Rate-optimal Binary Locally Repairable Codes with Joint Information Locality

Jung-Hyun Kim, **Mi-Young Nam**, **and Hong-Yeop Song**
Yonsei University, Korea
(jh.kim06, my.nam, hysong@yonsei.ac.kr)

CSDL

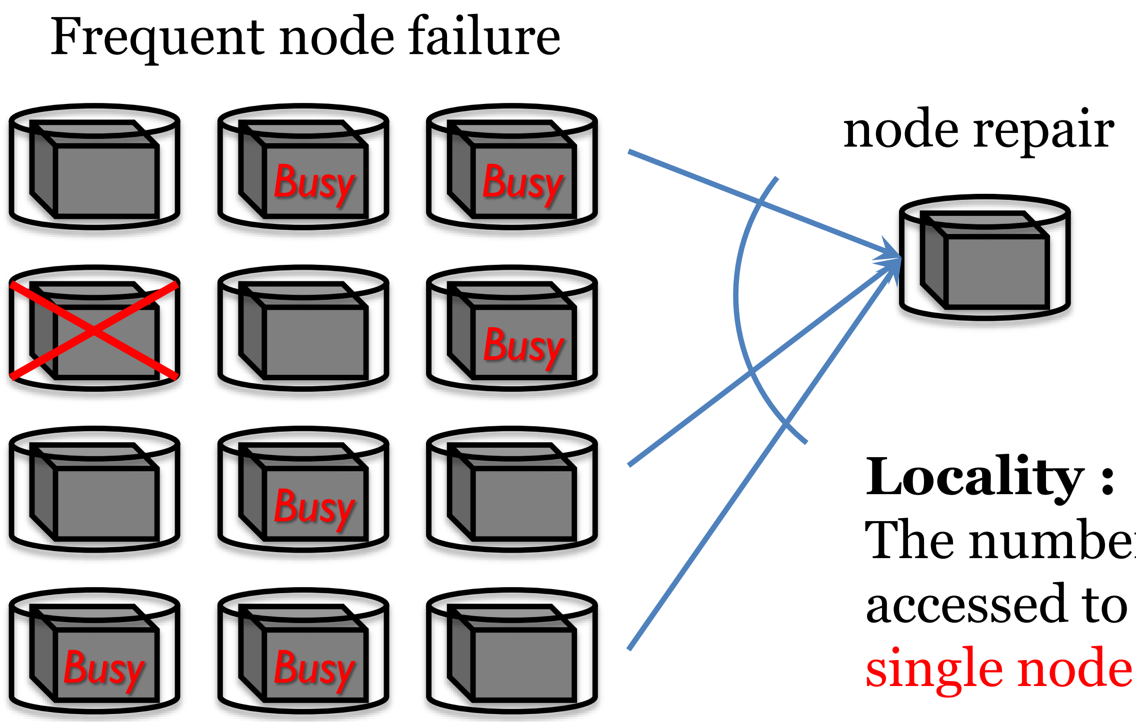
2015 / 10 / 12
2015 IEEE Information Theory Workshop

Outline

- **Introduction**
- Prior Work
- BLRC with Joint Inform. Locality
- Summary & Conclusion

Introduction

- Distributed Storage System (DSS)



Node failure :
A disk fault or
a node in use

Locality :
The number of nodes
accessed to repair a
single node failure

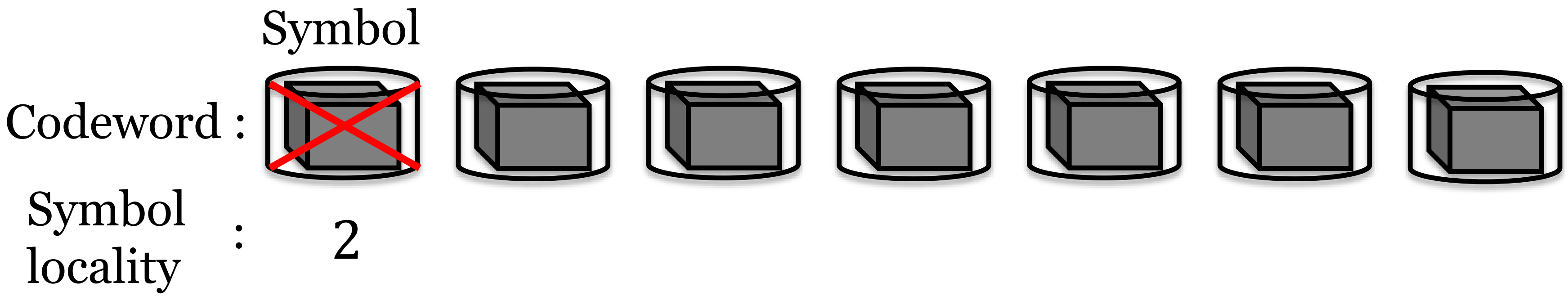
P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin, "On the locality of codeword symbols," IEEE Trans. Inform. Theory, vol. 58, no. 11, pp. 6925-6934, Nov. 2012.

Introduction

- Locally repairable code (LRC)
  - Codes with good (small) locality

Locality

- **Symbol locality** : # of symbols required to repair a failed symbol
- **(Code) locality** : the maximum value of symbol locality



Introduction

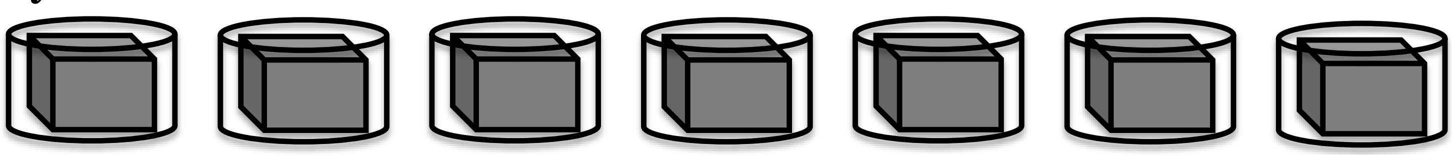
- Locally repairable code (LRC)
  - Codes with good (small) locality

Locality

- **Symbol locality** : # of symbols required to repair a failed symbol
- **(Code) locality** : the maximum value of symbol locality

Symbol

Codeword :



