Private Information Retrieval

Yuval Ishai

Technion

The Problem

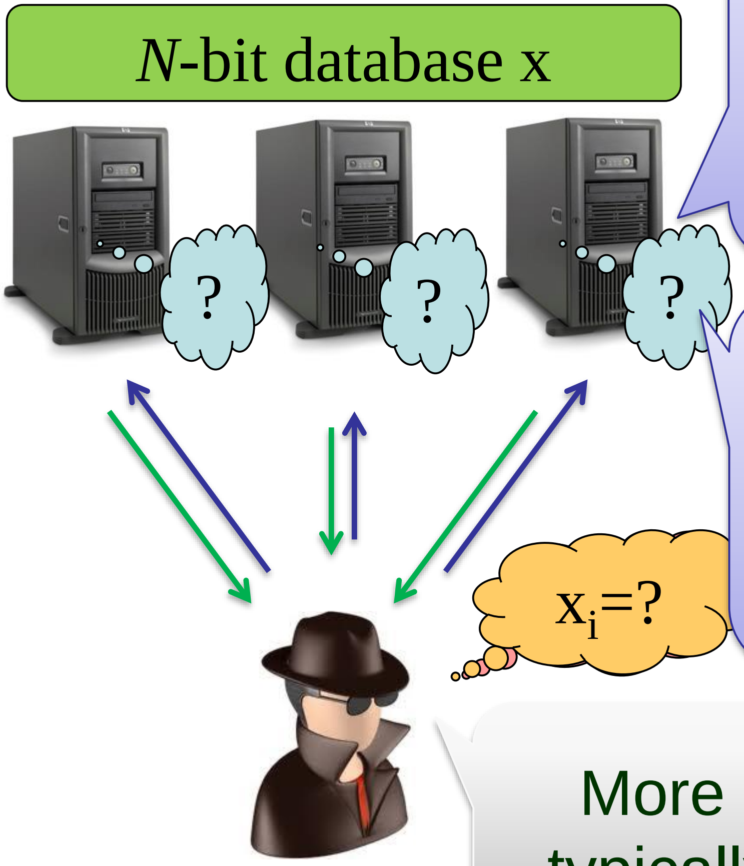
[Chor-Goldreich-Kushilevitz-Sudan95]

1-bit records
vs.
b-bit records

Trivial solution:
Download x

Main question:
minimize communication
(log$N$ vs. N)

building block for
sublinear MPC

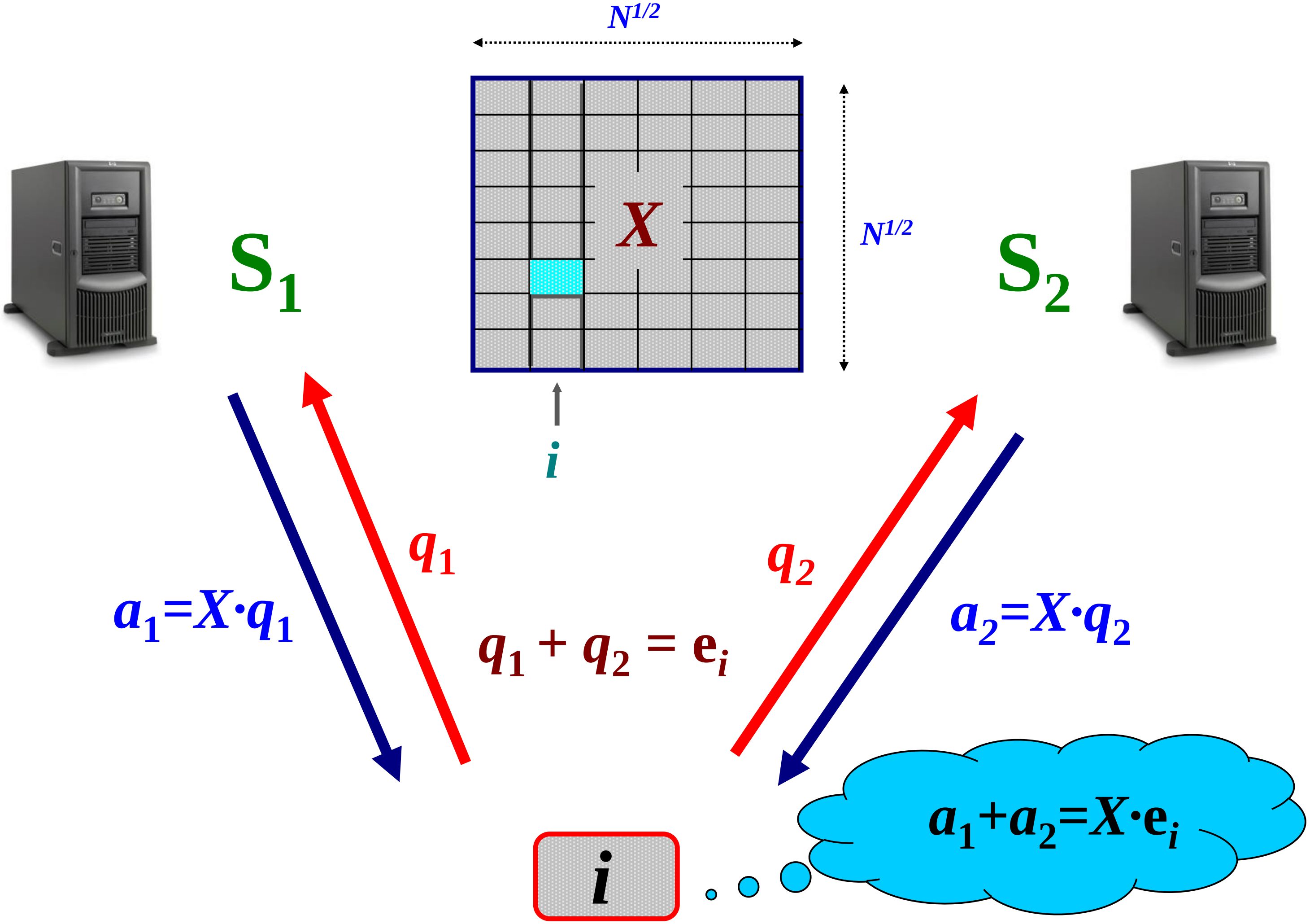


1-private
vs.
t-private

"Information-Theoretic"
vs.
Computational

More interaction:
typically not helpful

2-Server IT PIR example



$N^{1/2}$

S_1

X

$N^{1/2}$

S_2

i

q_1

q_2

$a_1=X\cdot q_1$

$q_1 + q_2 = \mathbf{e}_i$

$a_2=X\cdot q_2$

$a_1+a_2=X\cdot \mathbf{e}_i$

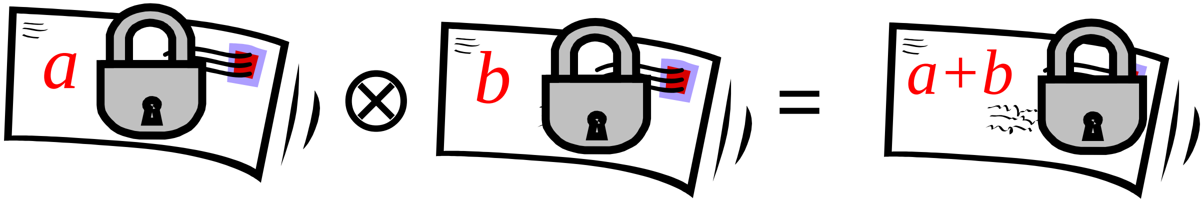
i

➔ **2-server PIR with $O(N^{1/2})$ communication**

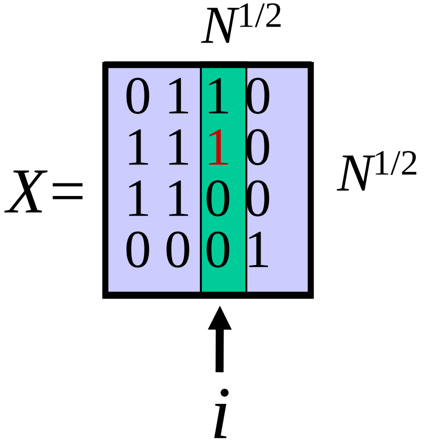
1-Server CPIR example

[Kushilevitz-Ostrovsky97]

Tool: additively homomorphic encryption



Protocol:



$$X = \begin{array}{cccc} 0 & 1 & 1 & 0 \\ 1 & 1 & {\color{red}1} & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{array}$$

- Client sends $\mathbf{E}(\mathbf{e}_i)$

  $E(0)\ E(0)\ E(1)\ E(0)\ (= c_1\ c_2\ c_3\ c_4)$

- Server replies with $\mathbf{E}(X \cdot \mathbf{e}_i)$

  $c_2 \otimes c_3$
  $c_1 \otimes c_2 \otimes c_3$
  $c_1 \otimes c_2$
  c_4

- Client recovers *i*th column of X

➔ 1-server CPIR with ~ $O(N^{1/2})$ communication

Why *Information-Theoretic* PIR?

Cons:

- Requires multiple servers
- Privacy against limited collusions
- Worse asymptotic complexity (with constant # servers *k*): $2^{(\log N)^{\epsilon}\, n}$ vs. polylog(*N*)

Pros:

- Challenging theoretical question
- Unconditional security
- Good concrete efficiency
- Allows for very short (logarithmic) queries or very short (constant-size) answers ➔ applications!
- Closely related to locally decodable codes

3 Regimes

- Short answers (O(1) bit from each server)
  - Application: PIR for long records
- Balanced communication
  - Typically reduces number of servers by factor of ~ 2
- Short queries (O(log$N$) bits to each server)
  - Application: PIR with preprocessing

Brief history

- For concreteness:
  - – 3-server protocols
  - – Answer length O(1)
- Lower bounds
  - – [Mann98,…,Woodruff07]: $c \cdot \log N$ for c>1
- Upper bounds
  - – [CGKS95] $O(N^{1/2})$
  - – [Yekhanin07] $N^{O(1/\log\log N)}$
  - – [Efremenko09] $N^{O(\sqrt{\log\log N}/\sqrt{\log N})}$

Assuming infinitely many Mersenne primes

Hidden constant > 100

Brief history

- For concreteness:
  - 3-server protocols
  - Answer length O(1)
- Lower bounds
  - [Mann98,…,Woodruff07]: $c \cdot \log N$ for c>1
- Upper bounds
  - [CGKS95] $O(N^{1/2})$
  - [Yekhanin07] $N^{O(1/\log\log N)}$
  - [Efremenko09] $N^{O(\sqrt{\log\log N}/\sqrt{\log N})}$

[Beimel-I-Kushilevitz-Orlov12]:
Hidden constant ≈ 6

Brief history

- For concreteness:
  - 3-server protocols
  - Answer length O(1)
- Lower bounds
  - [Mann98,…,Woodruff07]: $c \cdot \log N$ for c>1
- Upper bounds
  - [CGKS95] $O(N^{1/2})$
  - [Yekhanin07] $N^{O(1/\log\log N)}$
  - [Efremenko09] $N^{O(\sqrt{\log\log N}/\sqrt{\log N})}$

[Dvir-Gopi15]:
2 servers, balanced

A longer version

Complexity theory

[BF90, BFKR90]
Instance hiding,
locally random reductions

[KT00]
LDC vs. PIR

3rd Gen

[Yek07]
Breakthrough

[Efr09]
Best short answers

[DG15]
Best balanced

Crypto

1st Gen

[CGKS95]
PIR

[CG97]
2-server CPIR

[KO97]
1-server CPIR

2nd Gen

[BIKR02]
$N^{o(1/k)}$

[IK04,BIKK14]
=> MPC

[LVW17,LV18,…]
=> secret sharing

Rest of Talk

- The bigger picture
- 1^{st} (+ 2^{nd}) generation PIR
- PIR via homomorphic secret sharing
  - General blueprint for 3^{rd} generation PIR
- Open problems

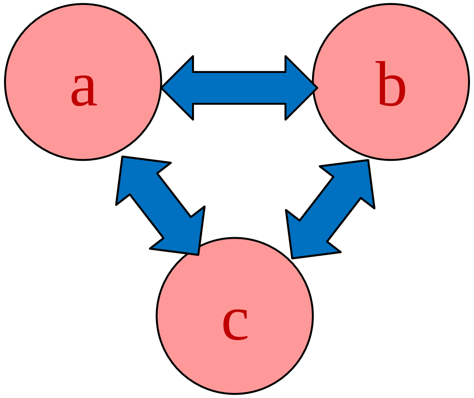
Communication Complexity of Cryptography

Fully Homomorphic Encryption

Gentry ‘09

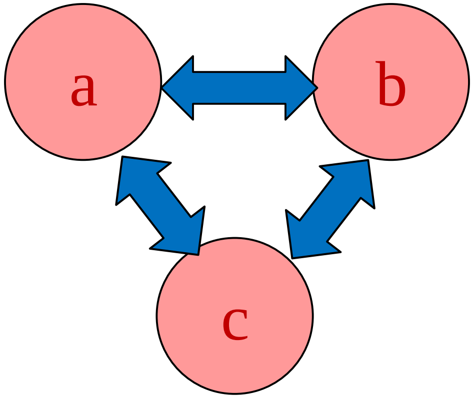
- Essentially settles communication complexity questions in complexity-based cryptography
- Main open questions
  - – Further improve assumptions (eliminate “circular security”)
  - – Improve concrete computational overhead
    - FHE >> PKE >> SKE >> one-time pad

Information-Theoretic MPC



	Communication Complexity	Secure Multiparty Computation (MPC)
Goal	Each party learns f(a,b,c)	Each party learns **only** f(a,b,c)

Information-Theoretic MPC



	Communication Complexity	**Secure Multiparty Computation (MPC)**
Goal	Each party learns f(a,b,c)	Each party learns **only** f(a,b,c)
Upper bound	O(n) (n = input length)	O(**size**(f)) [BGW88,CCD88]

Information-Theoretic MPC

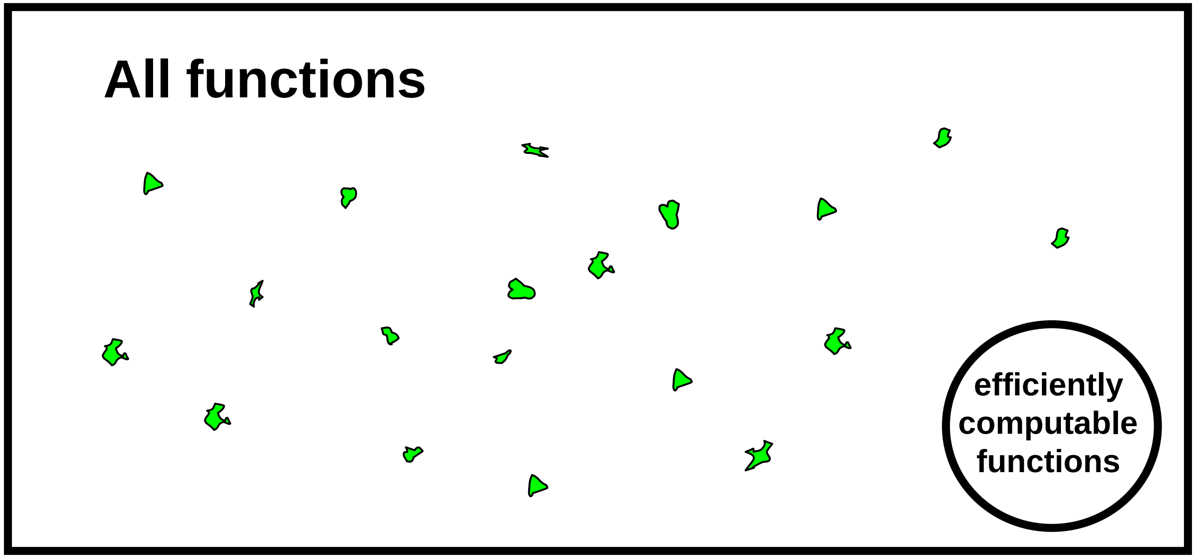
Big open question:
poly(n) communication for all f ?

