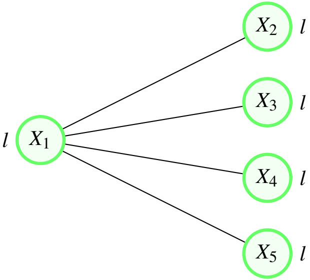
Node Repair on Connected Graphs

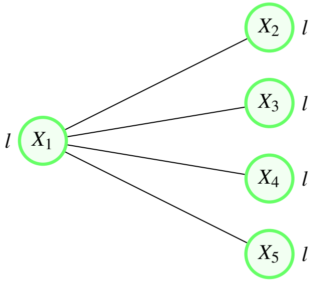
Adway Patra & Alexander Barg
(University of Maryland, College Park)

JMM 2022

Node repair in distributed storage

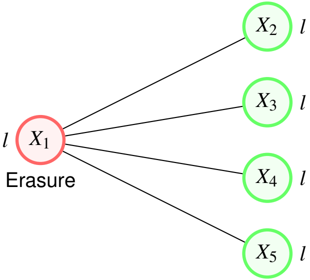


Node repair in distributed storage



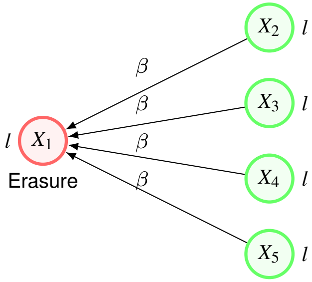
- Each codeword symbol stored in a node.

Node repair in distributed storage



- Each codeword symbol stored in a node.
- Correct erasures while trying to minimize total data "moved".

Node repair in distributed storage

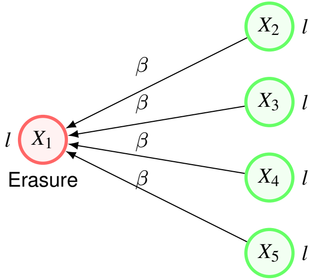


- Each codeword symbol stored in a node.
- Correct erasures while trying to minimize total data "moved".
- Total required transmission bounded by the Cut-set bound[1]

$$B \leqslant \sum_{i=0}^{k-1} \min\{l, (d-i)\beta\}$$

[1] Dimakis *et. al*, 2010

Node repair in distributed storage



- Each codeword symbol stored in a node.
- Correct erasures while trying to minimize total data "moved".
- Total required transmission bounded by the Cut-set bound[1]

$$B \leqslant \sum_{i=0}^{k-1} \min\{l, (d-i)\beta\}$$

- Different pairs of (l, β) satisfying the above with equality give rise to different points on the storage-bandwidth trade-off.

[1] Dimakis *et. al*, 2010

Moving away from traditional setting

Moving away from traditional setting

- General problem assumes d helper nodes are chosen from the direct neighbors of the failed node, i.e., high connectivity.

Moving away from traditional setting

- General problem assumes d helper nodes are chosen from the direct neighbors of the failed node, i.e., high connectivity.
- What can be done for sparsely connected graphs?

Moving away from traditional setting

- General problem assumes d helper nodes are chosen from the direct neighbors of the failed node, i.e., high connectivity.
- What can be done for sparsely connected graphs?
  - Simple answer: Relaying of repair data.

Moving away from traditional setting

- General problem assumes d helper nodes are chosen from the direct neighbors of the failed node, i.e., high connectivity.
- What can be done for sparsely connected graphs?
  - Simple answer: Relaying of repair data.
  - Choose a suitable spanning tree, having the failed node at the root.

Moving away from traditional setting

- General problem assumes d helper nodes are chosen from the direct neighbors of the failed node, i.e., high connectivity.
- What can be done for sparsely connected graphs?
  - Simple answer: Relaying of repair data.
  - Choose a suitable spanning tree, having the failed node at the root.
  - Each node forwards the helper data of its descendants, possibly supplementing with its own.

Moving away from traditional setting

- General problem assumes d helper nodes are chosen from the direct neighbors of the failed node, i.e., high connectivity.
- What can be done for sparsely connected graphs?
  - Simple answer: Relaying of repair data.
  - Choose a suitable spanning tree, having the failed node at the root.
  - Each node forwards the helper data of its descendants, possibly supplementing with its own.
- Same data gets transmitted multiple times.

Moving away from traditional setting

- General problem assumes d helper nodes are chosen from the direct neighbors of the failed node, i.e., high connectivity.
- What can be done for sparsely connected graphs?
  - Simple answer: Relaying of repair data.
  - Choose a suitable spanning tree, having the failed node at the root.
  - Each node forwards the helper data of its descendants, possibly supplementing with its own.
- Same data gets transmitted multiple times.
- Total required communication depends on the structure of the tree.

Moving away from traditional setting

- General problem assumes d helper nodes are chosen from the direct neighbors of the failed node, i.e., high connectivity.
- What can be done for sparsely connected graphs?
  - Simple answer: Relaying of repair data.
  - Choose a suitable spanning tree, having the failed node at the root.
  - Each node forwards the helper data of its descendants, possibly supplementing with its own.
- Same data gets transmitted multiple times.
- Total required communication depends on the structure of the tree.
  - For example, if the helpers are on a line, the failed node being at the end, then $\frac{d(d+1)\beta}{2} = \Theta(d^2)$ transmission required.

Moving away from traditional setting

- General problem assumes d helper nodes are chosen from the direct neighbors of the failed node, i.e., high connectivity.
- What can be done for sparsely connected graphs?
  - Simple answer: Relaying of repair data.
  - Choose a suitable spanning tree, having the failed node at the root.
  - Each node forwards the helper data of its descendants, possibly supplementing with its own.
- Same data gets transmitted multiple times.
- Total required communication depends on the structure of the tree.
  - For example, if the helpers are on a line, the failed node being at the end, then $\frac{d(d+1)\beta}{2} = \Theta(d^2)$ transmission required.

Question 1 : Is it possible to process the data to reduce communication?

Moving away from traditional setting

- General problem assumes d helper nodes are chosen from the direct neighbors of the failed node, i.e., high connectivity.
- What can be done for sparsely connected graphs?
  - Simple answer: Relaying of repair data.
  - Choose a suitable spanning tree, having the failed node at the root.
  - Each node forwards the helper data of its descendants, possibly supplementing with its own.
- Same data gets transmitted multiple times.
- Total required communication depends on the structure of the tree.
  - For example, if the helpers are on a line, the failed node being at the end, then $\frac{d(d+1)\beta}{2} = \Theta(d^2)$ transmission required.

