Decoding codes with locality

Julien Lavauzelle

LAGA, Université Paris 8 Vincennes – Saint-Denis

Symposium CAIPI

11-12/12/2025

Outline

Outline

1. Motivation

2. Local recovery

3. Regeneration

4. Local correction

Applications of codes

Main application of codes: **communication** in presence of noise.

- historically with combinatorial/algebraic block codes: binary Reed–Muller codes, Golay codes, BCH codes, etc.

Applications of codes

Main application of codes: **communication** in presence of noise.

- historically with combinatorial/algebraic block codes: binary Reed–Muller codes, Golay codes, BCH codes, etc.
- with convolutional codes and soft-decision decoders (e.g. turbo codes) since the 1990s

Applications of codes

Main application of codes: **communication** in presence of noise.

- historically with combinatorial/algebraic block codes: binary Reed–Muller codes, Golay codes, BCH codes, etc.
- with convolutional codes and soft-decision decoders (e.g. turbo codes) since the 1990s

Algebraic codes are now also used for **data storage**:

- on discs, e.g. Reed–Solomon codes in CDs, DVDs, hard disk drives,...
- for labelling or redirecting: bar codes, QR codes,...
- for massive data storage in **distributed storage systems** (DSS),
- etc.

Main application of codes: **communication** in presence of noise.

- historically with combinatorial/algebraic block codes: binary Reed–Muller codes, Golay codes, BCH codes, etc.
- with convolutional codes and soft-decision decoders (e.g. turbo codes) since the 1990s

Algebraic codes are now also used for **data storage**:

- on discs, e.g. Reed–Solomon codes in CDs, DVDs, hard disk drives,...
- for labelling or redirecting: bar codes, QR codes,...
- for massive data storage in **distributed storage systems** (DSS),
- etc.

Context of this talk: distributed storage systems.

Goal: efficiently store a set of **large files** on **several servers**.

Goal: efficiently store a set of **large files** on **several servers**.

Desired features:

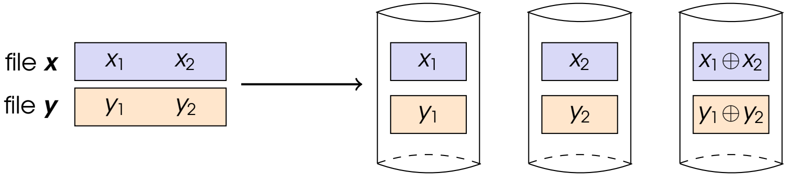
- "easy" to download data from servers:
  - low bandwidth
  - few servers to be contacted
  - multiple/balanced access to servers
- resilience to server failures, delay, etc.
- security against eavesdroppers, corruption, etc.

Goal: efficiently store a set of **large files** on **several servers**.

Desired features:

- "easy" to download data from servers:
  - low bandwidth
  - few servers to be contacted
  - multiple/balanced access to servers
- resilience to server failures, delay, etc.
- security against eavesdroppers, corruption, etc.

Codes are used to distribute files on the servers, and ensure these features.



A first simple storage model

Context:

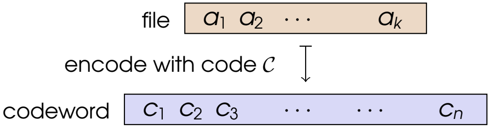
- n servers $\mathcal{S}_1, \dots, \mathcal{S}_n$,
- each file is split into information symbols $a_1, \dots, a_k$,
- we assume $a_i \in \mathbb{F}_q$ (up to file striping).

file $\boxed{a_1 \ a_2 \ \cdots \ a_k}$

A first simple storage model

Context:

- n servers $\mathcal{S}_1, \dots, \mathcal{S}_n$,
- each file is split into information symbols $a_1, \dots, a_k$,
- we assume $a_i \in \mathbb{F}_q$ (up to file striping).



Consider a code

$$\mathcal{C} : \mathbb{F}_q^k \to \mathbb{F}_q^n$$

Encode the file $\boldsymbol{a} = (a_1, \dots, a_k)$
into a codeword $\boldsymbol{c} = (c_1, \dots, c_n) \in \mathcal{C}$.

A first simple storage model

Context:

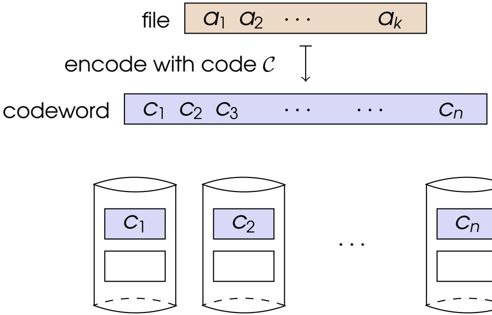
- n servers $\mathcal{S}_1, \dots, \mathcal{S}_n$,
- each file is split into information symbols $a_1, \dots, a_k$,
- we assume $a_i \in \mathbb{F}_q$ (up to file striping).

Consider a code

$$\mathcal{C} : \mathbb{F}_q^k \to \mathbb{F}_q^n$$

Encode the file $\boldsymbol{a} = (a_1, \dots, a_k)$
into a codeword $\boldsymbol{c} = (c_1, \dots, c_n) \in \mathcal{C}$.

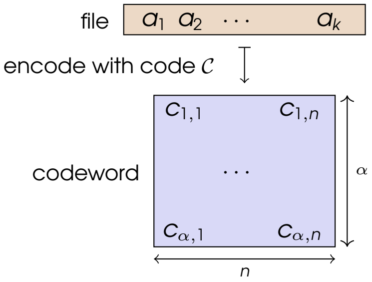
Server $\mathcal{S}_i$ stores symbol c_i.



Same context as before, but codes are **array codes**:

$$\mathcal{C} : \mathbb{F}_q^k \to \mathbb{F}_q^{\alpha \times n}$$

Encode the file $\boldsymbol{a} = (a_1, \dots, a_k)$
into a (matrix) codeword $\boldsymbol{C} = (\boldsymbol{c}_1, \dots, \boldsymbol{c}_n) \in \mathcal{C}$.

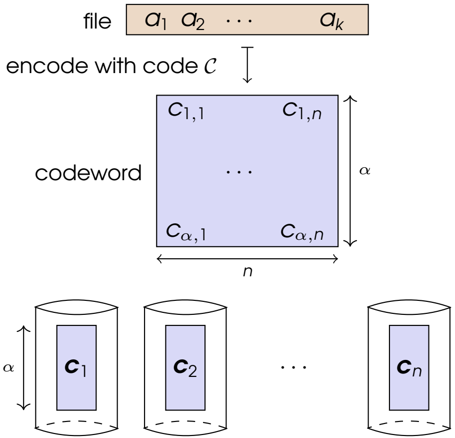


Same context as before, but codes are **array codes**:

$$\mathcal{C} : \mathbb{F}_q^k \to \mathbb{F}_q^{\alpha \times n}$$

Encode the file $\boldsymbol{a} = (a_1, \dots, a_k)$
into a (matrix) codeword $\boldsymbol{C} = (\boldsymbol{c}_1, \dots, \boldsymbol{c}_n) \in \mathcal{C}$.

Server $\mathcal{S}_i$ stores α symbols $\boldsymbol{c}_i \in \mathbb{F}_q^\alpha$.



Relevant quantities

Low redundancy: store files with low redundancy.

Extractability: download files with few servers access.

Recovery/regeneration: if some server $\mathcal{S}_i$ fails, recover/regenerate the data stored on it:

- with few servers access: see **locally recoverable codes**
- with low bandwidth: see **regenerating codes**

Outline

Illustration for locality

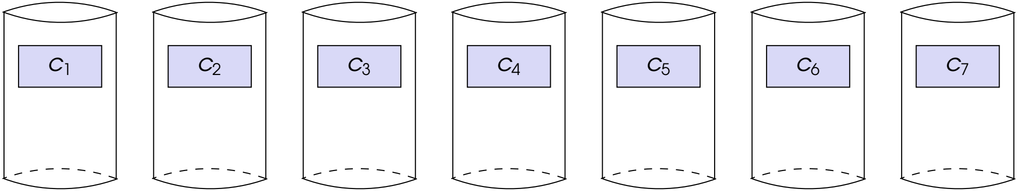


Illustration for locality

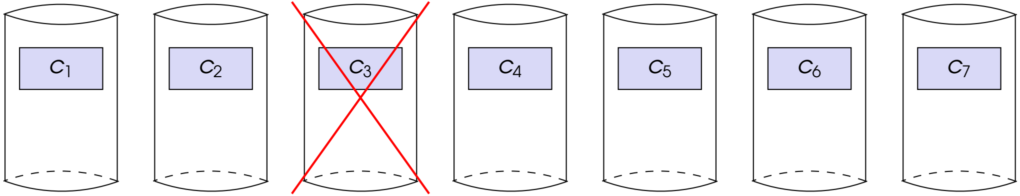


Illustration for locality

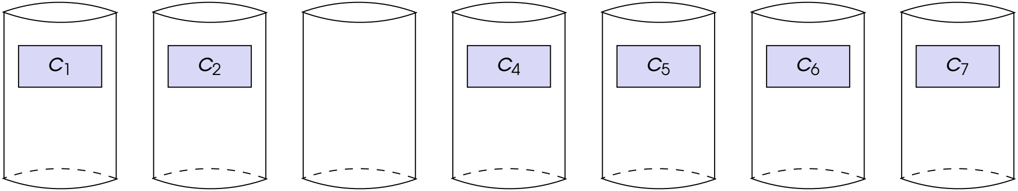


Illustration for locality

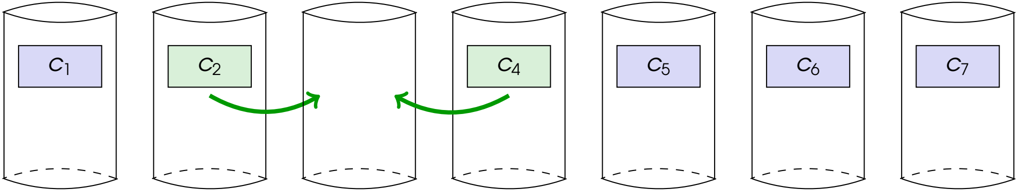


Illustration for locality

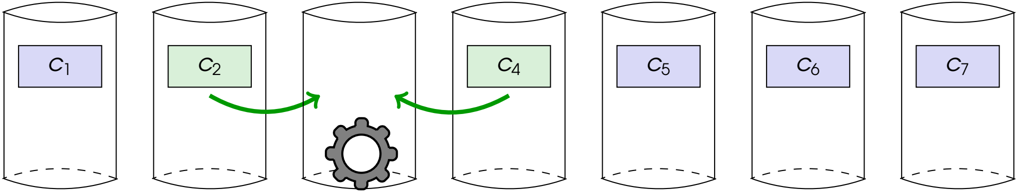
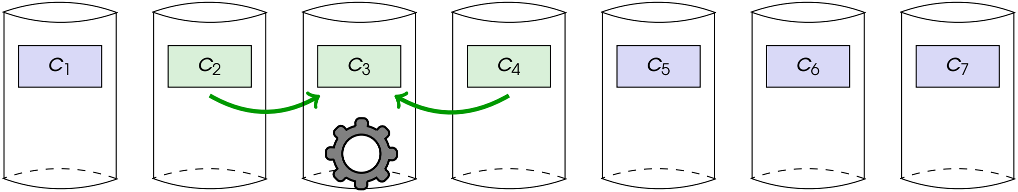


Illustration for locality



Symbol and code locality

On the Locality of Codeword Symbols. P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin. IEEE TIT. **2011**.

Symbol and code locality

On the Locality of Codeword Symbols. P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin. IEEE TIT. **2011**.

Let $\mathcal{C} \subseteq \mathbb{F}_q^n$ be a code of dimension k.

Def. A coordinate $i \in [1, n]$ has **locality** $r \geq 1$ in the code $\mathcal{C}$, if there exists a subset $\mathcal{R}_i \subset [1, n] \setminus \{i\}$ of size at most r such that:

$$\dim \mathcal{C}_{|\mathcal{R}_i} = \dim \mathcal{C}_{|\mathcal{R}_i \cup \{i\}}$$

Subset $\mathcal{R}_i$ is called a **helper set** for coordinate i.

coordinate i has locality r in $\mathcal{C}$ $\iff \exists(\lambda_{i_1}, \dots, \lambda_{i_r}) \in \mathbb{F}_q^r,\ \forall \boldsymbol{c} \in \mathcal{C}, c_i = \sum_j \lambda_{i_j} c_{i_j}$

On the Locality of Codeword Symbols. P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin. IEEE TIT. **2011**.

Let $\mathcal{C} \subseteq \mathbb{F}_q^n$ be a code of dimension k.

Def. A coordinate $i \in [1, n]$ has **locality** $r \geq 1$ in the code $\mathcal{C}$, if there exists a subset $\mathcal{R}_i \subset [1, n] \setminus \{i\}$ of size at most r such that:

$$\dim \mathcal{C}_{|\mathcal{R}_i} = \dim \mathcal{C}_{|\mathcal{R}_i \cup \{i\}}$$

Subset $\mathcal{R}_i$ is called a **helper set** for coordinate i.

$$\begin{aligned} \text{coordinate } i \text{ has locality } r \text{ in } \mathcal{C} \quad &\iff \exists(\lambda_{i_1}, \dots, \lambda_{i_r}) \in \mathbb{F}_q^r,\ \forall \boldsymbol{c} \in \mathcal{C}, c_i = \textstyle\sum_j \lambda_{i_j} c_{i_j} \\ &\iff \exists \boldsymbol{h} \in \mathcal{C}^\perp,\ i \in \mathrm{supp}(\boldsymbol{h}) \text{ and } |\mathrm{supp}(\boldsymbol{h})| \leq r + 1 \end{aligned}$$

On the Locality of Codeword Symbols. P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin. IEEE TIT. **2011**.

Let $\mathcal{C} \subseteq \mathbb{F}_q^n$ be a code of dimension k.

Def. A coordinate $i \in [1, n]$ has **locality** $r \geq 1$ in the code $\mathcal{C}$, if there exists a subset $\mathcal{R}_i \subset [1, n] \setminus \{i\}$ of size at most r such that:

$$\dim \mathcal{C}_{|\mathcal{R}_i} = \dim \mathcal{C}_{|\mathcal{R}_i \cup \{i\}}$$

Subset $\mathcal{R}_i$ is called a **helper set** for coordinate i.

$$\begin{aligned} \text{coordinate } i \text{ has locality } r \text{ in } \mathcal{C} \quad &\iff \exists(\lambda_{i_1}, \dots, \lambda_{i_r}) \in \mathbb{F}_q^r,\ \forall \boldsymbol{c} \in \mathcal{C}, c_i = \textstyle\sum_j \lambda_{i_j} c_{i_j} \\ &\iff \exists \boldsymbol{h} \in \mathcal{C}^\perp,\ i \in \mathrm{supp}(\boldsymbol{h}) \text{ and } |\mathrm{supp}(\boldsymbol{h})| \leq r + 1 \end{aligned}$$

Def. The code $\mathcal{C}$ is r**-locally recoverable** (r-LR) if each coordinate $i \in [1, n]$ has locality r in $\mathcal{C}$.

1. **Repetition code.** Parameters: $[n, 1, n]_q$.

$$\mathcal{C} = \{(a, a, \dots, a), a \in \mathbb{F}_q\}$$

Examples (1)

1. **Repetition code.** Parameters: $[n, 1, n]_q$.

$$\mathcal{C} = \{(a, a, \dots, a), a \in \mathbb{F}_q\}$$

Locality $r = 1$: each symbol is a copy of another.

Examples (1)

1. **Repetition code.** Parameters: $[n, 1, n]_q$.

$$\mathcal{C} = \{(a, a, \ldots, a), a \in \mathbb{F}_q\}$$

Locality $r = 1$: each symbol is a copy of another.

2. **Parity-check code.** Parameters: $[n, n-1, 2]_q$.

$$\mathcal{C} = \left\{ \boldsymbol{c} \in \mathbb{F}_q^n \mid \sum_{i=1}^{n} c_i = 0 \right\}$$

Examples (1)

1. **Repetition code.** Parameters: $[n, 1, n]_q$.

$$\mathcal{C} = \{(a, a, \dots, a), a \in \mathbb{F}_q\}$$

Locality $r = 1$: each symbol is a copy of another.

