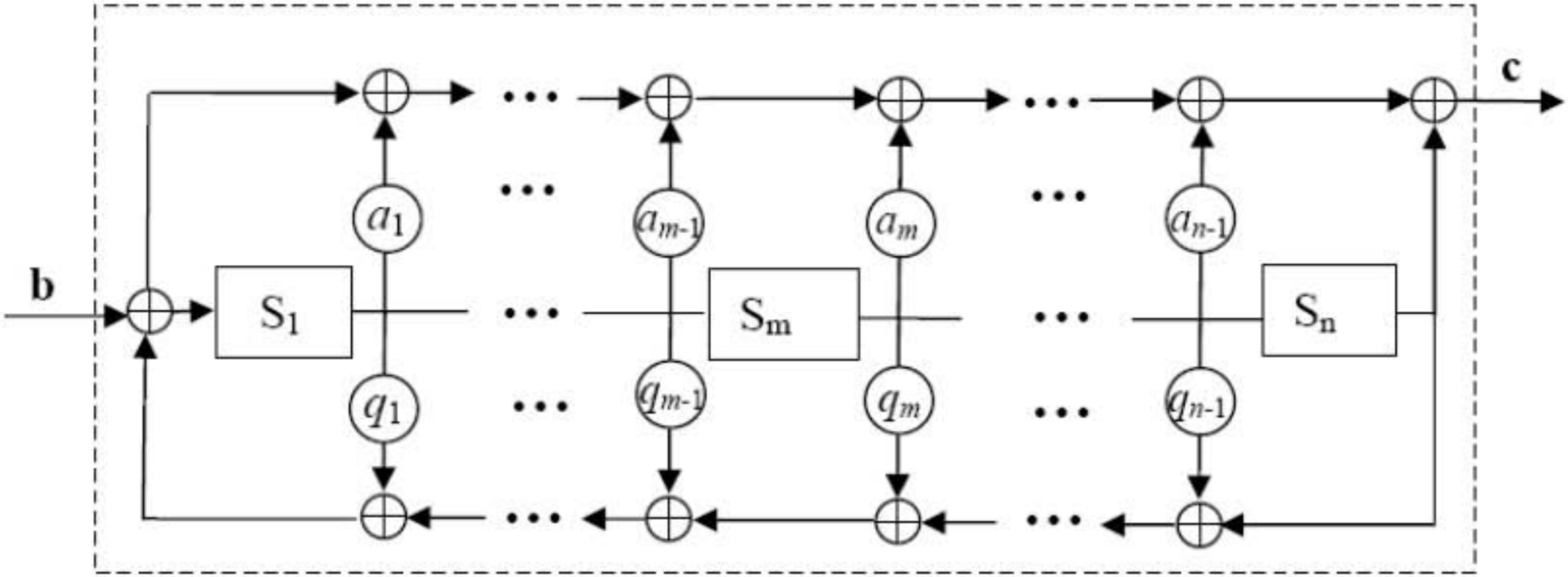
Reference: Li, Rahman, Vucetic, "Duality of channel encoding and decoding: part I – Rate 1 convolutional codes," submitted, available on Arxiv.

Outline

- o Introduction to Convolutional Codes
- o Duality Property for MAP Forward Decoding
- o Duality Property for MAP Backward Decoding
- o Duality Property for Bidirectional MAP Decoding
- o Conclusions

Convolutional Encoder

› It is so called because encoder performs convolutional process of information message **b** and encoder impulse response



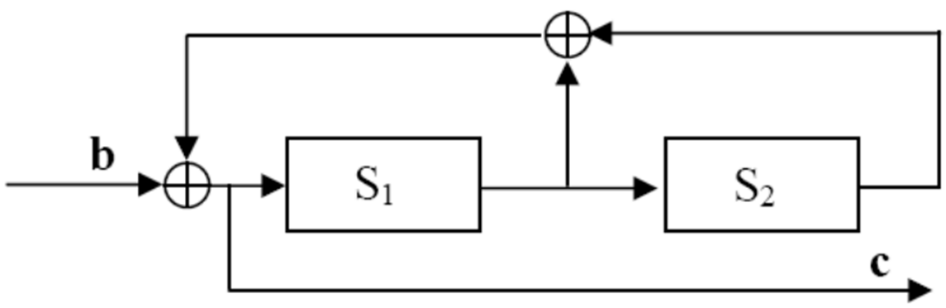
Rate-1 convolutional encoder, generated by g(x)=a(x)/q(x)

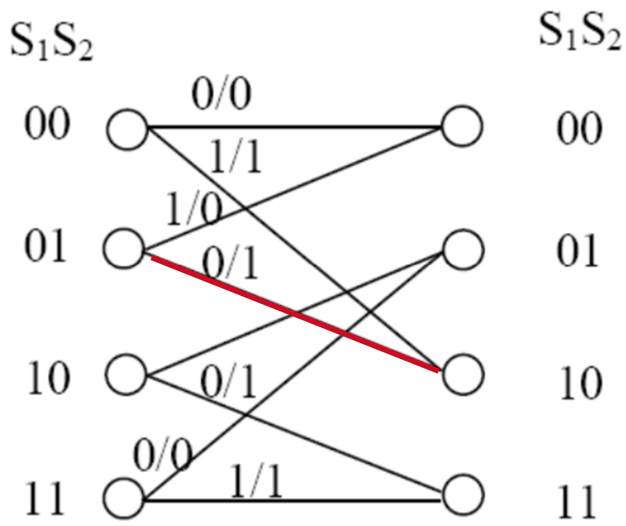
Trellis Diagram

› Encoding process is a finite machine

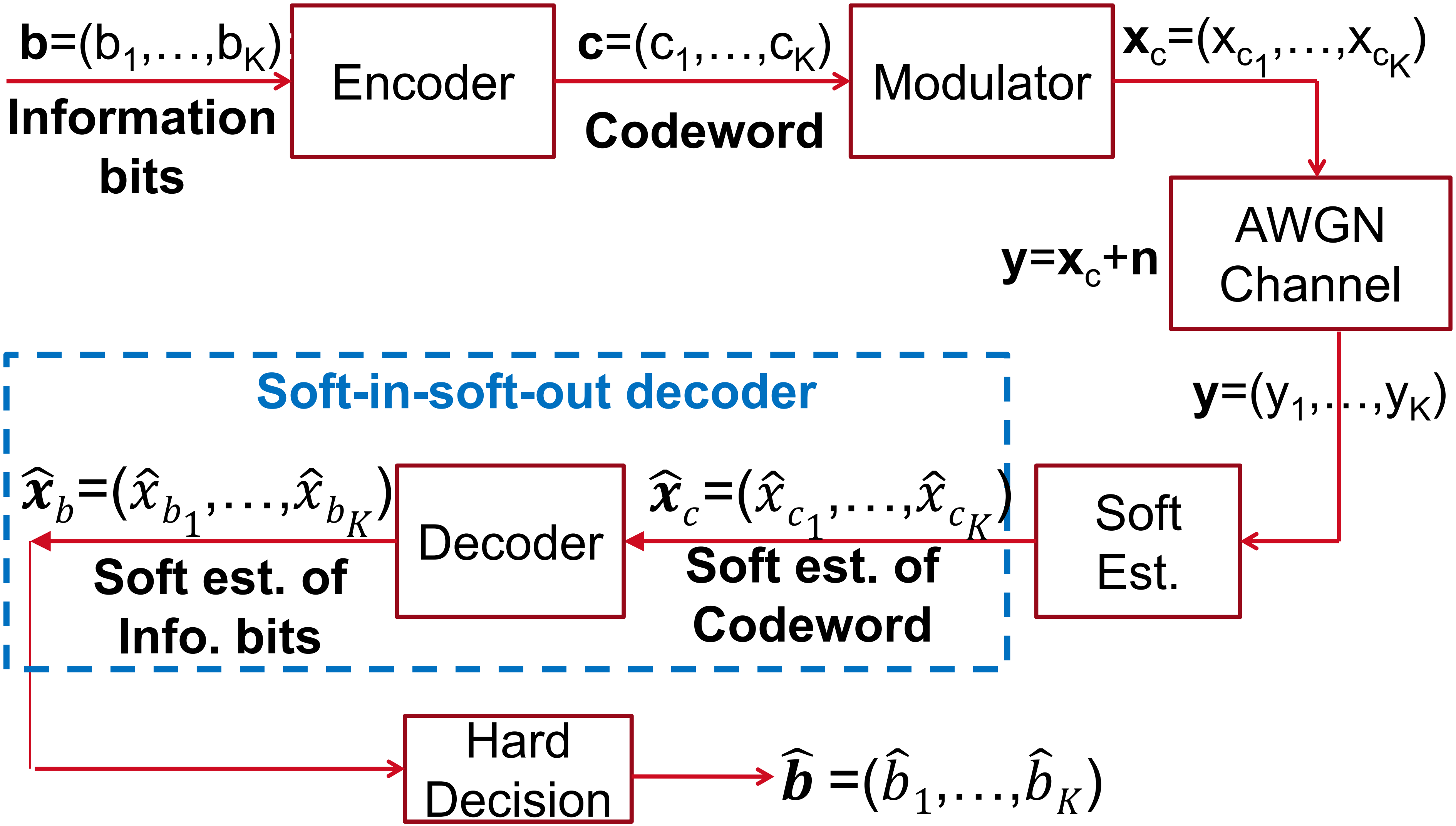
› Trellis diagram representation

› Trellis illustrates state transition and in/out relationship





Transceiver Structure



BCJR MAP Decoding

- BCJR MAP decoding is the optimal symbol-wise decoding algorithm
- It is a bidirectional decoding process
- It consists of a forward and a backward recursive calculation

