On Basing Private Information Retrieval on NP-Hardness

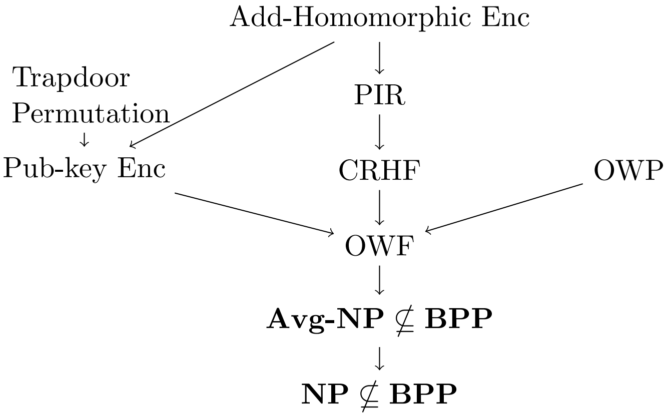
Tianren Liu[1] Vinod Vaikuntanathan[1]

[1]MIT CSAIL

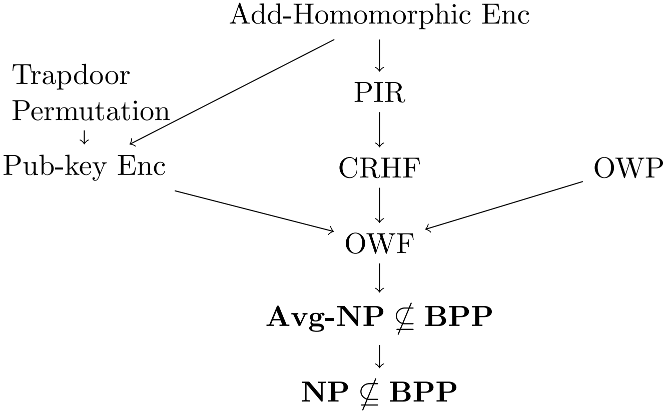
liutr@mit.edu, vinodv@csail.mit.edu

Thirteenth IACR Theory of Cryptography Conference

Assumptions and Primitives in Cryptography



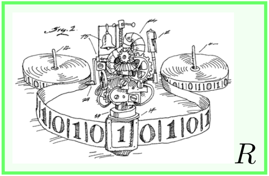
Assumptions and Primitives in Cryptography



Can we prove the security of a cryptographic primitive from the minimal assumption **NP** $\not\subseteq$ **BPP**? (Brassard 1979)

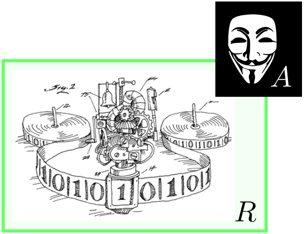
(Black-box) Security Proofs

To prove the security of X based on $\mathbf{NP} \not\subseteq \mathbf{BPP}$, find a (p.p.t.) reduction R s.t. for any oracle A that "breaks the security of X", R^A solves SAT



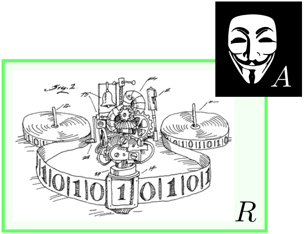
(Black-box) Security Proofs

To prove the security of X based on $\mathbf{NP} \not\subseteq \mathbf{BPP}$, find a (p.p.t.) reduction R s.t. for any oracle A that "breaks the security of X", R^A solves SAT



(Black-box) Security Proofs

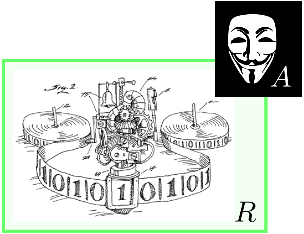
To prove the security of X based on $\mathbf{NP} \not\subseteq \mathbf{BPP}$, find a (p.p.t.) reduction R s.t. for any oracle A that "breaks the security of X", R^A solves SAT



$$(x) \begin{cases} \text{accepts w.p. } \geq 2/3, & \text{if } x \in \mathsf{SAT} \\ \text{accepts w.p. } \leq 1/3, & \text{if } x \notin \mathsf{SAT} \end{cases}$$