“fully homomorphic encryption of information-theoretic cryptography”

	tion plexity	Secure Multiparty Computation (MPC)
Goal	Each party learns f(a,b,c)	Each party learns **only** f(a,b,c)
Upper bound	O(n) (n = input length)	O(**size**(f)) [BGW88,CCD88]
Lower bound	Ω(n) (for most f)	Ω(n) (for most f)

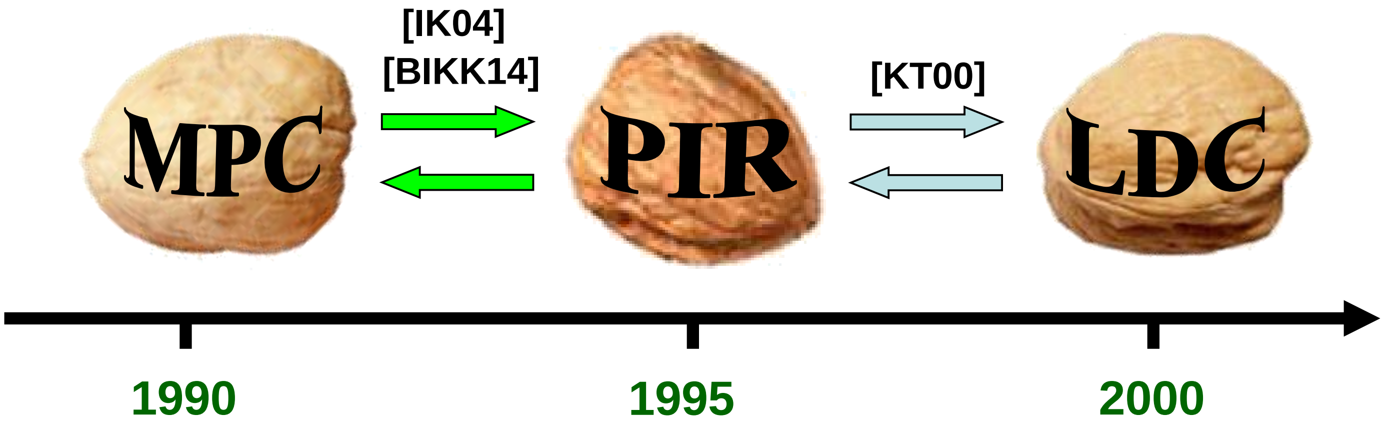
Question Reformulated

Is the communication complexity of MPC strongly correlated with the computational complexity of the function being computed?



🟩 = communication-efficient MPC

🟥 = no communication-efficient MPC



MPC

[IK04]
[BIKK14]

PIR

[KT00]

LDC

1990

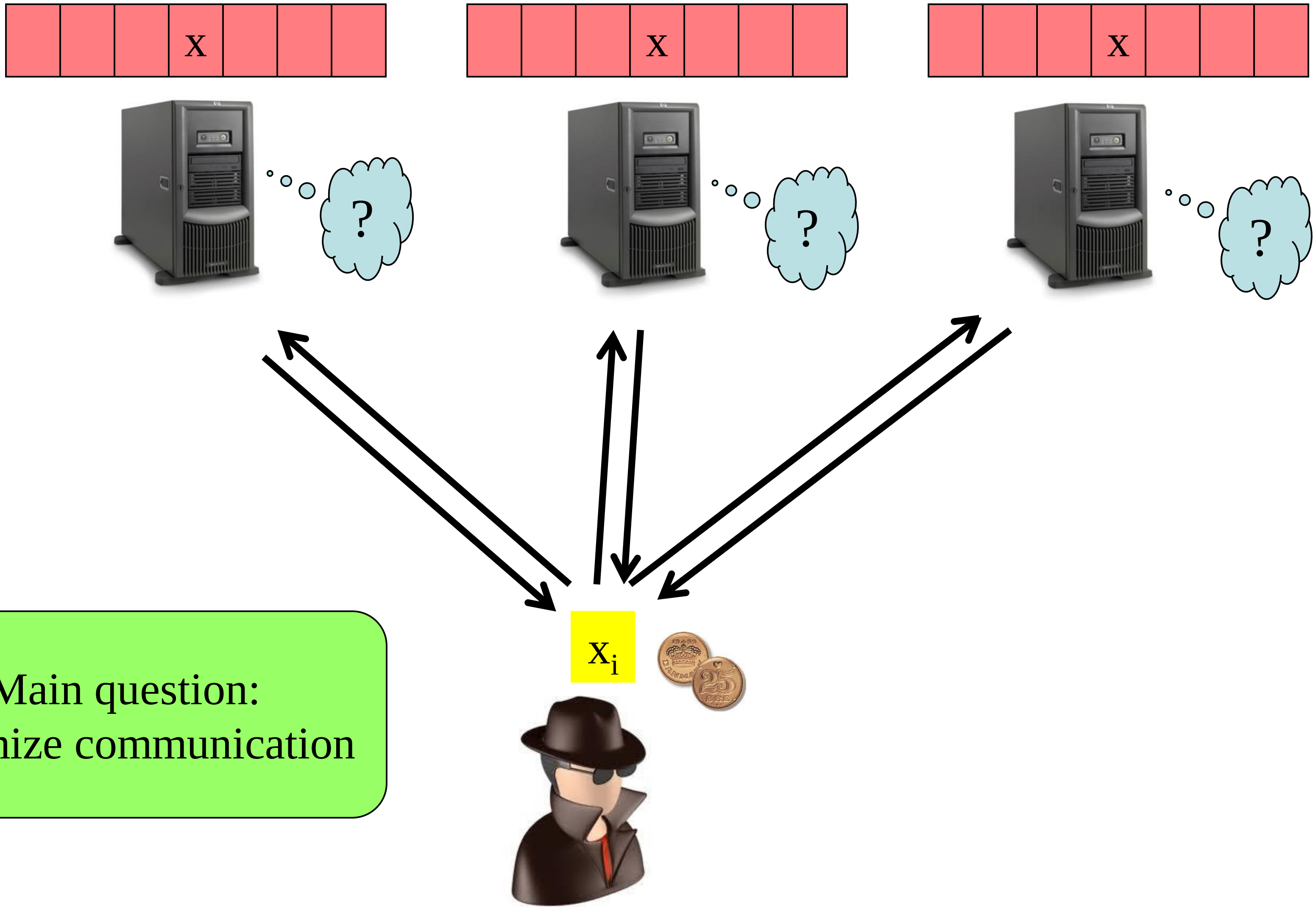
1995

2000

- The three problems are closely related

Back to 1st Generation…

Information-Theoretic PIR



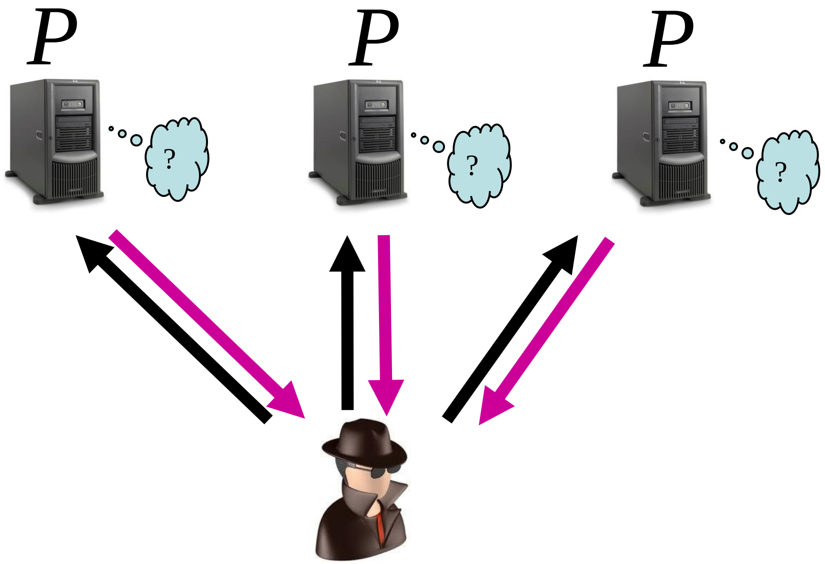
Main question:
minimize communication

Arithmetization

$$\mathrm{x} \quad \Rightarrow \quad P_{\mathrm{x}} \in F[Z_1,\ldots,Z_m]$$

$$i \quad \Rightarrow \quad \mathrm{z}_i \in F^m$$

$$\forall\ i \in [N], \quad P_{\mathrm{x}}(\mathrm{z}_i) = \mathrm{x}_i$$



$$z \Rightarrow P(z)$$

Parameters

Field F = GF(2)

Degree d = const.

#vars m s.t. $\binom{m}{d} \geq N \Rightarrow m = O(N^{1/d})$ suffices

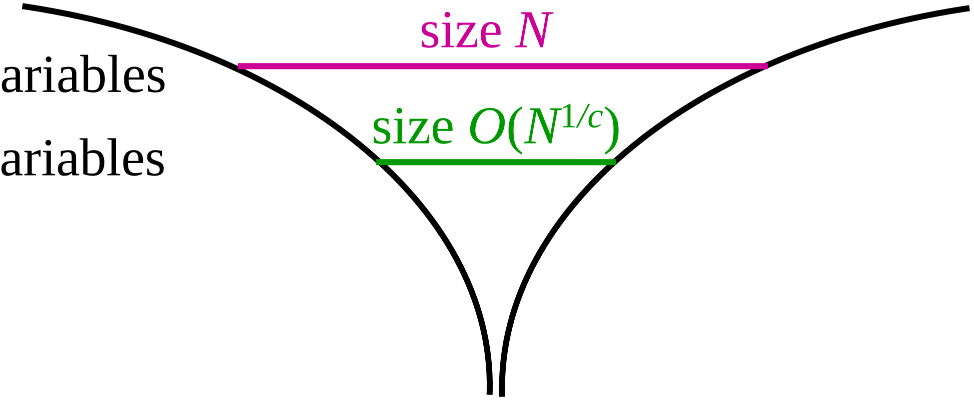
Ex. $d=3,\ m=8, N=\binom{8}{3}$

$\mathbf{z}_1$=11100000 $\mathbf{z}_2$=11010000 …. $\mathbf{z}_N$=00000111

$M_1 = Z_1Z_2Z_3$ $M_2 = Z_1Z_2Z_4$ $M_N = Z_6Z_7Z_8$

$$P_{\mathbf{x}} = \sum_{i=1}^{N} x_i M_i$$

Key Idea: Degree Reduction



size N

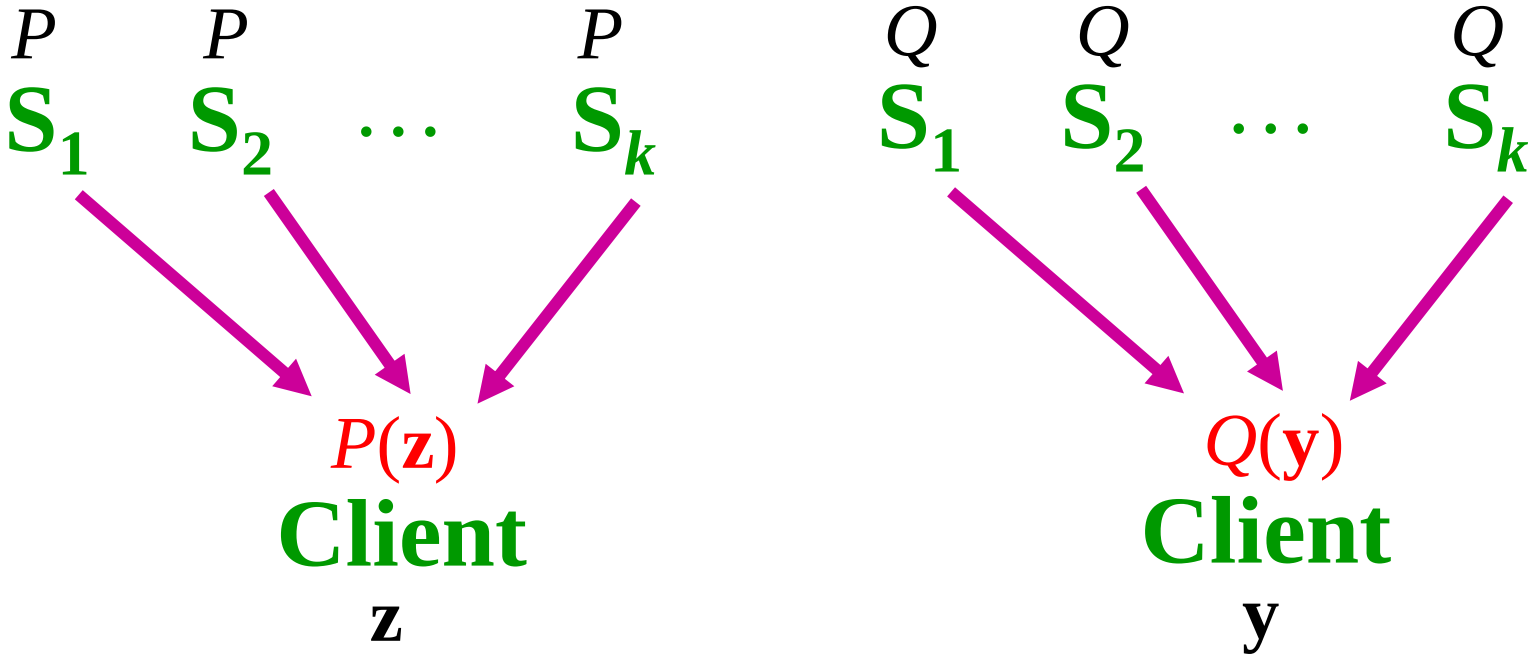
degree d, m variables

size $O(N^{1/c})$

degree d/c, m variables

Degree Reduction Using Partial Information

[BabaiKimmelLokam95,Beimel-I01]