2. **Parity-check code.** Parameters: $[n, n-1, 2]_q$.

$$\mathcal{C} = \left\{ \boldsymbol{c} \in \mathbb{F}_q^n \mid \sum_{i=1}^{n} c_i = 0 \right\}$$

Locality $r = n - 1$, since $c_i = -\sum_{j \neq i} c_j$, for all $\boldsymbol{c} \in \mathcal{C}$.

Examples (1)

1. **Repetition code.** Parameters: $[n, 1, n]_q$.

$$\mathcal{C} = \{(a, a, \dots, a), a \in \mathbb{F}_q\}$$

Locality $r = 1$: each symbol is a copy of another.

2. **Parity-check code.** Parameters: $[n, n-1, 2]_q$.

$$\mathcal{C} = \left\{ \boldsymbol{c} \in \mathbb{F}_q^n \mid \sum_{i=1}^{n} c_i = 0 \right\}$$

Locality $r = n - 1$, since $c_i = -\sum_{j \neq i} c_j$, for all $\boldsymbol{c} \in \mathcal{C}$.

3. **MDS codes.** Parameters: $[n, k, n-k+1]_q$ for $1 \le k \le n-1$.

$$\mathcal{C} \text{ is MDS} \iff \forall I \subset [1, n], \#I = k, \text{ we have } \mathcal{C}_{|I} = \mathbb{F}_q^k$$

Examples (1)

1. **Repetition code.** Parameters: $[n, 1, n]_q$.

$$\mathcal{C} = \{(a, a, \dots, a), a \in \mathbb{F}_q\}$$

Locality $r = 1$: each symbol is a copy of another.

2. **Parity-check code.** Parameters: $[n, n-1, 2]_q$.

$$\mathcal{C} = \left\{ \boldsymbol{c} \in \mathbb{F}_q^n \mid \sum_{i=1}^{n} c_i = 0 \right\}$$

Locality $r = n - 1$, since $c_i = -\sum_{j \neq i} c_j$, for all $\boldsymbol{c} \in \mathcal{C}$.

3. **MDS codes.** Parameters: $[n, k, n-k+1]_q$ for $1 \leq k \leq n-1$.

$$\mathcal{C} \text{ is MDS} \iff \forall I \subset [1, n], \#I = k, \text{ we have } \mathcal{C}_{|I} = \mathbb{F}_q^k$$

Locality $r = k$. Choose I such that $i \notin I$, and compute c_i from $\mathcal{C}_{|I}$.

Q. How small/large is the locality parameter r?

Properties (1)

Q. How small/large is the locality parameter r?

Prop. Let $\mathcal{C} \subseteq \mathbb{F}_q^n$ be an $[n, k]_q$-LRC with locality r. Then we have

$$d^\perp - 1 \leq r \leq k\,,$$

where $d^\perp$ is the minimum distance of $\mathcal{C}^\perp$.

Q. How small/large is the locality parameter r?

Prop. Let $\mathcal{C} \subseteq \mathbb{F}_q^n$ be an $[n, k]_q$-LRC with locality r. Then we have

$$d^\perp - 1 \le r \le k\,,$$

where $d^\perp$ is the minimum distance of $\mathcal{C}^\perp$.

Proof.

- If i has locality r, then $\exists \boldsymbol{h} \in \mathcal{C}^\perp$ of weight $r + 1$.

Q. How small/large is the locality parameter r?

Prop. Let $\mathcal{C} \subseteq \mathbb{F}_q^n$ be an $[n, k]_q$-LRC with locality r. Then we have

$$d^\perp - 1 \leq r \leq k\,,$$

where $d^\perp$ is the minimum distance of $\mathcal{C}^\perp$.

Proof.

- If i has locality r, then $\exists \boldsymbol{h} \in \mathcal{C}^\perp$ of weight $r + 1$.
- If $\dim \mathcal{C} = k$, some k coordinates allow to recover any codeword entirely (hence, c_i in particular).

Q. How small/large is the locality parameter r?

Prop. Let $\mathcal{C} \subseteq \mathbb{F}_q^n$ be an $[n, k]_q$-LRC with locality r. Then we have

$$d^\perp - 1 \leq r \leq k\,,$$

where $d^\perp$ is the minimum distance of $\mathcal{C}^\perp$.

Proof.

- If i has locality r, then $\exists \boldsymbol{h} \in \mathcal{C}^\perp$ of weight $r + 1$.
- If $\dim \mathcal{C} = k$, some k coordinates allow to recover any codeword entirely (hence, c_i in particular).

Remark. MDS codes attain the upper bound. **Worst locality** compared to their dimension.

One can design LRCs via their **parity-check matrix** $\boldsymbol{H} \in \mathbb{F}_q^{(n-k)\times n}$.

$$\mathcal{C} = \left\{ \boldsymbol{c} \in \mathbb{F}_q^n \mid \boldsymbol{H}\boldsymbol{c}^\top = \boldsymbol{0} \right\}.$$

For instance, if

$$\boldsymbol{H} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

then $\mathcal{C}$ has locality $r = 3$.

Properties (2)

Q. How large can be a code, depending on its locality parameter r?

Q. How large can be a code, depending on its locality parameter r?

Prop. Let $\mathcal{C} \subseteq \mathbb{F}_q^n$ be an $[n, k]_q$-LRC with locality r. Then :

$$k \leq \frac{nr}{r+1} .$$

Proof. $\mathcal{C}^\perp$ contains at least $\frac{n}{r+1}$ codewords with disjoint support, so $\dim \mathcal{C} \leq n - \frac{n}{r+1}$.

Properties (2)

Q. How large can be a code, depending on its locality parameter r?

Prop. Let $\mathcal{C} \subseteq \mathbb{F}_q^n$ be an $[n,k]_q$-LRC with locality r. Then :

$$k \leq \frac{nr}{r+1}\,.$$

Proof. $\mathcal{C}^\perp$ contains at least $\frac{n}{r+1}$ codewords with disjoint support, so $\dim \mathcal{C} \leq n - \frac{n}{r+1}$.

Q. Do we have a bound depending on the minimum distance d (as for Singleton bound)?

A Singleton-like bound

On the Locality of Codeword Symbols. P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin. IEEE TIT. **2011**.

Prop. Let $\mathcal{C}$ be an $[n, k, d]_q$-LRC with locality r. Then:

$$d \leq n - k + 2 - \left\lceil \frac{k}{r} \right\rceil .$$

On the Locality of Codeword Symbols. P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin. IEEE TIT. **2011**.

Prop. Let $\mathcal{C}$ be an $[n, k, d]_q$-LRC with locality r. Then:

$$d \leq n - k + 2 - \left\lceil \frac{k}{r} \right\rceil .$$

Proof.
Step 1: pick $i \in [1, n]$ and **shorten** the code $\mathcal{C}$ over $\{i\} \cup \mathcal{R}_i$, of size $\leq r + 1$.

On the Locality of Codeword Symbols. P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin. IEEE TIT. **2011**.

Prop. Let $\mathcal{C}$ be an $[n, k, d]_q$-LRC with locality r. Then:

$$d \leq n - k + 2 - \left\lceil \frac{k}{r} \right\rceil .$$

Proof.

Step 1: pick $i \in [1, n]$ and **shorten** the code $\mathcal{C}$ over $\{i\} \cup \mathcal{R}_i$, of size $\leq r + 1$.
Then $\mathcal{C}' := \mathrm{Short}(\mathcal{C}, \mathcal{R})$ is a code of

- length $n' \geq n - (r + 1)$;
- dimension $k' \geq k - r$ since $\dim \mathcal{C}_{|\mathcal{R}_i} = \dim \mathcal{C}_{|\mathcal{R}_i \cup \{i\}}$;
- minimum distance $d' \geq d$ by inclusion of codes;
- locality $r' \leq r$.

A Singleton-like bound

On the Locality of Codeword Symbols. P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin. IEEE TIT. **2011**.

Prop. Let $\mathcal{C}$ be an $[n, k, d]_q$-LRC with locality r. Then:

$$d \le n - k + 2 - \left\lceil \frac{k}{r} \right\rceil .$$

Proof.

Step 1: pick $i \in [1, n]$ and **shorten** the code $\mathcal{C}$ over $\{i\} \cup \mathcal{R}_i$, of size $\le r + 1$.
Then $\mathcal{C}' := \mathrm{Short}(\mathcal{C}, \mathcal{R})$ is a code of

- length $n' \ge n - (r + 1)$;
- dimension $k' \ge k - r$ since $\dim \mathcal{C}_{|\mathcal{R}_i} = \dim \mathcal{C}_{|\mathcal{R}_i \cup \{i\}}$;
- minimum distance $d' \ge d$ by inclusion of codes;
- locality $r' \le r$.

Step 2: iterate this shortening process as much as possible and apply the classical Singleton bound ($d \le n - k + 1$) on the shortened code.

Optimal codes?

Prop. Let $\mathcal{C}$ be an $[n, k, d]_q$-LRC with locality r. Then, on a :

$$d \leq n - k + 2 - \left\lceil \frac{k}{r} \right\rceil .$$

Remarks.

1. Also true for non-linear codes.
2. For $r = k$, we retrieve the classical Singleton bound (hence MDS codes reach this bound)

Optimal codes?

Prop. Let $\mathcal{C}$ be an $[n, k, d]_q$-LRC with locality r. Then, on a :

$$d \leq n - k + 2 - \left\lceil \frac{k}{r} \right\rceil .$$

Remarks.

1. Also true for non-linear codes.
2. For $r = k$, we retrieve the classical Singleton bound (hence MDS codes reach this bound)

Q. Do there exist optimal codes w.r.t. this "Singleton-like bound", for $k \neq r$?

A Family of Optimal Locally Recoverable Codes. I. Tamo, A. Barg. IEEE TIT. **2014**.

Idea. In **a well-chosen Reed–Solomon code** (large minimum distance), select a subspace of polynomials with local recovery properties.

Tamo–Barg codes: an example

A Family of Optimal Locally Recoverable Codes. I. Tamo, A. Barg. IEEE TIT. **2014**.

Idea. In **a well-chosen Reed–Solomon code** (large minimum distance), select a subspace of polynomials with local recovery properties.

Example. Consider subsets $\mathcal{X}_1 = \{1, 3, 9\}$, $\mathcal{X}_2 = \{2, 6, 5\}$, $\mathcal{X}_3 = \{4, 12, 10\}$ of $\mathbb{F}_q$ with $q = 13$. We have

$$\forall x_1 \in \mathcal{X}_1,\ x_1^3 = 1; \qquad \forall x_2 \in \mathcal{X}_2,\ x_2^3 = 8; \qquad \forall x_3 \in \mathcal{X}_3,\ x_3^3 = 12.$$

A Family of Optimal Locally Recoverable Codes. I. Tamo, A. Barg. IEEE TIT. **2014**.

Idea. In **a well-chosen Reed–Solomon code** (large minimum distance), select a subspace of polynomials with local recovery properties.

Example. Consider subsets $\mathcal{X}_1 = \{1,3,9\}$, $\mathcal{X}_2 = \{2,6,5\}$, $\mathcal{X}_3 = \{4,12,10\}$ of $\mathbb{F}_q$ with $q = 13$. We have

$$\forall x_1 \in \mathcal{X}_1,\ x_1^3 = 1; \qquad \forall x_2 \in \mathcal{X}_2,\ x_2^3 = 8; \qquad \forall x_3 \in \mathcal{X}_3,\ x_3^3 = 12.$$

Evaluate over $\boldsymbol{x} = (1,3,9,2,6,5,4,12,10)$ the following polynomial fonctions:

$$\mathcal{F} := \langle 1, X, \quad X^3, X^4, \quad X^6, X^7 \rangle_{\mathbb{F}_q}\,.$$

Tamo–Barg codes: an example

A Family of Optimal Locally Recoverable Codes. I. Tamo, A. Barg. IEEE TIT. **2014**.

Idea. In **a well-chosen Reed–Solomon code** (large minimum distance), select a subspace of polynomials with local recovery properties.

Example. Consider subsets $\mathcal{X}_1 = \{1, 3, 9\}$, $\mathcal{X}_2 = \{2, 6, 5\}$, $\mathcal{X}_3 = \{4, 12, 10\}$ of $\mathbb{F}_q$ with $q = 13$. We have

$$\forall x_1 \in \mathcal{X}_1,\ x_1^3 = 1; \qquad \forall x_2 \in \mathcal{X}_2,\ x_2^3 = 8; \qquad \forall x_3 \in \mathcal{X}_3,\ x_3^3 = 12.$$

Evaluate over $\boldsymbol{x} = (1, 3, 9, 2, 6, 5, 4, 12, 10)$ the following polynomial fonctions:

$$\mathcal{F} := \langle 1, X, \quad X^3, X^4, \quad X^6, X^7 \rangle_{\mathbb{F}_q} .$$

If $f(X) = \sum_i a_i X^i \in \mathcal{F}$, then for all $x \in \mathcal{A}_2$ (for instance)

$$f(x) = (a_0 + 8 \cdot a_3 + 8^2 a_6) \cdot 1 + (a_1 + 8 \cdot a_4 + 8^2 a_7) \cdot x = u_2 + v_2 \cdot x$$

Tamo–Barg codes: an example

A Family of Optimal Locally Recoverable Codes. I. Tamo, A. Barg. IEEE TIT. **2014**.

Idea. In **a well-chosen Reed–Solomon code** (large minimum distance), select a subspace of polynomials with local recovery properties.

Example. Consider subsets $\mathcal{X}_1 = \{1,3,9\}$, $\mathcal{X}_2 = \{2,6,5\}$, $\mathcal{X}_3 = \{4,12,10\}$ of $\mathbb{F}_q$ with $q = 13$. We have

$$\forall x_1 \in \mathcal{X}_1,\ x_1^3 = 1; \qquad \forall x_2 \in \mathcal{X}_2,\ x_2^3 = 8; \qquad \forall x_3 \in \mathcal{X}_3,\ x_3^3 = 12.$$

Evaluate over $\boldsymbol{x} = (1,3,9,2,6,5,4,12,10)$ the following polynomial fonctions:

$$\mathcal{F} := \langle 1, X, \quad X^3, X^4, \quad X^6, X^7 \rangle_{\mathbb{F}_q}.$$

If $f(X) = \sum_i a_i X^i \in \mathcal{F}$, then for all $x \in \mathcal{A}_2$ (for instance)

$$f(x) = (a_0 + 8 \cdot a_3 + 8^2 a_6) \cdot 1 + (a_1 + 8 \cdot a_4 + 8^2 a_7) \cdot x = u_2 + v_2 \cdot x$$

Consequence. If we know $f(x')$ and $f(x'')$ for both other evaluation points x' and x'' of $\mathcal{A}_2$, we can solve a linear system to get u_2 and v_2, and then $f(x)$.

Tamo–Barg codes: definition

A Family of Optimal Locally Recoverable Codes. I. Tamo, A. Barg. IEEE TIT. **2014**.

Tamo & Barg gave a **general construction** for any k, r, if q is large enough.

Tamo–Barg codes: definition

A Family of Optimal Locally Recoverable Codes. I. Tamo, A. Barg. IEEE TIT. **2014**.

Tamo & Barg gave a **general construction** for any k, r, if q is large enough.

Let $g \in \mathbb{F}_q[X]$ such that:

(i) $\deg(g) = r + 1$.

(ii) there exist pairwise distinct $y_1, \dots, y_s \in \mathbb{F}_q$ such that $\mathcal{X}_j := g^{-1}(\{y_j\})$ has size $r + 1$.

Tamo–Barg codes: definition

A Family of Optimal Locally Recoverable Codes. I. Tamo, A. Barg. IEEE TIT. **2014**.

Tamo & Barg gave a **general construction** for any k, r, if q is large enough.

Let $g \in \mathbb{F}_q[X]$ such that:

(i) $\deg(g) = r + 1$.

(ii) there exist pairwise distinct $y_1, \dots, y_s \in \mathbb{F}_q$ such that $\mathcal{X}_j := g^{-1}(\{y_j\})$ has size $r + 1$.

Define the evaluation vector

$$\boldsymbol{x} = (\underbrace{x_{1,1}, \dots, x_{1,r+1}}_{\mathcal{X}_1}, x_{2,1}, \dots, \dots, x_{s,1}, \dots, x_{s,r+1}) \in \mathbb{F}_q^{s(r+1)}$$

Tamo–Barg codes: definition

A Family of Optimal Locally Recoverable Codes. I. Tamo, A. Barg. IEEE TIT. **2014**.