Symbol locality :	2	2	3	4	3	5	2

Introduction

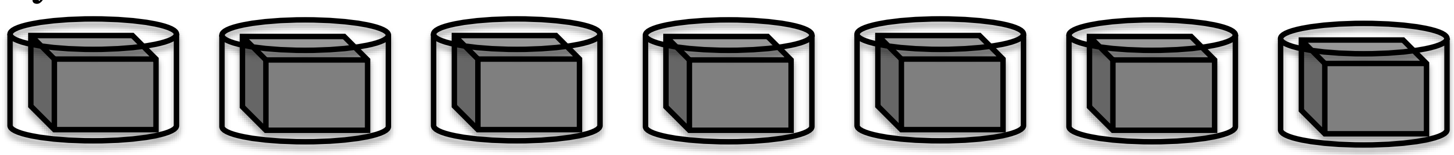
- Locally repairable code (LRC)
  - Codes with good (small) locality

Locality

- **Symbol locality** : # of symbols required to repair a failed symbol
- **(Code) locality** : the maximum value of symbol locality

Symbol

Codeword :



Symbol locality : 2 2 3 4 3 5 2

Code locality (r_1) : 5

Introduction

- Locally repairable code (LRC)
  - Codes with good (small) locality

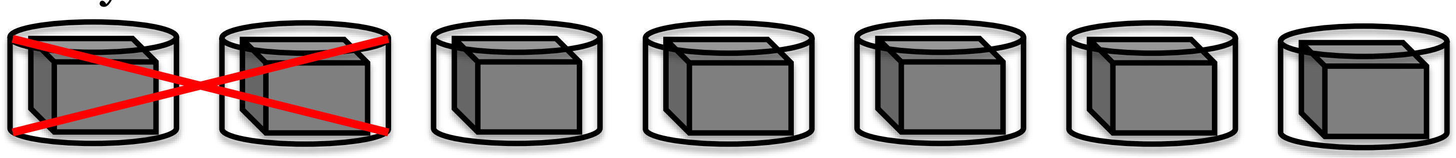
Locality (Generalized definition)

- ***ℓ*-locality (r_ℓ)** : locality for ℓ symbols repair

* 1-locality (r_1) is the same with "code locality" in the previous definition

Symbol set

Codeword :



A. S. Rawat, A. Mazumdar, and S. Vishwanath, "Cooperative local repair in distributed storage," arXiv Preprint arXiv:1409.3900, 2014.

Jung-Hyun Kim, Mi-Young Nam, Ki-Hyeon Park, and Hong-Yeop Song, "New Binary Locally Repairable Codes with Joint Locality and Average Locality," under revision, IEEE Trans. on Inf. Theory.

Outline

- Introduction
- **Prior Work**
- BLRC with Joint Inform. Locality
- Summary & Conclusion

Prior Work

(Binary) Simplex codes

$r_1 = 2$ (VERY GOOD)
$R = \frac{k}{2^{k-1}}$ (VERY LOW)

Only better code is repetition code ($r_1 = 1$), but its code rate is extremely low.

Prior Work

(Binary) Simplex codes

$r_1 = 2$ (VERY GOOD)
$R = \frac{k}{2^{k-1}}$ (VERY LOW)

Only better code is repetition code ($r_1 = 1$), but its code rate is extremely low.

Q1 : Can we improve the code rate maintaining the locality?

$(\boldsymbol{r_1} = \mathbf{2})$

Prior Work

(Binary) Simplex codes

$r_1 = 2$ (VERY GOOD)

$R = \frac{k}{2^k - 1}$ (VERY LOW)

Only better code is repetition code ($r_1 = 1$), but its code rate is extremely low.

Q1 : Can we improve the code rate maintaining the locality?

$(\boldsymbol{r_1 = 2})$

Simplex code
$(\boldsymbol{r_1 = 2})$

$$G_S = \begin{pmatrix} 1000 & 0110 & 10 & 1110 & 1 \\ 0100 & 1001 & 10 & 1101 & 1 \\ 0010 & 0101 & 01 & 1011 & 1 \\ 0001 & 1010 & 01 & 0111 & 1 \end{pmatrix} \qquad R_S = \frac{4}{15}$$

Prior

Complete graph code
$(\boldsymbol{r_1 = 2})$

$$G_{CG} = \begin{pmatrix} 1000 & 0110 & 10 \\ 0100 & 1001 & 10 \\ 0010 & 0101 & 01 \\ 0001 & 1010 & 01 \end{pmatrix} \qquad \boldsymbol{R_{CG} = \frac{4}{10}}$$

Complete multipartite graph code
$(\boldsymbol{r_1 = 2})$

$$G_{CMG} = \begin{pmatrix} 1000 & 0110 \\ 0100 & 1001 \\ 0010 & 0101 \\ 0001 & 1010 \end{pmatrix} \qquad \boldsymbol{R_{CMG} = \frac{4}{8}}$$

Jung-Hyun Kim, Mi-Young Nam, Ki-Hyeon Park, and Hong-Yeop Song, "New Binary Locally Repairable Codes with Joint Locality and Average Locality," under revision, IEEE Trans. on Inf. Theory.

Prior Work

Q2 : What about multiple failure patterns? (every ℓ-locality)

Prior Work

Q2 : What about multiple failure patterns? (every ℓ-locality)

Which one is better? C_1? or C_2?

$$G_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \quad G_2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 \end{pmatrix}$$

Prior Work

Q2 : What about multiple failure patterns? (every ℓ-locality)

Which one is better? C_1? or C_2?

$$G_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \quad G_2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 \end{pmatrix}$$

$r_1 = 2$ **>** $r_1 = 3$

Prior Work

Q2 : What about multiple failure patterns? (every ℓ-locality)

Which one is better? C_1? or C_2?

$$G_1 = \begin{pmatrix} 1&0&0&0&0&1&1&1&1&0&1 \\ 0&1&0&0&0&1&1&1&0&1&1 \\ 0&0&1&0&0&1&1&0&1&1&1 \\ 0&0&0&1&0&1&0&1&1&1&1 \\ 0&0&0&0&1&0&1&1&1&1&1 \end{pmatrix} \quad G_2 = \begin{pmatrix} 1&0&0&0&0&1&1&1&0&1&0 \\ 0&1&0&0&0&1&1&0&0&1&1 \\ 0&0&1&0&0&1&0&0&1&1&1 \\ 0&0&0&1&0&0&0&1&1&1&1 \\ 0&0&0&0&1&0&1&1&1&1&0 \end{pmatrix}$$

$r_1 = 2$ $>$ $r_1 = 3$

$r_2 = 5$ $<$ $r_2 = 4$

Prior Work

Q2 : What about multiple failure patterns? (every ℓ-locality)

Which one is better? C_1? or C_2?

$$G_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \quad G_2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 \end{pmatrix}$$

