Circular-Shift Linear Network Coding

Qifu (Tyler) Sun

Jun, 2017 @ INC, CUHK

Hanqi Tang,
Xiaolong Yang, Keping Long

北京科技大学
University of Science and Technology Beijing

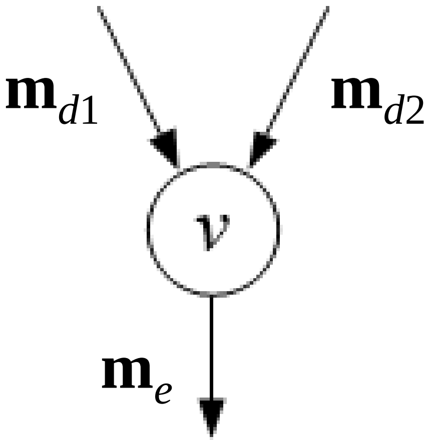
INC

Zongpeng Li

UNIVERSITY OF CALGARY

Different types of LNC: a recap

Every edge transmits a binary sequence $\mathbf{m}_e$ of length L.



$= \mathbf{k}_{d1,e}\mathbf{m}_{d1} + \mathbf{k}_{d2,e}\mathbf{m}_{d2}$

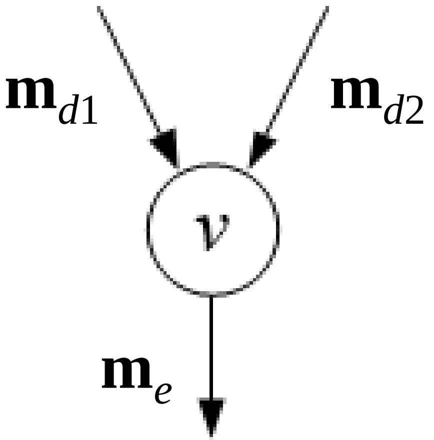
For scalar linear coding:

- $\mathbf{m}_e \in GF(2^L)$
- $\mathbf{m}_e$, $e \in Out(v)$, is determined by a linear function over $GF(2^L)$, i.e.,
- Local encoding kernels $\in GF(2^L)$
- Global encoding kernels $\in GF(2^L)^\omega$

  // ω: total no. source binary sequences

Different types of LNC: a recap

Every edge transmits a binary sequence $\mathbf{m}_e$ of length L.



$= \mathbf{m}_{d1}\mathbf{K}_{d1,e} + \mathbf{m}_{d2}\mathbf{K}_{d2,e}$

For vector linear coding:

- $\mathbf{m}_e \in \mathrm{GF}(2)^L$
- $\mathbf{m}_e$, $e \in \mathrm{Out}(v)$, is determined by L different linear functions over GF(2), i.e.,
- Local encoding kernels $\in \mathrm{GF}(2)^{L\times L}$
- Global encoding kernels $\in (\mathrm{GF}(2)^{L\times L})^{\omega}$

Reduce LNC implementation complexity

There have been continuous attempts to design LNC schemes with low implementation complexities.

- 1st straightforward approach: reduce block length L.
  - [1, 2] Vector LNC may yield solutions with lower implementation complexities compared with scalar LNC.

[1] Q. T. Sun et. al., "On vector linear solvability of multicast networks," *IEEE Trans. Comm.*, Dec. 2016.

[2] T. Etzion, A. Wachter-Zeh, "Vector network coding based on subspace codes outperforms scalar linear network coding," *IEEE ISIT*, 2016.

Reduce LNC implementation complexity

There have been continuous attempts to design LNC schemes with low implementation complexities.

- 2nd approach: choose appropriate LEKs
  - Ref. [3] studied permutation-based LNC: vector LNC with LEKs chosen from permutation matrices.

[3] S. Jaggi, Y. Cassuto, M. Effros, "Low complexity encoding for network codes," *IEEE ISIT*, 2006

When $L \to \infty$, randomly constructed permutation-based LNC schemes can asymptotically approach the multicast capacity.

