Camilla Hollanti
Department of Mathematics
and Systems Analysis
Aalto University, Finland

Private Information Retrieval:

Chasing Capacity with Algebraic Codes

Simons Institute
March 8 2024

Institute for Advanced Study
Technical University of Munich

Aalto University
School of Science

Joint with/Thanks go to:
R.Freij-Hollanti, O.Gnilke, L.Holzbaur
D.Karpuk, J.Li, R.Tajeddine, A.Wachter-Zeh,...

Presentation outline

- Introduction to private information retrieval (PIR)
- PIR capacity: known results and our conjecture ($\approx$ theorem)
- Star product PIR
- Full support-rank PIR and that theorem
- Current and future directions (beyond PIR)

Collaborators

Ragnar
Aalto

Dave
WithSecure

Oliver
Aalborg

Razane
Helsinki

Lukas
Infineon

Jie
Huawei

Antonia
TU Munich








Private information retrieval (PIR)

- With PIR, a user is able to download one out of m files $\{x^1, \ldots, x^m\}$ from a database without revealing the identity $i \in [m]$ of the file to the database holder.

Theorem

If the data is stored on only one server, perfect privacy cannot be achieved except by downloading the entire data.

$\rightarrow$ Distributed storage system (DSS) with n servers...

$\rightarrow$ ...encoded with an $[n, k]$ (MDS) code!

Short history of PIR

- PIR from replicated databases was introduced by Chor in 1995 and was an active topic thereafter [Cho+95; Bei+02; Dvi+16].
  $\rightarrow$ A sequence of papers reduced the communication cost to be sub-linear in m.

Short history of PIR

- PIR from replicated databases was introduced by Chor in 1995 and was an active topic thereafter [Cho+95; Bei+02; Dvi+16].
  $\rightarrow$ A sequence of papers reduced the communication cost to be sub-linear in m.

- More recently, a lot of renewed interest towards PIR from coded storage systems from various perspectives [Aug+14; Sha+14; Faz+15].

Short history of PIR

- PIR from replicated databases was introduced by Chor in 1995 and was an active topic thereafter [Cho+95; Bei+02; Dvi+16].
  $\rightarrow$ A sequence of papers reduced the communication cost to be sub-linear in m.

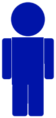
- More recently, a lot of renewed interest towards PIR from coded storage systems from various perspectives [Aug+14; Sha+14; Faz+15].

- 100s of papers since 2015; hits on IEEE Xplore:
  - 1995–2004: 7
  - 2005–2014: 72
  - 2015–2024: 393

A toy example with $[n,k]_q = [3,2]_2$ code

x_1^1 x_1^2 x_1^3	x_2^1 x_2^2 x_2^3	$x_1^1 + x_2^1$ $x_1^2 + x_2^2$ $x_1^3 + x_2^3$

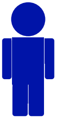
- $m = 3$ files. Want x^1. Server $j \in [n]$ stores $y_j^i = (x^i G)_j$, $i \in [n]$.



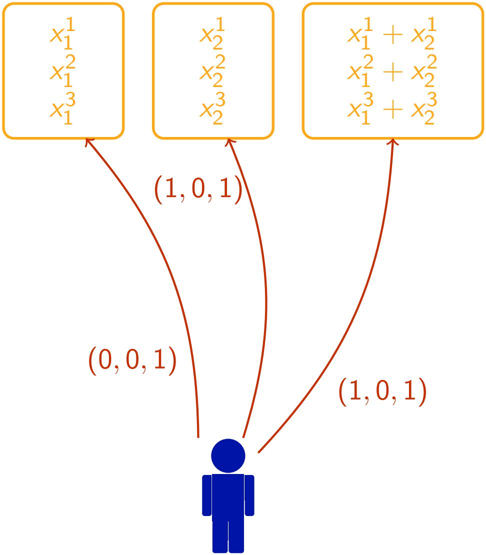
A toy example with $[n, k]_q = [3, 2]_2$ code

x_1^1	x_2^1	$x_1^1 + x_2^1$
x_1^2	x_2^2	$x_1^2 + x_2^2$
x_1^3	x_2^3	$x_1^3 + x_2^3$

- $m = 3$ files. Want x^1. Server $j \in [n]$ stores $y_j^i = (x^i G)_j$, $i \in [n]$.
- 1st round: Choose **random** $u = (1, 0, 1)$.

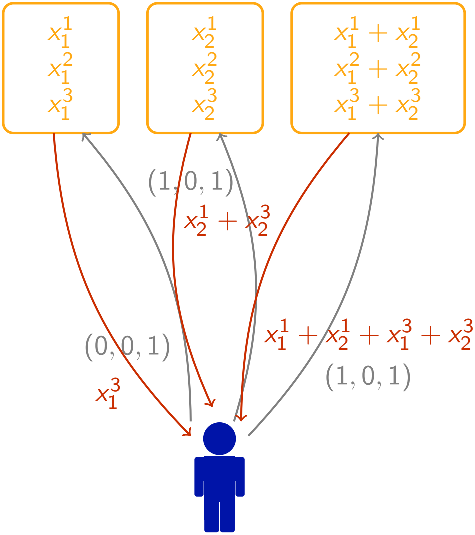


A toy example with $[n, k]_q = [3, 2]_2$ code



- $m = 3$ files. Want x^1. Server $j \in [n]$ stores $y_j^i = (x^i G)_j$, $i \in [n]$.
- 1st round: Choose **random** $u = (1, 0, 1)$.
- Send a query $q_1 = u + e_1 = (0, 0, 1)$ to the **1st** server, and $q_2 = q_3 = u = (1, 0, 1)$ to the other servers.

