On the construction of PIR schemes

Julien Lavauzelle

IRMAR, Université de Rennes 1

Séminaire CASYS

31/01/2019

Outline

1. Private information retrieval

2. PIR schemes for common storage systems
   - Distributed storage systems
   - A PIR scheme on RS-coded databases
   - A PIR scheme with regenerating codes

3. PIR schemes with low computation
   - Transversal designs and codes
   - A PIR scheme with transversal designs
   - Instances

4. Conclusion

Outline

1. Private information retrieval

2. PIR schemes for common storage systems
   - Distributed storage systems
   - A PIR scheme on RS-coded databases
   - A PIR scheme with regenerating codes

3. PIR schemes with low computation
   - Transversal designs and codes
   - A PIR scheme with transversal designs
   - Instances

4. Conclusion

Private information retrieval (PIR):

Given a **remote** database $F \in \Sigma^M$ and $i \in [1, M]$,
can we **retrieve** the entry/file F_i,
without leaking information on the index i?

Private information retrieval (PIR):

Given a **remote** database $F \in \Sigma^M$ and $i \in [1, M]$,
can we **retrieve** the entry/file F_i,
without leaking information on the index i?

Trivial solution: full download.

Definition of PIR

Introduced in:

Private Information Retrieval. Chor, Goldreich, Kushilevitz, Sudan. FOCS. **1995**.

Database F stored (in some way) on n servers $S_1, \dots, S_n$,
user U wants to recover F_i privately.

A **Private Information Retrieval protocol** is a set of algorithms $(\mathcal{Q}, \mathcal{A}, \mathcal{R})$:

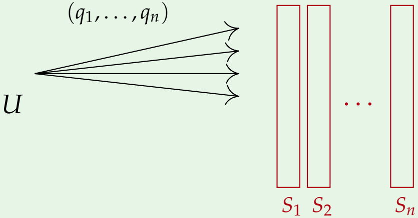
Introduced in:

Private Information Retrieval. Chor, Goldreich, Kushilevitz, Sudan. FOCS. **1995**.

Database F stored (in some way) on n servers $S_1, \dots, S_n$,
user U wants to recover F_i privately.

A **Private Information Retrieval protocol** is a set of algorithms $(\mathcal{Q}, \mathcal{A}, \mathcal{R})$:

1. U generates a query vector $\boldsymbol{q} = (q_1, \dots, q_n) \leftarrow \mathcal{Q}(i)$ and sends q_j to server S_j



Definition of PIR

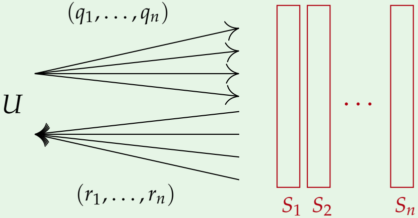
Introduced in:

Private Information Retrieval. Chor, Goldreich, Kushilevitz, Sudan. FOCS. **1995**.

Database F stored (in some way) on n servers $S_1, \dots, S_n$,
user U wants to recover F_i privately.

A **Private Information Retrieval protocol** is a set of algorithms $(\mathcal{Q}, \mathcal{A}, \mathcal{R})$:

1. U generates a query vector $\boldsymbol{q} = (q_1, \dots, q_n) \leftarrow \mathcal{Q}(i)$ and sends q_j to server S_j
2. Each server S_j computes $r_j = \mathcal{A}(q_j, F_{|S_j})$ and sends it back to U



Definition of PIR

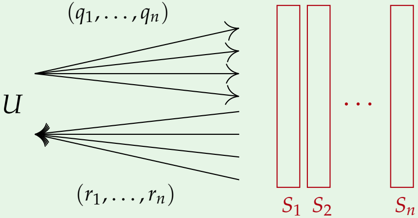
Introduced in:

Private Information Retrieval. Chor, Goldreich, Kushilevitz, Sudan. FOCS. **1995**.

Database F stored (in some way) on n servers $S_1, \dots, S_n$,
user U wants to recover F_i privately.

A **Private Information Retrieval protocol** is a set of algorithms $(\mathcal{Q}, \mathcal{A}, \mathcal{R})$:

1. U generates a query vector $\boldsymbol{q} = (q_1, \dots, q_n) \leftarrow \mathcal{Q}(i)$ and sends q_j to server S_j
2. Each server S_j computes $r_j = \mathcal{A}(q_j, F_{|S_j})$ and sends it back to U
3. U recovers $F_i = \mathcal{R}(\boldsymbol{q}, \boldsymbol{r}, i)$



Privacy

A **collusion of servers**: set of servers $\{S_j : j \in T\}$, where $T \subset [1, n]$, which exchange information about queries, data, etc.

$$t := \max\{|T|, T \subseteq [1, n] \text{ is a collusion}\} \geq 1$$

Privacy

A **collusion of servers**: set of servers $\{S_j : j \in T\}$, where $T \subset [1, n]$, which exchange information about queries, data, etc.

$$t := \max\{|T|, T \subseteq [1, n] \text{ is a collusion}\} \geq 1$$

- **Information-theoretic privacy:**

$$\mathrm{I}(i; \boldsymbol{q}_{|T}) = 0, \quad \forall T \subseteq [1, n], |T| \leq t.$$

- **Computational privacy:** by varying the index i, distributions of queries $\boldsymbol{q}_{|T} = \mathcal{Q}(i)_{|T}$ are computationally indistinguishable.

Privacy

A **collusion of servers**: set of servers $\{S_j : j \in T\}$, where $T \subset [1, n]$, which exchange information about queries, data, etc.

$$t := \max\{|T|, T \subseteq [1, n] \text{ is a collusion}\} \geq 1$$

• **Information-theoretic privacy:**

$$\mathrm{I}(i; \boldsymbol{q}_{|T}) = 0, \quad \forall T \subseteq [1, n], |T| \leq t.$$

• **Computational privacy:** by varying the index i, distributions of queries $\boldsymbol{q}_{|T} = \mathcal{Q}(i)_{|T}$ are computationally indistinguishable.

Theorem [CGKS95, CG97]. If $t = n$ (in particular if $n = 1$), then:

- for IT-privacy, **no better solution than full download**,
- computational privacy is possible (but remains **expensive** as of now).

We focus on **IT-privacy**
(hence we need $n \geq 2$ servers)

Main parameters of PIR schemes

We focus on **IT-privacy**
(hence we need $n \geq 2$ servers)

Parameters to be taken into account:

- **communication complexity** (upload and download)
- computation complexity (client and servers)
- global server storage overhead
- maximum size of collusions (t)

Main parameters of PIR schemes

We focus on **IT-privacy**
(hence we need $n \geq 2$ servers)

Parameters to be taken into account:

- **communication complexity** (upload and download)
- computation complexity (client and servers)
- global server storage overhead
- maximum size of collusions (t)

Several possible **settings**:

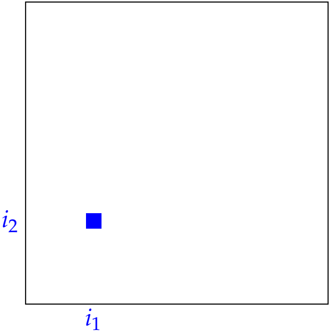
- bounded vs. unbounded number of entries in the database
- replicated database vs. coded database
- small entries vs. large entries
- dynamic database vs. static database
- unresponsive or byzantine servers

Seminal work [CGKS'95-98]

Private Information Retrieval. Chor, Goldreich, Kushilevitz, Sudan. FOCS. **1995**.

Settings:

- $|F| = M$ bits, with $M = L^2$, and $[1, M] \simeq [1, L]^2$.
- $n = 4$ servers $S_{00}, S_{01}, S_{10}, S_{11}$, each storing a replica of F.
- **Goal:** retrieve $F_i = F_{(i_1, i_2)}$, for $1 \leq i_1, i_2 \leq L$.

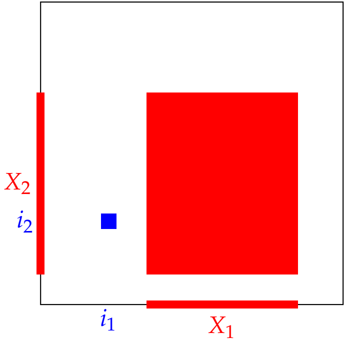


Seminal work [CGKS'95-98]

Private Information Retrieval. Chor, Goldreich, Kushilevitz, Sudan. FOCS. **1995**.

Settings:

- $|F| = M$ bits, with $M = L^2$, and $[1, M] \simeq [1, L]^2$.
- $n = 4$ servers $S_{00}, S_{01}, S_{10}, S_{11}$, each storing a replica of F.
- **Goal:** retrieve $F_i = F_{(i_1,i_2)}$, for $1 \leq i_1, i_2 \leq L$.