$r_1 = 2$	$>$	$r_1 = 3$
$r_2 = 5$	$<$	$r_2 = 4$
$(r_1, r_2) = (2, 5)$		$(r_1, r_2) = (3, 4)$

Prior Work

Q2 : What about multiple failure patterns? (every ℓ-locality)

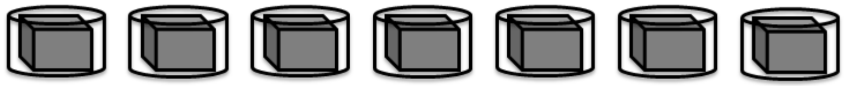
Which one is better? C_1? or C_2?

$$G_1 = \begin{pmatrix} 1&0&0&0&0&1&1&1&1&0&1 \\ 0&1&0&0&0&1&1&1&0&1&1 \\ 0&0&1&0&0&1&1&0&1&1&1 \\ 0&0&0&1&0&1&0&1&1&1&1 \\ 0&0&0&0&1&0&1&1&1&1&1 \end{pmatrix} \quad G_2 = \begin{pmatrix} 1&0&0&0&0&1&1&1&0&1&0 \\ 0&1&0&0&0&1&1&0&0&1&1 \\ 0&0&1&0&0&1&0&0&1&1&1 \\ 0&0&0&1&0&0&0&1&1&1&1 \\ 0&0&0&0&1&0&1&1&1&1&0 \end{pmatrix}$$

Joint locality

$$(r_1, r_2, r_3) = (2, 5, 5) \qquad (r_1, r_2, r_3) = (3, 4, 5)$$

Prior Work

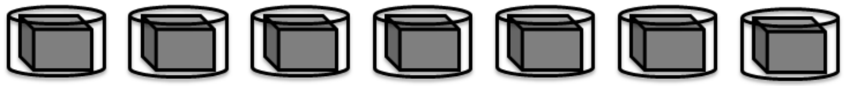


Symbol locality : 2 2 3 4 3 5 2

$\underbrace{\hspace{6cm}}$

1-locality (r_1) : 5 Maximum value (Worst case)

Prior Work



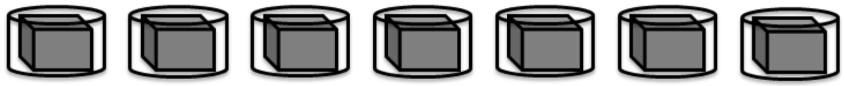
Symbol locality : 2 2 3 4 3 5 2

1-locality (r_1) : 5 Maximum value (Worst case)

Average 1-locality ($\bar{r_1}$) : 4 Average value

Prior Work

Q3 : Worst vs. Average?



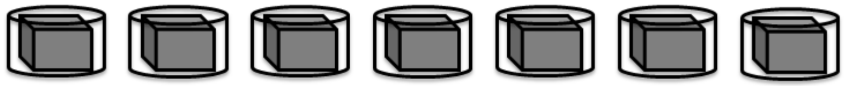
Symbol locality : 2 2 3 4 3 5 2

1-locality (r_1) : 5 — Maximum value (Worst case)

Average 1-locality ($\bar{r}_1$) : 4 — Average value

Which one is more reasonable measure?

Prior Work



Symbol locality : 2 2 3 4 3 5 2

Q3 : Worst vs. Average?

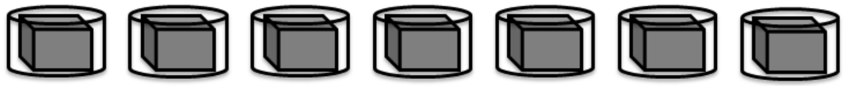
1-locality (r_1) : 5 — Maximum value (Worst case)

Average 1-locality ($\overline{r_1}$) : 4 — Average value

Which one is better? C_1? or C_2?

$$G_1 = \begin{pmatrix} 1&0&0&0&0&1&1&1&1&0&1 \\ 0&1&0&0&0&1&1&1&0&1&1 \\ 0&0&1&0&0&1&1&0&1&1&1 \\ 0&0&0&1&0&1&0&1&1&1&1 \\ 0&0&0&0&1&0&1&1&1&1&1 \end{pmatrix} \quad G_2 = \begin{pmatrix} 1&0&0&0&0&1&1&1&0&1&0 \\ 0&1&0&0&0&1&1&0&0&1&1 \\ 0&0&1&0&0&1&0&0&1&1&1 \\ 0&0&0&1&0&0&0&1&1&1&1 \\ 0&0&0&0&1&0&1&1&1&1&0 \end{pmatrix}$$

Prior Work



Symbol locality : 2 2 3 4 3 5 2

1-locality (r_1) : 5 Maximum value (Worst case)

Average 1-locality ($\overline{r_1}$) : 4 Average value

Q3 : Worst vs. Average?

Which one is better? C_1? or C_2?

$$G_1 = \begin{pmatrix} 1&0&0&0&0&1&1&1&1&0&1 \\ 0&1&0&0&0&1&1&1&0&1&1 \\ 0&0&1&0&0&1&1&0&1&1&1 \\ 0&0&0&1&0&1&0&1&1&1&1 \\ 0&0&0&0&1&0&1&1&1&1&1 \end{pmatrix} \quad G_2 = \begin{pmatrix} 1&0&0&0&0&1&1&1&0&1&0 \\ 0&1&0&0&0&1&1&0&0&1&1 \\ 0&0&1&0&0&1&0&0&1&1&1 \\ 0&0&0&1&0&0&0&1&1&1&1 \\ 0&0&0&0&1&0&1&1&1&1&0 \end{pmatrix}$$

Average locality

$$(\overline{r_1}, \overline{r_2}, \overline{r_3}) = \left(2, \frac{105}{33}, \frac{141}{33}\right) > (\overline{r_1}, \overline{r_2}, \overline{r_3}) = \left(3, \frac{132}{33}, \frac{145}{33}\right)$$

Main results

Joint locality

Code (dimension k)	**Code rate**	(r_1, r_2)	**Another metric?**
Simplex code	$\frac{k}{2^k - 1}$	$(2, 3)$	?
Complete graph code	$\frac{2}{k+1}$	$(2, 3)$	?
Complete multipartite graph code (p-partite)	$\frac{2}{k - \frac{k}{p} + 2}$	$(2, 4)$	?
New code?	?	$(2, 4)$	?

increasing code rate

Outline

- Introduction
- Prior Work
- **BLRC with Joint Inform. Locality**
- Summary & Conclusion

BLRC with Joint Inform. Locality

- Joint Information Locality
  - a set of numbers of symbols for repairing various erasure patterns of information symbols

BLRC with Joint Inform. Locality

- Joint Information Locality
  - a set of numbers of symbols for repairing various erasure patterns of information symbols

Can we design rate-optimal codes with joint inform. locality $(2,3)$ or $(2,4)$?