From permutation to circular-shifts

- When block length L is long, even permutation operations on the binary sequences may not have computational complexity as low as desired for real-world implementation.
- A natural further reduction is to choose circular-shift operations.
  - lower computational complexity;
  - amenable to implementation through atomic hardware operations.

Previous study of circular-shift LNC

- There have been considerations of adopting circular-shifts (& bitwise addition) for LNC encoding [4-6].
  - [4] focuses on (n, 2)-Combination Network, and constructs a linear solution involving circular-shift and bit truncation.
  - [5] shows the *existence* of an (L–1, L)-fractional circular-shift (*rotation-and-add*) linear solution for every multicast network.
  - [6] shows the *existence* of circular-shift-based regenerating codes.

[4] M. Xiao, M. Medard, T. Aulin, “A binary coding approach for combination networks and general erasure networks,” *IEEE ISIT*, 2007

[5] A. Keshavarz-Haddad, M. A. Khojastepour, “Rotate-and-add coding: a novel algebraic network coding scheme,” *IEEE ITW*, 2010

[6] H. Hou, K. W. Shum, M. Chen, H. Li, “BASIC codes: low-complexity regenerating codes for distributed storage systems,” *IEEE Trans. Inf. Theory*, 2016.

Previous study of circular-shift LNC

- There have been considerations of adopting circular-shifts (& bitwise addition) for LNC encoding [4-6].
  - [4] focuses on (n, 2)-Combination Network, and constructs a linear solution involving circular-shift and bit truncation.
  - [5, 6] from the perspective of cyclic convolutional coding

Due to lack of a systematic model

How to efficiently construct is unknown

[4] M. Xiao, M. Medard, T. Aulin, "A binary coding approach for combination networks and general erasure networks," *IEEE ISIT*, 2007

[5] A. Keshavarz-Haddad, M. A. Khojastepour, "Rotate-and-add coding: a novel algebraic network coding scheme," *IEEE ITW*, 2010

[6] H. Hou, K. W. Shum, M. Chen, H. Li, "BASIC codes: low-complexity regenerating codes for distributed storage systems," *IEEE Trans. Inf. Theory*, 2016.

Highlight of this talk

- Algebraically formulate circular-shift LNC as a special type of vector LNC.
- Establish an intrinsic connection between scalar LNC and circular-shift LNC for a general network.
- Efficiently construct an $(L-1, L)$-fractional circular-shift linear solution for some L on multicast networks.
- Insufficient to achieve the exact multicast capacity.

Algebraic formulation of circular-shift LNC

- Let $\mathbf{C}_L$ denote the *cyclic permutation matrix* of size L (over GF(2))

$$\mathbf{C}_L = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \ddots & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & 0 & 1 \\ 1 & 0 & \cdots & 0 & 0 \end{bmatrix}$$

$$\left(m_L, m_{L-1}, \ \ldots, m_1\right) \cdot \mathbf{C}_L^j = \left(m_j, \ldots, m_1, m_L, \ \ldots, m_{j+1}\right)$$

Lemma. Let α be a primitive L^{th} root of unity, where L is odd.

$$\mathbf{C}_L^j = \mathbf{V}_L \cdot \Lambda_\alpha^j \cdot \mathbf{V}_L^{-1} \qquad \forall j \geq 0$$

$$\mathbf{V}_L = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ 1 & \alpha & \ldots & \alpha^{L-1} \\ \vdots & \vdots & \ldots & \vdots \\ 1 & \alpha^{L-1} & \ldots & \alpha^{(L-1)(L-1)} \end{bmatrix} \qquad \mathbf{\Lambda}_\alpha = \begin{bmatrix} 1 & 0 & \ldots & 0 \\ 0 & \alpha & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \ldots & 0 & \alpha^{L-1} \end{bmatrix}$$

Algebraic formulation of circular-shift LNC

- Let $\mathbf{C}_L$ denote the *cyclic permutation matrix* of size L (over GF(2))

$$\mathbf{C}_L = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \ddots & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & 0 & 1 \\ 1 & 0 & \cdots & 0 & 0 \end{bmatrix}$$

$$\left(m_L, m_{L-1}, \ \ldots, m_1\right) \cdot \mathbf{C}_L^j$$
$$= \left(m_j, \ldots, m_1, m_L, \ \ldots, m_{j+1}\right)$$