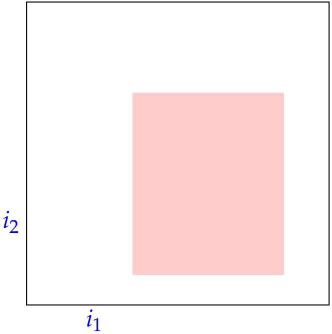


1. U generates at random two subsets X_1, X_2 of $[1, L]$. Then U sends:

Private Information Retrieval. Chor, Goldreich, Kushilevitz, Sudan. FOCS. **1995**.

Settings:

- $|F| = M$ bits, with $M = L^2$, and $[1, M] \simeq [1, L]^2$.
- $n = 4$ servers $S_{00}, S_{01}, S_{10}, S_{11}$, each storing a replica of F.
- **Goal:** retrieve $F_i = F_{(i_1, i_2)}$, for $1 \leq i_1, i_2 \leq L$.

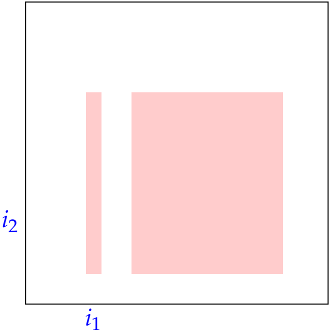


1. U generates at random two subsets X_1, X_2 of $[1, L]$. Then U sends:
   - $(\quad X_1 \quad , \quad X_2 \quad)$ to S_{00},

Private Information Retrieval. Chor, Goldreich, Kushilevitz, Sudan. FOCS. **1995**.

Settings:

- $|F| = M$ bits, with $M = L^2$, and $[1, M] \simeq [1, L]^2$.
- $n = 4$ servers $S_{00}, S_{01}, S_{10}, S_{11}$, each storing a replica of F.
- **Goal:** retrieve $F_i = F_{(i_1, i_2)}$, for $1 \leq i_1, i_2 \leq L$.



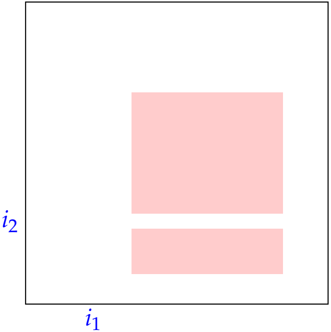
1. U generates at random two subsets X_1, X_2 of $[1, L]$. Then U sends:
   - (X_1, X_2) to S_{00},
   - $(X_1 \Delta \{i_1\}, X_2)$ to S_{10},

Seminal work [CGKS'95-98]

Private Information Retrieval. Chor, Goldreich, Kushilevitz, Sudan. FOCS. **1995**.

Settings:

- $|F| = M$ bits, with $M = L^2$, and $[1, M] \simeq [1, L]^2$.
- $n = 4$ servers $S_{00}, S_{01}, S_{10}, S_{11}$, each storing a replica of F.
- **Goal:** retrieve $F_i = F_{(i_1, i_2)}$, for $1 \leq i_1, i_2 \leq L$.



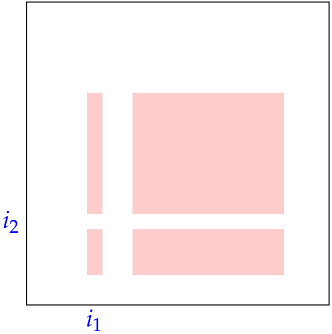
1. U generates at random two subsets X_1, X_2 of $[1, L]$. Then U sends:
   - (X_1, X_2) to S_{00},
   - $(X_1 \Delta \{i_1\}, X_2)$ to S_{10},
   - $(X_1, X_2 \Delta \{i_2\})$ to S_{01},

Seminal work [CGKS'95-98]

Private Information Retrieval. Chor, Goldreich, Kushilevitz, Sudan. FOCS. **1995**.

Settings:

- $|F| = M$ bits, with $M = L^2$, and $[1, M] \simeq [1, L]^2$.
- $n = 4$ servers $S_{00}, S_{01}, S_{10}, S_{11}$, each storing a replica of F.
- **Goal:** retrieve $F_i = F_{(i_1, i_2)}$, for $1 \leq i_1, i_2 \leq L$.



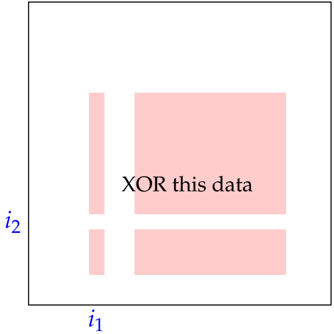
1. U generates at random two subsets X_1, X_2 of $[1, L]$. Then U sends:
   - (X_1, X_2) to S_{00},
   - $(X_1 \Delta \{i_1\}, X_2)$ to S_{10},
   - $(X_1, X_2 \Delta \{i_2\})$ to S_{01},
   - $(X_1 \Delta \{i_1\}, X_2 \Delta \{i_2\})$ to S_{11}.

Seminal work [CGKS'95-98]

Private Information Retrieval. Chor, Goldreich, Kushilevitz, Sudan. FOCS. **1995**.

Settings:

- $|F| = M$ bits, with $M = L^2$, and $[1, M] \simeq [1, L]^2$.
- $n = 4$ servers $S_{00}, S_{01}, S_{10}, S_{11}$, each storing a replica of F.
- **Goal:** retrieve $F_i = F_{(i_1, i_2)}$, for $1 \leq i_1, i_2 \leq L$.

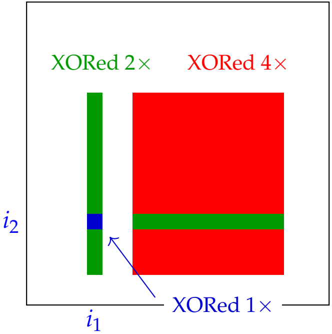


1. U generates at random two subsets X_1, X_2 of $[1, L]$. Then U sends:
   - (X_1, X_2) to S_{00},
   - $(X_1 \Delta \{i_1\}, X_2)$ to S_{10},
   - $(X_1, X_2 \Delta \{i_2\})$ to S_{01},
   - $(X_1 \Delta \{i_1\}, X_2 \Delta \{i_2\})$ to S_{11}.
2. At reception of (Z_1, Z_2), each server computes $a = \bigoplus_{z \in Z_1 \times Z_2} F_z$ and sends a to the user.

Private Information Retrieval. Chor, Goldreich, Kushilevitz, Sudan. FOCS. **1995**.

Settings:

- $|F| = M$ bits, with $M = L^2$, and $[1, M] \simeq [1, L]^2$.
- $n = 4$ servers $S_{00}, S_{01}, S_{10}, S_{11}$, each storing a replica of F.
- **Goal:** retrieve $F_i = F_{(i_1,i_2)}$, for $1 \leq i_1, i_2 \leq L$.



1. U generates at random two subsets X_1, X_2 of $[1, L]$. Then U sends:
   - (X_1, X_2) to S_{00},
   - $(X_1 \Delta \{i_1\}, X_2)$ to S_{10},
   - $(X_1, X_2 \Delta \{i_2\})$ to S_{01},
   - $(X_1 \Delta \{i_1\}, X_2 \Delta \{i_2\})$ to S_{11}.
2. At reception of (Z_1, Z_2), each server computes $a = \bigoplus_{z \in Z_1 \times Z_2} F_z$ and sends a to the user.
3. User XORs the 4 bits and retrieves F_i.

Features of the PIR scheme in [CGKS'95-98]

Correct, and **secure** if no collusion.

Correct, and **secure** if no collusion.

With $n = 4$ servers:

- **Communication:** $8\sqrt{M}$ uploaded bits, 4 downloaded bits,
- **Storage:** replication of F over 4 servers,
- **Complexity:**
  - for each server: in average, XOR of $(L/2)^2 = M/4$ bits
  - for the user: XOR of $n = 4$ bits.

Features of the PIR scheme in [CGKS'95-98]

Correct, and **secure** if no collusion.

With $n = 4$ servers:

- **Communication:** $8\sqrt{M}$ uploaded bits, 4 downloaded bits,
- **Storage:** replication of F over 4 servers,
- **Complexity:**
  - for each server: in average, XOR of $(L/2)^2 = M/4$ bits
  - for the user: XOR of $n = 4$ bits.

Generalisable to $n = 2^b$ servers:

- **Communication:** $b2^b M^{1/b} = n\log(n)M^{1/\log(n)}$ uploaded bits, n downloaded bits,
- **Storage:** replication of F over n servers,
- **Complexity:**
  - for each server: in average, XOR of M/n bits
  - for the user: XOR of n bits.

(Short) state of the art

- 1995: first definition [CGKS95]

- 2000: reduction from smooth locally decodable codes [KT00]

- 2000-10's: many improvements
  - PIR with 3 servers and subpolynomial communication [Yek08, Efr09]
  - PIR with 2 servers and subpolynomial communication [DG16]
  - lower storage overhead with *PIR codes* [FVY15]

- 2016-now: capacity-achieving schemes, schemes dedicated to storage systems
  - capacity of PIR [SJ17, BU18]
  - (nearly) capacity-achieving schemes [SRR14, CHY15, TR16, ...]