A toy example with $[n, k]_q = [3, 2]_2$ code

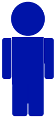


- $m = 3$ files. Want x^1. Server $j \in [n]$ stores $y_j^i = (x^i G)_j$, $i \in [n]$.
- 1st round: Choose **random** $u = (1, 0, 1)$.
- Send a query $q_1 = u + e_1 = (0, 0, 1)$ to the **1st** server, and $q_2 = q_3 = u = (1, 0, 1)$ to the other servers.
- Servers respond $r_j = \langle y_j, q_j \rangle$.
- Decode $x_1^1 = \sum_{j=1}^n r_j$.

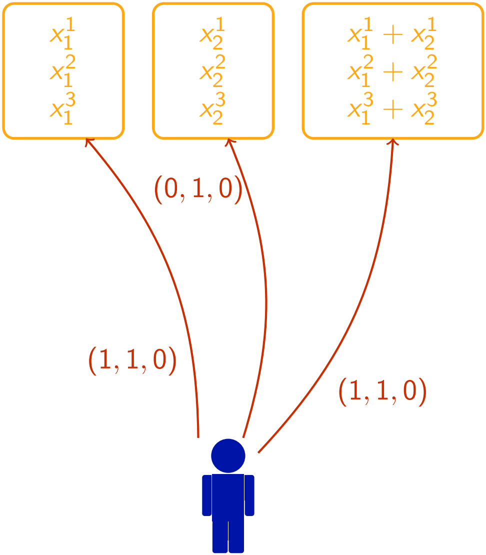
A toy example with $[n, k]_q = [3, 2]_2$ code

x_1^1	x_2^1	$x_1^1 + x_2^1$
x_1^2	x_2^2	$x_1^2 + x_2^2$
x_1^3	x_2^3	$x_1^3 + x_2^3$

- $m = 3$ files. Want x^1. Server $j \in [n]$ stores $y_j^i = (x^i G)_j$, $i \in [n]$.
- 2nd round: Choose **random** $u = (1, 1, 0)$.

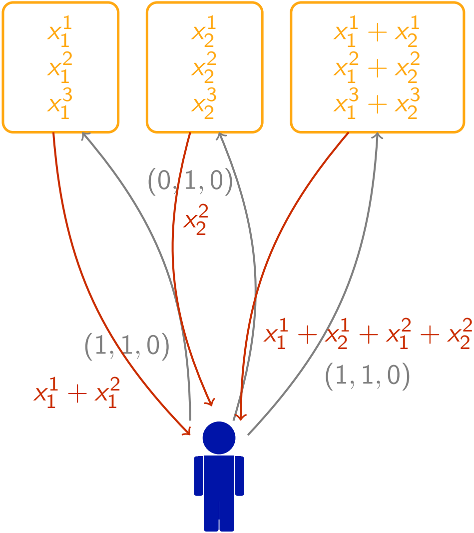


A toy example with $[n, k]_q = [3, 2]_2$ code



- $m = 3$ files. Want x^1. Server $j \in [n]$ stores $y_j^i = (x^i G)_j$, $i \in [n]$.
- 2nd round: Choose **random** $u = (1, 1, 0)$.
- $q_2 = u + e_1 = (0, 1, 0)$ to the **2nd** server, $q_1 = q_3 = u$.

A toy example with $[n,k]_q = [3,2]_2$ code



- $m = 3$ files. Want x^1. Server $j \in [n]$ stores $y_j^i = (x^i G)_j$, $i \in [n]$.
- 2nd round: Choose **random** $u = (1,1,0)$.
- $q_2 = u + e_1 = (0,1,0)$ to the **2nd** server, $q_1 = q_3 = u$.
- $r_j = \langle y_j, q_j \rangle$.
- $x_2^1 = \sum_{j=1}^n r_j$.
- Privacy holds if no *collusion*.

Collusion and t-PIR

Servers in a *colluding set* may exchange their obtained queries in order to reveal the identity of the desired file.

Definition

A PIR scheme *protects against* the colluding set $J \subseteq [n]$, if the projection of the overall query $\mathcal{Q}^i$ to J does not depend on the desired file $i \in [m]$.

Collusion and t-PIR

Servers in a *colluding set* may exchange their obtained queries in order to reveal the identity of the desired file.

Definition

A PIR scheme *protects against* the colluding set $J \subseteq [n]$, if the projection of the overall query $\mathcal{Q}^i$ to J does not depend on the desired file $i \in [m]$.

Definition

A *t-PIR scheme* protects against any colluding set of size $\leq t$.

Capacity of PIR

- The rate $\mathcal{R}$ of a PIR scheme is defined as

$$\mathcal{R} = \frac{\text{file size}}{\text{download size}}$$

Capacity of PIR

- The rate $\mathcal{R}$ of a PIR scheme is defined as

$$\mathcal{R} = \frac{\text{file size}}{\text{download size}}$$

- The capacity $\mathcal{C}$ of PIR is the maximum possible rate for a given model.

Capacity of PIR

- The rate $\mathcal{R}$ of a PIR scheme is defined as

$$\mathcal{R} = \frac{\text{file size}}{\text{download size}}$$

- The capacity $\mathcal{C}$ of PIR is the maximum possible rate for a given model.