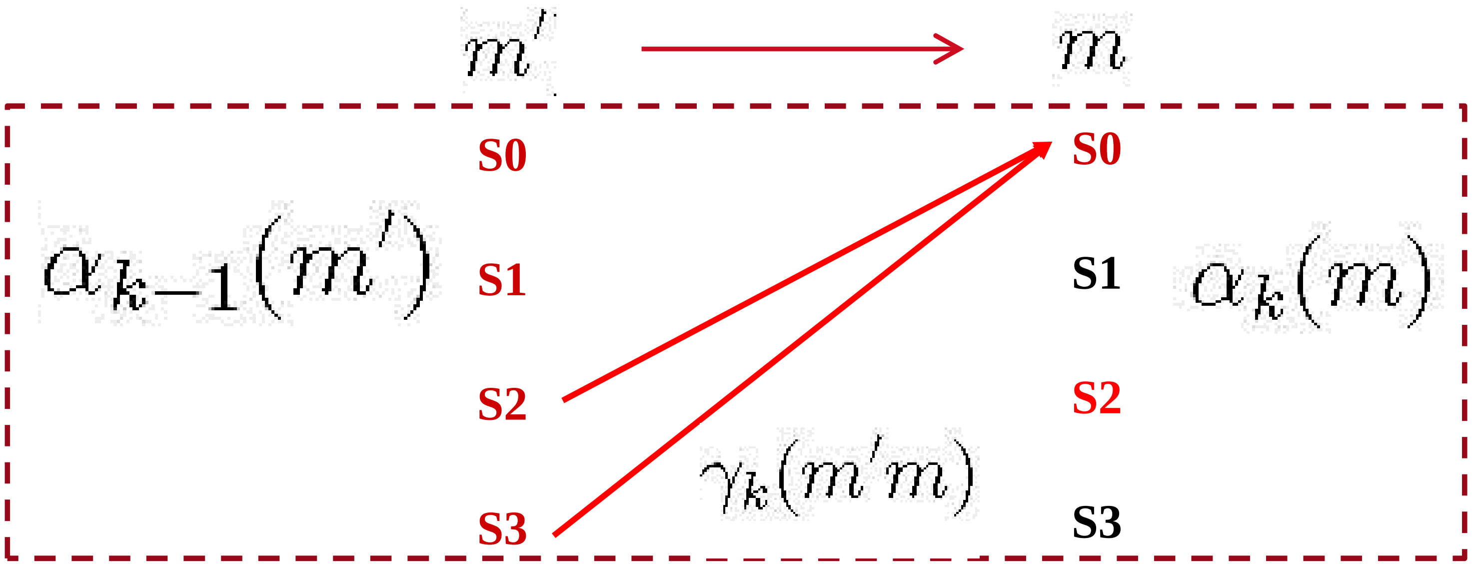
$$P_{b_k}(w) = P(b_k = w|y)$$

$$= \sum_{(m',m)\in U(b(k)=w)} \alpha_k(m')\beta_{k+1}(m)\gamma_k(m',m)$$

BCJR MAP Decoding

- $\alpha_{k-1}(m')$: Forward recursion variable

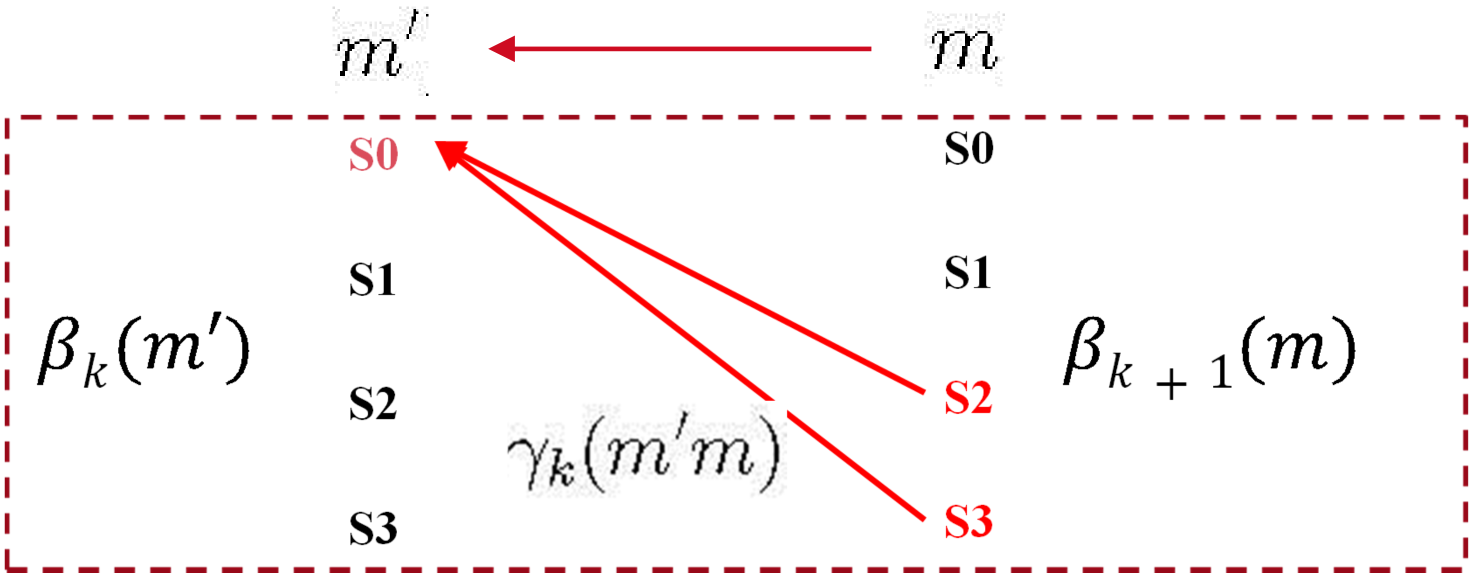
$$\alpha_k(m) = \sum_{m'} \alpha_{k-1}(m')\gamma_k(m'm)$$

- 

BCJR MAP Decoding

- $\beta_{k+1}(m')$: Backward recursion variable

$$\beta_k(m') = \sum_m \beta_{k+1}(m)\gamma_k(m',m)$$

- 

BCJR MAP Decoding

- Decompose the bidirectional MAP decoding into a forward and a backward decoding

$$P_{b_k}(\mathrm{w}|y_1, \ldots, y_k, \ldots, y_K)$$

- Forward decoding:

$$\vec{P}_{b_k}(\mathrm{w}|y_1, \ldots, y_k) = \sum \alpha_k(m')\gamma_k(m', m)$$

- Backward decoding:

$$\overleftarrow{P}_{b_k}(w|y_k, \ldots, y_K) = \sum \beta_{k+1}(m)\gamma_k(m', m)$$

- Represent the bidirectional MAP decoding output as the linear combination of forward and backward decoding output

Outline

- o Introduction to Convolutional Codes
- o Duality Property for MAP Forward Decoding
- o Duality Property for MAP Backward Decoding
- o Duality Property for Bidirectional MAP Decoding
- o Conclusions

Convolutional Codes

- Let
- $a(x) = x^n + a_{n-1}x^{n-1} + \cdots + a_1 x_1 + 1$
- $q(x) = x^n + q_{n-1}x^{n-1} + \cdots + q_1 x_1 + 1$
-
- Definitions:
- $g_{FBC}(x) = 1/q(x)$: feedback-only convolutional (FBC) code
-
- $g_{FFC}(x) = a(x)$:feed-forward only convolutional (FFC) code
-
- $g_{GC}(x) = a(x)/g(x)$: general convolutional (GC) code

Feedback-only CC (FBC)

- Example 1
- Consider a FBC code, generated by
- $$g_{FBC}(x) = \frac{1}{x^2+x+1}$$

$$\hat{\mathbf{x}}_{\mathbf{c}} = (\hat{x}_{c_1}, \cdots, \hat{x}_{c_K}) \longrightarrow \boxed{\text{SISO Decoder}} \longrightarrow \hat{\mathbf{x}}_{\mathbf{b}} = (\hat{x}_{b_1}, \cdots, \hat{x}_{b_K})$$