(Black-box) Security Proofs

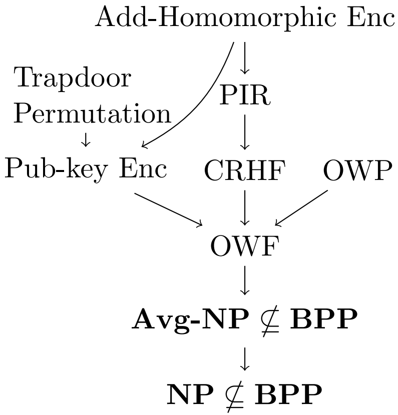
To prove the security of X based on $\mathbf{NP} \not\subseteq \mathbf{BPP}$, find a (p.p.t.) reduction R s.t. for any oracle A that "breaks the security of X", R^A solves SAT



$$(x) \begin{cases} \text{accepts w.p.} \geq 2/3, & \text{if } x \in \mathsf{SAT} \\ \text{accepts w.p.} \leq 1/3, & \text{if } x \notin \mathsf{SAT} \end{cases}$$

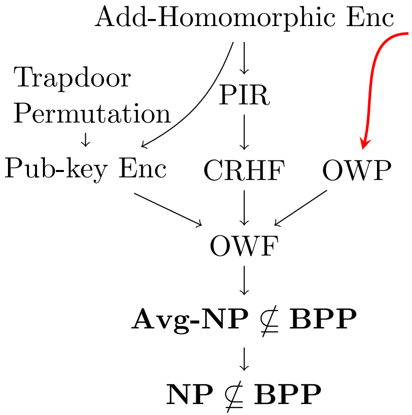
- Note: Black-box security proof but allow arbitrary construction.

Impossibility Results



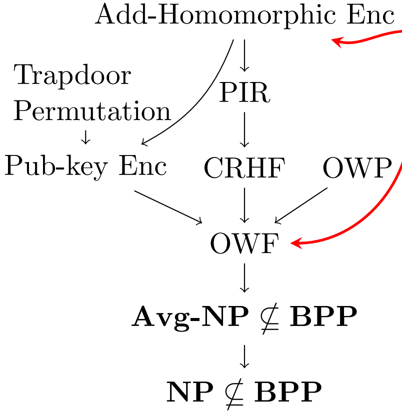
- No known cryptographic scheme based on **NP** $\not\subseteq$ **BPP**.
- Several negative results* (Brassard 1979, ...)

Impossibility Results



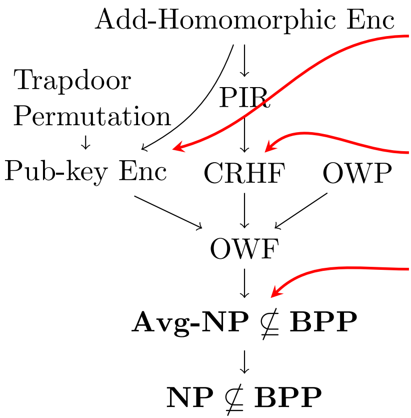
- One-way Permutations (Brassard 1979)

Impossibility Results (restricting the primitives)



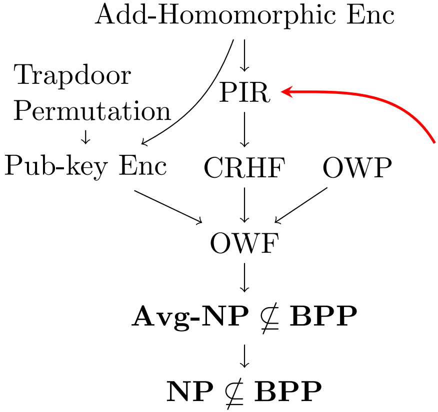
- Homomorphic Encryption*
  (Bogdanov-Lee 2013)
- One-way Functions*
  (Akavia-Goldreich-Goldwasser-Moshkovitz 2006, Bogdanov-Brzuska 2014)