Tamo & Barg gave a **general construction** for any k, r, if q is large enough.

Let $g \in \mathbb{F}_q[X]$ such that:

(i) $\deg(g) = r + 1$.

(ii) there exist pairwise distinct $y_1, \dots, y_s \in \mathbb{F}_q$ such that $\mathcal{X}_j := g^{-1}(\{y_j\})$ has size $r + 1$.

Define the evaluation vector

$$\boldsymbol{x} = (\underbrace{x_{1,1}, \dots, x_{1,r+1}}_{\mathcal{X}_1}, x_{2,1}, \dots, \dots, x_{s,1}, \dots, x_{s,r+1}) \in \mathbb{F}_q^{s(r+1)}$$

Define, for $0 \le \ell \le s$,

$$\mathcal{F}_\ell := \langle X^i g(X)^j \mid 0 \le i \le r-1, 0 \le j \le \ell - 1 \rangle_{\mathbb{F}_q} \quad \subseteq \mathbb{F}_q[X]_{<(r+1)\ell-1}\,.$$

Tamo–Barg codes: definition

A Family of Optimal Locally Recoverable Codes. I. Tamo, A. Barg. IEEE TIT. **2014**.

Tamo & Barg gave a **general construction** for any k, r, if q is large enough.

Let $g \in \mathbb{F}_q[X]$ such that:

(i) $\deg(g) = r + 1$.

(ii) there exist pairwise distinct $y_1, \dots, y_s \in \mathbb{F}_q$ such that $\mathcal{X}_j := g^{-1}(\{y_j\})$ has size $r + 1$.

Define the evaluation vector

$$\boldsymbol{x} = (\underbrace{x_{1,1}, \dots, x_{1,r+1}}_{\mathcal{X}_1}, x_{2,1}, \dots, \dots, x_{s,1}, \dots, x_{s,r+1}) \ \in \mathbb{F}_q^{s(r+1)}$$

Define, for $0 \le \ell \le s$,

$$\mathcal{F}_\ell := \langle X^i g(X)^j \mid 0 \le i \le r-1, 0 \le j \le \ell - 1 \rangle_{\mathbb{F}_q} \quad \subseteq \mathbb{F}_q[X]_{<(r+1)\ell-1} .$$

Def. The corresponding Tamo–Barg code is:

$$\mathrm{TB}_{r,s,\ell}(\boldsymbol{x}) = \{\mathrm{ev}_{\boldsymbol{x}}(f) \mid f \in \mathcal{F}_\ell\} \subseteq \mathbb{F}_q^{s(r+1)}$$

Tamo–Barg codes: optimality

Thm. The Tamo–Barg code $\mathrm{TB}_{r,s,\ell}(\boldsymbol{x}) = \{\mathrm{ev}_{\boldsymbol{x}}(f) \mid f \in \mathcal{F}_\ell\}$ with $\mathcal{F}_\ell := \langle X^i g(X)^j \rangle_{0 \le i \le r-1, 0 \le j \le \ell-1}$ has the following parameters:

1. length $n = (r+1)s$;
2. dimension $k = r\ell$;
3. minimum distance $d = (r+1)(s-\ell) + 2$
4. locality r.

Thm. The Tamo–Barg code $\mathrm{TB}_{r,s,\ell}(\boldsymbol{x}) = \{\mathrm{ev}_{\boldsymbol{x}}(f) \mid f \in \mathcal{F}_\ell\}$ with $\mathcal{F}_\ell := \langle X^i g(X)^j \rangle_{0 \le i \le r-1, 0 \le j \le \ell-1}$ has the following parameters:

1. length $n = (r+1)s$;
2. dimension $k = r\ell$;
3. minimum distance $d = (r+1)(s-\ell)+2$
4. locality r.

Proof.

1.2.3. as a subcode of a Reed–Solomon code of length $(r+1)s$ and dimension $(r+1)\ell - 1$

Thm. The Tamo–Barg code $\mathrm{TB}_{r,s,\ell}(\boldsymbol{x}) = \{\mathrm{ev}_{\boldsymbol{x}}(f) \mid f \in \mathcal{F}_\ell\}$ with $\mathcal{F}_\ell := \langle X^i g(X)^j \rangle_{0\le i\le r-1, 0\le j\le \ell-1}$ has the following parameters:

1. length $n = (r+1)s$;
2. dimension $k = r\ell$;
3. minimum distance $d = (r+1)(s-\ell)+2$
4. locality r.

Proof.

1.2.3. as a subcode of a Reed–Solomon code of length $(r+1)s$ and dimension $(r+1)\ell - 1$

4. by design, as in the example

Thm. The Tamo–Barg code $\mathrm{TB}_{r,s,\ell}(\boldsymbol{x}) = \{\mathrm{ev}_{\boldsymbol{x}}(f) \mid f \in \mathcal{F}_\ell\}$ with $\mathcal{F}_\ell := \langle X^i g(X)^j \rangle_{0 \le i \le r-1, 0 \le j \le \ell-1}$ has the following parameters:

1. length $n = (r+1)s$;
2. dimension $k = r\ell$;
3. minimum distance $d = (r+1)(s-\ell)+2$
4. locality r.

Proof.

1\. 2. 3. as a subcode of a Reed–Solomon code of length $(r+1)s$ and dimension $(r+1)\ell - 1$

4\. by design, as in the example

Corollary. Tamo–Barg codes are optimal w.r.t. the Singleton-like bound:

$$d = n - k + 2 - \left\lceil \frac{k}{r} \right\rceil .$$

There are explicit constructions of polynomials $g(x)$, for instance:

Explicit Tamo–Barg codes

There are explicit constructions of polynomials $g(x)$, for instance:

1. Let H be a **multiplicative** subgroup of $\mathbb{F}_q^\times$ of order $r+1$.
   The polynomial
   $$g(x) = \prod_{h \in H}(x-h) = x^{r+1} - 1$$
   is constant over each multiplicative coset $\mathcal{R}_i := a_i H$ of H.

There are explicit constructions of polynomials $g(x)$, for instance:

1. Let H be a **multiplicative** subgroup of $\mathbb{F}_q^\times$ of order $r+1$.
   The polynomial
   $$g(x) = \prod_{h\in H}(x-h) = x^{r+1} - 1$$
   is constant over each multiplicative coset $\mathcal{R}_i := a_i H$ of H.

2. Let H' be an **additive** subgroup of $\mathbb{F}_q$.
   The polynomial
   $$g(x) = \prod_{h'\in H'}(x-h')$$
   is also constant over each multiplicative coset $\mathcal{R}_i := b_i + H'$ of H'.

There are explicit constructions of polynomials $g(x)$, for instance:

1. Let H be a **multiplicative** subgroup of $\mathbb{F}_q^\times$ of order $r+1$.
   The polynomial
   $$g(x) = \prod_{h \in H}(x - h) = x^{r+1} - 1$$
   is constant over each multiplicative coset $\mathcal{R}_i := a_i H$ of H.

2. Let H' be an **additive** subgroup of $\mathbb{F}_q$.
   The polynomial
   $$g(x) = \prod_{h' \in H'}(x - h')$$
   is also constant over each multiplicative coset $\mathcal{R}_i := b_i + H'$ of H'.

Question. Good constructions of LRCs for $q < n$?

We can **generalize** this idea with AG codes.

Towards a generalization of Tamo–Barg codes

We can **generalize** this idea with AG codes.

Let $\mathcal{X}$ and $\mathcal{Y}$ be two smooth projective absolutely irreductible curves over $\mathbb{F}_q$.

(before: $\mathcal{X} = \mathcal{Y} = \mathbb{P}^1$).

We can **generalize** this idea with AG codes.

Let $\mathcal{X}$ and $\mathcal{Y}$ be two smooth projective absolutely irreductible curves over $\mathbb{F}_q$.

(before: $\mathcal{X} = \mathcal{Y} = \mathbb{P}^1$).

Let $g : \mathcal{X} \to \mathcal{Y}$ be a rational map. Assume $\mathbb{F}_q(\mathcal{X}) = \mathbb{F}_q(\mathcal{Y})(x)$ is separable, with

$$x^{r+1} + \sum_{i=0}^{r} b_i x^i = 0, \quad b_i \in \mathbb{F}_q(\mathcal{Y})$$

(before: $g \in \mathbb{F}_q[X]$ of degree $r+1$)

We can **generalize** this idea with AG codes.

Let $\mathcal{X}$ and $\mathcal{Y}$ be two smooth projective absolutely irreductible curves over $\mathbb{F}_q$.

(before: $\mathcal{X} = \mathcal{Y} = \mathbb{P}^1$).

Let $g : \mathcal{X} \to \mathcal{Y}$ be a rational map. Assume $\mathbb{F}_q(\mathcal{X}) = \mathbb{F}_q(\mathcal{Y})(x)$ is separable, with

$$x^{r+1} + \sum_{i=0}^{r} b_i x^i = 0, \quad b_i \in \mathbb{F}_q(\mathcal{Y})$$

(before: $g \in \mathbb{F}_q[X]$ of degree $r+1$)

Let $\mathcal{P} = \{P_1, \dots, P_s\} \subseteq \mathcal{Y}(\mathbb{F}_q)$ and D be an effective divisor of degree ℓ, with $\mathrm{supp}(D) \cap \mathcal{P} = \varnothing$.

(before: $P_j = y_j$ et $D = \ell P_\infty$)

We can **generalize** this idea with AG codes.

Let $\mathcal{X}$ and $\mathcal{Y}$ be two smooth projective absolutely irreductible curves over $\mathbb{F}_q$.

(before: $\mathcal{X} = \mathcal{Y} = \mathbb{P}^1$).

Let $g : \mathcal{X} \to \mathcal{Y}$ be a rational map. Assume $\mathbb{F}_q(\mathcal{X}) = \mathbb{F}_q(\mathcal{Y})(x)$ is separable, with

$$x^{r+1} + \sum_{i=0}^{r} b_i x^i = 0, \quad b_i \in \mathbb{F}_q(\mathcal{Y})$$

(before: $g \in \mathbb{F}_q[X]$ of degree $r+1$)

Let $\mathcal{P} = \{P_1, \dots, P_s\} \subseteq \mathcal{Y}(\mathbb{F}_q)$ and D be an effective divisor of degree ℓ, with $\mathrm{supp}(D) \cap \mathcal{P} = \varnothing$.

(before: $P_j = y_j$ et $D = \ell P_\infty$)

Let $\mathcal{X}_j := \{x_{i,j}\}_{0 \le i \le r} \subseteq \mathcal{X}(\mathbb{F}_q)$ for $1 \le j \le s$ be such that $g(x_{i,j}) = P_j$.

(similar as before)

We can **generalize** this idea with AG codes.

Let $\mathcal{X}$ and $\mathcal{Y}$ be two smooth projective absolutely irreductible curves over $\mathbb{F}_q$.

(before: $\mathcal{X} = \mathcal{Y} = \mathbb{P}^1$).

Let $g : \mathcal{X} \to \mathcal{Y}$ be a rational map. Assume $\mathbb{F}_q(\mathcal{X}) = \mathbb{F}_q(\mathcal{Y})(x)$ is separable, with

$$x^{r+1} + \sum_{i=0}^{r} b_i x^i = 0, \quad b_i \in \mathbb{F}_q(\mathcal{Y})$$

(before: $g \in \mathbb{F}_q[X]$ of degree $r+1$)

Let $\mathcal{P} = \{P_1, \dots, P_s\} \subseteq \mathcal{Y}(\mathbb{F}_q)$ and D be an effective divisor of degree ℓ, with $\mathrm{supp}(D) \cap \mathcal{P} = \varnothing$.

(before: $P_j = y_j$ et $D = \ell P_\infty$)

Let $\mathcal{X}_j := \{x_{i,j}\}_{0 \le i \le r} \subseteq \mathcal{X}(\mathbb{F}_q)$ for $1 \le j \le s$ be such that $g(x_{i,j}) = P_j$.

(similar as before)

Finally, assume that $b_i \in L(n_i D)$ for some $n_i \ge 1$, and denote by $f_1, \dots, f_m \in \mathbb{F}_q(\mathcal{Y})$ a basis of $L(D)$. Then define:

$$\mathcal{F} := \langle x^i f_j \rangle_{0 \le i \le r-1, 1 \le j \le m} \subset \mathbb{F}_q(\mathcal{X})$$

(before: $\mathcal{F} = \{x^i g(x)^j \mid \dots\}$)

Locally Recoverable Codes on Algebraic Curves. A. Barg, I. Tamo, S. Vladut. IEEE TIT. **2017**.

Def. According to previous notation, we define a Barg–Tamo–Vladut code as:

$$\mathrm{BTV} = \{\mathrm{ev}_{\cup \mathcal{X}_j}(f) \mid f \in \mathcal{F}\} \subseteq \mathbb{F}_q^{s(r+1)}$$

Locally Recoverable Codes on Algebraic Curves. A. Barg, I. Tamo, S. Vladut. IEEE TIT. **2017**.

Def. According to previous notation, we define a Barg–Tamo–Vladut code as:

$$\mathrm{BTV} = \{\mathrm{ev}_{\cup \mathcal{X}_j}(f) \mid f \in \mathcal{F}\} \subseteq \mathbb{F}_q^{s(r+1)}$$

Thm. Code BTV has the following parameters:

- length $n = (r+1)s$,
- dimension $k = rm \geq t(\ell - g_{\mathcal{Y}} + 1)$,
- minimum distance $d \geq n - \ell(r+1) - (r-1)h$,
- locality r,

where h is the degree of $x \in \mathbb{F}_q(\mathcal{X})$.

Barg–Tamo–Vladut's construction

Locally Recoverable Codes on Algebraic Curves. A. Barg, I. Tamo, S. Vladut. IEEE TIT. **2017**.

Def. According to previous notation, we define a Barg–Tamo–Vladut code as:

$$\mathrm{BTV} = \{\mathrm{ev}_{\cup \mathcal{X}_j}(f) \mid f \in \mathcal{F}\} \subseteq \mathbb{F}_q^{s(r+1)}$$

Thm. Code BTV has the following parameters:

- length $n = (r+1)s$,
- dimension $k = rm \geq t(\ell - g_{\mathcal{Y}} + 1)$,
- minimum distance $d \geq n - \ell(r+1) - (r-1)h$,
- locality r,

where h is the degree of $x \in \mathbb{F}_q(\mathcal{X})$.

Consequence. Explicit asymptotic families of codes exceeding a Gilbert–Varshamov-like bound for LRCs.

Other constructions

- "Pyramid codes":
  Pyramid codes: Flexible schemes to trade space for access efficiency in reliable data storage systems. C. Huang, M. Chen, and J. Li. Proc. IEEE ISNCA. **2007**.

Other constructions

- "Pyramid codes":
  Pyramid codes: Flexible schemes to trade space for access efficiency in reliable data storage systems. C. Huang, M. Chen, and J. Li. Proc. IEEE ISNCA. **2007**.
- Cyclic LRCs:
  Cyclic LRC codes, binary LRC codes, and upper bounds on the distance of cyclic codes. I. Tamo, A. Barg, S. Goparaju, and A. R. Calderbank. Int. J. Inf. Coding Theory. **2016**.

Other constructions

- "Pyramid codes":
  Pyramid codes: Flexible schemes to trade space for access efficiency in reliable data storage systems. C. Huang, M. Chen, and J. Li. Proc. IEEE ISNCA. **2007**.
- Cyclic LRCs:
  Cyclic LRC codes, binary LRC codes, and upper bounds on the distance of cyclic codes. I. Tamo, A. Barg, S. Goparaju, and A. R. Calderbank. Int. J. Inf. Coding Theory. **2016**.
- LRCs using algebraic geometry:
  Construction of optimal locally repairable codes via automorphism groups of rational function fields. L. Jin, L. Ma, and C. Xing. IEEE TIT. **2019**.
  Locally recoverable codes on surfaces. C. Salgado, A. Várilly-Alvarado, and J. F. Voloch. IEEE TIT. **2021**.