- We call a scheme asymptotically capacity achieving if

$$\mathcal{R} = \lim_{m \to \infty} \mathcal{C}\,.$$

PIR capacity and constructions

- Several capacity results and scheme constructions have been reported (non-exhaustive list!):
  - replication [Sun+17; Tia+19]
  - MDS-coded storage [Ban+18; Zhu+19]
  - colluding servers [Sun+18b]
  - **MDS and colluding** [Fre+17; Taj+18; D'O+18]

PIR capacity and constructions

- Several capacity results and scheme constructions have been reported (non-exhaustive list!):
  - replication [Sun+17; Tia+19]
  - MDS-coded storage [Ban+18; Zhu+19]
  - colluding servers [Sun+18b]
  - **MDS and colluding** [Fre+17; Taj+18; D'O+18]

  - symmetric PIR (SPIR) [Wan+17b; Wan+17a; Wan+17c]
  - single-server PIR with side information [Hei+18]
  - non-MDS storage [Fre+19; Kum+19]
  - ...
  - graph-based PIR [Sad+23]

Capacity of PIR

- A storage system with m files has the following (conjectured, mostly proven) capacities:

	replication	$[n, k]$-coded
no collusion	† $\frac{1-1/n}{1-(1/n)^m} \xrightarrow{m\to\infty} 1-\frac{1}{n}$	¶ $\frac{1-k/n}{1-(k/n)^m} \xrightarrow{m\to\infty} 1-\frac{k}{n}$
t-collusion	‡ $\frac{1-t/n}{1-(t/n)^m} \xrightarrow{m\to\infty} 1-\frac{t}{n}$	§ $\frac{1-\frac{t+k-1}{n}}{1-(\frac{t+k-1}{n})^m} \xrightarrow{m\to\infty} 1-\frac{t+k-1}{n}$

† Sun–Jafar [Sun+17].

¶ Banawan–Ulukus [Ban+18].

‡ Sun–Jafar [Sun+18b].

§ Freij-Hollanti *et al.* [Fre+17]; Sun–Jafar [Sun+18a]; Holzbaur *et al.* [Hol+22].

Capacity of PIR

- A storage system with m files has the following (conjectured, mostly proven) capacities:

	replication	$[n,k]$-coded
no collusion	† $\frac{1-1/n}{1-(1/n)^m} \xrightarrow{m\to\infty} 1-\frac{1}{n}$	¶ $\frac{1-k/n}{1-(k/n)^m} \xrightarrow{m\to\infty} 1-\frac{k}{n}$
t-collusion	‡ $\frac{1-t/n}{1-(t/n)^m} \xrightarrow{m\to\infty} 1-\frac{t}{n}$	§ $\cancel{\frac{1-\frac{t+k-1}{n}}{1-(\frac{t+k-1}{n})^m}} \xrightarrow{m\to\infty} 1-\frac{t+k-1}{n}$

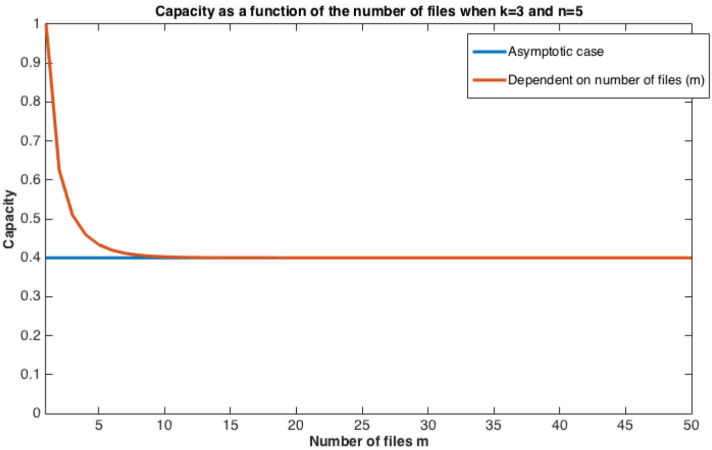
† Sun–Jafar [Sun+17].

¶ Banawan–Ulukus [Ban+18].

‡ Sun–Jafar [Sun+18b].

§ Freij-Hollanti *et al.* [Fre+17]; Sun–Jafar [Sun+18a]; Holzbaur *et al.* [Hol+22].

Fast convergence: coded case ($k > 1, t = 1$)



Capacity of uncoded PIR

Theorem (Sun–Jafar 2016)

The capacity of replicated n server PIR for a storage system containing m files is given by

$$\left(1+\frac{1}{n}+\cdots+\frac{1}{n^{m-1}}\right)^{-1}=\frac{1-\frac{1}{n}}{1-\frac{1}{n^m}}\xrightarrow{m\to\infty}1-\frac{1}{n}$$

Proof (sketch):

- Information theoretic argument to provide an upper bound.
- Scheme that achieves this bound.

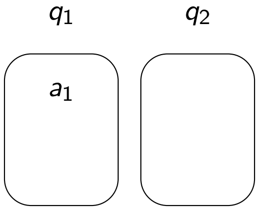
Sun–Jafar construction for replicated data

- $m = 2$ files a and b, stored on $n = 2$ servers via replication.
- "Sub-packetize" files into $n^m = 4$ (!!!) symbols a_1, a_2, a_3, a_4 and b_1, b_2, b_3, b_4. Assume a user wants a.
- A general retrieval strategy is as follows:

Sun–Jafar construction for replicated data

- $m = 2$ files a and b, stored on $n = 2$ servers via replication.
- "Sub-packetize" files into $n^m = 4$ (!!!) symbols a_1, a_2, a_3, a_4 and b_1, b_2, b_3, b_4. Assume a user wants a.
- A general retrieval strategy is as follows:

$\rightarrow$ Download a_1 from the 1st server.