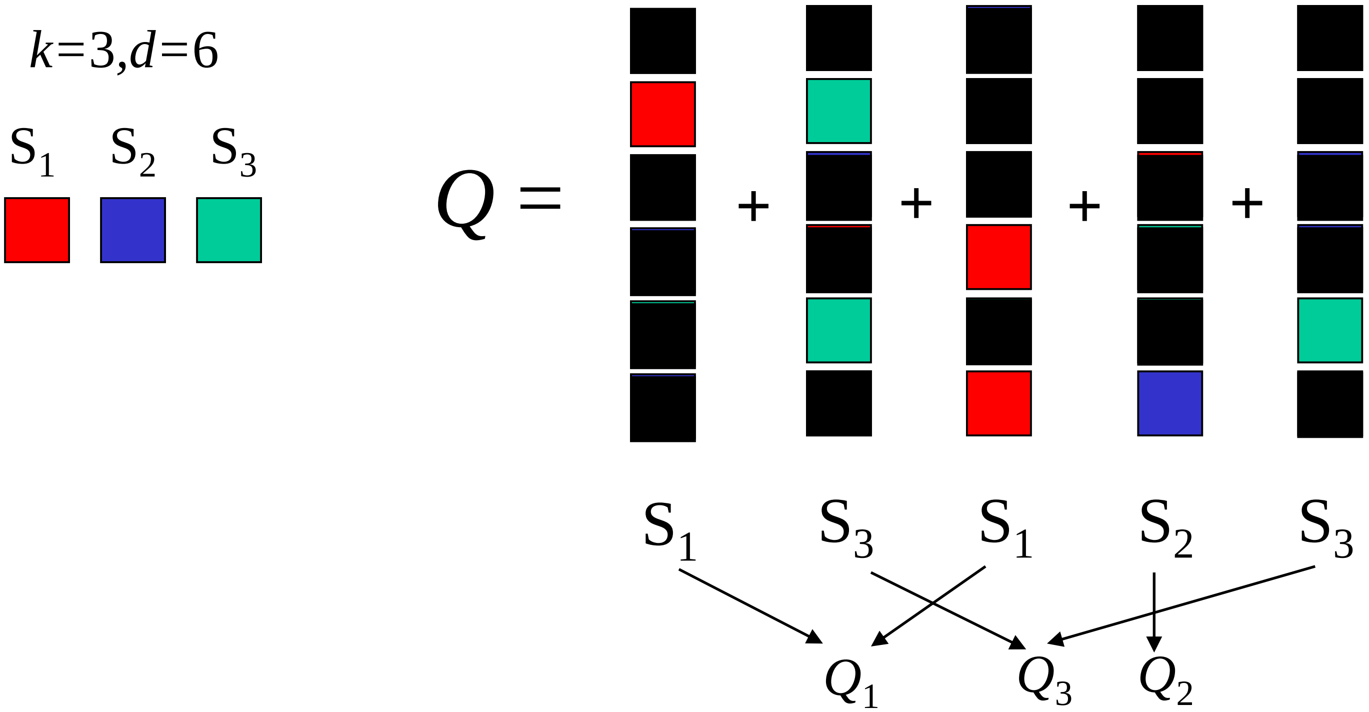


z is hidden from servers

Each entry of **y** is known to *all but one* server

CNF (aka replicated) secret sharing

$k=3, d=6$



$Q(\mathbf{y})=Q_1(\mathbf{y})+Q_2(\mathbf{y})+Q_3(\mathbf{y})$

$\deg Q_j \le d/k = 2$

➔ $Q(\mathbf{y})$ communicated with $O(N^{1/3})$ bits

Privately Evaluating P(z)

$O(m) = O(N^{1/d})$

- Client picks random $\mathbf{y}_1,\ldots,\mathbf{y}_k$ s.t. $\mathbf{y}_1+\ldots+\mathbf{y}_k=\mathbf{z}$, and sends to S_j all **y**'s *except* $\mathbf{y}_j$.
- Servers define an mk-variate degree-d polynomial $Q(\mathbf{Y}_1,\ldots,\mathbf{Y}_k)= P(\mathbf{Y}_1+\ldots+\mathbf{Y}_k)$.
- Each S_j computes degree-(d/k) poly. Q_j , such that $Q(\mathbf{y})= Q_1(\mathbf{y})+\ldots+Q_k(\mathbf{y})$.

$O(N^{1/k})$

- S_j sends a description of Q_j to Client.
- Client computes $\Sigma Q_j(\mathbf{y})=x_i$.

A Closer Look

- $\forall M\ \exists S_j$ missing at most $\lfloor d/k \rfloor$ variables.

  $\Rightarrow \deg Q_j \le \lfloor d/k \rfloor$

Useful parameters:

Best 1st Gen *binary* PIR

- $d=k-1 \Rightarrow$ query length $O(N^{1/(k-1)})$
  $\lfloor d/k \rfloor = 0 \Rightarrow$ answer length 1

Best 1st Gen *balanced* PIR

- $d=2k-1 \Rightarrow$ query length $O(N^{1/(2k-1)})$
  $\lfloor d/k \rfloor = 1 \Rightarrow$ answer length $O(N^{1/(2k-1)})$

A Closer Look

- $\forall\ M\ \ \exists\ S_j$ missing

  $\Rightarrow \deg Q_j \leq \lfloor d/k \rfloor$

Woodruff-Yekhanin05:
Better $O_k(.)$ dependence
via Shamir + partial derivatives

Useful parameters:

Best
1st Gen
binary
PIR

- $d=k\text{-}1 \quad \Rightarrow$ query length $O(N^{1/(k-1)})$

  $\lfloor d/k \rfloor = 0 \Rightarrow$ answer length 1

Best
1st Gen
balanced
PIR

- $d=2k\text{-}1 \ \Rightarrow$ query length $O(N^{1/(2k-1)})$

  $\lfloor d/k \rfloor = 1 \Rightarrow$ answer length $O(N^{1/(2k-1)})$

Best
current
short-query
PIR

- $d=O(\log N) \ \Rightarrow$ query length $O(\log N)$

  $\lfloor d/k \rfloor \cong d/k \Rightarrow$ answer length $O(N^{1/k+\epsilon})$

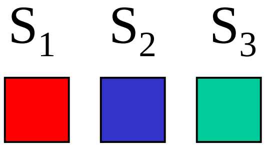
2^{nd} Gen: Breaking the $O(n^{1/(2k-1)})$ Barrier

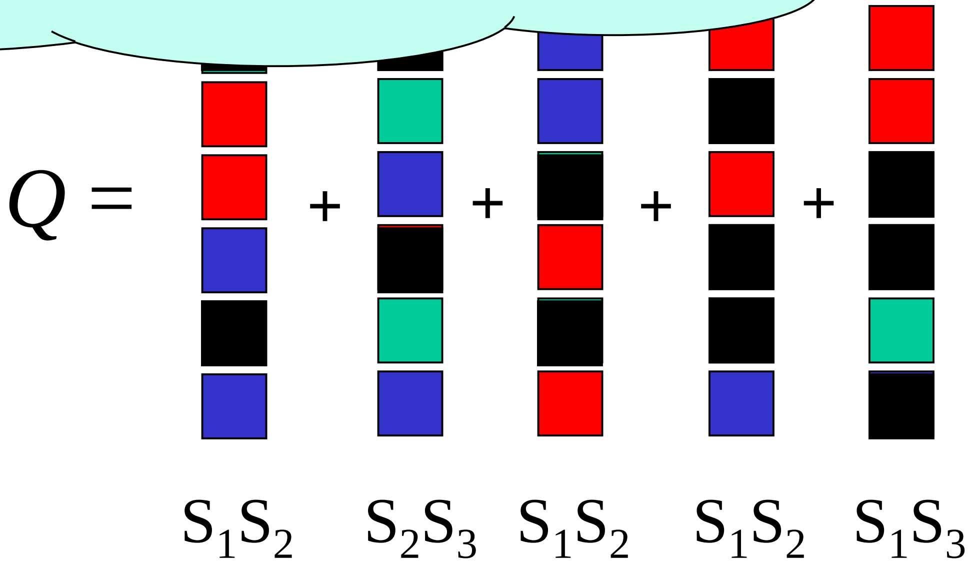
[Beimel-I-Kushilevitz-Raymond02]

- Rough idea: apply multiple “partial” degree reduction steps to boost the integer truncation affect.
- Generalized degree reduction:
  Assign each monomial to the *k’* servers *V* which jointly
  miss th

- Implementation: complicated and messy
- Essentially subsumed by 3^{rd} Gen PIR

$k=3, d=6, k'=2$





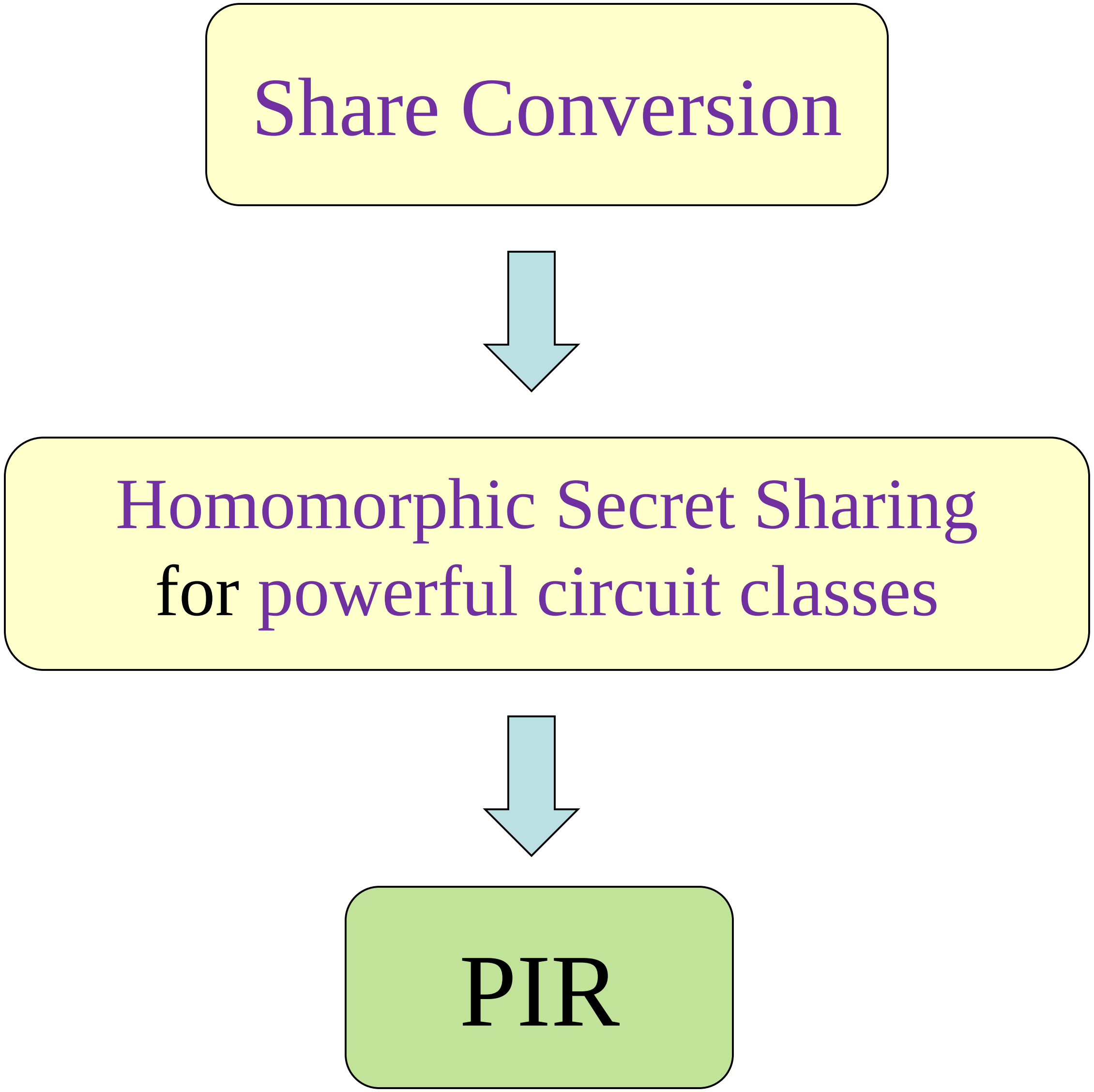
$$Q(\mathbf{y}) = \sum_{|V|=k'} Q_V(\mathbf{y})$$

Information-Theoretic PIR:

A Homomorphic Secret Sharing View

Coding view + missing details:
Klim’s talks

Blueprint for 3rd Gen PIR



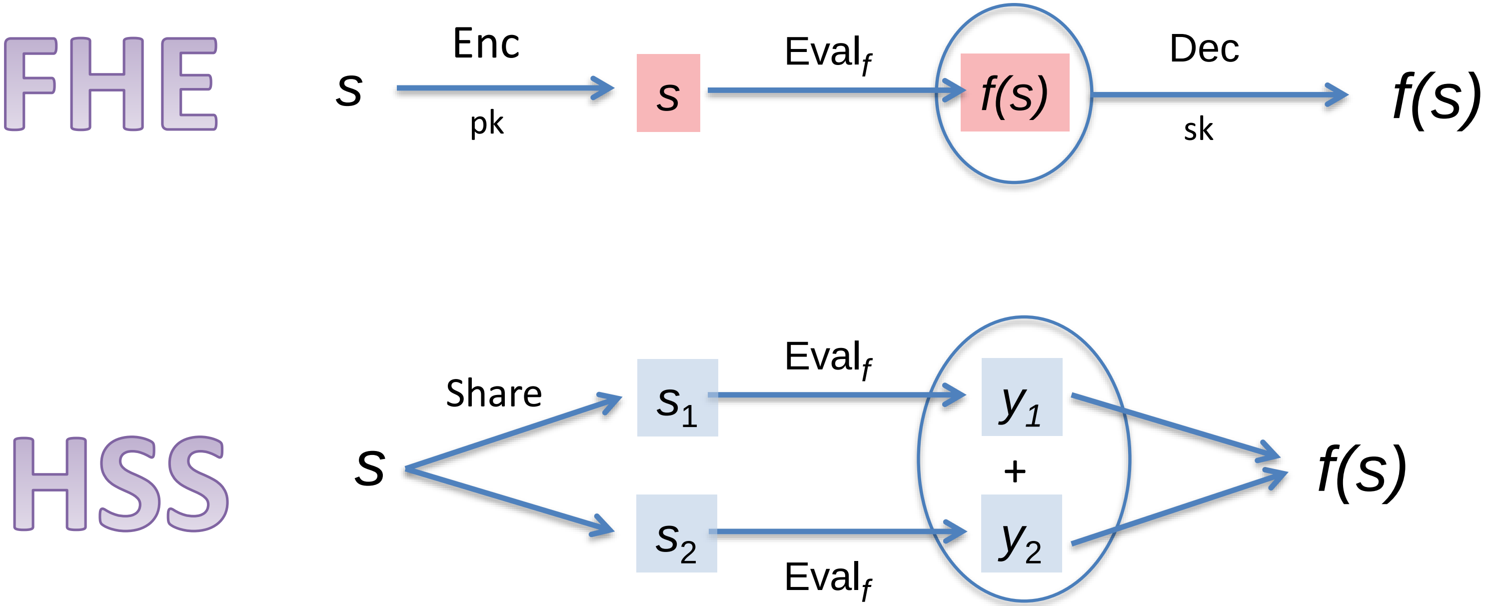
Share Conversion

Homomorphic Secret Sharing
for powerful circuit classes

PIR

Homomorphic Secret Sharing

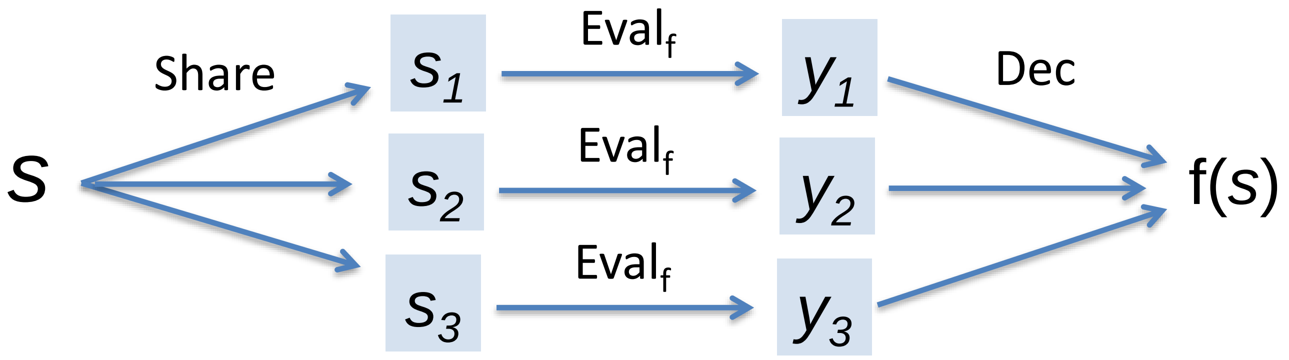
Relaxing FHE?



- Assuming 2+ non-colluding parties (sometimes not an issue!)
- No need for keys
- IT security or broader computational assumptions
- Additive decoding, better efficiency

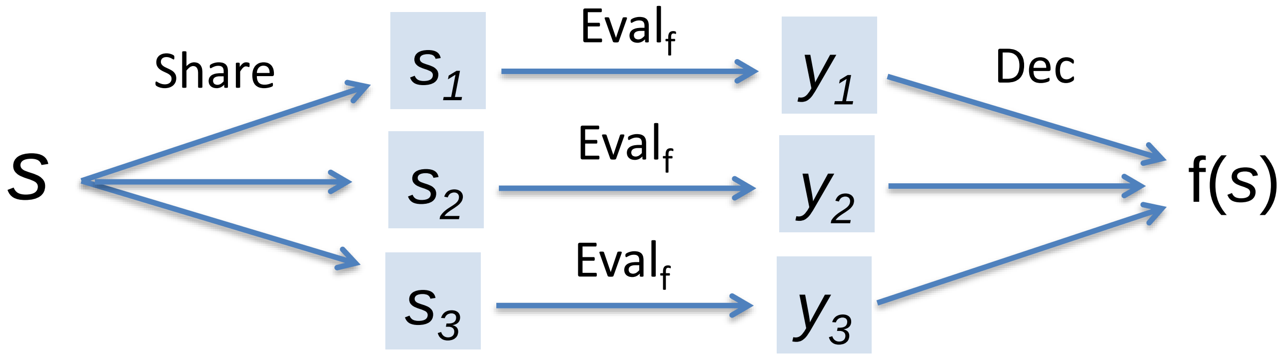
Many useful HSS flavors…

[Benaloh86, Boyle-Gilboa-I16, BGI-Lin-Tessaro18]



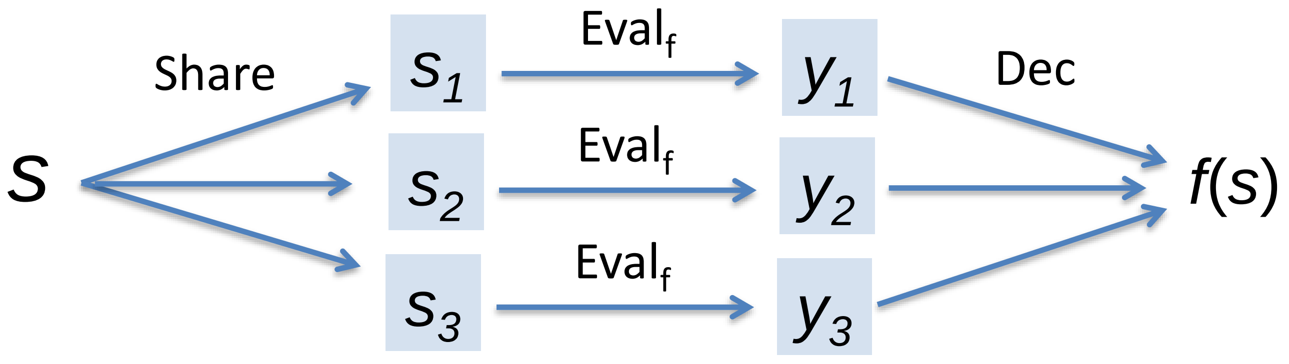
- **(k,t)**-HSS: k shares, each t keep s secret
- **Secrecy**: perfect vs. computational
- **Decoding**: additive vs. general
- **Single input vs. multi-input**

This Talk



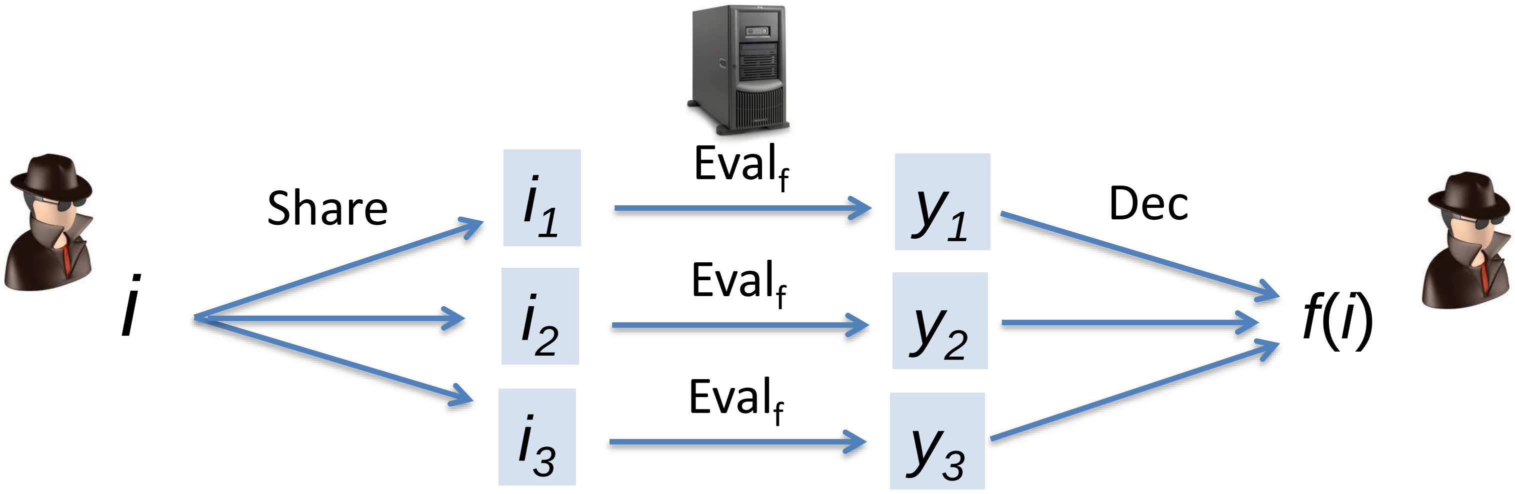
- **(k,1)**-HSS
- **Secrecy**: perfect
- **Decoding**: additive or general
- **Single input**

HSS Parameters



- **Function class F**
- **Input share size**
- **Output share size**

PIR as instance of HSS



Share

i

i_1

i_2

i_3

Eval_f

Eval_f

Eval_f

y_1

y_2

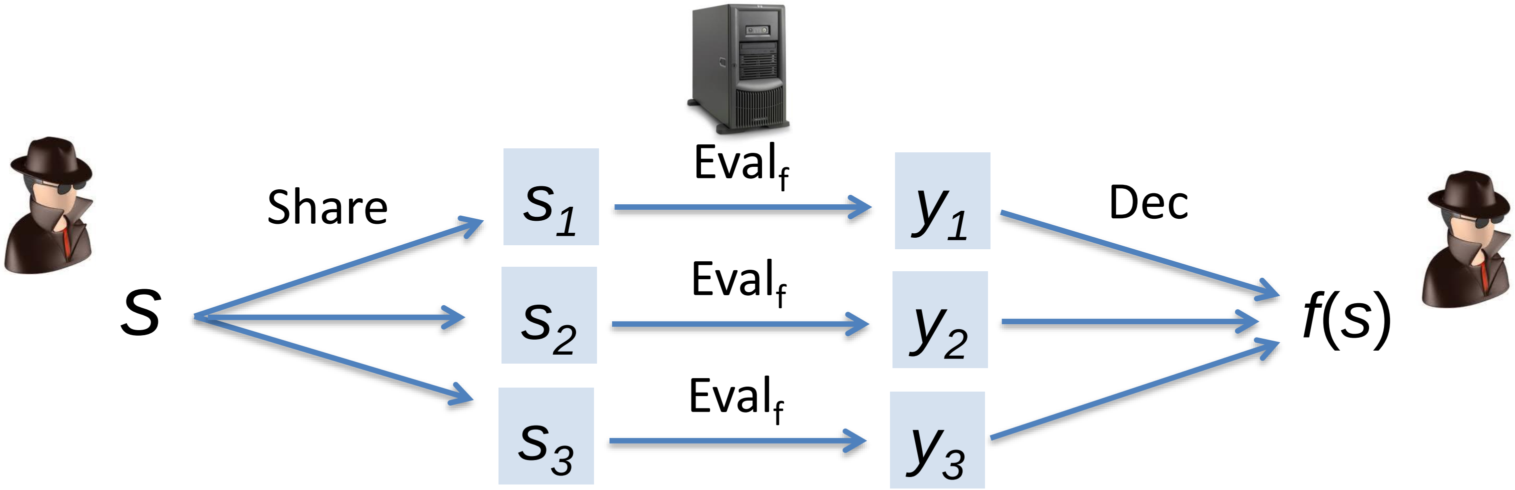
y_3

Dec

$f(i)$

- **Function class F:** all $f:\{0,1\}^n \rightarrow \{0,1\}$ for $n=\log N$
  - – For database x, $f(i)=x_i$
- **Input share size:** as small as we can…
- **Output share size:** O(1) (short answer regime)

PIR from arbitrary HSS?



s

Share

s_1 $\xrightarrow{\text{Eval}_f}$ y_1

s_2 $\xrightarrow{\text{Eval}_f}$ y_2

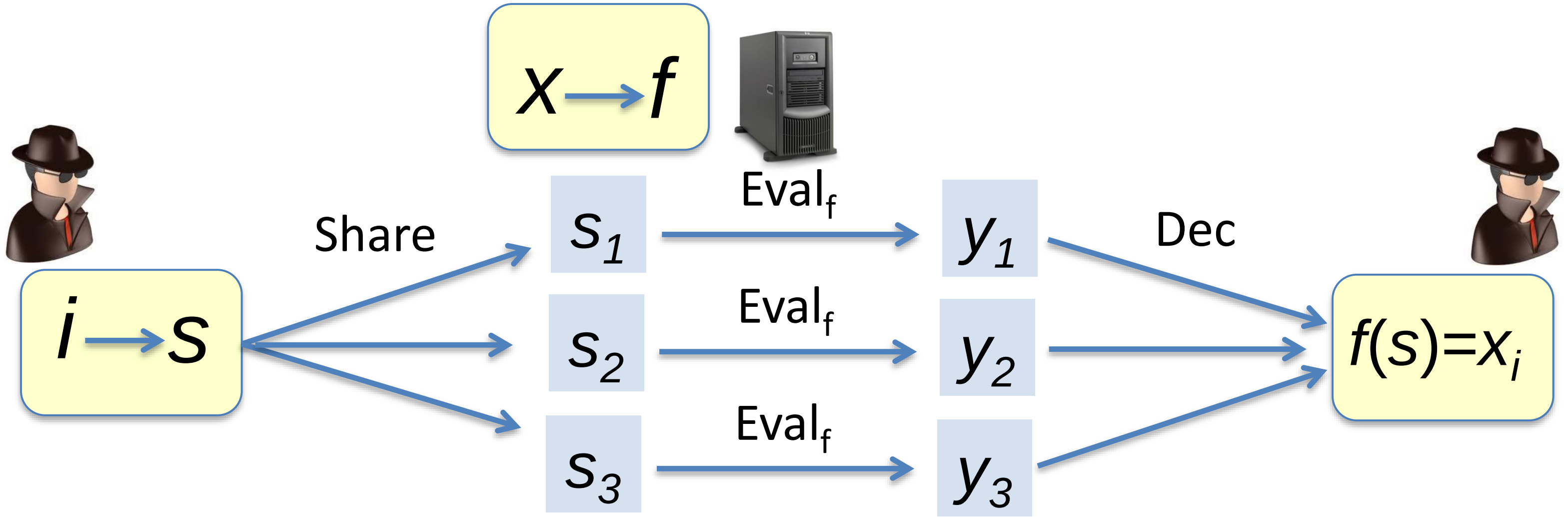
s_3 $\xrightarrow{\text{Eval}_f}$ y_3

Dec

$f(s)$

- **Function class F:** any set of $f:\{0,1\}^m \rightarrow \{0,1\}$
- **Input share size:** $\boldsymbol{\alpha(m)}$ (O(m) by default)
- **Output share size:** $\boldsymbol{\beta(m)}$ (O(1) by default)

PIR from arbitrary HSS?



