DARNEC'15

Design and Application of Random Network Codes

Istanbul Technical University

Access-optimal MSR codes with optimal sub-packetization over small fields

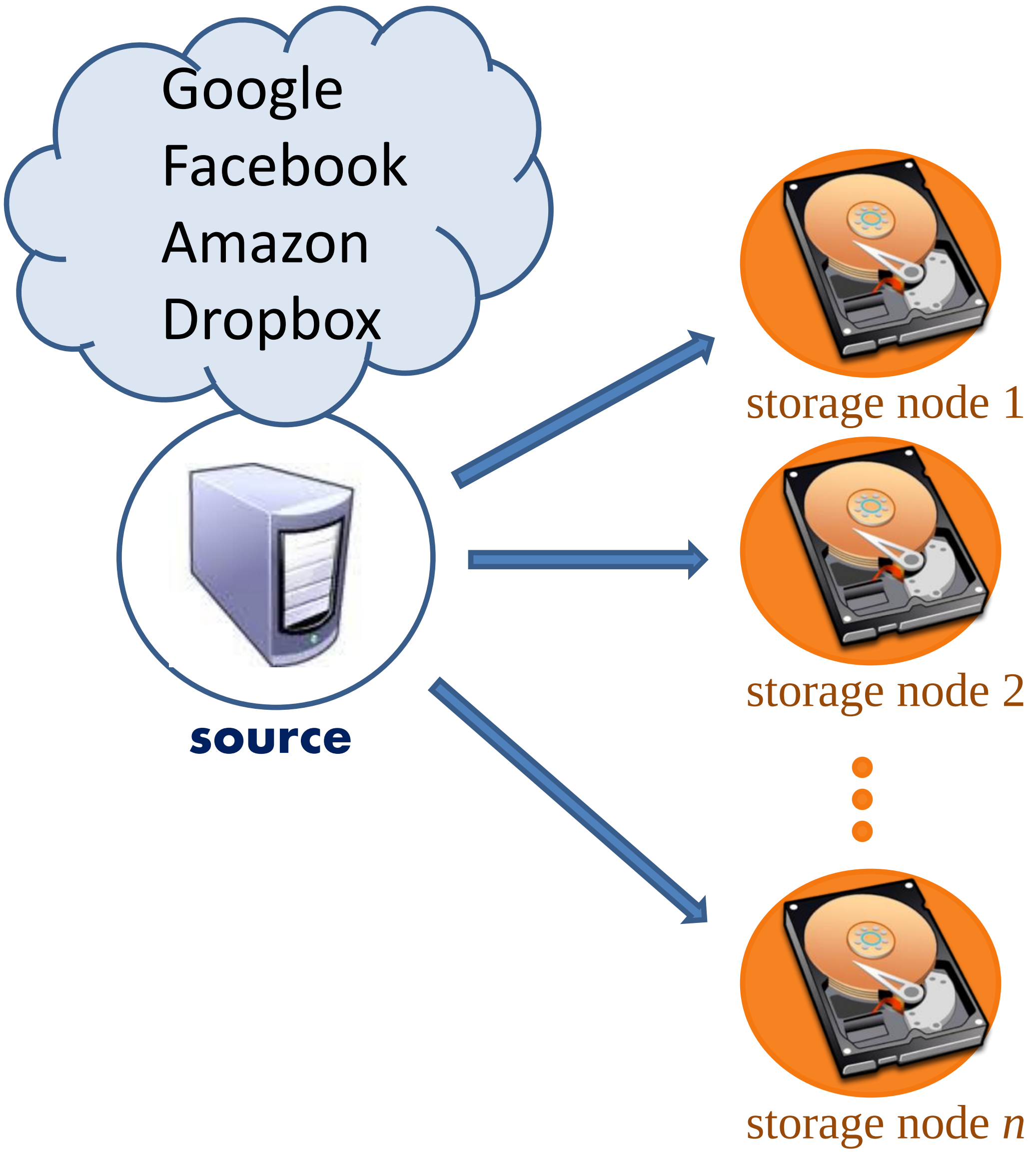
Natalia Silberstein

Joint work with

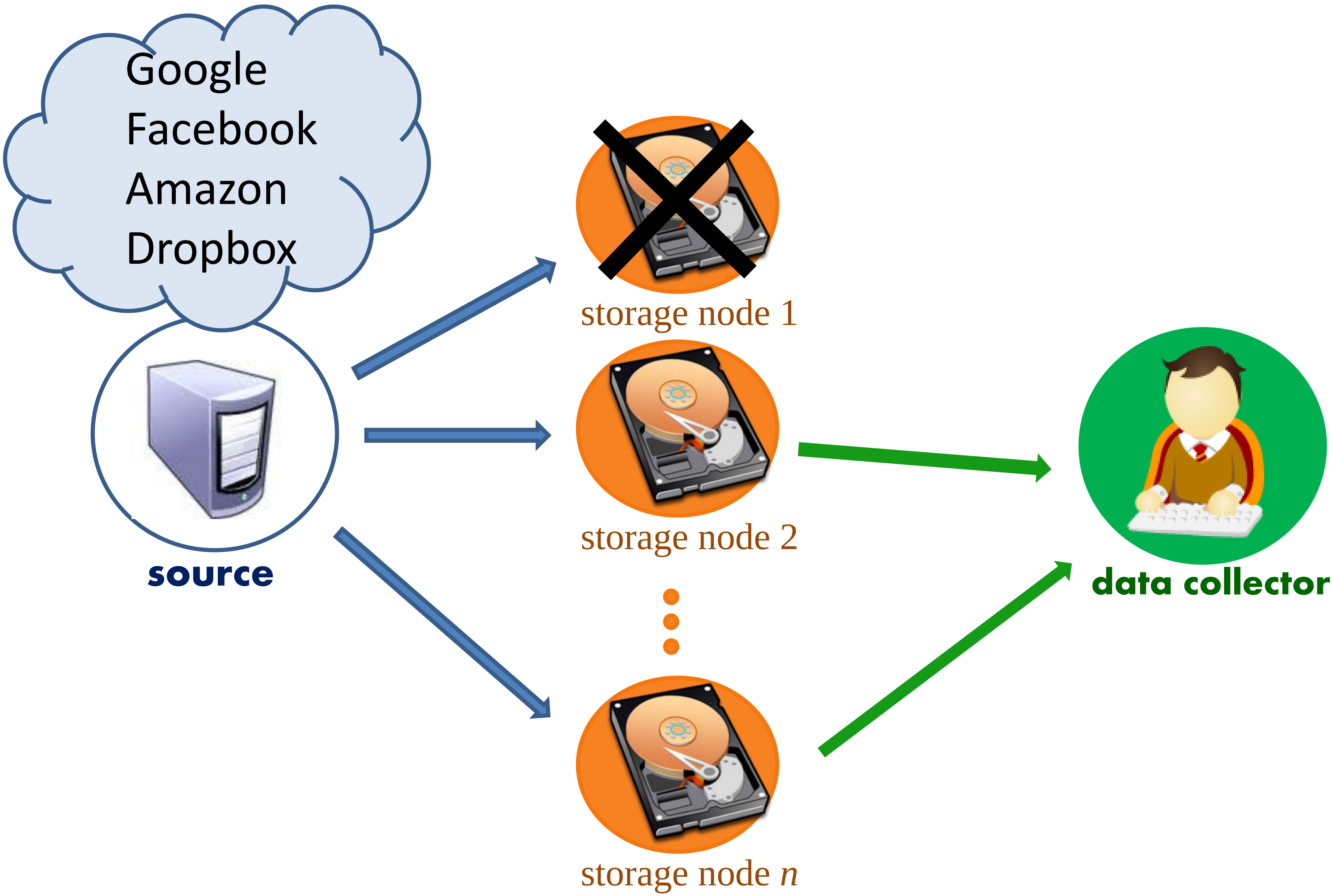
Netanel Raviv and Tuvi Etzion

Technion
Israel Institute of Technology

Distributed storage system (DSS)

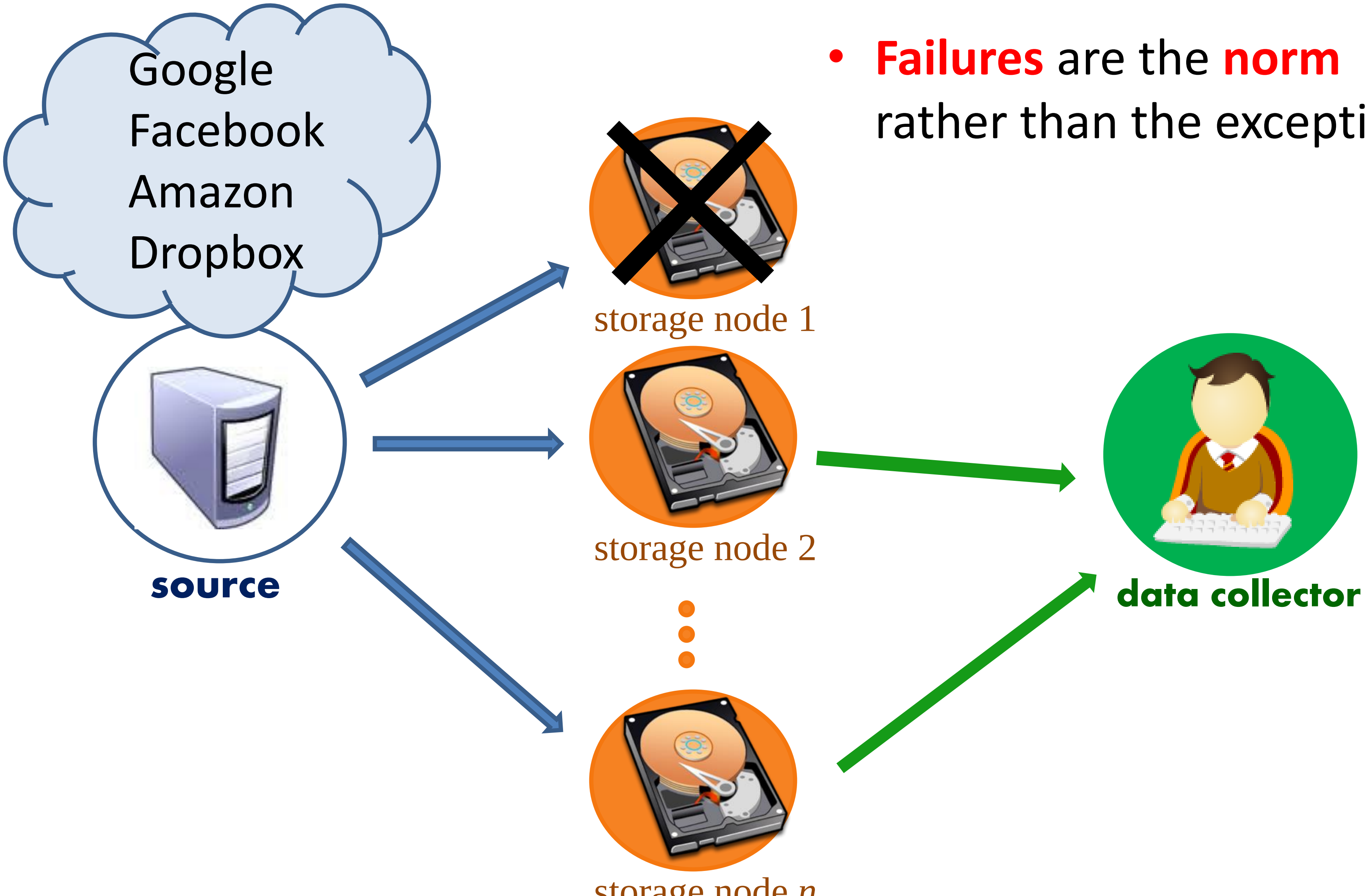


Distributed storage system (DSS)

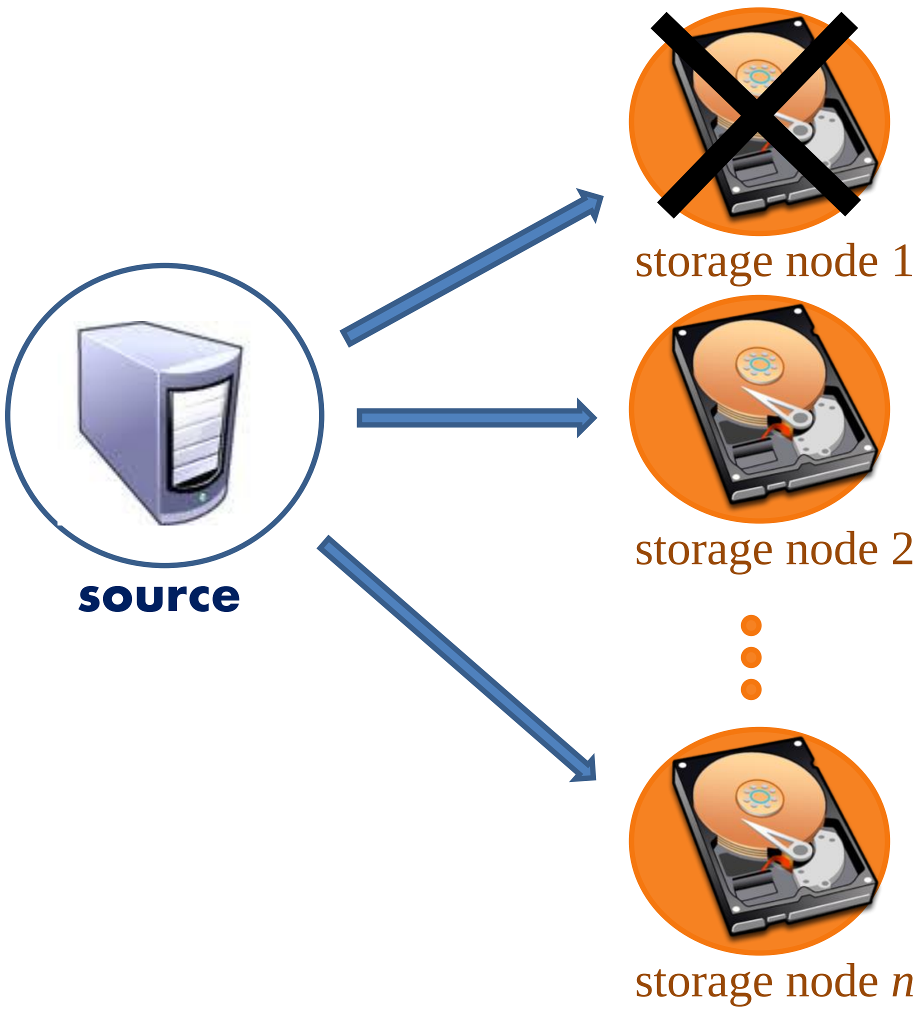


Distributed storage system (DSS)

- **Failures** are the **norm** rather than the exception

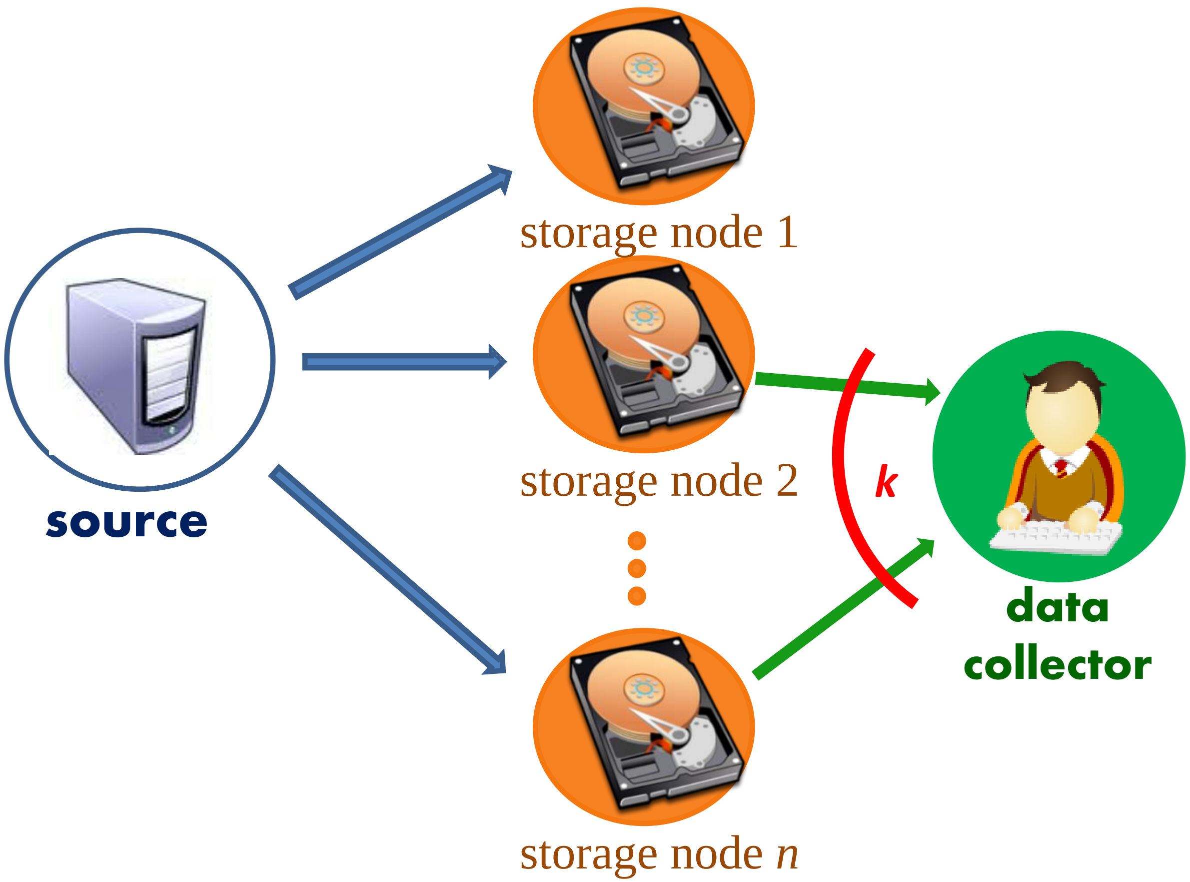


Distributed storage system (DSS)