Question 1 : Is it possible to process the data to reduce communication?

Question 2 : If so, then to what extent?

The bounds: How much can we process?

The bounds: How much can we process?

- Node v_i holds random variable X_i.

The bounds: How much can we process?

- Node v_i holds random variable X_i.
- For failed node v_f, d helper nodes $v_1, \cdots, v_d$. Helper node v_i would have sent S_i^f to v_f in case of direct connectivity.

The bounds: How much can we process?

- Node v_i holds random variable X_i.
- For failed node v_f, d helper nodes $v_1, \cdots, v_d$. Helper node v_i would have sent S_i^f to v_f in case of direct connectivity.
- Operating at the MSR point: $H(X_f) = l$, $H(S_i^f) = \beta = \frac{l}{d-k+1}$.

The bounds: How much can we process?

- Node v_i holds random variable X_i.
- For failed node v_f, d helper nodes $v_1, \cdots, v_d$. Helper node v_i would have sent S_i^f to v_f in case of direct connectivity.
- Operating at the MSR point: $H(X_f) = l, H(S_i^f) = \beta = \frac{l}{d-k+1}$.
- $H(S_i^f|X_i) = 0, H(X_f|S_1^f, \cdots, S_d^f) = 0$.

The bounds: How much can we process?

- Node v_i holds random variable X_i.
- For failed node v_f, d helper nodes $v_1, \cdots, v_d$. Helper node v_i would have sent S_i^f to v_f in case of direct connectivity.
- Operating at the MSR point: $H(X_f) = l$, $H(S_i^f) = \beta = \frac{l}{d-k+1}$.
- $H(S_i^f|X_i) = 0, H(X_f|S_1^f, \cdots, S_d^f) = 0.$

Lemma

Let $v_f, f \in [n]$ be the failed node. For a subset of the helper nodes $E \subset D$ let R_E^f be a function of S_E^f such that

$$H(X_f|R_E^f, S_{D\setminus E}^f) = 0.$$

1) If $|E| \geqslant d-k+1$, then

$$H(R_E^f) \geqslant l.$$

2) If $|E| \leqslant d-k$, then

$$H(R_E^f) \geqslant \frac{|E|l}{d-k+1}.$$

Proof of Lemma

- Given $X_{D\setminus E}$ the information contained in R_E^f is sufficient to repair v_f, i.e.,

$$H(W_f | R_E^f, W_{D\setminus E}) = 0.$$

Proof of Lemma

- Given $X_{D\setminus E}$ the information contained in R_E^f is sufficient to repair v_f, i.e.,

$$H(W_f|R_E^f, W_{D\setminus E}) = 0.$$

- Take a set $A \subset E$ with $|A| = k - 1 - |D\setminus E|$. Now,

$$H(R_E^f, W_{D\setminus E}, W_A) = H(R_E^f, W_{D\setminus E}, W_f, W_A) \geqslant kl$$

by the MDS property.

Proof of Lemma

- Given $X_{D\setminus E}$ the information contained in R_E^f is sufficient to repair v_f, i.e.,

$$H(W_f|R_E^f, W_{D\setminus E}) = 0.$$

- Take a set $A \subset E$ with $|A| = k - 1 - |D\setminus E|$. Now,

$$H(R_E^f, W_{D\setminus E}, W_A) = H(R_E^f, W_{D\setminus E}, W_f, W_A) \geqslant kl$$

  by the MDS property.

-

$$\begin{aligned} H(R_E^f, W_{D\setminus E}, W_A) &\leqslant H(R_E^f) + H(W_{D\setminus E}, W_A) \\ &= H(R_E^f) + (k-1)l \end{aligned}$$

Proof of Lemma

- Given $X_{D\backslash E}$ the information contained in R_E^f is sufficient to repair v_f, i.e.,

$$H(W_f|R_E^f, W_{D\backslash E}) = 0.$$

- Take a set $A \subset E$ with $|A| = k - 1 - |D\backslash E|$. Now,

$$H(R_E^f, W_{D\backslash E}, W_A) = H(R_E^f, W_{D\backslash E}, W_f, W_A) \geqslant kl$$

  by the MDS property.

-

$$\begin{aligned} H(R_E^f, W_{D\backslash E}, W_A) &\leqslant H(R_E^f) + H(W_{D\backslash E}, W_A) \\ &= H(R_E^f) + (k-1)l \end{aligned}$$

- The proof of Part (2) is similar.

Lower bound on communication

Lower bound on communication

Definition

Repair graph: The subgraph spanned by the failed node and d helper nodes closest to it in terms of graph distance.

$\Gamma_i(v_f)$: set of helper nodes at distance i from the failed node.

Lower bound on communication

Definition

Repair graph: The subgraph spanned by the failed node and d helper nodes closest to it in terms of graph distance.

$\Gamma_i(v_f)$: set of helper nodes at distance i from the failed node.

Proposition

Let R_j^f be the random variable denoting the information flow from the j-th layer to the $(j-1)$-th layer. Then

$$H(R_j^f) \geqslant \min\left\{l, \frac{|\cup_{i=j}^{t}\Gamma_i(v_f)| \cdot l}{d-k+1}\right\}$$

Lower bound on communication

Definition

Repair graph: The subgraph spanned by the failed node and d helper nodes closest to it in terms of graph distance.

$\Gamma_i(v_f)$: set of helper nodes at distance i from the failed node.

Proposition

Let R_j^f be the random variable denoting the information flow from the j-th layer to the $(j-1)$-th layer. Then

$$H(R_j^f) \geqslant \min\left\{l, \frac{|\cup_{i=j}^t \Gamma_i(v_f)| \cdot l}{d-k+1}\right\}$$

Proof.

Take $E = \cup_{i=j}^t \Gamma_i(v_f)$. □

Lower bound on communication

If the repair graph is a tree, then

Lower bound on communication

If the repair graph is a tree, then

Proposition

Let $J_f = \{v \in V(T_f) \backslash \{v_f\} : |D^*(v)| \geqslant d-k+2\}$*. The total communication complexity* β_{total} *for the repair of node* v_f *on the repair tree* T_f *is bounded as*

$$\beta_{total} \geqslant \sum_{v \in J_f} l + \sum_{v \in V(T_f) \backslash (\{v_f\} \cup J_f)} \frac{|D^*(v)| l}{d-k+1}.$$

where $D^*(v)$: *set of descendants of* v *including itself.*

Lower bound on communication

If the repair graph is a tree, then

Proposition

Let $J_f = \{v \in V(T_f)\backslash\{v_f\} : |D^*(v)| \geqslant d-k+2\}$. *The total communication complexity* β_{total} *for the repair of node* v_f *on the repair tree* T_f *is bounded as*

$$\beta_{total} \geqslant \sum_{v \in J_f} l + \sum_{v \in V(T_f)\backslash(\{v_f\}\cup J_f)} \frac{|D^*(v)|l}{d-k+1}.$$

where $D^*(v)$: *set of descendants of* v *including itself.*

Proof.