Impossibility Results (restricting the reductions)



- Public-key Encryption Scheme, via “smart” reduction (Goldreich-Goldwasser 1998)
- Collision-resistant Hash Functions, via constant-adaptive reduction (Haitner-Mahmoody-Xiao 2009)
- Average-case NP, via non-adaptive reduction (Bogdanov-Trevisan 2006)

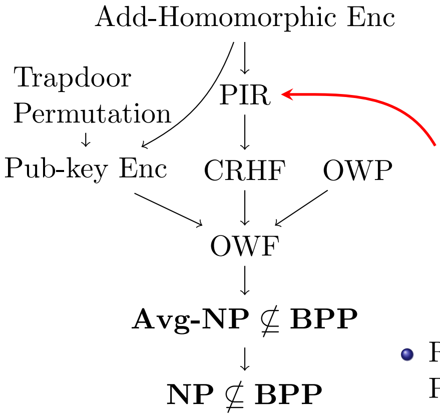
Our Result: Private Information Retrieval [CGKS95, KO97]



Theorem (Informal)

Let Π be a single-server one-round PIR scheme. Security of Π can not be based on NP-hardness unless polynomial hierarchy collapses.

Our Result: Private Information Retrieval [CGKS95, KO97]



Theorem (Informal)

Let Π be a single-server one-round PIR scheme. Security of Π can not be based on NP-hardness unless polynomial hierarchy collapses.

- Rule out approximately correct PIR.
- Rule out PIR with communication complexity $n - o(n)$.

Proof Overview

Lemma 1 (Single-server one-round) PIR can be broken with an SZK oracle

Lemma 2 Language $L \in \mathbf{BPP}^{\mathbf{SZK}} \implies L \in \mathbf{AM} \cap \mathbf{coAM}$ (Mahmoody & Xiao, 2010)

Thus: if there is a reduction from SAT to breaking PIR, then SAT $\in \mathbf{AM} \cap \mathbf{coAM}$.

Lemma 3 $\mathbf{NP} \not\subseteq \mathbf{coAM}$ unless polynomial hierarchy collapses (Boppana, Håstad & Zachos, 1987)

Thus: if there is a reduction from SAT to breaking PIR, then polynomial hierarchy collapses.

Proof Overview

Lemma 1 (Single-server one-round) PIR can be broken with an SZK oracle

Lemma 2 Language $L \in \mathbf{BPP}^{\mathbf{SZK}} \implies L \in \mathbf{AM} \cap \mathbf{coAM}$
(Mahmoody & Xiao, 2010)

Thus: if there is a reduction from SAT to breaking PIR, then SAT $\in \mathbf{AM} \cap \mathbf{coAM}$.

Lemma 3 $\mathbf{NP} \not\subseteq \mathbf{coAM}$ unless polynomial hierarchy collapses
(Boppana, Håstad & Zachos, 1987)

Thus: if there is a reduction from SAT to breaking PIR, then polynomial hierarchy collapses.

Proof Overview

Lemma 1 (Single-server one-round) PIR can be broken with an SZK oracle

Lemma 2 Language $L \in \mathbf{BPP}^{\mathbf{SZK}} \implies L \in \mathbf{AM} \cap \mathbf{coAM}$

(Mahmoody & Xiao, 2010)

Thus: if there is a reduction from SAT to breaking PIR, then SAT $\in \mathbf{AM} \cap \mathbf{coAM}$.

Lemma 3 $\mathbf{NP} \not\subseteq \mathbf{coAM}$ unless polynomial hierarchy collapses

(Boppana, Håstad & Zachos, 1987)

Thus: if there is a reduction from SAT to breaking PIR, then polynomial hierarchy collapses.

Proof Overview

Lemma 1 (Single-server one-round) PIR can be broken with an SZK oracle

Lemma 2 Language $L \in \mathbf{BPP}^{\mathbf{SZK}} \implies L \in \mathbf{AM} \cap \mathbf{coAM}$
(Mahmoody & Xiao, 2010)

Thus: if there is a reduction from SAT to breaking PIR, then $\mathsf{SAT} \in \mathbf{AM} \cap \mathbf{coAM}$.