BLRC with Joint Inform. Locality

- Joint Information Locality
  - a set of numbers of symbols for repairing various erasure patterns of information symbols

Can we design rate-optimal codes with joint inform. locality $(2,3)$ or $(2,4)$?

We begin with a **simple graph**.

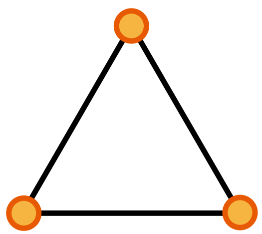
an unweighted, undirected, connected graph containing no loops or multiple edges

BLRC with Joint Inform. Locality

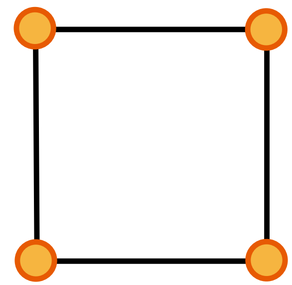
simple graph

$$k = \#v$$
$$n = \#v + \#e$$
$$\longrightarrow \text{If } \#e \uparrow, \text{ then } \frac{k}{n} \downarrow$$

Vertex : inform. symbol

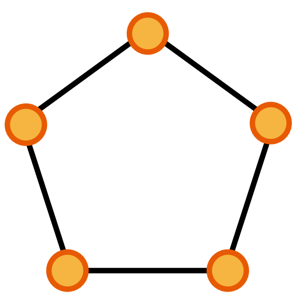


$$G = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

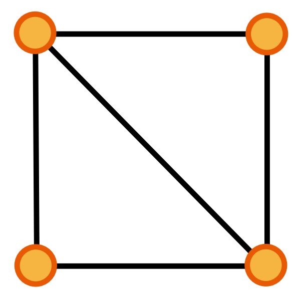


$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{bmatrix}$$

edge : parity symbol



$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$



$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

BLRC with Joint Inform. Locality

- Simple graph-based code construction
  - Minimum distance

    obtained straightforwardly

We found this expression.

$$d = \min_{S \subseteq V} [|Cut(S, S^c)| + |S|]$$

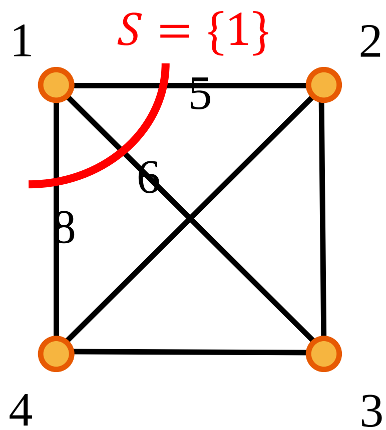
where V is the set of all the vertices.

BLRC with Joint Inform. Locality

- Simple graph-based code construction
  - Minimum distance $\quad d = \min\limits_{S \subseteq V} [|Cut(S, S^c)| + |S|]$

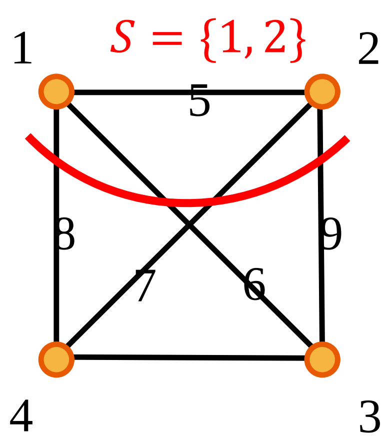
    obtained straightforwardly

    where V is the set of all the vertices.



$$\begin{array}{cc} & \begin{array}{cccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{array} \\ G_S & \\ G = & \begin{bmatrix} \boxed{1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0} \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix} \end{array}$$

$$c_1 = [1\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 0]$$



$$\begin{array}{cc} & \begin{array}{cccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{array} \\ G_S & \\ G = & \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix} \end{array} \quad \oplus$$

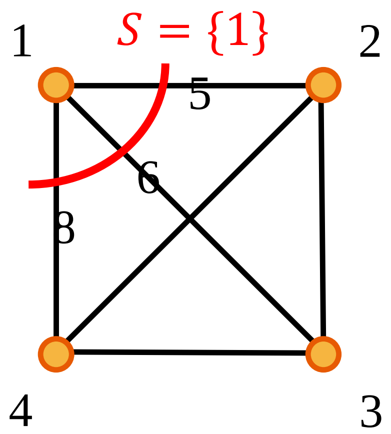
$$c_2 = [1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0]$$

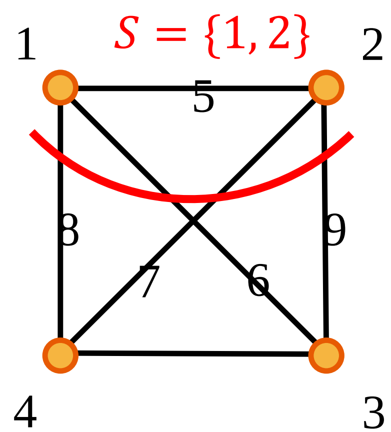
BLRC with Joint Inform. Locality

- Simple graph-based code construction
  - Minimum distance $d = \min_{S \subseteq V} [|Cut(S, S^c)| + |S|]$

    obtained straightforwardly

    where V is the set of all the vertices.





To repair 2 failed symbols,

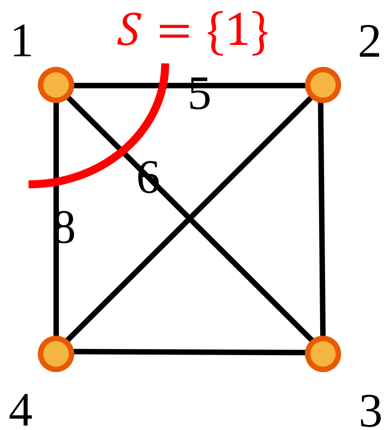
$d \geq 3 \iff$ For $\forall v$ $\deg(v) \geq 2$