□

Lower bound on communication

If the repair graph is a tree, then

Proposition

Let $J_f = \{v \in V(T_f) \backslash \{v_f\} : |D^*(v)| \geqslant d-k+2\}$. *The total communication complexity* β_{total} *for the repair of node* v_f *on the repair tree* T_f *is bounded as*

$$\beta_{total} \geqslant \sum_{v \in J_f} l + \sum_{v \in V(T_f) \backslash (\{v_f\} \cup J_f)} \frac{|D^*(v)| l}{d-k+1}.$$

where $D^*(v)$: *set of descendants of* v *including itself.*

Proof.

- For every non-root node $v \notin J_f$, we have $|D^*(v)| \leqslant d-k$.

□

Lower bound on communication

If the repair graph is a tree, then

Proposition

Let $J_f = \{v \in V(T_f) \backslash \{v_f\} : |D^*(v)| \geqslant d-k+2\}$. The total communication complexity β_{total} for the repair of node v_f on the repair tree T_f is bounded as

$$\beta_{total} \geqslant \sum_{v \in J_f} l + \sum_{v \in V(T_f) \backslash (\{v_f\} \cup J_f)} \frac{|D^*(v)| l}{d-k+1}.$$

where $D^*(v)$: set of descendants of v including itself.

Proof.

- For every non-root node $v \notin J_f$, we have $|D^*(v)| \leqslant d-k$.
- Any outflow of information out of the subtree spanned by $D^*(v)$ passes through the node v.

□

Lower bound on communication

If the repair graph is a tree, then

Proposition

Let $J_f = \{v \in V(T_f) \backslash \{v_f\} : |D^*(v)| \geqslant d-k+2\}$. The total communication complexity β_{total} for the repair of node v_f on the repair tree T_f is bounded as

$$\beta_{total} \geqslant \sum_{v \in J_f} l + \sum_{v \in V(T_f) \backslash (\{v_f\} \cup J_f)} \frac{|D^*(v)| l}{d-k+1}.$$

where $D^*(v)$: set of descendants of v including itself.

Proof.

- For every non-root node $v \notin J_f$, we have $|D^*(v)| \leqslant d-k$.
- Any outflow of information out of the subtree spanned by $D^*(v)$ passes through the node v.
- Needs to transmit at least $|D^*(v)| \cdot l/(d-k+1)$ symbols to its immediate parent in T_f.

□

Lower bound on communication

If the repair graph is a tree, then

Proposition

Let $J_f = \{v \in V(T_f)\backslash\{v_f\} : |D^*(v)| \geqslant d-k+2\}$. The total communication complexity β_{total} for the repair of node v_f on the repair tree T_f is bounded as

$$\beta_{total} \geqslant \sum_{v \in J_f} l + \sum_{v \in V(T_f)\backslash(\{v_f\}\cup J_f)} \frac{|D^*(v)|l}{d-k+1}.$$

where $D^*(v)$: set of descendants of v including itself.

Proof.

- For every non-root node $v \notin J_f$, we have $|D^*(v)| \leqslant d-k$.
- Any outflow of information out of the subtree spanned by $D^*(v)$ passes through the node v.
- Needs to transmit at least $|D^*(v)| \cdot l/(d-k+1)$ symbols to its immediate parent in T_f.
- Every node $v \in J_f$ needs to transmit at least l symbols to its immediate parent.

□

Achieving the bounds: The Product Matrix Framework[2]

[2] Rashmi *et al.*, 2011

Achieving the bounds: The Product Matrix Framework[2]

- Paremeters: $[n, k, d = 2(k-1), l = k-1, \beta = 1, M = k(k-1)]$

[2] Rashmi *et al.*, 2011

Achieving the bounds: The Product Matrix Framework[2]

- Paremeters: $[n, k, d = 2(k-1), l = k-1, \beta = 1, M = k(k-1)]$
- Take two symmetric polynomials $s_1(y,z), s_2(y,z)$ over field F_q of degree at most $k-2$. The coefficients carry the message symbols.

[2]Rashmi *et al.*, 2011

Achieving the bounds: The Product Matrix Framework[2]

- Paremeters: $[n, k, d = 2(k-1), l = k-1, \beta = 1, M = k(k-1)]$
- Take two symmetric polynomials $s_1(y,z), s_2(y,z)$ over field F_q of degree at most $k-2$. The coefficients carry the message symbols.
- For distinct points $a_1, \cdots, a_n$, node i stores $g^{(i)}(z) = s_1(a_i, z) + a_i^{k-1} s_2(a_i, z)$.

[2]Rashmi *et al.*, 2011

Achieving the bounds: The Product Matrix Framework[2]

- Paremeters: $[n, k, d = 2(k-1), l = k-1, \beta = 1, M = k(k-1)]$
- Take two symmetric polynomials $s_1(y,z), s_2(y,z)$ over field F_q of degree at most $k-2$. The coefficients carry the message symbols.
- For distinct points $a_1, \cdots, a_n$, node i stores $g^{(i)}(z) = s_1(a_i, z) + a_i^{k-1} s_2(a_i, z)$.

Recall: What we want to prove

A set of size at least $d-k+1$ needs to transmit exactly l symbols for repair.

[2] Rashmi *et al.*, 2011

Achieving the bounds: The Product Matrix Framework[2]

- Paremeters: $[n, k, d = 2(k-1), l = k-1, \beta = 1, M = k(k-1)]$
- Take two symmetric polynomials $s_1(y,z), s_2(y,z)$ over field F_q of degree at most $k-2$. The coefficients carry the message symbols.
- For distinct points $a_1, \cdots, a_n$, node i stores $g^{(i)}(z) = s_1(a_i, z) + a_i^{k-1} s_2(a_i, z)$.

Recall: What we want to prove

A set of size at least $d-k+1$ needs to transmit exactly l symbols for repair.

- Failed node f, helper set D, $A \subset D$ with $|A| = d-k+1 = k-1$.