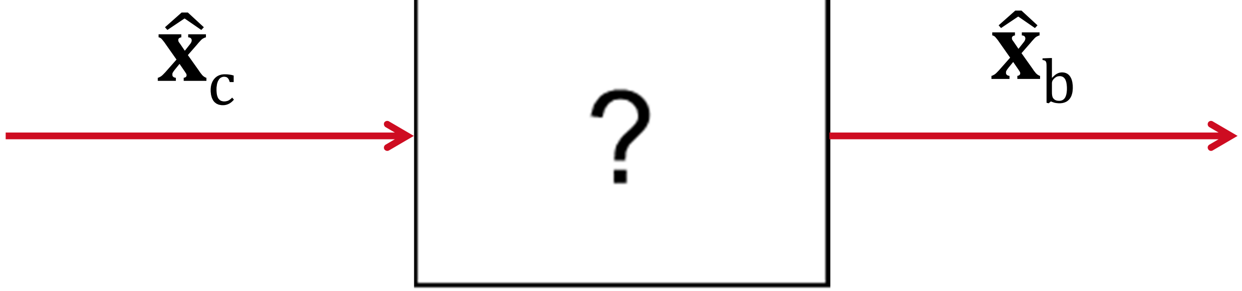
Soft estimate seq. of received codeword **c**

SISO Decoder

Soft estimate seq. of info **b**

Example 1

- What is the transfer function of the decoder?
- What is the explicit relationship of decoder input $\hat{\mathbf{x}}_c$ and output $\hat{\mathbf{x}}_b$



Decoder

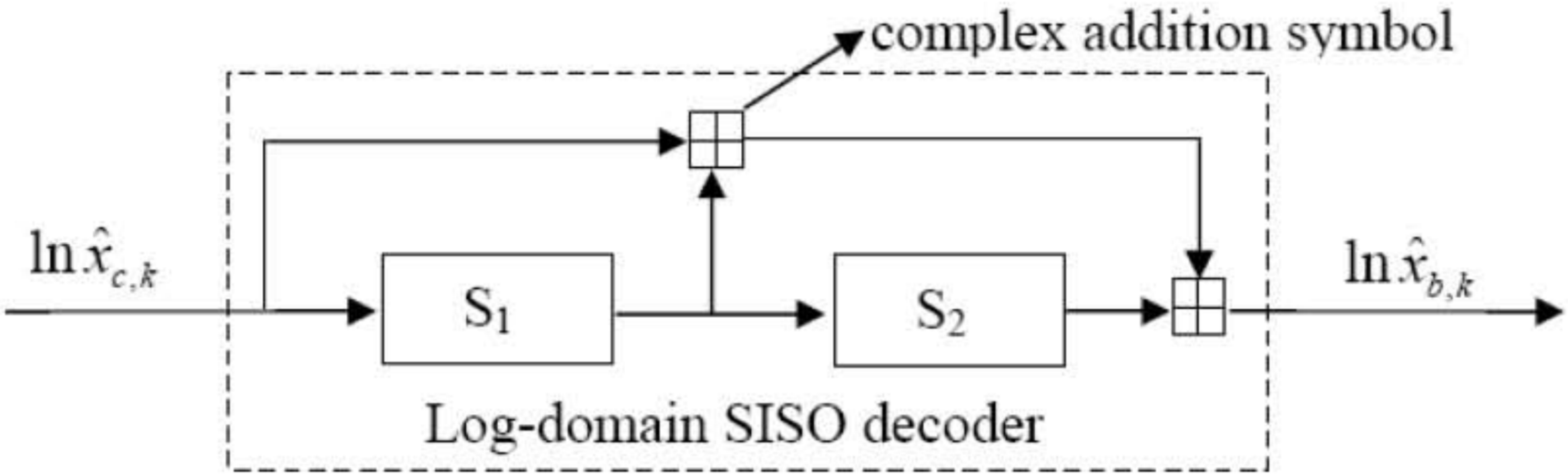
Example 1

- The forward decoder input $\hat{x}_{c_k}$ and decoder output $\hat{x}_{b_k}$ has the following linear relationship,
- $$\hat{x}_{b_k} = \hat{x}_{c_k} \hat{x}_{c_{k-1}} \hat{x}_{c_{k-2}}$$
- or in the logarithm domain

$$ln \hat{x}_{b_k} = ln \hat{x}_{c_k} + ln \hat{x}_{c_{k-1}} + ln \hat{x}_{c_{k-2}}$$

Log-domain Forward Decoder Structure of Example 1

›

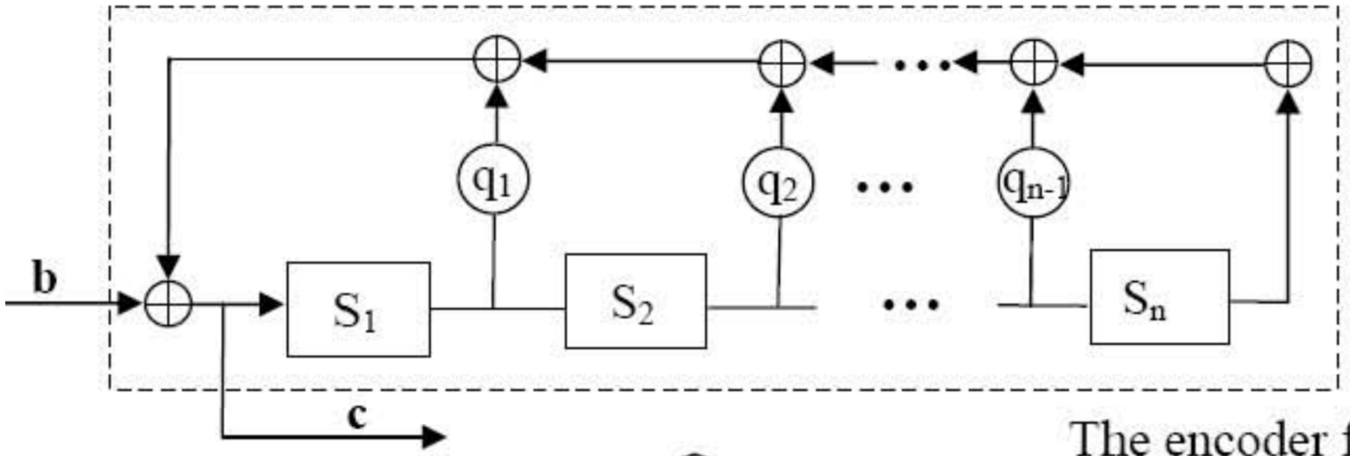


Decoder implemented using an equivalent encoder, generated by the polynomial $1/g(x) = x^2+x+1$

The Log-domain forward decoding of the code $g(x) = 1/(x^2+x+1)$ is simply the convolutional encoder, generated by the polynomial $1/g(x) = x^2+x+1$.

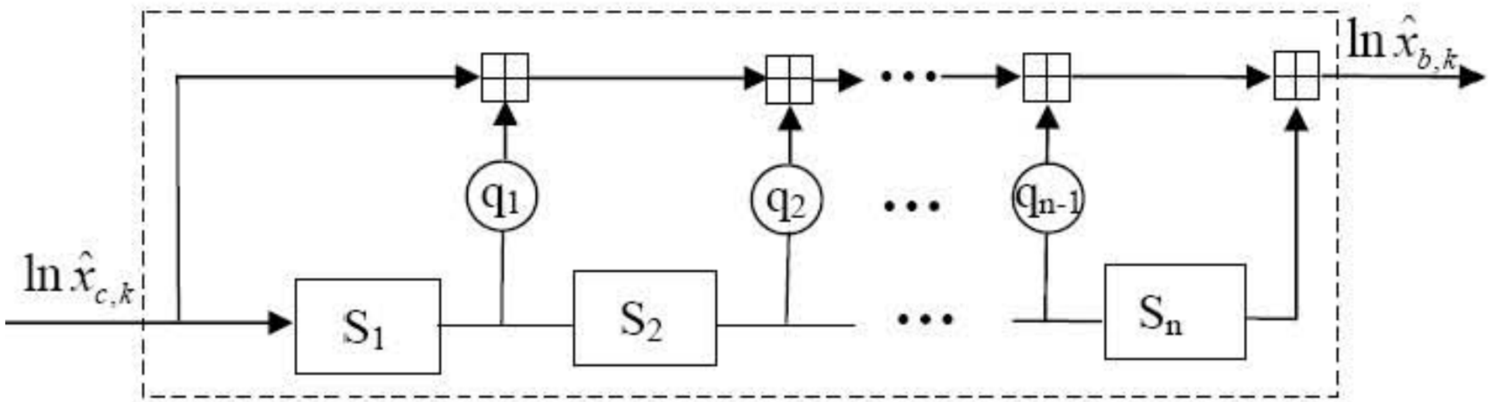
Duality for Feedback-only CC (FBC)

- **Theorem 1 -** For a FBC code, generated by a generator polynomial
- $$g_{FBC}(x) = 1/q(x),$$
- we define its dual encoder as the encoder with the inverse polynomial of $g_{FBC}(x)$, given by
- $$q_{FBC}(x) = 1/g_{FBC}(x) = q(x).$$
- Then the log-domain SISO forward decoding of the FBC code can be simply implemented by its dual encoder in a complex field.



