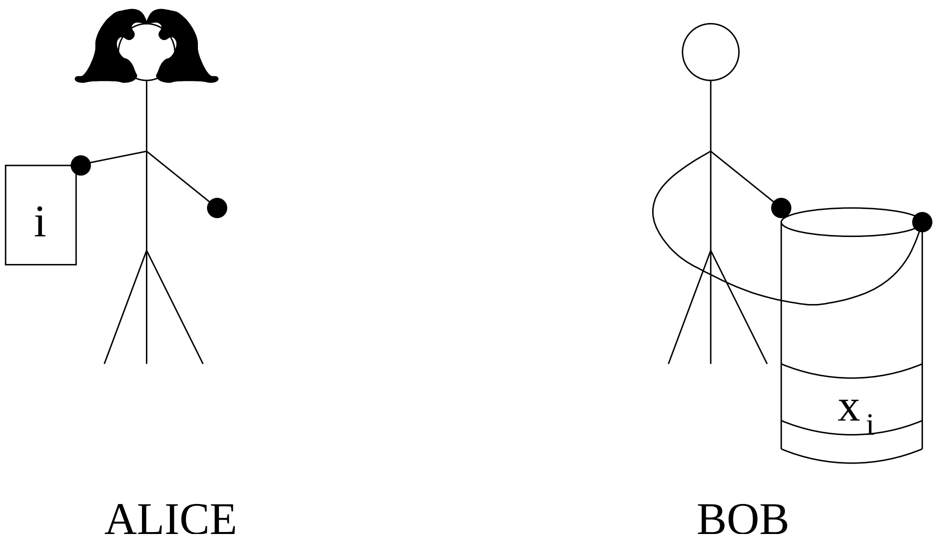
On Breaking the $O(n^{\frac{1}{2k-1}})$ Barrier for Information-Theoretic Private Information Retrieval

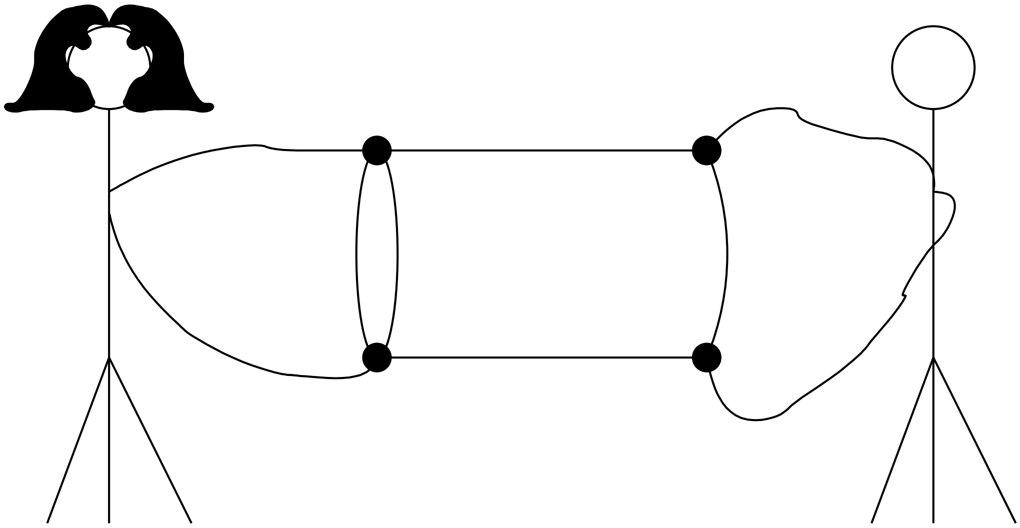
Amos Beimel   Yuval Ishai   Eyal Kushilevitz
Jean-Francois Raymond