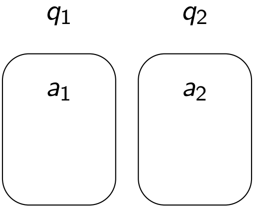


Sun–Jafar construction for replicated data

- $m = 2$ files a and b, stored on $n = 2$ servers via replication.
- "Sub-packetize" files into $n^m = 4$ (!!!) symbols a_1, a_2, a_3, a_4 and b_1, b_2, b_3, b_4. Assume a user wants a.
- A general retrieval strategy is as follows:

$\rightarrow$ Download a_1 from the 1st server.

$\rightarrow$ Symmetrize over both servers



Sun–Jafar construction for replicated data

- $m = 2$ files a and b, stored on $n = 2$ servers via replication.
- "Sub-packetize" files into $n^m = 4$ (!!!) symbols a_1, a_2, a_3, a_4 and b_1, b_2, b_3, b_4. Assume a user wants a.
- A general retrieval strategy is as follows:

$\rightarrow$ Download a_1 from the 1st server.

$\rightarrow$ Symmetrize over both servers

$\rightarrow$ Symmetrize over files.

q_1	q_2
a_1	a_2
b_1	b_2

Sun–Jafar construction for replicated data

- $m = 2$ files a and b, stored on $n = 2$ servers via replication.
- "Sub-packetize" files into $n^m = 4$ (!!!) symbols a_1, a_2, a_3, a_4 and b_1, b_2, b_3, b_4. Assume a user wants a.
- A general retrieval strategy is as follows:

→ Download a_1 from the 1st server.

→ Symmetrize over both servers

→ Symmetrize over files.

→ Use b_i as interference.

q_1	q_2
a_1	a_2
b_1	b_2
$a_3 + b_2$	

Sun–Jafar construction for replicated data

- $m = 2$ files a and b, stored on $n = 2$ servers via replication.
- "Sub-packetize" files into $n^m = 4$ (!!!) symbols a_1, a_2, a_3, a_4 and b_1, b_2, b_3, b_4. Assume a user wants a.
- A general retrieval strategy is as follows:

→ Download a_1 from the 1st server.

→ Symmetrize over both servers

→ Symmetrize over files.

→ Use b_i as interference.

→ Symmetrize again.

q_1	q_2
a_1	a_2
b_1	b_2
$a_3 + b_2$	$a_4 + b_1$

Sun–Jafar construction for replicated data

- $m = 2$ files a and b, stored on $n = 2$ servers via replication.
- "Sub-packetize" files into $n^m = 4$ (!!!) symbols a_1, a_2, a_3, a_4 and b_1, b_2, b_3, b_4. Assume a user wants a.
- A general retrieval strategy is as follows:

$\rightarrow$ Download a_1 from the 1st server.

$\rightarrow$ Symmetrize over both servers

$\rightarrow$ Symmetrize over files.

$\rightarrow$ Use b_i as interference.

$\rightarrow$ Symmetrize again.

$\rightarrow$ Random permutation provides privacy.

q_1	q_2
a_3	a_1
b_3	b_1
$a_4 + b_1$	$a_2 + b_3$

Sun–Jafar construction for replicated data

- $m = 2$ files a and b, stored on $n = 2$ servers via replication.
- "Sub-packetize" files into $n^m = 4$ (!!!) symbols a_1, a_2, a_3, a_4 and b_1, b_2, b_3, b_4. Assume a user wants a.
- A general retrieval strategy is as follows:

$\rightarrow$ Download a_1 from the 1st server.

$\rightarrow$ Symmetrize over both servers

$\rightarrow$ Symmetrize over files.

$\rightarrow$ Use b_i as interference.

$\rightarrow$ Symmetrize again.

$\rightarrow$ Random permutation provides privacy.

q_1	q_2
a_3	a_1
b_3	b_1
$a_4 + b_1$	$a_2 + b_3$

- Rate $= \frac{4}{6} = \frac{2}{3}$.

Proof of converse (sketch)

Theorem (Sun-Jafar, 2016)

The rate of any replicated PIR scheme on n servers with m files satisfies

$$\mathcal{R} = \frac{\textit{file size}}{\textit{download size}} \leq \frac{1 - \frac{1}{n}}{1 - \frac{1}{n^m}}.$$

Proof of converse (sketch)

Theorem (Sun-Jafar, 2016)

The rate of any replicated PIR scheme on n servers with m files satisfies

$$\mathcal{R} = \frac{\textit{file size}}{\textit{download size}} \leq \frac{1 - \frac{1}{n}}{1 - \frac{1}{n^m}}.$$

In other words,

$$\frac{\#\text{symbols downloaded}}{\#\text{desired symbols retrieved}} \geq 1 + \frac{1}{n} + \frac{1}{n^2} + \cdots + \frac{1}{n^{m-1}}$$

Proof of converse (sketch)

Theorem (Sun-Jafar, 2016)

The rate of any replicated PIR scheme on n servers with m files satisfies

$$\mathcal{R} = \frac{\textit{file size}}{\textit{download size}} \leq \frac{1 - \frac{1}{n}}{1 - \frac{1}{n^m}}.$$

In other words,

$$\frac{\#\text{symbols downloaded}}{\#\text{desired symbols retrieved}} \geq 1 + \frac{1}{n} + \frac{1}{n^2} + \cdots + \frac{1}{n^{m-1}}$$

File size $:= L$. *Claim:*

$$\#\text{symbols downloaded} \geq L + \frac{L}{n} + \frac{L}{n^2} + \cdots + \frac{L}{n^{m-1}}.$$

Proof of converse (sketch)

Proof idea (induction over m):

- Need to download $\geq L$ symbols of the file we want.
  Clear if $m = 1$.