- For $1 \le \delta \le L$, define $\mathcal{C}_\delta$ as

$$\mathcal{C}_\delta = \left\{ \sum_{j=0}^{L-1} a_j \mathrm{C}_L^j : a_j \in \{0,1\}, \sum_{j=0}^{L-1} a_j \le \delta \right\}$$

$$\boxed{\mathcal{C}_1 \subset \mathcal{C}_2 \subset \ldots \subset \mathcal{C}_L}$$

// matrices that are summation at most δ cyclic-permutation matrices

Algebraic formulation of circular-shift LNC

Definition. An L-dimensional *circular-shift linear code of order δ* is an L-dimensional vector linear code with LEKs selected from $\mathcal{C}_\delta$.

Remarks.
$\mathcal{C}_L$ forms a *commutative subring* of the (non-commutative) ring of $L \times L$ binary matrices.

- For $1 \le \delta \le L$, define $\mathcal{C}_\delta$ as

$$\mathcal{C}_\delta = \left\{ \sum_{j=0}^{L-1} a_j \mathbf{C}_L^j : a_j \in \{0,1\}, \sum_{j=0}^{L-1} a_j \le \delta \right\} \quad \boxed{\mathcal{C}_1 \subset \mathcal{C}_2 \subset \ldots \subset \mathcal{C}_L}$$

// matrices that are summation at most δ cyclic-permutation matrices

Algebraic formulation of circular-shift LNC

Definition. An L-dimensional *circular-shift linear code of order* δ is an L-dimensional vector linear code with LEKs selected from $\mathcal{C}_\delta$.

Remarks.

$\mathcal{C}_L$ forms a *commutative subring* of the (non-commutative) ring of $L \times L$ binary matrices.

- Circular-shift LNC conforms to the assumption in the algebraic framework of vector LNC in [7].
- In the context of [8], a circular-shift linear code of order L can be regarded as a linear code over the $\mathcal{C}_L$-module $GF(2)^L$.

[7] J. B. Ebrahimi, C. Fragouli, "Algebraic algorithm for vecor network coding" *IEEE Trans. Inf. Theory*, 2011.
[8] J. Connelly, K. Zeger, "Linear network coding over rings part II: vector codes and non-commutative alphabets," *IEEE Trans. Inf. Theory*, 2017.

Algebraic formulation of circular-shift LNC

Definition. An L-dimensional *circular-shift linear code of order* δ is an L-dimensional vector linear code with LEKs selected from $\mathcal{C}_\delta$.

Remarks.

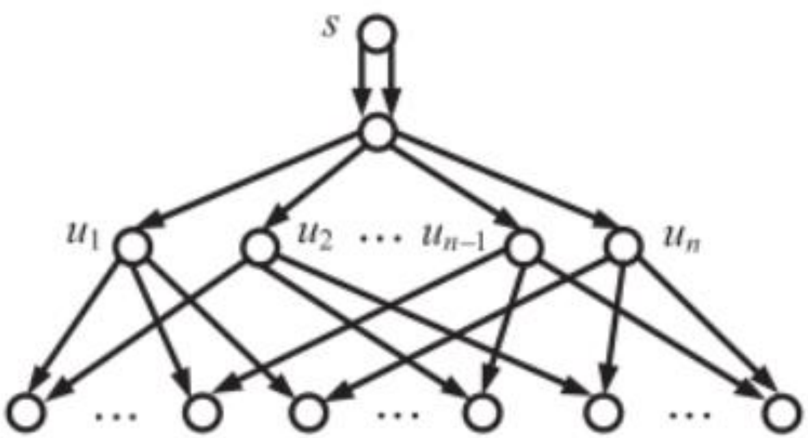
The *rotate-and-add linear code* in [5], can be regarded as a *circular-shift linear code of order 1* without **0** as LEK.