The encoder for the code generated by

$$g_{FBC}(x) = 1/q(x) = 1/(x^n + \cdots q_1 x + 1)$$



The dual encoder of the log-domain SISO forward decoder for the code $g_{FBC}(x)$, given by

$$q_{FBC}(x) = 1/g_{FBC}(x) = x^n + \cdots q_1 x + 1$$

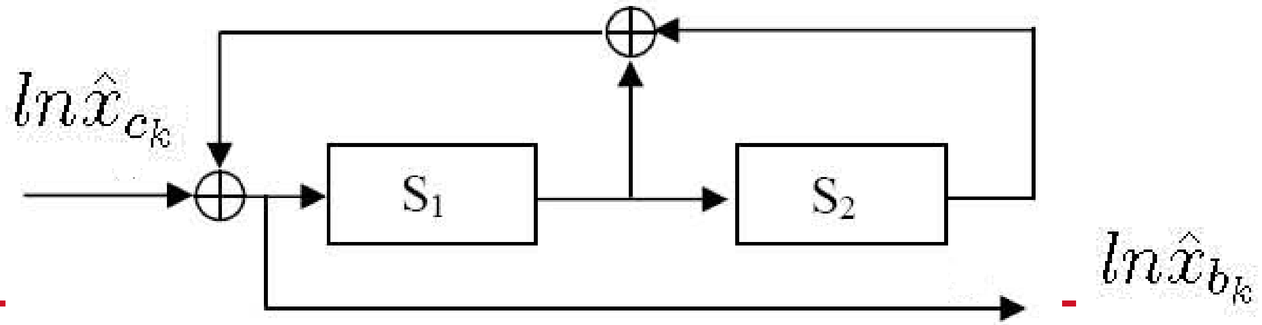
Duality for Feedforward-only CC (FFC)

- Theorem 1 reveals the encoding-decoding duality property for feedback only code (FBC)
- Does the Theorem 1 apply to the FFC codes and general recursive codes?

Log soft input $ln\hat{x}_{c_k}$	Log soft output of the dual encoder $ln\ddot{x}_{b_k}$	Desired soft decoding output $ln\hat{x}_{b_k}$
$ln\hat{x}_{c_1}$	$ln\hat{x}_{c_1}$	$ln\hat{x}_{c_1}$
$ln\hat{x}_{c_2}$	$ln\hat{x}_{c_2} + ln\hat{x}_{c_1}$	$ln\hat{x}_{c_2} + ln\hat{x}_{c_1}$
$ln\hat{x}_{c_3}$	$ln\hat{x}_{c_3} + ln\hat{x}_{c_2} + \boxed{ln\hat{x}_{c_1} + ln\hat{x}_{c_1}}$	$ln\hat{x}_{c_3} + ln\hat{x}_{c_2}$
$ln\hat{x}_{c_4}$	$ln\hat{x}_{c_4}+ln\hat{x}_{c_3} + ln\hat{x}_{c_1}+ \boxed{ln\hat{x}_{c_2}+ln\hat{x}_{c_2}+ln\hat{x}_{c_1}+ln\hat{x}_{c_1}}$	$ln\hat{x}_{c_4}+ln\hat{x}_{c_3} + ln\hat{x}_{c_1}$
$ln\hat{x}_{c_5}$	$ln\hat{x}_{c_5}+ln\hat{x}_{c_4} + ln\hat{x}_{c_2}+ln\hat{x}_{c_1} + \boxed{ln\hat{x}_{c_3} +ln\hat{x}_{c_3}+ln\hat{x}_{c_2} +ln\hat{x}_{c_2}}$ $\boxed{+ln\hat{x}_{c_1}+ln\hat{x}_{c_1}+ln\hat{x}_{c_1}+ln\hat{x}_{c_1}}$	$ln\hat{x}_{c_5}+ln\hat{x}_{c_4} + ln\hat{x}_{c_2}+ln\hat{x}_{c_1}$
$\vdots$	$\vdots$	$\vdots$

$$g_{FFC}(x) = x^2 + x + 1$$

Log soft input $ln\hat{x}_{c_k}$	Memory S_1 of the dual encoder	Memory S_2 of the dual encoder	Log soft output of the dual encoder $ln\ddot{x}_{b_k}$
$ln\hat{x}_{c_1}$	0	0	$ln\hat{x}_{c_1}$
$ln\hat{x}_{c_2}$	$ln\hat{x}_{c_1}$	0	$ln\hat{x}_{c_2} + ln\hat{x}_{c_1}$
$ln\hat{x}_{c_3}$	$ln\hat{x}_{c_2} + \boxed{ln\hat{x}_{c_1}}$	$\boxed{ln\hat{x}_{c_1}}$	$ln\hat{x}_{c_3} + ln\hat{x}_{c_2} + \boxed{ln\hat{x}_{c_1} + ln\hat{x}_{c_1}}$
$ln\hat{x}_{c_4}$	$ln\hat{x}_{c_3}+ln\hat{x}_{c_2} + ln\hat{x}_{c_1}+ln\hat{x}_{c_1}$	$ln\hat{x}_{c_2} + ln\hat{x}_{c_1}$	$ln\hat{x}_{c_4}+ln\hat{x}_{c_3} + ln\hat{x}_{c_1}+ \boxed{ln\hat{x}_{c_2}+ln\hat{x}_{c_2}+ln\hat{x}_{c_1}+ln\hat{x}_{c_1}}$
$ln\hat{x}_{c_5}$	$ln\hat{x}_{c_4}+ln\hat{x}_{c_3} + ln\hat{x}_{c_1} +ln\hat{x}_{c_2}+ln\hat{x}_{c_2}+ln\hat{x}_{c_1}+ln\hat{x}_{c_1}$	$ln\hat{x}_{c_3}+ln\hat{x}_{c_2} + ln\hat{x}_{c_1}+ln\hat{x}_{c_1}$	$ln\hat{x}_{c_5}+ln\hat{x}_{c_4} + ln\hat{x}_{c_2}+ln\hat{x}_{c_1} + \boxed{ln\hat{x}_{c_3} +ln\hat{x}_{c_3}+ln\hat{x}_{c_2} +ln\hat{x}_{c_2}} \boxed{+ln\hat{x}_{c_1}+ln\hat{x}_{c_1}+ln\hat{x}_{c_1}+ln\hat{x}_{c_1}}$



Duality for Feedforward-only CC (FFC)

- Theorem 1 cannot be applied to FCC due to
- (1) the recursive structure of the FFC dual encoder
  (2) complex field addition operations in dual encoder

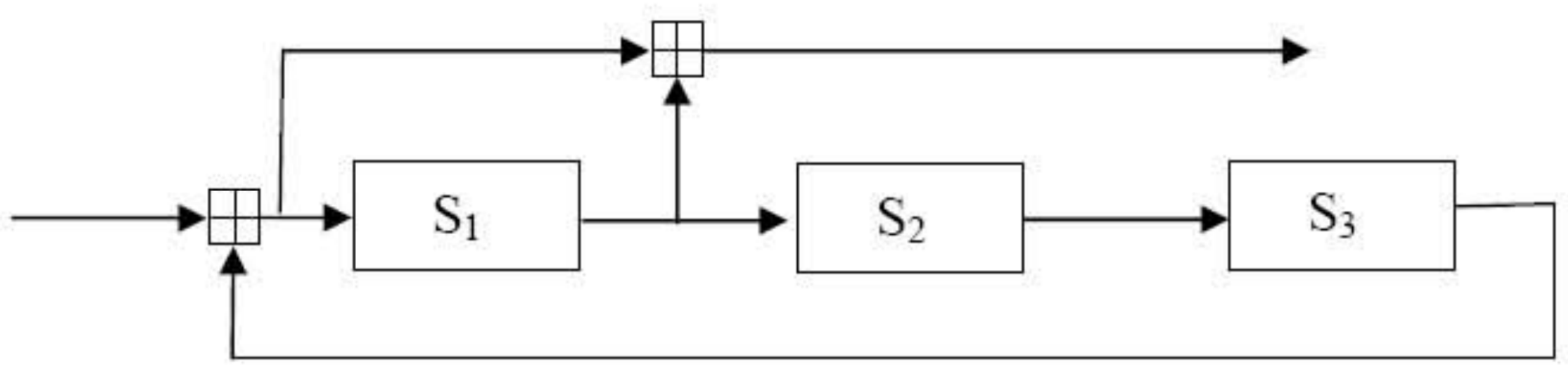
- Additional terms come from the common terms of the shift register contents of dual encoder
-
- Modification of dual encoder structure without changing its generate polynomial is required

$$\frac{1}{a(x)}\boxed{\frac{z(x)}{z(x)}}$$

Duality for Feedforward-only CC (FFC)

