On basing one-way permutations on NP-hard problems under quantum reductions

Nai-Hui Chia  
(PennState to UT Austin)

Joint work with Sean Hallgren (PennState) and Fang Song (Portland State to TAMU)
How do people say a crypto system is computationally secure?

Many experts put lots of efforts on breaking system Y for a very long time. After 50 yrs... Still cannot find an efficient algorithm for Y.

Okay, Y is secure.
How do people say a crypto system is computationally secure?

Many experts put lots of efforts on breaking system Y for a very long time. After 50yrs...

Still cannot find an efficient algorithm for Y

Okay, Y is secure

Do we really need to wait 50yrs?
How do people say a crypto system is computationally secure?

Many experts put lots of efforts on breaking system Y for a very long time.

Still cannot find an efficient algorithm for Y

Okay, Y is secure

Do we really need to wait 50yrs?

- SAT has already been studied for >50yrs.
- SAT is hard (NP-complete)
- $P \neq NP$ (people believe)

Use SAT to show Problem Y is hard.
Show Y is hard by a reduction from SAT: SAT $\leq$ Y

SAT $\leq$ Y:
- An efficient algorithm $A$ solving SAT by using an oracle for Y.
- Algorithm $A$ and (Questions, Answers) can be either classical or quantum!

SAT $\leq$ Y $\Rightarrow$ No efficient algorithm can break system Y unless NP = P.
Consider Y as inverting one-way functions

- Functions which are easy to compute but hard to invert.
- A fundamental cryptographic primitive. The existence of one-way functions implies
  - Pseudorandom generators
  - Digital signature scheme
  - Message Authentication Codes
  - .......
Consider Y as inverting one-way functions

- Functions which are easy to compute but hard to invert.
- A fundamental cryptographic primitive. The existence of one-way functions implies
  - Pseudorandom generators
  - Digital signature scheme
  - Message Authentication Codes
  - ......

Can inverting one-way functions be as hard as SAT?
One-way functions

- Functions which are easy to compute but hard to invert.
- A fundamental cryptographic primitive. It implies
  - Pseudorandom generators
  - Digital signature scheme
  - Message Authentication Codes
  - ........

Can inverting one-way functions be as hard as SAT?

- SAT $\leq^c$ Inverting a one-way permutation $\Rightarrow$ PH collapses [Brassard96].
- SAT $\leq^c$ Inverting a one-way function $\Rightarrow$ PH collapses,
  - when the reductions are non-adaptive [AGGM05] or the functions are preimage verifiable [AGGM05, BB15].
One-way functions

- Functions which are easy to compute but hard to invert.
- A fundamental cryptographic primitive. It implies
  - Pseudorandom generators
  - Digital signature scheme
  - Message Authentication Codes
  - .......

Can inverting one-way functions be as hard as SAT?

- $\text{SAT} \leq^c_{\text{c}}\text{Inverting a one-way permutation} \Rightarrow \text{PH collapses}$ [Brassard96].
- $\text{SAT} \leq^c_{\text{c}}\text{Inverting a one-way function} \Rightarrow \text{PH collapses},$
  - when the reductions are non-adaptive [AGGM05] or the functions are preimage verifiable [AGGM05, BB15].

Only classical reductions are considered!
We are interested in quantum reductions

Hard problems (e.g., NP-hard problems)

Computational tasks (e.g., inverting one-way functions)

Do these reductions exist?
Our results

SAT $\leq_q$ Inverting a one-way permutation (Inv-OWP) $\Rightarrow$ coNP $\subseteq$ AM $\Rightarrow$ PH collapses [Brassard96].

SAT $\leq_c$ Inverting a one-way function $\Rightarrow$ PH collapses,
  ○ when the reductions are non-adaptive[BT06] or the functions are preimage verifiable[].