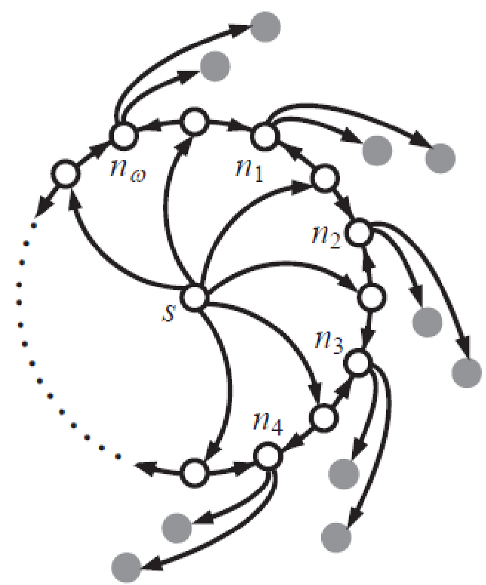
[5] A. Keshavarz-Haddad, M. A. Khojastepour, "Rotate-and-add coding: a novel algebraic network coding scheme," *IEEE ITW*, 2010.

Insufficiency of circular-shift LNC

Definition. An L-dimensional *circular-shift linear code of order* δ is an L-dimensional vector linear code with LEKs selected from $\mathcal{C}_\delta$.

Proposition. Both the (n, 2)-Combination Network ($n \geq 4$) and the Swirl Network [9] with parameter $\omega \geq 4$ are *not* circular-shift linearly solvable of order L for any $L \geq 1$.





[9] Q. T. Sun et. al., "Multicast network coding and field sizes," *IEEE Trans. Inf. Theory*, 2015

Insufficiency of circular-shift LNC

Definition. An L-dimensional *circular-shift linear code of order* δ is an L-dimensional vector linear code with LEKs selected from $\mathcal{C}_\delta$.

Proposition. Both the $(n, 2)$-Combination Network ($n \geq 4$) and the Swirl Network [9] with parameter $\omega \geq 4$ are *not* circular-shift linearly solvable of order L for any $L \geq 1$. They do not have permutation-based linear solutions for any $L \geq 1$ either.

- Circular-shift LNC and permutation-based LNC are insufficient to achieve the exact multicast capacity.
- The best to expect for circular-shift LNC is 1-bit redundancy per edge transmission. It is feasible & can be efficiently constructed!

Need review the concept of *fractional LNC.*

Fractional LNC (on multicast networks)

In an (L', L)-fractional linear code (over GF(2), $L' \leq L$)

- Every edge transmits a binary sequence of length L.
- The LEKs are selected from $\mathrm{GF}(2)^{L\times L}$.

same as vector code

- The ω binary sequences $\mathbf{m}_1', \mathbf{m}_2', \ldots, \mathbf{m}_\omega'$ generated at s are of length L'.
- The source needs an $\omega L' \times \omega L$ matrix $\mathbf{G}_s$ to generate the ω binary sequences $\mathbf{m}_e$ of length L for Out(s).

$$[\mathbf{m}_e]_{e\in \mathrm{Out}(s)} = [\mathbf{m}_j']_{1\leq j\leq \omega} \cdot \mathbf{G}_s$$

additional settings at source s

// L-dimensional vector linear codes are (L, L)-fractional linear codes with $\mathbf{G}_s = \mathbf{I}_{\omega L}$.

Fractional LNC (on multicast networks)

In an (L', L)-fractional linear code (over GF(2), $L' \le L$)

- Every edge transmits a binary sequence of length L.
- The LEKs are selected from $\mathrm{GF}(2)^{L\times L}$.

same as vector code

- The ω binary sequences $\mathbf{m}_1', \mathbf{m}_2', \ldots, \mathbf{m}_\omega'$ generated at s are of length L'.
- The source needs an $\omega L' \times \omega L$ matrix $\mathbf{G}_s$ to generate the ω binary sequences $\mathbf{m}_e$ of length L for Out(s).

$$[\mathbf{m}_e]_{e\in \mathrm{Out}(s)} = [\mathbf{m}_j']_{1\le j\le \omega} \cdot \mathbf{G}_s$$

additional settings at source s

Definition. An (L', L) circular-shift linear code of order δ is an (L', L)-fractional linear code with LEKs chosen from $\mathcal{C}_\delta$.