- **Failures** are the **norm** rather than the exception
- **Redundancy** for reliability
  - **Replication**
  - **Erasure coding**

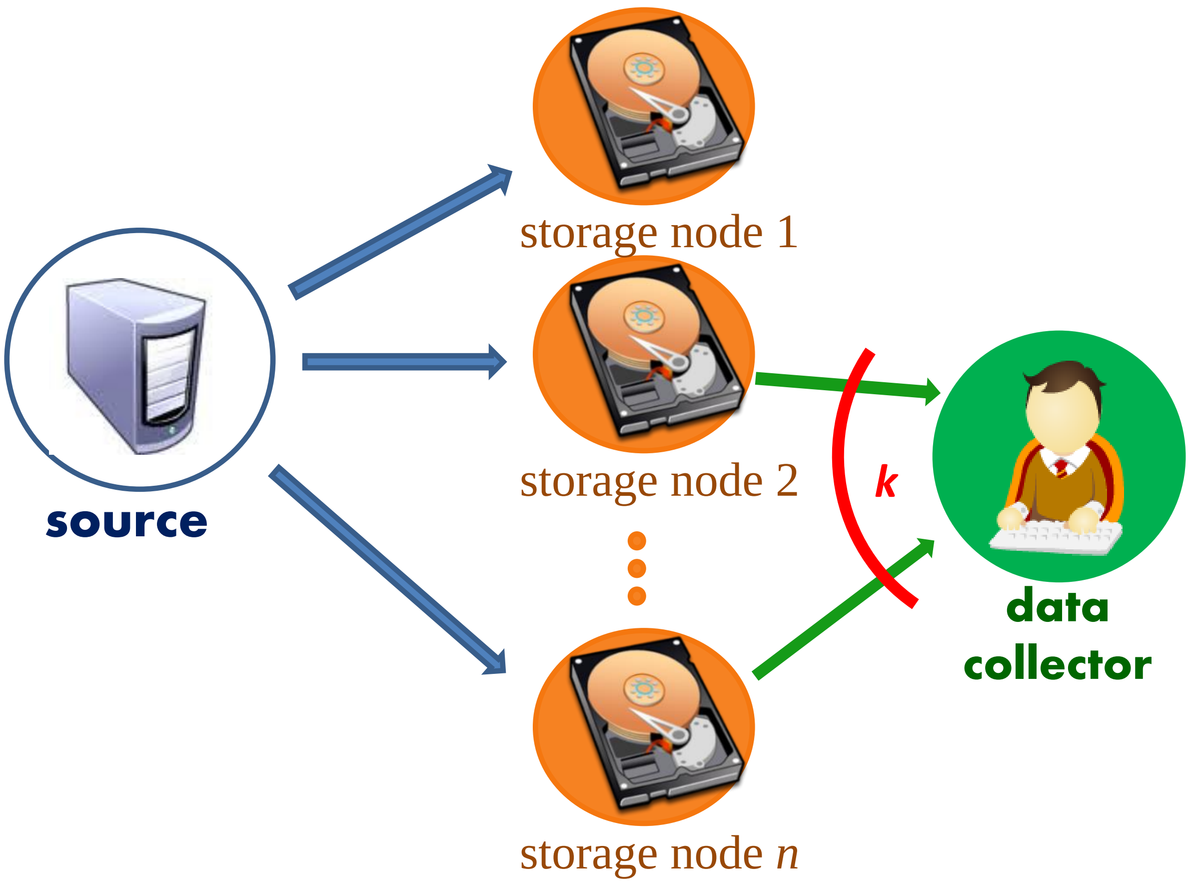
Coding for distributed storage system



Using an (n, k) MDS code:

- Partition the original data into k packets.
- Generate n packets. Store each packet in a different node.

Coding for distributed storage system



Using an (n, k) MDS code:

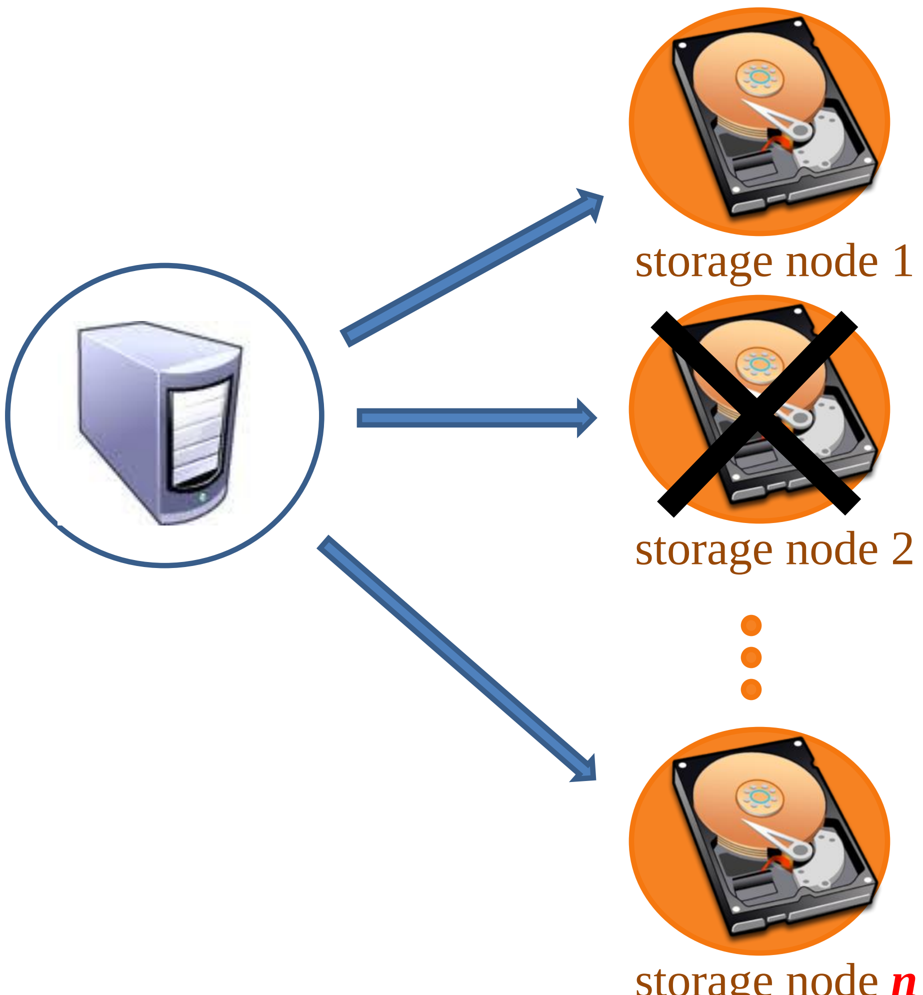
- Partition the original data into k packets.
- Generate n packets. Store each packet in a different node.

(n, k) **MDS property**: reconstruct the stored data from any k nodes

$\Updownarrow$

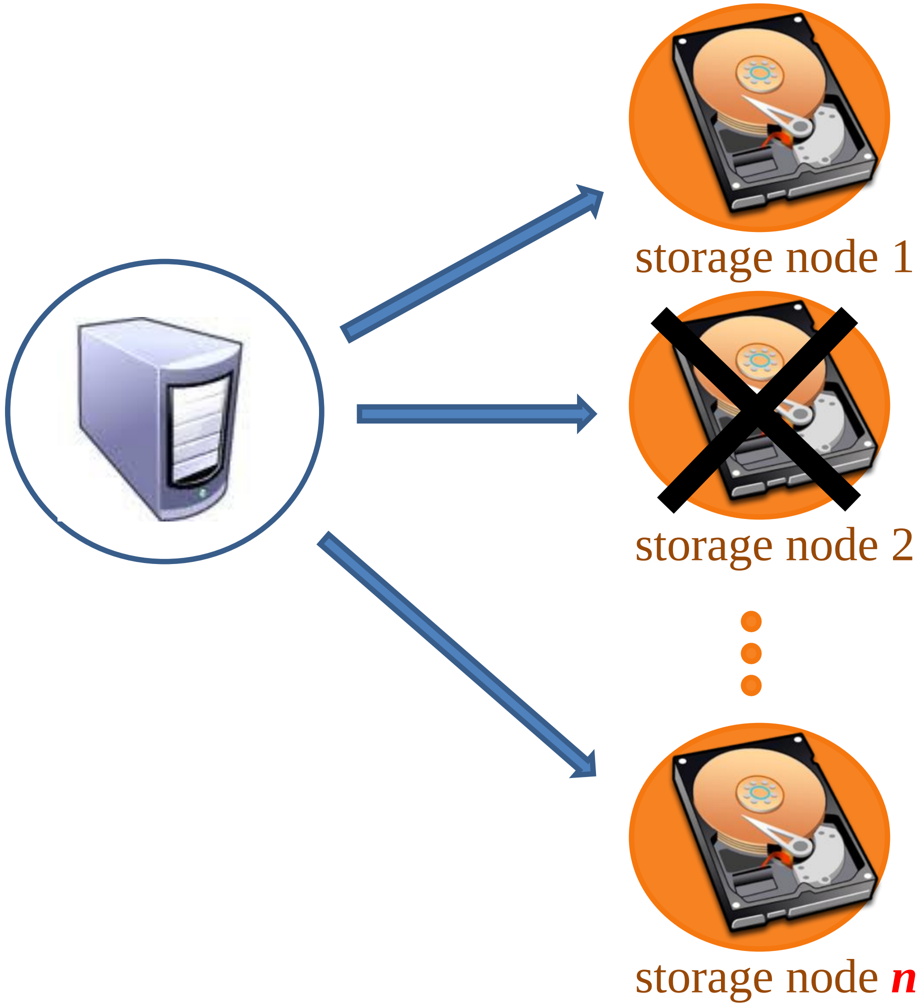
The system can tolerate any $n - k$ node failures

Node repair



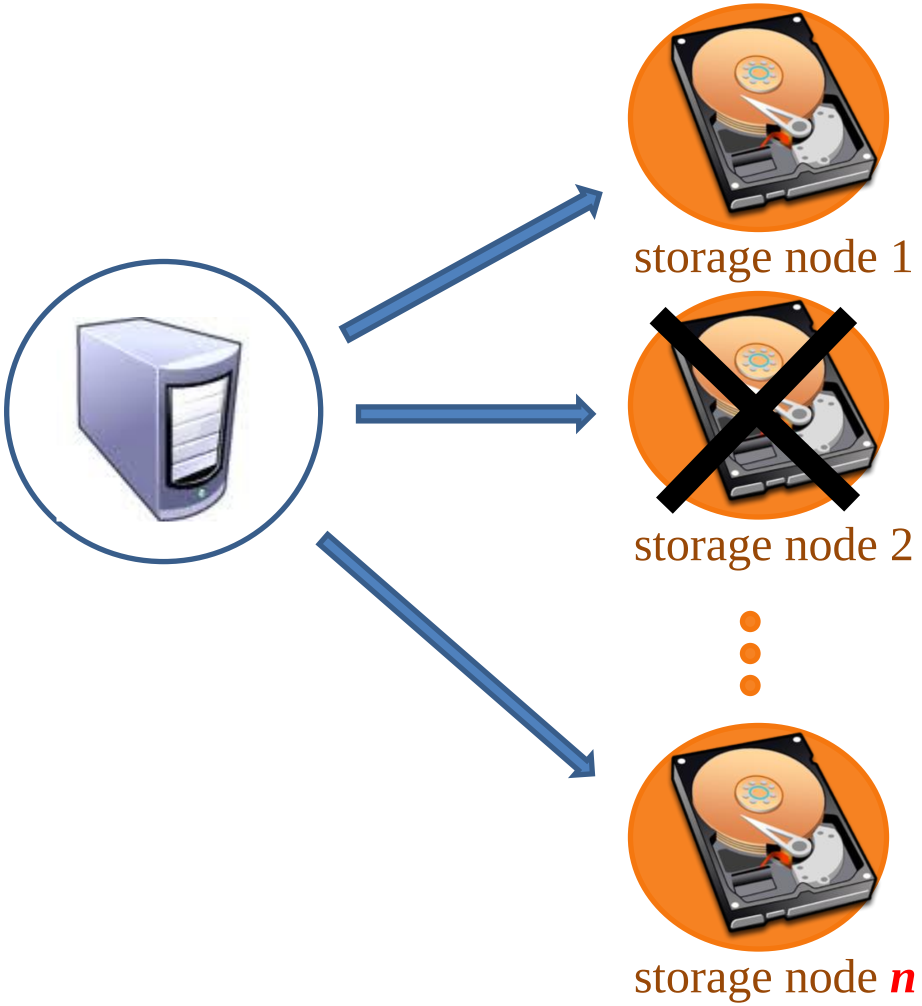
- If only **one** node fails, how to rebuild the redundancy?