Other constructions

- "Pyramid codes":
  Pyramid codes: Flexible schemes to trade space for access efficiency in reliable data storage systems. C. Huang, M. Chen, and J. Li. Proc. IEEE ISNCA. **2007**.
- Cyclic LRCs:
  Cyclic LRC codes, binary LRC codes, and upper bounds on the distance of cyclic codes. I. Tamo, A. Barg, S. Goparaju, and A. R. Calderbank. Int. J. Inf. Coding Theory. **2016**.
- LRCs using algebraic geometry:
  Construction of optimal locally repairable codes via automorphism groups of rational function fields. L. Jin, L. Ma, and C. Xing. IEEE TIT. **2019**.
  Locally recoverable codes on surfaces. C. Salgado, A. Várilly-Alvarado, and J. F. Voloch. IEEE TIT. **2021**.
- Via sum-rank codes, linearized and subspace polynomials.
  Universal and dynamic locally repairable codes with maximal recoverability via sum-rank codes. U. Martínez-Penas and F. R. Kschischang. IEEE TIT. **2019**.
  Explicit maximally recoverable codes with locality. P. Gopalan, C. Huang, B. Jenkins, and S. Yekhanin. IEEE TIT. **2014**.

In pratice

LRCs have been tested and implemented:

- in *Windows Azure*: with an $(n = 16, k = 12, r = 6)$ and then an $(n = 18, k = 14, r = 7)$ LRC;
- in *Facebook storage system*, with an $(n = 16, k = 10, r = 5)$ LRC;
- in *Ceph* storage cluster

First extension: local recovery with correction

LRCs usually allow to recover 1 erased symbol (by accessing $\leq r$ other symbols).

Problem. Recover many erasures? Correct errors instead of erasures?

First extension: local recovery with correction

LRCs usually allow to recover 1 erased symbol (by accessing $\leq r$ other symbols).

Problem. Recover many erasures? Correct errors instead of erasures?

First solution: improve the "local distance" to $\delta > 2$.

LRCs usually allow to recover 1 erased symbol (by accessing $\leq r$ other symbols).

Problem. Recover many erasures? Correct errors instead of erasures?

First solution: improve the "local distance" to $\delta > 2$.

Def. A code $\mathcal{C} \subseteq \mathbb{F}_q^n$ is (r, δ)–locally recoverable if, for all $1 \leq i \leq n$, there exists a subset $\mathcal{R}_i \subset [1, n]$, such that $i \notin \mathcal{R}_i$, $|\mathcal{R}_i| \leq r$, and

- $\dim \mathcal{C}_{|\mathcal{R}_i} = \dim \mathcal{C}_{|\mathcal{R}_i \cup \{i\}}$,
- $\mathrm{d}_{\min}(\mathcal{C}_{|\mathcal{R}_i \cup \{i\}}) \geq \delta$.

A Singleton-like bound for (r, δ)-LRCs:

Prop. If $\mathcal{C}$ is an (r, δ)-LR code with parameters $[n, k, d]_q$, then

$$d \leq n - k + 1 - (\delta - 1)\left(\left\lceil \frac{k}{r} \right\rceil - 1\right) .$$

A Singleton-like bound for (r, δ)-LRCs:

Prop. If $\mathcal{C}$ is an (r, δ)-LR code with parameters $[n, k, d]_q$, then

$$d \leq n - k + 1 - (\delta - 1)\left(\left\lceil \frac{k}{r} \right\rceil - 1\right) .$$

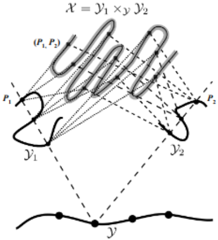
Optimal construction: generalization of Tamo–Barg codes.
→ modify the polynomial space into:

$$\mathcal{F} := \{x^i g(x)^j \mid 0 \leq i \leq r + 1 - \delta, 0 \leq j \leq \ell\} .$$

Second solution: many helper sets for the same coordinate.

Repair locality with multiple erasure tolerance. A. Wang and Z. Zhang. IEEE TIT. **2014**.

Def. A code $\mathcal{C}$ has locality r and **availability** t if, for all $i \in [1, n]$, the coordinate i admits t pairwise disjoint helper sets.



Remark. Nice geometric constructions with fiber products of curves.

Illustration and reference:

Locally recoverable codes with availability $t \geq 2$ from fiber products of curves. K. Haymaker, B. Malmskog, G. Matthews. Adv. Math. Comm.. **2018**.

Outline

1. Motivation

2. Local recovery

3. Regeneration

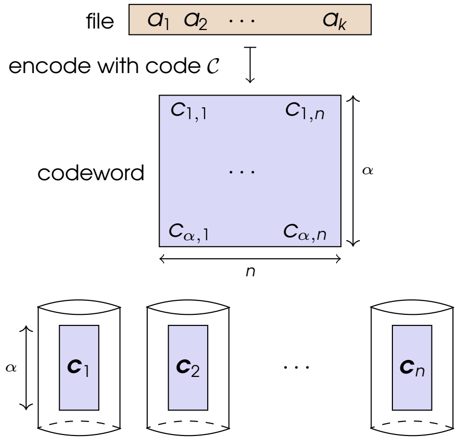
4. Local correction

Same context as before, but codes are **array codes**:

$$\mathcal{C} : \mathbb{F}_q^k \to \mathbb{F}_q^{\alpha \times n}$$

Encode the file $\boldsymbol{a} = (a_1, \dots, a_k)$
into a (matrix) codeword $\boldsymbol{C} = (\boldsymbol{c}_1, \dots, \boldsymbol{c}_n) \in \mathcal{C}$.

Server $\mathcal{S}_i$ stores α symbols $\boldsymbol{c}_i \in \mathbb{F}_q^\alpha$.



Def. An $(n, s, t, \alpha, \beta, k)$-**regenerating code** is a code $\mathcal{C} \subseteq \mathbb{F}_q^{\alpha \times n}$ of dimension k such that:

- each codeword $\boldsymbol{C} \in \mathcal{C}$ is fully determined by **any** s-subset of its columns *(data reconstruction property)*,
- each column $\boldsymbol{c}_i$ of $\boldsymbol{C}$ can be "regenerated", by downloading at most $\beta \leq \alpha$ symbols from **any** t-subset of columns *(regeneration property)*.

(rem: unusual notation; more usually $(n, k, d, \alpha, \beta, M)$-regenerating code)

Note. The regeneration condition enforces $t\beta \geq \alpha$.

Illustration for regeneration

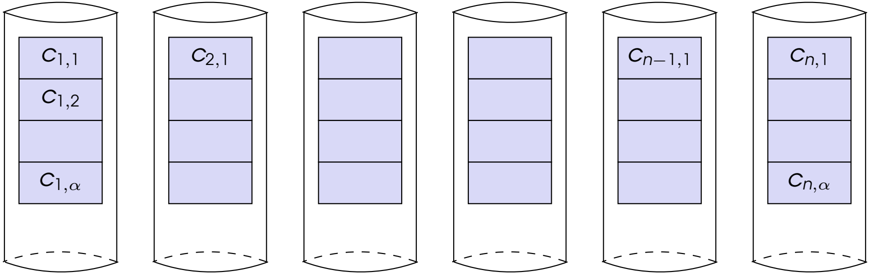


Illustration for regeneration

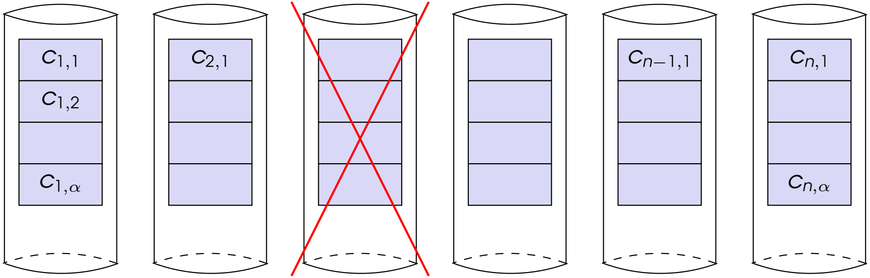
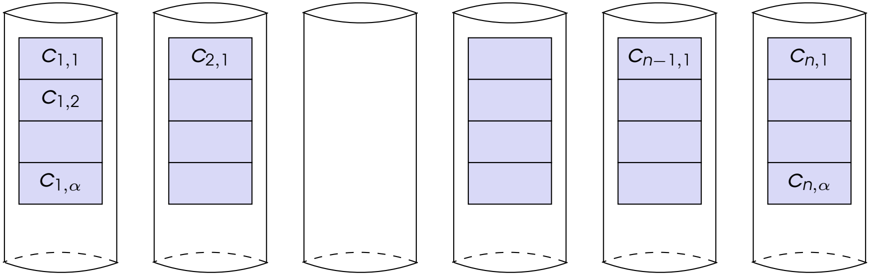
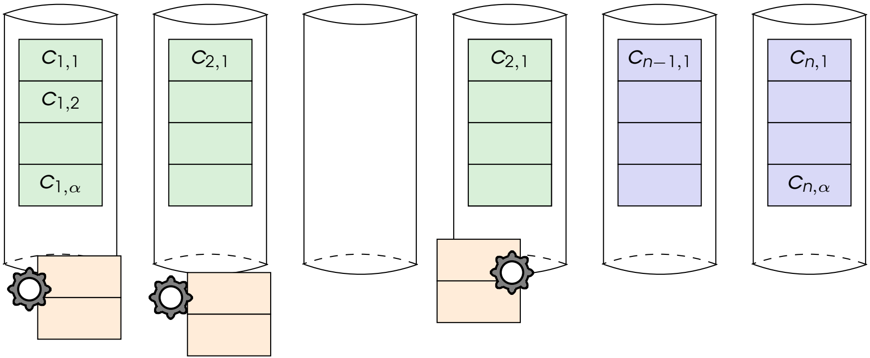


Illustration for regeneration

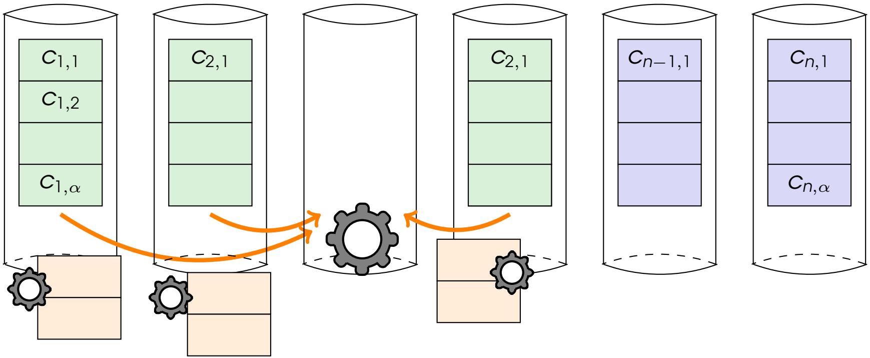




$t = 3$ servers are contacted for regeneration

$\beta = 2$ symbols are sent by each server for regeneration

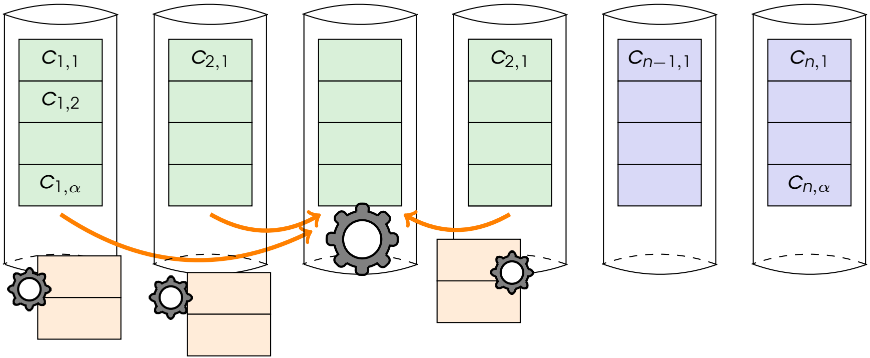
Illustration for regeneration



$t = 3$ servers are contacted for regeneration

$\beta = 2$ symbols are sent by each server for regeneration

Illustration for regeneration



$t = 3$ servers are contacted for regeneration

$\beta = 2$ symbols are sent by each server for regeneration

Cut-set bound

Main bound (cut-set bound): Any $(n, s, t, \alpha, \beta, k)$-regenerating code satisfies

$$k \leq \sum_{i=0}^{s-1} \min(\alpha, (t-i)\beta)\,.$$

Rem. For fixed s and t, there are many optimal points (depending on α and β).

For instance, if $t\beta = \alpha$, the bound defines a **minimum-bandwidth repair (MBR) point**:

$$k = \left(st - \frac{s(s-1)}{2}\right)\beta\,.$$

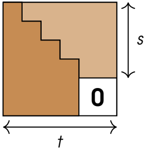
Deterministic regenerating codes for distributed storage. Y. Wu, A. G. Dimakis, and K. Ramchandran. Allerton conf. on control, computing, and communication. **2007**.

Product-matrix MBR codes

Optimal Exact-Regenerating Codes for Distributed Storage at the MSR and MBR Points via a Product-Matrix Construction. Rashmi, Shah, Kumar. IEEE-TIT. **2011**.

For the example, we set $\beta = 1$, hence $\alpha = t$.
Set also $k = \left(st - \frac{s(s-1)}{2}\right)\beta = st - \frac{s(s-1)}{2}$.

📄 *Optimal Exact-Regenerating Codes for Distributed Storage at the MSR and MBR Points via a Product-Matrix Construction*. Rashmi, Shah, Kumar. IEEE-TIT. **2011**.

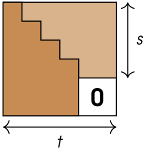


For the example, we set $\beta = 1$, hence $\alpha = t$.
Set also $k = \left(st - \frac{s(s-1)}{2}\right)\beta = st - \frac{s(s-1)}{2}$.

1. **Message symbols** are first arranged in a $(t \times t)$ **symmetric** matrix of the form

$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{S} & \boldsymbol{T}^\top \\ \boldsymbol{T} & \boldsymbol{0} \end{pmatrix}$$

Optimal Exact-Regenerating Codes for Distributed Storage at the MSR and MBR Points via a Product-Matrix Construction. Rashmi, Shah, Kumar. IEEE-TIT. **2011**.



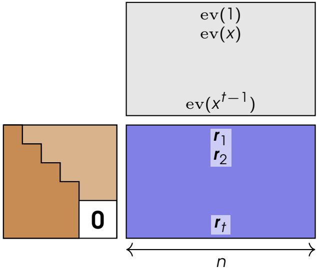
For the example, we set $\beta = 1$, hence $\alpha = t$.
Set also $k = \left(st - \frac{s(s-1)}{2}\right)\beta = st - \frac{s(s-1)}{2}$.

1. **Message symbols** are first arranged in a $(t \times t)$ **symmetric** matrix of the form

$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{S} & \boldsymbol{T}^\top \\ \boldsymbol{T} & \mathbf{0} \end{pmatrix}$$

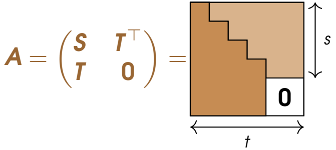
2. Let $\boldsymbol{G}$ be a $(t \times n)$ generator matrix for a Reed-Solomon code $\mathrm{RS}_q(t, n)$, echelonized in degree. **Codewords** are then:

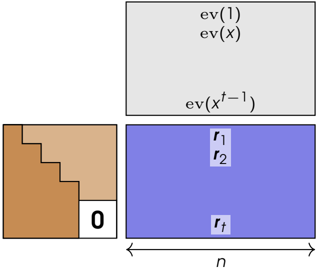
$$\boldsymbol{C} = \boldsymbol{A}\boldsymbol{G} \in \mathbb{F}_q^{t \times n}.$$



MBR PM codes: data reconstruction

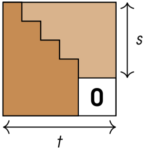
Goal: recover $A = \begin{pmatrix} S & T^\top \\ T & 0 \end{pmatrix}$ from the codeword C, with access to only s columns.

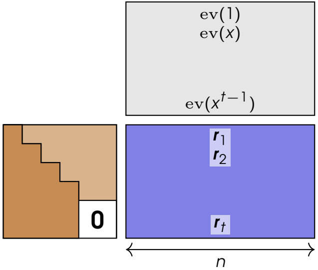




Goal: recover $A = \begin{pmatrix} S & T^\top \\ T & 0 \end{pmatrix}$ from the codeword C, with access to only s columns.

Step 1: find T.