Construction of (L–1, L) circular-shift linear solutions

- *L: a prime with primitive root 2.*
- α: a primitive L^{th} root of unity.

Lemma. For each element $k \in \mathrm{GF}(2^{L-1})$, there is a *unique* polynomial over GF(2) $g(x) = a_{L-1}x^{L-1} + \ldots + a_1x^1 + a_0$ s.t.

(∗) $k = g(\alpha)$, and it has at most $(L-1)/2$ nonzero coefficients.

Theorem. Consider an *arbitrary* scalar linear solution over $\mathrm{GF}(2^{L-1})$ with LEKs $g_{d,e}(\alpha)$ and decoding matrix $\mathbf{D}_t(\alpha)$ for receiver t.

Define an $(L-1, L)$-fractional linear code (over GF(2)):

- Out(s) transmits $[0\ \mathbf{m}_1']$, …, $[0\ \mathbf{m}_\omega']$ // $\mathbf{G}_s = \mathbf{I}_\omega \otimes [\mathbf{0}\ \mathbf{I}_{L-1}]$
- LEKs $\mathbf{K}_{d,e} = g_{d,e}(\mathbf{C}_L) \in \mathcal{C}_{(L-1)/2}$

This code is a circular-shift linear solution of order $(L-1)/2$.

The decoding matrix for t is $\mathbf{D}_t(\mathbf{C}_L)\cdot(\mathbf{I}_\omega \otimes \hat{\mathbf{I}}_L)$ // $\hat{\mathbf{I}}_L = \begin{bmatrix} 1 \ldots 1 \\ \mathbf{I}_{L-1} \end{bmatrix}$

Construction of (L–1, L) circular-shift linear solutions

Remarks.

- The mapping from $k_{d,e} \in \mathrm{GF}(2^{L-1})$ to $\mathbf{K}_{d,e} \in \mathcal{C}_{(L-1)/2}$ *is one-to-one correspondence*. However, it is *not* an isomorphism.
- The theorem holds for general networks as well.

Theorem. Consider an *arbitrary* scalar linear solution over $\mathrm{GF}(2^{L-1})$ with LEKs $g_{d,e}(\alpha)$ and decoding matrix $\mathbf{D}_t(\alpha)$ for receiver t.

Define an $(L-1, L)$-fractional linear code (over GF(2)):

- Out(s) transmits $[0\ \mathbf{m}_1'], \ldots, [0\ \mathbf{m}_\omega']$ // $\mathbf{G}_s = \mathbf{I}_\omega \otimes [\mathbf{0}\ \mathbf{I}_{L-1}]$
- LEKs $\mathbf{K}_{d,e} = g_{d,e}(\mathbf{C}_L) \in \mathcal{C}_{(L-1)/2}$

This code is a circular-shift linear solution of order $(L-1)/2$.

The decoding matrix for t is $\mathbf{D}_t(\mathbf{C}_L)\cdot(\mathbf{I}_\omega \otimes \hat{\mathbf{I}}_L)$ // $\hat{\mathbf{I}}_L = \begin{bmatrix} 1\ldots1 \\ \mathbf{I}_{L-1} \end{bmatrix}$

Construction of (L–1, L) circular-shift linear solutions

Proof Key. $\mathbf{C}_L^j = \mathbf{V}_L \cdot \Lambda_\alpha^j \cdot \mathbf{V}_L^{-1} \quad \forall j \geq 0$

$$\mathbf{\Lambda}_\alpha = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & \alpha & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \alpha^{L-1} \end{bmatrix}$$

If $(g_{d,e}(\alpha))$ is a scalar linear solution, then $(g_{d,e}(\alpha^j))$ is a scalar linear solution $\forall\ 1 \leq j \leq L$.