1. Private information retrieval

2. PIR schemes for common storage systems
   - Distributed storage systems
   - A PIR scheme on RS-coded databases
   - A PIR scheme with regenerating codes

3. PIR schemes with low computation
   - Transversal designs and codes
   - A PIR scheme with transversal designs
   - Instances

4. Conclusion

1. Private information retrieval

2. PIR schemes for common storage systems
Distributed storage systems
A PIR scheme on RS-coded databases
A PIR scheme with regenerating codes

3. PIR schemes with low computation
Transversal designs and codes
A PIR scheme with transversal designs
Instances

4. Conclusion

Context

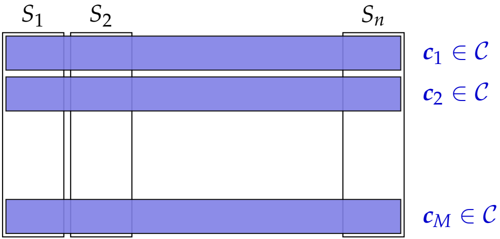
Storage systems use codes to cope with node failures.

- Before 2010: mostly replication or parity-check.
- 2010's: MDS storage (*e.g.* $[14, 10]$ Reed-Solomon code for Facebook).
- Recently: codes with locality (*e.g.* Hadoop Xorbas).

Storage systems use codes to cope with node failures.

- Before 2010: mostly replication or parity-check.
- 2010's: MDS storage (*e.g.* $[14, 10]$ Reed-Solomon code for Facebook).
- Recently: codes with locality (*e.g.* Hadoop Xorbas).

Given a code $\mathcal{C}$ of length n:



Definition (Reed-Solomon code). Let $\boldsymbol{x} = (x_1, \dots, x_n) \in \mathbb{F}_q^n$, pairwise distinct.

$$\mathrm{RS}_q(k, n) := \{(f(x_1), \dots, f(x_n)), f \in \mathbb{F}_q[X], \deg f < k\}$$

Definition (Reed-Solomon code). Let $\boldsymbol{x} = (x_1, \dots, x_n) \in \mathbb{F}_q^n$, pairwise distinct.

$$\mathrm{RS}_q(k,n) := \{(f(x_1), \dots, f(x_n)), f \in \mathbb{F}_q[X], \deg f < k\}$$

$\mathcal{C} = \mathrm{RS}_q(k,n)$ is **MDS**:

- every codeword $\boldsymbol{c} \in \mathcal{C}$ can be reconstructed from any k-subset of coordinates of $\boldsymbol{c}$,
- any subset of $d^{\perp}(\mathcal{C}) - 1 = k$ coordinates of $\boldsymbol{c}$ are independent.

Definition (Reed-Solomon code). Let $\boldsymbol{x} = (x_1, \dots, x_n) \in \mathbb{F}_q^n$, pairwise distinct.

$$\mathrm{RS}_q(k,n) := \{(f(x_1), \dots, f(x_n)), f \in \mathbb{F}_q[X], \deg f < k\}$$

$\mathcal{C} = \mathrm{RS}_q(k,n)$ is **MDS**:

- every codeword $\boldsymbol{c} \in \mathcal{C}$ can be reconstructed from any k-subset of coordinates of $\boldsymbol{c}$,
- any subset of $d^{\perp}(\mathcal{C}) - 1 = k$ coordinates of $\boldsymbol{c}$ are independent.

File storage:

$$\text{a file} \quad F_i \in \Sigma \simeq \mathbb{F}_{q^s}^k \quad \text{is encoded into} \quad \boldsymbol{c}_i \in \mathrm{RS}_q(k,n) \otimes \mathbb{F}_{q^s}$$

Definition (Reed-Solomon code). Let $\boldsymbol{x} = (x_1, \dots, x_n) \in \mathbb{F}_q^n$, pairwise distinct.

$$\mathrm{RS}_q(k,n) := \{(f(x_1), \dots, f(x_n)), f \in \mathbb{F}_q[X], \deg f < k\}$$

$\mathcal{C} = \mathrm{RS}_q(k,n)$ is **MDS**:

- every codeword $\boldsymbol{c} \in \mathcal{C}$ can be reconstructed from any k-subset of coordinates of $\boldsymbol{c}$,
- any subset of $d^{\perp}(\mathcal{C}) - 1 = k$ coordinates of $\boldsymbol{c}$ are independent.

File storage:

$$\text{a file} \quad F_i \in \Sigma \simeq \mathbb{F}_{q^s}^k \quad \text{is encoded into} \quad \boldsymbol{c}_i \in \mathrm{RS}_q(k,n) \otimes \mathbb{F}_{q^s}$$

Main assumption (can be discussed):

$$s \gg M$$

1. Private information retrieval

2. PIR schemes for common storage systems
Distributed storage systems
A PIR scheme on RS-coded databases
A PIR scheme with regenerating codes

3. PIR schemes with low computation
Transversal designs and codes
A PIR scheme with transversal designs
Instances

4. Conclusion

Usual goal (assuming $s \gg M$): a large *PIR rate*

$$\rho := \frac{|F_i|}{|\boldsymbol{r}|} .$$

Presentation

Usual goal (assuming $s \gg M$): a large *PIR rate*

$$\rho := \frac{|F_i|}{|\boldsymbol{r}|}\,.$$

Next, we present a PIR scheme for RS-coded databases.

- Originally [TR16], then extended and reformulated [TGKFH18, TGR18].
- Scalable.
- Optimal PIR rate for $t = 1$ and $M \to \infty$.
- PIR rate conjectured optimal for $M \to \infty$.

[TR16] *PIR from MDS Coded Data in Distributed Storage Systems*. Tajeddine, El Rouayheb. ISIT. **2016**.

[TGKFH18] *Robust PIR from Coded Systems with Byzantine and Colluding Servers*. Tajeddine, Gnilke, Karpuk, Freij-Hollanti, Hollanti. ISIT. **2018**.

[TGR18] *PIR from MDS Coded Data in Distributed Storage Systems*. Tajeddine, Gnilke, El Rouayheb. IEEE-TIT. **2018**.

The protocol: query generation

Notation:
$$\boldsymbol{a} \star \boldsymbol{b} := (a_1 b_1, \dots, a_n b_n)$$
$$\mathcal{C} \star \mathcal{C}' := \langle \{ \boldsymbol{c} \star \boldsymbol{c}' \mid \boldsymbol{c} \in \mathcal{C}, \boldsymbol{c}' \in \mathcal{C}' \} \rangle$$

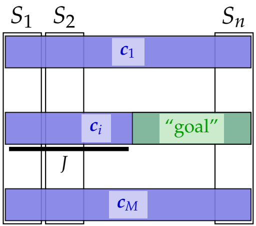
Notation:
$$\boldsymbol{a} \star \boldsymbol{b} := (a_1 b_1, \dots, a_n b_n)$$
$$\mathcal{C} \star \mathcal{C}' := \langle \{ \boldsymbol{c} \star \boldsymbol{c}' \mid \boldsymbol{c} \in \mathcal{C}, \boldsymbol{c}' \in \mathcal{C}' \} \rangle$$

System parameters:

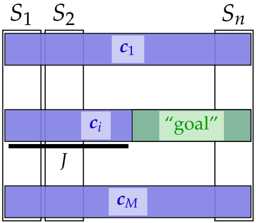
$\mathcal{C} \subseteq \mathbb{F}_q^n$ the *storage code*, $\boldsymbol{C} \in \mathcal{C}^M$ the coded database

$\mathcal{D} \subseteq \mathbb{F}_q^n$ a *query code* of dual distance $d^\perp(\mathcal{D}) = t + 1$

$J \subseteq [1, n]$ an information set for $\mathcal{C} \star \mathcal{D}$, and $\bar{J} := [1, n] \setminus J$



Notation:
$$\boldsymbol{a} \star \boldsymbol{b} := (a_1 b_1, \dots, a_n b_n)$$
$$\mathcal{C} \star \mathcal{C}' := \langle \{ \boldsymbol{c} \star \boldsymbol{c}' \mid \boldsymbol{c} \in \mathcal{C}, \boldsymbol{c}' \in \mathcal{C}' \} \rangle$$



System parameters:

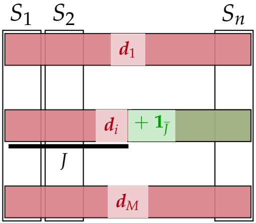
$\mathcal{C} \subseteq \mathbb{F}_q^n$ the *storage code*, $\boldsymbol{C} \in \mathcal{C}^M$ the coded database

$\mathcal{D} \subseteq \mathbb{F}_q^n$ a *query code* of dual distance $d^\perp(\mathcal{D}) = t + 1$

$J \subseteq [1, n]$ an information set for $\mathcal{C} \star \mathcal{D}$, and $\bar{J} := [1, n] \setminus J$

Queries:

1. the user generates at random M words $\boldsymbol{d}_1, \dots, \boldsymbol{d}_M \in \mathcal{D}$ and defines $\boldsymbol{Q}$ as follows:
2. the j-th column of $\boldsymbol{Q}$ is sent to server S_j