Node repair



- If only **one** node fails, how to rebuild the redundancy?
- **Naïve method:** to reconstruct the whole data from k nodes

Node repair



- If only **one** node fails, how to rebuild the redundancy?
- **Naïve method:** to reconstruct the whole data from ***k*** nodes
- <u>Solution:</u> ***regenerating codes***[DimGodWaiRam07] for efficient single node repairs

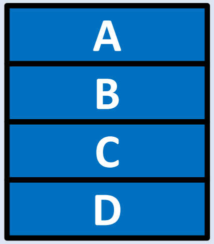
Regenerating codes – the idea

- To reduce the repair bandwidth, during a node repair
  - communicate with more nodes,
  - but download only part of their stored data

Regenerating codes – the idea

- To reduce the repair bandwidth, during a node repair
  - communicate with more nodes,
  - but download only part of their stored data

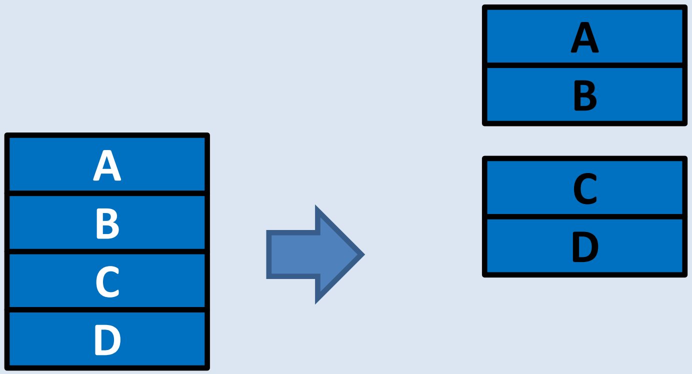
Example: (4,2) MDS code



Regenerating codes – the idea

- To reduce the repair bandwidth, during a node repair
  - communicate with more nodes,
  - but download only part of their stored data

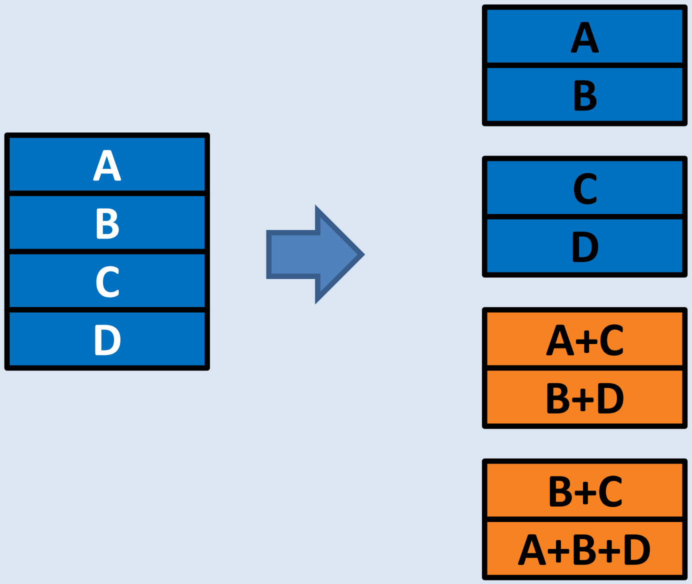
Example: (4,2) MDS code



Regenerating codes – the idea

- To reduce the repair bandwidth, during a node repair
  - communicate with more nodes,
  - but download only part of their stored data

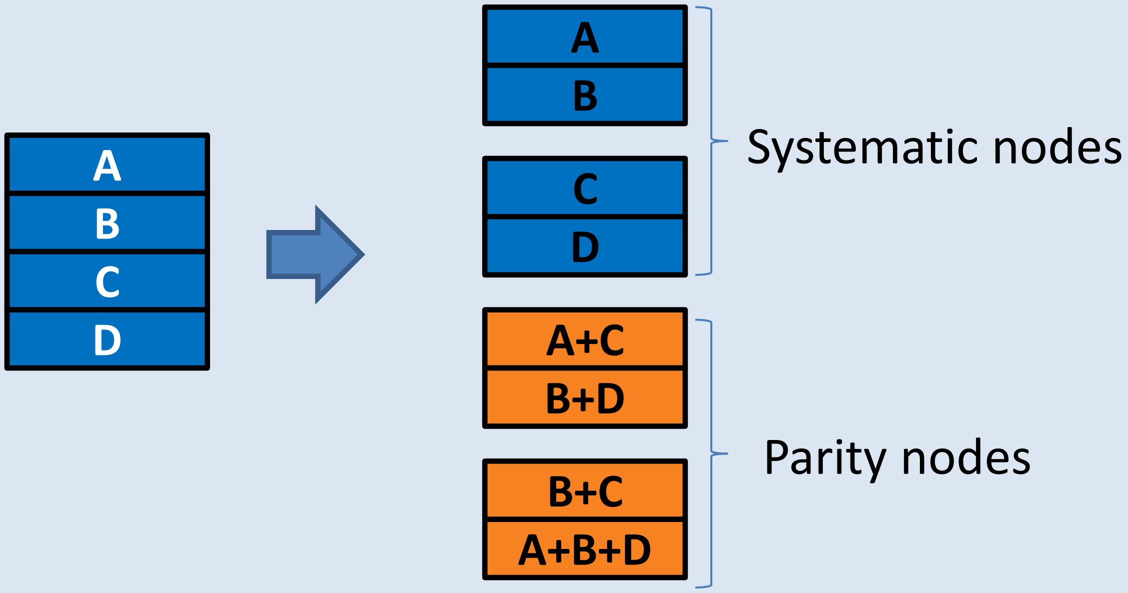
Example: (4,2) MDS code



Regenerating codes – the idea

- To reduce the repair bandwidth, during a node repair
  - communicate with more nodes,
  - but download only part of their stored data

Example: (4,2) MDS code

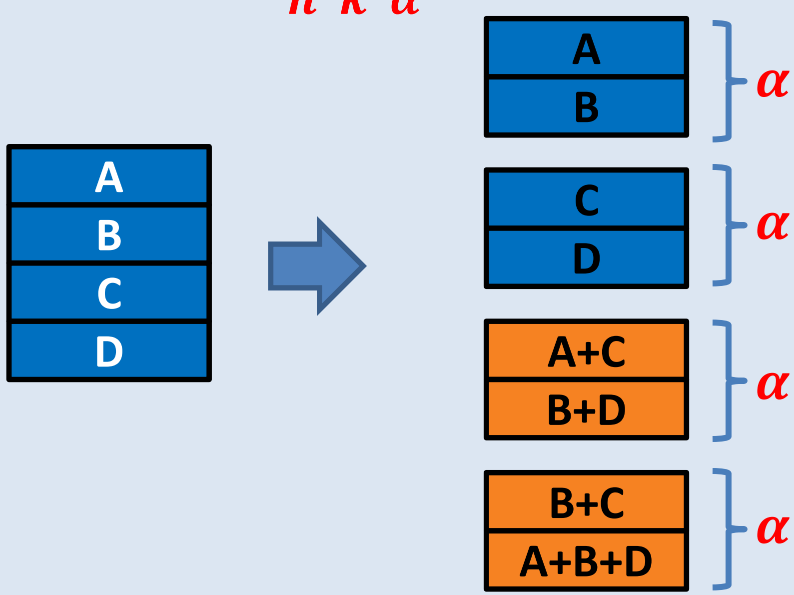


Regenerating codes – the idea

- To reduce the repair bandwidth, during a node repair
  - communicate with more nodes,
  - but download only part of their stored data

Example: (4,2,2) MDS code

$n\ k\ \alpha$

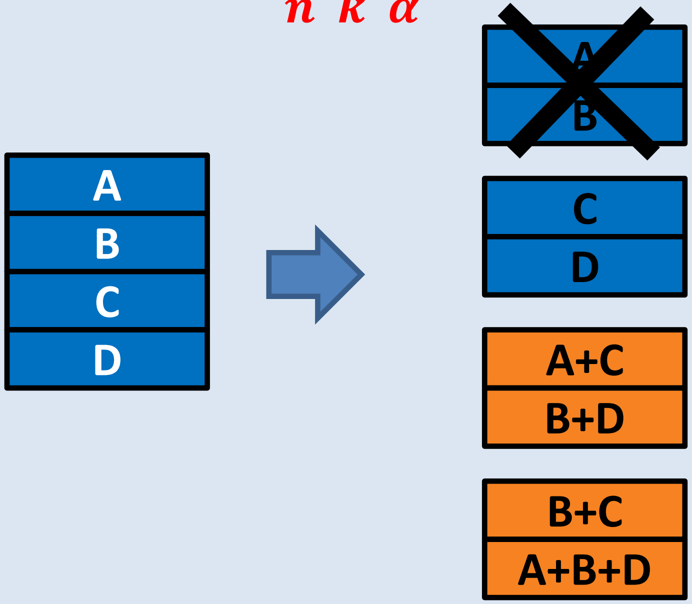


Regenerating codes – the idea

- To reduce the repair bandwidth, during a node repair
  - communicate with more nodes,
  - but download only part of their stored data

Example: (4,2,2) MDS code

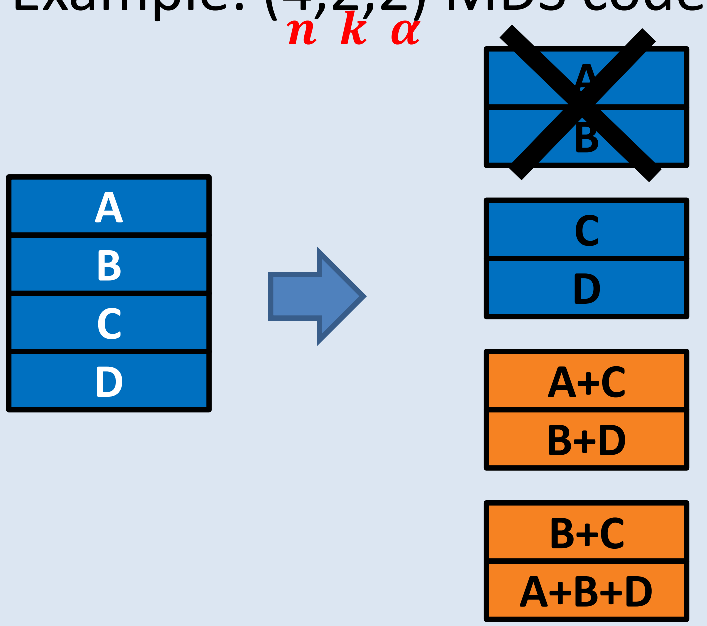
n k α



Regenerating codes – the idea

- To reduce the repair bandwidth, during a node repair
  - communicate with more nodes,
  - but download only part of their stored data

Example: (4,2,2) MDS code



Naïve repair:

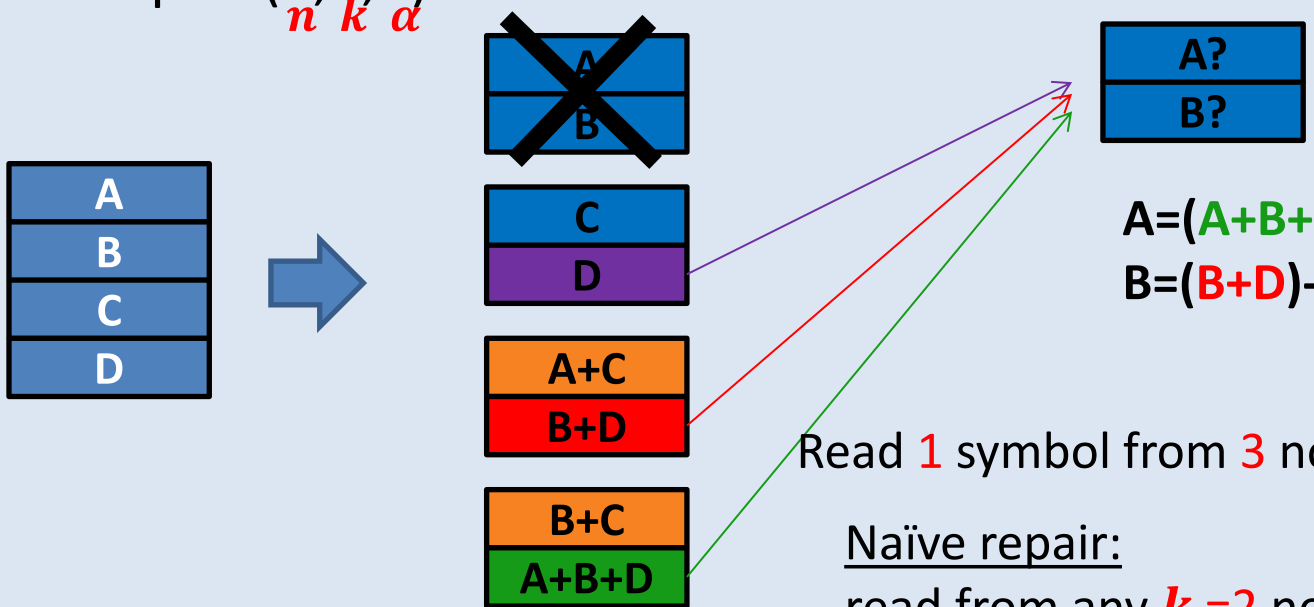
read from any k =2 nodes => **4** symbols

Regenerating codes – the idea

- To reduce the repair bandwidth, during a node repair
  - communicate with more nodes,
  - but download only part of their stored data

Example: (4,2,2) MDS code

n k α



A=(A+B+D)-(B+D)

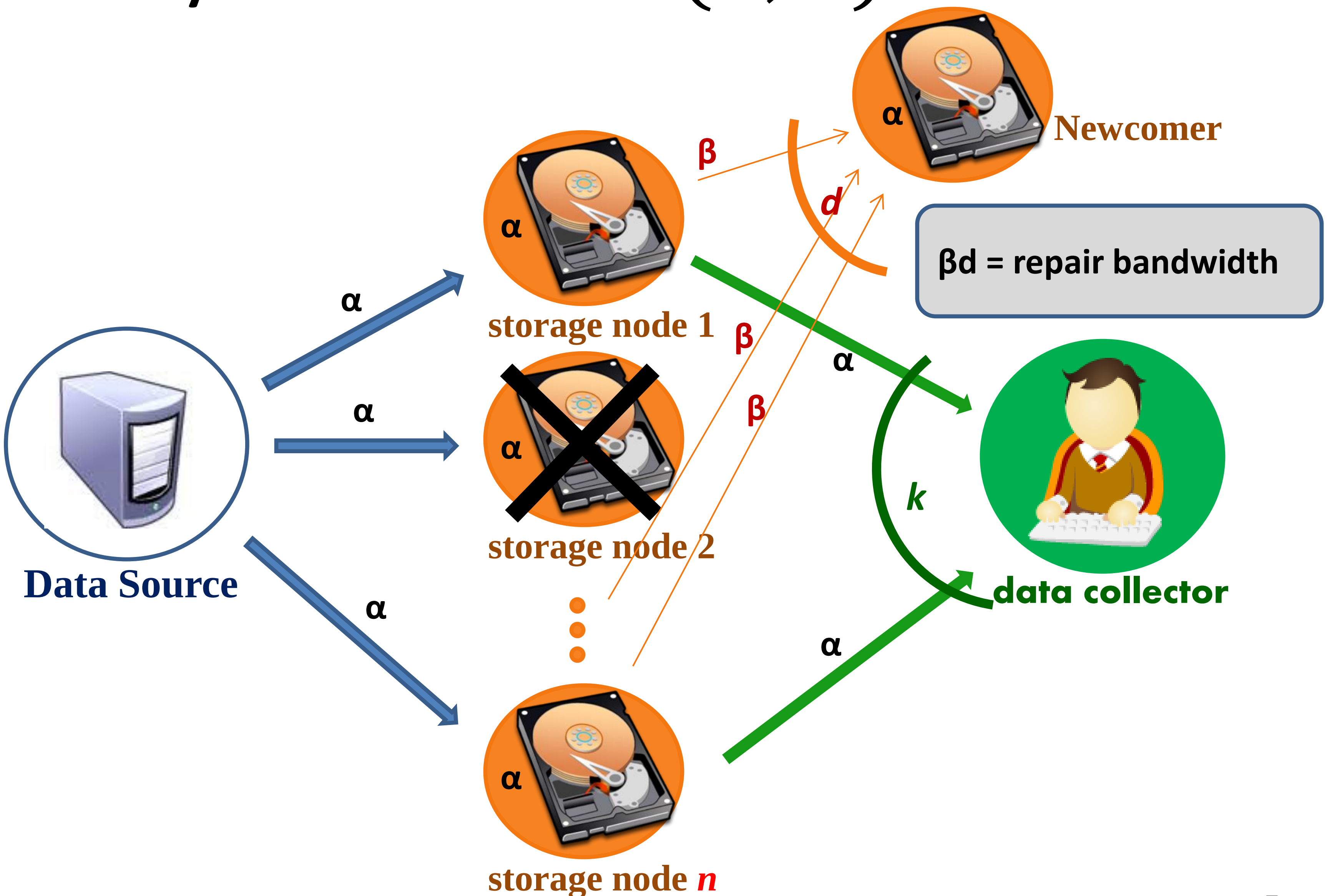
B=(B+D)-D

Read 1 symbol from 3 nodes => 3 symbols

<u>Naïve repair:</u>

read from any k =2 nodes => 4 symbols

System model: (n, k)-DSS



Storage-repair bandwidth tradeoff

- [DimGodWaiRam07] established a tradeoff between storage α vs. repair bandwidth $\gamma = \beta d$ for an (n, k) - DSS
- 2 extreme points of the trade-off:

MSR (minimum storage) : $(\alpha_{MSR}, \gamma_{MSR}) = \left(\frac{M}{k}, \frac{Md}{k(d-k+1)}\right)$

MBR (minimum bandwidth) : $(\alpha_{MBR}, \gamma_{MBR}) = \left(\frac{2Md}{2kd-k^2+k}, \frac{2Md}{2kd-k^2+k}\right)$

Storage-repair bandwidth tradeoff

- [DimGodWaiRam07] established a tradeoff between storage α vs. repair bandwidth $\gamma = \beta d$ for an (n, k) - DSS
- 2 extreme points of the trade-off:

MSR (minimum storage) : $(\alpha_{MSR}, \gamma_{MSR}) = \left(\frac{M}{k}, \frac{Md}{k(d-k+1)}\right)$