[2] Rashmi *et al.*, 2011

Achieving the bounds: The Product Matrix Framework[2]

- Paremeters: $[n, k, d = 2(k-1), l = k-1, \beta = 1, M = k(k-1)]$
- Take two symmetric polynomials $s_1(y,z), s_2(y,z)$ over field F_q of degree at most $k-2$. The coefficients carry the message symbols.
- For distinct points $a_1, \cdots, a_n$, node i stores $g^{(i)}(z) = s_1(a_i, z) + a_i^{k-1} s_2(a_i, z)$.

Recall: What we want to prove

A set of size at least $d-k+1$ needs to transmit exactly l symbols for repair.

- Failed node f, helper set D, $A \subset D$ with $|A| = d-k+1 = k-1$.
- For $h \in D$ define $l^{(h)}(z) = \sum_{j=0}^{d-1} l_j^h z^j := \prod_{\substack{i \in D \\ i \neq h}} \frac{z - a_i}{a_h - a_i}$

[2]Rashmi *et al.*, 2011

Achieving the bounds: The Product Matrix Framework[2]

- Paremeters: $[n, k, d = 2(k-1), l = k-1, \beta = 1, M = k(k-1)]$
- Take two symmetric polynomials $s_1(y,z), s_2(y,z)$ over field F_q of degree at most $k-2$. The coefficients carry the message symbols.
- For distinct points $a_1, \cdots, a_n$, node i stores $g^{(i)}(z) = s_1(a_i, z) + a_i^{k-1} s_2(a_i, z)$.

Recall: What we want to prove

A set of size at least $d-k+1$ needs to transmit exactly l symbols for repair.

- Failed node f, helper set D, $A \subset D$ with $|A| = d-k+1 = k-1$.
- For $h \in D$ define $l^{(h)}(z) = \sum_{j=0}^{d-1} l_j^h z^j := \prod_{\substack{i \in D \\ i \neq h}} \frac{z-a_i}{a_h - a_i}$
- The set A transmits the l-vector

$$\xi(f, A) = \sum_{h \in A} g^{(h)}(a_f) \begin{bmatrix} l_0^h + a_f^{k-1} l_{k-1}^h \\ l_1^h + a_f^{k-1} l_k^h \\ \vdots \\ l_{k-2}^h + a_f^{k-1} l_{2k-3}^h \end{bmatrix}.$$

[2]Rashmi *et al.*, 2011

Achieving the bounds: Proving correctness

Achieving the bounds: Proving correctness

Claim

The content of the failed node f coincides with the vector $\xi(f,D)$, i.e., $g^{(f)}(z) = \sum_{i=0}^{l-1} (\xi(f,D))_i z^i$.

Achieving the bounds: Proving correctness

Claim

The content of the failed node f coincides with the vector $\xi(f,D)$, i.e., $g^{(f)}(z) = \sum_{i=0}^{l-1}(\xi(f,D))_i z^i$.

- Consider the polynomial $H(z) = s_1(a_f, z) + z^{k-1} s_2(a_f, z) = \sum_{j=0}^{d-1} h_j z^j$, $\deg(H) \leqslant 2k-3 = d-1$.

Achieving the bounds: Proving correctness

Claim

The content of the failed node f coincides with the vector $\xi(f, D)$, i.e., $g^{(f)}(z) = \sum_{i=0}^{l-1} (\xi(f, D))_i z^i$.

- Consider the polynomial $H(z) = s_1(a_f, z) + z^{k-1} s_2(a_f, z) = \sum_{j=0}^{d-1} h_j z^j$, $\deg(H) \leqslant 2k - 3 = d - 1$.
- The polynomial corresponding to the failed node defined before can be written as

$$g^{(f)}(z) = \sum_{j=0}^{k-2} (h_j + a_f^{k-1} h_{k-1+j}) z^j.$$

Achieving the bounds: Proving correctness

Claim

The content of the failed node f coincides with the vector $\xi(f,D)$, i.e., $g^{(f)}(z) = \sum_{i=0}^{l-1}(\xi(f,D))_i z^i$.

- Consider the polynomial $H(z) = s_1(a_f,z) + z^{k-1}s_2(a_f,z) = \sum_{j=0}^{d-1} h_j z^j$, $\deg(H) \leqslant 2k-3 = d-1$.
- The polynomial corresponding to the failed node defined before can be written as

$$g^{(f)}(z) = \sum_{j=0}^{k-2}(h_j + a_f^{k-1}h_{k-1+j})z^j.$$

- Rephrasing, the contents of the node f is

$$(h_0 + a_f^{k-1}h_{k-1}, h_1 + a_f^{k-1}h_k, \ldots, h_{k-2} + a_f^{k-1}h_{2k-3})^{\mathsf{T}}.$$

Achieving the bounds: Proving correctness

Claim

The content of the failed node f coincides with the vector $\xi(f, D)$, i.e., $g^{(f)}(z) = \sum_{i=0}^{l-1} (\xi(f, D))_i z^i$.

- Consider the polynomial $H(z) = s_1(a_f, z) + z^{k-1} s_2(a_f, z) = \sum_{j=0}^{d-1} h_j z^j$, $\deg(H) \leqslant 2k - 3 = d - 1$.
- The polynomial corresponding to the failed node defined before can be written as

$$g^{(f)}(z) = \sum_{j=0}^{k-2} (h_j + a_f^{k-1} h_{k-1+j}) z^j.$$

- Rephrasing, the contents of the node f is

$$(h_0 + a_f^{k-1} h_{k-1}, h_1 + a_f^{k-1} h_k, \ldots, h_{k-2} + a_f^{k-1} h_{2k-3})^{\mathsf{T}}.$$

- At the same time, we can write $H(z)$ in the Lagrange form $H(z) = \sum_{h \in D} H(a_h) l^{(h)}(z)$. where $H(a_h) = g^{(h)}(a_f)$ due to the symmetry of s_1, s_2.

Achieving the bounds: Proving correctness

Claim

The content of the failed node f coincides with the vector $\xi(f,D)$, i.e., $g^{(f)}(z) = \sum_{i=0}^{l-1}(\xi(f,D))_i z^i$.