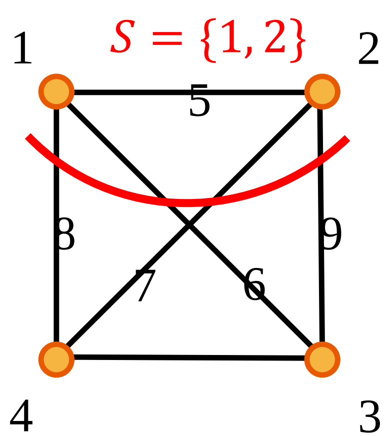
BLRC with Joint Inform. Locality

- Simple graph-based code construction
  - Minimum distance $d = \min_{S \subseteq V} [|Cut(S, S^c)| + |S|]$

    obtained straightforwardly

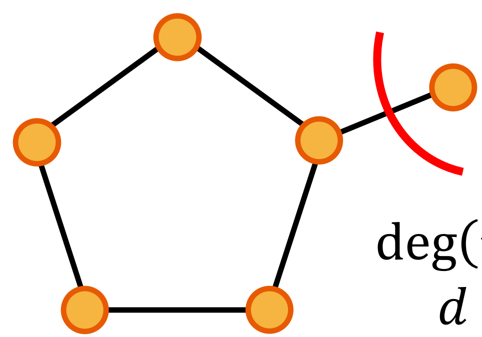
    where V is the set of all the vertices.





To repair 2 failed symbols,

$$d \geq 3 \;\Leftrightarrow\; \text{For } \forall v \;\; \deg(v) \geq 2$$

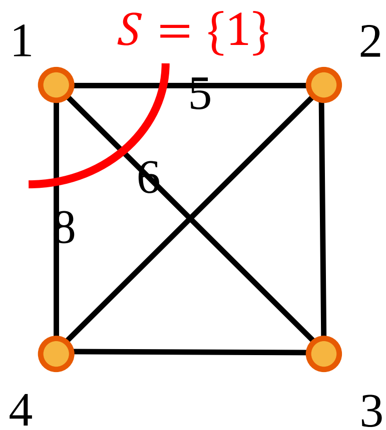


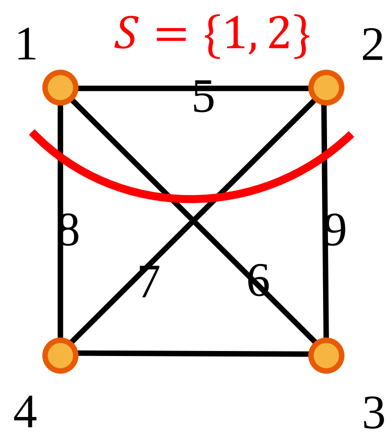
BLRC with Joint Inform. Locality

- Simple graph-based code construction
  - Minimum distance $d = \min_{S \subseteq V}[|Cut(S, S^c)| + |S|]$

obtained straightforwardly

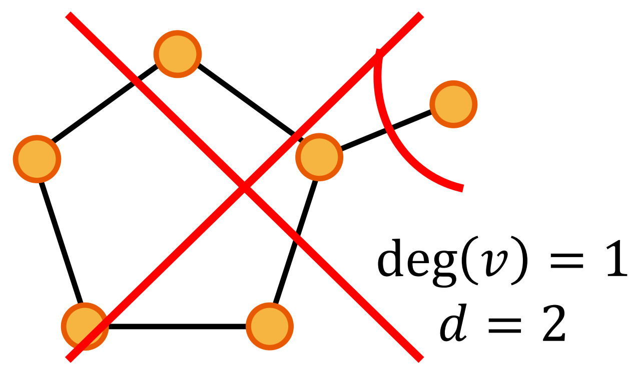
where V is the set of all the vertices.





To repair 2 failed symbols,

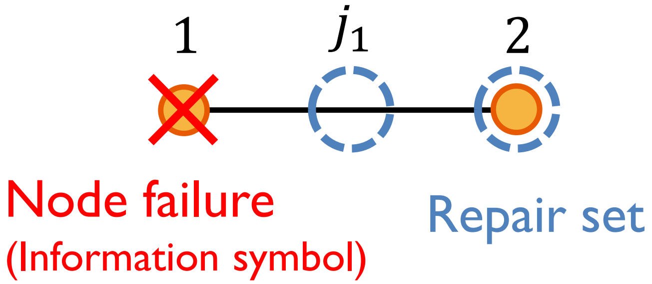
$d \geq 3 \;\Leftrightarrow\;$ For $\forall v$ $\deg(v) \geq 2$