$x \rightarrow f$

Share

Eval$_f$

Eval$_f$

Eval$_f$

Dec

$i \rightarrow s$

s_1

s_2

s_3

y_1

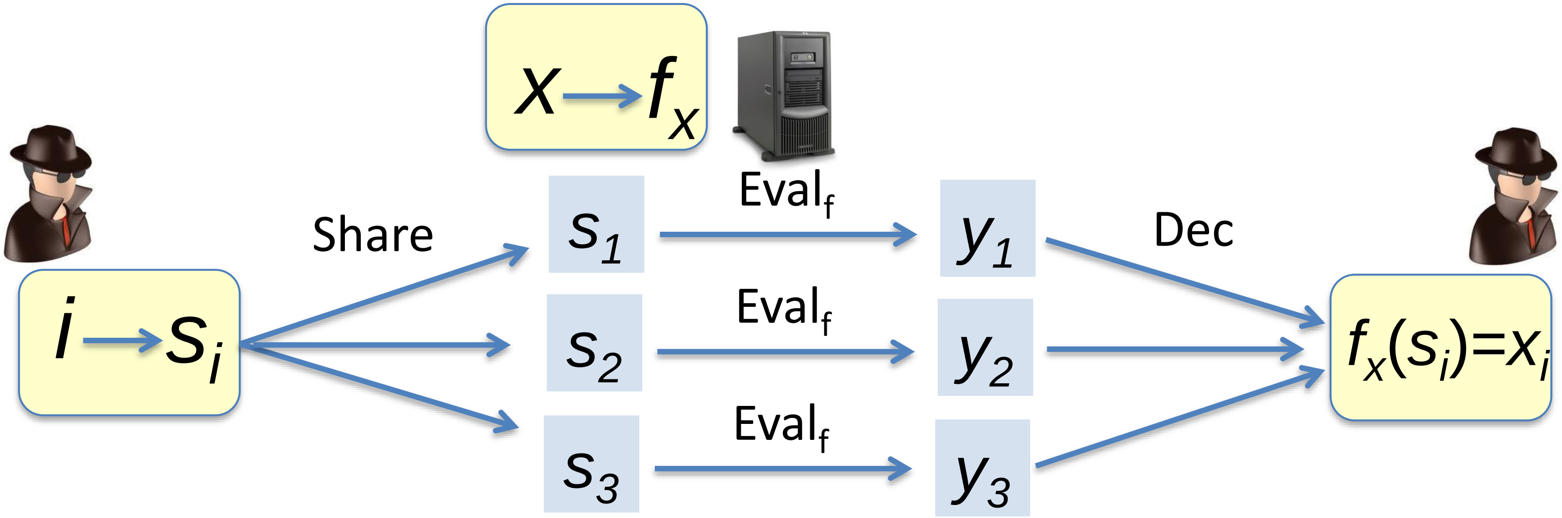
y_2

y_3

$f(s)=x_i$

- **Function class F:** any set of $f:\{0,1\}^m \rightarrow \{0,1\}$
- **Input share size:** $\boldsymbol{\alpha(m)}$ (O(m) by default)
- **Output share size:** $\boldsymbol{\beta(m)}$ (O(1) by default)

What should F satisfy?



$x \rightarrow f_x$

$i \rightarrow s_i$

Share

s_1 $\xrightarrow{\text{Eval}_f}$ y_1

s_2 $\xrightarrow{\text{Eval}_f}$ y_2

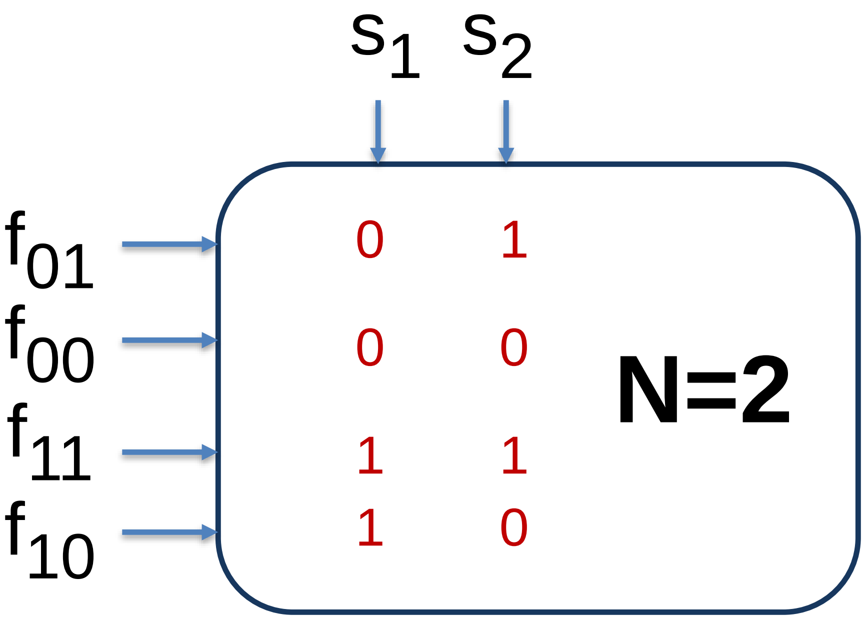
s_3 $\xrightarrow{\text{Eval}_f}$ y_3

Dec

$f_x(s_i)=x_i$

VC-dimension(F) ≥ N

∃ "shattered" input set $S=\{s_1,\ldots,s_N\}$ such that every $x:S\rightarrow\{0,1\}$ is a restriction of some $f_x \in F$ to S.



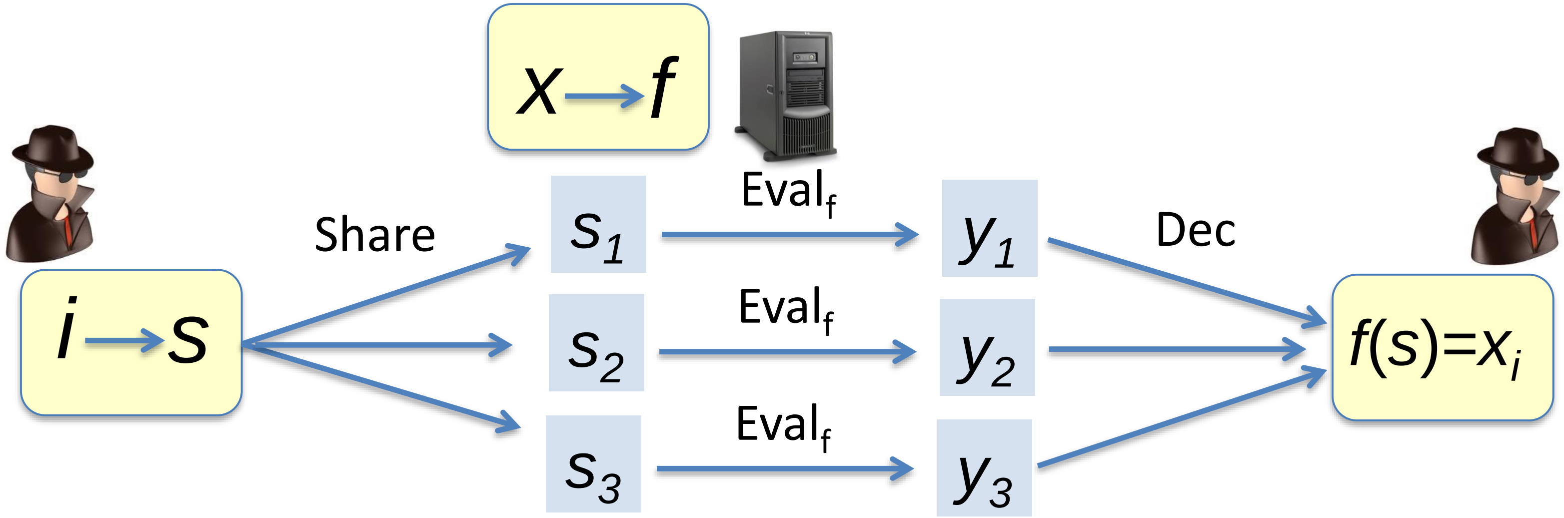
	s_1	s_2
f_{01}	0	1
f_{00}	0	0
f_{11}	1	1
f_{10}	1	0

N=2

VC-dimension(F) $\geq$ N

$\exists$ **“shattered” input set $S=\{s_1,\ldots,s_N\}$ such that every $x:S \rightarrow \{0,1\}$ is a restriction of some $f_x \in F$ to S.**

PIR from arbitrary HSS?



$x \longrightarrow f$

$i \longrightarrow s$

Share

s_1

s_2

s_3

Eval$_f$

Eval$_f$

Eval$_f$

y_1

y_2

y_3

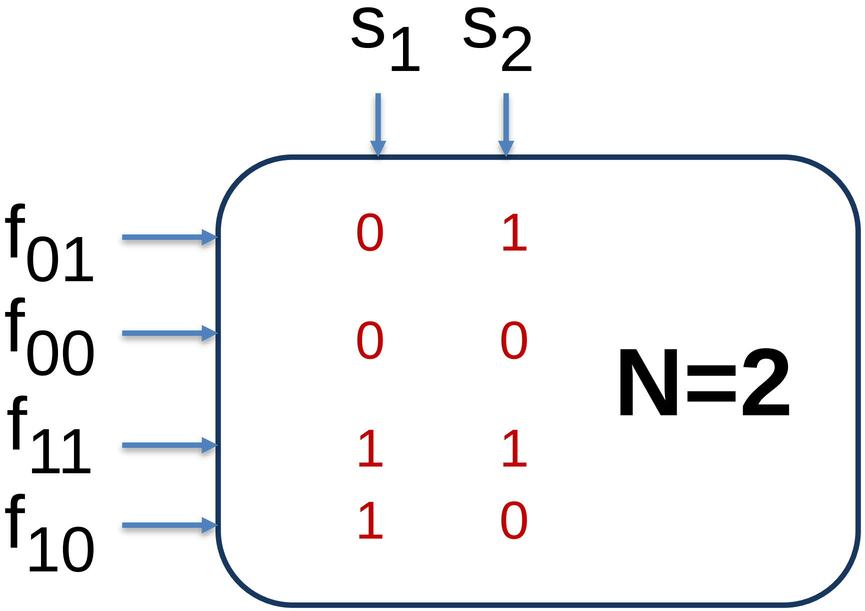
Dec

$f(s)=x_i$

- **Function class F:** any set of $f:\{0,1\}^m \rightarrow \{0,1\}$
- **Input share size:** $\boldsymbol{\alpha}$ (O(m) by default)
- **Output share size:** $\boldsymbol{\beta}$ (O(1) by default)

➔ PIR with N=VC-dim(F), α-bit queries, β-bit answers

Properties of VC dimension

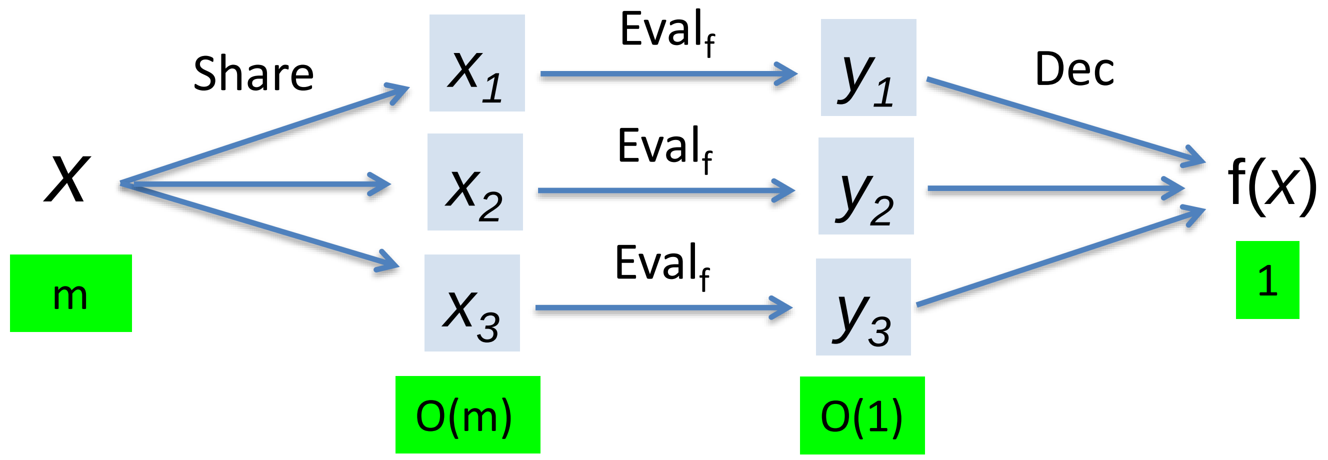


	s_1	s_2
f_{01}	0	1
f_{00}	0	0
f_{11}	1	1
f_{10}	1	0

N=2

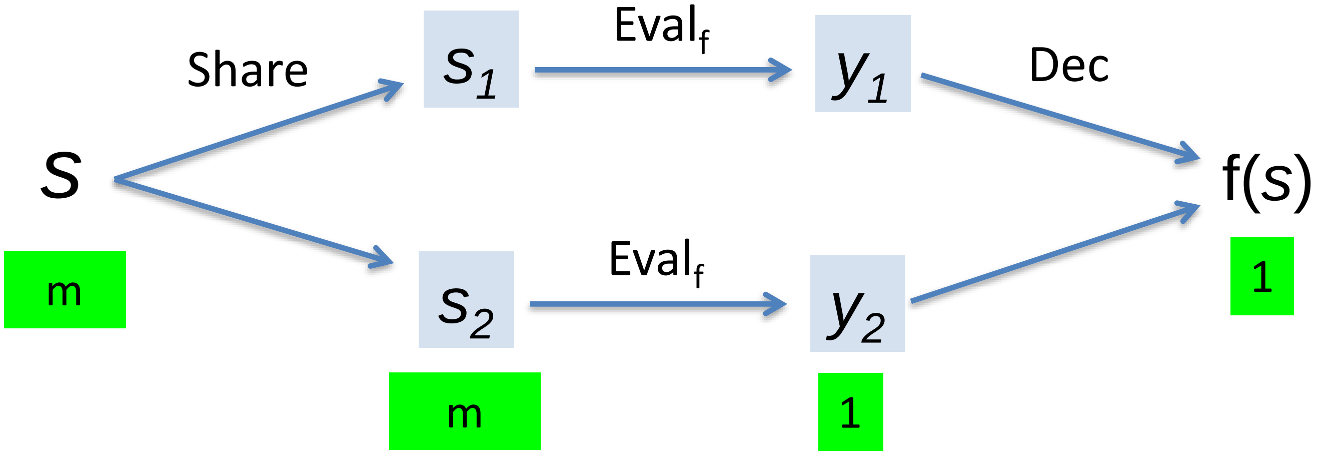
- If F is a linear space, VC-dim(F)=dim(F)
  - – F = deg-d polynomials over GF(2) => dim(F)=$O(m^d)$
  - – HSS for deg-d polynomials => PIR with N=$O(m^d)$
- Sauer lemma: For $|F|>>2^m$, VC-dim(F)=$\Omega(\log|F|)$
  - – HSS for really big F ➔ really good PIR!

The big question



- Given k, which F can be supported?