For $d = n-1$, γ_{MSR} is minimized:

$$\left(\alpha_{MSR}, \gamma^{min}{}_{MSR}\right) = \left(\frac{M}{k}, \frac{M}{k} \cdot \frac{n-1}{n-k}\right)$$

MBR (minimum bandwidth) : $(\alpha_{MBR}, \gamma_{MBR}) = \left(\frac{2Md}{2kd-k^2+k}, \frac{2Md}{2kd-k^2+k}\right)$

For $d = n-1$, γ_{MBR} is minimized:

$$\left(\alpha_{MBR}, \gamma^{min}{}_{MBR}\right) = \left(\frac{M}{k} \cdot \frac{2n-2}{2n-k-1}, \frac{M}{k} \cdot \frac{2n-2}{2n-k-1}\right)$$

Storage-repair bandwidth tradeoff

- [DimGodWaiRam07] established a tradeoff between storage α vs. repair bandwidth $\gamma = \beta d$ for an (n, k) - DSS
- 2 extreme points of the trade-off:

MSR (minimum storage) : $(\alpha_{MSR}, \gamma_{MSR}) = \left(\frac{M}{\cdots}, \frac{Md}{\cdots}\right)$

Each node sends a fraction $\frac{1}{n-k}$ of its stored symbols

For $d = n - 1$, γ_{MSR} is minimized:

$$\left(\alpha_{MSR}, {\gamma^{min}}_{MSR}\right) = \left(\frac{M}{k}, \frac{M}{k} \cdot \frac{n-1}{n-k}\right)$$

MBR (minimum bandwidth) : $(\alpha_{MBR}, \gamma_{MBR}) = \left(\frac{2Md}{2kd-k^2+k}, \frac{2Md}{2kd-k^2+k}\right)$

For $d = n - 1$, γ_{MBR} is minimized:

$$\left(\alpha_{MBR}, {\gamma^{min}}_{MBR}\right) = \left(\frac{M}{k} \cdot \frac{2n-2}{2n-k-1}, \frac{M}{k} \cdot \frac{2n-2}{2n-k-1}\right)$$

Storage-repair bandwidth tradeoff

- [DimGodWaiRam07] established a tradeoff between storage α vs. repair bandwidth $\gamma = \beta d$ for an (n, k) - DSS
- 2 extreme points of the trade-off:

MSR (minimum storage) : $(\alpha_{MSR}, \gamma_{MSR}) = ($ [illegible] $\frac{Md}{(d-k+1)})$

For $d = n - 1$, γ_{M} [illegible]

MDS array codes

$(\alpha_{MSR}, {\gamma^{min}}_{MSR}) = (\frac{M}{k}, \frac{M}{k} \cdot \frac{n-1}{n-k})$

MBR (minimum bandwidth) : $(\alpha_{MBR}, \gamma_{MBR}) = (\frac{2Md}{2kd-k^2+k}, \frac{2Md}{2kd-k^2+k})$

For $d = n - 1$, γ_{MBR} is minimized:

$$(\alpha_{MBR}, {\gamma^{min}}_{MBR}) = (\frac{M}{k} \cdot \frac{2n-2}{2n-k-1}, \frac{M}{k} \cdot \frac{2n-2}{2n-k-1})$$

EXAMPLE: HDFS RAID RS code vs. MSR/MBR

640 MB file => 10 blocks

(14,10) –RS code

1 2 3 4 5 6 7 8 9 10

P1 P2 P3 P4

Data size M = 640 MB

Storage nodes n = 14

DC connects to any k = 10 nodes

Newcomer node connects to d = 10 nodes

	RS code	MSR code	MBR code
Storage per node α	**64 MB**		
Repair bandwidth βd	**640 MB**		

EXAMPLE: HDFS RAID RS code vs. MSR/MBR

640 MB file => 10 blocks

(14,10) –RS code

1 2 3 4 5 6 7 8 9 10

P1 P2 P3 P4

Data size M = 640 MB

Storage nodes n = 14

DC connects to any k = 10 nodes

Newcomer node connects to d = 13 nodes

$$\left(\alpha_{MSR}, {\gamma^{min}}_{MSR}\right) = \left(\frac{M}{k}, \frac{M}{k} \cdot \frac{n-1}{n-k}\right)$$

$$\left(\alpha_{MBR}, {\gamma^{min}}_{MBR}\right) = \left(\frac{M}{k} \cdot \frac{2n-2}{2n-k-1}, \frac{M}{k} \cdot \frac{2n-2}{2n-k-1}\right)$$

	RS code	**MSR code**	**MBR code**
Storage per node α	64 MB	**64 MB**	**98 MB**
Repair bandwidth βd	640 MB	**208 MB**	**98 MB**

This talk:

Minimum storage regenerating (MSR) codes
with the following properties:

This talk:

Minimum storage regenerating (MSR) codes with the following properties:

1. Systematic (k nodes store the original data)

This talk:

Minimum storage regenerating (MSR) codes with the following properties:

1. Systematic (k nodes store the original data)
2. Minimized repair bandwidth of systematic nodes

$$d = n - 1; \left(\alpha_{MSR}, {\gamma^{min}}_{MSR}\right) = \left(\frac{M}{k}, \frac{M}{k} \cdot \frac{n-1}{n-k}\right)$$

This talk:

Minimum storage regenerating (MSR) codes with the following properties:

1. Systematic (k nodes store the original data)
2. Minimized repair bandwidth of systematic nodes

$$d = n - 1;\ \left(\alpha_{MSR}, {\gamma^{min}}_{MSR}\right) = \left(\frac{M}{k}, \frac{M}{k} \cdot \frac{n-1}{n-k}\right)$$

3. High rate $\frac{\boldsymbol{k}}{\boldsymbol{n}}$ (# of parity nodes $r = n - k$ is 2 or 3)

This talk:

Minimum storage regenerating (MSR) codes with the following properties:

1. Systematic (k nodes store the original data)
2. Minimized repair bandwidth of systematic nodes

$$d = n - 1;\ \left(\alpha_{MSR}, {\gamma^{min}}_{MSR}\right) = \left(\frac{M}{k}, \frac{M}{k} \cdot \frac{n-1}{n-k}\right)$$

3. High rate $\boldsymbol{\frac{k}{n}}$ (# of parity nodes $r = n - k$ is 2 or 3)
4. Small finite field

This talk:

Minimum storage regenerating (MSR) codes with the following properties:

1. Systematic (k nodes store the original data)
2. Minimized repair bandwidth of systematic nodes

$$d = n - 1;\ \left(\alpha_{MSR}, {\gamma^{min}}_{MSR}\right) = \left(\frac{M}{k}, \frac{M}{k} \cdot \frac{n-1}{n-k}\right)$$

3. High rate $\frac{\boldsymbol{k}}{\boldsymbol{n}}$ (# of parity nodes $r = n - k$ is 2 or 3)
4. Small finite field
5. Accessing the minimum number of helper node's symbols during repairs

   # of read symbols = # of transmitted symbols

This talk:

Minimum storage regenerating (MSR) codes with the following properties:

1. Systematic (k nodes store the original data)
2. Minimized repair bandwidth of systematic nodes

$$d = n - 1;\ (\alpha_{MSR}, {\gamma^{min}}_{MSR}) = \left(\frac{M}{k}, \frac{M}{k} \cdot \frac{n-1}{n-k}\right)$$

3. High rate $\boldsymbol{\frac{k}{n}}$ (# of parity nodes $r = n - k$ is 2 or 3)
4. Small finite field
5. Accessing the minimum number of helper node's symbols during repairs

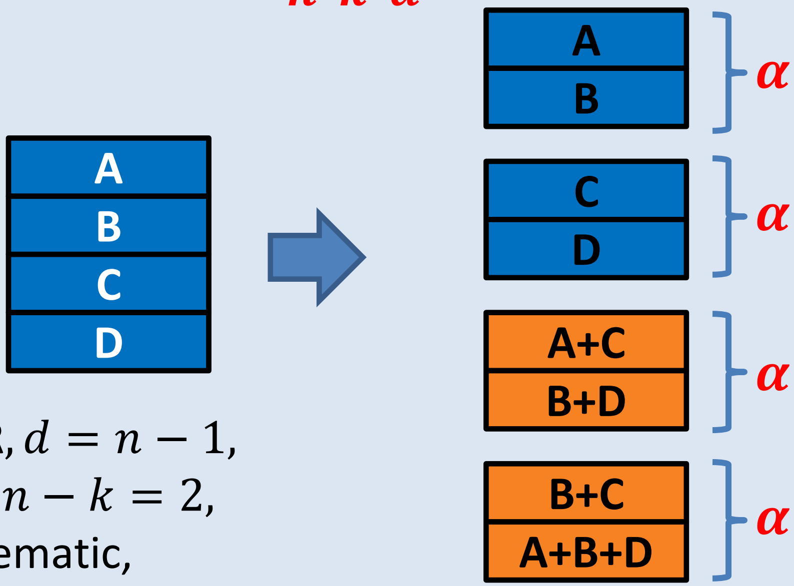
   # of read symbols = # of transmitted symbols
6. Minimum sub-packetization factor $\boldsymbol{\alpha = r^{k/r}}$

Optimal access codes

- Accessing the minimum number of helper node's symbols during repairs

  # of read symbols = # of transmitted symbols

Example: (4,2,2) MDS code
n k α



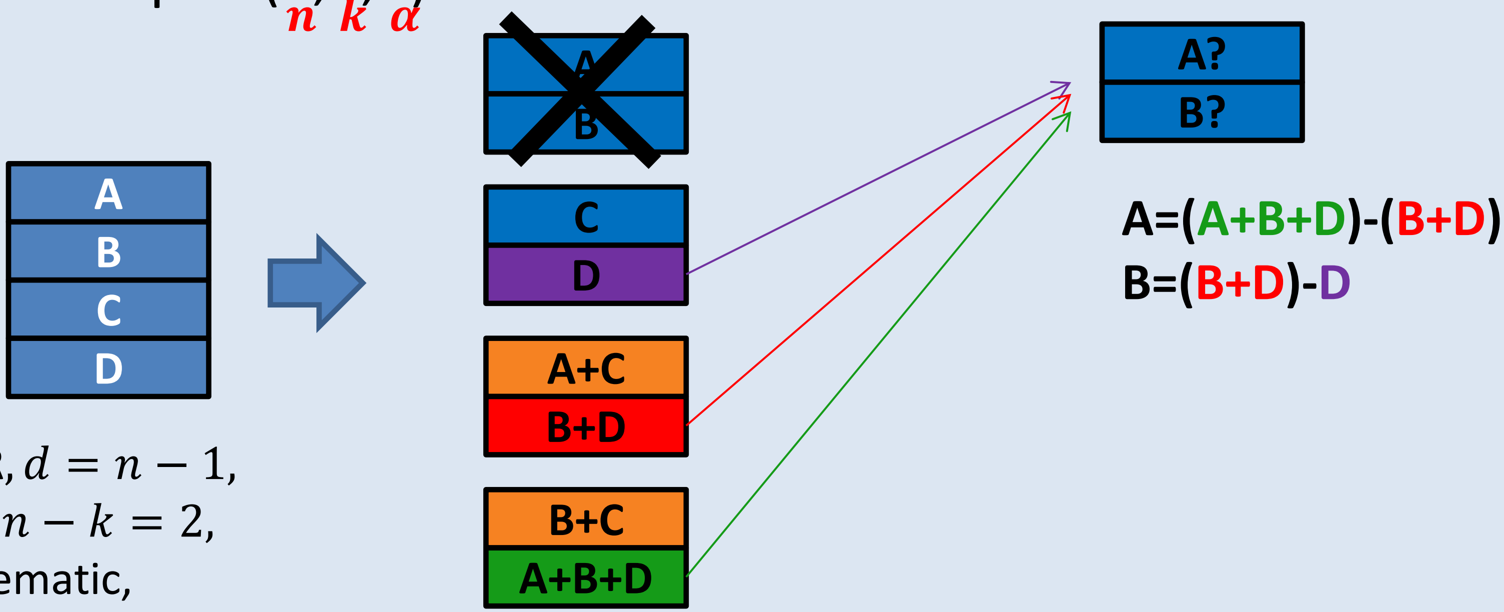
MSR, $d = n - 1$,
$r = n - k = 2$,
systematic,
binary

Optimal access codes

- Accessing the minimum number of helper node's symbols during repairs

  # of read symbols = # of transmitted symbols

Example: (4,2,2) MDS code (n, k, α)



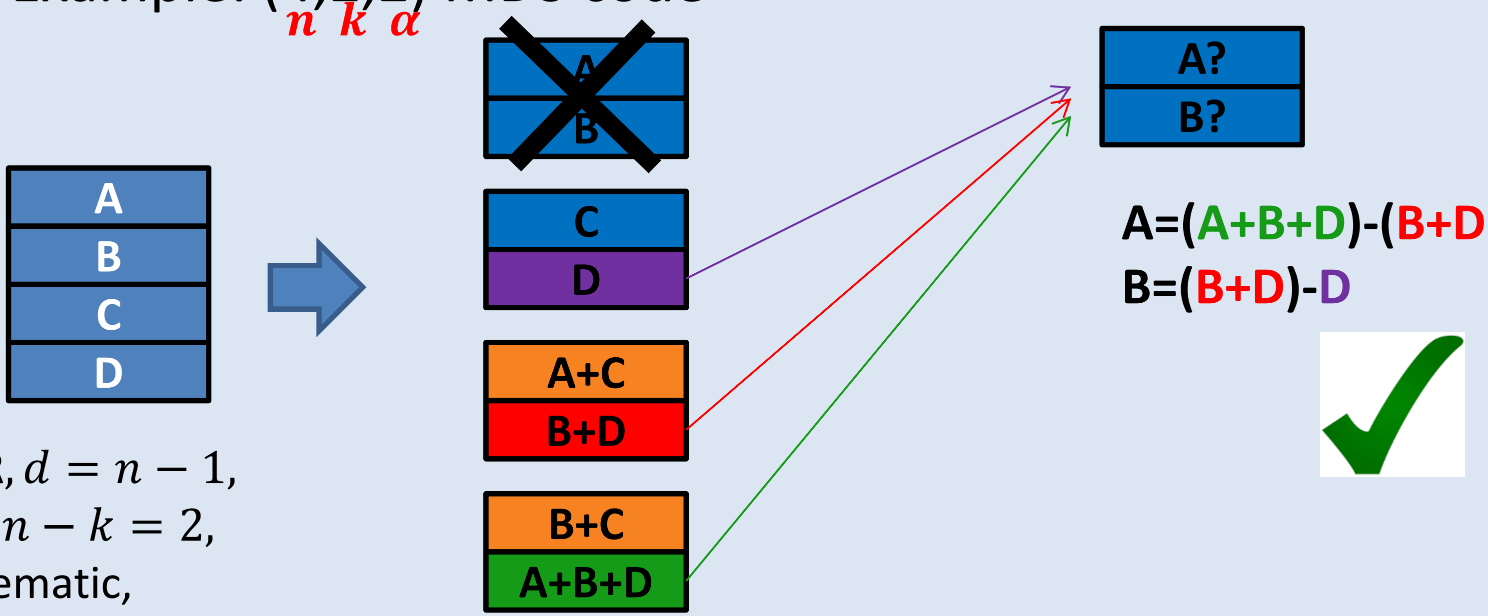
MSR, $d = n - 1$,
$r = n - k = 2$,
systematic,
binary