The protocol: query generation

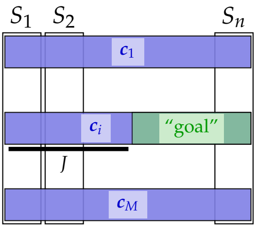
Notation:
$$\boldsymbol{a} \star \boldsymbol{b} := (a_1 b_1, \dots, a_n b_n)$$
$$\mathcal{C} \star \mathcal{C}' := \langle \{ \boldsymbol{c} \star \boldsymbol{c}' \mid \boldsymbol{c} \in \mathcal{C}, \boldsymbol{c}' \in \mathcal{C}' \} \rangle$$

System parameters:

$\mathcal{C} \subseteq \mathbb{F}_q^n$ the *storage code*, $\boldsymbol{C} \in \mathcal{C}^M$ the coded database

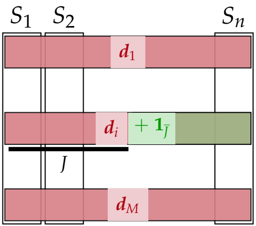
$\mathcal{D} \subseteq \mathbb{F}_q^n$ a *query code* of dual distance $d^\perp(\mathcal{D}) = t + 1$

$J \subseteq [1, n]$ an information set for $\mathcal{C} \star \mathcal{D}$, and $\bar{J} := [1, n] \setminus J$



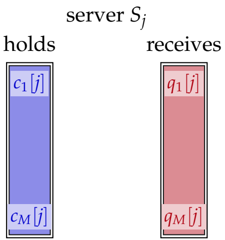
Queries:

1. the user generates at random M words $\boldsymbol{d}_1, \dots, \boldsymbol{d}_M \in \mathcal{D}$ and defines $\boldsymbol{Q}$ as follows:
2. the j-th column of $\boldsymbol{Q}$ is sent to server S_j



Remark: queries remain private against collusions of servers of size $\leq t$.

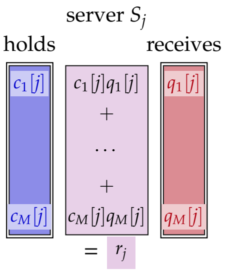
Server answers: server S_j receives as a query a column $\boldsymbol{Q}^{(j)} \in \mathbb{F}_q^M$ of $\boldsymbol{Q}$,



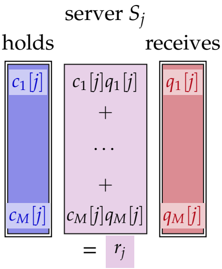
The protocol: server answers and reconstruction

Server answers: server S_j receives as a query a column $\boldsymbol{Q}^{(j)} \in \mathbb{F}_q^M$ of $\boldsymbol{Q}$, and has to compute

$$r_j = \langle \boldsymbol{Q}^{(j)}, \boldsymbol{C}^{(j)} \rangle \in \mathbb{F}_q.$$



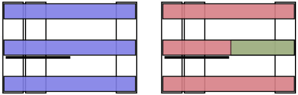
The protocol: server answers and reconstruction



Server answers: server S_j receives as a query a column $\boldsymbol{Q}^{(j)} \in \mathbb{F}_q^M$ of $\boldsymbol{Q}$, and has to compute

$$r_j = \langle \boldsymbol{Q}^{(j)}, \boldsymbol{C}^{(j)} \rangle \in \mathbb{F}_q.$$

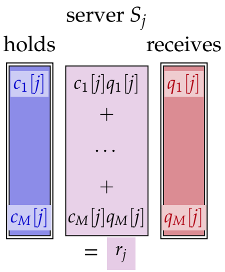
Reconstruction:



The protocol: server answers and reconstruction

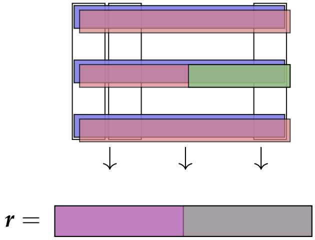
Server answers: server S_j receives as a query a column $\boldsymbol{Q}^{(j)} \in \mathbb{F}_q^M$ of $\boldsymbol{Q}$, and has to compute

$$r_j = \langle \boldsymbol{Q}^{(j)}, \boldsymbol{C}^{(j)} \rangle \in \mathbb{F}_q.$$



Reconstruction: The user collects

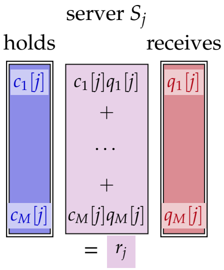
$$\boldsymbol{r} = (r_1, \dots, r_n) = \underbrace{\sum_{m=1}^{M} \boldsymbol{d}_m \star \boldsymbol{c}_m}_{\in \mathcal{C} \star \mathcal{D}} + \underbrace{\boldsymbol{1}_{\bar{J}} \star \boldsymbol{c}_i}_{= \boldsymbol{c}_i \text{ on } \bar{J}}$$



The protocol: server answers and reconstruction

Server answers: server S_j receives as a query a column $\boldsymbol{Q}^{(j)} \in \mathbb{F}_q^M$ of $\boldsymbol{Q}$, and has to compute

$$r_j = \langle \boldsymbol{Q}^{(j)}, \boldsymbol{C}^{(j)} \rangle \in \mathbb{F}_q .$$

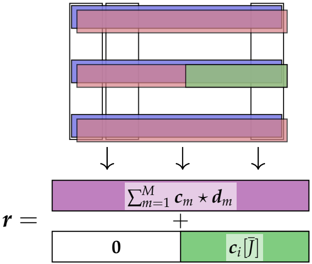


Reconstruction: The user collects

$$\boldsymbol{r} = (r_1, \dots, r_n) = \underbrace{\sum_{m=1}^{M} \boldsymbol{d}_m \star \boldsymbol{c}_m}_{\in \mathcal{C} \star \mathcal{D}} + \underbrace{\boldsymbol{1}_{\bar{J}} \star \boldsymbol{c}_i}_{= \boldsymbol{c}_i \text{ on } \bar{J}}$$

and interpolates on J to recover

- $\sum_{m=1}^{M} \boldsymbol{d}_m \star \boldsymbol{c}_m$,
- then $\boldsymbol{c}_i[|\bar{J}]$.



Features for 1 run of the protocol.

- download cost: n symbols over $\mathbb{F}_{q^s}$
- upload cost: an $(M \times n)$-matrix over $\mathbb{F}_q$ (negligible if $s \gg M$)
- retrieval of $|\bar{J}| = n - \dim(\mathcal{C} \star \mathcal{D})$ symbols of the desired file
- the protocol is **private** against collusions of size $\leq d^{\perp}(\mathcal{D}) - 1$

Features for 1 run of the protocol.

- download cost: n symbols over $\mathbb{F}_{q^s}$
- upload cost: an $(M \times n)$-matrix over $\mathbb{F}_q$ (negligible if $s \gg M$)
- retrieval of $|\bar{J}| = n - \dim(\mathcal{C} \star \mathcal{D})$ symbols of the desired file
- the protocol is **private** against collusions of size $\leq d^{\perp}(\mathcal{D}) - 1$

For **Reed-Solomon codes**: $\mathcal{C} = \mathrm{RS}_q(k, n)$ and $\mathcal{D} = \mathrm{RS}_q(t, n)$:

$$d^{\perp}(\mathcal{D}) - 1 = t \qquad \text{and} \qquad \mathcal{C} \star \mathcal{D} = \mathrm{RS}_q(k + t - 1, n) \Rightarrow |\bar{J}| = n - k - t + 1$$

Analysis

Features for 1 run of the protocol.

- download cost: n symbols over $\mathbb{F}_{q^s}$
- upload cost: an $(M \times n)$-matrix over $\mathbb{F}_q$ (negligible if $s \gg M$)
- retrieval of $|\bar{J}| = n - \dim(\mathcal{C} \star \mathcal{D})$ symbols of the desired file
- the protocol is **private** against collusions of size $\leq d^{\perp}(\mathcal{D}) - 1$

For **Reed-Solomon codes**: $\mathcal{C} = \mathrm{RS}_q(k, n)$ and $\mathcal{D} = \mathrm{RS}_q(t, n)$:

$$d^{\perp}(\mathcal{D}) - 1 = t \qquad \text{and} \qquad \mathcal{C} \star \mathcal{D} = \mathrm{RS}_q(k + t - 1, n) \Rightarrow |\bar{J}| = n - k - t + 1$$

If $(n - k - t + 1) \mid k$, then **repeating** several runs gives a (download) **PIR rate**:

$$\rho = \frac{n - k - t + 1}{n} = 1 - \frac{k + t - 1}{n}.$$

Features for 1 run of the protocol.