Theorem. Consider an *arbitrary* scalar linear solution over GF(2^{L-1}) with LEKs $g_{d,e}(\alpha)$ and decoding matrix $\mathbf{D}_t(\alpha)$ for receiver t.

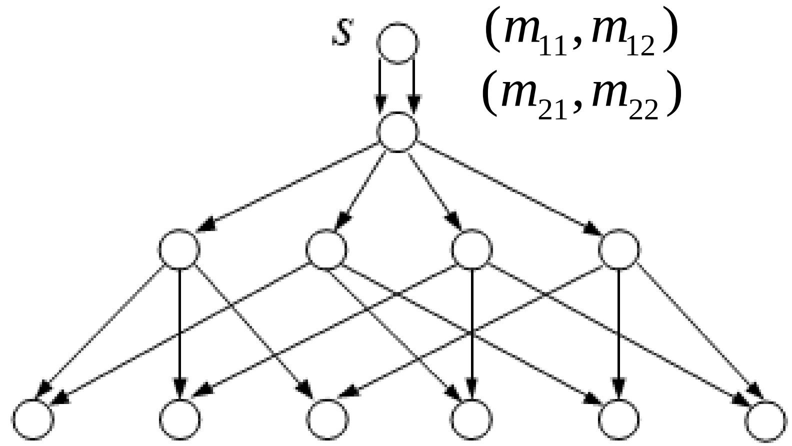
Define an ($L-1$, L)-fractional linear code (over GF(2)):

- Out(s) transmits $[0\ \mathbf{m}_1']$, …, $[0\ \mathbf{m}_\omega']$ // $\mathbf{G}_s = \mathbf{I}_\omega \otimes [\mathbf{0}\ \mathbf{I}_{L-1}]$
- LEKs $\mathbf{K}_{d,e} = g_{d,e}(\mathbf{C}_L) \in \mathcal{C}_{(L-1)/2}$

This code is a circular-shift linear solution of order (L–1)/2.

The decoding matrix for t is $\mathbf{D}_t(\mathbf{C}_L)\cdot(\mathbf{I}_\omega \otimes \hat{\mathbf{I}}_L)$ // $\hat{\mathbf{I}}_L = \begin{bmatrix} 1 \dots 1 \\ \mathbf{I}_{L-1} \end{bmatrix}$

Example



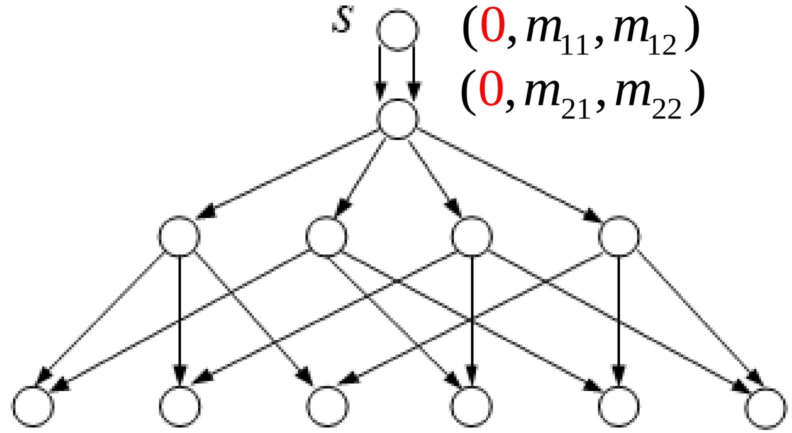
Let $L = 3$, α be a primitive 3rd root of unity.

Given a scalar linear solution over GF(2^2) w/ GEKs $\begin{pmatrix} 0 \\ 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ \alpha \end{pmatrix} \begin{pmatrix} 1 \\ \alpha^2 \end{pmatrix}$

Decoding matrix for the rightmost receiver: $\begin{bmatrix} \alpha^2 & 1 \\ \alpha & 1 \end{bmatrix}$

$$// \quad \begin{bmatrix} 1 & 1 \\ \alpha & \alpha^2 \end{bmatrix} \begin{bmatrix} \alpha^2 & 1 \\ \alpha & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Example



Let $L = 3$, α be a primitive 3rd root of unity.