› For the above example, z(x)=1+x

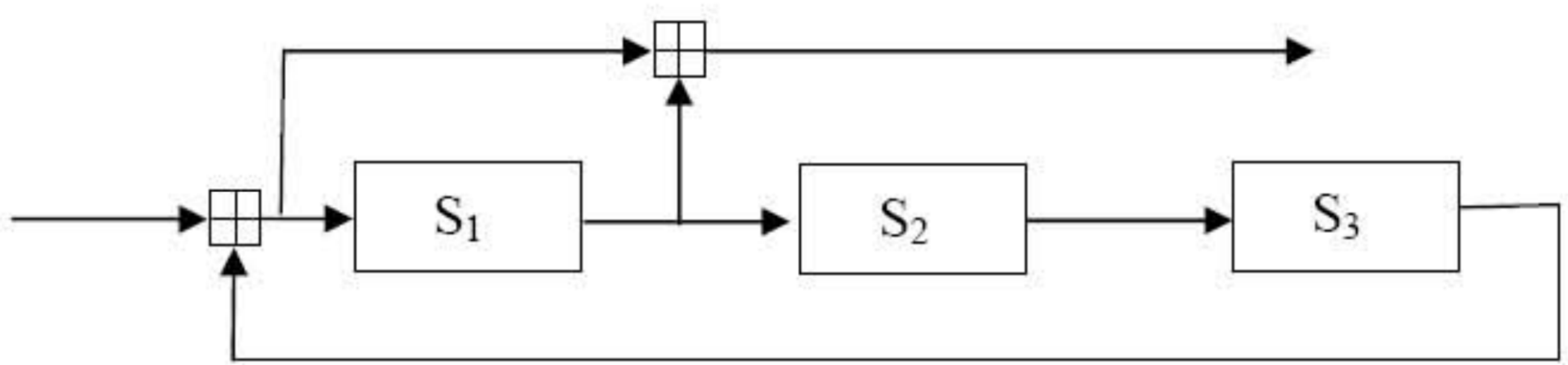
$$\frac{1}{x^2+x+1}\boxed{\frac{x+1}{x+1}} = \frac{x+1}{x^3+1}$$



Duality for Feedforward-only CC (FFC)

› For the above example, z(x)=1+x

$$\frac{1}{x^2+x+1}\boxed{\frac{x+1}{x+1}}=\frac{x+1}{x^3+1}$$



Log soft input $ln\hat{x}_{c_k}$	Memory S_1	Memory S_2	Memory S_3	Log soft output of the modified dual encoder $ln\ddot{x}_{b_k}$	Desired soft decoding output $ln\hat{x}_{b_k}$
$ln\hat{x}_{c_1}$	0	0	0	$ln\hat{x}_{c_1}$	$ln\hat{x}_{c_1}$
$ln\hat{x}_{c_2}$	$ln\hat{x}_{c_1}$	0	0	$ln\hat{x}_{c_2} + ln\hat{x}_{c_1}$	$ln\hat{x}_{c_2} + ln\hat{x}_{c_1}$
$ln\hat{x}_{c_3}$	$ln\hat{x}_{c_2}$	$ln\hat{x}_{c_1}$	0	$ln\hat{x}_{c_3} + ln\hat{x}_{c_2}$	$ln\hat{x}_{c_3} + ln\hat{x}_{c_2}$
$ln\hat{x}_{c_4}$	$ln\hat{x}_{c_3}$	$ln\hat{x}_{c_2}$	$ln\hat{x}_{c_1}$	$ln\hat{x}_{c_4}+ln\hat{x}_{c_3} + ln\hat{x}_{c_1}$	$ln\hat{x}_{c_4}+ln\hat{x}_{c_3} + ln\hat{x}_{c_1}$
$ln\hat{x}_{c_5}$	$ln\hat{x}_{c_4}+ln\hat{x}_{c_1}$	$ln\hat{x}_{c_3}$	$ln\hat{x}_{c_2}$	$ln\hat{x}_{c_5}+ln\hat{x}_{c_4} + ln\hat{x}_{c_2}+ln\hat{x}_{c_1}$	$ln\hat{x}_{c_5}+ln\hat{x}_{c_4} + ln\hat{x}_{c_2}+ln\hat{x}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

Duality for Feedforward-only CC (FFC)

› We first define *a minimum complementary polynomial* of $a(x) = x^n + a_{n-1}x^{n-1} + \cdots + a_1 x_1 + 1$ as the polynomial of the smallest degree

$$z(x) = x^l + z_{l-1}x^{l-1} + \cdots + z_1 x + 1$$

such that

$$a(x)z(x) = x^{n+l} + 1$$

Duality for Feedforward-only CC (FFC)

- **Theorem 2** - For a FFC code, generated by a generator polynomial
- $$g_{FFC}(x) = a(x),$$
- let z(x) represent its minimum complementary polynomial of degree l. The log-domain SISO forward decoding of the FFC code can be implemented by its dual encoder with the generator polynomial of

$$q_{FFC}(x) = \frac{z(x)}{a(x)z(x)} = \frac{z(x)}{x^{n+l}+1}$$

$$= \frac{x^l + z_{l-1}x^{l-1} + \cdots + z_1 x + 1}{x^{n+l}+1}$$

Duality for General Recursive CC (GC)

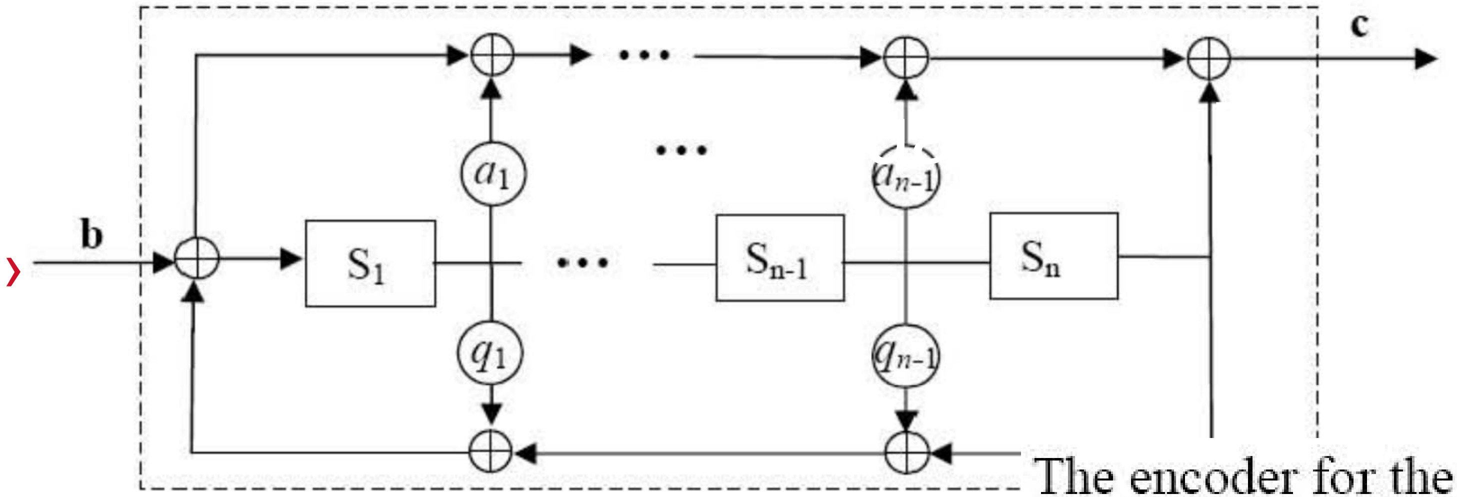
- **Corollary 1**- For a GC code, generated by a generator polynomial

$$g_{GC}(x) = \frac{a(x)}{g(x)} = \frac{x^n + \cdots + a_1 x + 1}{x^n + \cdots + g_1 x + 1}$$

- let z(x) represent the minimum complementary polynomial of *a*(*x*). The log-domain SISO forward decoding of the GC code can be implemented by its dual encoder with the generator polynomial of