Introduction



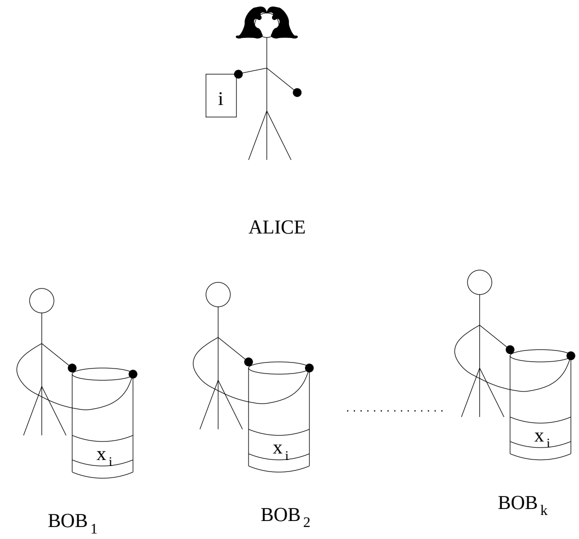
- Alice wants x_i
- Bob should not know what Alice is interested in.

Simple Protocol



- Bob hands everything to Alice
- Chor et. al. [CGKS95] prove this is optimal if there is only one copy of the database.

Clone BOB!



- Alice sends queries to $Bob_1, \cdots, Bob_k$ and reconstructs x_i from their answers.
- Clones do not talk to each other
  - *Privacy* No Bob_j by itself gets an idea about i.
- How many bits were exchanged ?

Overview of the talk

- The Model
- Results
- A Lower Bound
- Previous Work
- High Level View of the Solution
- Details....
- Extensions of the model

The Model

- Assume that the records are of 1 bit
  - k Servers $S_1, \cdots, S_k$; each has n bit string x
  - User, $\mathcal{U}$; knows n and wants to privately retrieve x_i, $i \in [n]$.
- A k-server PIR, $\mathcal{P} = (\mathcal{Q}, \mathcal{A}, \mathcal{C})$.
  - Step 1 $\mathcal{U}$: $\mathcal{Q}(n, k, i)$ outputs $(q_1, \cdots, q_k)$ [randomized] + aux.
    $\mathcal{U}$ sends q_j to S_j.
  - Step 2 S_j: $a_j = \mathcal{A}(k, j, x, q_j)$ [deterministic].
    $\mathcal{U}$ receives a_j.
  - Step 3 $\mathcal{U}$ calculates x_i by $\mathcal{C}(k, n, a_1 \cdots a_k, aux)$.
- *Correctness*: $\forall k, n, x, i$; $\mathcal{U}$ outputs the correct value of x_i with probability 1.
- *Privacy* $\forall k, n, i_1, i_2$ and $j \in [k]$; the distributions $\mathcal{Q}_j(k, n, i_1)$ and $\mathcal{Q}_j(k, n, i_2)$ are identical.
- *Communication Complexity*: Total number of bits exchanged
  - Max over all $x \in \{0, 1\}^n$, $i \in [n]$ and random inputs.
- *Linear* Protocols.

What numbers are we talking about?

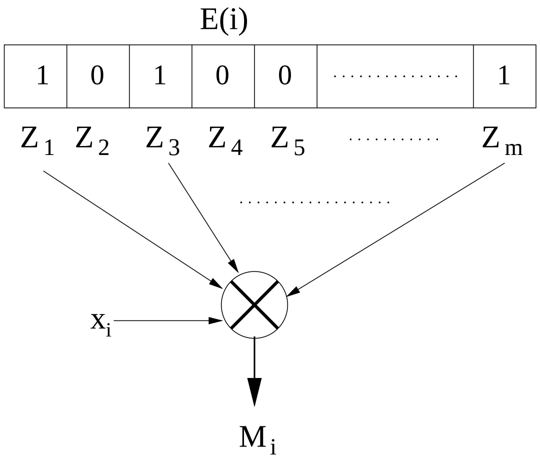
- Previous Best For fixed k, (Ambainis, 1997) $O(n^{\frac{1}{2k-1}})$
- This Work $n^{O(\frac{\log\log k}{k\log k})}$

A Trivial Lower Bound

- Communication Complexity of any PIR is $\Omega(\log_2 n)$.
- Follows from standard communication complexity arguments
  - Does not require privacy.