- download cost: n symbols over $\mathbb{F}_{q^s}$
- upload cost: an $(M \times n)$-matrix over $\mathbb{F}_q$ (negligible if $s \gg M$)
- retrieval of $|\bar{J}| = n - \dim(\mathcal{C} \star \mathcal{D})$ symbols of the desired file
- the protocol is **private** against collusions of size $\leq d^{\perp}(\mathcal{D}) - 1$

For **Reed-Solomon codes**: $\mathcal{C} = \mathrm{RS}_q(k, n)$ and $\mathcal{D} = \mathrm{RS}_q(t, n)$:

$$d^{\perp}(\mathcal{D}) - 1 = t \qquad \text{and} \qquad \mathcal{C} \star \mathcal{D} = \mathrm{RS}_q(k + t - 1, n) \Rightarrow |\bar{J}| = n - k - t + 1$$

If $(n - k - t + 1) \mid k$, then **repeating** several runs gives a (download) **PIR rate**:

$$\rho = \frac{n - k - t + 1}{n} = 1 - \frac{k + t - 1}{n}\,.$$

Otherwise, **striping** methods allow to achieve the same PIR rate.

Outline

1. Private information retrieval

2. PIR schemes for common storage systems
   Distributed storage systems
   A PIR scheme on RS-coded databases
   A PIR scheme with regenerating codes

3. PIR schemes with low computation
   Transversal designs and codes
   A PIR scheme with transversal designs
   Instances

4. Conclusion

!!! Sorry for the notation !!!

!!! Sorry for the notation !!!

Definition: $\mathcal{C}$ is an $(n, k, d, \alpha, \beta, B)$-**regenerating code** if:

- $\mathcal{C}$ is a linear space of dimension B, consisting in $(\alpha \times n)$-matrices over $\mathbb{F}_q$,
- every $\boldsymbol{c} \in \mathcal{C}$ is fully determined by any k-subset of columns,
- every column of $\boldsymbol{c}$ can be "repaired", by downloading $\beta \leq \alpha$ symbols from any d-subset of columns (hence $d\beta \geq \alpha$).

!!! Sorry for the notation !!!

Definition: $\mathcal{C}$ is an $(n, k, d, \alpha, \beta, B)$-**regenerating code** if:

- $\mathcal{C}$ is a linear space of dimension B, consisting in $(\alpha \times n)$-matrices over $\mathbb{F}_q$,
- every $\boldsymbol{c} \in \mathcal{C}$ is fully determined by any k-subset of columns,
- every column of $\boldsymbol{c}$ can be "repaired", by downloading $\beta \leq \alpha$ symbols from any d-subset of columns (hence $d\beta \geq \alpha$).

Main bound (cut-set bound [WDR07]):

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

!!! Sorry for the notation !!!

Definition: $\mathcal{C}$ is an $(n, k, d, \alpha, \beta, B)$-**regenerating code** if:

- $\mathcal{C}$ is a linear space of dimension B, consisting in $(\alpha \times n)$-matrices over $\mathbb{F}_q$,
- every $\boldsymbol{c} \in \mathcal{C}$ is fully determined by any k-subset of columns,
- every column of $\boldsymbol{c}$ can be "repaired", by downloading $\beta \leq \alpha$ symbols from any d-subset of columns (hence $d\beta \geq \alpha$).

Main bound (cut-set bound [WDR07]):

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

A particular optimal point (minimum-bandwidth repair, MBR): $d\beta = \alpha$.
Then,

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

Product-matrix MBR codes [RSK11]

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

We set $\beta = 1$, hence $\alpha = d$.

Product-matrix MBR codes [RSK11]

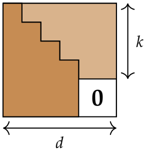
Optimal Exact-Regenerating Codes for Distributed Storage at the MSR and MBR Points via a Product-Matrix Construction. Rashmi, Shah, Kumar. IEEE-TIT. **2011**.

We set $\beta = 1$, hence $\alpha = d$.

1. Message symbols are arranged in a $(d \times d)$-matrix

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

   where $\boldsymbol{S}$ is $(k \times k)$-symmetric.



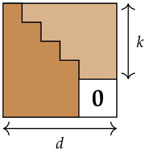
Optimal Exact-Regenerating Codes for Distributed Storage at the MSR and MBR Points via a Product-Matrix Construction. Rashmi, Shah, Kumar. IEEE-TIT. **2011**.

We set $\beta = 1$, hence $\alpha = d$.

1. Message symbols are arranged in a $(d \times d)$-matrix

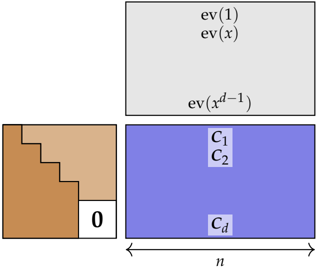
$$A = \begin{pmatrix} S & T^\top \\ T & \mathbf{0} \end{pmatrix}$$

   where S is $(k \times k)$-symmetric.



2. Let G be a $(d \times n)$ generator matrix for $\mathrm{RS}_q(d, n)$, echelonized in degree (*i.e.* a Vandermonde matrix). Codewords are then:

$$C = AG \in \mathbb{F}_q^{d \times n} .$$



Product-matrix MBR codes [RSK11]

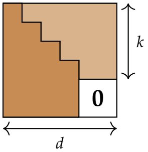
Optimal Exact-Regenerating Codes for Distributed Storage at the MSR and MBR Points via a Product-Matrix Construction. Rashmi, Shah, Kumar. IEEE-TIT. **2011**.

We set $\beta = 1$, hence $\alpha = d$.

1. Message symbols are arranged in a $(d \times d)$-matrix

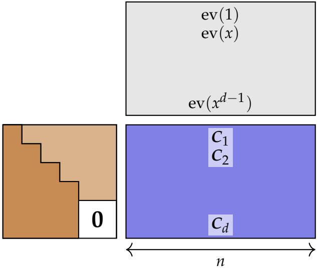
$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{S} & \boldsymbol{T}^\top \\ \boldsymbol{T} & \boldsymbol{0} \end{pmatrix}$$

   where $\boldsymbol{S}$ is $(k \times k)$-symmetric.



2. Let $\boldsymbol{G}$ be a $(d \times n)$ generator matrix for $\mathrm{RS}_q(d, n)$, echelonized in degree (*i.e.* a Vandermonde matrix). Codewords are then:

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



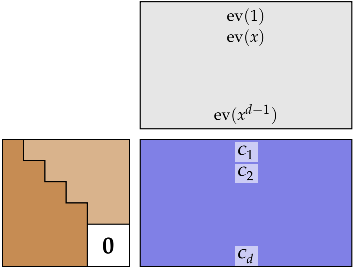
Remark: row $\boldsymbol{C}_j$ of $\boldsymbol{C}$ is a word of a RS code

- of dimension k, if $j > k$,
- of dimension $d > k$ otherwise.

PIR scheme on PM-MBR codes with no collusion

Private Information Retrieval Schemes with Regenerating Codes. L., Tajeddine, Freij-Hollanti, Hollanti. `arxiv:1811.02898`. **2018**.

PIR scheme with **no collusion** ($t = 1$).

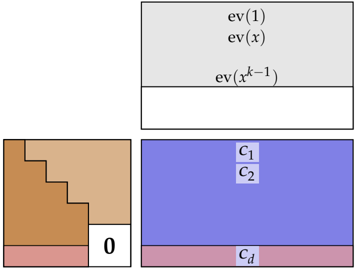


PIR scheme on PM-MBR codes with no collusion

Private Information Retrieval Schemes with Regenerating Codes. L., Tajeddine, Freij-Hollanti, Hollanti. `arxiv:1811.02898`. **2018**.

PIR scheme with **no collusion** ($t = 1$).

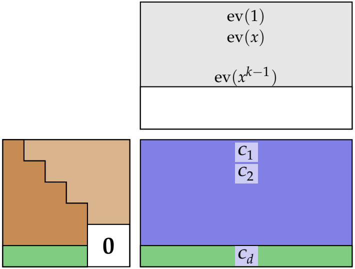
- For row $j = d$ down to $k + 1$:
  - Run a RS(k)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.



Private Information Retrieval Schemes with Regenerating Codes. L., Tajeddine, Freij-Hollanti, Hollanti. `arxiv:1811.02898`. **2018**.

PIR scheme with **no collusion** ($t = 1$).

- For row $j = d$ down to $k + 1$:
  - Run a RS(k)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.



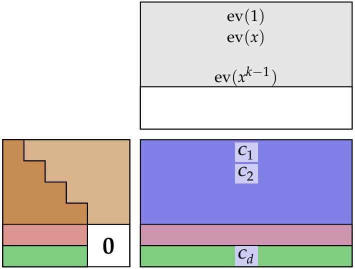
Retrieval rate: $1 - \frac{k}{n}$

PIR scheme on PM-MBR codes with no collusion

Private Information Retrieval Schemes with Regenerating Codes. L., Tajeddine, Freij-Hollanti, Hollanti. `arxiv:1811.02898`. **2018**.

PIR scheme with **no collusion** ($t = 1$).

- For row $j = d$ down to $k + 1$:
  - Run a $\mathrm{RS}(k)$-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.