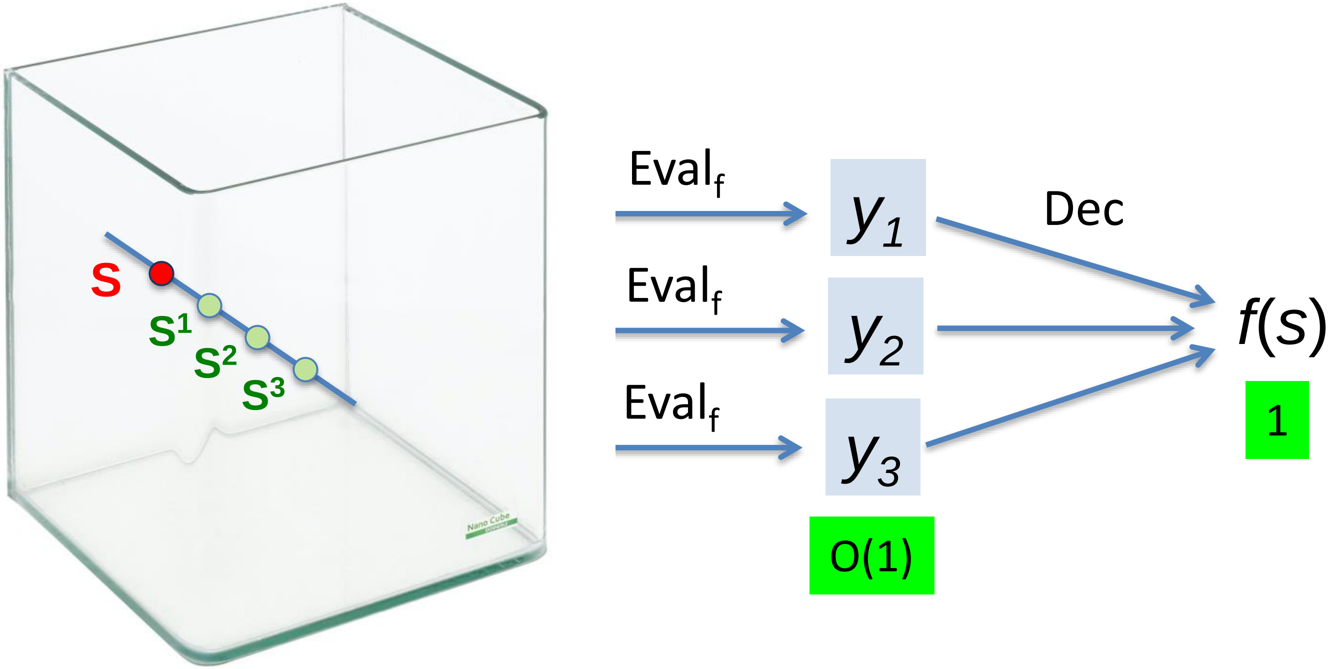
k=2



- F = linear functions $L: \mathbb{F}^m \to \mathbb{F}$
  - Share: additive secret sharing $s_1 + s_2 = s$
  - Eval$_L(s_i) = L(s_i)$
  - Recon$(\mathrm{y}_1, \mathrm{y}_2) = \mathrm{y}_1 + \mathrm{y}_2$

- VC-dim(F) = m
  - 2-server PIR with $\alpha = N,\ \beta = 1$
  - Essentially best possible with $\beta = 1$ [CGKS95,KT00,GKST02,BFG06]

k=3

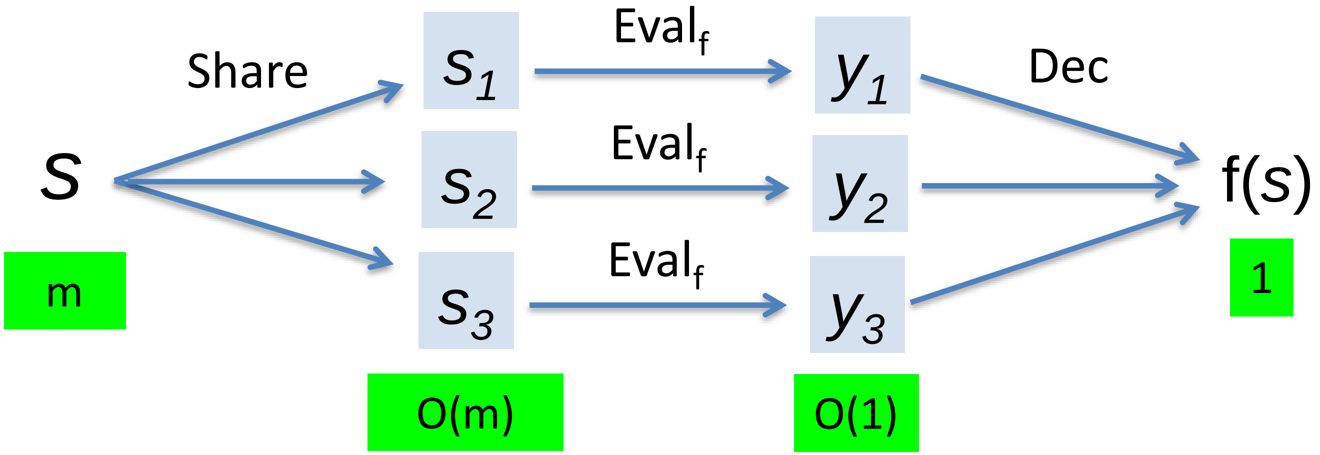


- F = degree-2 polynomials $p\colon \mathbb{F}^m \rightarrow \mathbb{F}$
  - Share: points on a random line passing through s
  - $\text{Eval}_p(s_i) = p(s_i)$
  - $\text{Recon}(y_1, y_2, y_3) = \ P(0)$ for

What else can we do?

- VC-dim(F) = $O(m^2)$
  - 3-server PIR with $\alpha = O(\sqrt{N}),\ \beta = 1$
  - Until 2007, conjectured to be best possible

k=3



Share

Eval$_f$

Dec

S

s_1

s_2

s_3

y_1

y_2

y_3

f(s)

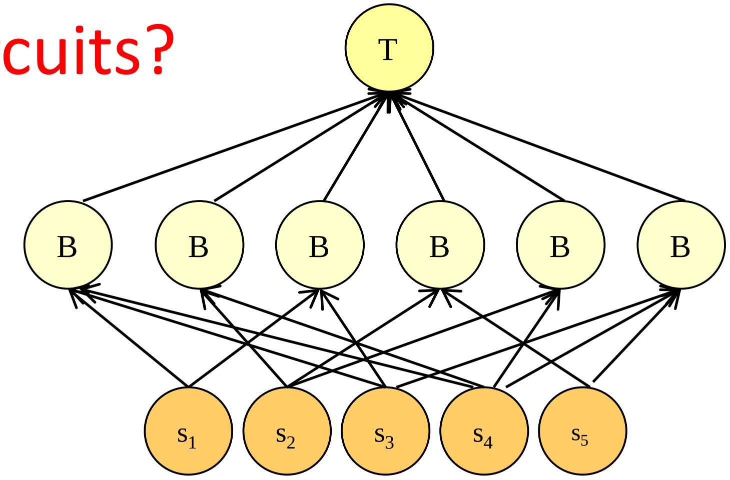
m

O(m)

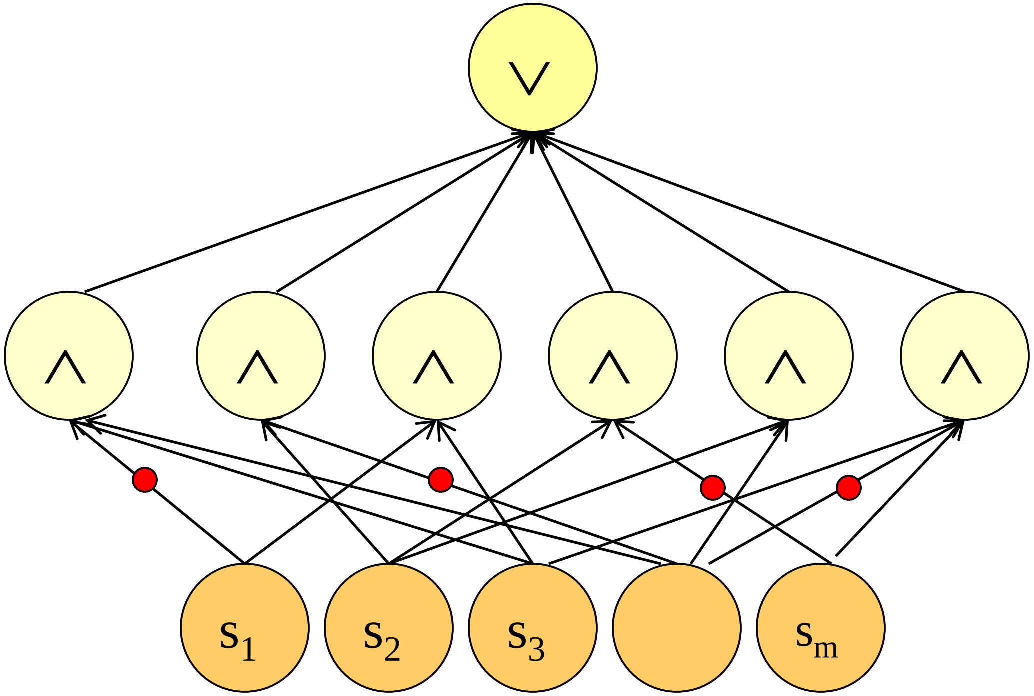
O(1)

1

- F = exotic class of depth-2 circuits?
  - B,T = symmetric gates

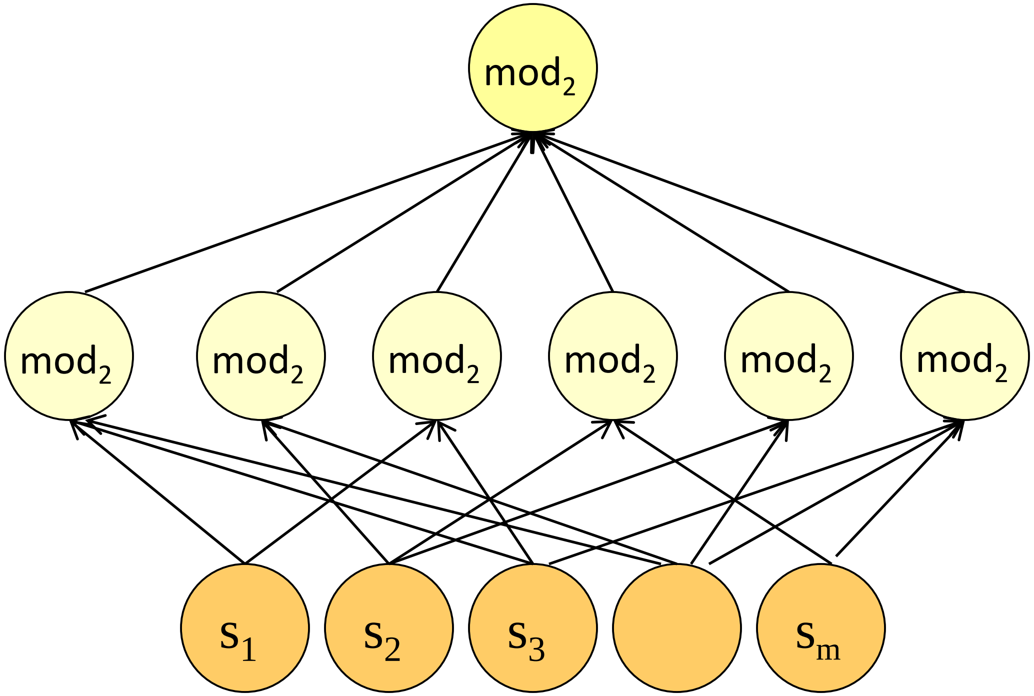


Power of Depth-2 Circuits



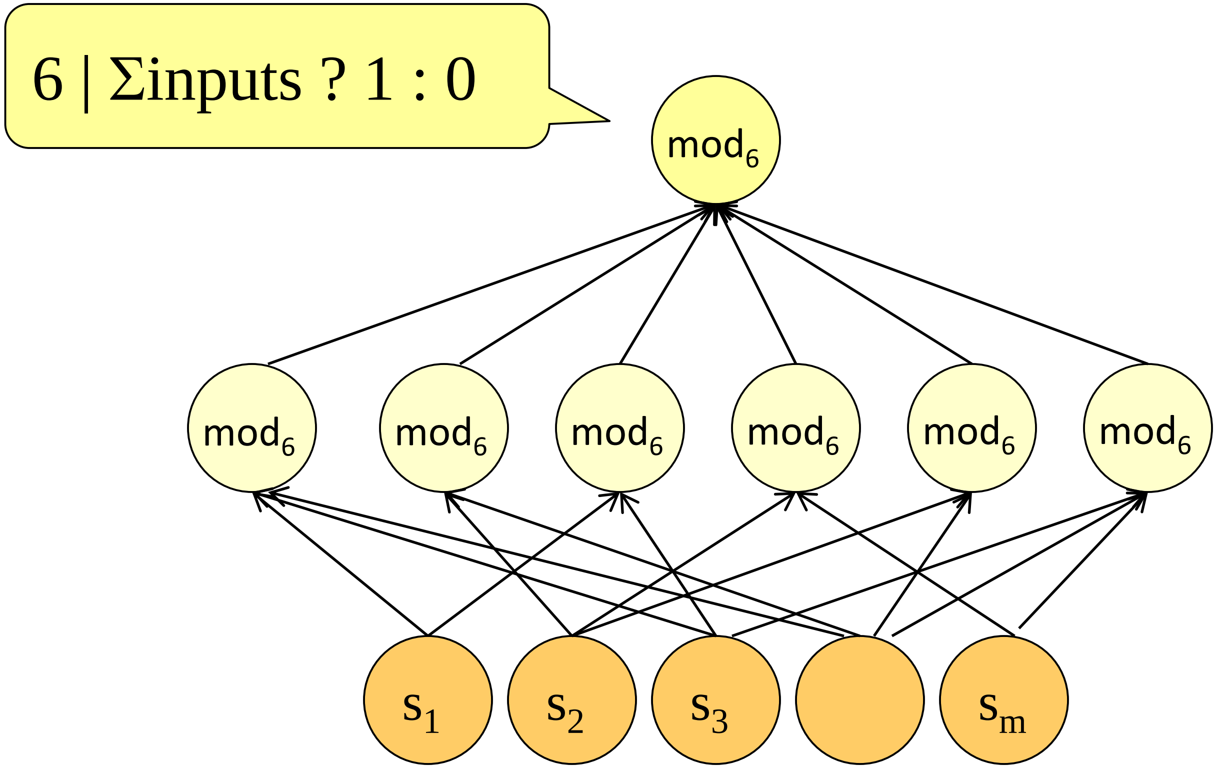
All 2^{2^m} functions

Power of Depth-2 Circuits



Only 2^m functions

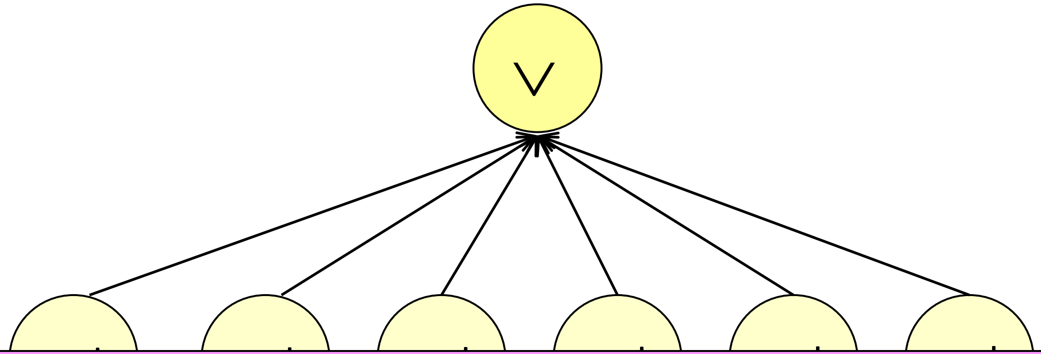
Power of Depth-2 Circuits



6 | Σinputs ? 1 : 0

All 2^{2^m} functions!