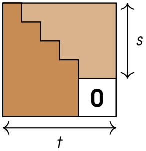
$A = \begin{pmatrix} S & T^\top \\ T & 0 \end{pmatrix} =$ 

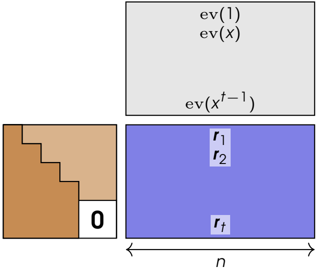


Goal: recover $\boldsymbol{A} = \begin{pmatrix} \boldsymbol{S} & \boldsymbol{T}^\top \\ \boldsymbol{T} & \boldsymbol{0} \end{pmatrix}$ from the codeword $\boldsymbol{C}$, with access to only s columns.

Step 1: find $\boldsymbol{T}$. Last $t - s$ rows $\boldsymbol{r}_{s+1}, \dots, \boldsymbol{r}_t$ of $\boldsymbol{C}$ are:

$$\boldsymbol{r}_i = (T_{1,i}, \dots, T_{s,i}, 0, \dots, 0) \cdot \boldsymbol{G} \quad \in \mathrm{RS}_q(s, n)\,.$$

$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{S} & \boldsymbol{T}^\top \\ \boldsymbol{T} & \boldsymbol{0} \end{pmatrix} =$ 



MBR PM codes: data reconstruction

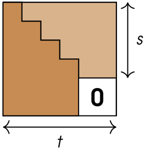
Goal: recover $A = \begin{pmatrix} S & T^\top \\ T & 0 \end{pmatrix}$ from the codeword C, with access to only s columns.

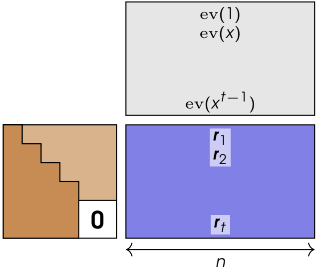
<ins>Step 1:</ins> find T. Last $t - s$ rows $r_{s+1}, \dots, r_t$ of C are:

$$r_i = (T_{1,i}, \dots, T_{s,i}, 0, \dots, 0) \cdot G \quad \in \mathrm{RS}_q(s, n)\,.$$

$\rightarrow$ we can recover $T_{1,i}, \dots, T_{s,i}$ with any s symbols of r_i.

$\rightarrow$ we can recover T with any s columns of C.

$A = \begin{pmatrix} S & T^\top \\ T & 0 \end{pmatrix} =$ 



MBR PM codes: data reconstruction

Goal: recover $\boldsymbol{A} = \begin{pmatrix} \boldsymbol{S} & \boldsymbol{T}^\top \\ \boldsymbol{T} & \boldsymbol{0} \end{pmatrix}$ from the codeword $\boldsymbol{C}$, with access to only s columns.

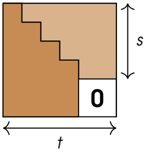
Step 1: find $\boldsymbol{T}$. Last $t - s$ rows $\boldsymbol{r}_{s+1}, \dots, \boldsymbol{r}_t$ of $\boldsymbol{C}$ are:

$$\boldsymbol{r}_i = (T_{1,i}, \dots, T_{s,i}, 0, \dots, 0) \cdot \boldsymbol{G} \quad \in \mathrm{RS}_q(s, n) \,.$$

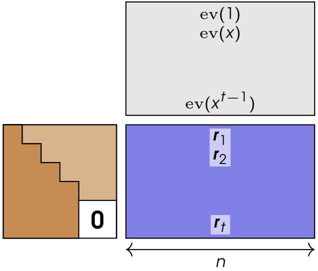
$\rightarrow$ we can recover $T_{1,i}, \dots, T_{s,i}$ with any s symbols of $\boldsymbol{r}_i$.

$\rightarrow$ we can recover $\boldsymbol{T}$ with any s columns of $\boldsymbol{C}$.

$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{S} & \boldsymbol{T}^\top \\ \boldsymbol{T} & \boldsymbol{0} \end{pmatrix} =$$



Step 2: find $\boldsymbol{S}$.

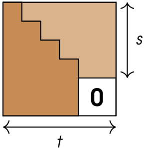


MBR PM codes: data reconstruction

Goal: recover $\boldsymbol{A} = \begin{pmatrix} \boldsymbol{S} & \boldsymbol{T}^\top \\ \boldsymbol{T} & \boldsymbol{0} \end{pmatrix}$ from the codeword $\boldsymbol{C}$, with access to only s columns.

Step 1: find $\boldsymbol{T}$. Last $t - s$ rows $\boldsymbol{r}_{s+1}, \dots, \boldsymbol{r}_t$ of $\boldsymbol{C}$ are:

$$\boldsymbol{r}_i = (T_{1,i}, \dots, T_{s,i}, 0, \dots, 0) \cdot \boldsymbol{G} \quad \in \mathrm{RS}_q(s, n)\,.$$

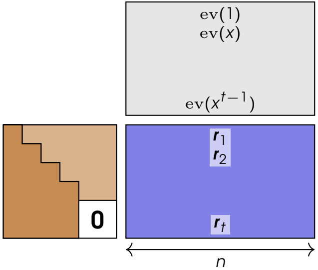
$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{S} & \boldsymbol{T}^\top \\ \boldsymbol{T} & \boldsymbol{0} \end{pmatrix} =$ 

$\rightarrow$ we can recover $T_{1,i}, \dots, T_{s,i}$ with any s symbols of $\boldsymbol{r}_i$.

$\rightarrow$ we can recover $\boldsymbol{T}$ with any s columns of $\boldsymbol{C}$.

Step 2: find $\boldsymbol{S}$. First s rows $\boldsymbol{r}_1, \dots, \boldsymbol{r}_s$ of $\boldsymbol{C}$ are:

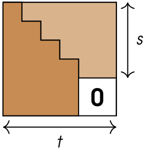
$$\boldsymbol{r}_i = (S_{1,i}, \dots, S_{s,i}, 0, \dots, 0) \cdot \boldsymbol{G} + \underbrace{(0, \dots, 0, T_{i,1}, \dots, T_{i,s}) \cdot \boldsymbol{G}}_{=\boldsymbol{z}_i \ \text{(known)}}$$



Goal: recover $\boldsymbol{A} = \begin{pmatrix} \boldsymbol{S} & \boldsymbol{T}^\top \\ \boldsymbol{T} & \boldsymbol{0} \end{pmatrix}$ from the codeword $\boldsymbol{C}$, with access to only s columns.

Step 1: find $\boldsymbol{T}$. Last $t - s$ rows $\boldsymbol{r}_{s+1}, \dots, \boldsymbol{r}_t$ of $\boldsymbol{C}$ are:

$$\boldsymbol{r}_i = (T_{1,i}, \dots, T_{s,i}, 0, \dots, 0) \cdot \boldsymbol{G} \quad \in \mathrm{RS}_q(s, n)\,.$$

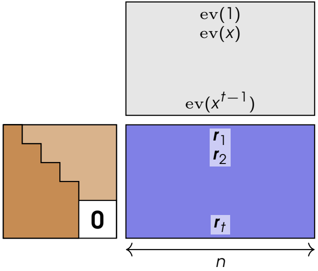
$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{S} & \boldsymbol{T}^\top \\ \boldsymbol{T} & \boldsymbol{0} \end{pmatrix} =$ 

- $\rightarrow$ we can recover $T_{1,i}, \dots, T_{s,i}$ with any s symbols of $\boldsymbol{r}_i$.
- $\rightarrow$ we can recover $\boldsymbol{T}$ with any s columns of $\boldsymbol{C}$.

Step 2: find $\boldsymbol{S}$. First s rows $\boldsymbol{r}_1, \dots, \boldsymbol{r}_s$ of $\boldsymbol{C}$ are:

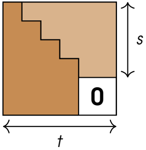
$$\boldsymbol{r}_i = (S_{1,i}, \dots, S_{s,i}, 0, \dots, 0) \cdot \boldsymbol{G} + \underbrace{(0, \dots, 0, T_{i,1}, \dots, T_{i,s}) \cdot \boldsymbol{G}}_{=\boldsymbol{z}_i \ \text{(known)}}$$

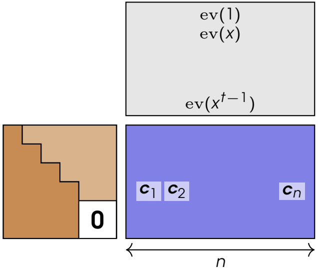
- $\rightarrow$ vector $\boldsymbol{r}_i - \boldsymbol{z}_i$ is an element $\mathrm{RS}_q(s, n)$.
- $\rightarrow$ we recover $S_{1,i}, \dots, S_{s,i}$, hence $\boldsymbol{S}$, as in Step 1, *i.e.* with any s columns of $\boldsymbol{C}$.



Goal: recover a column $\boldsymbol{c}_i$ of $\boldsymbol{C}$, by retrieving only $\beta = 1$ symbol from t servers.
Important: we allow servers to **compute** on data.

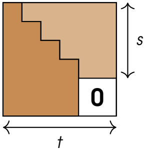
$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{S} & \boldsymbol{T}^\top \\ \boldsymbol{T} & \boldsymbol{0} \end{pmatrix} =$$

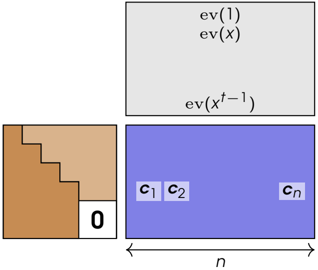




Goal: recover a column $\boldsymbol{c}_i$ of $\boldsymbol{C}$, by retrieving only $\beta = 1$ symbol from t servers.
Important: we allow servers to **compute** on data.
Consider:

- $J \subset [1, n]$ any subset of coordinates of size t
- $\boldsymbol{U} = \boldsymbol{G}_{|J}^{-1} \in \mathbb{F}_q^{t \times t}$
- $\boldsymbol{g}_j$ a column of $\boldsymbol{G}$, for $j \in J$.

$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{S} & \boldsymbol{T}^\top \\ \boldsymbol{T} & \boldsymbol{0} \end{pmatrix} =$ 



MBR PM codes: regeneration

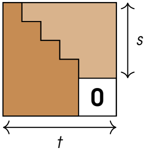
Goal: recover a column $\boldsymbol{c}_i$ of $\boldsymbol{C}$, by retrieving only $\beta = 1$ symbol from t servers.
Important: we allow servers to **compute** on data.
Consider:

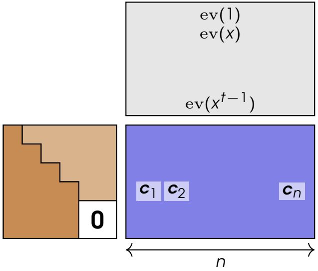
- $J \subset [1, n]$ any subset of coordinates of size t
- $\boldsymbol{U} = \boldsymbol{G}_{|J}^{-1} \in \mathbb{F}_q^{t \times t}$
- $\boldsymbol{g}_j$ a column of $\boldsymbol{G}$, for $j \in J$.

Since $\boldsymbol{A}$ is symmetric and $\boldsymbol{c}_j = \boldsymbol{A}\boldsymbol{g}_j$:

$$\langle \boldsymbol{g}_i, \boldsymbol{c}_j \rangle = \langle \boldsymbol{g}_i, \boldsymbol{A}\boldsymbol{g}_j \rangle = \langle \boldsymbol{A}^\top \boldsymbol{g}_i, \boldsymbol{g}_j \rangle = \langle \boldsymbol{A}\boldsymbol{g}_i, \boldsymbol{g}_j \rangle = \langle \boldsymbol{c}_i, \boldsymbol{g}_j \rangle$$

$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{S} & \boldsymbol{T}^\top \\ \boldsymbol{T} & \boldsymbol{0} \end{pmatrix} =$$



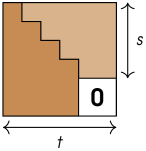


MBR PM codes: regeneration

Goal: recover a column $\boldsymbol{c}_i$ of $\boldsymbol{C}$, by retrieving only $\beta = 1$ symbol from t servers.
Important: we allow servers to **compute** on data.
Consider:

- $J \subset [1, n]$ any subset of coordinates of size t
- $\boldsymbol{U} = \boldsymbol{G}_{|J}^{-1} \in \mathbb{F}_q^{t \times t}$
- $\boldsymbol{g}_j$ a column of $\boldsymbol{G}$, for $j \in J$.

$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{S} & \boldsymbol{T}^\top \\ \boldsymbol{T} & \boldsymbol{0} \end{pmatrix} =$$

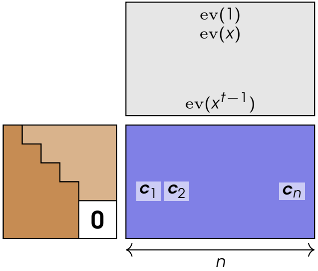


Since $\boldsymbol{A}$ is symmetric and $\boldsymbol{c}_j = \boldsymbol{A}\boldsymbol{g}_j$:

$$\langle \boldsymbol{g}_i, \boldsymbol{c}_j \rangle = \langle \boldsymbol{g}_i, \boldsymbol{A}\boldsymbol{g}_j \rangle = \langle \boldsymbol{A}^\top \boldsymbol{g}_i, \boldsymbol{g}_j \rangle = \langle \boldsymbol{A}\boldsymbol{g}_i, \boldsymbol{g}_j \rangle = \langle \boldsymbol{c}_i, \boldsymbol{g}_j \rangle$$

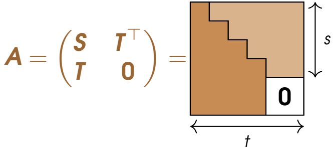
<u>Step 1:</u> for $j \in J$, server $\mathcal{S}_j$ computes $y_j = \langle \boldsymbol{g}_i, \boldsymbol{c}_j \rangle \in \mathbb{F}_q$. Then,

$$(y_1, \dots, y_t) =: \boldsymbol{y} = \boldsymbol{c}_i \boldsymbol{G}_{|J} \,.$$



Goal: recover a column $\boldsymbol{c}_i$ of $\boldsymbol{C}$, by retrieving only $\beta = 1$ symbol from t servers.
Important: we allow servers to **compute** on data.
Consider:

- $J \subset [1, n]$ any subset of coordinates of size t
- $\boldsymbol{U} = \boldsymbol{G}_{|J}^{-1} \in \mathbb{F}_q^{t\times t}$
- $\boldsymbol{g}_j$ a column of $\boldsymbol{G}$, for $j \in J$.



Since $\boldsymbol{A}$ is symmetric and $\boldsymbol{c}_j = \boldsymbol{A}\boldsymbol{g}_j$:

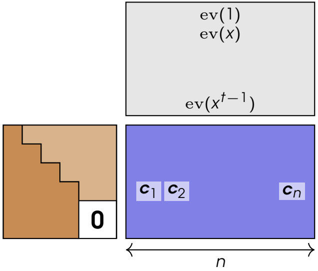
$$\langle \boldsymbol{g}_i, \boldsymbol{c}_j \rangle = \langle \boldsymbol{g}_i, \boldsymbol{A}\boldsymbol{g}_j \rangle = \langle \boldsymbol{A}^\top \boldsymbol{g}_i, \boldsymbol{g}_j \rangle = \langle \boldsymbol{A}\boldsymbol{g}_i, \boldsymbol{g}_j \rangle = \langle \boldsymbol{c}_i, \boldsymbol{g}_j \rangle$$

Step 1: for $j \in J$, server $\mathcal{S}_j$ computes $y_j = \langle \boldsymbol{g}_i, \boldsymbol{c}_j \rangle \in \mathbb{F}_q$. Then,

$$(y_1, \dots, y_t) =: \boldsymbol{y} = \boldsymbol{c}_i \boldsymbol{G}_{|J}\,.$$

Step 2: after collection of $\boldsymbol{y} = (y_1, \dots, y_t)$, compute and recover

$$\boldsymbol{y}\boldsymbol{U} = \boldsymbol{c}_i \boldsymbol{G}_{|J} \boldsymbol{U} = \boldsymbol{c}_i\,.$$



Other works

- Quite surprisingly, **repair of Reed–Solomon codes over extension fields** $\mathbb{F}_q/\mathbb{F}_p$, using the $\mathbb{F}_p$-vector space structure and trace-dual bases.
  Repairing Reed–Solomon codes. V. Guruswami and M. Wootters. IEEE TIT. **2017**.
- Sequential recovery.
  Codes with locality for two erasures. N. Prakash, V. Lalitha, S. B. Balaji, and P. V. Kumar. IEEE TIT. **2019**.
- Hierarchical locality
  Codes with hierarchical locality. B. Sasidharan, G. K. Agarwal, and P. V. Kumar. IEEE ISIT. **2015**.
- Maximally Recoverable Codes
  On the maximally recoverable property for multi-protection group codes. M. Chen, C. Huang, and J. Li. ISIT. **2007**.
- A survey:
  Codes for Distributed Storage. Ramkumar, Balaji, Sasidharan, Vajha, Krishnan, Kumar. Foundations and Trends in Communication and Information Theory. **2022**.