Using Polynomials for Bob

- n-bit x is associated with $P_x(Z_1, \cdots, Z_m)$ [over $GF(2)$] of degree d.
- Definition of P_x
  - $m = \Theta(n^{\frac{1}{d}})$
  - Define a one to one *encoding* $E : [n] \to \{0,1\}^m$.
    * $E(i)$ has weight d.
  - $P_x(Z_1, \cdots, Z_m) = \sum_{i=1}^{n} M_i$
    * $M_i = x_i \cdot \Pi_{E(i)_l=1} Z_l$.



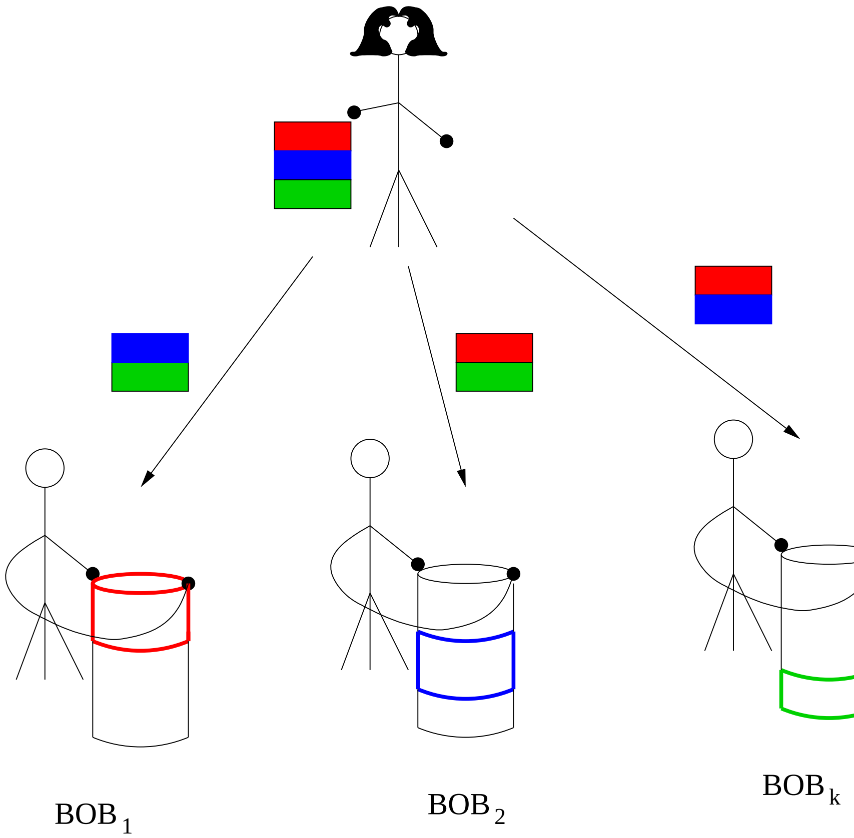
- $P_x(E(i)) = x_i$.

What does all this mean ?