- Consider the polynomial $H(z) = s_1(a_f,z) + z^{k-1}s_2(a_f,z) = \sum_{j=0}^{d-1} h_j z^j$, $\deg(H) \leqslant 2k-3 = d-1$.
- The polynomial corresponding to the failed node defined before can be written as

$$g^{(f)}(z) = \sum_{j=0}^{k-2}(h_j + a_f^{k-1}h_{k-1+j})z^j.$$

- Rephrasing, the contents of the node f is

$$(h_0 + a_f^{k-1}h_{k-1}, h_1 + a_f^{k-1}h_k, \ldots, h_{k-2} + a_f^{k-1}h_{2k-3})^\intercal.$$

- At the same time, we can write $H(z)$ in the Lagrange form $H(z) = \sum_{h\in D} H(a_h)l^{(h)}(z)$. where $H(a_h) = g^{(h)}(a_f)$ due to the symmetry of s_1, s_2.
- It follows that the coefficients of the polynomial $g^{(f)}(z)$ is nothing but $\xi(f,D)$.

PM codes in Multi-linear Algebra framework

PM codes in Multi-linear Algebra framework

Some notation

PM codes in Multi-linear Algebra framework

Some notation

- For vector space U, define functional $\phi : U \to F$. For $V \subseteq U$, $\phi \restriction V$ is the restriction of ϕ to V. The dual space of U is U^*.

PM codes in Multi-linear Algebra framework

Some notation

- For vector space U, define functional $\phi : U \to F$. For $V \subseteq U$, $\phi \restriction V$ is the restriction of ϕ to V. The dual space of U is U^*.
- The tensor product of U and V:

$$U \otimes V = \{\sum_{ij} a_{ij}\bar{u}_i \otimes \bar{v}_j, a_{ij} \in F\}.$$

$T^pV := V \otimes V \otimes \cdots \otimes V, dim(T^pV) = n^p$

PM codes in Multi-linear Algebra framework

Some notation

- For vector space U, define functional $\phi : U \to F$. For $V \subseteq U$, $\phi \restriction V$ is the restriction of ϕ to V. The dual space of U is U^*.
- The tensor product of U and V:

$$U \otimes V = \{\sum_{ij} a_{ij}\bar{u}_i \otimes \bar{v}_j, a_{ij} \in F\}.$$

$T^p V := V \otimes V \otimes \cdots \otimes V, dim(T^p V) = n^p$

- The symmetric product of U and V:

$$U \odot V = \{\sum_{i \leqslant j} a_{ij}\bar{u}_i \odot \bar{v}_j, a_{ij} \in F\}$$

$S^p V = V \odot V \odot \cdots \odot V$, $dim(S^p V) = \binom{n+p-1}{p}$

PM codes in Multi-linear Algebra framework

PM codes in Multi-linear Algebra framework

Code Construction [Duursma *et al.* 2021]

PM codes in Multi-linear Algebra framework

Code Construction [Duursma *et al.* 2021]

- Let $X = F^2$ and $Y = F^{k-1}$. Let $L := X \otimes S^2 Y$ and note that $\dim(L) = 2 \cdot \binom{k}{2} = M$. The encoding $\phi : L \to F$ correspond to the message symbols.

PM codes in Multi-linear Algebra framework

Code Construction [Duursma *et al.* 2021]

- Let $X = F^2$ and $Y = F^{k-1}$. Let $L := X \otimes S^2 Y$ and note that $\dim(L) = 2 \cdot \binom{k}{2} = M$. The encoding $\phi : L \to F$ correspond to the message symbols.
- For each $i \in [n]$, a pair of vectors $x_i \in X$ and $y_i \in Y$ such that
  (i) Any 2 vectors of x_i's spans X.
  (ii) Any $(k-1)$-subset of y_i's spans Y.
  (iii) Any d subspaces $x_i \otimes y_i$ span $X \otimes Y$.

PM codes in Multi-linear Algebra framework

Code Construction [Duursma *et al.* 2021]

- Let $X = F^2$ and $Y = F^{k-1}$. Let $L := X \otimes S^2 Y$ and note that $\dim(L) = 2 \cdot \binom{k}{2} = M$. The encoding $\phi : L \to F$ correspond to the message symbols.
- For each $i \in [n]$, a pair of vectors $x_i \in X$ and $y_i \in Y$ such that
  - (i) Any 2 vectors of x_i's spans X.
  - (ii) Any $(k-1)$-subset of y_i's spans Y.
  - (iii) Any d subspaces $x_i \otimes y_i$ span $X \otimes Y$.
- The contents of node i corresponds to the restriction $\phi \restriction x_i \otimes y_i \odot Y \in (x_i \otimes y_i \odot Y)^*$.

PM codes in Multi-linear Algebra framework

Code Construction [Duursma *et al.* 2021]

- Let $X = F^2$ and $Y = F^{k-1}$. Let $L := X \otimes S^2 Y$ and note that $\dim(L) = 2 \cdot \binom{k}{2} = M$. The encoding $\phi : L \to F$ correspond to the message symbols.
- For each $i \in [n]$, a pair of vectors $x_i \in X$ and $y_i \in Y$ such that
  - (i) Any 2 vectors of x_i's spans X.
  - (ii) Any $(k-1)$-subset of y_i's spans Y.
  - (iii) Any d subspaces $x_i \otimes y_i$ span $X \otimes Y$.
- The contents of node i corresponds to the restriction $\phi \restriction x_i \otimes y_i \odot Y \in (x_i \otimes y_i \odot Y)^*$.

Intermediate Processing

PM codes in Multi-linear Algebra framework

Code Construction [Duursma *et al.* 2021]

- Let $X = F^2$ and $Y = F^{k-1}$. Let $L := X \otimes S^2 Y$ and note that $\dim(L) = 2 \cdot \binom{k}{2} = M$. The encoding $\phi : L \to F$ correspond to the message symbols.
- For each $i \in [n]$, a pair of vectors $x_i \in X$ and $y_i \in Y$ such that
  (i) Any 2 vectors of x_i's spans X.
  (ii) Any $(k-1)$-subset of y_i's spans Y.
  (iii) Any d subspaces $x_i \otimes y_i$ span $X \otimes Y$.
- The contents of node i corresponds to the restriction $\phi \restriction x_i \otimes y_i \odot Y \in (x_i \otimes y_i \odot Y)^*$.

Intermediate Processing

- We wish to recover the restriction $\phi \restriction x_f \otimes y_f \odot Y$.