Proof of converse (sketch)

Proof idea (induction over m):

- Need to download $\geq L$ symbols of the file we want.
  Clear if $m = 1$.

- Induction assumption (m files): We must download in total

$$D_m := L + \frac{L}{n} + \frac{L}{n^2} + \cdots + \frac{L}{n^{m-1}}$$

  symbols from the files $1, \ldots, m$.

Proof of converse (sketch)

Proof idea (induction over m):

- Need to download $\geq L$ symbols of the file we want.
  Clear if $m = 1$.

- Induction assumption (m files): We must download in total

$$D_m := L + \frac{L}{n} + \frac{L}{n^2} + \cdots + \frac{L}{n^{m-1}}$$

  symbols from the files $1, \ldots, m$.

- From some server, need

$$\geq \frac{D_m}{n} = \frac{1}{n}\left(L + \frac{L}{n} + \frac{L}{n^2} + \cdots + \frac{L}{n^{m-1}}\right)$$

  symbols from the files $1, \ldots, m$.

Proof of converse (sketch)

- Induction step: Assume we want file $m+1$.
- Then by induction assumption, from some server need $\frac{D_m}{n}$ symbols from the files $1, \ldots, m$.

Proof of converse (sketch)

- Induction step: Assume we want file $m+1$.
- Then by induction assumption, from some server need $\frac{D_m}{n}$ symbols from the files $1, \ldots, m$.
- In addition, need to download L symbols from file $m+1$.
- Total download:

$$L + \frac{D_m}{n} = L + \left(\frac{L}{n} + \frac{L}{n^2} + \frac{L}{n^3} + \cdots + \frac{L}{n^m}\right) = D_{m+1}$$

$\square$

The proofs for the coded storage case and for the colluding case are similar. Combine to get **coded AND colluding** case? Hard!

Conjecture for PIR capacity with $t > 1, k > 1$

Conjecture [Fre+17, Conj. 1] Let C be a linear $[n, k, d]$ code. Consider m files and let $1 \leq t \leq n-k$. Any t-PIR scheme has rate

$$\mathcal{R} \leq \frac{1 - \frac{k+t-1}{n}}{1 - (\frac{k+t-1}{n})^m} \overset{m\to\infty}{\longrightarrow} 1 - \frac{k+t-1}{n} \ .$$

Conjecture for PIR capacity with $t > 1, k > 1$

Conjecture [Fre+17, Conj. 1] Let C be a linear $[n, k, d]$ code. Consider m files and let $1 \leq t \leq n - k$. Any t-PIR scheme has rate

$$\mathcal{R} \leq \frac{1 - \frac{k+t-1}{n}}{1 - (\frac{k+t-1}{n})^m} \xrightarrow{m \to \infty} 1 - \frac{k+t-1}{n}.$$

- Disproved in [Sun+18a] for $m = 2$, $k = t = 2$, $n = 4$. The PIR rate is $3/5$, while the conjecture states $4/7$.

Conjecture for PIR capacity with $t > 1, k > 1$

Conjecture [Fre+17, Conj. 1] Let C be a linear $[n, k, d]$ code. Consider m files and let $1 \leq t \leq n - k$. Any t-PIR scheme has rate

$$\mathcal{R} \leq \frac{1 - \frac{k+t-1}{n}}{1 - (\frac{k+t-1}{n})^m} \xrightarrow{m\to\infty} 1 - \frac{k+t-1}{n} \ .$$

- Disproved in [Sun+18a] for $m = 2$, $k = t = 2$, $n = 4$. The PIR rate is $3/5$, while the conjecture states $4/7$.
- The query scheme in the counter-example is not full support-rank!
- Proved for full support-rank schemes in [Hol+22].
- *How*, and what is "full support-rank"?

Codes from star products

- For two vectors $x, y \in \mathbb{F}_q^n$, define the *star product*

$$x \star y := (x_1 y_1, \ldots, x_n y_n) \, .$$

- Let C and D be linear codes in $\mathbb{F}_q^n$. Define the *star product code* as the linear span

$$C \star D := \langle \{ c \star d | c \in C, d \in D \} \rangle \, .$$

[†] Mirandola–Zémor [Mir+15]: Apart from pairs $C, C^\perp$ and their products, the only pairs that get to this bound are *generalized Reed–Solomon (GRS) codes*.

Codes from star products

- For two vectors $x, y \in \mathbb{F}_q^n$, define the *star product*

$$x \star y := (x_1 y_1, \ldots, x_n y_n) .$$

- Let C and D be linear codes in $\mathbb{F}_q^n$. Define the *star product code* as the linear span

$$C \star D := \langle \{ c \star d | c \in C, d \in D \} \rangle .$$

- *Product Singleton Bound*[†]:

$$d_{C \star D} \leq n - \dim(C) - \dim(D) + 2$$

[†] Mirandola–Zémor [Mir+15]: Apart from pairs $C, C^{\perp}$ and their products, the only pairs that get to this bound are *generalized Reed–Solomon (GRS) codes*.