Lemma 3 $\mathbf{NP} \not\subseteq \mathbf{coAM}$ unless polynomial hierarchy collapses
(Boppana, Håstad & Zachos, 1987)

Thus: if there is a reduction from SAT to breaking PIR, then polynomial hierarchy collapses.

Proof Overview

Lemma 1 (Single-server one-round) PIR can be broken with an SZK oracle

Lemma 2 Language $L \in \mathbf{BPP}^{\mathbf{SZK}} \implies L \in \mathbf{AM} \cap \mathbf{coAM}$

(Mahmoody & Xiao, 2010)

Thus: if there is a reduction from SAT to breaking PIR, then SAT $\in \mathbf{AM} \cap \mathbf{coAM}$.

Lemma 3 $\mathbf{NP} \not\subseteq \mathbf{coAM}$ unless polynomial hierarchy collapses

(Boppana, Håstad & Zachos, 1987)

Thus: if there is a reduction from SAT to breaking PIR, then polynomial hierarchy collapses.

Proof Overview

Lemma 1 (Single-server one-round) PIR can be broken with an SZK oracle

Lemma 2 Language $L \in \mathbf{BPP}^{\mathbf{SZK}} \implies L \in \mathbf{AM} \cap \mathbf{coAM}$
(Mahmoody & Xiao, 2010)

Thus: if there is a reduction from SAT to breaking PIR, then $\mathsf{SAT} \in \mathbf{AM} \cap \mathbf{coAM}$.

Lemma 3 $\mathbf{NP} \not\subseteq \mathbf{coAM}$ unless polynomial hierarchy collapses
(Boppana, Håstad & Zachos, 1987)

Thus: if there is a reduction from SAT to breaking PIR, then polynomial hierarchy collapses.

Proof Overview

Lemma 1 (Single-server one-round) PIR can be broken with an SZK oracle

Lemma 2 Language $L \in \mathbf{BPP}^{\mathbf{SZK}} \implies L \in \mathbf{AM} \cap \mathbf{coAM}$
(Mahmoody & Xiao, 2010)

Thus: if there is a reduction from SAT to breaking PIR, then SAT $\in \mathbf{AM} \cap \mathbf{coAM}$.

Lemma 3 $\mathbf{NP} \not\subseteq \mathbf{coAM}$ unless polynomial hierarchy collapses
(Boppana, Håstad & Zachos, 1987)

Thus: if there is a reduction from SAT to breaking PIR, then polynomial hierarchy collapses.

Proof Overview

Lemma 1 (Single-server one-round) PIR can be broken with an SZK oracle

Lemma 2 Language $L \in \mathbf{BPP}^{\mathbf{SZK}} \implies L \in \mathbf{AM} \cap \mathbf{coAM}$
(Mahmoody & Xiao, 2010)

Thus: if there is a reduction from SAT to breaking PIR, then SAT $\in \mathbf{AM} \cap \mathbf{coAM}$.

Lemma 3 $\mathbf{NP} \not\subseteq \mathbf{coAM}$ unless polynomial hierarchy collapses
(Boppana, Håstad & Zachos, 1987)

Thus: if there is a reduction from SAT to breaking PIR, then polynomial hierarchy collapses.

Single-server *One-round* Private Information Retrieval

Client

- Index $i \in \{1, \ldots, n\}$

One Server

- Data $\mathbf{x} \in \{0, 1\}^n$

Single-server *One-round* Private Information Retrieval

Client

- Index $i \in \{1, \ldots, n\}$
- Client send a query $\xrightarrow{\quad q \quad}$

One Server

- Data $\mathbf{x} \in \{0, 1\}^n$

Single-server *One-round* Private Information Retrieval

Client

- Index $i \in \{1, \ldots, n\}$
- Client send a query $\xrightarrow{q}$

One Server

- Data $\mathbf{x} \in \{0,1\}^n$
- $\xleftarrow{a}$ Server answer

Single-server *One-round* Private Information Retrieval

Client

- Index $i \in \{1, \dots, n\}$
- Client send a query $\xrightarrow{q}$

One Server

- Data $\mathbf{x} \in \{0,1\}^n$
- $\xleftarrow{a}$ Server answer

- Correctness:
  The client learns x_i

Single-server *One-round* Private Information Retrieval

Client

- Index $i \in \{1, \ldots, n\}$
- Client send a query $\xrightarrow{\quad q \quad}$