Perspectives

Locality (and regeneration?) for codes defined w.r.t. **other metrics**:

- in **rank-metric**, attempted in 2017 by Kadhe, El Rouayheb, Duursma & Sprintson in *Codes with Locality in the Rank and Subspace Metrics* (framework to be discussed);
- in **weighted metrics**, *e.g.* for codes over the integers.

Locality for **dynamic** coded databases.

Locality and codes for **peer-to-peer storage** systems, with a large amount of disconnected nodes.

Locality with specific **storage topologies**, with nodes adjacency given by a graph.

Locality with **service rate** constaints.

Locality for **coded distributed computing**.

Outline

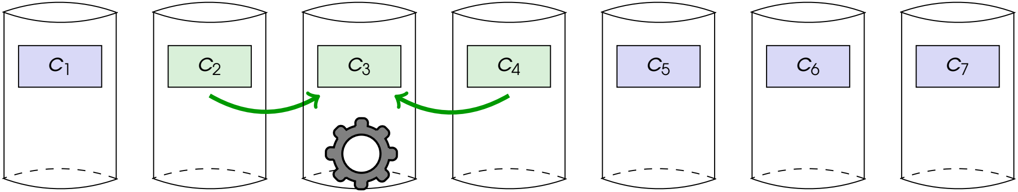
1. Motivation

2. Local recovery

3. Regeneration

4. Local correction

Illustration for locality



Reminder: availability (1)

Def. A code $\mathcal{C}$ has locality r and **availability** t if, for all $1 \leq i \leq n$, the coordinate i admits at least t pairwise disjoint helper sets.

Reminder: availability (1)

Def. A code $\mathcal{C}$ has locality r and **availability** t if, for all $1 \le i \le n$, the coordinate i admits at least t pairwise disjoint helper sets.

Tensor product of codes $\mathcal{A} \subseteq \mathbb{F}_q^m$ and $\mathcal{B} \in \mathbb{F}_q^n$:

$$\mathcal{A} \otimes \mathcal{B} = \left\{ \boldsymbol{c} \in \mathbb{F}_q^{m \times n} \;\middle|\; \begin{array}{l} \text{each column of } \boldsymbol{c} \text{ is in } \mathcal{A} \\ \text{each row of } \boldsymbol{c} \text{ is in } \mathcal{B} \end{array} \right\} \subseteq \mathbb{F}_q^{m \times n}$$

Def. A code $\mathcal{C}$ has locality r and **availability** t if, for all $1 \leq i \leq n$, the coordinate i admits at least t pairwise disjoint helper sets.

Tensor product of codes $\mathcal{A} \subseteq \mathbb{F}_q^m$ and $\mathcal{B} \in \mathbb{F}_q^n$:

$$\mathcal{A} \otimes \mathcal{B} = \left\{ \boldsymbol{c} \in \mathbb{F}_q^{m \times n} \;\middle|\; \begin{array}{l} \text{each column of } \boldsymbol{c} \text{ is in } \mathcal{A} \\ \text{each row of } \boldsymbol{c} \text{ is in } \mathcal{B} \end{array} \right\} \subseteq \mathbb{F}_q^{m \times n}$$

There are $t = 2$ ways to retrieve $c_{i,j}$ efficiently:

1. select the i-th row of $\boldsymbol{c}$ (except $c_{i,j}$) and use a parity-check equation of $\mathcal{B}$:

$$c_{i,j} = \sum_{\ell \neq j} \lambda_\ell c_{i,\ell}$$

2. select the j-th column of $\boldsymbol{c}$ (except $c_{i,j}$) and use a parity-check equation of $\mathcal{A}$:

$$c_{i,j} = \sum_{\ell \leq i} \mu_\ell c_{\ell,j}$$

Def. A code $\mathcal{C}$ has locality r and **availability** t if, for all $1 \leq i \leq n$, the coordinate i admits at least t pairwise disjoint helper sets.

Tensor product of codes $\mathcal{A} \subseteq \mathbb{F}_q^m$ and $\mathcal{B} \in \mathbb{F}_q^n$:

$$\mathcal{A} \otimes \mathcal{B} = \left\{ \boldsymbol{c} \in \mathbb{F}_q^{m \times n} \;\middle|\; \begin{array}{l} \text{each column of } \boldsymbol{c} \text{ is in } \mathcal{A} \\ \text{each row of } \boldsymbol{c} \text{ is in } \mathcal{B} \end{array} \right\} \subseteq \mathbb{F}_q^{m \times n}$$

There are $t = 2$ ways to retrieve $c_{i,j}$ efficiently:

1. select the i-th row of $\boldsymbol{c}$ (except $c_{i,j}$) and use a parity-check equation of $\mathcal{B}$:

$$c_{i,j} = \sum_{\ell \neq j} \lambda_\ell c_{i,\ell}$$

2. select the j-th column of $\boldsymbol{c}$ (except $c_{i,j}$) and use a parity-check equation of $\mathcal{A}$:

$$c_{i,j} = \sum_{\ell \leq i} \mu_\ell c_{\ell,j}$$

Prop. If $\mathcal{A}$ and $\mathcal{B}$ are r-LRCs (without availability), then $\mathcal{A} \otimes \mathcal{B}$ is also r-LR, with availability $t = 2$.

Reminder: availability (2)

This can be extended to m-wise tensor products (or m-th tensor powers) of codes:

Prop. Let $\mathcal{C} \subseteq \mathbb{F}_q^n$ be an $[n, k, d]_q$-LRC with locality r. Then

$$\mathcal{C}^{\otimes t} := \mathcal{C} \otimes \cdots \otimes \mathcal{C}$$

is a code of length n^t, dimension k^t and minimum distance d^t. It is also an LRC with locality r and availability t.

This can be extended to *m*-wise tensor products (or *m*-th tensor powers) of codes:

Prop. Let $\mathcal{C} \subseteq \mathbb{F}_q^n$ be an $[n, k, d]_q$-LRC with locality r. Then

$$\mathcal{C}^{\otimes t} := \mathcal{C} \otimes \cdots \otimes \mathcal{C}$$

is a code of length n^t, dimension k^t and minimum distance d^t. It is also an LRC with locality r and availability t.

In a **worst case scenario**, $t + 1$ erasures lie on a coordinate i and on 1 coordinate of each of its t helper sets. Then,

- the symbol c_i cannot be decoded;
- we get no information about c_i.

This can be extended to m-wise tensor products (or m-th tensor powers) of codes:

Prop. Let $\mathcal{C} \subseteq \mathbb{F}_q^n$ be an $[n, k, d]_q$-LRC with locality r. Then

$$\mathcal{C}^{\otimes t} := \mathcal{C} \otimes \cdots \otimes \mathcal{C}$$

is a code of length n^t, dimension k^t and minimum distance d^t. It is also an LRC with locality r and availability t.

In a **worst case scenario**, $t+1$ erasures lie on a coordinate i and on 1 coordinate of each of its t helper sets. Then,

- the symbol c_i cannot be decoded;
- we get no information about c_i.

For tensor codes, t is only $\log(\text{length}(\mathcal{C}))$...

We would like to retrieve **some information** about c_i, with few access to $\boldsymbol{c}$, and even with a large amount of errors/erasures.

Another paradigm

Usual decoding algorithms for a code $\mathcal{C} \subseteq \mathbb{F}_q^n$:

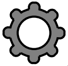
$$\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e} \in \mathbb{F}_q^n \longrightarrow \boldsymbol{c} \in \mathcal{C}$$

Properties.

1. Allow to recover $\boldsymbol{c}$ entirely, despite a large fraction of errors.
2. For algebraic codes, mostly deteministic algorithms.
3. But complexity often in $\Omega(n^2)$, at least in $\Omega(n)$.

Another paradigm

Usual decoding algorithms for a code $\mathcal{C} \subseteq \mathbb{F}_q^n$:



$$\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e} \in \mathbb{F}_q^n \quad \longrightarrow \quad \boldsymbol{c} \in \mathcal{C}$$

Properties.

1. Allow to recover $\boldsymbol{c}$ entirely, despite a large fraction of errors.
2. For algebraic codes, mostly deteministic algorithms.
3. But complexity often in $\Omega(n^2)$, at least in $\Omega(n)$.

New goal. Recover only **one** symbol c_i of a codeword $\boldsymbol{c} \in \mathcal{C}$, but in sublinear time $o(n)$, thus **probabilistically**.

1. How many errors can be handled?
2. Which probability of success?

Locally correctable codes

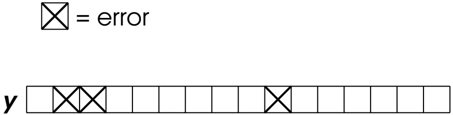
Consider a **locality** parameter $\ell \geq 2$, an **error ratio** $\tau \in (0, 1)$, a **soundness parameter** $\varepsilon < 1/2$.

Consider a **locality** parameter $\ell \geq 2$, an **error ratio** $\tau \in (0,1)$, a **soundness parameter** $\varepsilon < 1/2$.

Def. A code $\mathcal{C} \in \mathbb{F}_q^n$ is said $(\ell, \tau, \varepsilon)$**-locally correctable** if there exists a (probabilistic) algorithm Cor such that, for all noisy codewords $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e}$ with $\boldsymbol{c} \in \mathcal{C}$ and $|\boldsymbol{e}| \leq \tau n$, we have

1. *(soundness)* $\mathbb{P}(\mathrm{Cor}^{(\boldsymbol{y})}(i) \neq c_i) \leq \varepsilon$;
2. *(locality)* $\mathrm{Cor}^{(\boldsymbol{y})}(i)$ makes at most ℓ queries to $\boldsymbol{y}$.

Rem. $\mathbb{P}$ is computed w.r.t. random coins of Cor, not over $\boldsymbol{e}$.



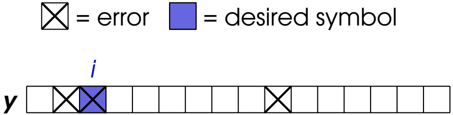
Locally correctable codes

Consider a **locality** parameter $\ell \geq 2$, an **error ratio** $\tau \in (0,1)$, a **soundness parameter** $\varepsilon < 1/2$.

Def. A code $\mathcal{C} \in \mathbb{F}_q^n$ is said $(\ell, \tau, \varepsilon)$**-locally correctable** if there exists a (probabilistic) algorithm Cor such that, for all noisy codewords $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e}$ with $\boldsymbol{c} \in \mathcal{C}$ and $|\boldsymbol{e}| \leq \tau n$, we have

1. *(soundness)* $\mathbb{P}(\mathrm{Cor}^{(\boldsymbol{y})}(i) \neq c_i) \leq \varepsilon$;
2. *(locality)* $\mathrm{Cor}^{(\boldsymbol{y})}(i)$ makes at most ℓ queries to $\boldsymbol{y}$.

Rem. $\mathbb{P}$ is computed w.r.t. random coins of Cor, not over $\boldsymbol{e}$.

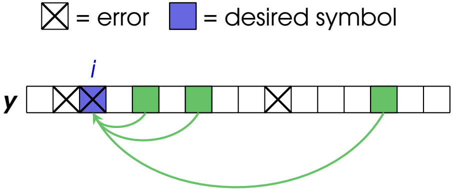


Consider a **locality** parameter $\ell \geq 2$, an **error ratio** $\tau \in (0, 1)$, a **soundness parameter** $\varepsilon < 1/2$.

Def. A code $\mathcal{C} \in \mathbb{F}_q^n$ is said $(\ell, \tau, \varepsilon)$**-locally correctable** if there exists a (probabilistic) algorithm Cor such that, for all noisy codewords $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e}$ with $\boldsymbol{c} \in \mathcal{C}$ and $|\boldsymbol{e}| \leq \tau n$, we have

1. *(soundness)* $\mathbb{P}(\mathrm{Cor}^{(\boldsymbol{y})}(i) \neq c_i) \leq \varepsilon$;
2. *(locality)* $\mathrm{Cor}^{(\boldsymbol{y})}(i)$ makes at most ℓ queries to $\boldsymbol{y}$.

Rem. $\mathbb{P}$ is computed w.r.t. random coins of Cor, not over $\boldsymbol{e}$.

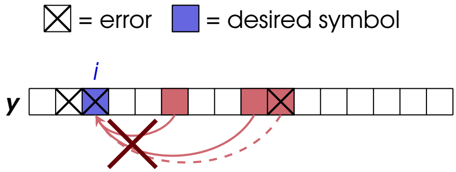


Consider a **locality** parameter $\ell \geq 2$, an **error ratio** $\tau \in (0, 1)$, a **soundness parameter** $\varepsilon < 1/2$.

Def. A code $\mathcal{C} \in \mathbb{F}_q^n$ is said $(\ell, \tau, \varepsilon)$**-locally correctable** if there exists a (probabilistic) algorithm Cor such that, for all noisy codewords $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e}$ with $\boldsymbol{c} \in \mathcal{C}$ and $|\boldsymbol{e}| \leq \tau n$, we have

1. *(soundness)* $\mathbb{P}(\mathrm{Cor}^{(\boldsymbol{y})}(i) \neq c_i) \leq \varepsilon$;
2. *(locality)* $\mathrm{Cor}^{(\boldsymbol{y})}(i)$ makes at most ℓ queries to $\boldsymbol{y}$.

Rem. $\mathbb{P}$ is computed w.r.t. random coins of Cor, not over $\boldsymbol{e}$.

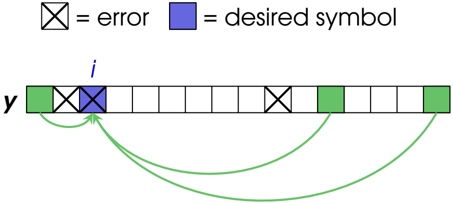


Consider a **locality** parameter $\ell \geq 2$, an **error ratio** $\tau \in (0, 1)$, a **soundness parameter** $\varepsilon < 1/2$.

Def. A code $\mathcal{C} \in \mathbb{F}_q^n$ is said $(\ell, \tau, \varepsilon)$**-locally correctable** if there exists a (probabilistic) algorithm Cor such that, for all noisy codewords $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e}$ with $\boldsymbol{c} \in \mathcal{C}$ and $|\boldsymbol{e}| \leq \tau n$, we have

1. *(soundness)* $\mathbb{P}(\mathrm{Cor}^{(\boldsymbol{y})}(i) \neq c_i) \leq \varepsilon$;
2. *(locality)* $\mathrm{Cor}^{(\boldsymbol{y})}(i)$ makes at most ℓ queries to $\boldsymbol{y}$.

Rem. $\mathbb{P}$ is computed w.r.t. random coins of Cor, not over $\boldsymbol{e}$.



Relevant quantities

Goals.

- small **locality** ℓ, depending on n.

Relevant quantities

Goals.

- small **locality** ℓ, depending on n.
- large **information rate** k/n, if possible $k/n \in \Omega(1)$.

Relevant quantities

Goals.

- small **locality** ℓ, depending on n.
- large **information rate** k/n, if possible $k/n \in \Omega(1)$.
- minimum distance is of secondary importance.

Hadamard code

Def. The **Hadamard code** consists in evaluation vectors of linear forms $\langle \boldsymbol{a} \mid \cdot \rangle$ (= homogeneous polynomials of degree 1) over all points $\boldsymbol{x} \in \mathbb{F}_q^k$:

$$\mathrm{Had}_k = \{ (\langle \boldsymbol{a} \mid \boldsymbol{x} \rangle)_{\boldsymbol{x} \in \mathbb{F}_q^k}, \boldsymbol{a} \in \mathbb{F}_q^k \} \subseteq \mathbb{F}_q^{q^k}$$

Parameters: $[n = q^k, k, q^k - q^{k-1}]_q$.