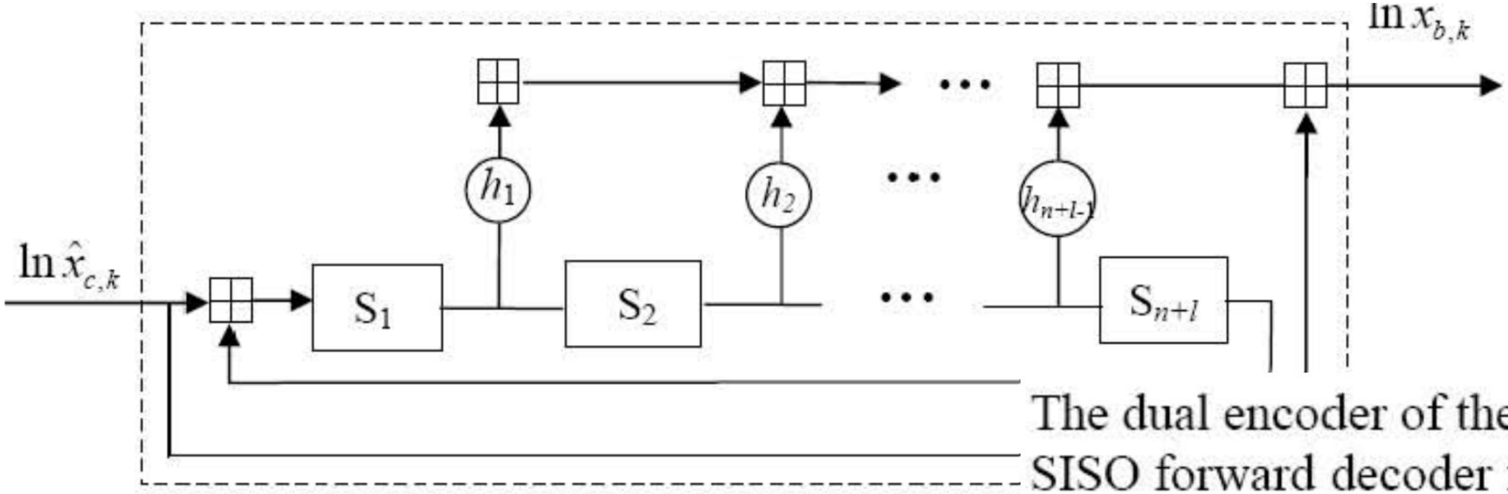
$$q_{GC}(x) = \frac{g(x)z(x)}{a(x)z(x)} = \frac{g(x)z(x)}{x^{n+l}+1}$$

$$= 1 + \frac{h_{n+l-1}x^{n+l-1} + \cdots + h_1 x}{x^{n+l}+1}$$



The encoder for the code generated by

$$g_{GC}(x) = \frac{a(x)}{q(x)} = \frac{x^n + \cdots a_1 x + 1}{x^n + \cdots q_1 x + 1}$$



The dual encoder of the log-domain SISO forward decoder for the code $g_{GC}(x)$, given by

$$q_{GC}(x) = \frac{q(x)z(x)}{x^{n+l}+1} = 1 + \frac{h_{n+l-1}x^{n+l-1} + \cdots h_1 x}{x^{n+l}+1}$$

Outline

- o Introduction to Convolutional Codes
- o Duality Property for MAP Forward Decoding
- o Duality Property for MAP Backward Decoding
- o Duality Property for Bidirectional MAP Decoding
- o Conclusions

Reverse Memory Labeling Encoder

› The encoder of reverse memory labeling of the code

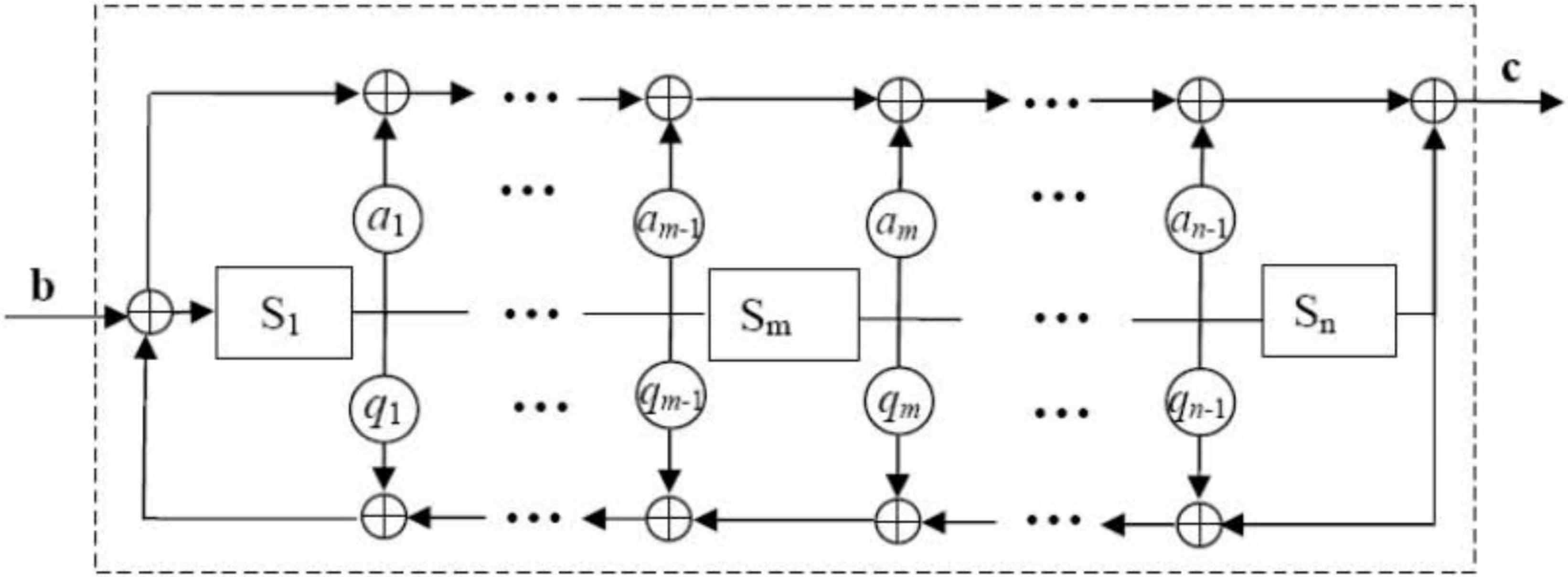
$$g(x) = \frac{a(x)}{q(x)} = \frac{x^n + \cdots + a_1 x + 1}{x^n + \cdots + q_1 x + 1}$$

is obtained by changing

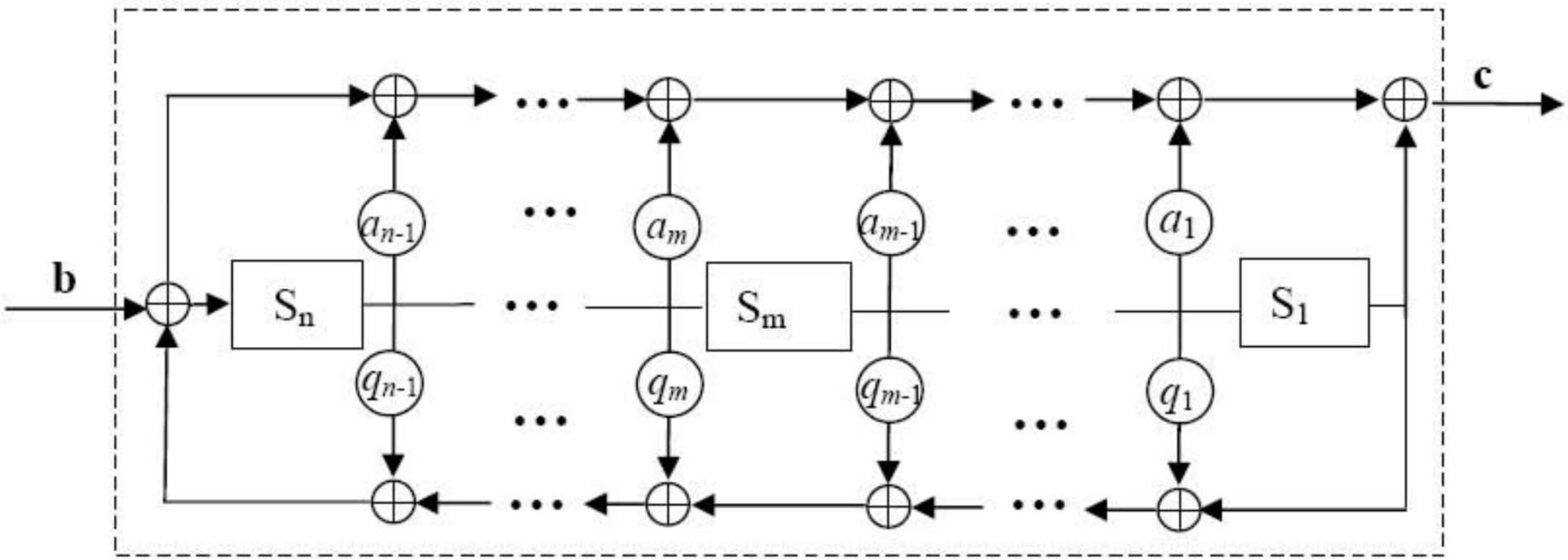
(1) the labeling of the k-th encoder shift register of g(x) from S_k to S_{n+1-k},

(2) the feedforward encoder coefficient from a_k to a_{n-k}, k=1,…,n

(3) The feedback encoder coefficients from q_k to q_{n-k}, k=1,2,…,n.