One Server

- Data $\mathbf{x} \in \{0,1\}^n$
- $\xleftarrow{\quad a \quad}$ Server answer

- Correctness:
  The client learns x_i
  (W.p. $1 - \varepsilon$.)

Single-server *One-round* Private Information Retrieval

Client

- Index $i \in \{1, \dots, n\}$
- Client send a query $\xrightarrow{q}$

One Server

- Data $\mathbf{x} \in \{0,1\}^n$
- $\xleftarrow{a}$ Server answer

Client

- Correctness:
  The client learns x_i
  (W.p. $1 - \varepsilon$.)

One Server

- Privacy:
  The server learn nothing about i

Single-server *One-round* Private Information Retrieval

Client

- Index $i \in \{1, \ldots, n\}$
- Client send a query $\xrightarrow{q}$

One Server

- Data $\mathbf{x} \in \{0,1\}^n$
- $\xleftarrow{a}$ Server answer

Client

- Correctness:
  The client learns x_i
  (W.p. $1 - \varepsilon$.)

One Server

- Privacy:
  The server learn nothing about i

An Oracle Breaking Single-server One-round PIR

Given a query q, guess the index with probability $> 1/n + 1/\text{poly}$.

Break PIR with SZK oracle (Lemma 1)

Client

- Index $i \in \{1, \ldots, n\}$
- Generate a query $\xrightarrow{\quad q \quad}$

Break PIR with SZK oracle (Lemma 1)

Client

- Index $i \in \{1, \ldots, n\}$
- Generate a query $\xrightarrow{q}$

Server

- Random $\mathbf{x} \in \{0,1\}^n$

Break PIR with SZK oracle (Lemma 1)

Client

- Index $i \in \{1, \dots, n\}$
- Generate a query $\xrightarrow{q}$

Server

- Random $\mathbf{x} \in \{0,1\}^n$
- $\xleftarrow{a}$ Server answers

Break PIR with SZK oracle (Lemma 1)

Client		Server
• Index $i \in \{1, \ldots, n\}$		• Random $\mathbf{x} \in \{0,1\}^n$
• Generate a query	$\xrightarrow{q}$	
	$\xleftarrow{a}$	• Server answers

- **Claim 1:** $I(x_i; a)$ is big*.

*The randomness is from $\mathbf{x}$ and from the proceduce generating the answer.

Break PIR with SZK oracle (Lemma 1)

Client

- Index $i \in \{1, \ldots, n\}$
- Generate a query $\xrightarrow{q}$

Server

- Random $\mathbf{x} \in \{0,1\}^n$
- $\xleftarrow{a}$ Server answers

- **Claim 1:** $I(x_i; a)$ is big*.

  Proof: Correctness. □

*The randomness is from $\mathbf{x}$ and from the proceduce generating the answer.

Break PIR with SZK oracle (Lemma 1)

Client

- Index $i \in \{1, \ldots, n\}$
- Generate a query $\xrightarrow{q}$

Server

- Random $\mathbf{x} \in \{0,1\}^n$
- $\xleftarrow{a}$ Server answers

- **Claim 1:** $I(x_i; a) = 1$ assuming perfect correctness

  Proof: Correctness. $\square$

*The randomness is from $\mathbf{x}$ and from the proceduce generating the answer.

Break PIR with SZK oracle (Lemma 1)

Client

- Index $i \in \{1, \dots, n\}$
- Generate a query $\xrightarrow{q}$

Server

- Random $\mathbf{x} \in \{0,1\}^n$
- $\xleftarrow{a}$ Server answers

- **Claim 1:** $I(x_i; a) = 1$ assuming perfect correctness
  Proof: Correctness. $\square$
- **Claim 2:** $\sum_{j=1}^{n} I(x_j; a) \leq H(a) \leq |a|$.