Optimal access codes

- Accessing the minimum number of helper node's symbols during repairs

  # of read symbols = # of transmitted symbols

Example: (4,2,2) MDS code



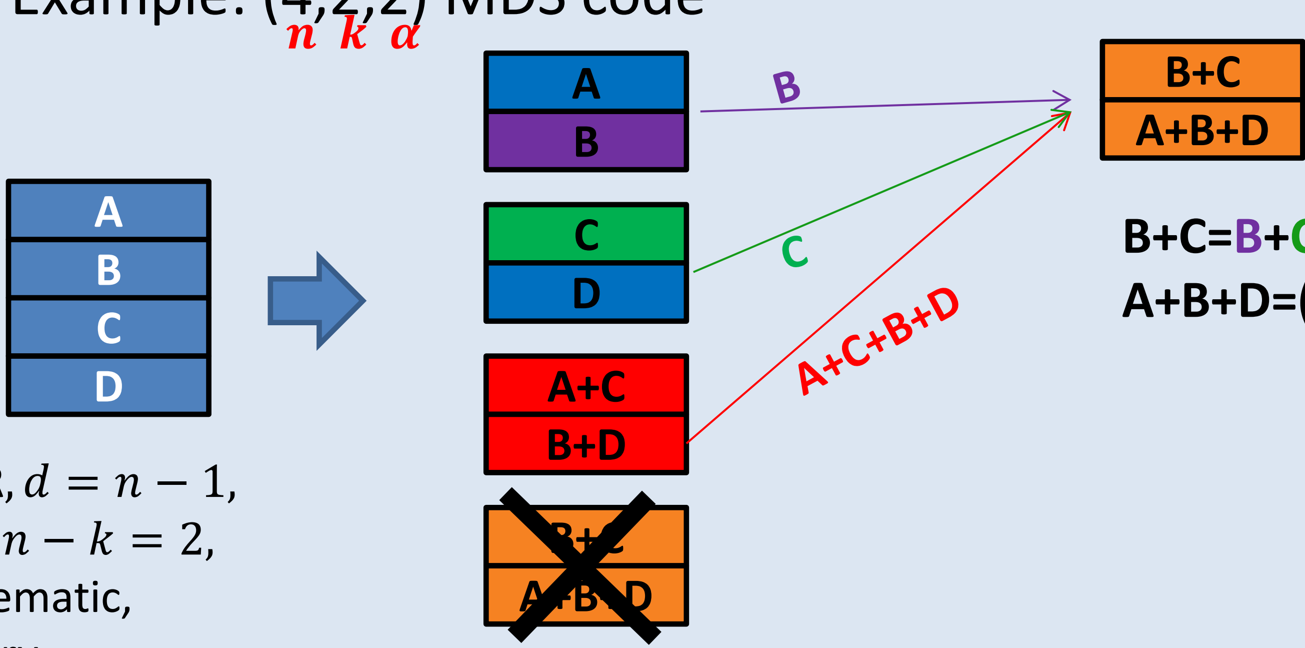
MSR, $d = n - 1$,
$r = n - k = 2$,
systematic,
binary

Optimal access codes

- Accessing the minimum number of helper node's symbols during repairs

# of read symbols = # of transmitted symbols

Example: (4,2,2) MDS code



B+C=B+C

A+B+D=(A+C+B+D)+ C

MSR, $d = n - 1$, $r = n - k = 2$, systematic, binary

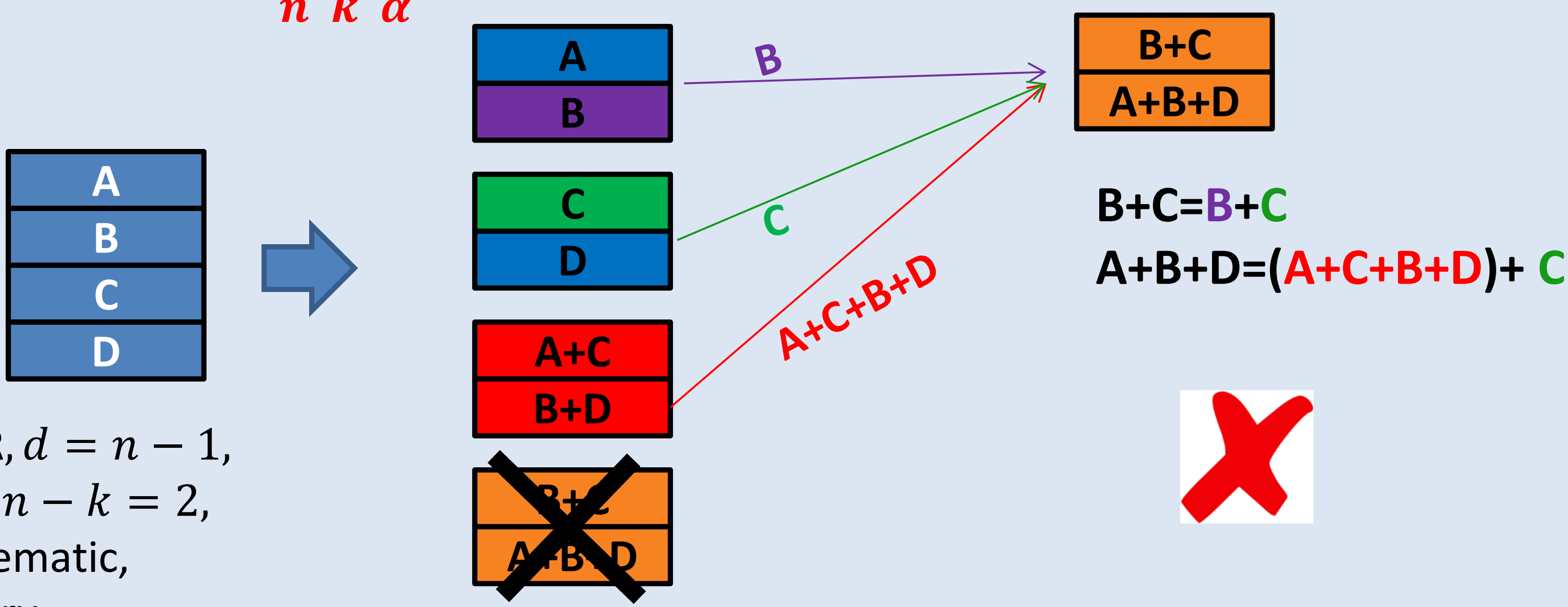
Optimal access codes

- Accessing the minimum number of helper node's symbols during repairs

  # of read symbols = # of transmitted symbols

Example: (4,2,2) MDS code

n k α



MSR, $d = n - 1$,
$r = n - k = 2$,
systematic,
binary

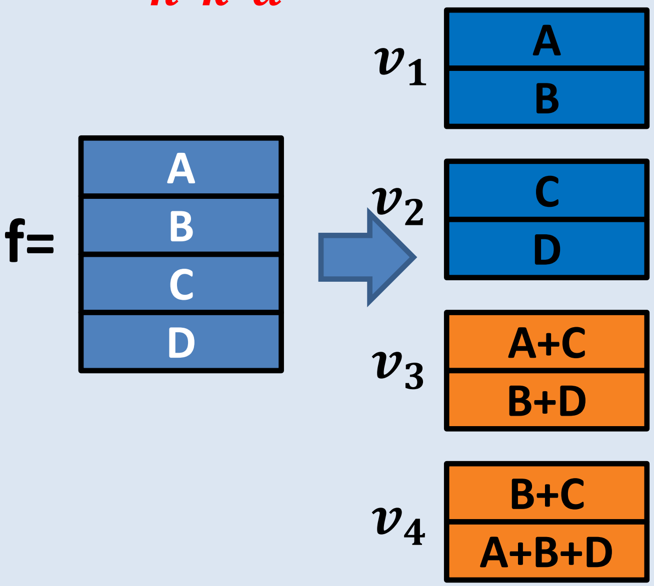
Our approach

- Subspace condition*
  – The problem of construction of MSR codes is described by the algebraic problem of construction of certain matrices and subspaces
- Graph theory
  – Perfect matchings in complete hypergraphs

* I. Tamo, Z. Wang, and J. Bruck, “Access versus bandwidth in codes for storage", IEEE Trans. Inf. Theory, 2014.

Example: Subspace condition

(4,2,2) MSR code
$n\ k\ \alpha$



Encoding:

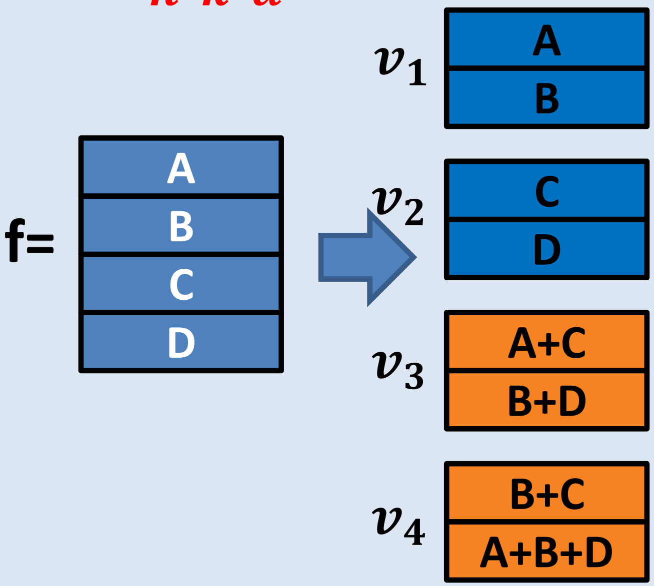
$$v_3 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} v_1 + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} v_2$$

$$v_4 = \begin{pmatrix} \mathbf{0} & \mathbf{1} \\ \mathbf{1} & \mathbf{1} \end{pmatrix} v_1 + \begin{pmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} \end{pmatrix} v_2$$

$$\mathbf{f}\begin{pmatrix} \mathrm{I} & 0 & \mathrm{I} & A_1 \\ 0 & \mathrm{I} & \mathrm{I} & A_2 \end{pmatrix}$$

Example: Subspace condition

(4,2,2) MSR code
$n \; k \; \alpha$



Encoding:

$$v_3 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} v_1 + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} v_2$$

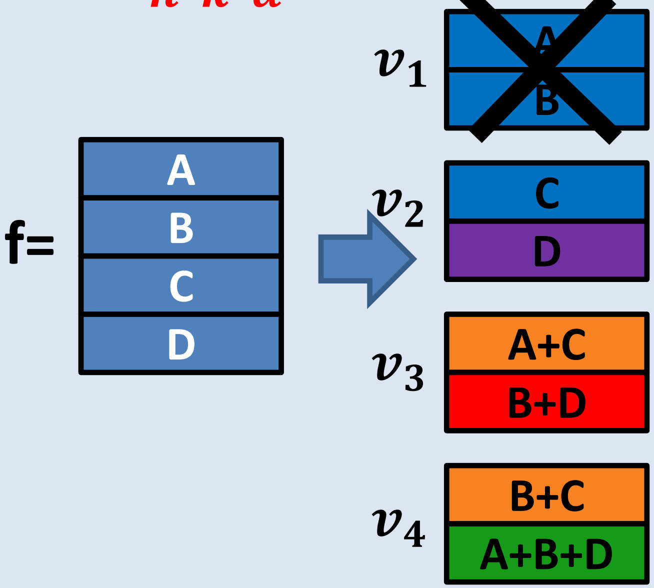
$$v_4 = \begin{pmatrix} \mathbf{0} & \mathbf{1} \\ \mathbf{1} & \mathbf{1} \end{pmatrix} v_1 + \begin{pmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} \end{pmatrix} v_2$$

$$\mathbf{f}\begin{pmatrix} \mathrm{I} & 0 & \boxed{\begin{matrix} \mathrm{I} & A_1 \\ \mathrm{I} & A_2 \end{matrix}} \\ 0 & \mathrm{I} & \end{pmatrix}$$

A_1, A_2, $A_1 - A_2$ **are invertible**

Example: Subspace condition

(4,2,2) MSR code
n k α



Encoding:

$$v_3 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} v_1 + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} v_2$$

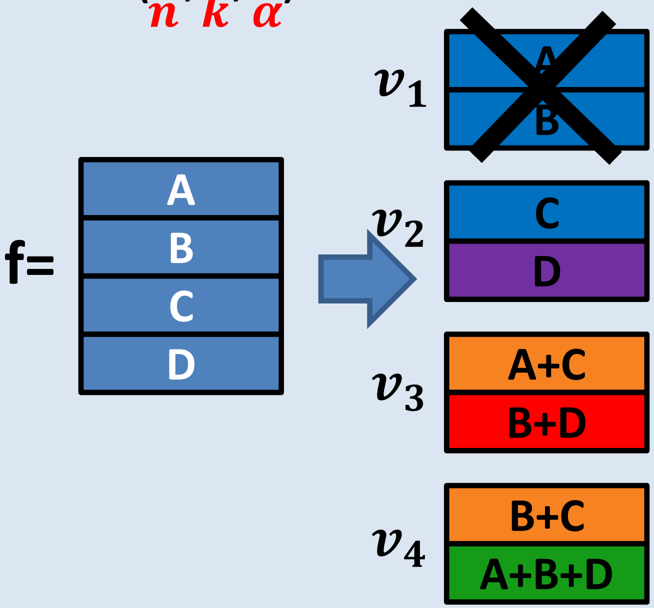
$$v_4 = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} v_1 + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} v_2$$

$$\mathbf{f}\begin{pmatrix} I & 0 & I & A_1 \\ 0 & I & I & A_2 \end{pmatrix}$$

A_1, A_2, $A_1 - A_2$ **are invertible**

Example: Subspace condition

(4,2,2) MSR code
$n \; k \; \alpha$



Encoding:

$$v_3 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} v_1 + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} v_2$$

$$v_4 = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} v_1 + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} v_2$$

$$\mathbf{f}\begin{pmatrix} I & 0 & I & A_1 \\ 0 & I & I & A_2 \end{pmatrix}$$

A_1, A_2, $A_1 - A_2$ are invertible

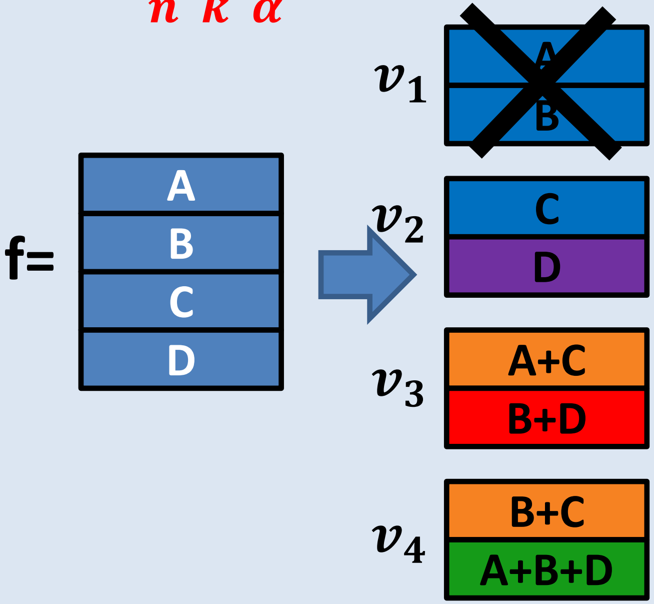
To repair node 1, node j sends $S_{1j}v_j$ and new node 1 gets

$$\begin{pmatrix} S_{12}v_2 \\ S_{13}v_3 \\ S_{14}v_4 \end{pmatrix} = \begin{pmatrix} S_{12}v_2 \\ S_{13}(v_1+v_2) \\ S_{14}(A_1v_1 + A_2v_2) \end{pmatrix} = \begin{pmatrix} 0 & S_{12} \\ S_{13} & S_{13} \\ S_{14}A_1 & S_{14}A_2 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$$

$S_{12} = S_{13} = S_{14} = (0,1)$

Example: Subspace condition

(4,2,2) MSR code
$n\ k\ \alpha$



Encoding:

$$v_3 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} v_1 + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} v_2$$

$$v_4 = \begin{pmatrix} \mathbf{0} & \mathbf{1} \\ \mathbf{1} & \mathbf{1} \end{pmatrix} v_1 + \begin{pmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} \end{pmatrix} v_2$$

$$\mathbf{f}\begin{pmatrix} \mathrm{I} & 0 & \boxed{\begin{matrix}\mathrm{I} & A_1 \\ \mathrm{I} & A_2\end{matrix}} \\ 0 & \mathrm{I} & \end{pmatrix}$$

A_1, A_2, $A_1 - A_2$ are invertible

To repair node 1, node j sends $S_{1j}v_j$ and new node 1 gets

$$\begin{pmatrix} S_{12}v_2 \\ S_{13}v_3 \\ S_{14}v_4 \end{pmatrix} = \begin{pmatrix} S_{12}v_2 \\ S_{13}(v_1+v_2) \\ S_{14}(A_1v_1 + A_2v_2) \end{pmatrix} = \begin{pmatrix} 0 & S_{12} \\ S_{13} & S_{13} \\ S_{14}A_1 & S_{14}A_2 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$$

