PRIVATE INFORMATION RETRIEVAL

Benny Chor
Oded Goldreich
Eyal Kushilevitz
Madhu Sudan
THE PROBLEM/MODEL

- \( k \) (replicated) databases
  each holding the same \( \Pi \) items

- Privacy: no single database should know \( i \)

- Cost: communication
  (between user & databases)

\( k=2 \)
SOLUTIONS — PIR SCHEMES

• For $K=2$ DATABASES: $\text{Cost} = 12 \cdot \sqrt[3]{n}$

• $\forall K \geq 2$ DATABASES: $\text{Cost} = O(\sqrt{\frac{k}{n}})$

• For $K = \frac{1}{3} \log_2 n$, $\text{Cost} = \frac{1}{3} \log_2^2 n \cdot \log \log_2 n$

THE TRIVIAL SOLUTION ALLOW $K=1$

BUT REQUIRES $\text{Cost} = n$. 
A RELATED MODEL - "INSTANCE HIDING"

- $k$ "POWERFUL" COMPUTERS

---

**USER:** INSTANCE $i \in \{0,1\}^k$

WANTS $f(i)$, WHERE $f$ IS HARD TO COMPUTE (FOR $U$).

---

- **PRIVACY:** AS BEFORE
- **COST:** USER OPERATES IN $\text{POLY}(|i|)$-TIME.

---

**RELATION TO PIR**

\[
\begin{align*}
[n] & \quad \rightarrow \quad 2^k,13^k \equiv [2^k] \\
\text{DATABASE} & \quad \rightarrow \quad f(1), f(2), \ldots, f(2^k)
\end{align*}
\]

---

SUMMARY: PIR = $\#H$ scaled down $+$ change of focus/complexity

---

IN PIR = $\text{POLY}$

IN I.H. $= \text{POLY}$
A RELATED MODEL - "INSTANCE HIDING"

- k "POWERFUL" COMPUTERS

\[ \text{[AFK]}, k=1, \text{NEG.} \]
\[ \text{[BF]}, k=\log_2 n \]
\[ \text{[BFR]} \forall k \geq 2 \]

USER: INSTANCE \( i \in \{0,1\}^l \)
WANTS \( f(i) \), WHERE \( f \)
IS HARD TO COMPUTE (FOR \( U \)).

- PRIVACY: AS BEFORE
- COST: USER OPERATES IN \( \text{POLY}(|i|) \)-TIME.

RELATION TO PIR

\[ \begin{align*}
[\text{n}] & \quad \rightarrow \quad [30,13^l] \equiv [2^l] \\
\text{DATABASE} & \quad \rightarrow \quad f(1), f(2), \ldots, f(2^l)
\end{align*} \]

\[ \Rightarrow \text{SUMMARY: PIR = \# scaled down} \]
\[ \text{+ change of focus/complexity} \]
A SIMPLE PIR FOR $k=2$

$\text{DATABASE} = x \in \mathbb{Z}_3^n$

$\text{DESIRED ITEM} \quad i \in [n]$

$\text{USER SELECTS UNIFORMLY } s \in [n]$

- SENDS $s$ to $\text{DATABASE}_1$
- SENDS $s \oplus x_i$ to $\text{DATABASE}_2$

$\text{DATABASE}$, UPON RECEIVING $r \in [n]$, RETURNS $x_i \oplus r$

$\text{USER XORs THE 2 BITS.}$

- $\text{PRIVACY: OK.}$
- $\text{COST: LINEAR IN } n$

NO BETTER THAN "TRIVIAL".

So why did I waste your time? ...
because this idea can be generalized to something useful.
Generalization to $k=2^d$ (e.g., $d=3$)

View $x = x_1, \ldots, x_n$ as residing in a cube:

$$m = \sqrt[n-1]{n}$$

$$[n] = [m] \times [m] \times [m]$$

\[
\text{WANTED} \quad x_i, c_j, e_k
\]

\[
\text{USER SELECTS UNIFORMLY} \quad S_1, S_2, S_3 \in [m]
\]

- Sends $S_1 \oplus 2^i c_j \oplus 3^e k$, $S_2 \oplus 2^i c_j \oplus 3^e k$, $S_3 \oplus 2^i c_j \oplus 3^e k$

  to database 'number' $\mathcal{D}_{1, 2, 3} \in \{0, 1\}^3$.

Database, upon receiving $R_1, R_2, R_3 \in [m]$,

- returns $\oplus \oplus \oplus x_i, j, k$,

  \[
  j \in \mathcal{R}_1, k \in \mathcal{R}_2, l \in \mathcal{R}_3
  \]

User XORs the 8 bits

- Privacy: OK. 😊

- Cost: $O(m) = O(\sqrt[3]{n})$. 😊

But this is with 8 DATABASES
And I've promised such performance with 2.
A CLOSER LOOK AT $\binom{3}{n}$ SOLUTION

COMMUNICATION BETWEEN USER & DATABASE

USER \[\begin{array}{c}
3 \text{ SUBSETS} \\
= \frac{3 \cdot m \text{ BITS}}{}
\end{array}\] Database

\[\begin{array}{c}
1 \text{ (XOR) BIT} \\
\end{array}\]

- Suppose $DB_{000}$ receives $(R_1, R_2, R_3)$.
- It knows that $DB_{100}$ has received one of the following $(R_{1j} \oplus 3, j, R_2, R_3)$, $j = 1, \ldots, m$.
- It can "simulate" $DB_{100}$ by replying with all $m$ possibilities. ($m$ bits!)