PM codes in Multi-linear Algebra framework

Code Construction [Duursma *et al.* 2021]

- Let $X = F^2$ and $Y = F^{k-1}$. Let $L := X \otimes S^2 Y$ and note that $\dim(L) = 2 \cdot \binom{k}{2} = M$. The encoding $\phi : L \to F$ correspond to the message symbols.
- For each $i \in [n]$, a pair of vectors $x_i \in X$ and $y_i \in Y$ such that
  (i) Any 2 vectors of x_i's spans X.
  (ii) Any $(k-1)$-subset of y_i's spans Y.
  (iii) Any d subspaces $x_i \otimes y_i$ span $X \otimes Y$.
- The contents of node i corresponds to the restriction $\phi \restriction x_i \otimes y_i \odot Y \in (x_i \otimes y_i \odot Y)^*$.

Intermediate Processing

- We wish to recover the restriction $\phi \restriction x_f \otimes y_f \odot Y$.
- Helper node $i \in D$ sends ϕ evaluated at $x_i \otimes y_i \odot y_f$.

PM codes in Multi-linear Algebra framework

Code Construction [Duursma *et al.* 2021]

- Let $X = F^2$ and $Y = F^{k-1}$. Let $L := X \otimes S^2 Y$ and note that $\dim(L) = 2 \cdot \binom{k}{2} = M$. The encoding $\phi : L \to F$ correspond to the message symbols.
- For each $i \in [n]$, a pair of vectors $x_i \in X$ and $y_i \in Y$ such that
  - (i) Any 2 vectors of x_i's spans X.
  - (ii) Any $(k-1)$-subset of y_i's spans Y.
  - (iii) Any d subspaces $x_i \otimes y_i$ span $X \otimes Y$.
- The contents of node i corresponds to the restriction $\phi \restriction x_i \otimes y_i \odot Y \in (x_i \otimes y_i \odot Y)^*$.

Intermediate Processing

- We wish to recover the restriction $\phi \restriction x_f \otimes y_f \odot Y$.
- Helper node $i \in D$ sends ϕ evaluated at $x_i \otimes y_i \odot y_f$.
- $x_f \otimes y_f \odot \bar{y}_1 = x_f \otimes \bar{y}_1 \odot y_f = \sum_{i \in D} a_i (x_i \otimes y_i) \odot y_f$.

PM codes in Multi-linear Algebra framework

Code Construction [Duursma *et al.* 2021]

- Let $X = F^2$ and $Y = F^{k-1}$. Let $L := X \otimes S^2 Y$ and note that $\dim(L) = 2 \cdot \binom{k}{2} = M$. The encoding $\phi : L \to F$ correspond to the message symbols.
- For each $i \in [n]$, a pair of vectors $x_i \in X$ and $y_i \in Y$ such that
  - (i) Any 2 vectors of x_i's spans X.
  - (ii) Any $(k-1)$-subset of y_i's spans Y.
  - (iii) Any d subspaces $x_i \otimes y_i$ span $X \otimes Y$.
- The contents of node i corresponds to the restriction $\phi \restriction x_i \otimes y_i \odot Y \in (x_i \otimes y_i \odot Y)^*$.

Intermediate Processing

- We wish to recover the restriction $\phi \restriction x_f \otimes y_f \odot Y$.
- Helper node $i \in D$ sends ϕ evaluated at $x_i \otimes y_i \odot y_f$.
- $x_f \otimes y_f \odot \bar{y}_1 = x_f \otimes \bar{y}_1 \odot y_f = \sum_{i \in D} a_i (x_i \otimes y_i) \odot y_f$.
- Set A transmits $\sum_{i \in A} a_i \phi(x_i \otimes y_i \odot y_f)$.

PM codes in Multi-linear Algebra framework

Code Construction [Duursma *et al.* 2021]

- Let $X = F^2$ and $Y = F^{k-1}$. Let $L := X \otimes S^2 Y$ and note that $\dim(L) = 2 \cdot \binom{k}{2} = M$. The encoding $\phi : L \to F$ correspond to the message symbols.
- For each $i \in [n]$, a pair of vectors $x_i \in X$ and $y_i \in Y$ such that
  - (i) Any 2 vectors of x_i's spans X.
  - (ii) Any $(k-1)$-subset of y_i's spans Y.
  - (iii) Any d subspaces $x_i \otimes y_i$ span $X \otimes Y$.
- The contents of node i corresponds to the restriction $\phi \restriction x_i \otimes y_i \odot Y \in (x_i \otimes y_i \odot Y)^*$.

Intermediate Processing

- We wish to recover the restriction $\phi \restriction x_f \otimes y_f \odot Y$.
- Helper node $i \in D$ sends ϕ evaluated at $x_i \otimes y_i \odot y_f$.
- $x_f \otimes y_f \odot \bar{y}_1 = x_f \otimes \bar{y}_1 \odot y_f = \sum_{i \in D} a_i (x_i \otimes y_i) \odot y_f$.
- Set A transmits $\sum_{i \in A} a_i \phi(x_i \otimes y_i \odot y_f)$.

Extension

PM codes in Multi-linear Algebra framework

Code Construction [Duursma *et al.* 2021]

- Let $X = F^2$ and $Y = F^{k-1}$. Let $L := X \otimes S^2 Y$ and note that $\dim(L) = 2 \cdot \binom{k}{2} = M$. The encoding $\phi : L \to F$ correspond to the message symbols.
- For each $i \in [n]$, a pair of vectors $x_i \in X$ and $y_i \in Y$ such that
  - (i) Any 2 vectors of x_i's spans X.
  - (ii) Any $(k-1)$-subset of y_i's spans Y.
  - (iii) Any d subspaces $x_i \otimes y_i$ span $X \otimes Y$.
- The contents of node i corresponds to the restriction $\phi \restriction x_i \otimes y_i \odot Y \in (x_i \otimes y_i \odot Y)^*$.

Intermediate Processing

- We wish to recover the restriction $\phi \restriction x_f \otimes y_f \odot Y$.
- Helper node $i \in D$ sends ϕ evaluated at $x_i \otimes y_i \odot y_f$.
- $x_f \otimes y_f \odot \bar{y}_1 = x_f \otimes \bar{y}_1 \odot y_f = \sum_{i \in D} a_i (x_i \otimes y_i) \odot y_f$.
- Set A transmits $\sum_{i \in A} a_i \phi(x_i \otimes y_i \odot y_f)$.

Extension

- This framework can be extended to get an MSR code with $n, k, d = \frac{(k-1)t}{t-1}, l = \binom{k-1}{t-1}, M = t\binom{k}{t}$ for any $t \geqslant 2$.