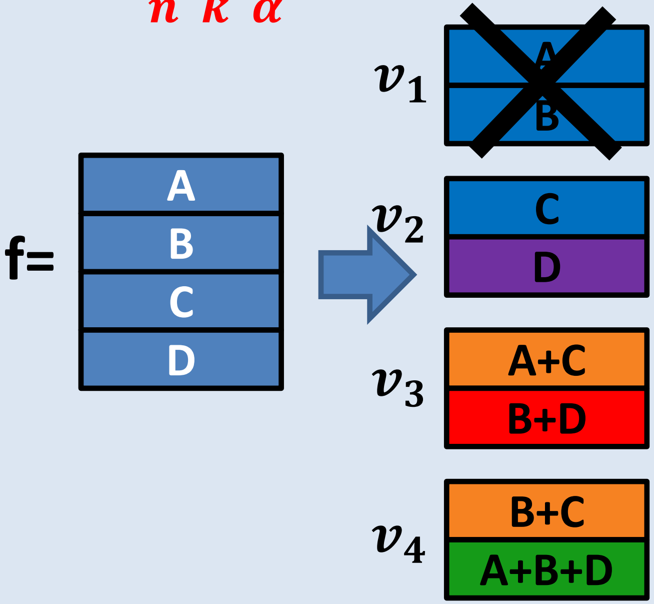
$S_{12} = S_{13} = S_{14} =$(0,1)

To recover v_1 it should hold

1. rank$\begin{pmatrix} S_{13} \\ S_{14}A_1 \end{pmatrix} = 2$
2. rank$\begin{pmatrix} S_{12} \\ S_{13} \\ S_{14}A_2 \end{pmatrix} = 1$

Example: Subspace condition

(4,2,2) MSR code
n k α



Encoding:

$$v_3 = \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix} v_1 + \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix} v_2$$

$$v_4 = \begin{pmatrix}\mathbf{0} & \mathbf{1}\\ \mathbf{1} & \mathbf{1}\end{pmatrix} v_1 + \begin{pmatrix}\mathbf{1} & \mathbf{0}\\ \mathbf{0} & \mathbf{1}\end{pmatrix} v_2$$

$$\mathbf{f}\left(\begin{matrix}\mathrm{I} & 0\\ 0 & \mathrm{I}\end{matrix}\;\boxed{\begin{matrix}\mathrm{I} & A_1\\ \mathrm{I} & A_2\end{matrix}}\right)$$

A_1, A_2, $A_1 - A_2$ are invertible

To repair node 1, node j sends $S_{1j}v_j$ and new node 1 gets

$$\begin{pmatrix} S_{12}v_2 \\ S_{13}v_3 \\ S_{14}v_4 \end{pmatrix} = \begin{pmatrix} S_{12}v_2 \\ S_{13}(v_1+v_2) \\ S_{14}(A_1v_1 + A_2v_2) \end{pmatrix} = \begin{pmatrix} 0 & S_{12} \\ S_{13} & S_{13} \\ S_{14}A_1 & S_{14}A_2 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$$

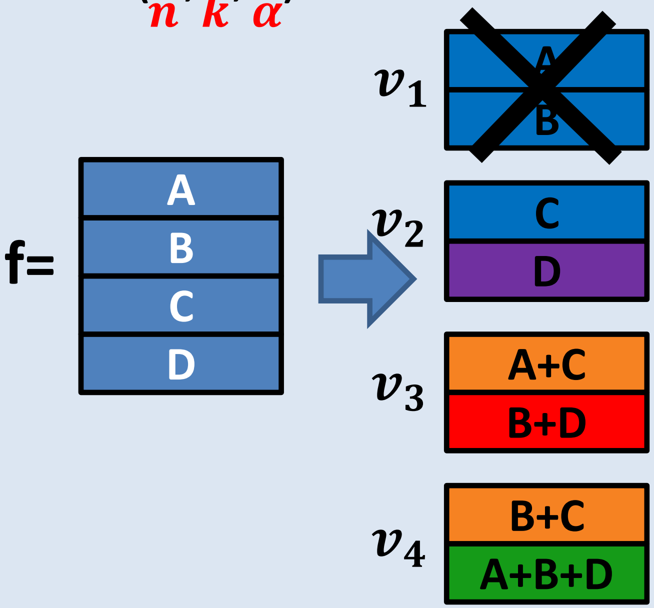
$S_{12} = S_{13} = S_{14} =(0,1)$

To recover v_1 it should hold

1. $\mathrm{rank}\begin{pmatrix} S_{13} \\ S_{14}A_1 \end{pmatrix} = 2 \quad \Leftrightarrow \quad <S_{13}> \cap < S_{14}A_1 >=\{0\} \Leftrightarrow <S_{13}> + < S_{14}A_1 >=\mathbb{F}_2^2$
2. $\mathrm{rank}\begin{pmatrix} S_{12} \\ S_{13} \\ S_{14}A_2 \end{pmatrix} = 1 \quad \Leftrightarrow \quad < S_{12} >=<S_{13}>=< S_{14}A_2>$

Example: Subspace condition

(4,2,2) MSR code
n k α



Encoding:

$$v_3 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} v_1 + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} v_2$$

$$v_4 = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} v_1 + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} v_2$$

$$\mathbf{f}\begin{pmatrix} I & 0 & I & A_1 \\ 0 & I & I & A_2 \end{pmatrix}$$

A_1, A_2, $A_1 - A_2$ **are invertible**

$S_{12} = S_{13} = S_{14} =(0,1)$

To repair node 1, node j sends $S_{1j} v_j$ and new node 1 gets

$$\begin{pmatrix} S_{12} v_2 \\ S_{13} v_3 \\ S_{14} v_4 \end{pmatrix} = \begin{pmatrix} S_{12} v_2 \\ S_{13}(v_1 + v_2) \\ S_{14}(A_1 v_1 + A_2 v_2) \end{pmatrix} = \begin{pmatrix} 0 & S_{12} \\ S_{13} & S_{13} \\ S_{14} A_1 & S_{14} A_2 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$$

To recover v_1 it should hold

1. $\operatorname{rank}\begin{pmatrix} S_{13} \\ S_{14} A_1 \end{pmatrix} = 2 \Leftrightarrow$ $<S_{13}> \cap < S_{14} A_1 >=\{0\} \Leftrightarrow <S_{13}> + < S_{14} A_1 >=\mathbb{F}_2^2$
2. $\operatorname{rank}\begin{pmatrix} S_{12} \\ S_{13} \\ S_{14} A_2 \end{pmatrix} = 1 \Leftrightarrow$ $< S_{12} >=< S_{13} >=< S_{14} A_2 >$

Subspace condition

Subspace condition for $(n, k, \alpha) - MSR$ code

- **Encoding:** a file $f \in \mathbb{F}_q^{k\alpha}$ is partitioned into k parts of size α:

$$\mathbf{f} = (v_1, v_2, \dots, v_k), v_i \in \mathbb{F}_q^{\alpha}$$

$$\mathbf{f} \to \mathbf{f} \cdot G \quad = (C_1, \dots, C_n) \quad ,$$

where $G = \begin{pmatrix} \mathrm{I} & & & \mathrm{I} & A_{11} \cdots A_{(n-k-1)1} \\ & \ddots & & \vdots & \vdots \quad \ddots \quad \vdots \\ & & \mathrm{I} & \mathrm{I} & A_{1k} \cdots A_{(n-k-1)k} \end{pmatrix}$, $A_{ij} \in \mathbb{F}_q^{\alpha \times \alpha}$

$$C_j = v_j, 1 \le j \le k$$

$$C_{k+i} = \sum_{j=1}^{k} A_{ij} C_j, 1 \le i \le n - k$$

Subspace condition for $(n,k,\alpha)-MSR$ code

- **Encoding:** a file $f \in \mathbb{F}_q^{k\alpha}$ is partitioned into k parts of size α:

$$\mathbf{f} = (v_1, v_2, \dots, v_k), v_i \in \mathbb{F}_q^{\alpha}$$

$$\mathbf{f} \to \mathbf{f} \cdot G \quad = (C_1, \dots, C_n) \quad ,$$

where $G = \begin{pmatrix} \mathrm{I} & & & \mathrm{I} & A_{11} & \cdots & A_{(n-k-1)1} \\ & \ddots & & \vdots & \vdots & \ddots & \vdots \\ & & \mathrm{I} & \mathrm{I} & A_{1k} & \cdots & A_{(n-k-1)k} \end{pmatrix}$, $A_{ij} \in \mathbb{F}_q^{\alpha\times\alpha}$

G'

$$C_j = v_j, 1 \le j \le k$$

$$C_{k+i} = \sum_{j=1}^{k} A_{ij} C_j, 1 \le \ i \le \ n-k$$

Every square block submatrix of G' is **invertible**

Subspace condition for $(n, k, \alpha) - MSR$ code

- **Encoding:** a file $f \in \mathbb{F}_q^{k\alpha}$ is partitioned into k parts of size α:

$$\mathbf{f} = (v_1, v_2, \ldots, v_k), v_i \in \mathbb{F}_q^{\alpha}$$

$$\mathbf{f} \to \mathbf{f} \cdot G \quad = (C_1, \ldots, C_n) \quad ,$$

where $G = \left(\begin{array}{ccc|cccc} \mathrm{I} & & & \mathrm{I} & A_{11} & \cdots & A_{(n-k-1)1} \\ & \ddots & & \vdots & \vdots & \ddots & \vdots \\ & & \mathrm{I} & \mathrm{I} & A_{1k} & \cdots & A_{(n-k-1)k} \end{array} \right)$, $A_{ij} \in \mathbb{F}_q^{\alpha \times \alpha}$

G'

$$C_j = v_j, 1 \leq j \leq k$$

$$C_{k+i} = \sum_{j=1}^{k} A_{ij} C_j, 1 \leq \ i \leq \ n - k$$

Every square block submatrix of G' is **invertible**

The nonsingular property

Subspace condition for $(n, k, \alpha) - MSR$ code

- **Repair of node t:**

  node $j \in [n] \backslash \{\text{t}\}$ sends $S_t\, C_j$, where $S_t \in \mathbb{F}_q^{\frac{\alpha}{r} \times \alpha}$, $r = n - k$.

  $\mathbb{S}_t := < S_t >$ is called the repairing subspace

  (the repairing subspace are independent on the helper node*)

  $\mathbb{S}_t \subseteq \mathbb{F}_q^{\alpha}$, $\dim(\mathbb{S}_t) = \frac{\alpha}{r}$.

* I. Tamo, Z. Wang, and J. Bruck, "Access versus bandwidth in codes for storage", IEEE Trans. Inf. Theory, 2014.

Subspace condition for $(n, k, \alpha) - MSR$ code

- **Repair of node t:**

  node $j \in [n] \backslash \{t\}$ sends $S_t\, C_j$, where $S_t \in \mathbb{F}_q^{\frac{\alpha}{r} \times \alpha}, r = n - k$.

  $\mathbb{S}_t := < S_t >$ is called the repairing subspace

  (the repairing subspace are independent on the helper node*)

  $\mathbb{S}_t \subseteq \mathbb{F}_q^{\alpha}, \dim(\mathbb{S}_t) = \frac{\alpha}{r}$.

The optimal access property ⇔ every $\mathbb{S}_t = < e_{i_1}, \dots, e_{i_{\frac{l}{r}}} >$

* I. Tamo, Z. Wang, and J. Bruck, “Access versus bandwidth in codes for storage", IEEE Trans. Inf. Theory, 2014.

Subspace condition for $(n, k, \alpha) - MSR$ code

- **Repair of node t:**

node $j \in [n]\backslash\{\mathrm{t}\}$ sends $S_t\, C_j$, where $S_t \in \mathbb{F}_q^{\frac{\alpha}{r} \times \alpha}$, $r = n - k$.

$\mathbb{S}_t :=< S_t >$ is called the repairing subspace

(the repairing subspace are independent on the helper node*)

$\mathbb{S}_t \subseteq \mathbb{F}_q^{\alpha}$, $\dim(\mathbb{S}_t) = \frac{\alpha}{r}$.

The optimal access property $\Leftrightarrow$ every $\mathbb{S}_t =< e_{i_1}, \ldots, e_{i_{\frac{l}{r}}} >$

1. $\mathbb{S}_t + \mathbb{S}_t A_{1t} + \ldots + \mathbb{S}_t A_{(r-1)t} = \mathbb{F}_q^{\alpha}$

The independence property

2. $\mathbb{S}_t = \mathbb{S}_t A_{ij}, j \neq t,$ where $\mathbb{S}_t A_{it} := \mathrm{rs}(S_t A_{it})$

The invariance property

* I. Tamo, Z. Wang, and J. Bruck, “Access versus bandwidth in codes for storage", IEEE Trans. Inf. Theory, 2014.

Sufficient Condition

- **Theorem** [TWB14]: Let α and r be integers s.t. $r|\alpha$. If there exist subspaces $\mathbb{S}_1, \dots, \mathbb{S}_k \subseteq \mathbb{F}_q^{\alpha}$ of dimension α/r and encoding matrices $A_{ij} \in \mathbb{F}_q^{\alpha\times\alpha}, i \in [r-1], j \in [k]$ which satisfy

1. The nonsingular property:

   Every square block submatrix of $\begin{pmatrix} I & A_{11} & \cdots & A_{(r-1)1} \\ \vdots & \vdots & \ddots & \vdots \\ I & A_{1k} & \cdots & A_{(r-1)k} \end{pmatrix}$ is invertible

2. The independence property:

$$\mathbb{S}_t + \mathbb{S}_t A_{1t} + \dots + \mathbb{S}_t A_{(r-1)t} = \mathbb{F}_q^{\alpha}$$

3. The invariance property:

$$\mathbb{S}_t = \mathbb{S}_t A_{ij}, j \neq t$$

Then the corresponding code is an (n, k, α) −MSR code.

Sub-Packetization Bound

- Node capacity α symbols = sub-packetization factor
- Optimal bandwidth => $\boldsymbol{\alpha > 1}$

Sub-Packetization Bound

- Node capacity α symbols = sub-packetization factor
- Optimal bandwidth => $\boldsymbol{\alpha} > \mathbf{1}$
- **Q:** For a given (n, k), what is the minimum node capacity α required to achieve the optimal bandwidth and optimal access?

Sub-Packetization Bound

- Node capacity α symbols = sub-packetization factor
- Optimal bandwidth => $\boldsymbol{\alpha} > \mathbf{1}$
- **Q:** For a given (n, k), what is the minimum node capacity α required to achieve the optimal bandwidth and optimal access?

- For a given node capacity of α symbols and a given number of parity nodes r, what is the maximum number of systematic nodes k for an optimal access MSR code?

Sub-Packetization Bound

- Node capacity α symbols = sub-packetization factor
- Optimal bandwidth => $\boldsymbol{\alpha > 1}$
- **Q:** For a given (n, k), what is the minimum node capacity α required to achieve the optimal bandwidth and optimal access?

- For a given node capacity of α symbols and a given number of parity nodes r, what is the maximum number of systematic nodes k for an optimal access MSR code?
- **Theorem** [TWB14]**:** Let k be the largest number of systematic nodes in an $(k+r, k, \alpha)$ optimal access MSR code, then

$$k = r\log_r \alpha$$

Our goal

For $k = r\log_r \alpha,$ construct sets of

- subspaces $\mathbb{S}_1, \ldots, \mathbb{S}_k \subseteq \mathbb{F}_q^{\alpha}$ of dimension α/r spanned by vectors of the standard basis
- Invertible matrices $A_{ij} \in \mathbb{F}_q^{\alpha\times\alpha}, 1 \leq i \leq r-1, 1 \leq j \leq k$

which satisfy

1. The nonsingular property: every square block submatrix of
$$\begin{pmatrix} \mathrm{I}\, A_{11} \cdots A_{(r-1)1} \\ \vdots \quad \vdots \quad \ddots \quad \vdots \\ \mathrm{I} A_{1k} \cdots A_{(r-1)k} \end{pmatrix}$$ is invertible
2. The independence property: $\mathbb{S}_t + \mathbb{S}_t A_{1t} + \ldots + \mathbb{S}_t A_{(r-1)t} = \mathbb{F}_q^{\alpha}$
3. The invariance property: $\mathbb{S}_t = \mathbb{S}_t A_{ij}, j \neq t$

for a field size q as small as possible.

In particular for $A_{ij} = (A_j)^i$:

For $k = r\log_r \alpha$, construct a set $\{(\boldsymbol{A_i}, \mathbb{S}_i)\}_{\mathbf{i=1}}^{\mathbf{k}}$ of

- subspaces $\mathbb{S}_1, \dots, \mathbb{S}_k \subseteq \mathbb{F}_q^\alpha$ of dimension α/r spanned by vectors of the standard basis
- Invertible matrices $A_1, \dots, A_k \in \mathbb{F}_q^{\alpha\times\alpha}$

which satisfy

1. The nonsingular property: every square block submatrix of

$$\begin{pmatrix} \mathrm{I} & A_1 & \cdots & {A_1}^{r-1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{I} & A_k & \cdots & {A_k}^{r-1} \end{pmatrix}$$ is invertible

2. The independence property:

$$\mathbb{S}_t + \mathbb{S}_t A_t + \mathbb{S}_t A_t^2 + \dots + \mathbb{S}_t\, {A_t}^{r-1} = \mathbb{F}_q^\alpha$$

3. The invariance property: $\mathbb{S}_t = \mathbb{S}_t A_j, j \neq t$

for a field size q as small as possible.