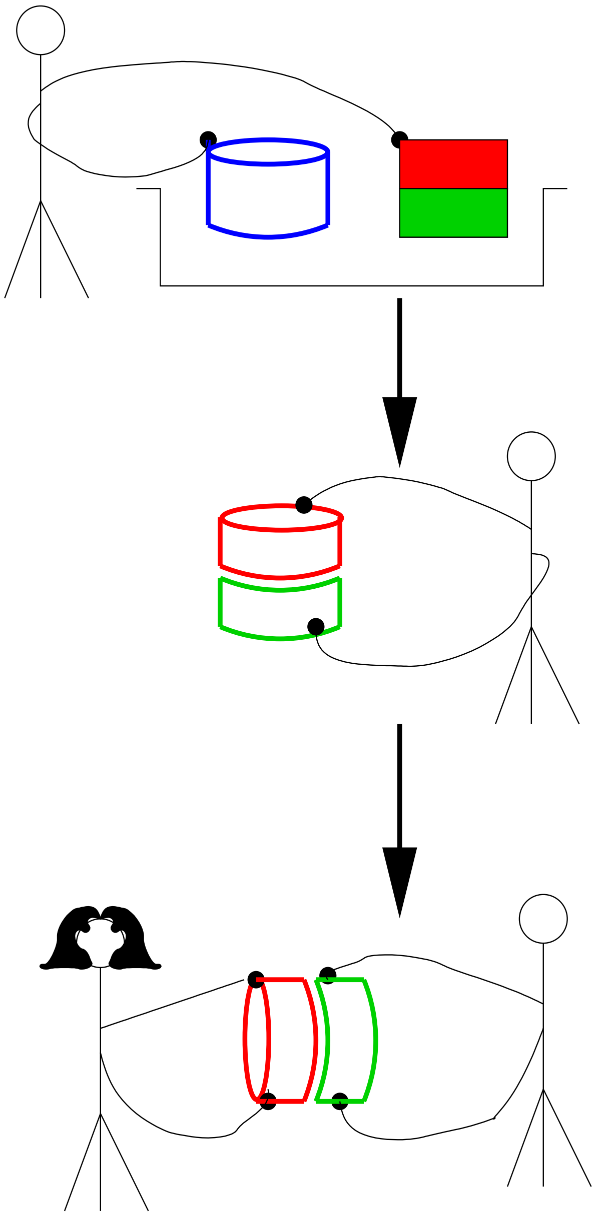
- Each server, S_j knows *all* the coefficients of P_x.
  - Servers know the encoding E.
- $\mathcal{U}$ knows $i \Rightarrow \mathcal{U}$ knows $E(i)$.
- $\mathcal{U}$ has to evaluate $P_x(E(i))$ while keeping $E(i)$ secret.
- **Simple Protocol** $k = 1$. S_1 sends all the $O(n)$ coefficients to $\mathcal{U}$.
  - Nothing yet!