(a) The encoder of $g(x) = a(x)/q(x)$



(b) The encoder of $g(x) = a(x)/q(x)$ with reverse memory labeling

Backward Decoding Duality

› **Theorem 3** - For a GC code, generated by a generator polynomial

$$g_{GC}(x) = \frac{a(x)}{g(x)} = \frac{x^n + \cdots + a_1 x + 1}{x^n + \cdots + g_1 x + 1}$$

› let z(x) represent the minimum complementary polynomial of *a*(*x*). The log-domain SISO backward decoding of the GC code can be implemented by its dual encoder with **reverse memory labeling** and the generator polynomial of

$$q_{GC}(x) = \frac{g(x)z(x)}{a(x)z(x)} = \frac{g(x)z(x)}{x^{n+l}+1}$$

$$= 1 + \frac{h_{n+l-1}x^{n+l-1} + \cdots + h_1 x}{x^{n+l}+1}$$

Outline

- o Introduction to Convolutional Codes
- o Duality Property for MAP Forward Decoding
- o Duality Property for MAP Backward Decoding
- o Duality Property for Bidirectional MAP Decoding
- o Conclusions

Duality for Bidirectional MAP Decoding

- Define
- $\overrightarrow{\widehat{x}}_{b_k}$: soft outputs of the forward dual encoder
-
- $\overleftarrow{\widehat{x}}_{b_k}$: soft outputs of the backward dual encoder
-
- $\widehat{x}_{b_k}$: BCJR bidirectional MAP decoder output

- Then we can represent $\widehat{x}_{b_k}$ as

$$\widehat{x}_{b_k} = \overrightarrow{\omega_k}\,\overrightarrow{\hat{x}}_{b_k} + \overleftarrow{\omega_k}\,\overleftarrow{\hat{x}}_{b_k}$$

Duality for Bidirectional MAP Decoding

$\overrightarrow{\omega_k}$ and $\overleftarrow{\omega_k}$ are the combining coefficients in real domain applied to the forward and backward dual encoder outputs

- They are obtained by comparing the bidirectional BCJR MAP decoding output expressions with the forward and backward dual encoder output expressions
- The combining coefficients for various 4-state and 8-state codes are obtained
- Results are verified by simulations. The dual decoder has the exactly the same performance as the BCJR MAP decoder

Duality for Bidirectional MAP Decoding

› For example, the combining coefficients for the 4 states FCC code $[5]_8$, generated by $g_{FFC}(x) = x^2 + 1$,

$$\overrightarrow{\omega_k} = \overleftarrow{\omega_k} = \begin{cases} (1+E)/4, & \text{for } k = 1,3,5,...,\\ (1+O)/4, & \text{for } k = 2,4,6,.... \end{cases}$$

where

$$O = \prod_{l=1}^{\lceil K/2 \rceil} \hat{x}_{c_{2l-1}}, \quad E = \prod_{l=1}^{\lceil K/2 \rceil} \hat{x}_{c_{2l}}$$

and $\tilde{x}_{c_k}$ is the soft estimation of the received coded symbol c_k

Duality for Bidirectional MAP Decoding

› For example, the combining coefficients for the 4 states FCC code $[5/7]_8$, *generated by* $g_{GC}(x) = \frac{x^2+1}{x^2+x+1}$,

$$\overrightarrow{\omega_k} = \overleftarrow{\omega_k} = \begin{cases} (1 + O/\hat{x}_{c_k}^2)/4, & \text{for } k = 1, 3, 5, \ldots, \\ (1 + E/\hat{x}_{c_k}^2)/4, & \text{for } k = 2, 4, 6, \ldots. \end{cases}$$

where

$$O = \prod_{l=1}^{\lceil K/2 \rceil} \hat{x}_{c_{2l-1}}, \quad E = \prod_{l=1}^{\lceil K/2 \rceil} \hat{x}_{c_{2l}}$$

and $\tilde{x}_{c_k}$ is the soft estimation of the received coded symbol c_k

Outline

- o Introduction to Convolutional Codes
- o Duality Property for MAP Forward Decoding
- o Duality Property for MAP Backward Decoding
- o Duality Property for Bidirectional MAP Decoding
- o Conclusions

Conclusions

- Revisited the BCJR SISO MAP forward and backward decoding process for the rate-1 convolutional codes
- Dual encoder structures of forward and backward decoding for various rate-1 convolutional codes are derived
- Dual decoder can reduce the decoder complexity from exponential to linear computation in terms of the memory order
- Duality property also reveals for the first time the explicit decoder input-output relationship, which can be used to facilitate the performance and decoding convergence analysis

Conclusions

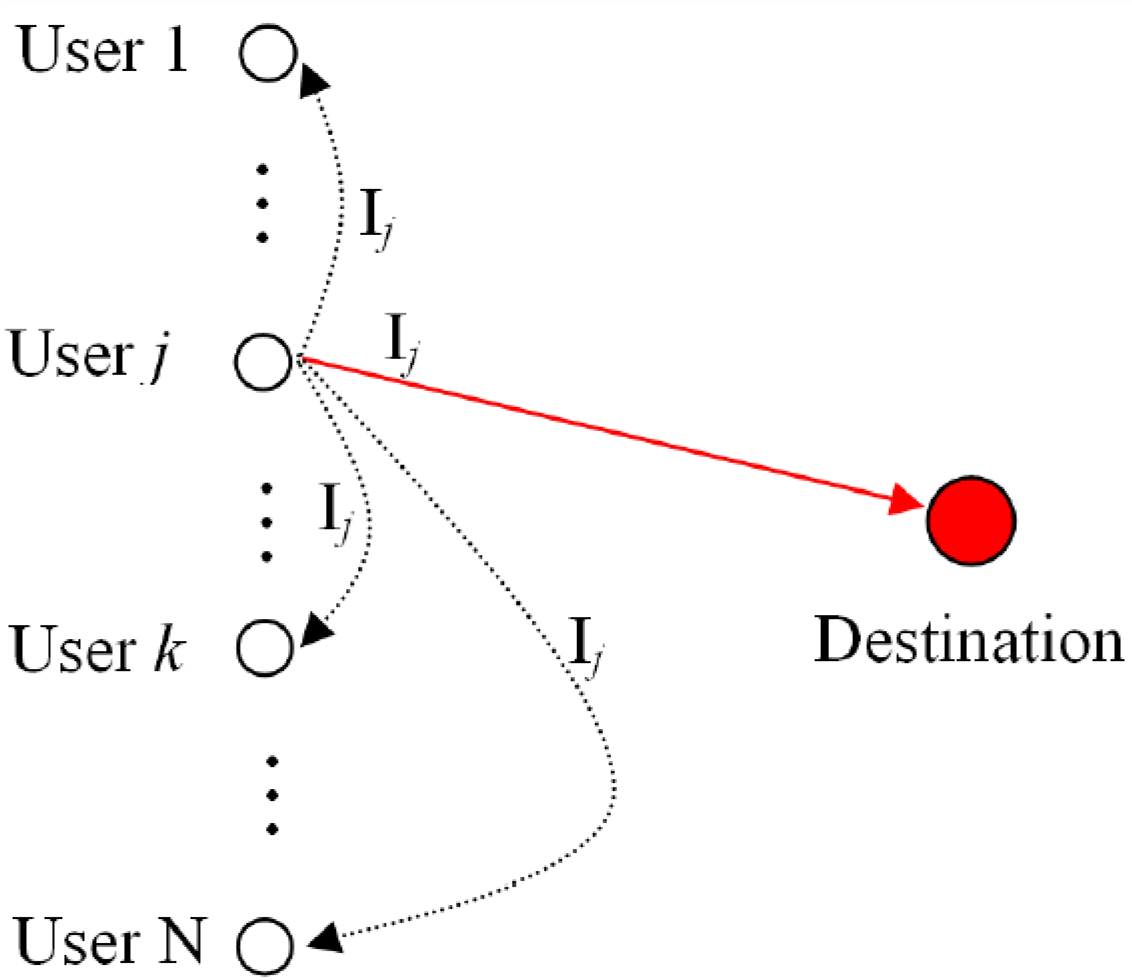
- Rate-1 codes are mainly used as the component codes in concatenated codes
- The ISI channels can also be represented by a rate-1 convolutional encoding process
- Currently working on the duality properties for general convolutional codes and other codes