In particular for $A_{ij} = (A_j)^i$:

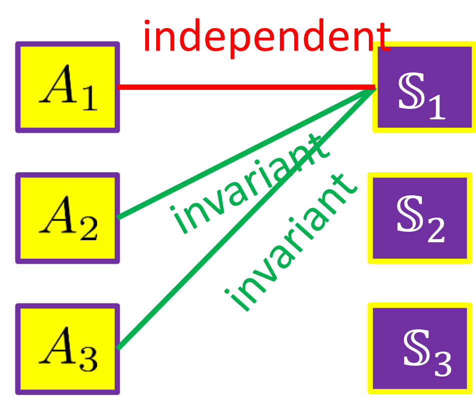
For $k = r\log_r\alpha$, construct a set $\{(A_i, \mathbb{S}_i)\}_{\mathrm{i=1}}^{\mathrm{k}}$ of

- subspaces $\mathbb{S}_1, \ldots, \mathbb{S}_k \subseteq \mathbb{F}_q^\alpha$ of dimension α/r spanned by vectors of the standard basis
- Invertible matrices $A_1, \ldots, A_k \in \mathbb{F}_q^{\alpha\times\alpha}$

which satisfy

1. The nonsingular property: every square block submatrix of

$$\begin{pmatrix} \mathrm{I} & A_1 & \cdots & A_1{}^{r-1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{I} & A_k & \cdots & A_k{}^{r-1} \end{pmatrix}$$ is invertible



2. The independence property:

$$\mathbb{S}_t + \mathbb{S}_t A_t + \mathbb{S}_t A_t^2 + \ldots + \mathbb{S}_t A_t{}^{r-1} = \mathbb{F}_q^\alpha$$

3. The invariance property: $\mathbb{S}_t = \mathbb{S}_t A_j, j \neq t$

for a field size q as small as possible.

Known results

Constructions of access-optimal MSR codes with the minimum sub-packetization factor $\boldsymbol{\alpha = r^{k/r}}$:

number r of parity nodes	finite field size [TWB 12]	finite field size [ASK15]
2	$k+1$	$k+1$
3	$k^2 3^{k/3-1}+1$	$\binom{k+3}{k} 3^{k/3+1}$
r	$k^{r-1} r^{k/r-1}+1$	$\binom{n}{k} r^{k/r+1}$

[TWB 12] I. Tamo, Z. Wang, and J. Bruck, "Long MDS codes for optimal repair bandwidth," ISIT12.
[ASK15] G. K. Agarwal, B. Sasidharan, and P. V. Kumar, "An alternative construction of an access-optimal regenerating codes with optimal sub-packetization level," NCC 2015.

Our results

Constructions of access-optimal MSR codes with the minimum sub-packetization factor $\boldsymbol{\alpha = r^{k/r}}$:

number r of parity nodes	finite field size [TWB 12]	finite field size [ASK15]	**Our finite field size q**
2	$k+1$	$k+1$	$\boldsymbol{k/2+1}$
3	$k^2 3^{k/3-1}+1$	$\binom{k+3}{k} 3^{k/3+1}$	$\boldsymbol{2k+1}$ **odd** $\boldsymbol{q}$ $\boldsymbol{k+1}$ **even** $\boldsymbol{q}$
r	$k^{r-1} r^{k/r-1}+1$	$\binom{n}{k} r^{k/r+1}$	

[TWB 12] I. Tamo, Z. Wang, and J. Bruck, “Long MDS codes for optimal repair bandwidth," ISIT12.
[ASK15] G. K. Agarwal, B. Sasidharan, and P. V. Kumar, “An alternative construction of an access-optimal regenerating codes with optimal sub-packetization level,” NCC 2015.

Our Construction

- Consider the matrix

$$A \triangleq \begin{pmatrix} 0 & 0 & \cdots & 1 \\ 1 & 0 & \cdots & 0 \\ & & & \\ 0 & \ddots & 0 & 0 \\ 0 & \cdots & 1 & 0 \end{pmatrix} \in \mathbb{F}_q^{r \times r}$$

$$\mathrm{minpoly} A = x^r - 1$$

- <u>Notice</u>:

$$e_0 \overset{A}{\mapsto} e_{r\text{-}1} \overset{A}{\mapsto} e_{r\text{-}2} \overset{A}{\mapsto} \cdots e_1$$

Our Construction

- Consider the matrix

$$A \triangleq \begin{pmatrix} 0 & 0 & \cdots & 1 \\ 1 & 0 & \cdots & 0 \\ 0 & \ddots & 0 & 0 \\ 0 & \cdots & 1 & 0 \end{pmatrix} \in \mathbb{F}_q^{r \times r}$$

$$\operatorname{minpoly} A = x^r - 1$$

- <u>Notice</u>:

$$e_0 \overset{A}{\mapsto} e_{r\text{-}1} \overset{A}{\mapsto} e_{r\text{-}2} \overset{A}{\mapsto} \cdots e_1 \Rightarrow \langle e_0 \rangle + \langle e_0 \rangle A + \cdots + \langle e_0 \rangle A^{r-1} = \mathbb{F}_q^{\alpha}$$

- The subspace $\langle e_0 \rangle$ is an ***independent*** subspace for *A*.

The independence property:

$$\mathbb{S}_t + \mathbb{S}_t A_t + \mathbb{S}_t A_t^2 + \ldots + \mathbb{S}_t {A_t}^{r-1} = \mathbb{F}_q^{\alpha}$$

The invariance property:

$$\mathbb{S}_t = \mathbb{S}_t A_j, j \neq t$$

Our Construction

- Consider the matrix

$$A \triangleq \begin{pmatrix} 0 & 0 & \cdots & 1 \\ 1 & 0 & \cdots & 0 \\ 0 & \ddots & 0 & 0 \\ 0 & \cdots & 1 & 0 \end{pmatrix} \in \mathbb{F}_q^{r \times r}$$

$$\mathrm{minpoly}\, A = x^r - 1$$

- <u>Notice</u>:

$$e_0 \overset{A}{\mapsto} e_{r\text{-}1} \overset{A}{\mapsto} e_{r\text{-}2} \overset{A}{\mapsto} \cdots e_1 \Rightarrow \langle e_0 \rangle + \langle e_0 \rangle A + \cdots + \langle e_0 \rangle A^{r-1} = \mathbb{F}_q^{\alpha}$$

- The subspace $\langle e_0 \rangle$ is an ***independent*** subspace for *A*.
- An eigenspace is an ***invariant*** subspace of A

The independence property:

$$\mathbb{S}_t + \mathbb{S}_t A_t + \mathbb{S}_t A_t^2 + \ldots + \mathbb{S}_t {A_t}^{r-1} = \mathbb{F}_q^{\alpha}$$

The invariance property:

$$\mathbb{S}_t = \mathbb{S}_t A_j, j \neq t$$

Our Construction

- Consider the matrix

$$A \triangleq \begin{pmatrix} 0 & 0 & \cdots & 1 \\ 1 & 0 & \cdots & 0 \\ 0 & \ddots & 0 & 0 \\ 0 & \cdots & 1 & 0 \end{pmatrix} \in \mathbb{F}_q^{r \times r}$$

$$\text{minpoly}\, A = x^r - 1$$

- <u>Notice</u>:

$$e_0 \overset{A}{\mapsto} e_{r\text{-}1} \overset{A}{\mapsto} e_{r\text{-}2} \quad \overset{A}{\mapsto} \cdots e_1 \Rightarrow \langle e_0 \rangle + \langle e_0 \rangle A + \cdots + \langle e_0 \rangle A^{r-1} = \mathbb{F}_q^{\alpha}$$

- The subspace $\langle e_0 \rangle$ is an ***independent*** subspace for *A*.
- An eigenspace is an ***invariant*** subspace of A
  - The eigenvalues of *A* are the roots of unity of order r

    $$\gamma_0 = 1, \gamma_1, \ldots, \gamma_{r-1}$$

  - The eigenvectors are

    $$\left\{ \left(1, \gamma_i, \gamma_i^2, \cdots, \gamma_i^{r-1}\right) \right\}_{i=0}^{r-1}.$$

The independence property:

$\mathbb{S}_t + \mathbb{S}_t A_t + \mathbb{S}_t A_t^2 + \ldots + \mathbb{S}_t A_t^{r-1} = \mathbb{F}_q^{\alpha}$

The invariance property:

$\mathbb{S}_t = \mathbb{S}_t A_j, j \neq t$

Our Construction

- Let $\mathrm{diag}_{\alpha/r}(A) = \begin{pmatrix} A & & \\ & \ddots & \\ & & A \end{pmatrix}$ $\quad A \triangleq \begin{pmatrix} 0 & 0 & \cdots & 1 \\ 1 & 0 & \cdots & 0 \\ 0 & \ddots & 0 & 0 \\ 0 & \cdots & 1 & 0 \end{pmatrix}$
- The subspace $\mathbb{S} = \langle e_0, e_r, e_{2r}, \dots, e_{\alpha-r} \rangle$ is an ***independent*** subspace

Our Construction

- Let $\mathrm{diag}_{\alpha/r}(A) = \begin{pmatrix} A & & \\ & \ddots & \\ & & A \end{pmatrix}$ $\quad A \triangleq \begin{pmatrix} 0 & 0 & \cdots & 1 \\ 1 & 0 & \cdots & 0 \\ 0 & \ddots & 0 & 0 \\ 0 & \cdots & 1 & 0 \end{pmatrix}$
- The subspace $\mathbb{S} = \langle e_0, e_r, e_{2r}, \ldots, e_{\alpha-r} \rangle$ is an ***independent*** subspace
- Let $M \triangleq P_M^{-1} \mathrm{diag}_{\alpha/r}(A) P_M = \begin{pmatrix} p_0 \\ \vdots \\ p_{\alpha\text{-}1} \end{pmatrix}^{-1} \begin{pmatrix} A & & \\ & \ddots & \\ & & A \end{pmatrix} \begin{pmatrix} p_0 \\ \vdots \\ p_{\alpha\text{-}1} \end{pmatrix}$
- The subspace $\langle \{p_0, p_r, p_{2r}, \ldots\} \rangle$ is an ***independent*** subspace for *M*.
- Subspaces of the form $\left\langle \{p_{ri} + \gamma_j p_{ri+1} + \gamma_j^2 p_{ri+2} + \ldots\}_{i=0}^{\alpha/r-1} \right\rangle$ are eigenspaces, and thus also ***invariant*** subspaces.

Problems:

- Choose the change-of-basis matrices P_M such that the ***invariance*** and the ***independence*** property (for all M's) are satisfied.
- Modify each M (multiply by a field constant λ)such that the ***nonsingular*** property is satisfied

Problems:

- Choose the change-of-basis matrices P_M such that the ***invariance*** and the ***independence*** property (for all M's) are satisfied.
- Modify each M (multiply by a field constant λ)such that the ***nonsingular*** property is satisfied

Solution:

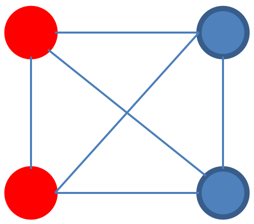
Based on perfect matchings in uniform hypergraphs

Perfect matchings in uniform hypergraphs

- r-uniform hypergraph - edges are sets of r vertices
  (if $r = 2$ then a standard graph)
- Matching – a set of mutually disjoint edges
- Perfect matching – a matching that covers all the vertices

Perfect matchings in uniform hypergraphs

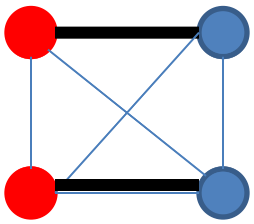
- r-uniform hypergraph - edges are sets of r vertices (if $r = 2$ then a standard graph)
- Matching – a set of mutually disjoint edges
- Perfect matching – a matching that covers all the vertices
- Let K_{α}^{r} be a complete r-uniform hypergraph whose α vertices are colored in r colors.



K_4^2

Perfect matchings in uniform hypergraphs

- r-uniform hypergraph - edges are sets of r vertices (if $r = 2$ then a standard graph)
- Matching – a set of mutually disjoint edges
- Perfect matching – a matching that covers all the vertices
- Let K_α^r be a complete r-uniform hypergraph whose α vertices are colored in r colors.
  Perfect colored matching in K_α^r – a perfect matching where no edge contains two vertices of the same color.

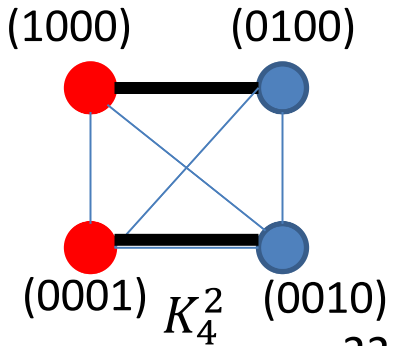


K_4^2

Perfect matchings in uniform hypergraphs

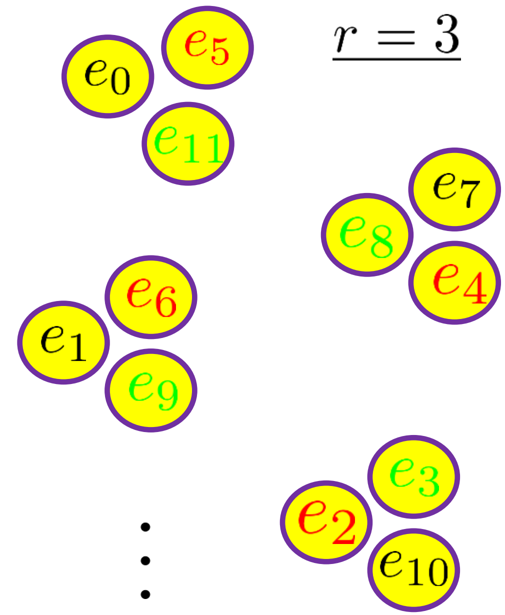
- r-uniform hypergraph - edges are sets of r vertices (if $r = 2$ then a standard graph)
- Matching – a set of mutually disjoint edges
- Perfect matching – a matching that covers all the vertices
- Let K_α^r be a complete r-uniform hypergraph whose α vertices are colored in r colors.
  Perfect colored matching in K_α^r – a perfect matching where no edge contains two vertices of the same color.

- We identify α unit vectors $e_0, \ldots, e_{\alpha-1}$ of length α with the α vertices of K_α^r.



Change Basis Matrices from Matchings

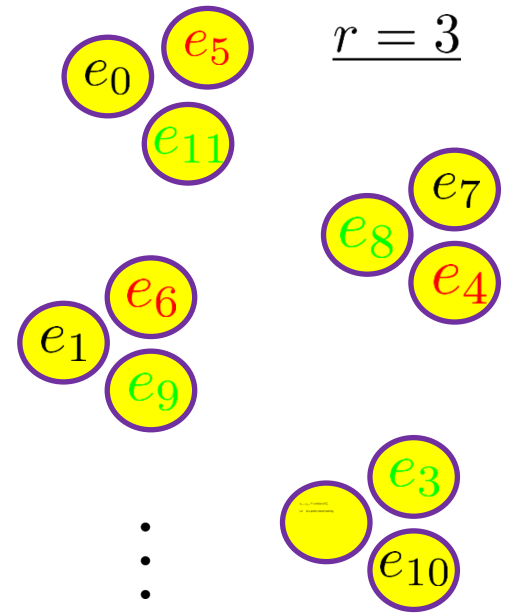
$e_0, \ldots, e_{\alpha-1} \Leftrightarrow \alpha$ vertices of K_α^r



Change Basis Matrices from Matchings

$e_0, \ldots, e_{\alpha-1} \Leftrightarrow \alpha$ vertices of K_α^r

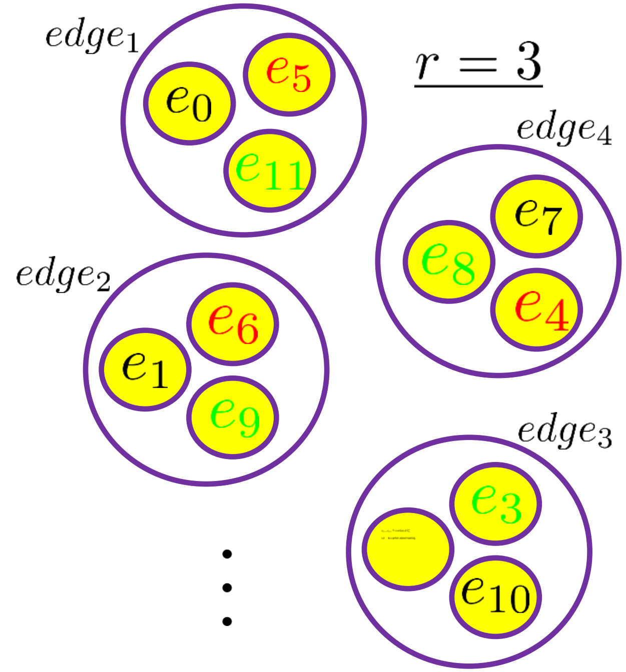
Let $\mathcal{M}$ be a perfect colored matching.