Power of Depth-2 Circuits



Related to size of:

- Set systems with restricted intersections [BF80, Gromlusz00]
- Matching vector sets [Yekanin07,Efremenko09,DvirGopalanYekhanin10]
- Degree of representing “OR” modulo 6 [BarringtonBeigelRudich92]

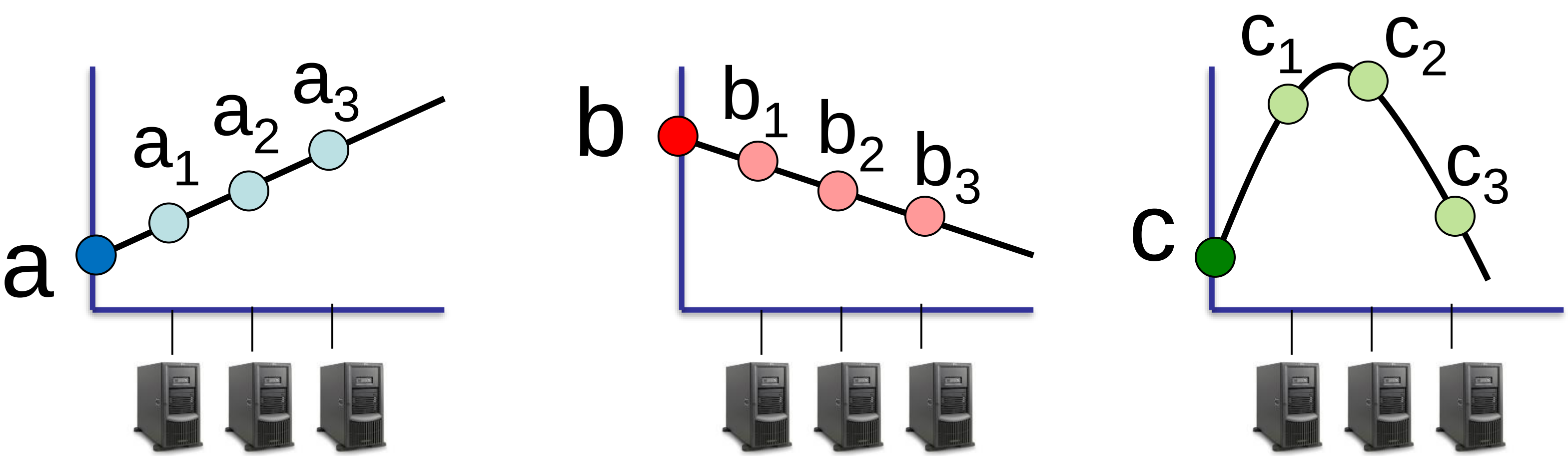
$2^{m^{\log m}}$ <

any functions!

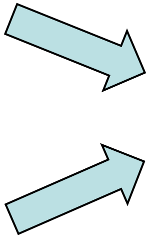
$<<2^{2^m}$

How many ???

Another view of deg-2 HSS:
What can we compute with Shamir?



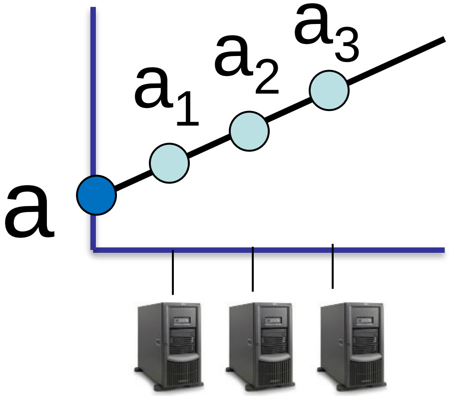
- Local addition
  - – Does not increase degree
- Local multiplication
  - – Increases degree to 2 (ok!)
  - – Outputs can be added

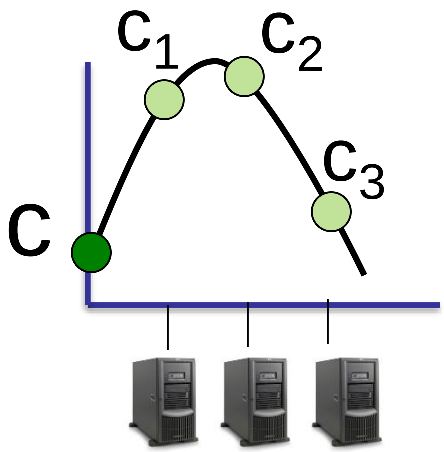


HSS
for deg-2
polynomials

B: $\times_2$ T: +

Yet another view:
Squaring is enough





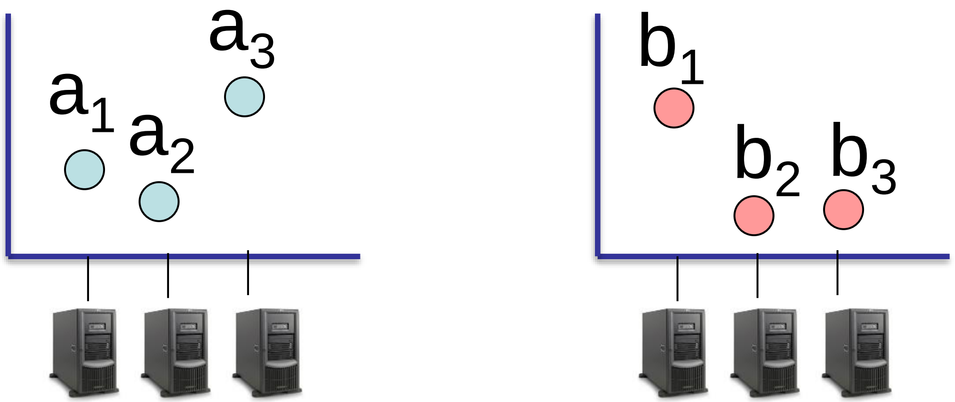
- Local addition
  - Does not increase degree
- Local squaring
  - Increases degree to 2
  - Outputs can be added

HSS
for deg-2
polynomials

B: SQ T: +

Going Crazy?

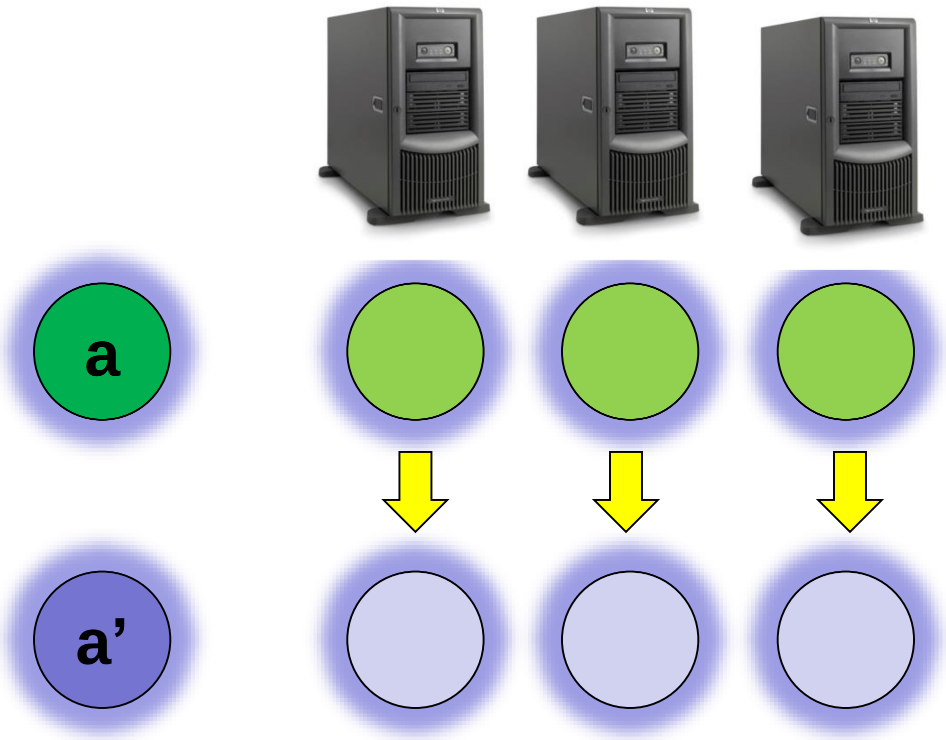
- CRAZY secret sharing



- CRAZY computations on shares ⇨ HSS for CRAZY functions

- Problem: Dec output will depend not only on inputs, but also on randomness of Share.

Share Conversion



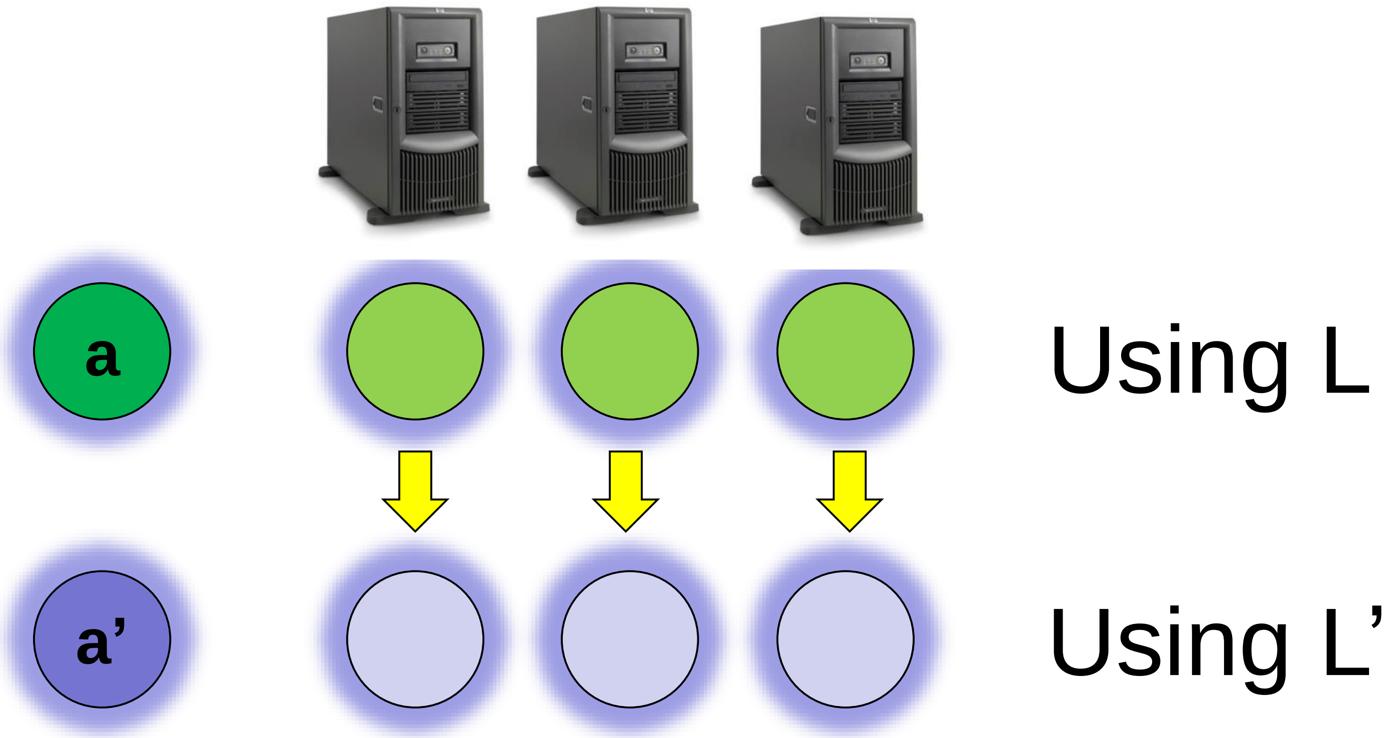
Using L

Using L’

(a,a’) satisfy a given relation

B: output a’ for a=sum of inputs T: + / OR

Which L and L' to choose?



(a,a') satisfy a given relation

Which L and L’ to choose?