Outline

o Other Work on Wireless Network Coding

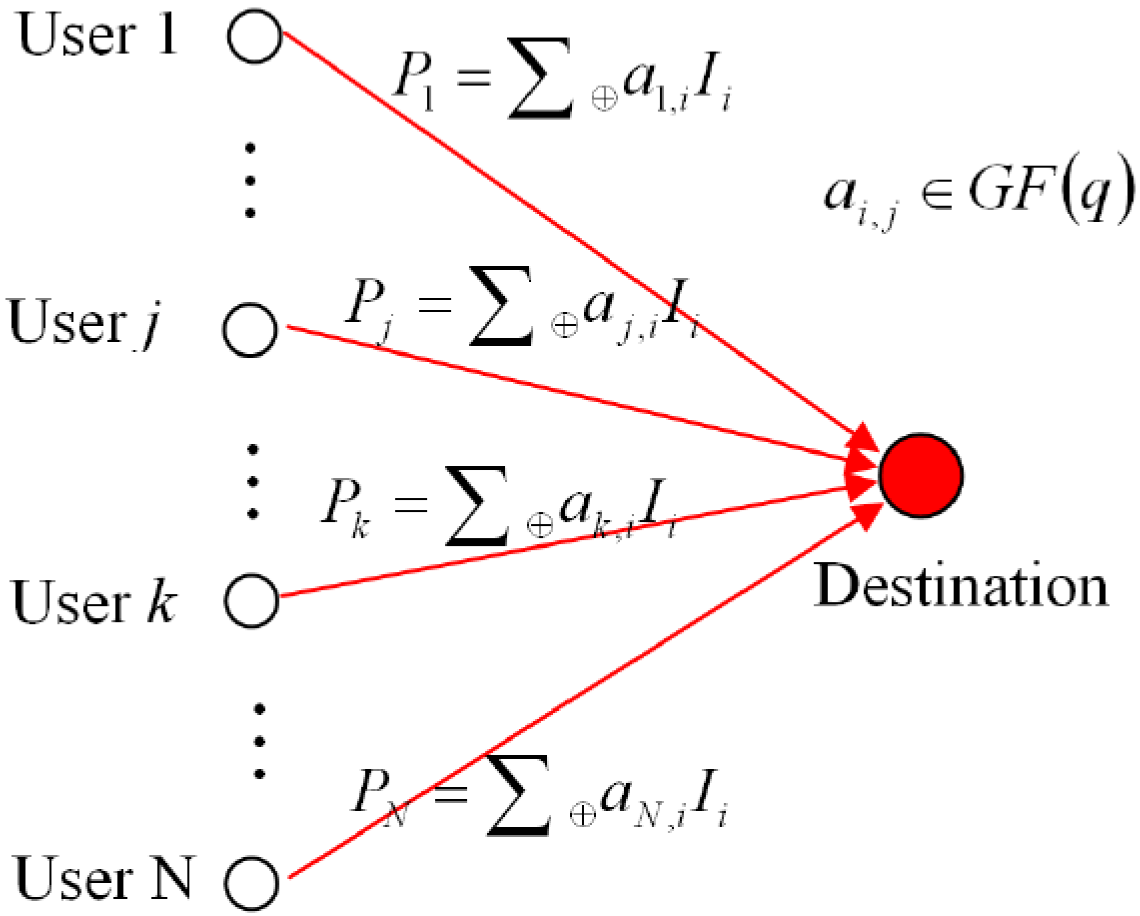
Generalized Distributed Network-Channel Coding

- Designed for multi-user cooperative wireless networks



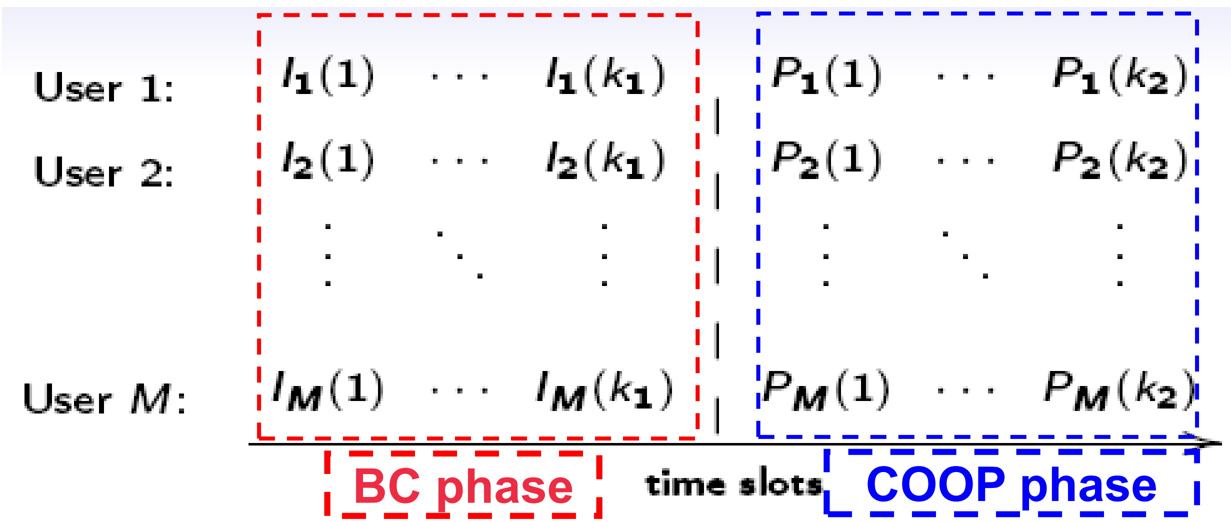
Generalized Distributed Network-Channel Coding

› **How to design the network code to maximize the diversity when taking into account inter-user link failure**



Generalized Distributed Network-Channel Coding

- BC phase: Each user broadcasts k1 independent information packets
- Coop Phase: Each user transmits k2 parity packets consisting of nonbinary linear combinations of its k1 own information packets and the k1(M − 1) partners' information packets (if decoded correctly)



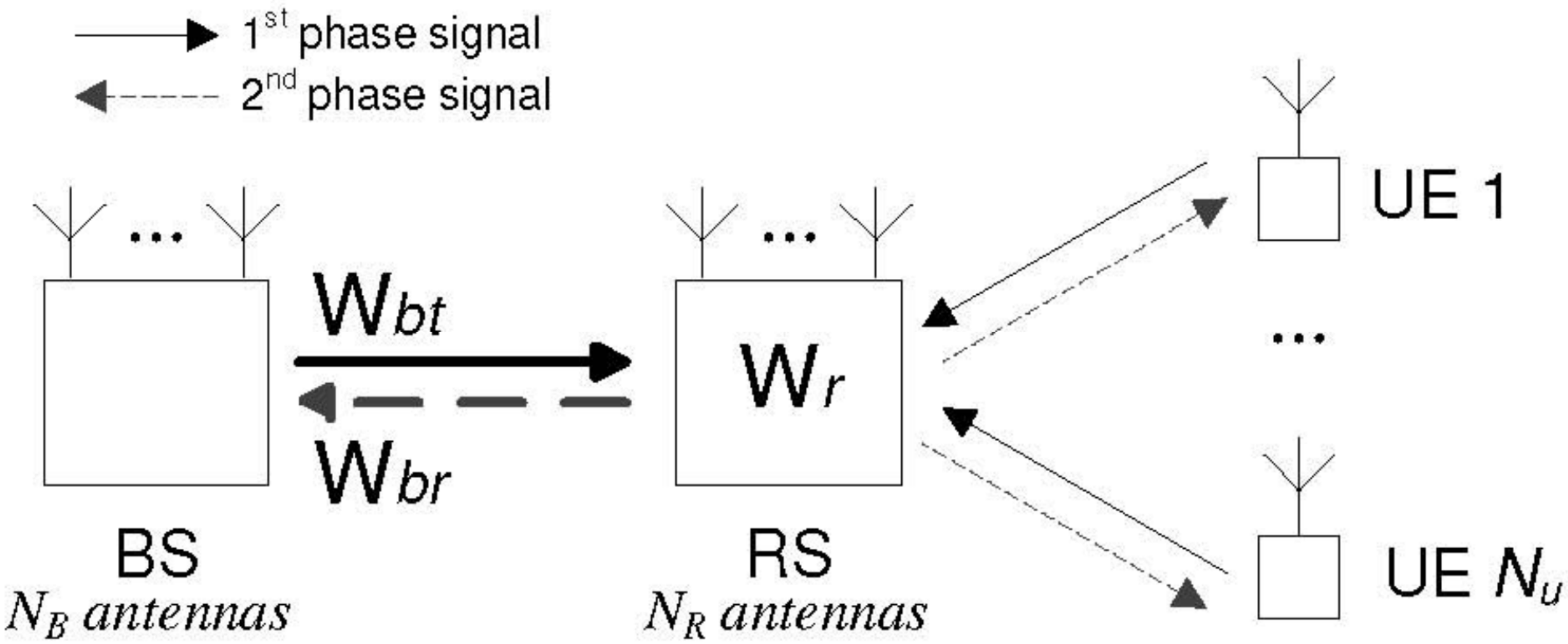
Generalized Distributed Network-Channel Coding

- Represent the network transfer matrix as a generator matrix of a linear block code
- The design of the network codes that maximizes the diversity order is equivalent to the design of linear block codes over a nonbinary finite field under the Hamming metric.
- A maximum distance separable (MDS) block code over a sufficiently large finite field as the network transfer matrix is the necessary condition for full diversity order under link failure model.

Two-Way Cellular Networks

› **BS exchanges information with multiple users via a MIMO relay**

How to design precoding matrix at the BS and relay to maximize the bidirectional sum rate?



Thanks