Change Basis Matrices from Matchings

$e_0, \ldots, e_{\alpha-1} \Leftrightarrow \alpha$ vertices of K^r_α

Let $\mathcal{M}$ be a perfect colored matching.



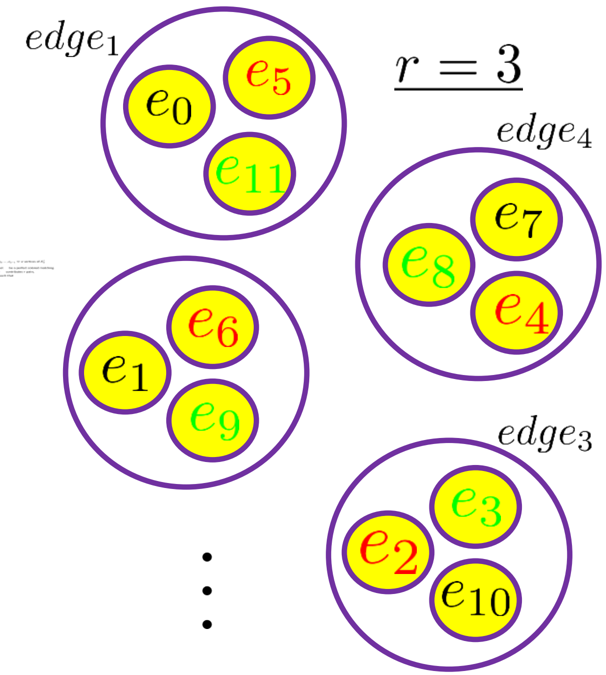
Change Basis Matrices from Matchings

$e_0, \ldots, e_{\alpha-1} \Leftrightarrow \alpha$ vertices of K_α^r

Let $\mathcal{M}$ be a perfect colored matching.

- $\mathcal{M}$ contributes r pairs, $\{(S_{M_i}, A_{M_i})\}_{i=1}^{r}$

such that



Change Basis Matrices from Matchings

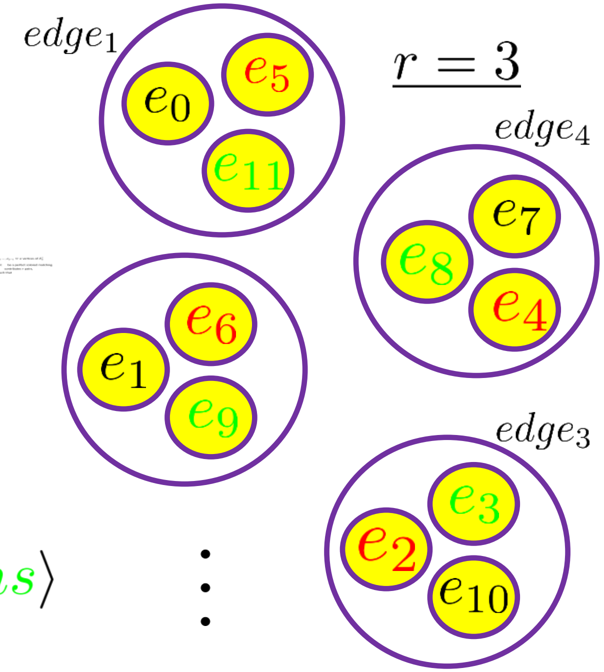
$e_0, \ldots, e_{\alpha-1} \Leftrightarrow \alpha$ vertices of K_α^r

Let $\mathcal{M}$ be a perfect colored matching.

- $\mathcal{M}$ contributes r pairs, $\{(S_{M_i}, A_{M_i})\}_{i=1}^r$

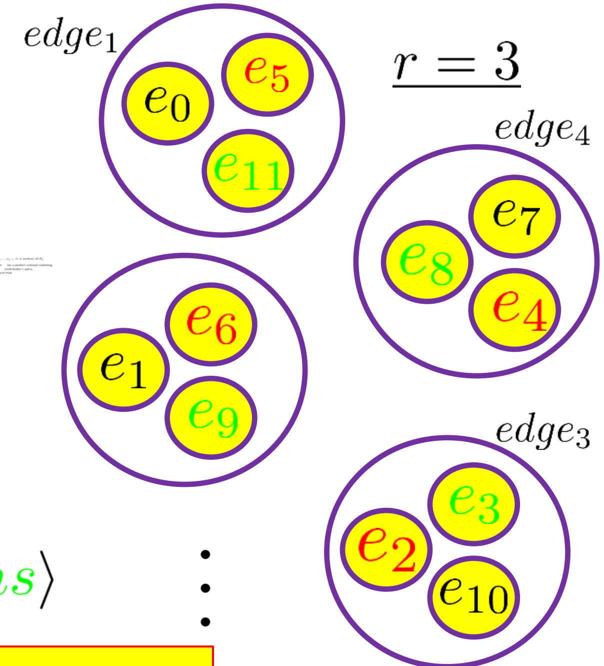
such that

$S_{M_1} = \langle reds \rangle \quad S_{M_2} = \langle blacks \rangle \quad S_{M_3} = \langle greens \rangle$



Change Basis Matrices from Matchings

$e_0, \ldots, e_{\alpha-1} \Leftrightarrow \alpha$ vertices of K_α^r



Let $\mathcal{M}$ be a perfect colored matching.

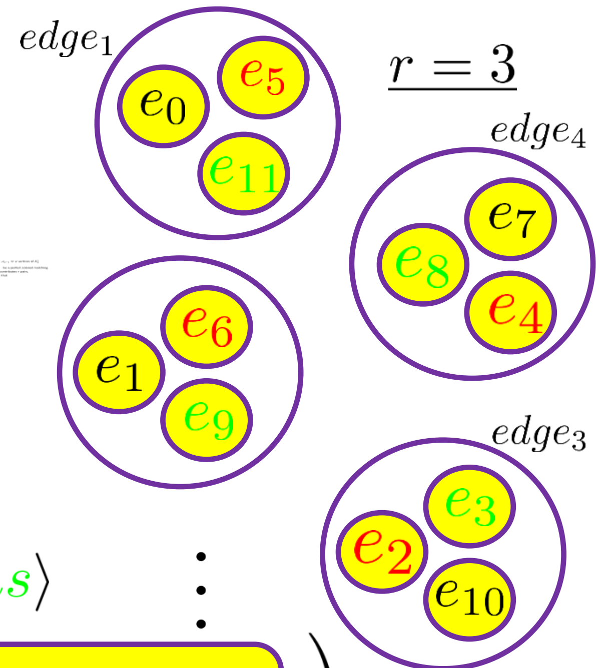
- $\mathcal{M}$ contributes r pairs, $\{(S_{M_i}, A_{M_i})\}_{i=1}^{r}$ such that

$S_{M_1} = \langle reds \rangle \quad S_{M_2} = \langle blacks \rangle \quad S_{M_3} = \langle greens \rangle \quad \vdots$

All repair subspaces are spanned by *unit* vectors then **access-optimal** code

Change Basis Matrices from Matchings

$e_0, \dots, e_{\alpha-1} \Leftrightarrow \alpha$ vertices of K_α^r



Let $\mathcal{M}$ be a perfect colored matching.

- $\mathcal{M}$ contributes r pairs, $\{(S_{M_i}, A_{M_i})\}_{i=1}^r$ such that

$$S_{M_1} = \langle reds \rangle \quad S_{M_2} = \langle blacks \rangle \quad S_{M_3} = \langle greens \rangle \quad \vdots$$

$$A_{M_i} \triangleq P_{M_i}^{-1} \mathrm{diag}_{\alpha/r}(A) P_{M_i}, \qquad P_{M_i} \triangleq \begin{pmatrix} \boxed{\text{a function of } edge_1} \\ \vdots \\ \boxed{\text{a function of } edge_{\alpha/r}} \end{pmatrix}$$

Change Basis Matrices from Matchings

Ensure that
S_{M_i} is an independent subspace
S_{M_j} is an eigenspace, for $i \neq j$

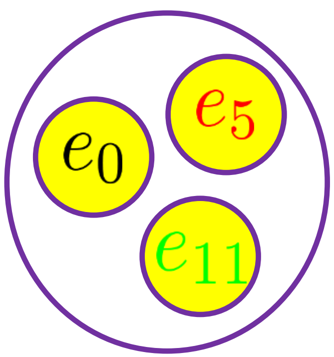
$$P_{M_i} \triangleq \begin{pmatrix} \boxed{\text{a function of } edge_1} \\ \vdots \\ \boxed{\text{a function of } edge_{\alpha/r}} \end{pmatrix}$$

Change Basis Matrices from Matchings

Ensure that
S_{M_i} is an independent subspace
S_{M_j} is an eigenspace, for $i \neq j$

$$P_{M_i} \triangleq \begin{pmatrix} \text{a function of } edge_1 \\ \vdots \\ \text{a function of } edge_{\alpha/r} \end{pmatrix}$$

e.g.,



$edge_1$

$$\text{a function of } edge_1 = \begin{pmatrix} 1 & 0 & 0 \\ 1 & -\frac{\gamma_1}{\gamma_1 - 1} & \frac{1}{\gamma_1 - 1} \\ 1 & \frac{1}{\gamma_1 - 1} & -\frac{\gamma_1}{\gamma_1 - 1} \end{pmatrix} \cdot \begin{pmatrix} e_0 \\ e_5 \\ e_{11} \end{pmatrix}$$

For A_{M_2} : S_{M_2} Independent subspace;

S_{M_1} Eigenspace for γ_1

S_{M_3} Eigenspace for γ_2

From One Matching to Many Matchings

- One matching contributes $\boldsymbol{r}$ pairs
- More than one matching?

From One Matching to Many Matchings

- One matching contributes $\boldsymbol{r}$ pairs
- More than one matching?
- Construct codes from *different* matchings which satisfy a mutual relation:
  - **Each edge in one matching is *monochromatic* in any other matching.**

From One Matching to Many Matchings

- One matching contributes $\boldsymbol{r}$ pairs
- More than one matching?
- Construct codes from *different* matchings which satisfy a mutual relation:
  - **Each edge in one matching is *monochromatic* in any other matching.**
- Can construct $\boldsymbol{k/r}$ such mathchings

$\boldsymbol{k}$ pairs: $\{(S^1_{M_i}, A^1_{M_i})\}_{i=1}^r \cup \ldots \cup \{(S^{k/r}_{M_i}, A^{k/r}_{M_i})\}_{i=1}^r$

The Nonsingular Property

- So far, the construction works for *any* number r of parities.
- What about the ***nonsingular*** property?
  - $\forall$ square block submatrix of $\begin{pmatrix} \mathrm{I} & A_1 & \cdots & A_1^{r-1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{I} & A_k & \cdots & A_k^{r-1} \end{pmatrix}$ is nonsingular

The Nonsingular Property

- So far, the construction works for *any* number r of parities.
- What about the ***nonsingular*** property?
  - $\forall$ square block submatrix of $\begin{pmatrix} I & A_1 & \cdots & A_1^{r-1} \\ \vdots & \vdots & \ddots & \vdots \\ I & A_k & \cdots & A_k^{r-1} \end{pmatrix}$ is nonsingular
- A_i is nonsingular

- For $\underline{\boldsymbol{r = 2}}$ parities:
  - $\mathrm{rank}(A_i - A_j) = \alpha$

The Nonsingular Property

- So far, the construction works for *any* number r of parities.
- What about the ***nonsingular*** property?

– $\forall$ square block submatrix of $\begin{pmatrix} I\, A_1 \cdots {A_1}^{r-1} \\ \vdots \;\; \vdots \;\; \ddots \;\; \vdots \\ I\, A_k \cdots {A_k}^{r-1} \end{pmatrix}$ is nonsingular

- A_i is nonsingular

- For $\underline{\boldsymbol{r=2}}$ parities:
  - $\text{rank}(A_i - A_j) = \alpha$

- For $\underline{\boldsymbol{r=3}}$ parities:
  - $\text{rank}(A_i - A_j) = \alpha$
  - $\text{rank}(A_i^2 - A_j^2) = \alpha$
  - Non singularity of $\begin{pmatrix} I & A_i & A_i^2 \\ I & A_j & A_j^2 \\ I & A_k & A_k^2 \end{pmatrix}$.

The Nonsingular Property

- <u>Our solution:</u>

  $A_i \Rightarrow \lambda_i A_i \quad, 0 \neq \lambda_i \in \mathbb{F}_q, \lambda_i = \lambda_j$ for the same matching

The Nonsingular Property

- Our solution:

  $A_i \Rightarrow \lambda_i A_i \quad ,0 \neq \lambda_i \in \mathbb{F}_q, \lambda_i = \lambda_j$ for the same matching

- For $\boldsymbol{r = 2}$ parities:

  $$\boldsymbol{q = k/2 + 1}$$

- For $\boldsymbol{r = 3}$ parities:

  $$\boldsymbol{q \geq \begin{cases} 2k + 1 \text{ for odd } q \\ k + 1 \text{ for even } q \end{cases}}$$

Our results

Constructions of access-optimal MSR codes with the minimum sub-packetization factor $\boldsymbol{\alpha = r^{k/r}}$:

number r of parity nodes	finite field size [TWB 12]	finite field size [ASK15]	**Our finite field size q**
2	$k+1$	$k+1$	$\boldsymbol{k/2+1}$

[TWB 12] I. Tamo, Z. Wang, and J. Bruck, "Long MDS codes for optimal repair bandwidth," ISIT12.
[ASK15] G. K. Agarwal, B. Sasidharan, and P. V. Kumar, "An alternative construction of an access-optimal regenerating codes with optimal sub-packetization level," NCC 2015.

Our results

Constructions of access-optimal MSR codes with the minimum sub-packetization factor $\boldsymbol{\alpha = r^{k/r}}$:

number r of parity nodes	finite field size [TWB 12]	finite field size [ASK15]	**Our finite field size q**
2	$k+1$	$k+1$	$\boldsymbol{k/2+1}$
3	$k^2 3^{k/3-1}+1$	$\binom{k+3}{k} 3^{k/3+1}$	**$2k+1$ odd q** **$k+1$ even q**

[TWB 12] I. Tamo, Z. Wang, and J. Bruck, "Long MDS codes for optimal repair bandwidth," ISIT12.
[ASK15] G. K. Agarwal, B. Sasidharan, and P. V. Kumar, "An alternative construction of an access-optimal regenerating codes with optimal sub-packetization level," NCC 2015.

Our results

Constructions of access-optimal MSR codes with the minimum sub-packetization factor $\boldsymbol{\alpha = r^{k/r}}$:

number r of parity nodes	finite field size [TWB 12]	finite field size [ASK15]	**Our finite field size q**
2	$k+1$	$k+1$	$\boldsymbol{k/2+1}$
3	$k^2 3^{k/3-1}+1$	$\binom{k+3}{k} 3^{k/3+1}$	**$2k+1$ odd q** **$k+1$ even q**
r	$k^{r-1} r^{k/r-1} + 1$	$\binom{n}{k} r^{k/r+1}$	**?**

[TWB 12] I. Tamo, Z. Wang, and J. Bruck, “Long MDS codes for optimal repair bandwidth," ISIT12.
[ASK15] G. K. Agarwal, B. Sasidharan, and P. V. Kumar, “An alternative construction of an access-optimal regenerating codes with optimal sub-packetization level,” NCC 2015.

Future Research

- Nonsingularity for more than three parities.
- Reduce the field size.

Future Research

- Nonsingularity for more than three parities.
- Reduce the field size.

- <u>In general</u>:

  Non-systematic node failure.

  More than one simultaneous failures?

Thank you!