Star-product PIR scheme

- We proposed a fully general coded retrieval scheme protecting against t-collusion [Fre+17][†].
- Asymptotically capacity achieving at the known points ($k = 1$, $t = 1$), when employed with GRS codes.
- Also achieves the (asymptotic) capacity of the above conjecture.

[†] You can find a couple of extra slides for details after the thank-you slide.

Star-product PIR scheme

- We proposed a fully general coded retrieval scheme protecting against t-collusion [Fre+17][†].
- Asymptotically capacity achieving at the known points ($k = 1$, $t = 1$), when employed with GRS codes.
- Also achieves the (asymptotic) capacity of the above conjecture.
- **Novelty:**
  - *Earlier:* n queries from the entire space $\mathbb{F}_q^m$.
  - *Star product scheme:* m queries from an $[n, t]$ code $D \subseteq \mathbb{F}_q^n$. $\rightarrow$ smart (star-product) interplay of the $[n, k]$ storage code C and the query code D.

[†] You can find a couple of extra slides for details after the thank-you slide.

Rate vs. capacity

- What the user receives is a codeword in $C \star D$ with errors in known positions.
- These errors can be treated as erasures and we know that the code $C \star D$ can correct up to $d_{C \star D} - 1$ erasures.

Rate vs. capacity

- What the user receives is a codeword in $C \star D$ with errors in known positions.
- These errors can be treated as erasures and we know that the code $C \star D$ can correct up to $d_{C\star D} - 1$ erasures.
- From the *product singleton bound* we know that

$$d_{C\star D} \leq n - \dim(C) - \dim(D) + 2$$

Rate vs. capacity

- What the user receives is a codeword in $C \star D$ with errors in known positions.
- These errors can be treated as erasures and we know that the code $C \star D$ can correct up to $d_{C\star D} - 1$ erasures.
- From the *product singleton bound* we know that

$$d_{C\star D} \leq n - k - t + 2$$

Rate vs. capacity

- What the user receives is a codeword in $C \star D$ with errors in known positions.
- These errors can be treated as erasures and we know that the code $C \star D$ can correct up to $d_{C\star D} - 1$ erasures.
- From the *product singleton bound* we know that

$$d_{C\star D} \leq n - k - t + 2$$

- For GRS pairs C, D this gives us a rate of

$$\mathcal{R} = \frac{n-k-t+1}{n} = 1 - \frac{k+t-1}{n}$$

Rate vs. capacity

- What the user receives is a codeword in $C \star D$ with errors in known positions.
- These errors can be treated as erasures and we know that the code $C \star D$ can correct up to $d_{C\star D} - 1$ erasures.
- From the *product singleton bound* we know that

$$d_{C\star D} \leq n - k - t + 2$$

- For GRS pairs C, D this gives us a rate of

$$\mathcal{R} = \frac{n - k - t + 1}{n} = 1 - \frac{k + t - 1}{n}$$

- Asymptotically capacity (and conjecture) achieving:

Capacity of $\star$-product schemes

Any "strongly linear" scheme can be replaced by a star product scheme for the same privacy model, without losing in the PIR rate:

Theorem ([Hol+22])

Consider a strongly linear PIR scheme from a storage code C and a query scheme as above. Then the rate is bounded by

$$\mathcal{R} \leq 1 - \frac{k+t-1}{n}$$

for any number of files m.

This bound coincides with the asymptotic capacity conjecture, which is achieved for any number of files with star product PIR.

Full support-rank PIR

- Clearly, it is suboptimal to send linearly dependent queries to servers.
- However, submatrices of the query matrix may be dependent [Sun+18a], *i.e.*, have supported columns that are linearly dependent.
- The technical assumption of full support-rank restricts all supported columns $\mathcal{T}$, $|\mathcal{T}| \leq t$, to be independent:

Definition

A linear PIR scheme is of *full support-rank* if for every query realization $q \in \mathbb{F}^{\alpha m \times \beta n}$, any subset $\mathcal{T} \subseteq [n]$ of $|\mathcal{T}| \leq t$ servers, and any file index $i \in [m]$

$$\mathrm{rank}(q[\psi_\alpha(i), \psi_\beta(\mathcal{T})]) = |\,\mathsf{colsupp}(q[\psi_\alpha(i), \psi_\beta(\mathcal{T})])|.$$

Capacity of full support-rank PIR

Theorem ([Hol+22])

The capacity of full support-rank linear PIR from $[n, k]$-MDS coded storage with t colluding servers is

$$\mathcal{C} = \frac{1 - \frac{k+t-1}{n}}{1 - \left(\frac{k+t-1}{n}\right)^m} \xrightarrow{m\to\infty} 1 - \frac{k+t-1}{n}.$$

- The converse follows the converse proof for the symmetric case [Wan+17a] with some additional lemmas.
- A capacity-achieving scheme can be constructed from [Fre+17; D'O+18].

Capacity of full support-rank PIR

Theorem ([Hol+22])

The capacity of full support-rank linear PIR from $[n, k]$-MDS coded storage with t colluding servers is

$$\mathcal{C} = \frac{1 - \frac{k+t-1}{n}}{1 - \left(\frac{k+t-1}{n}\right)^m} \xrightarrow{m \to \infty} 1 - \frac{k+t-1}{n}.$$

- The converse follows the converse proof for the symmetric case [Wan+17a] with some additional lemmas.
- A capacity-achieving scheme can be constructed from [Fre+17; D'O+18].
- The proof settles the earlier conjecture for linear PIR schemes for full support-rank schemes, which seems to cover *almost* everything.

How to interpret the above result?