PM codes in Multi-linear Algebra framework

Code Construction [Duursma *et al.* 2021]

- Let $X = F^2$ and $Y = F^{k-1}$. Let $L := X \otimes S^2 Y$ and note that $\dim(L) = 2 \cdot \binom{k}{2} = M$. The encoding $\phi : L \to F$ correspond to the message symbols.
- For each $i \in [n]$, a pair of vectors $x_i \in X$ and $y_i \in Y$ such that
  - (i) Any 2 vectors of x_i's spans X.
  - (ii) Any $(k-1)$-subset of y_i's spans Y.
  - (iii) Any d subspaces $x_i \otimes y_i$ span $X \otimes Y$.
- The contents of node i corresponds to the restriction $\phi \restriction x_i \otimes y_i \odot Y \in (x_i \otimes y_i \odot Y)^*$.

Intermediate Processing

- We wish to recover the restriction $\phi \restriction x_f \otimes y_f \odot Y$.
- Helper node $i \in D$ sends ϕ evaluated at $x_i \otimes y_i \odot y_f$.
- $x_f \otimes y_f \odot \bar{y}_1 = x_f \otimes \bar{y}_1 \odot y_f = \sum_{i \in D} a_i (x_i \otimes y_i) \odot y_f$.
- Set A transmits $\sum_{i \in A} a_i \phi(x_i \otimes y_i \odot y_f)$.

Extension

- This framework can be extended to get an MSR code with $n, k, d = \frac{(k-1)t}{t-1}, l = \binom{k-1}{t-1}, M = t\binom{k}{t}$ for any $t \geqslant 2$.
- The same ideas of intermediate processing applies.

Extension to Interior Point Codes: Moving away from MSR

Extension to Interior Point Codes: Moving away from MSR

- Now focus on non-MSR interior points of the trade-off: $l > (d-k+1)\beta$.

Extension to Interior Point Codes: Moving away from MSR

- Now focus on non-MSR interior points of the trade-off: $l > (d - k + 1)\beta$.
- Code constructions are not MDS anymore.

Extension to Interior Point Codes: Moving away from MSR

- Now focus on non-MSR interior points of the trade-off: $l > (d-k+1)\beta$.
- Code constructions are not MDS anymore.
- A gap exists between functional and exact repair.

Extension to Interior Point Codes: Moving away from MSR

- Now focus on non-MSR interior points of the trade-off: $l > (d - k + 1)\beta$.
- Code constructions are not MDS anymore.
- A gap exists between functional and exact repair.

What about lower bounds?

Extension to Interior Point Codes: Moving away from MSR

- Now focus on non-MSR interior points of the trade-off: $l > (d - k + 1)\beta$.
- Code constructions are not MDS anymore.
- A gap exists between functional and exact repair.

What about lower bounds?

- Same lower bound is still applicable, i.e., any set of size at least $d - k + 1$ needs to transmit l symbols.

Extension to Interior Point Codes: Moving away from MSR

- Now focus on non-MSR interior points of the trade-off: $l > (d-k+1)\beta$.
- Code constructions are not MDS anymore.
- A gap exists between functional and exact repair.

What about lower bounds?

- Same lower bound is still applicable, i.e., any set of size at least $d-k+1$ needs to transmit l symbols.
- Unlike the MSR point, this bound might not be tight anymore.

Extension to Interior Point Codes: Moving away from MSR

- Now focus on non-MSR interior points of the trade-off: $l > (d - k + 1)\beta$.
- Code constructions are not MDS anymore.
- A gap exists between functional and exact repair.

What about lower bounds?

- Same lower bound is still applicable, i.e., any set of size at least $d - k + 1$ needs to transmit l symbols.
- Unlike the MSR point, this bound might not be tight anymore.
- Improvement: For any $E \subset D$, $H(R_E^f) \geqslant \frac{|E|l}{d}$. The bound is tighter for $|E| > (d - k + 1)\beta d/l$.

Extension to Interior Point Codes: Moving away from MSR

- Now focus on non-MSR interior points of the trade-off: $l > (d - k + 1)\beta$.
- Code constructions are not MDS anymore.
- A gap exists between functional and exact repair.

What about lower bounds?

- Same lower bound is still applicable, i.e., any set of size at least $d - k + 1$ needs to transmit l symbols.
- Unlike the MSR point, this bound might not be tight anymore.
- Improvement: For any $E \subset D$, $H(R_E^f) \geqslant \frac{|E|l}{d}$. The bound is tighter for $|E| > (d - k + 1)\beta d/l$.

What can we achieve

Extension to Interior Point Codes: Moving away from MSR

- Now focus on non-MSR interior points of the trade-off: $l > (d-k+1)\beta$.
- Code constructions are not MDS anymore.
- A gap exists between functional and exact repair.

What about lower bounds?

- Same lower bound is still applicable, i.e., any set of size at least $d-k+1$ needs to transmit l symbols.
- Unlike the MSR point, this bound might not be tight anymore.
- Improvement: For any $E \subset D$, $H(R_E^f) \geqslant \frac{|E|l}{d}$. The bound is tighter for $|E| > (d-k+1)\beta d/l$.

What can we achieve

- Explicit code constructions for interior points: Determinant Codes [Elyasi *et al.* 2016], Moulin Codes [Duursma *et al.* 2021] can be fit into the Intermediate Processing framework.

Extension to Interior Point Codes: Moving away from MSR

- Now focus on non-MSR interior points of the trade-off: $l > (d-k+1)\beta$.
- Code constructions are not MDS anymore.
- A gap exists between functional and exact repair.

What about lower bounds?

- Same lower bound is still applicable, i.e., any set of size at least $d-k+1$ needs to transmit l symbols.
- Unlike the MSR point, this bound might not be tight anymore.
- Improvement: For any $E \subset D$, $H(R_E^f) \geqslant \frac{|E|l}{d}$. The bound is tighter for $|E| > (d-k+1)\beta d/l$.

What can we achieve

- Explicit code constructions for interior points: Determinant Codes [Elyasi *et al.* 2016], Moulin Codes [Duursma *et al.* 2021] can be fit into the Intermediate Processing framework.
- A set of size at least $d-k+1$ need only transmit l symbols.

Application to random graphs

Application to random graphs

- Erdös-Rényi random graph $\mathcal{G}_{n,p}$: n nodes, each edge present with probability p independently of others.

Application to random graphs

- Erdös-Rényi random graph $\mathcal{G}_{n,p}$: n nodes, each edge present with probability p independently of others.

- An $[n, k, d]$ MSR code defined with C_i stored in node i.

Application to random graphs

- Erdös-Rényi random graph $\mathcal{G}_{n,p}$: n nodes, each edge present with probability p independently of others.
- An $[n, k, d]$ MSR code defined with C_i stored in node i.
- For asymptotically positive rates, $d = \Theta(n)$.