SAT $\leq_q$ Inverting a one-way permutation (Inv-OWP) $\Rightarrow$
coNP $\subseteq$ QIP(2), where
  ● our result has the restrictions that the reductions are non-adaptive and the distribution of the questions to the oracle are not far from the uniform distribution.
  ● It is not known if coNP $\subseteq$ QIP(2).
NP-hard Problems $\leq^c_{c}$ Inv-OWP $\Rightarrow$ coNP $\subseteq$ AM

**Theorem [Brassad96]:** SAT $\leq^c_{c}$ Inv-OWP $\Rightarrow$ coNP $\subseteq$ AM $\Rightarrow$ The polynomial hierarchy collapses to the second level.

The goal is to construct a “constant-round protocol” for SAT by using the reduction.
We say $L \in AM$ if

- (completeness) if $x \in L$, there is a prover (Merlin) can convince Arthur (the verifier) that $x \in L$.
- (soundness) if $x \notin L$, no prover (Merlin) can convince Arthur that $x \in L$. 

Arthur-Merlin Protocol

Verifier (Arthur)

Prover (Merlin)

Two classical messages exchanged.

$x$

$r$: a random string

c: a proof

$A(x,r,c)=L(x)$
SAT ≤ₜ Inv-OWP ⇒ SAT ∈ AM

1. The verifier sends his random string to the prover.
   ○ The prover knows y after having the random string.
2. The prover sends y and x (where f(x)=y) to the verifier.
   ○ A malicious prover may send (y’, x’) ≠ (y, x).
3. The verifier verifies whether y is the question and f(x) = y. If not, reject.
4. The verifier runs the reduction R⁰ if he doesn’t reject in step 3.
Can we use this protocol for quantum reductions?

1. The verifier sends his random string to the prover.
   - The prover knows y after having the random string.
2. The prover sends y and x (where f(x)=y) to the verifier.
   - A malicious prover may send (y', x') ≠ (y, x).
3. The verifier verifies whether y is the question and f(x) = y. If not, reject.
4. The verifier runs the reduction $R^o$ if he doesn’t reject in step 3.
No, quantum reductions are more tricky

Each question can be in superposition
- $|Q\rangle_{123} = \sum_q c_q |q\rangle_1 |0\rangle_2 |w_q\rangle_3$
- $|c_q|^2$ can be viewed as the weight of question $q$.

The answer is also in superposition
- $|A\rangle_{123} = \sum_q c_q |q\rangle_1 |f^{-1}(q)\rangle_2 |w_q\rangle_3$
Why does the classical protocol fail?

Each question can be in superposition

- $|Q\rangle_{123} = \sum_q c_q |q\rangle_1 |0\rangle_2 |w_q\rangle_3$
- $|c_q|^2$ can be viewed as the weight of question $q$.

The answer is also in superposition

- $|A\rangle_{123} = \sum_q c_q |q\rangle_1 |f^{-1}(q)\rangle_2 |w_q\rangle_3$

- Simulating the reduction SAT $\leq_q$ Inv-OWP only gives “quantum interactive proof” protocol.
- The prover can cheat by giving correct $(q, f^{-1}(q))$, but changing the weight $c_q$. 

Reduction $U_R$
(An efficient quantum algorithm)

$O$
(An oracle for Inv-OWP)
Goal: $\text{SAT} \leq_q \text{Inv-OWP} \Rightarrow \text{SAT} \in \text{QIP}(2)$

We say $L \in \text{QIP}(2)$ if

- **(completeness)** if $x \in L$, the prover can convince the verifier that $x \in L$.
- **(soundness)** if $x \notin L$, no prover can convince the verifier that $x \in L$. 

![Diagram of Prover and Verifier](image_url)
Goal: $\text{SAT} \leq_q \text{Inv-OWP} \Rightarrow \text{SAT} \in \text{QIP}(2)$ under uniform quantum reductions

We say $L \in \text{QIP}(2)$ if

- (completeness) if $x \in L$, the prover can convince the verifier that $x \in L$.
- (soundness) if $x \notin L$, no prover can convince the verifier that $x \in L$.