BLRC with Joint Inform. Locality

- Simple graph-based code construction

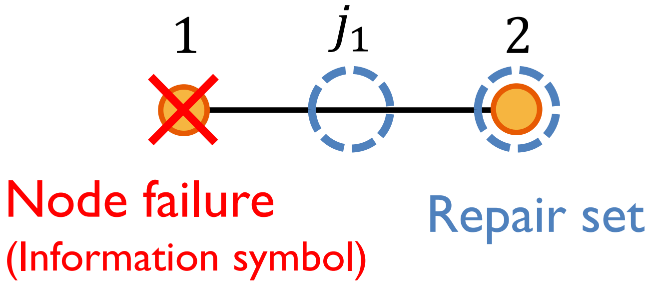
Lemma 1. Always $(r_1)_{info} = 2$



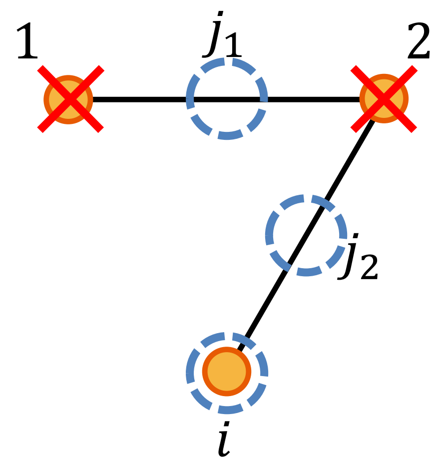
BLRC with Joint Inform. Locality

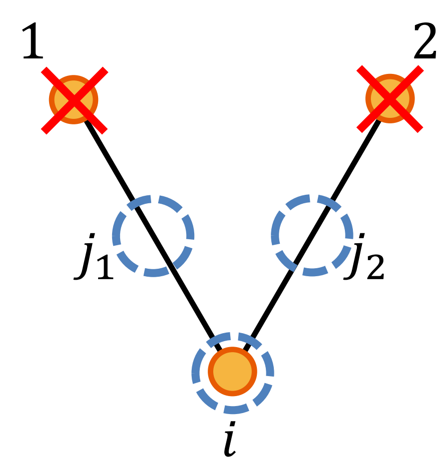
- Simple graph-based code construction

Lemma 1. Always $(r_1)_{info} = 2$



Lemma 2. If every vertex pair is in 2-hop distance, $(r_2)_{info} = 3$





BLRC with Joint Inform. Locality

2-hop distance vs. higher rate

BLRC with Joint Inform. Locality

2-hop distance vs. higher rate

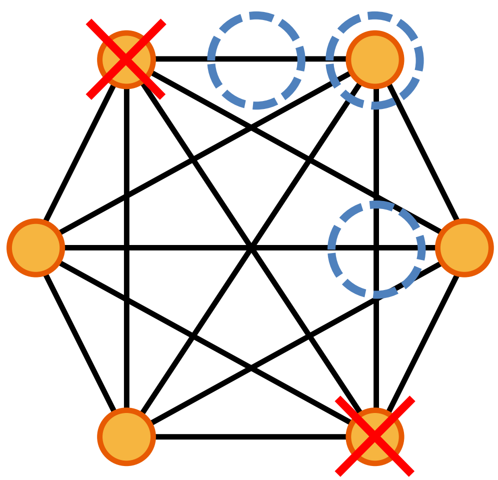
# of edges ↑
Too many edges ⇒ low rate

# of edges ↓
Too few edges ⇒ ~~2-hop~~

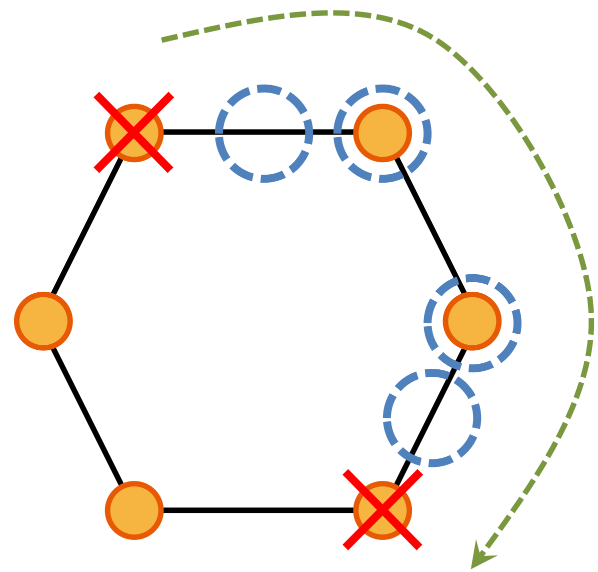
BLRC with Joint Inform. Locality

2-hop distance vs. higher rate

# of edges ↑	# of edges ↓
Too many edges ⇒ low rate	Too few edges ⇒ 2-~~hop~~



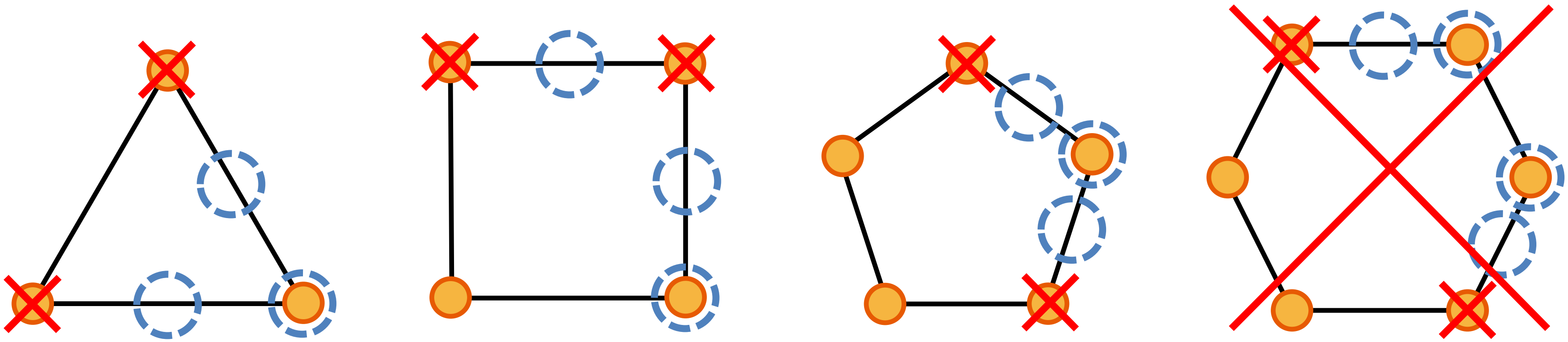
2-hop
Low rate



High rate
3-hop

BLRC with Joint Inform. Locality

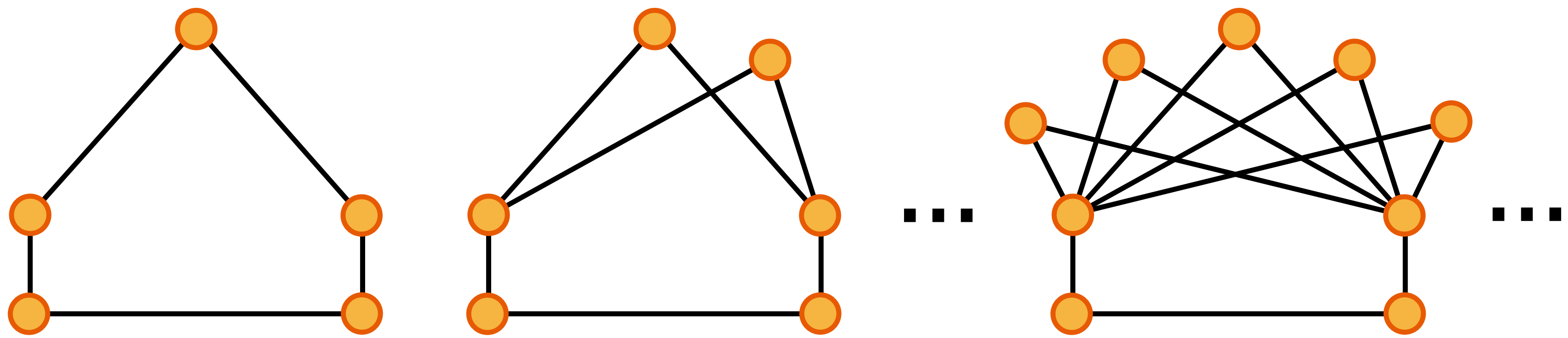
Lemma 3. $(r_1, r_2)_{info} = (2, 3)$,
if and only if any vertex pair is in either of triangle, quadrangle, or pentagon.