Application to random graphs

- Erdös-Rényi random graph $\mathcal{G}_{n,p}$: n nodes, each edge present with probability p independently of others.
- An $[n, k, d]$ MSR code defined with C_i stored in node i.
- For asymptotically positive rates, $d = \Theta(n)$.
- Operate in $p \gg \frac{\log n}{n}$ region, i.e., connected region.

Application to random graphs

- Erdös-Rényi random graph $\mathcal{G}_{n,p}$: n nodes, each edge present with probability p independently of others.
- An $[n, k, d]$ MSR code defined with C_i stored in node i.
- For asymptotically positive rates, $d = \Theta(n)$.
- Operate in $p \gg \frac{\log n}{n}$ region, i.e., connected region.
  - Node repair possible with high probability.

Application to random graphs

- Erdös-Rényi random graph $\mathcal{G}_{n,p}$: n nodes, each edge present with probability p independently of others.
- An $[n, k, d]$ MSR code defined with C_i stored in node i.
- For asymptotically positive rates, $d = \Theta(n)$.
- Operate in $p \gg \frac{\log n}{n}$ region, i.e., connected region.
  - Node repair possible with high probability.
- Threshold behavior: We say that t*-layer repair* of the failed node v *is possible* if

$$P(|N_t(v)| \geqslant d) \to 1 \text{ as } n \to \infty.$$

Application to random graphs: Repair threshold

Proposition

Let $d = \delta n, 0 < \delta < 1$ be a constant and let t be a fixed integer. Then t is the threshold depth for repair if

$$(np)^{t-1} = o(n), \quad p^t n^{t-1} - 2\log n \to \infty.$$

Application to random graphs: Repair threshold

Proposition

Let $d = \delta n, 0 < \delta < 1$ be a constant and let t be a fixed integer. Then t is the threshold depth for repair if

$$(np)^{t-1} = o(n), \quad p^t n^{t-1} - 2\log n \to \infty.$$

- Proof by classical results:

Application to random graphs: Repair threshold

Proposition

Let $d = \delta n, 0 < \delta < 1$ be a constant and let t be a fixed integer. Then t is the threshold depth for repair if

$$(np)^{t-1} = o(n), \quad p^t n^{t-1} - 2\log n \to \infty.$$

- Proof by classical results:
  - If the above conditions hold, then $\text{diam}(G) = t$ with high probability [See Bollobas '81; Frieze and Karoński 2015].

Application to random graphs: Repair threshold

Proposition

Let $d = \delta n, 0 < \delta < 1$ be a constant and let t be a fixed integer. Then t is the threshold depth for repair if

$$(np)^{t-1} = o(n), \quad p^t n^{t-1} - 2\log n \to \infty.$$

- Proof by classical results:
  - If the above conditions hold, then $\mathrm{diam}(G) = t$ with high probability [See Bollobas '81; Frieze and Karoński 2015].
  - If $(np)^{t-1} = o(n)$ and $d = \Theta(n)$, then with high probability, d nodes are not reached in $t-1$ layers [Chung et. al. 2001].

Application to random graphs: Repair bandwidth

- How much savings do we get by using the "Intermediate Processing" technique compared to simple "Accumulate and Forward" relaying?

Application to random graphs: Repair bandwidth

- How much savings do we get by using the "Intermediate Processing" technique compared to simple "Accumulate and Forward" relaying?

Proposition

The repair bandwidth β_{AF} *satisfies* $\mathbb{P}(\beta_{\mathrm{AF}} \geqslant td - o(n)) \to 1$

Application to random graphs: Repair bandwidth

- How much savings do we get by using the "Intermediate Processing" technique compared to simple "Accumulate and Forward" relaying?

Proposition

The repair bandwidth β_{AF} *satisfies* $\mathbb{P}(\beta_{\mathrm{AF}} \geqslant td - o(n)) \to 1$

- Most nodes are at distance t. For the AF setting, required bandwidth scaled by t compared to the directly connected scenario.

Application to random graphs: Repair bandwidth

- How much savings do we get by using the "Intermediate Processing" technique compared to simple "Accumulate and Forward" relaying?

Proposition

The repair bandwidth β_{AF} satisfies $\mathbb{P}(\beta_{\mathrm{AF}} \geqslant td - o(n)) \to 1$

- Most nodes are at distance t. For the AF setting, required bandwidth scaled by t compared to the directly connected scenario.

Theorem

Let t be the threshold for repair. For $d = \Theta(n)$ let $d - k = \chi(n)$ be a function of n such that $\chi(n)n^{s-1}p^s \to 0$ where $s \leqslant t - 1$ is the largest integer for which this condition holds. Then $\mathbb{P}(\beta_{\mathrm{IP}} \leqslant (t - s)d + o(n)) \to 1.$

Application to random graphs: Repair bandwidth

- How much savings do we get by using the "Intermediate Processing" technique compared to simple "Accumulate and Forward" relaying?

Proposition

The repair bandwidth β_{AF} *satisfies* $\mathbb{P}(\beta_{\mathrm{AF}} \geqslant td - o(n)) \to 1$

- Most nodes are at distance t. For the AF setting, required bandwidth scaled by t compared to the directly connected scenario.

Theorem

Let t *be the threshold for repair. For* $d = \Theta(n)$ *let* $d - k = \chi(n)$ *be a function of* n *such that* $\chi(n) n^{s-1} p^s \to 0$ *where* $s \leqslant t - 1$ *is the largest integer for which this condition holds. Then* $\mathbb{P}(\beta_{\mathrm{IP}} \leqslant (t - s)d + o(n)) \to 1$.

- Using Intermediate Processing, the scaling of the bandwidth can be brought down from t to $t - s$.

Concluding remarks

- In sparsely connected graphs, it is possible to do better than simple relaying.

Concluding remarks

- In sparsely connected graphs, it is possible to do better than simple relaying.
- The intermediate processing technique is applicable to all $\mathbb{F}$-linear MSR codes as well as interior point codes and achieves the minimum possible communication in some cases.

Concluding remarks

- In sparsely connected graphs, it is possible to do better than simple relaying.
- The intermediate processing technique is applicable to all $\mathbb{F}$-linear MSR codes as well as interior point codes and achieves the minimum possible communication in some cases.
- For random graphs $\mathcal{G}_{n,p}$, in certain regimes, intermediate processing can give significant reductions in communication overhead.