Hadamard code

Def. The **Hadamard code** consists in evaluation vectors of linear forms $\langle \boldsymbol{a} \mid \cdot \rangle$ (= homogeneous polynomials of degree 1) over all points $\boldsymbol{x} \in \mathbb{F}_q^k$:

$$\mathrm{Had}_k = \{(\langle \boldsymbol{a} \mid \boldsymbol{x} \rangle)_{\boldsymbol{x} \in \mathbb{F}_q^k}, \boldsymbol{a} \in \mathbb{F}_q^k\} \subseteq \mathbb{F}_q^{q^k}$$

Parameters: $[n = q^k, k, q^k - q^{k-1}]_q$.

Local correction algorithm for $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e}$, at location indexed by $\boldsymbol{x} \in \mathbb{F}_q^k$:

- pick uniformly at random $\boldsymbol{u} \in \mathbb{F}_q^k \setminus \{0\}$;
- return $y_{\boldsymbol{x}+\boldsymbol{u}} - y_{\boldsymbol{u}} \in \mathbb{F}_q$.

Hadamard code

Def. The **Hadamard code** consists in evaluation vectors of linear forms $\langle \boldsymbol{a} \mid \cdot \rangle$ (= homogeneous polynomials of degree 1) over all points $\boldsymbol{x} \in \mathbb{F}_q^k$:

$$\mathrm{Had}_k = \{(\langle \boldsymbol{a} \mid \boldsymbol{x} \rangle)_{\boldsymbol{x} \in \mathbb{F}_q^k}, \boldsymbol{a} \in \mathbb{F}_q^k\} \subseteq \mathbb{F}_q^{q^k}$$

Parameters: $[n = q^k, k, q^k - q^{k-1}]_q$.

Local correction algorithm for $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e}$, at location indexed by $\boldsymbol{x} \in \mathbb{F}_q^k$:

- pick uniformly at random $\boldsymbol{u} \in \mathbb{F}_q^k \setminus \{0\}$;
- return $y_{\boldsymbol{x}+\boldsymbol{u}} - y_{\boldsymbol{u}} \in \mathbb{F}_q$.

If $|\boldsymbol{e}| \leq \tau n$, then by union bound:

$$\mathbb{P}(\mathrm{Cor}^{(\boldsymbol{y})}(\boldsymbol{x}) = c_{\boldsymbol{x}})$$

Def. The **Hadamard code** consists in evaluation vectors of linear forms $\langle \boldsymbol{a} \mid \cdot \rangle$ (= homogeneous polynomials of degree 1) over all points $\boldsymbol{x} \in \mathbb{F}_q^k$:

$$\mathrm{Had}_k = \{ (\langle \boldsymbol{a} \mid \boldsymbol{x} \rangle)_{\boldsymbol{x} \in \mathbb{F}_q^k}, \boldsymbol{a} \in \mathbb{F}_q^k \} \subseteq \mathbb{F}_q^{q^k}$$

Parameters: $[n = q^k, k, q^k - q^{k-1}]_q$.

Local correction algorithm for $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e}$, at location indexed by $\boldsymbol{x} \in \mathbb{F}_q^k$:

- pick uniformly at random $\boldsymbol{u} \in \mathbb{F}_q^k \setminus \{0\}$;
- return $y_{\boldsymbol{x}+\boldsymbol{u}} - y_{\boldsymbol{u}} \in \mathbb{F}_q$.

If $|\boldsymbol{e}| \leq \tau n$, then by union bound:

$$\mathbb{P}(\mathrm{Cor}^{(\boldsymbol{y})}(\boldsymbol{x}) = c_{\boldsymbol{x}}) \ \geq \ \mathbb{P}_{\boldsymbol{u} \leftarrow \mathbb{F}_q^k}\Big(y_{\boldsymbol{u}+\boldsymbol{x}} = c_{\boldsymbol{u}+\boldsymbol{x}} \text{ and } y_{\boldsymbol{u}} = c_{\boldsymbol{u}} \Big) \ \geq \ 1 - 2\tau\,.$$

Def. The **Hadamard code** consists in evaluation vectors of linear forms $\langle \boldsymbol{a} \mid \cdot \rangle$ (= homogeneous polynomials of degree 1) over all points $\boldsymbol{x} \in \mathbb{F}_q^k$:

$$\mathrm{Had}_k = \left\{ (\langle \boldsymbol{a} \mid \boldsymbol{x} \rangle)_{\boldsymbol{x} \in \mathbb{F}_q^k}, \boldsymbol{a} \in \mathbb{F}_q^k \right\} \subseteq \mathbb{F}_q^{q^k}$$

Parameters: $[n = q^k, k, q^k - q^{k-1}]_q$.

Local correction algorithm for $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e}$, at location indexed by $\boldsymbol{x} \in \mathbb{F}_q^k$:

- pick uniformly at random $\boldsymbol{u} \in \mathbb{F}_q^k \setminus \{0\}$;
- return $y_{\boldsymbol{x}+\boldsymbol{u}} - y_{\boldsymbol{u}} \in \mathbb{F}_q$.

If $|\boldsymbol{e}| \leq \tau n$, then by union bound:

$$\mathbb{P}(\mathrm{Cor}^{(\boldsymbol{y})}(\boldsymbol{x}) = c_{\boldsymbol{x}}) \;\geq\; \mathbb{P}_{\boldsymbol{u} \leftarrow \mathbb{F}_q^k}\Big(y_{\boldsymbol{u}+\boldsymbol{x}} = c_{\boldsymbol{u}+\boldsymbol{x}} \text{ and } y_{\boldsymbol{u}} = c_{\boldsymbol{u}} \Big) \;\geq\; 1 - 2\tau\,.$$

Thm. Hadamard code is $(\ell = 2, \tau, \varepsilon = 2\tau)$–locally correctable, for all $\tau < 1/2$.

(**Rem.** This is also an LRC with $r = 2$ and very high availability $t \simeq n/2$.)

Summary for the Hadamard code

Summary.

- Hadamard code has optimal locality $\ell = 2$
- Hadamard code has dimension $k \in O(\log(n))$; hence rate $\to 0$.

Summary for the Hadamard code

Summary.

- Hadamard code has optimal locality $\ell = 2$
- Hadamard code has dimension $k \in O(\log(n))$; hence rate $\to 0$.

However, this is **essentially optimal:** any LCC with $\ell = 2$ is of expenential length.

Exponential Lower Bound for 2-query Locally Decodable Codes via a Quantum Argument. I. Kerenidis, R. de Wolf. J. Comput. Syst. Sci.. **2004**.

Summary for the Hadamard code

Summary.

- Hadamard code has optimal locality $\ell = 2$
- Hadamard code has dimension $k \in O(\log(n))$; hence rate $\to 0$.

However, this is **essentially optimal:** any LCC with $\ell = 2$ is of expenential length.

Exponential Lower Bound for 2-query Locally Decodable Codes via a Quantum Argument. I. Kerenidis, R. de Wolf. J. Comput. Syst. Sci.. **2004**.

Q. Can we reach constant rate $k/n \in \Omega(1)$ with sublinear locality $\ell = o(n)$?

Def. Let $m \geq 1$ and $r \leq q-1$. The **Reed–Muller code** of order m and degree r over $\mathbb{F}_q$ is:

$$\mathrm{RM}_q(m,r) = \left\{ \mathrm{ev}_{\mathbb{F}_q^m}(f) := (f(\boldsymbol{x}))_{\boldsymbol{x} \in \mathbb{F}_q^m} \mid f \in \mathbb{F}_q[X_1, \dots, X_m],\ \deg f \leq r \right\}$$

($r \leq q-1$ can be relaxed)

Def. Let $m \geq 1$ and $r \leq q-1$. The **Reed–Muller code** of order m and degree r over $\mathbb{F}_q$ is:

$$\mathrm{RM}_q(m,r) = \left\{ \mathrm{ev}_{\mathbb{F}_q^m}(f) := (f(\boldsymbol{x}))_{\boldsymbol{x}\in\mathbb{F}_q^m} \mid f \in \mathbb{F}_q[X_1,\dots,X_m],\ \deg f \leq r \right\}$$

($r \leq q-1$ can be relaxed)

If $r \leq q-1$, then parameters of $\mathrm{RM}_q(r,m)$ are $[q^m,\ \binom{m+r}{r},\ (q-r)q^{m-1}]_q$.

Def. Let $m \geq 1$ and $r \leq q-1$. The **Reed–Muller code** of order m and degree r over $\mathbb{F}_q$ is:

$$\mathrm{RM}_q(m,r) = \left\{ \mathrm{ev}_{\mathbb{F}_q^m}(f) := (f(\boldsymbol{x}))_{\boldsymbol{x}\in\mathbb{F}_q^m} \mid f \in \mathbb{F}_q[X_1,\dots,X_m],\ \deg f \leq r \right\}$$

($r \leq q-1$ can be relaxed)

If $r \leq q-1$, then parameters of $\mathrm{RM}_q(r,m)$ are $[q^m, \binom{m+r}{r}, (q-r)q^{m-1}]_q$.

Remark. For $m=1$, we have a Reed–Solomon code.

Let $\boldsymbol{c} = \mathrm{ev}_{\mathbb{F}_q^m}(f) \in \mathrm{RM}_q(m, r)$ and $\phi : \mathbb{F}_q \to \mathbb{F}_q^m$ an affine map:

$$\phi(T) = (a_1 T + b_1, \dots, a_m T + b_m) \qquad \text{with } (a_1, \dots, a_m) \neq (0, \dots, 0)$$

Then,

$$f \circ \phi \in \mathbb{F}_q[T] \qquad \text{and} \qquad \deg(f \circ \phi) \leq r$$

Let $\boldsymbol{c} = \mathrm{ev}_{\mathbb{F}_q^m}(f) \in \mathrm{RM}_q(m, r)$ and $\phi : \mathbb{F}_q \to \mathbb{F}_q^m$ an affine map:

$$\phi(T) = (a_1 T + b_1, \dots, a_m T + b_m) \qquad \text{with } (a_1, \dots, a_m) \neq (0, \dots, 0)$$

Then,

$$f \circ \phi \in \mathbb{F}_q[T] \qquad \text{and} \qquad \deg(f \circ \phi) \leq r$$

In other words,

$$\mathrm{ev}_{\mathbb{F}_q}(f \circ \phi) \text{ is a codeword of } \mathrm{RS}_q(r+1)\,.$$

Local properties of Reed–Muller codes

Let $\boldsymbol{c} = \mathrm{ev}_{\mathbb{F}_q^m}(f) \in \mathrm{RM}_q(m, r)$ and $\phi : \mathbb{F}_q \to \mathbb{F}_q^m$ an affine map:

$$\phi(T) = (a_1 T + b_1, \ldots, a_m T + b_m) \qquad \text{with } (a_1, \ldots, a_m) \neq (0, \ldots, 0)$$

Then,

$$f \circ \phi \in \mathbb{F}_q[T] \qquad \text{and} \qquad \deg(f \circ \phi) \leq r$$

In other words,

$$\mathrm{ev}_{\mathbb{F}_q}(f \circ \phi) \text{ is a codeword of } \mathrm{RS}_q(r + 1)\,.$$

Let $L := \phi(\mathbb{F}_q)$ be the corresponding affine line of $\mathbb{F}_q^m$. Up to reordering the coordinates,

$$\mathrm{ev}_{\mathbb{F}_q}(f \circ \phi) = \mathrm{ev}(f)_{|L}$$

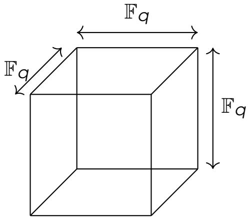
Local correction of Reed–Muller codes (1)

Local correcting algorithm for Reed–Muller codes $\mathrm{RM}_q(r, m)$ of degree $r \leq q - 2$.

Input:

- a coordinate $\boldsymbol{x} \in \mathbb{F}_q^m$
- an oracle access to symbols of $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e} \in \mathbb{F}_q^{q^m}$ such that $\boldsymbol{c} \in \mathrm{RM}_q(r, m)$ and $|\boldsymbol{e}| \leq \tau q^m$.

Expected output: symbol $\boldsymbol{c}_{\boldsymbol{x}} \in \mathbb{F}_q$



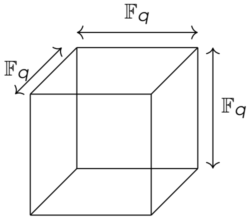
Local correcting algorithm for Reed–Muller codes $\mathrm{RM}_q(r, m)$ of degree $r \le q - 2$.

Input:

- a coordinate $\boldsymbol{x} \in \mathbb{F}_q^m$
- an oracle access to symbols of $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e} \in \mathbb{F}_q^{q^m}$ such that $\boldsymbol{c} \in \mathrm{RM}_q(r, m)$ and $|\boldsymbol{e}| \le \tau q^m$.

Expected output: symbol $\boldsymbol{c}_{\boldsymbol{x}} \in \mathbb{F}_q$

1. draw at random an affine line $L \subset \mathbb{F}_q^m$ passing through $\boldsymbol{x}$,
2. collect $\boldsymbol{y}_{|L \setminus \{\boldsymbol{x}\}}$, and correct it with any bounded-radius Reed–Solomon correcting algorithm,
3. output the (corrected) value of $y_{\boldsymbol{x}}$.



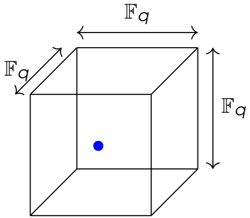
Local correcting algorithm for Reed–Muller codes $\mathrm{RM}_q(r, m)$ of degree $r \leq q - 2$.

Input:

- a coordinate $\boldsymbol{x} \in \mathbb{F}_q^m$
- an oracle access to symbols of $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e} \in \mathbb{F}_q^{q^m}$ such that $\boldsymbol{c} \in \mathrm{RM}_q(r, m)$ and $|\boldsymbol{e}| \leq \tau q^m$.

Expected output: symbol $\boldsymbol{c}_{\boldsymbol{x}} \in \mathbb{F}_q$

1. draw at random an affine line $L \subset \mathbb{F}_q^m$ passing through $\boldsymbol{x}$,
2. collect $\boldsymbol{y}_{|L \setminus \{\boldsymbol{x}\}}$, and correct it with any bounded-radius Reed–Solomon correcting algorithm,
3. output the (corrected) value of $y_{\boldsymbol{x}}$.



Local correcting algorithm for Reed–Muller codes $\mathrm{RM}_q(r, m)$ of degree $r \leq q-2$.

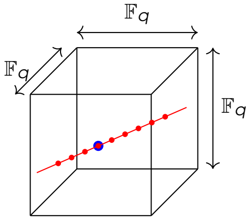
Input:

– a coordinate $\boldsymbol{x} \in \mathbb{F}_q^m$

– an oracle access to symbols of $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e} \in \mathbb{F}_q^{q^m}$ such that $\boldsymbol{c} \in \mathrm{RM}_q(r, m)$ and $|\boldsymbol{e}| \leq \tau q^m$.

Expected output: symbol $\boldsymbol{c}_{\boldsymbol{x}} \in \mathbb{F}_q$

1. draw at random an affine line $L \subset \mathbb{F}_q^m$ passing through $\boldsymbol{x}$,
2. collect $\boldsymbol{y}_{|L \setminus \{\boldsymbol{x}\}}$, and correct it with any bounded-radius Reed–Solomon correcting algorithm,
3. output the (corrected) value of $y_{\boldsymbol{x}}$.



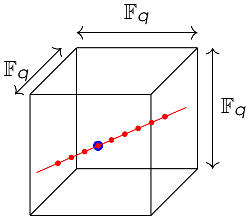
Local correcting algorithm for Reed–Muller codes $\mathrm{RM}_q(r,m)$ of degree $r \le q-2$.

Input:

- a coordinate $\boldsymbol{x} \in \mathbb{F}_q^m$
- an oracle access to symbols of $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e} \in \mathbb{F}_q^{q^m}$ such that $\boldsymbol{c} \in \mathrm{RM}_q(r,m)$ and $|\boldsymbol{e}| \le \tau q^m$.