$r_2 = 4$

BLRC with Joint Inform. Locality

- Crown code
  - Rate-optimal code with joint information locality $(r_1, r_2)_{info} = (\mathbf{2}, \mathbf{3})$



$k = 5$ $k = 6$ $k = 9$

For every positive integer $k \geq 5$, the code construction is possible.

BLRC with Joint Inform. Locality

- Crown code

Theorm 1.

$(r_1, r_2)_{info} = (2, 3)$ $\rightleftarrows$ Any vertex pair should be in either of $\square$ or $\pentagon$, and no more. The graph should contain at least one $\pentagon$.

Rate-optimal

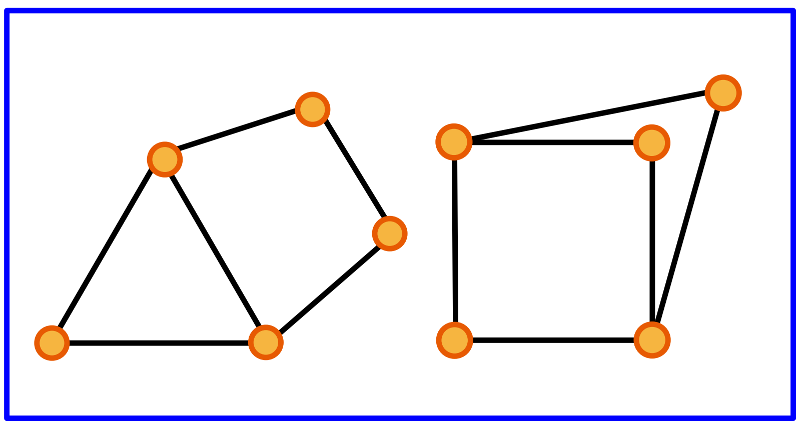
BLRC with Joint Inform. Locality

- Crown code

Theorm 1.

$(r_1, r_2)_{info} = (2,3)$ ⇄ Any vertex pair should be in either of □ or ⬠, and no more. The graph should contain at least one ⬠.

Rate-optimal



$(r_1, r_2)_{info} = (2,3)$

Not rate-optimal

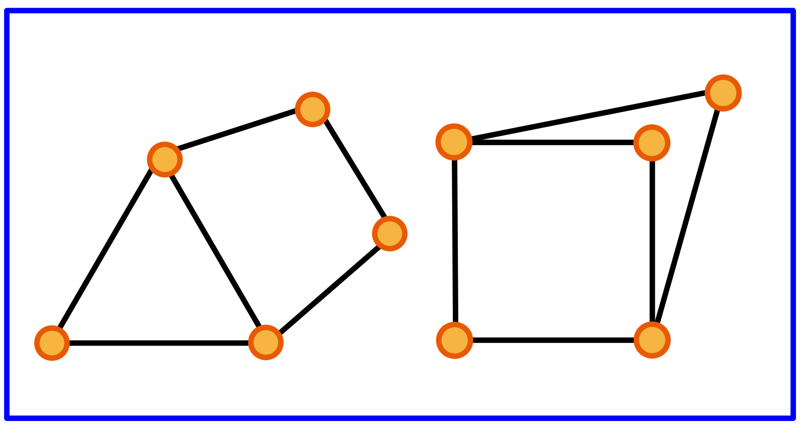
BLRC with Joint Inform. Locality

- Crown code

Theorm 1.

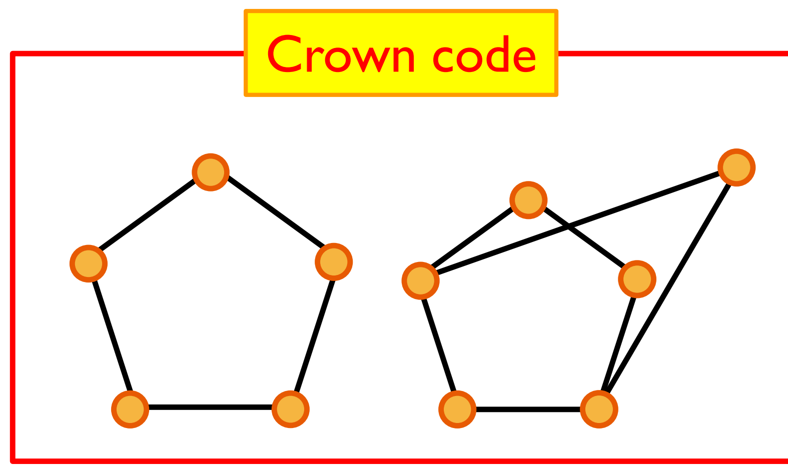
$(r_1, r_2)_{info} = (2, 3)$ $\rightleftarrows$ Any vertex pair should be in either of □ or ⬠, and no more. The graph should contain at least one ⬠.