- Our definition of full support-rank PIR captures the linear independency of the queries that all general capacity-achieving schemes have in common.
- In order to exceed the conjectured capacity, it is *necessary* for some restrictions of the queries to subsets of t servers to be linearly dependent.

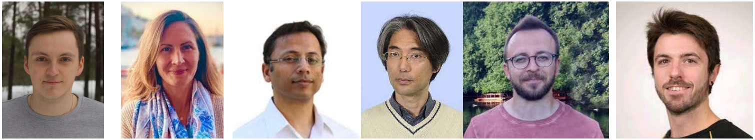
How to interpret the above result?

- Our definition of full support-rank PIR captures the linear independency of the queries that all general capacity-achieving schemes have in common.
- In order to exceed the conjectured capacity, it is *necessary* for some restrictions of the queries to subsets of t servers to be linearly dependent.
- It is exactly this property that allows the scheme of [Sun+18a], which is *not* of full support-rank, to exceed the full support-rank capacity.
- It seems difficult to extend the counter-example for $m > 2$ while maintaining a good rate.

Conclusion

- Intro to star product PIR from coded storage.
- Overview of capacity results.
- Strongly linear and full support-rank PIR capacity (almost) proving earlier conjectures.
- Importance of star product schemes in terms of practical implementation: small field sizes and low sub-packetization.
- Highly generalizable
  $\rightarrow$ stragglers, adversaries, networks, streaming, distributed computation, interference alignment, quantum,...

Beyond PIR: current and future directions



Okko Elif Syed Masahito Tefjol Matteo

- Secure (analog) distributed matrix multiplication (SDMM): Okko Makkonen
- Cross-subspace alignment codes (CSA) for PIR and SDMM: cf. Jafar et al.
- Generalizations of the above using algebraic geometry codes: Okko, Dave, Elif Sacikara (+Gretchen)
- Quantum PIR: Matteo Allaix, Lukas, Tefjol Pllaha, Masahito Hayashi+group, Syed Jafar+group

References I

[Aug+14] D. Augot, F. Levy-Dit-Vehel, and A. Shikfa, “A storage-efficient and robust private information retrieval scheme allowing few servers”, in *Cryptology and Network Security*, Springer, 2014, pp. 222–239.

[Ban+18] K. Banawan and S. Ulukus, “The capacity of private information retrieval from coded databases”, *IEEE Transactions on Information Theory*, vol. 64, no. 3, pp. 1945–1956, 2018.

[Bei+02] A. Beimel, Y. Ishai, E. Kushilevitz, and J.-F. Raymond, “Breaking the $O(n^{1/(2k-1)})$ barrier for information-theoretic private information retrieval”, in *The 43rd Annual IEEE Symposium on Foundations of Computer Science, 2002. Proceedings.*, IEEE, 2002, pp. 261–270.

[Cho+95] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan, “Private information retrieval”, in *Proceedings of IEEE 36th Annual Foundations of Computer Science*, IEEE, 1995, pp. 41–50.

References II

[D'O+19] R. G. D'Oliveira and S. El Rouayheb, "One-shot PIR: Refinement and lifting", *IEEE Transactions on Information Theory*, vol. 66, no. 4, pp. 2443–2455, 2019.

[Dvi+16] Z. Dvir and S. Gopi, "2-server PIR with sub-polynomial communication", *Journal of the ACM (JACM)*, vol. 63, no. 4, 2016.

[Faz+15] A. Fazeli, A. Vardy, and E. Yaakobi, "Codes for distributed PIR with low storage overhead", in *Information Theory (ISIT), 2015 IEEE International Symposium on*, 2015, pp. 2852–2856.

[Hol+22] L. Holzbaur, R. Freij-Hollanti, J. Li, and C. Hollanti, "Toward the capacity of private information retrieval from coded and colluding servers", *IEEE Transactions on Information Theory*, 2022.

[Mir+15] D. Mirandola and G. Zémor, "Critical pairs for the product Singleton bound", *IEEE Transactions on Information Theory*, vol. 61, no. 9, pp. 4928–4937, 2015.

References III

[Sad+23] B. Sadeh, Y. Gu, and I. Tamo, “Bounds on the capacity of private information retrieval over graphs”, *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 261–273, 2023.

[Sha+14] N. B. Shah, K. V. Rashmi, and K. Ramchandran, “One extra bit of download ensures perfectly private information retrieval”, in *2014 IEEE International Symposium on Information Theory*, 2014.

[Sun+17] H. Sun and S. A. Jafar, “The capacity of private information retrieval”, *IEEE Transactions on Information Theory*, vol. 63, no. 7, pp. 4075–4088, 2017.

[Taj+18] R. Tajeddine, O. W. Gnilke, and S. El Rouayheb, “Private information retrieval from MDS coded data in distributed storage systems”, *IEEE Transactions on Information Theory*, vol. 64, no. 11, pp. 7081–7093, 2018.

References IV

[Taj+19] R. Tajeddine, O. W. Gnilke, D. Karpuk, R. Freij-Hollanti, and C. Hollanti, "Private information retrieval from coded storage systems with colluding, Byzantine, and unresponsive servers", *IEEE Transactions on Information Theory*, vol. 65, no. 6, pp. 3898–3906, 2019.

[Tia+19] C. Tian, H. Sun, and J. Chen, "Capacity-achieving private information retrieval codes with optimal message size and upload cost", *IEEE Transactions on Information Theory*, vol. 65, no. 11, pp. 7613–7627, 2019.