Basic Idea

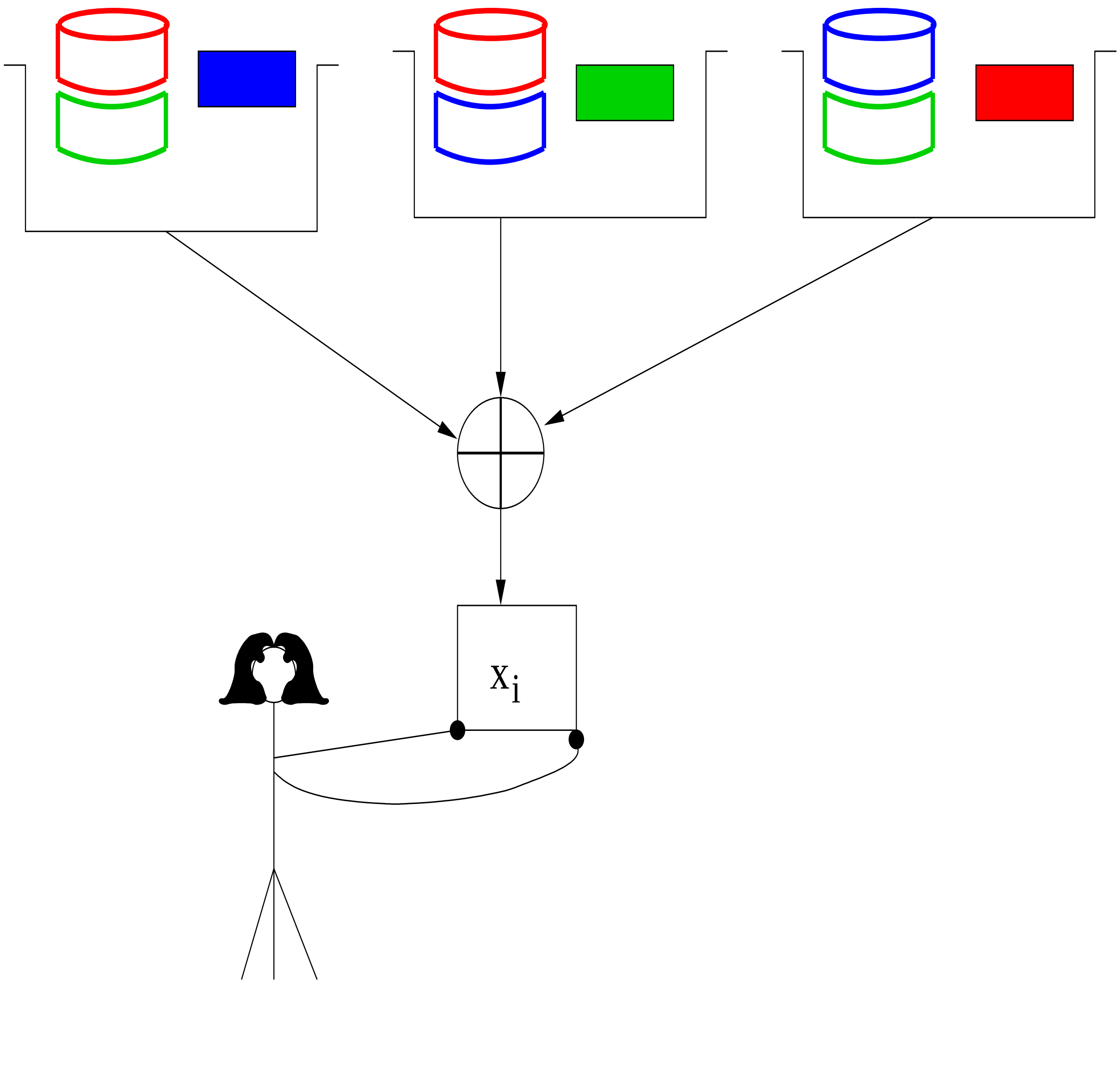
- $\mathcal{U}$ chooses (uniformly and independently) at random $\vec{y}_1, \cdots, \vec{y}_k \in \{0,1\}^m$ such that $E(i) = \sum_{j=1}^k \vec{y}_j$.
- $\mathcal{U}$ sends to S_j all $\vec{y}_i$s except $\vec{y}_j$.
- *Privacy*: Each of the $(k-1)$ $\vec{y}$'s sent to S_j are uniformly and independently distributed.
- Each S_j is "assigned" some set of monomials of P_x.



What does BOB_2 do ?

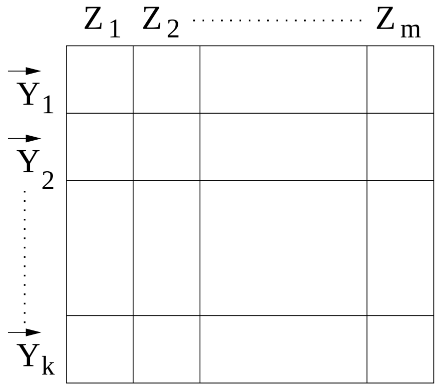


What does Alice do ?



We need variables!