Rate-optimal



$(r_1, r_2)_{info} = (2, 3)$

Not rate-optimal



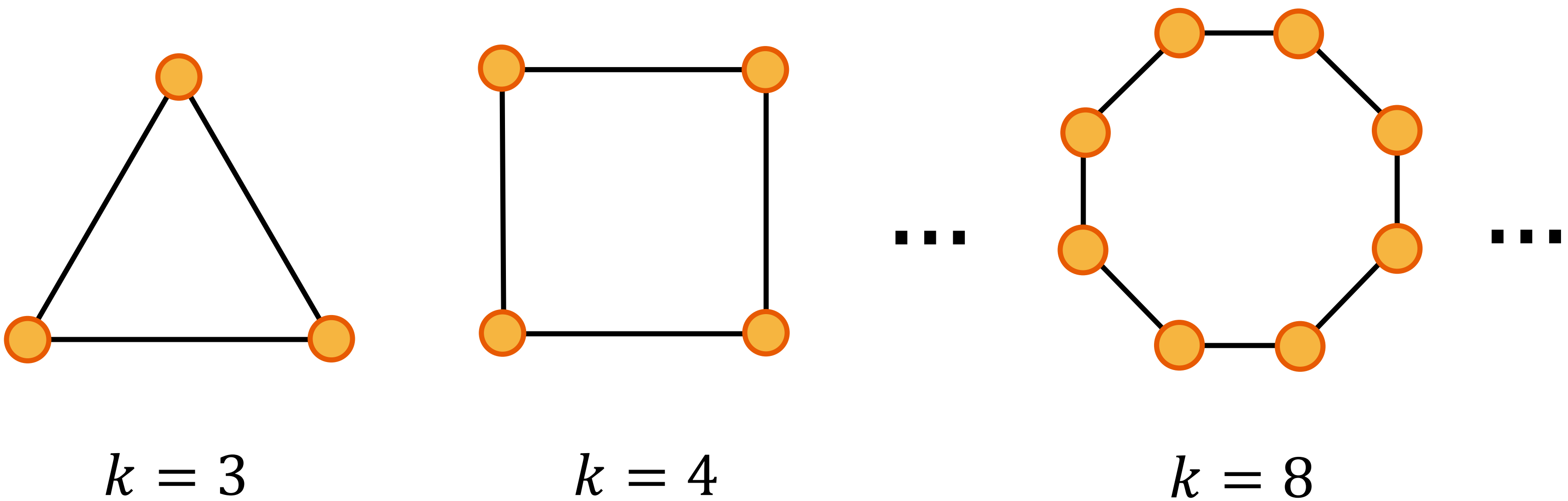
$(r_1, r_2)_{info} = (2, 3)$

Rate-optimal

BLRC with Joint Inform. Locality

- Ring code
  - Rate-optimal code with joint information locality $(r_1, r_2)_{info} = (\mathbf{2}, \mathbf{4})$

More than 2-hop is ok



For every positive integer $k \geq 3$, the code construction is possible.
When $k = 5$, $(r_1, r_2)_{info} = (2, 3)$ since it is also a crown code.

BLRC with Joint Inform. Locality

- Ring code

Theorm 2.

$(r_1, r_2)_{info} = (2, 4)$ $\rightleftarrows$ The graph should be single cycle structure

Rate-optimal

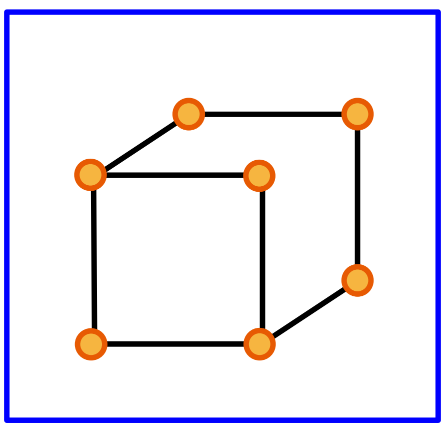
BLRC with Joint Inform. Locality

- Ring code

Theorm 2.

$(r_1, r_2)_{info} = (2, 4)$ $\rightleftarrows$ The graph should be single cycle structure

Rate-optimal



$(r_1, r_2)_{info} = (2, 4)$

Not rate-optimal

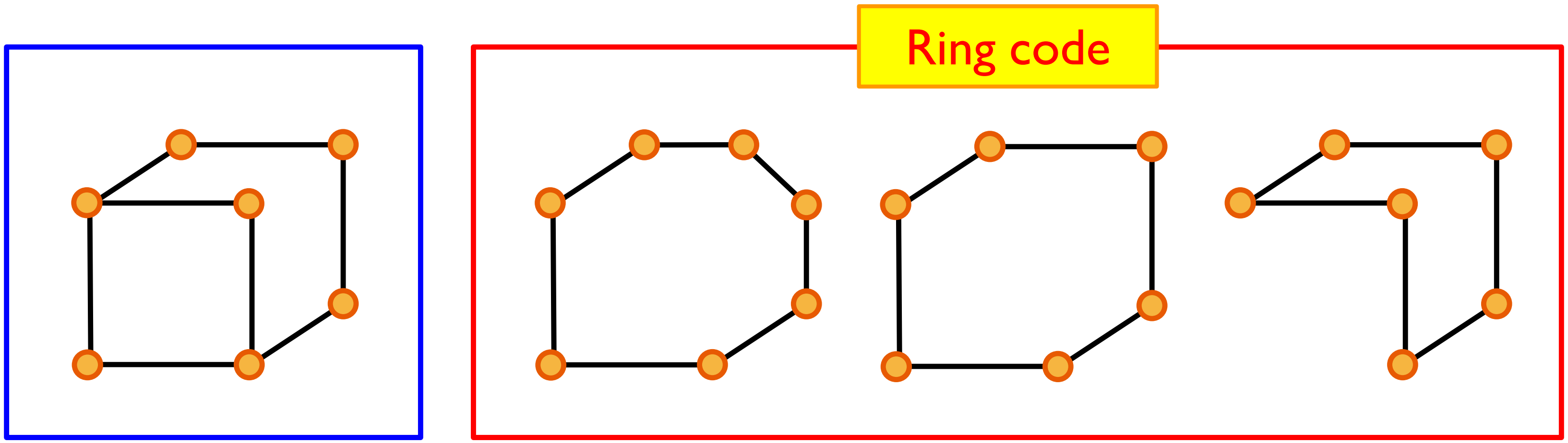
BLRC with Joint Inform. Locality

- Ring code

Theorm 2.

$(r_1, r_2)_{info} = (2, 4)$ $\rightleftarrows$ The graph should be single cycle structure

Rate-optimal



$(r_1, r_2)_{info} = (2, 4)$
Not rate-optimal

$(r_1, r_2)_{info} = (2, 4)$
Rate-optimal

Outline

- Introduction
- Prior Work
- BLRC with Joint Inform. Locality
- **Summary & Conclusion**

Summary

Average locality

Joint locality

Joint inform. locality

Code (dimension k)	**Code rate**	(r_1, r_2)	$\overline{r_2}$	$(r_1, r_2)_{info}$	$(\overline{r_2})_{info}$
Simplex code	$\frac{k}{2^k - 1}$	$(2, 3)$	3	$(2, 3)$	3
Complete graph code	$\frac{2}{k + 1}$	$(2, 3)$	3	$(2, 3)$	3
Complete multipartite graph code (p-partite)	$\frac{2}{k - \frac{k}{p} + 2}$	$(2, 4)$	$3 + \frac{2\binom{k/p}{2}^2 \binom{p}{2}}{\binom{n}{2}}$	$(2, 3)$	3
Crown code	$\frac{k}{3k - 5}$	$(2, 4)$	$3 + \frac{2k^2 - 4k - 10}{9k^2 - 33k + 30}$	$(2, 3)$	3
Ring code	$\frac{1}{2}$	$(2, 4)$	$4 - \frac{7}{2k-1}$	$(2, 4)$	$4 - \frac{4}{k-1}$

Concluding Remarks

- The rate of Crown/Ring codes gives a **global lower bound**, since it is Rate-optimal within a framework of codes based on simple graph. How good is it?

- LRC construction **not based on** simple graph

- Binary LRC with joint inform. locality $(r_1, r_2, \boldsymbol{r_3}, \boldsymbol{r_4})$

- **Non-binary LRC** construction with the same G for either Crown or Ring code