[Zhu+19] J. Zhu, Q. Yan, C. Qi, and X. Tang, "A new capacity-achieving private information retrieval scheme with (almost) optimal file length for coded servers", *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1248–1260, 2019.

[D'O+18] R. G. L. D'Oliveira and S. E. Rouayheb, "Lifting private information retrieval from two to any number of messages", in *2018 IEEE International Symposium on Information Theory (ISIT)*, 2018, pp. 1744–1748.

References V

[Fre+17] R. Freij-Hollanti, O. W. Gnilke, C. Hollanti, and D. A. Karpuk, "Private information retrieval from coded databases with colluding servers", *SIAM Journal on Applied Algebra and Geometry*, vol. 1, no. 1, pp. 647–664, 2017.

[Fre+19] R. Freij-Hollanti, O. W. Gnilke, C. Hollanti, A. Horlemann-Trautmann, D. Karpuk, and I. Kubjas, "t-private information retrieval schemes using transitive codes", *IEEE Transactions on Information Theory*, vol. 65, no. 4, pp. 2107–2118, 2019.

[Hei+18] A. Heidarzadeh, F. Kazemi, and A. Sprintson, "Capacity of single-server single-message private information retrieval with coded side information", in *2018 IEEE Information Theory Workshop (ITW)*, 2018, pp. 1–5.

[Kum+19] S. Kumar, H. Lin, E. Rosnes, and A. G. i. Amat, "Achieving maximum distance separable private information retrieval capacity with linear codes", *IEEE Transactions on Information Theory*, pp. 1–1, 2019.

References VI

[Sun+18a] H. Sun and S. A. Jafar, "Private information retrieval from MDS coded data with colluding servers: Settling a conjecture by Freij-Hollanti et al.", *IEEE Transactions on Information Theory*, vol. 64, no. 2, pp. 1000–1022, 2018.

[Sun+18b] ——, "The capacity of robust private information retrieval with colluding databases", *IEEE Transactions on Information Theory*, vol. 64, no. 4, pp. 2361–2370, 2018.

[Wan+17a] Q. Wang and M. Skoglund, "Linear symmetric private information retrieval for MDS coded distributed storage with colluding servers", in *2017 IEEE Information Theory Workshop (ITW)*, 2017, pp. 71–75.

[Wan+17b] ——, "Secure symmetric private information retrieval from colluding databases with adversaries", in *2017 55th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2017, pp. 1083–1090.

[Wan+17c] ——, "Symmetric private information retrieval for MDS coded distributed storage", in *2017 IEEE International Conference on Communications (ICC)*, 2017, pp. 1–6.


THANKS!!
Aalto University
School of Science

Linear and strongly linear (SL) PIR

Definition

A PIR scheme is *linear* if the responses are given by $A_j^i = \left\langle Q_j^i, Y_j \right\rangle, \forall j \in [n]$.

Definition

A linear PIR scheme is *strongly linear* (SL) if each symbol of the desired file x^i is obtained as a deterministic linear function over $\mathbb{F}_q$ of the response vector $(A_1^i, \ldots A_n^i)$, not depending on the randomness used to produce the queries.

- Strong linearity is important in practice, since it allows for **small field size** and **low sub-packetization level** $O(k(n-k))$ (in contrast to $O(n^m)$).

PIR codes from star products

- Consider m files $x^1, \ldots, x^m$.

$$\begin{array}{cc} x^1 & \\ & \\ x^m & \end{array} \begin{pmatrix} \boxed{x_1^1 \quad \ldots \quad x_k^1} \\ \vdots \quad \ddots \quad \vdots \\ \boxed{x_1^m \quad \ldots \quad x_k^m} \end{pmatrix}$$

PIR codes from star products

- Consider m files $x^1, \ldots, x^m$.
- We encode this data using an $[n, k, d_C]$ storage code C with generator matrix G_C and store it on n servers.

$$\begin{array}{c} x^1 \\ \\ x^m \end{array} \begin{pmatrix} \boxed{x_1^1 \quad \ldots \quad x_k^1} \\ \vdots \quad \ddots \quad \vdots \\ \boxed{x_1^m \quad \ldots \quad x_k^m} \end{pmatrix} \cdot G_C = \begin{array}{ccc} \text{Server 1} & & \text{Server n} \\ \left(\boxed{\begin{matrix} y_1^1 \\ \vdots \\ y_1^m \end{matrix}} \right. & \begin{matrix} \ldots \\ \ddots \\ \ldots \end{matrix} & \left. \boxed{\begin{matrix} y_n^1 \\ \vdots \\ y_n^m \end{matrix}} \right) \end{array}$$

PIR codes from star products

- Consider m files $x^1, \ldots, x^m$.
- We encode this data using an $[n, k, d_C]$ storage code C with generator matrix G_C and store it on n servers.

$$
\begin{array}{c} x^1 \\ \\ x^m \end{array}
\begin{pmatrix} \boxed{x_1^1 \quad \ldots \quad x_k^1} \\ \vdots \quad \ddots \quad \vdots \\ \boxed{x_1^m \quad \ldots \quad x_k^m} \end{pmatrix} \cdot G_C =
\begin{array}{ccc} \text{Server 1} & & \text{Server n} \\ \left(\boxed{\begin{matrix} y_1^1 \\ \vdots \\ y_1^m \end{matrix}} \right. & \begin{matrix} \ldots \\ \ddots \\ \ldots \end{matrix} & \left. \boxed{\begin{matrix} y_n^1 \\ \vdots \\ y_n^m \end{matrix}} \right) \end{array}
$$

- Protects against failure of up to $d_C - 1$ servers.