[Cramer-Damgård-I05]: a’=a

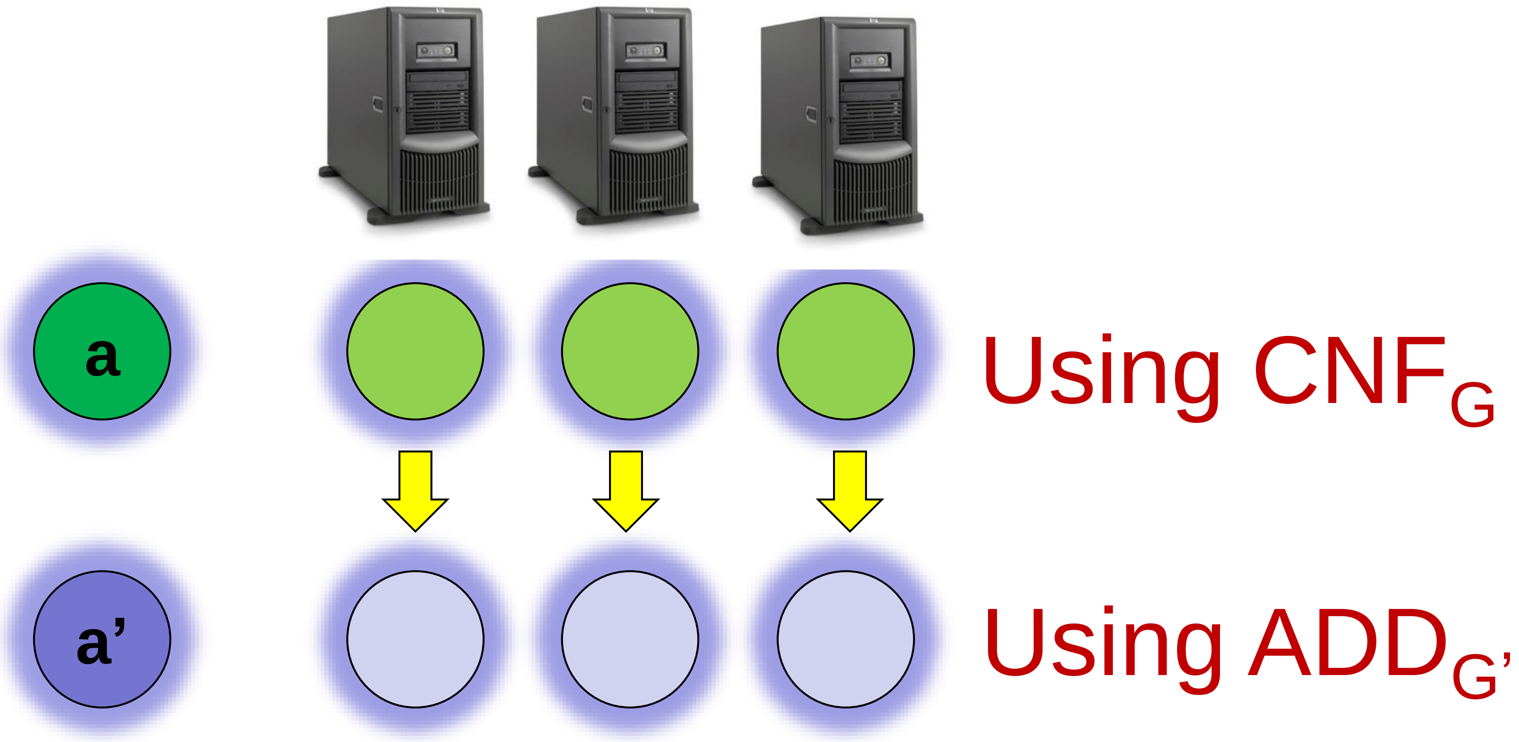
“CNF secret-sharing” is maximal
“DNF secret-sharing” is minimal

Using L

Using L’

(a,a’) satisfy a given relation

Which L and L’ to choose?



Using CNF$_G$

Using ADD$_{G'}$

(a,a’) satisfy a given relation

Applying Share Conversion

- Which circuit classes can we realize?
  - deg-2 polynomials VC-dim = m^2
  - $\mathrm{OR}^\circ\mathrm{mod}_6$ VC-dim = ??

Requires either:

- k>3 servers, or
- Promise on the Hamming weight of inputs for gates ”S-matching vectors” – Klim’s talk

Applying Share Conversion

- Which circuit classes can we realize?
  - – deg-2 polynomials VC-dim = m^2
  - – $\mathrm{OR}^\circ\mathrm{mod}_c$ VC-dim = $m^{\Omega(\log m)}$

- Efremenko09: c=511 conversion from Shamir’
- BIKO12: c=6 conversion from CNF

Improves constant in exponent

Applying Share Conversion

- Which circuit classes can we realize?
  - deg-2 polynomials VC-dim = m^2
  - $\mathrm{OR}\circ\mathrm{mod}_c$ VC-dim = $m^{\Omega(\log m)}$
  - $\mathrm{OR}\circ\mathrm{AND}_d\circ\mathrm{mod}_c$ not much better…

- Wishful thinking: logarithmic PIR
  - $\mathrm{mod}_6\circ\mathrm{mod}_6$ VC-dim = 2^m
  - suitable share conversion can be ruled out

A Practical Instance?

- 3 Servers, database size N
- Communication
  - – Client: $7N^{1/4}$-bit queries (compare with $1.4N^{1/2}$)
    - Feasible also for a virtual database of hash values
  - – Servers: 2-bit answers ((b+1) bits for b-bit records)
- Computation
  - – Servers: 54 XORs for each nonzero record
  - – Client: takes XOR of 3 answers

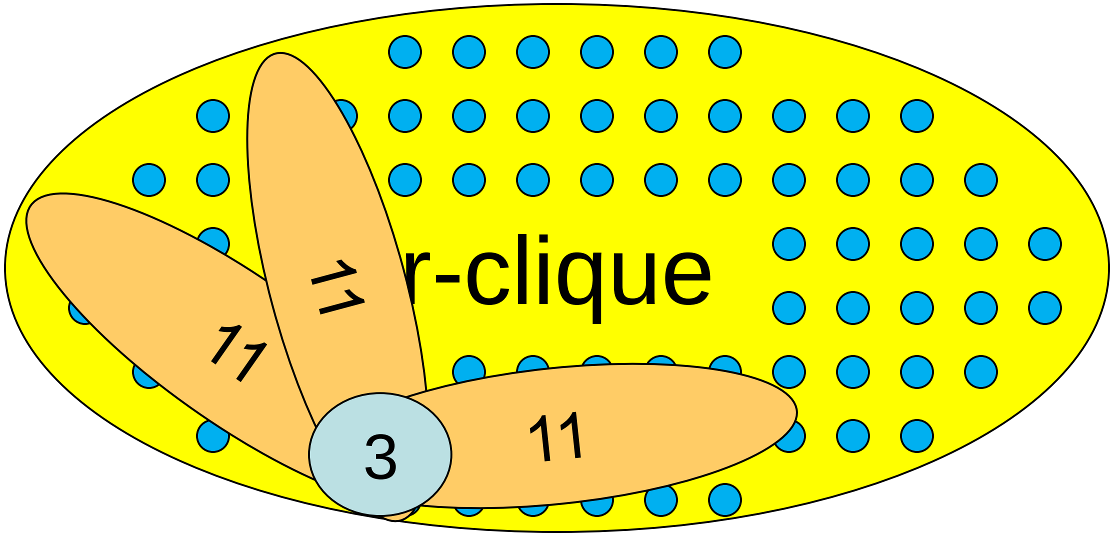
Secret Sauce I:

Big Set System with Limited mod-6 Intersections

- Goal: find N subsets T_i of [h] such that:
  - $|T_i| \equiv 1 \quad (\text{mod } 6)$
  - $|T_i \cap T_j| \in \{0,3,4\} \quad (\text{mod } 6)$
- h = query length; N = database size
- [Frankl83]: $h=\binom{r}{2}$, $N=\binom{r-3}{8}$
  - $h \approx 7N^{1/4}$
- Better asymptotic constructions: Klim’s talk

Secret Sauce I:

Big Set System with Limited mod-6 Intersections



h=$\binom{r}{2}$; N=$\binom{r-3}{8}$; $|T_i|$=$\binom{11}{2}$=55≡1 (mod 6)

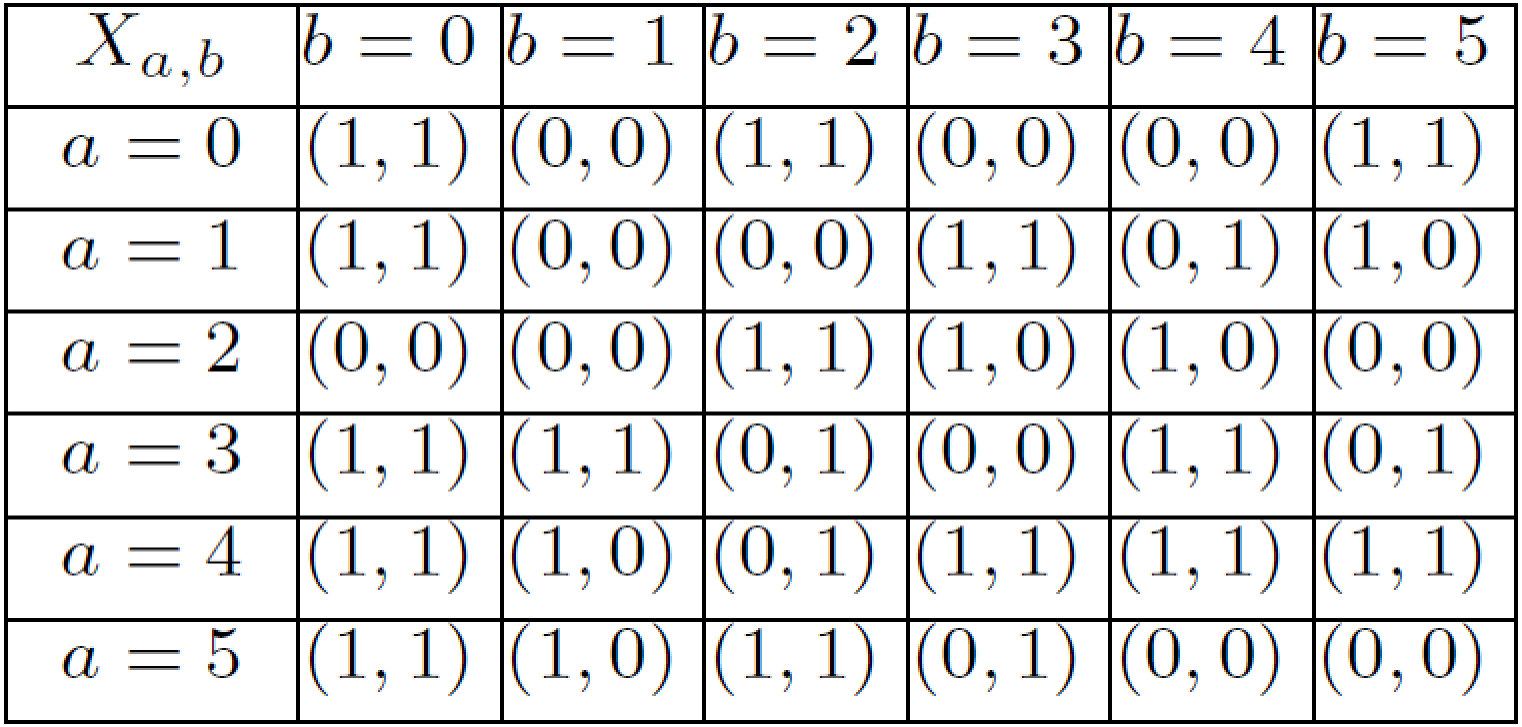
$|T_i \cap T_j|$=$\binom{t}{2}$, 3≤t ≤10 ∈ {0,3,4} (mod 6)

Secret Sauce II:

Convert CNF over Z_6 to ADD over $Z_2{}^2$

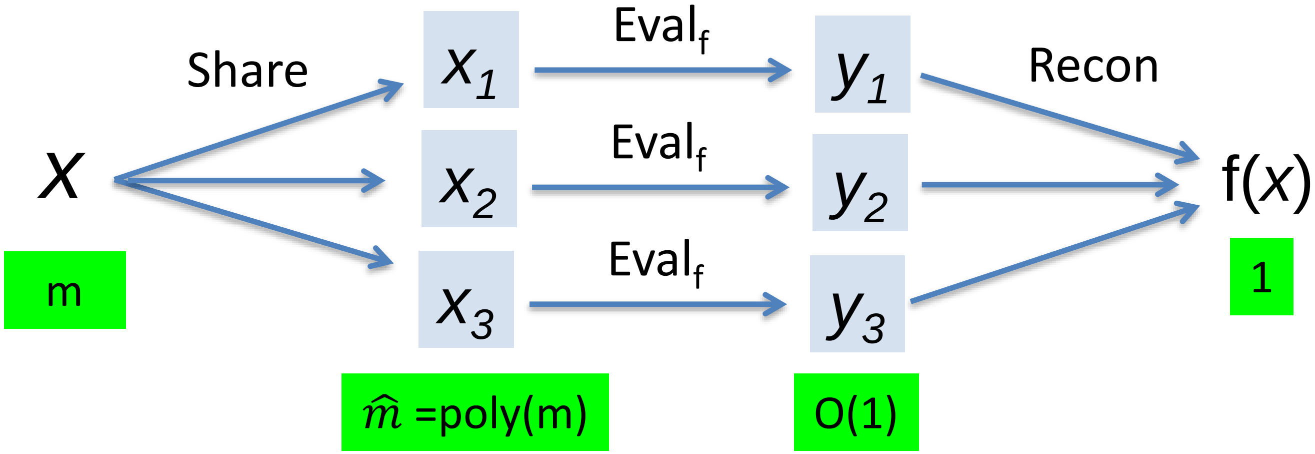
a=0 ➔ a’≠0

a=1,3,4 ➔ a’=0

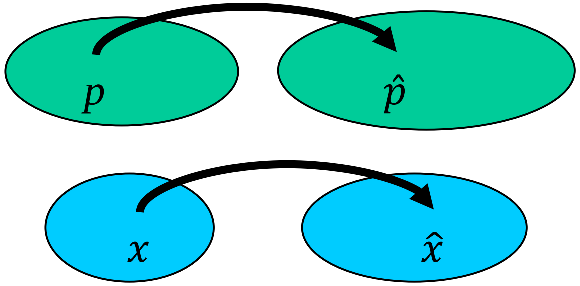


$X_{a,b}$	$b=0$	$b=1$	$b=2$	$b=3$	$b=4$	$b=5$
$a=0$	$(1,1)$	$(0,0)$	$(1,1)$	$(0,0)$	$(0,0)$	$(1,1)$
$a=1$	$(1,1)$	$(0,0)$	$(0,0)$	$(1,1)$	$(0,1)$	$(1,0)$
$a=2$	$(0,0)$	$(0,0)$	$(1,1)$	$(1,0)$	$(1,0)$	$(0,0)$
$a=3$	$(1,1)$	$(1,1)$	$(0,1)$	$(0,0)$	$(1,1)$	$(0,1)$
$a=4$	$(1,1)$	$(1,0)$	$(0,1)$	$(1,1)$	$(1,1)$	$(1,1)$
$a=5$	$(1,1)$	$(1,0)$	$(1,1)$	$(0,1)$	$(0,0)$	$(0,0)$

An intriguing HSS question



- How big should $\hat{m}$ be for F = degree-3 polynomials?
- Natural approach: reduce to degree-2 case
  - Embedding degree 3 to degree 2:
    - Map deg-3 $p[X_1, \dots, X_m] \rightarrow$ deg-2 $\hat{p}[X_1, \dots, X_{\hat{m}}]$
    - Map $x \in \mathbb{F}^m \rightarrow \hat{x} \in \mathbb{F}^{\hat{m}}$
    - $p(x) = \hat{p}(\hat{x})$
  - How big should $\hat{m}$ be in such an embedding?
    - Gap between easy bounds: $\Omega(m^{1.5}) \leq \hat{m} \leq O(m^2)$



deg 3, m vars

deg 2, $\hat{m}$ vars

Open Questions

- Improve upper bounds for IT PIR
  - polylog(N) with constant k?
  - Beat O($N^{1/k}$) in short-query regime?
- Understand power of IT HSS
  - New classes via new share conversions?
  - Other use cases for $\mathrm{OR}^{\circ}\mathrm{mod}_c$ circuits?
- Improved t-private PIR with $N^{o(1)}$ communication
  - 3^t servers [Barkol-I-Weinreb08] (short answers)
  - 2^t servers [BIW08] + [Dvir-Gopi15] (balanced)
- Better lower bounds
  - Any fundamental barriers?