Uniform quantum reductions:

- Each query is a uniform superposition
  - $|Q\rangle = \sum_q |q\rangle |0\rangle |w_q\rangle$
- The answer is also in uniform superposition
  - $|A\rangle = \sum_q |f^{-1}(q)\rangle |w_q\rangle$
A protocol with “trap”

The main idea: If the prover cheats, he has $\frac{1}{2}$ probability to cheat on the trap state. The verifier can catch him by verifying the trap state!

- The prover cannot distinguish the trap and the real query.
- $|S>$ can be efficiently verified by the verifier.
A protocol with “trap”

1. Send the register $M$ of $|Q>$ or $|T>$ uniformly at random.
   - $|Q> = \sum_q (|q/>0>)_M (|w_q>q>)_V$
   - $|T> = \sum_q (|0>q>)_M (|0>q>)_V$

2. An honest prover will send $|A>$ or $|S>$.
   - $|A> = \sum_q |q> |f^{-1}(q)> |w_q>q>$
   - $|S> = \sum_q |q> |f^{-1}(q)> |0> |q>$

3. The verifier does the following.
   - In case $|Q>$:
     - Run the reduction and accept if the reduction accepts.
   - In case $|T>$:
     - Run the unitary $U$: $|S> \Rightarrow |0>$. Measure the output in the standard basis. If the outcome is $|0>$, accepts.

- $|A> \Rightarrow |0>$ may not be efficient.
- $U: |S> \Rightarrow |0>$ is efficient.
Analysis of the trap protocol

1. Send the register $\mathcal{M}$ of $|Q\rangle$ or $|T\rangle$ uniformly at random.
   - $|Q\rangle = \sum_q (|q\rangle|0\rangle)_\mathcal{M} (|w_q\rangle|q\rangle)_\mathcal{V}$
   - $|T\rangle = \sum_q (|q\rangle|0\rangle)_\mathcal{M} (|0\rangle|q\rangle)_\mathcal{V}$

2. The prover does not know which state he gets.

3. The verifier does the following.
   - In case $|Q\rangle$:
     - Run the reduction and accept if the reduction accepts.
   - In case $|T\rangle$:
     - Run the unitary $U$: $|S\rangle \Rightarrow |0...0\rangle$ and measure the output in the standard basis. If the outcome is $|0\rangle$, accepts.

- The prover does not know which state he gets.
- No matter which operator the prover applies, it will
  - Change $|S\rangle$ a lot
    - Suppose $|S'\rangle$ is far from $|S\rangle$. By applying $U$: $|S\rangle \Rightarrow |0...0\rangle$, $|S'\rangle$ is far from $|0...0\rangle$.
  - Or changes $|A\rangle$ little.
    - Suppose $|A'\rangle = |A\rangle$. By applying the reduction, $|A'\rangle$ will be rejected with high probability.

In these two cases, the verifier rejects with high probability.
**Theorem:** $\text{SAT} \leq_{uq} \text{Inv-OWP} \Rightarrow \text{coNP} \subseteq \text{QIP}(2)$.

The result $\text{coNP} \subseteq \text{QIP}(2)$ is not as strong as PH collapses. However, it is a nontrivial consequence of the existence of quantum reductions.

The “trap” protocol can be easily extended to quantum reductions with multiple non-adaptive queries.

We can deal with other non-uniform distributions which are not far from the uniform distribution by quantum resampling.
Open questions

- Can we deal with other distributions or adaptive queries?
- We shall revisit other no-go theorems for crypto primitives.
  - For cryptographic primitives which security are not based on NP-complete problems under classical reductions, can NP-complete problems reduce to them if quantum reductions are allowed?
    - E.g., Private information retrieval (PIR), FHE, Inv-OWF, ...
- Can we give more evidences that coNP is not in QIP(2)?
- Can we find other consequence which is stronger than coNP ⊆ QIP(2)?
  - E.g., coNP ⊆ QAM or QMA.
- Can we find an example where we can prove quantum reductions are more powerful than classical reductions?
- Generally, people think quantum algorithms make crypto systems less computationally secure. But, maybe it can make crypto systems secure by reducing hard problems to these systems.