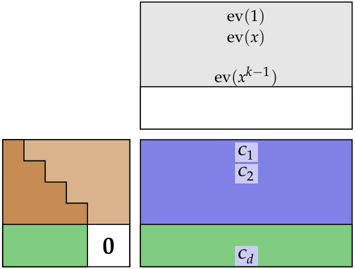


PIR scheme on PM-MBR codes with no collusion

Private Information Retrieval Schemes with Regenerating Codes. L., Tajeddine, Freij-Hollanti, Hollanti. `arxiv:1811.02898`. **2018**.

PIR scheme with **no collusion** ($t = 1$).

- For row $j = d$ down to $k + 1$:
  - Run a RS(k)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.



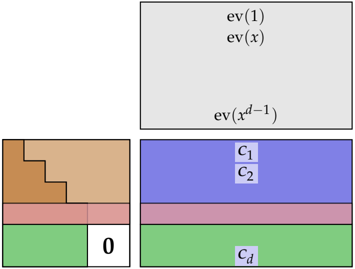
Retrieval rate: $1 - \frac{k}{n}$

PIR scheme on PM-MBR codes with no collusion

Private Information Retrieval Schemes with Regenerating Codes. L., Tajeddine, Freij-Hollanti, Hollanti. `arxiv:1811.02898`. **2018**.

PIR scheme with **no collusion** ($t = 1$).

- For row $j = d$ down to $k + 1$:
  - Run a RS(k)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.
- For row $j = k$ down to 1:
  - Run a RS(j)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Use symmetry of $\boldsymbol{A}$ and previously recovered data for the reconstruction (high-degree terms can be eliminated).
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.

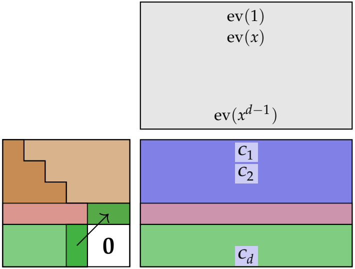


PIR scheme on PM-MBR codes with no collusion

Private Information Retrieval Schemes with Regenerating Codes. L., Tajeddine, Freij-Hollanti, Hollanti. `arxiv:1811.02898`. **2018**.

PIR scheme with **no collusion** ($t = 1$).

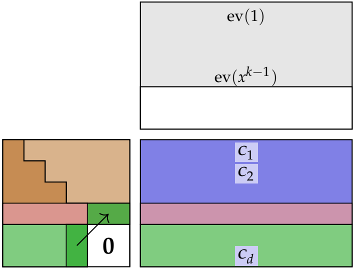
- For row $j = d$ down to $k + 1$:
  - Run a RS(k)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.
- For row $j = k$ down to 1:
  - Run a RS(j)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Use symmetry of $\boldsymbol{A}$ and previously recovered data for the reconstruction (high-degree terms can be eliminated).
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.



Private Information Retrieval Schemes with Regenerating Codes. L., Tajeddine, Freij-Hollanti, Hollanti. `arxiv:1811.02898`. **2018**.

PIR scheme with **no collusion** ($t = 1$).

- For row $j = d$ down to $k + 1$:
  - Run a RS(k)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.
- For row $j = k$ down to 1:
  - Run a RS(j)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Use symmetry of $\boldsymbol{A}$ and previously recovered data for the reconstruction (high-degree terms can be eliminated).
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.

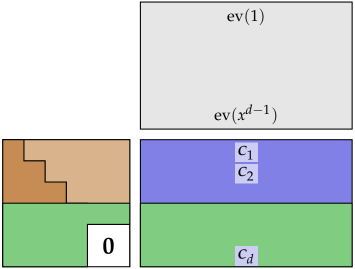


PIR scheme on PM-MBR codes with no collusion

Private Information Retrieval Schemes with Regenerating Codes. L., Tajeddine, Freij-Hollanti, Hollanti. `arxiv:1811.02898`. **2018**.

PIR scheme with **no collusion** ($t = 1$).

- For row $j = d$ down to $k + 1$:
  - Run a RS(k)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.
- For row $j = k$ down to 1:
  - Run a RS(j)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Use symmetry of $\boldsymbol{A}$ and previously recovered data for the reconstruction (high-degree terms can be eliminated).
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.



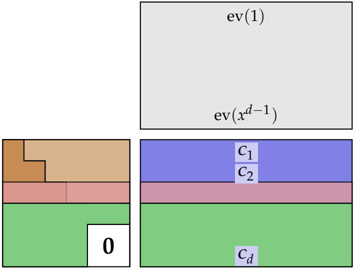
Retrieval rate: $1 - \frac{k}{n}$

PIR scheme on PM-MBR codes with no collusion

Private Information Retrieval Schemes with Regenerating Codes. L., Tajeddine, Freij-Hollanti, Hollanti. `arxiv:1811.02898`. **2018**.

PIR scheme with **no collusion** ($t = 1$).

- For row $j = d$ down to $k + 1$:
  - Run a RS(k)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.
- For row $j = k$ down to 1:
  - Run a RS(j)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Use symmetry of $\boldsymbol{A}$ and previously recovered data for the reconstruction (high-degree terms can be eliminated).
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.

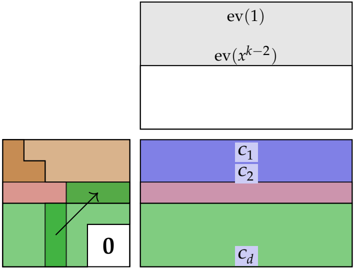


PIR scheme on PM-MBR codes with no collusion

Private Information Retrieval Schemes with Regenerating Codes. L., Tajeddine, Freij-Hollanti, Hollanti. `arxiv:1811.02898`. **2018**.

PIR scheme with **no collusion** ($t = 1$).

- For row $j = d$ down to $k + 1$:
  - Run a RS(k)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.
- For row $j = k$ down to 1:
  - Run a RS(j)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Use symmetry of $\boldsymbol{A}$ and previously recovered data for the reconstruction (high-degree terms can be eliminated).
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.

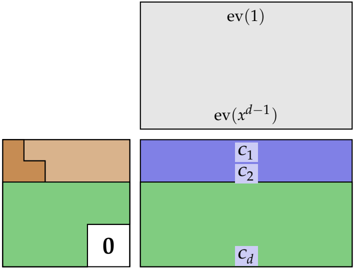


PIR scheme on PM-MBR codes with no collusion

Private Information Retrieval Schemes with Regenerating Codes. L., Tajeddine, Freij-Hollanti, Hollanti. `arxiv:1811.02898`. **2018**.

PIR scheme with **no collusion** ($t = 1$).

- For row $j = d$ down to $k + 1$:
  - Run a RS(k)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.
- For row $j = k$ down to 1:
  - Run a RS(j)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Use symmetry of $\boldsymbol{A}$ and previously recovered data for the reconstruction (high-degree terms can be eliminated).
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.



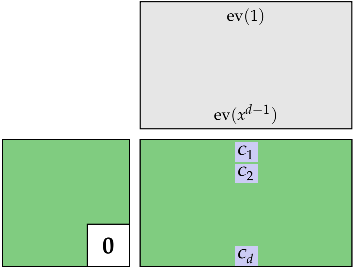
Retrieval rate: $1 - \frac{k-1}{n}$

PIR scheme on PM-MBR codes with no collusion

Private Information Retrieval Schemes with Regenerating Codes. L., Tajeddine, Freij-Hollanti, Hollanti. `arxiv:1811.02898`. **2018**.

PIR scheme with **no collusion** ($t = 1$).

- For row $j = d$ down to $k + 1$:
  - Run a RS(k)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.
- For row $j = k$ down to 1:
  - Run a RS(j)-coded PIR scheme with randomness $\boldsymbol{D}$.
  - Use symmetry of $\boldsymbol{A}$ and previously recovered data for the reconstruction (high-degree terms can be eliminated).
  - Interpolate random values $\sum \boldsymbol{d}_m \star \boldsymbol{C}_{j,m}$.
  - Recover row $\boldsymbol{C}_j$, then row $\boldsymbol{A}_j$.



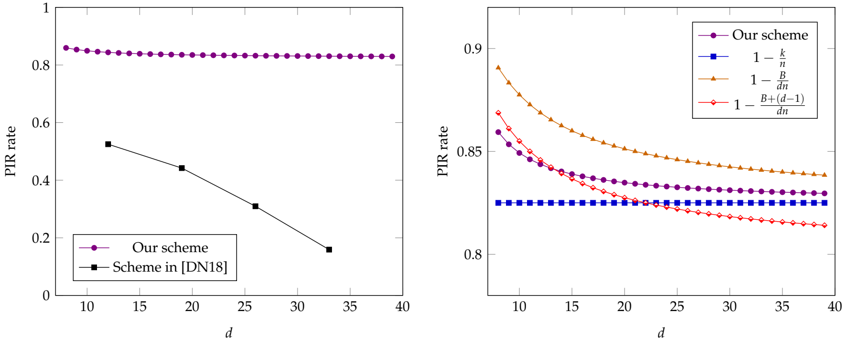
Retrieval rate: $1 - \frac{j}{n}$

We get a PIR rate:

$$\rho = \frac{1 - \frac{k}{n}}{1 - \frac{k(k+1)(k-1)}{nB}} \; > \; 1 - \frac{k}{n}$$

We get a PIR rate:

$$\rho = \frac{1 - \frac{k}{n}}{1 - \frac{k(k+1)(k-1)}{nB}} > 1 - \frac{k}{n}$$



Comparison of PIR rates for $n = 40$ and $k = 7$.