$\Rightarrow$ $DB_{000}$ simulates $DB_{000}, DB_{010}, DB_{001}$
$\Rightarrow$ $DB_{111}$ simulates $DB_{011}, DB_{101}, DB_{110}$

$\Rightarrow$ 2 database scheme with

Privacy: Ok.
Cost: $O(\frac{3}{\sqrt{n}})$. 😊
IN GENERAL \( (d \geq 3) \)

- A PIR for \( 2^d \) databases with communication cost:
  \[
  \text{User} \quad \frac{d}{\sqrt{n}} \quad \text{subsets of } \binom{n}{\sqrt{n}}
  \]
  \[
  \text{Database} \quad d \cdot \sqrt{n} \quad \text{bits}
  \]
  \[
  \quad \quad \quad \quad \quad \quad \quad \quad 1 \ (\text{XOR}) \ \text{bit}
  \]

- A covering code (of radius 1) for \( 3^d \) using \( k \) codewords:
  \[
  \left( \frac{2^d}{d+1} \leq k \leq 2^d \right) \quad \text{"volume bound"}
  \]

\[\Rightarrow\] A PIR for \( k \) databases with communication cost \( \propto 2^d \cdot d \cdot \sqrt{n} \).

<table>
<thead>
<tr>
<th>( d )</th>
<th>( 2^d )</th>
<th>( k )</th>
<th>Volume Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>8</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>16</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>32</td>
<td>7</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>64</td>
<td>12</td>
<td>10</td>
</tr>
<tr>
<td>7</td>
<td>128</td>
<td>16</td>
<td>16</td>
</tr>
</tbody>
</table>
Covering Codes (of Rad' 1)

C \subseteq 013^d \text{ is a covering code for } 3013^d \text{ if }
\forall x \in 3013^d \exists y \in C \text{ s.t. } \text{dist}(x, y) \leq 1.

Hamming distance.

E.g. \{000, 111\} is a covering code for 3013^3.

Volume bound \implies \text{each codeword covers } d+1 \text{ strings}

\implies \#\text{codewords} \geq \frac{2^d}{d+1}

Equality for perfect cover.
Polynomial Interpolation PIRs

\[ x \in \mathbb{R}^n \implies X : [n] \to \mathbb{R}_0^* \]

**APPLICATION**: \([BF]\)

\[ \Pi_{i_1, \ldots, i_q} \in \mathcal{F} \quad (q \in \{1, \text{m}3\}) \]

\[ f(t) \equiv X(i_1 + t \cdot i_{r_1}, \ldots, i_q + t \cdot i_{r_q}) \]

where \((i_1, \ldots, i_q)\) is the item sought.

Obtain from \(j^{th}\) database the value \(f(j)\)

Interpolate to obtain \(f(0) = X(i_1, \ldots, i_q)\).
Polynomial Interpolation PIRs

\[ x \in \mathbb{Z}^{m} \rightarrow x : [n] \rightarrow \mathbb{Z}^{m} \]

\[ x : (\mathbb{F}_{d}^{m}) \rightarrow \mathbb{Z}^{m}, \quad m \geq \frac{d}{\sqrt{n}} \]

\[ \hat{x} : \mathbb{F}^{m} \rightarrow \mathbb{F} \quad \text{degree} \ d \quad \text{multi-Lin' Ext'} \]

I.E., \( \hat{x}(z_{1}, \ldots, z_{m}) \equiv \sum_{\sigma \in \sigma(n)} (T_{1} z_{1} \cdots T_{d} z_{d}) \cdot x(s) \)

Application: \[ [BF] \]

\[ \pi_{1}, \ldots, \pi_{q} \in \mathbb{F} \]

\[ \ell(t) \equiv \hat{x}(i_{1} + t \cdot \pi_{1}, \ldots, i_{q} + t \cdot \pi_{q}) \]

where \( (i_{1}, \ldots, i_{q}) \) is the item sought.

obtain from \( j^{th} \) database the value \( \ell(j) \)

interpolate to obtain \( \ell(0) = \hat{x}(i_{1}, \ldots, i_{q}) \).
OPEN PROBLEM

BEST PIR SCHEMES

<table>
<thead>
<tr>
<th>#DB</th>
<th>COST</th>
<th>CONJ.</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>$O(\frac{3}{2}n)$</td>
<td>$L(\frac{3}{2}n)$</td>
</tr>
<tr>
<td>$k&gt;2$</td>
<td>$O(\frac{k}{2}n)$</td>
<td>$L(\frac{k}{2}n)$</td>
</tr>
</tbody>
</table>

"BEST" LOWER BOUND

For $k=2$, if user is only allowed a single boolean query, then $|query| = \Omega(n)$. 
ADDENDUM (JULY 2008)

The conjectured lowerbound for multiple-server PIRs (i.e., the case of $k > 2$) was disproved a couple of years afterwards by Ambainis in his paper ``An Upper Bound On The Communication Complexity of Private Information Retrieval'' (24th ICALP, LNCS 1256, pages 401--407, 1997). Ambainis established an upper bound of $n^{1/(2k-1)}$, which became my revised conjecture for a tight result.

This conjecture was disproved by Beimel, Ishai, Kushilevitz, and Raymond in their paper ``Breaking the $O(n^{1/(2k-1)})$ barrier for information-theoretic private information retrieval'' (43rd FOCS, pages 261--270, 2002).
My last attempt at a conjecture was that for every \( k \) there exists a constant \( c = c(k) \) such that a \( k \)-server PIR requires communication complexity \( n^c \). Furthermore, I expected \( c(k) \) to equal the reciprocal of some small polynomial.

This last conjecture seems to be disproven by Yekhanin in his paper "Towards 3-Query Locally Decodable Codes of Subexponential Length" (39th STOC, pages 266--274, 2007).

I dare not make further conjectures....