*The randomness is from $\mathbf{x}$ and from the proceduce generating the answer.

Break PIR with SZK oracle (Lemma 1)

Client		Server
• Index $i \in \{1, \ldots, n\}$		• Random $\mathbf{x} \in \{0,1\}^n$
• Generate a query	$\xrightarrow{q}$	
	$\xleftarrow{a}$	• Server answers

- **Claim 1:** $I(x_i; a) = 1$ assuming perfect correctness

  Proof: Correctness. □
- **Claim 2:** $\sum_{j=1}^{n} I(x_j; a) \leq H(a) \leq |a|$.

  Proof: As $x_1, \ldots, x_n$ are independent. □

*The randomness is from $\mathbf{x}$ and from the proceduce generating the answer.

Break PIR with SZK oracle (Lemma 1)

Client

- Index $i \in \{1, \ldots, n\}$
- Generate a query $\xrightarrow{q}$

Server

- Random $\mathbf{x} \in \{0,1\}^n$
- $\xleftarrow{a}$ Server answers

- **Claim 1:** $I(x_i; a) = 1$ assuming perfect correctness

  Proof: Correctness. □
- **Claim 2:** $\sum_{j=1}^{n} I(x_j; a) \leq H(a) \leq |a|$.

  Proof: As $x_1, \ldots, x_n$ are independent. □
- **Corollary:** $\sum_{j=1}^{n} I(x_j; a)$ is small.

*The randomness is from $\mathbf{x}$ and from the proceduce generating the answer.

Idea: Reduce Breaking PIR to Estimating Entropy

Given a query q, guess the index

- **Claim 1:** $I(x_i; a) = 1$ assuming perfect correctness
- **Claim 2:** $\sum_{j=1}^{n} I(x_j; a) \leq H(a) \leq |a|$.

Idea: Reduce Breaking PIR to Estimating Entropy

Given a query q, guess the index

- Emulate how the server answer q when $\mathbf{x} \in \{0,1\}^n$ is random
  Estimate $I(x_j; a)$ for each $j \in \{1, \dots, n\}$ using SZK oracle (on next slide)

- **Claim 1:** $I(x_i; a) = 1$ assuming perfect correctness
- **Claim 2:** $\sum_{j=1}^{n} I(x_j; a) \leq H(a) \leq |a|$.

Idea: Reduce Breaking PIR to Estimating Entropy

Given a query q, guess the index

- Emulate how the server answer q when $\mathbf{x} \in \{0,1\}^n$ is random
  Estimate $I(x_j; a)$ for each $j \in \{1, \ldots, n\}$ using SZK oracle (on next slide)
- Output a random i' w.p. $\dfrac{I(x_{i'}; a)}{\sum_{j=1}^{n} I(x_j; a)}$

- **Claim 1:** $I(x_i; a) = 1$ assuming perfect correctness
- **Claim 2:** $\sum_{j=1}^{n} I(x_j; a) \leq H(a) \leq |a|$.

Idea: Reduce Breaking PIR to Estimating Entropy

Given a query q, guess the index

- Emulate how the server answer q when $\mathbf{x} \in \{0,1\}^n$ is random
  Estimate $I(x_j; a)$ for each $j \in \{1, \ldots, n\}$ using SZK oracle (on next slide)
- Output a random i' w.p. $\dfrac{I(x_{i'}; a)}{\sum_{j=1}^{n} I(x_j; a)}$
- Guess correctly w.p. $\geq \dfrac{1}{|a|}$

- **Claim 1:** $I(x_i; a) = 1$ assuming perfect correctness
- **Claim 2:** $\sum_{j=1}^{n} I(x_j; a) \leq H(a) \leq |a|$.