1. Private information retrieval

2. PIR schemes for common storage systems
Distributed storage systems
A PIR scheme on RS-coded databases
A PIR scheme with regenerating codes

3. PIR schemes with low computation
Transversal designs and codes
A PIR scheme with transversal designs
Instances

4. Conclusion

Previous schemes:

- low communication complexity
- computationally inefficient (linear in $|F| = \sum_{m=1}^{M} |F_m|$)

Our goal:

- optimal computation ($|r_j|$ for each server S_j)
- remove the assumption $s \gg M$
- moderate communication complexity

1. Private information retrieval

2. PIR schemes for common storage systems
- Distributed storage systems
- A PIR scheme on RS-coded databases
- A PIR scheme with regenerating codes

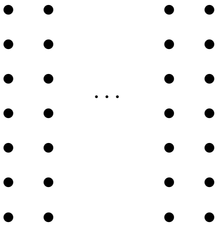
3. PIR schemes with low computation
- Transversal designs and codes
- A PIR scheme with transversal designs
- Instances

4. Conclusion

Transversal designs

A **transversal design** $\mathrm{TD}(n, s) = (X, \mathcal{B}, \mathcal{G})$ is given by:

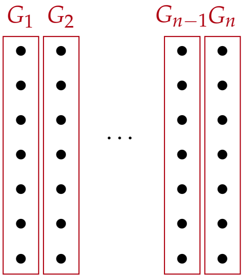
- X a set of *points*, $|X| = N = ns$,



A **transversal design** $\mathrm{TD}(n,s) = (X, \mathcal{B}, \mathcal{G})$ is given by:

- X a set of *points*, $|X| = N = ns$,
- *groups* $\mathcal{G} = \{G_j\}_{1 \le j \le n}$ satisfying

$$X = \coprod_{j=1}^{n} G_j \text{ and } |G_j| = s\,,$$

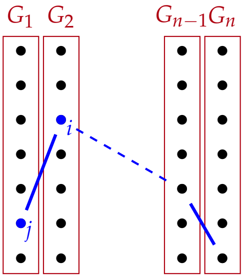


A **transversal design** $\mathrm{TD}(n,s) = (X, \mathcal{B}, \mathcal{G})$ is given by:

- X a set of *points*, $|X| = N = ns$,
- *groups* $\mathcal{G} = \{G_j\}_{1 \le j \le n}$ satisfying

$$X = \coprod_{j=1}^{n} G_j \text{ and } |G_j| = s\,,$$

- *blocks* $B \in \mathcal{B}$ satisfying
  - $B \subset X$ and $|B| = n$;
  - for all $\{i,j\} \subset X$, $\{i,j\}$ lie:
    either in a single group $G \in \mathcal{G}$,
    or in a unique block $B \in \mathcal{B}$



Let $\mathcal{T}$ be a transversal design $\mathrm{TD}(n, s) = (X, \mathcal{B}, \mathcal{G})$.

Its **incidence matrix** $\boldsymbol{M}$ has size $|\mathcal{B}| \times |X|$ and is defined by:

$$M_{i,j} = \begin{cases} 1 & \text{if } x_j \in B_i \\ 0 & \text{otherwise.} \end{cases}$$

Codes from designs

Let $\mathcal{T}$ be a transversal design $\mathrm{TD}(n,s) = (X, \mathcal{B}, \mathcal{G})$.

Its **incidence matrix** $\boldsymbol{M}$ has size $|\mathcal{B}| \times |X|$ and is defined by:

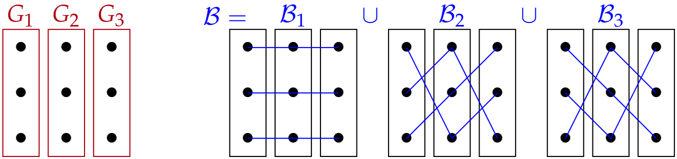
$$M_{i,j} = \begin{cases} 1 & \text{if } x_j \in B_i \\ 0 & \text{otherwise.} \end{cases}$$

The **code** $\mathcal{C}$ **based on** $\mathcal{T}$ **over** $\mathbb{F}_q$ is the $\mathbb{F}_q$-linear code admitting $\boldsymbol{M}$ as a parity-check matrix ($\mathcal{C}^\perp$ is generated by $\boldsymbol{M}$).

- $\mathrm{length}(\mathcal{C}) = |X|$,
- $\dim(\mathcal{C}) = \dim(\ker \boldsymbol{M})$,
- every $B \in \mathcal{B}$ gives an $\boldsymbol{h} \in \mathcal{C}^\perp$ such that $\mathrm{wt}(\boldsymbol{h}_{|G_j}) = 1$, $\forall j = 1, \dots, n$.

Example

The transversal design $\mathrm{TD}(3,3)$ represented by:



gives an incidence matrix

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

Its rank over $\mathbb{F}_3$ is 6 $\quad\Longrightarrow\quad$ the associated code $\mathcal{C}$ is a $[9,3]_3$ code.

1. Private information retrieval

2. PIR schemes for common storage systems
- Distributed storage systems
- A PIR scheme on RS-coded databases
- A PIR scheme with regenerating codes

3. PIR schemes with low computation
- Transversal designs and codes
- A PIR scheme with transversal designs
- Instances

4. Conclusion

The PIR scheme

Let $\mathcal{C} \subseteq \mathbb{F}_q^N$ be a code based on a $\mathrm{TD}(n, s)$.

Let $\mathcal{C} \subseteq \mathbb{F}_q^N$ be a code based on a $\mathrm{TD}(n, s)$.

- **Initialisation.** User U encodes $F \mapsto \boldsymbol{c} \in \mathcal{C}$, and gives $\boldsymbol{c}_{|G_j}$ to server S_j.

The PIR scheme

Let $\mathcal{C} \subseteq \mathbb{F}_q^N$ be a code based on a $\mathrm{TD}(n, s)$.

- **Initialisation.** User U encodes $F \mapsto \boldsymbol{c} \in \mathcal{C}$, and gives $\boldsymbol{c}_{|G_j}$ to server S_j.
- **To recover** $F_i = c_i$, with $i \in X$:
  1. User U randomly picks a block $B \in \mathcal{B}$ containing i.
     Then U defines:

$$q_j = \mathcal{Q}(i)_j := \begin{cases} \text{unique } \in B \cap G_j & \text{if } i \notin G_j \\ \text{a random point in } G_j & \text{otherwise.} \end{cases}$$

  2. Each server S_j sends back c_{q_j}
  3. U recovers

$$c_i = -\sum_{j:\, i \notin G_j} c_{q_j} = -\sum_{b \in B \setminus \{i\}} c_b$$

Theorem. This PIR protocol is information-theoretically private.

Proof:

- the only server which holds F_i received a random query;
- for each other server S_j, query q_j gives no information on the block B which has been picked $\Rightarrow$ no information leaks on i.

Theorem. This PIR protocol is information-theoretically private.

Proof:

- the only server which holds F_i received a random query;
- for each other server S_j, query q_j gives no information on the block B which has been picked $\Rightarrow$ no information leaks on i.

Features.

- communication complexity: $n \log s$ uploaded bits, $n \log q$ downloaded bits
- computational complexity:
  - **only** 1 **read for each server** (somewhat optimal)
  - $\leq n$ additions over $\mathbb{F}_q$ for the user
- storage overhead: $(ns - M) \log q$ bits, where $M = \dim(\mathcal{C})$

Privacy and parameters

Theorem. This PIR protocol is information-theoretically private.

Proof:

- the only server which holds F_i received a random query;
- for each other server S_j, query q_j gives no information on the block B which has been picked $\Rightarrow$ no information leaks on i.

Features.

- communication complexity: $n \log s$ uploaded bits, $n \log q$ downloaded bits
- computational complexity:
  - **only** 1 **read for each server** (somewhat optimal)
  - $\leq n$ additions over $\mathbb{F}_q$ for the user
- storage overhead: $(ns - M) \log q$ bits, where $M = \dim(\mathcal{C})$

Question: transversal designs with good $\dim(\mathcal{C})$ depending on (n, s)?

Outline

1. Private information retrieval

2. PIR schemes for common storage systems
   - Distributed storage systems
   - A PIR scheme on RS-coded databases
   - A PIR scheme with regenerating codes

3. PIR schemes with low computation
   - Transversal designs and codes
   - A PIR scheme with transversal designs
   - Instances

4. Conclusion

$\mathcal{T}_A$, the **classical affine transversal design**:

- $X = \mathbb{F}_q^m$, $m \geq 2$,
- $\mathcal{G}$ a set of q disjoint hyperplanes partitionning X,
- $\mathcal{B} = \{$affine lines L secant to each group of $\mathcal{G}\}$.