Given a scalar linear solution over GF(2^2) w/ GEKs $\begin{pmatrix} 0 \\ 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ \alpha \end{pmatrix} \begin{pmatrix} 1 \\ \alpha^2 \end{pmatrix}$

Decoding matrix for the rightmost receiver: $\begin{bmatrix} \alpha^2 & 1 \\ \alpha & 1 \end{bmatrix}$

Establish a (2, 3)-fractional linear code w/ GEKs $\begin{pmatrix} \mathbf{0} \\ \mathbf{I}_3 \end{pmatrix} \begin{pmatrix} \mathbf{I}_3 \\ \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{I}_3 \\ \mathbf{C}_3 \end{pmatrix} \begin{pmatrix} \mathbf{I}_3 \\ \mathbf{C}_3^2 \end{pmatrix}$

For the rightmost receiver: $(0, m_{11}, m_{12}, 0, m_{21}, m_{22})\begin{pmatrix} \mathbf{I}_3 \\ \mathbf{C}_3 \end{pmatrix} = (m_{22}, m_{11}, m_{12} + m_{21}),$

$$(0, m_{11}, m_{12}, 0, m_{21}, m_{22})\begin{pmatrix} \mathbf{I}_3 \\ \mathbf{C}_3^2 \end{pmatrix} = (m_{21}, m_{11} + m_{22}, m_{12})$$

$$(m_{22}, m_{11}, m_{12} + m_{21}, m_{21}, m_{11} + m_{22}, m_{12})\begin{bmatrix} \mathbf{C}_3^2 & \mathbf{I}_3 \\ \mathbf{C}_3 & \mathbf{I}_3 \end{bmatrix}(\mathbf{I}_2 \otimes \tilde{\mathbf{I}}_3) = (m_{11}, m_{12}, m_{21}, m_{22})$$

Efficient construction of circular-shift LNC

- When $2^{L-1} \geq |T|$, as a scalar linear solution over GF(2^{L-1}) can be efficiently constructed, an (L–1, L) circular-shift linear solution of order (L–1)/2 can also be efficiently constructed.
- For an arbitrary subset F of GF(2^{L-1}), as long as $|F| \geq |T|$, a scalar linear solution with LEKs selected from F can be efficiently constructed.
- For any $1 \leq \delta \leq (L-1)/2$, as long as $\binom{L}{0} + \binom{L}{1} + \ldots + \binom{L}{\delta} \geq |T|$ an (L–1, L) circular-shift linear solution of order δ can be obtained by efficiently constructing a scalar linear code over GF(2^{L-1}) with LEKs selected from

$$F = \{a_{L-1}\alpha^{L-1} + \ldots + a_1\alpha^1 + a_0 : \text{at most } \delta \text{ nonzero coefficients } a_j\}$$

Complexity comparison

- Theoretically compare the encoding and decoding complexity between circular-shift LNC and scalar LNC.
- Same as in [6], ignore the complexity of computing $\mathbf{m}_d\mathbf{C}_L^j$ (can be software implemented by modifying the pointer to the starting address in the sequence).
  - L binary operations for $\mathbf{m}_d(\mathbf{C}_L^j+\mathbf{C}_L^i)$
- For an $(L-1, L)$ circular-shift linear solution of degree δ:
  - Encoding: $L(\delta|\mathrm{In}(v)|-1)$ binary operations for $\mathbf{m}_e = \sum_{d\in\mathrm{In}(v)}\mathbf{m}_d\mathbf{K}_{d,e}$
  - Decoding: $\omega^2L(L-1)/2$ binary operations // decoding matrix is $\mathbf{D}_t(\mathbf{C}_L)\cdot(\mathbf{I}_\omega \otimes \hat{\mathbf{I}}_L)$

Each block entry in $\mathbf{D}_t(\mathbf{C}_L) \in \mathcal{C}_{(L-1)/2}$

[6] H. Hou, K. W. Shum, M. Chen, H. Li, "BASIC codes: low-complexity regenerating codes for distributed storage systems," *IEEE Trans. Inf. Theory*, 2016.