Expected output: symbol $\boldsymbol{c}_{\boldsymbol{x}} \in \mathbb{F}_q$

1. draw at random an affine line $L \subset \mathbb{F}_q^m$ passing through $\boldsymbol{x}$,
2. collect $\boldsymbol{y}_{|L\setminus\{\boldsymbol{x}\}}$, and correct it with any bounded-radius Reed–Solomon correcting algorithm,
3. output the (corrected) value of $y_{\boldsymbol{x}}$.



Remark: we require $r \le q-2$ so that the local Reed–Solomon code has distance ≥ 2.

Local correction of Reed–Muller codes (2)

Let

$$\tau = \frac{q-r-2}{2(q-1)} > 0$$

be such that the Reed-Solomon code $\mathrm{RS}_q(r+1)$ can correct 1 erasure and $\tau(q-1)$ errors.

Prop. The Reed–Muller code $\mathrm{RM}_q(r,m)$ is locally correctable, with **locality** $\ell = q-1$, for all errors of relative weight $\delta < \tau/2$, and with soundness δ/τ.

Local correction of Reed–Muller codes (2)

Let
$$\tau = \frac{q-r-2}{2(q-1)} > 0$$
be such that the Reed-Solomon code $\mathrm{RS}_q(r+1)$ can correct 1 erasure and $\tau(q-1)$ errors.

Prop. The Reed–Muller code $\mathrm{RM}_q(r,m)$ is locally correctable, with **locality** $\ell = q-1$, for all errors of relative weight $\delta < \tau/2$, and with soundness δ/τ.

Let $\boldsymbol{x}$ be a coordinate to be corrected.
$$\begin{aligned}
\mathbb{P}(\text{decoding fails at } \boldsymbol{x}) &\leq \mathbb{P}_L(\#\text{errors in } L > \tau(q-1)) \\
&\leq \frac{\mathbb{E}_B[\#\text{errors in } L]}{\tau(q-1)+1} && \text{(Markov's inequality)} \\
&< \frac{\delta(q-1)}{\tau(q-1)} = \frac{\delta}{\tau} && \text{(linearity of } \mathbb{E}\text{)}
\end{aligned}$$

Let

$$\tau = \frac{q-r-2}{2(q-1)} > 0$$

be such that the Reed-Solomon code $\mathrm{RS}_q(r+1)$ can correct 1 erasure and $\tau(q-1)$ errors.

Prop. The Reed–Muller code $\mathrm{RM}_q(r,m)$ is locally correctable, with **locality** $\ell = q-1$, for all errors of relative weight $\delta < \tau/2$, and with soundness δ/τ.

Let $\boldsymbol{x}$ be a coordinate to be corrected.

$$\begin{aligned}
\mathbb{P}(\text{decoding fails at } \boldsymbol{x}) &\leq \mathbb{P}_L(\#\text{errors in } L > \tau(q-1)) \\
&\leq \frac{\mathbb{E}_B[\#\text{errors in } L]}{\tau(q-1)+1} && \text{(Markov's inequality)} \\
&< \frac{\delta(q-1)}{\tau(q-1)} = \frac{\delta}{\tau} && \text{(linearity of } \mathbb{E}\text{)}
\end{aligned}$$

Parameters:

Local correction of Reed–Muller codes (2)

Let
$$\tau = \frac{q-r-2}{2(q-1)} > 0$$
be such that the Reed-Solomon code $\mathrm{RS}_q(r+1)$ can correct 1 erasure and $\tau(q-1)$ errors.

Prop. The Reed–Muller code $\mathrm{RM}_q(r,m)$ is locally correctable, with **locality** $\ell = q-1$, for all errors of relative weight $\delta < \tau/2$, and with soundness δ/τ.

Let $\boldsymbol{x}$ be a coordinate to be corrected.

$$\begin{aligned}
\mathbb{P}(\text{decoding fails at } \boldsymbol{x}) &\le \mathbb{P}_L(\#\text{errors in } L > \tau(q-1)) \\
&\le \frac{\mathbb{E}_B[\#\text{errors in } L]}{\tau(q-1)+1} && \text{(Markov's inequality)} \\
&< \frac{\delta(q-1)}{\tau(q-1)} = \frac{\delta}{\tau} && \text{(linearity of } \mathbb{E})
\end{aligned}$$

Parameters:

– Locality $\ell = O(n^{1/m}) = o(n)$

Local correction of Reed–Muller codes (2)

Let

$$\tau = \frac{q - r - 2}{2(q-1)} > 0$$

be such that the Reed-Solomon code $\mathrm{RS}_q(r+1)$ can correct 1 erasure and $\tau(q-1)$ errors.

Prop. The Reed–Muller code $\mathrm{RM}_q(r,m)$ is locally correctable, with **locality** $\ell = q-1$, for all errors of relative weight $\delta < \tau/2$, and with soundness δ/τ.

Let $\boldsymbol{x}$ be a coordinate to be corrected.

$$\begin{aligned}
\mathbb{P}(\text{decoding fails at } \boldsymbol{x}) &\leq \mathbb{P}_L(\#\text{errors in } L > \tau(q-1)) \\
&\leq \frac{\mathbb{E}_B[\#\text{errors in } L]}{\tau(q-1)+1} && \text{(Markov's inequality)} \\
&< \frac{\delta(q-1)}{\tau(q-1)} = \frac{\delta}{\tau} && \text{(linearity of } \mathbb{E}\text{)}
\end{aligned}$$

Parameters:

- Locality $\ell = O(n^{1/m}) = o(n)$
- Information rate

$$\frac{k}{n} \simeq \frac{(r/q)^m}{m!} \simeq \frac{(1-2\tau)^m}{m!} < \frac{1}{m!} \cdots$$

Looking for higher rates (1)

LCC with sublinear locality and rate $\frac{k}{n} \geq 1/2$.

LCC with sublinear locality and rate $\frac{k}{n} \geq 1/2$.

New affine-invariant codes from lifting. A. Guo, S. Kopparty, M. Sudan. ITCS. **2013**.

1. **Affine-invariant "lifted codes":**

$$\mathcal{C} = \left\{ \mathrm{ev}_{\mathbb{F}_q^m}(f) \mid f \in \mathbb{F}_q[X_1, \dots, X_m], \forall \text{ line } L, ev_L(f) \in \mathrm{RS}_q(r) \right\} \subseteq \mathbb{F}_q^{q^m} .$$

LCC with sublinear locality and rate $\frac{k}{n} \geq 1/2$.

New affine-invariant codes from lifting. A. Guo, S. Kopparty, M. Sudan. ITCS. **2013**.

1. Affine-invariant "lifted codes":

$$\mathcal{C} = \left\{ \mathrm{ev}_{\mathbb{F}_q^m}(f) \mid f \in \mathbb{F}_q[X_1, \dots, X_m], \forall \text{ line } L, ev_L(f) \in \mathrm{RS}_q(r) \right\} \subseteq \mathbb{F}_q^{q^m}.$$

They share the **same local properties** as Reed–Muller codes.
Quite surprisingly, in some cases, they reach much higher rates than RM codes ($k/n \to 1$ in some contexts).

Looking for higher rates (2)

LCC with sublinear locality and rate $\frac{k}{n} \geq 1/2$.

LCC with sublinear locality and rate $\frac{k}{n} \geq 1/2$.

High-Rate Codes with Sublinear-Time Decoding. S. Kopparty, S. Saraf, S. Yekhanin. J. ACM. **2014**.

2. Multiplicity codes: evaluation of m-variate polynomials **and their Hasse derivative** up to some order s.

$$\mathcal{C} = \left\{ \mathrm{ev}^{<s}_{\mathbb{F}_q^m}(f) \mid f \in \mathbb{F}_q[X_1, \dots, X_m] \right\} \subseteq (\mathbb{F}_q^u)^{q^m}, \quad \text{where } u := \binom{m+s}{s}$$

Looking for higher rates (2)

LCC with sublinear locality and rate $\frac{k}{n} \geq 1/2$.

High-Rate Codes with Sublinear-Time Decoding. S. Kopparty, S. Saraf, S. Yekhanin. J. ACM. **2014**.

2. Multiplicity codes: evaluation of m-variate polynomials **and their Hasse derivative** up to some order s.

$$\mathcal{C} = \left\{ \mathrm{ev}_{\mathbb{F}_q^m}^{<s}(f) \mid f \in \mathbb{F}_q[X_1, \dots, X_m] \right\} \subseteq (\mathbb{F}_q^u)^{q^m}, \quad \text{where } u := \binom{m+s}{s}$$

Idea: Reed–Muller codes require $r \leq q-1$ for locality; multiplicity codes allow $r \leq s(q-1)$.

Reach higher rates than lifted codes.

Fast encoding algorithm.

Minor drawback: large alphabet.

LCC with sublinear locality and rate $\frac{k}{n} \geq 1/2$.

High-Rate Codes with Sublinear-Time Decoding. S. Kopparty, S. Saraf, S. Yekhanin. J. ACM. **2014**.

2. Multiplicity codes: evaluation of m-variate polynomials **and their Hasse derivative** up to some order s.

$$\mathcal{C} = \left\{ \mathrm{ev}^{<s}_{\mathbb{F}_q^m}(f) \mid f \in \mathbb{F}_q[X_1, \dots, X_m] \right\} \subseteq (\mathbb{F}_q^u)^{q^m}, \quad \text{where } u := \binom{m+s}{s}$$

Idea: Reed–Muller codes require $r \leq q-1$ for locality; multiplicity codes allow $r \leq s(q-1)$.

- Reach higher rates than lifted codes.
- Fast encoding algorithm.
- Minor drawback: large alphabet.

Rem. Can be mixed up with lifted codes: "lifted multiplicity codes".

Looking for higher rates (3)

LCC with sublinear locality and rate $\frac{k}{n} \geq 1/2$.

LCC with sublinear locality and rate $\frac{k}{n} \geq 1/2$.

Local Correctability of Expander Codes. B. Hemenway, R. Ostrovsky, M. Wootters. Inf. Comp.. **2015**.

3. Some families of **expander codes:** codes indexed by edges E of a d-regular expander graph $\mathcal{G} = (V, E)$, with local constraints in the neighbourhood of each vertex.

$$\mathcal{C}(\mathcal{G}, \mathcal{C}_0) = \left\{ \boldsymbol{c} \in \mathbb{F}_q^E \mid \forall v \in V, \boldsymbol{c}_{|\Gamma(v)} \in \mathcal{C}_0 \right\}$$

where $\mathcal{C}_0$ is a small "local code" and $\Gamma(v)$ is the set of edges adjacent to $v \in V$.

Rem. if $\mathcal{C}_0$ has rate R_0, then $\mathcal{C}(\mathcal{G}, \mathcal{C}_0)$ has rate $2R_0 - 1$.

Definition. A *t-design* of parameters (n, ℓ, λ) consists in:

- a set X of points, $|X| = n$,
- a set $\mathcal{B}$ of blocks $B \subset X$, $|B| = \ell$

such that every t-set in X appears in exactly λ blocks.

Block designs

Definition. A *t-design* of parameters (n, ℓ, λ) consists in:

- a set X of points, $|X| = n$,
- a set $\mathcal{B}$ of blocks $B \subset X$, $|B| = \ell$

such that every t-set in X appears in exactly λ blocks.

Incidence matrix of a design:

$$\text{blocks in } \mathcal{B} \left\{ \overbrace{\begin{pmatrix} & & & & * & & & & \\ 0 & \cdots & 0 & \mathbf{1} & \cdots & \mathbf{1} & 0 & \cdots & 0 \\ & & & & * & & & & \end{pmatrix}}^{\text{points in } X} \right. \longleftarrow \text{indicator vector of } \textbf{block } B$$

Def. The **code based on the design** $\mathcal{D} = (X, \mathcal{B})$ is the code $\mathcal{C} = \mathrm{Code}(\mathcal{D}) \subseteq \mathbb{F}_q^X$ admitting the incidence matrix of $\mathcal{D}$ as a parity-check matrix. In other words:

$$\mathrm{Code}(\mathcal{D}) = \{\boldsymbol{c} \in \mathbb{F}_q^X \mid \forall B \in \mathcal{B}, \boldsymbol{c}_{|B} \in \mathrm{Parity}\}$$

Remark. The dimension of $\mathrm{Code}(\mathcal{D})$ is highly dependent on the field $\mathbb{F}_q$

Codes based on designs, and generalization

Def. The **code based on the design** $\mathcal{D} = (X, \mathcal{B})$ is the code $\mathcal{C} = \mathrm{Code}(\mathcal{D}) \subseteq \mathbb{F}_q^X$ admitting the incidence matrix of $\mathcal{D}$ as a parity-check matrix. In other words:

$$\mathrm{Code}(\mathcal{D}) = \{\boldsymbol{c} \in \mathbb{F}_q^X \mid \forall B \in \mathcal{B}, \boldsymbol{c}_{|B} \in \mathrm{Parity}\}$$

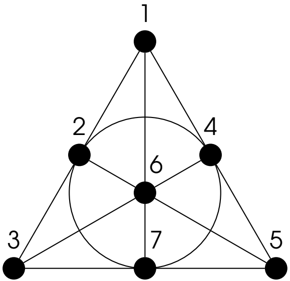
Remark. The dimension of $\mathrm{Code}(\mathcal{D})$ is highly dependent on the field $\mathbb{F}_q$

Def. Let $(\mathcal{C}_B)_{B \in \mathcal{B}}$ be a family of codes indexed by blocks $B \in \mathcal{B}$. The **generalised design-based code** based on $(\mathcal{D}, \mathcal{B})$ is

$$\mathrm{Code}(\mathcal{D}, (\mathcal{C}_B)_B) := \{\boldsymbol{c} \in \mathbb{F}_q^X \mid \forall B \in \mathcal{B}, \boldsymbol{c}_{|B} \in \mathcal{C}_B\}.$$

Design-based codes: an example

Fano plane $\mathcal{D}$



Corresponding incidence matrix

$$\boldsymbol{H} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

The design-based code $\mathcal{C} = \mathrm{Code}(\mathcal{D})$ is:

- a $[7, 3, 4]_2$ code if $q = 2$ (simplex code);
- a $[7, 1, 7]_2$ code if $q = 3$;
- the zero code for other primes q.

Design-based codes are locally correctable

Generalised design-based code $\mathcal{C} = \mathrm{Code}(\mathcal{D}, (\mathcal{C}_B)_B)$, where

- $\mathcal{D} = (X, \mathcal{B})$ is a t-$(n, \ell + 1, \lambda)$-design
- $\tau \in (0, \frac{1}{2})$ is fixed
- $(\mathcal{C}_B)_{B \in \mathcal{B}}$ is such that each code in $\mathcal{C}_B$ corrects a fraction τ of errors.

Algorithm. Local correction of $\boldsymbol{y} \in \mathbb{F}_q^X$ at $x \in X$

1. Pick uniformly at random a block $B \in \mathcal{B}$ such that $x \in B$.
2. Correct $\boldsymbol{y}_{|B}$ as a noisy codeword from $\mathcal{C}_B$, and output the (corrected) symbol y_x.

Design-based codes are locally correctable

Proposition ($t = 2$). For every $\delta < \tau/2$, $\mathrm{Code}(\mathcal{D}, (\mathcal{C}_B)_B)$ is a $(\ell, \delta, \varepsilon)$-LCC, where

$$\varepsilon = \delta/\tau\,.$$

Let i be a coordinate to be corrected.

$$\begin{aligned}
\mathbb{P}(\text{decoding fails at } i) &\le \mathbb{P}_B(\#\text{errors in } B_i \ge \tau\ell) \\
&\le \frac{\mathbb{E}_B[\#\text{errors in } B_i]}{\tau\ell} && \text{(Markov's inequality)} \\
&\lesssim \frac{\delta\ell}{\tau\ell} && \text{(linearity of } \mathbb{E}\text{)}
\end{aligned}$$

Proposition ($t = 3$). For every $\delta < \tau - 1/\sqrt{2\ell}$, $\mathrm{Code}(\mathcal{D}, (\mathcal{C}_B)_B)$ is a $(\ell, \delta, \varepsilon)$-LCC where

$$\varepsilon = \frac{\delta(1-\delta)}{(\tau-\delta)^2} \cdot \frac{1}{\ell} \le \frac{1}{\tau^2\ell} \cdot \delta\,.$$

(proof: Bienaymé–Tchebychev inequality)

Perspectives

- Designs-based codes with large dimension?
- Applications in crypto, *e.g.* private information retrieval