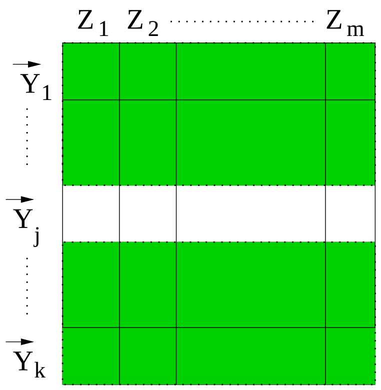
- $\vec{y}_j$ consists of m values $y_{j,h}$; $h = 1, \cdots, m$
  - $y_{j,h}$ is the h^{th} bit of $\vec{y}_j$.
- Consider $\forall h \in [m]$, $Z_h = \sum_{j=1}^{k} Y_{j,h}$



- $P_x(E(i))$ is obtained by substituting $y_{j,h}$ for $Y_{j,h}$
  - Remember $E(i) = \sum_{j=1}^{k} \vec{y}_j$.
- Q_x is P_x viewed as a polynomial in $Y_{j,h}$; $j \in [k]$, $h \in [m]$.

The Protocol Outline

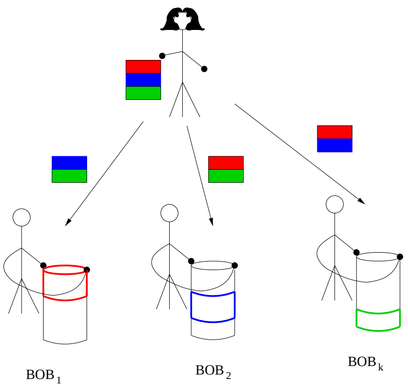
- Q_x is a degree d polynomial in mk variables.
- M be any monomial of Q_x
  - M depends on at most d variables
  - Each variable in M is missed by one server

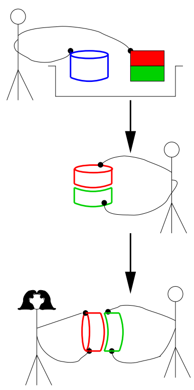


  - There exists a server that misses at most $\lfloor d/k \rfloor$ of the variables of M.
    - "assign" M to this server

Protocol with complexity $O(n^{\frac{1}{k-1}})$

- Let $d = k - 1$
  - $\lfloor d/k \rfloor = 0$, i.e., the server to which M is assigned knows the values of all the variables.
  - The assigned server computes the value of M
  - Each server sends back sum of the values of all assigned monomials
- User sends $m(k-1)k$ bits and gets back $1 \cdot k$ bits





- For fixed k, Comm. Complexity $= O(m) = (n^{\frac{1}{d}}) = O(n^{\frac{1}{k-1}})$.

Protocol with complexity $O(n^{\frac{1}{2k-1}})$

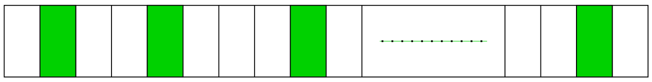
- Let $d = 2k - 1$
  - $\lfloor d/k \rfloor = 1$, i.e., the server to which M is assigned knows the values of all but one variable.
  - After all substitutions, let the sum of monomials assigned to S_j be $P_j(Y_{j,1}, \cdots, Y_{j,m})$
    - P_j has degree 1
  - Each server send back $m + 1$ coefficients
- User send $m(k-1)k$ bits and gets back $(m+1)k$ bits
- Comm. Complexity $= O(m) = O(n^{\frac{1}{2k-1}})$.

Great, Step on the gas!

- Increasing d further does not work
- $d = ck - 1$; $c > 2$
  - Partial polynomial P_j would have $\Theta(m^{c-1})$ coefficients
- Comm. Complexity $= O(m^{c-1}) = O(n^{\frac{c-1}{ck-1}})$.
  - $\frac{c-1}{ck-1} > \frac{1}{2k-1}$.

Okay, so what does this paper do?