Idea: Reduce Breaking PIR to Estimating Entropy

Given a query q, guess the index

- Emulate how the server answer q when $\mathbf{x} \in \{0,1\}^n$ is random
  Estimate $I(x_j; a)$ for each $j \in \{1, \ldots, n\}$ using SZK oracle (on next slide)
- Output a random i' w.p. $\dfrac{I(x_{i'}; a)}{\sum_{j=1}^{n} I(x_j; a)}$
- Guess correctly w.p. $\geq \dfrac{1 - h(\varepsilon)}{|a|}$

- **Claim 1:** $\mathbb{E}_q[I(x_i; a)] \geq 1 - h(\varepsilon)$ assuming correctness w.h.p.
- **Claim 2:** $\sum_{j=1}^{n} I(x_j; a) \leq H(a) \leq |a|$.

Mutual Information and SZK

- Mutual information
  $I(x_i; a) = H(x_i) + H(a) - H(x_i, a) = H(x_i) - H(x_i|a)$
- Entropy Approximation is in **SZK**:
  - Given a circuit generating a distribution D, and h
  - To distinguish between $H(D) \geq h + \frac{1}{\text{poly}}$ and $H(D) \leq h - \frac{1}{\text{poly}}$
- Can estimate entropy given an SZK oracle

Mutual Information and SZK

- Mutual information
  $I(x_i; a) = H(x_i) + H(a) - H(x_i, a) = H(x_i) - H(x_i|a)$
- Entropy Approximation is in **SZK**:
  - Given a circuit generating a distribution D, and h
  - To distinguish between $H(D) \geq h + \frac{1}{\text{poly}}$ and $H(D) \leq h - \frac{1}{\text{poly}}$
- Can estimate entropy given an SZK oracle

Mutual Information and SZK

- Mutual information
  $I(x_i; a) = H(x_i) + H(a) - H(x_i, a) = H(x_i) - H(x_i|a)$
- Entropy Approximation is in **SZK**:
  - Given a circuit generating a distribution D, and h
  - To distinguish between $H(D) \geq h + \frac{1}{\text{poly}}$ and $H(D) \leq h - \frac{1}{\text{poly}}$
- Can estimate entropy given an SZK oracle

Mutual Information and SZK

- Mutual information
  $I(x_i; a) = H(x_i) + H(a) - H(x_i, a) = H(x_i) - H(x_i|a)$
- Entropy Approximation is in **SZK**:
  - Given a circuit generating a distribution D, and h
  - To distinguish between $H(D) \geq h + \frac{1}{\text{poly}}$ and $H(D) \leq h - \frac{1}{\text{poly}}$
- Can estimate entropy given an SZK oracle

Client: i, index

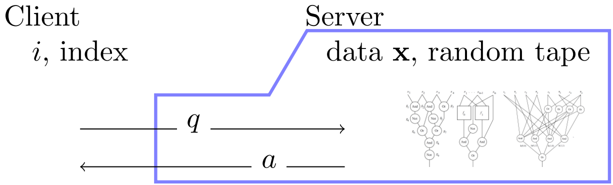
Server: data $\mathbf{x}$, random tape

Client $\xrightarrow{\ q\ }$ Server

Client $\xleftarrow{\ a\ }$ Server

Mutual Information and SZK

- Mutual information
  $I(x_i; a) = H(x_i) + H(a) - H(x_i, a) = H(x_i) - H(x_i|a)$
- Entropy Approximation is in **SZK**:
  - Given a circuit generating a distribution D, and h
  - To distinguish between $H(D) \geq h + \frac{1}{\text{poly}}$ and $H(D) \leq h - \frac{1}{\text{poly}}$
- Can estimate entropy given an SZK oracle



Mutual Information and SZK

- Mutual information
  $I(x_i; a) = H(x_i) + H(a) - H(x_i, a) = H(x_i) - H(x_i|a)$
- Entropy Approximation is in **SZK**:
  - Given a circuit generating a distribution D, and h
  - To distinguish between $H(D) \geq h + \frac{1}{\text{poly}}$ and $H(D) \leq h - \frac{1}{\text{poly}}$
- Can <u>estimate</u> entropy given an SZK oracle

Client

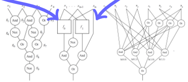
i, index

Server

data $\mathbf{x}$, random tape

q, fixed

a



Recall Proof Overview

Lemma 1 (Single-server one-round) PIR can be broken with an SZK oracle

Lemma 2 Language $L \in \mathbf{BPP}^{\mathbf{SZK}} \implies L \in \mathbf{AM} \cap \mathbf{coAM}$
(Mahmoody & Xiao, 2010)

Thus: if there is a reduction from SAT to breaking PIR, then $\mathsf{SAT} \in \mathbf{AM} \cap \mathbf{coAM}$.