Complexity comparison

- $|T| \leq L < 2^m$, m+1, L+1 are primes with primitive root 2.

Number of Binary Operations per Source Information Bit

	Encoding	Decoding
Scalar over GF(2^m)	$> 2\eta m$	$> \omega(2m+1)$
$(m, m+1)$ circular-shift of degree $\frac{m}{2}$	$\frac{1}{2}\eta m$	$\frac{1}{2}\omega(m+1)$
$(L, L+1)$ circular-shift of degree 1	$\eta - 1$	$\frac{1}{2}\omega(L+1) < \frac{1}{2}\omega 2^m$

Complexity comparison

- $|T| \le L < 2^m$, m+1, L+1 are primes with primitive root 2.

Number of Binary Operations per Source Information Bit

	Encoding	Decoding
Scalar over GF(2^m)	$> 2\eta m$	$> \omega(2m+1)$
$(m, m+1)$ circular-shift of degree $\frac{m}{2}$	$\frac{1}{2}\eta m$	$\frac{1}{2}\omega(m+1)$
$(L, L+1)$ circular-shift of degree 1	$\eta - 1$	$\frac{1}{2}\omega(L+1) < \frac{1}{2}\omega 2^m$

- Reason: necessary block length is $\lceil \log_2 |T| \rceil$ vs $|T|$.
- The interesting tradeoff makes circular-shift LNC more flexible to be applied in networks with different computational constraints.

Summary

- Circular-shift LNC cannot achieve the exact capacity for some multicast networks.
- For prime L with primitive root 2, an intrinsic connection is established between scalar LNC over GF(2^L) and (L–1, L) circular-shift LNC for general networks.
- For any $1 \leq \delta \leq (L-1)/2$, as long as $\binom{L}{0} + \binom{L}{1} + \ldots + \binom{L}{\delta} \geq |T|$ an (L–1, L) circular-shift linear solution of order δ can be efficiently constructed.
- There is an interesting tradeoff between encoding and decoding complexity with different choice of degree δ.

Concluding Remarks

- Circular-shift LNC cannot achieve the exact capacity for some multicast networks.
- For prime L with primitive root 2, an intrinsic connection is established between scalar LNC over GF(2^L) and (L–1, L) circular-shift LNC for general networks.
- 3, 5, 11, 13, 19, 29, 37, 53, 59, 61, 67, 83, 101, 107, 131, 139, 149, 163, 173, 179, 181, 197, 211, 227, 269, 293, 317, 347, 349, 373, 379, 389, 419, 421, 443, 461, 467, 491, 509, 523, 541, 547, 557, 563, 587, 613, 619, 653, 659, 661, 677, 701, 709, 757, 773, 787, 797 …
- It is unknown whether every multicast network is asymptotically circular-shift linearly solvable.

Concluding Remarks

Circular-shift LNC (of a degree 1) vs permutation-based LNC

- L+1 vs L!
- Random coding for both can yield an asymptotic linear solution with high probability;
- No obvious disadvantage of circular-shift LNC w.r.t. successful probability of random construction.
- Circular-shift LNC has advantage on shorter overheads for random coding.

Concluding Remarks

In the deterministic framework,

- For practical purpose, we only studied (L–1, L) circular-shift LNC over GF(2).
- This work can be theoretically extended to
  - be over GF(p);
  - construct an (L', L) circular-shift linear solution.

  // once $\lim_{L\to\infty} L'/L = 1$, we can prove "every multicast network is asymptotically circular-shift linearly solvable".

- Q. T. Sun, et. al., "Circular-shift linear network coding," *ISIT'17*.
- H. Tang, et. al, "A random coding analysis of circular-shift linear network coding," *Poster session, ISIT'17* & Croucher IT summer school'17.