- We have $P_x(\sum_{j=1}^{k} \vec{y}_j) = \sum_{V \subseteq [k]} P_V(z_V)$.
  - P_V known to each server in V
  - $\mathcal{U}$ knows z_V.
- Each server misses atmost λ variables in each monomial in P_V.
- Knowing z_V, a small # of coeff. req. to obtain $P_V(z_V)$
  - Infact $Z_V = E(i)$
    * Recollect: $Wt(E(i)) = d$.
    * Most of the bits in $E(i)$ is 0 $\Rightarrow$ most M in $P_V(E(i)) = 0$.
    * User needs to retrieve only those M where all vars $= 1$.
  - Identity of coeff. $\Rightarrow$ info. about i
  - Apply PIR *recursively* for each coeff.



Why would recursion work?

- Parameters λ, k'
- $|V| \geq k'$ (except $|V| = 1$ is allowed)
- Each $j \in V$ misses atmost λ vars. in each M of P_V.
  - All but $\leq \lambda|V|$ vars. known to all servers in V
  - After substiuting the values known to servers in V, P_V has degree $\leq \lambda|V|$
    * Choose λ so that $\lambda|V| < d$.
  - # vars. in P_V is m (same as P_x)
  - Total # of coeff. in $P_V = O(m^{\lambda|V|})$.
  - # of coeff. to retrieve $\leq 2^d$.
- Get 2^d bits out of $O(m^{\lambda|V|})$ bits
  - 2^d executions of $|V|$-server PIR on database size $O(n^{\frac{\lambda|V|}{d}})$

Let us fix the Parameters

1. λ, $k' \leq k$.
2. $d \leq (\lambda+1)k - (\lambda-1)k' + (\lambda-2)$.
3. M be any monomial of Q_x
   - M has degree atmost d.
   - variables $Y_{j,h}$, where $j \in [k]$ and $h \in [m]$.
4. There is a server that misses atmost one variable or $|V(M)| \geq k'$.

Proof of 4: Assume otherwise.

- k servers miss atleast 2 variables.
- Atmost $k'-1$ servers miss atmost λ variables of M.
  - Atleast $k-(k'-1)$ servers miss atleast λ variables.

$\Rightarrow$

- M has atleast $(k-(k'-1))(\lambda+1)+(k'-1)\cdot 2 \geq d+1$ variables

A Contradiction.

A quick analysis

- $\mathcal{U}$ sends out $k(k-1)m$ bits
- Servers for each P_j return $(m+1)$ bits
- $C(n,k) = m(k-1)k + (m+1)k + \sum_{V \subseteq [n]: |V| \geq k'} C(m^{\lambda |V|}, |V|)$.
  - $C(n,k) = O(n^{\frac{1}{d}} + \sum_{l=k'}^{k} \binom{k}{l} C(n^{\frac{\lambda l}{d}}, l))$.
- Forall i s.t. $k \geq (i-1)!$, we can choose parameters to get
  $C(n,k) = O(n^{\frac{2}{ik}})$
- Choose $i = \lceil \frac{\log k}{\log \log k} \rceil$.

- $x_i = P_x(E(i))$
  $= \sum_{j-1}^{k} P_j(\vec{y}_j) + \sum_{V \subseteq [k]; |V| \geq k'} P_V(E(i))$

Extensions of basic PIR model

- Computational PIR
  Kushilevitz and Ostrovsky [KO97]
  - It can be done by 1 server.
- User is protected against collusions of upto t servers
  Chor et al [CGKS95]
- Protect databases from the user also.
  Naor and Pinkas [NP99]
- Private Information Storage (reading as well as writing)
  Ostrovsky and Shoup [OS97]

References

CSKG95 B. Chor, O. Goldreich, E. Kushilevitz and M. Sudan, "Private Information Retrieval," in *STOC 95*

KO97 E. Kushilevitz and R. Ostrovsky, "Replication is not needed: Single database, computationally-private information retrieval," in *FOCS 97*

NP99 M. Naor and B. Pinkas, "Oblivious Transfer and polynomial evaluation," in *STOC 99*

OS97 R. Ostrovsky and V. Shoup, "Private Information Storage," in *STOC 97*.