Lemma 3 $\mathbf{NP} \not\subseteq \mathbf{coAM}$ unless polynomial hierarchy collapses
(Boppana, Håstad & Zachos, 1987)

Thus: if there is a reduction from SAT to breaking PIR, then polynomial hierarchy collapses.

Open problem: Multiple-round

Multiple-round PIR

- Could we rule out multiple-round PIR?

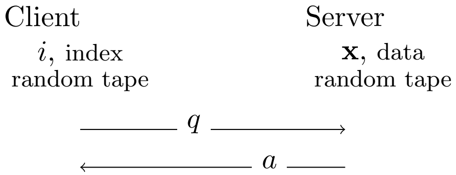
One-round PIR

Open problem: Multiple-round

Multiple-round PIR

- Could we rule out multiple-round PIR?

One-round PIR



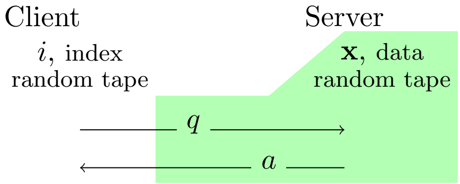
Open problem: Multiple-round

Multiple-round PIR

- Could we rule out multiple-round PIR?

One-round PIR

- Given the view of server, it's easy to generate another view.



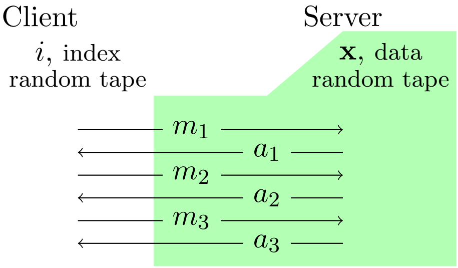
Open problem: Multiple-round

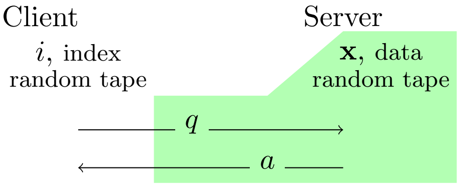
Multiple-round PIR

- Could we rule out multiple-round PIR?

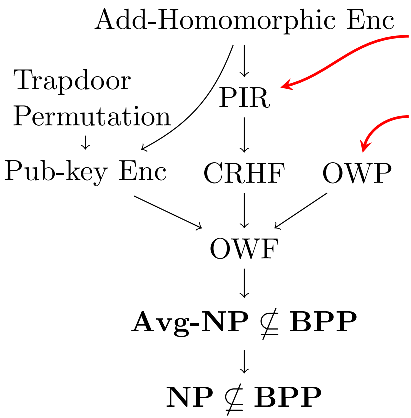
One-round PIR

- Given the view of server, it's easy to generate another view.



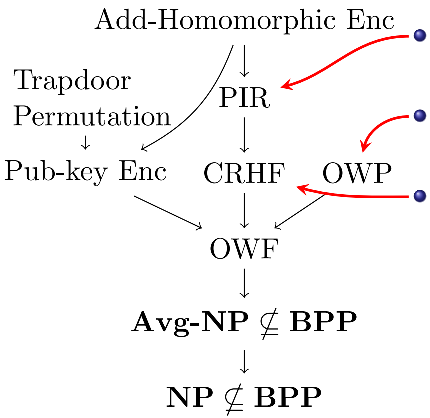


Open problem: CRHF



- PIR
  (This work)
- One-way Permutations
  (Brassard 1979)

Open problem: CRHF



- PIR
  (This work)
- One-way Permutations
  (Brassard 1979)
- Could we rule out reduction from SAT to finding collisions?
  (TCC 2017?)

Thank you!