The code has:

- length $ns = q^m$,
- "locality" $n = q$.

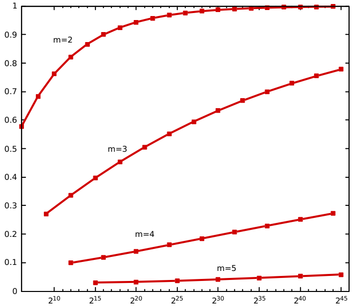
$\mathcal{T}_A$, the **classical affine transversal design**:

- $X = \mathbb{F}_q^m$, $m \geq 2$,
- $\mathcal{G}$ a set of q disjoint hyperplanes partitionning X,
- $\mathcal{B} = \{$affine lines L secant to each group of $\mathcal{G}\}$.

The code has:

- length $ns = q^m$,
- "locality" $n = q$.

rate M/N



length $N = ns = 2^{em}$

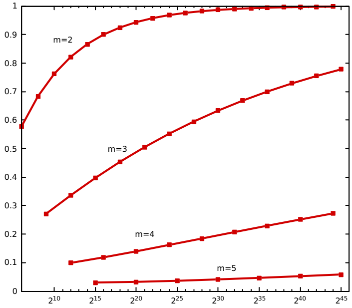
$\mathcal{T}_A$, the **classical affine transversal design**:

- $X = \mathbb{F}_q^m$, $m \geq 2$,
- $\mathcal{G}$ a set of q disjoint hyperplanes partitionning X,
- $\mathcal{B} = \{$affine lines L secant to each group of $\mathcal{G}\}$.

The code has:

- length $ns = q^m$,
- "locality" $n = q$.

rate M/N



length $N = ns = 2^{em}$

Question: better instances?

An **orthogonal array** $\mathrm{OA}(t, n, s)$ of strength t is a list A of words
- of length n,
- over a finite set S, $|S| = s$,
- such that, for every $I \subset [1, n]$ of size t, $A_{|I} = S^t$.

Equivalently, an $\mathrm{OA}(t, n, s)$ is a code $A \subset S^n$ with dual distance $t + 1$.

$$S = \{a, b\}$$

$$\mathrm{OA}(2,3,2) = \begin{bmatrix} a & b & b \\ b & b & a \\ b & a & b \\ a & a & a \end{bmatrix}$$

Instances with orthogonal arrays

An **orthogonal array** $\mathrm{OA}(t,n,s)$ of strength t is a list A of words

- of length n,
- over a finite set S, $|S| = s$,
- such that, for every $I \subset [1,n]$ of size t, $A_{|I} = S^t$.

Equivalently, an $\mathrm{OA}(t,n,s)$ is a code $A \subset S^n$ with dual distance $t+1$.

$S = \{a,b\}$

Construction OA → TD :

- $X = S \times [1,n]$
- $\mathcal{G} = \{S \times \{i\}, 1 \leq i \leq n\}$

$$\mathrm{OA}(2,3,2) = \begin{bmatrix} a & b & b \\ b & b & a \\ b & a & b \\ a & a & a \end{bmatrix}$$

$\boxed{\begin{matrix}(a,1) \\ (b,1)\end{matrix}}$ $\boxed{\begin{matrix}(a,2) \\ (b,2)\end{matrix}}$ $\boxed{\begin{matrix}(a,3) \\ (b,3)\end{matrix}}$

Instances with orthogonal arrays

An **orthogonal array** $\mathrm{OA}(t,n,s)$ of strength t is a list A of words

- of length n,
- over a finite set S, $|S| = s$,
- such that, for every $I \subset [1,n]$ of size t, $A_{|I} = S^t$.

Equivalently, an $\mathrm{OA}(t,n,s)$ is a code $A \subset S^n$ with dual distance $t+1$.

$S = \{a,b\}$

Construction OA $\to$ TD :

- $X = S \times [1,n]$
- $\mathcal{G} = \{S \times \{i\}, 1 \le i \le n\}$
- $\mathcal{B} = \{\{(c_i,i), 1 \le i \le n\}, \boldsymbol{c} \in \mathrm{OA}\}$

$$\mathrm{OA}(2,3,2) = \begin{bmatrix} a & b & b \\ b & b & a \\ b & a & b \\ a & a & a \end{bmatrix}$$

$$\boxed{\begin{matrix}(a,1)\\ \\(b,1)\end{matrix}} \quad \boxed{\begin{matrix}(a,2)\\ \\(b,2)\end{matrix}} \quad \boxed{\begin{matrix}(a,3)\\ \\(b,3)\end{matrix}}$$

Instances with orthogonal arrays

An **orthogonal array** $\mathrm{OA}(t, n, s)$ of strength t is a list A of words

- of length n,
- over a finite set S, $|S| = s$,
- such that, for every $I \subset [1, n]$ of size t, $A_{|I} = S^t$.

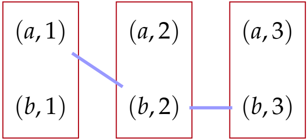
Equivalently, an $\mathrm{OA}(t, n, s)$ is a code $A \subset S^n$ with dual distance $t + 1$.

$S = \{a, b\}$

Construction OA → TD :

- $X = S \times [1, n]$
- $\mathcal{G} = \{S \times \{i\}, 1 \leq i \leq n\}$
- $\mathcal{B} = \{\{(c_i, i), 1 \leq i \leq n\}, \boldsymbol{c} \in \mathrm{OA}\}$

$$\mathrm{OA}(2,3,2) = \begin{bmatrix} a & b & b \\ b & b & a \\ b & a & b \\ a & a & a \end{bmatrix}$$



An **orthogonal array** $\mathrm{OA}(t,n,s)$ of strength t is a list A of words

- of length n,
- over a finite set S, $|S| = s$,
- such that, for every $I \subset [1,n]$ of size t, $A_{|I} = S^t$.

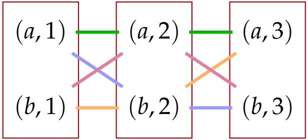
Equivalently, an $\mathrm{OA}(t,n,s)$ is a code $A \subset S^n$ with dual distance $t+1$.

$S = \{a,b\}$

Construction OA $\rightarrow$ TD :

- $X = S \times [1,n]$
- $\mathcal{G} = \{S \times \{i\}, 1 \leq i \leq n\}$
- $\mathcal{B} = \{\{(c_i, i), 1 \leq i \leq n\}, \boldsymbol{c} \in \mathrm{OA}\}$

$$\mathrm{OA}(2,3,2) = \begin{bmatrix} a & b & b \\ b & b & a \\ b & a & b \\ a & a & a \end{bmatrix}$$



Resisting collusions

Proposition. For $t = 2$, an $\mathrm{OA}(t, n, s)$ gives a $\mathrm{TD}(n, s)$.

Proposition. For $t = 2$, an $\mathrm{OA}(t, n, s)$ gives a $\mathrm{TD}(n, s)$.

Experimentally, for $t = 2$ and small n and s, codes based on classical affine TDs have the largest dimension.

Resisting collusions

Proposition. For $t = 2$, an $\mathrm{OA}(t, n, s)$ gives a $\mathrm{TD}(n, s)$.

Experimentally, for $t = 2$ and small n and s, codes based on classical affine TDs have the largest dimension.

For $t \geq 3$, we get TDs such that:

for every t-set T of points lying in t different groups,
there exists a unique block $B \in \mathcal{B}$ such that $T \subset B$.

$\Rightarrow$ The PIR protocol resists $t - 1$ colluding servers.

Resisting collusions

Proposition. For $t = 2$, an $\mathrm{OA}(t, n, s)$ gives a $\mathrm{TD}(n, s)$.

Experimentally, for $t = 2$ and small n and s, codes based on classical affine TDs have the largest dimension.

For $t \geq 3$, we get TDs such that:

for every t-set T of points lying in t different groups,
there exists a unique block $B \in \mathcal{B}$ such that $T \subset B$.

$\Rightarrow$ The PIR protocol resists $t - 1$ colluding servers.

- OAs with $t > 2$ exist (*e.g.* from Reed-Solomon codes)
- But associated TDs lead to codes with poor rates (except for $t \ll n$)

Private Information Retrieval from Transversal Designs. L.. IEEE-TIT. **2019**.

Outline

1. Private information retrieval

2. PIR schemes for common storage systems
- Distributed storage systems
- A PIR scheme on RS-coded databases
- A PIR scheme with regenerating codes

3. PIR schemes with low computation
- Transversal designs and codes
- A PIR scheme with transversal designs
- Instances

4. Conclusion

Conclusion

Private information retrieval:

- concentrated a lot of recent research,
- involves nice mathematical tools,
- but in practice ... relies on questionable assumptions (collusions, size of entries, communication channels)

Private information retrieval:

- concentrated a lot of recent research,
- involves nice mathematical tools,
- but in practice ... relies on questionable assumptions (collusions, size of entries, communication channels)

Questions?