Zero-Knowledge Against Quantum Attacks
John Watrous (2009)

Sebastian Verschoor

QIC710: Introduction to Quantum Information Processing
University of Waterloo

December 8th, 2016
Interactive proof systems

Zero-knowledge
  Definition
  Applications of zero-knowledge
  Example: graph isomorphism

Quantum Attacks
  Quantum Zero-Knowledge
  Quantum rewinding lemma
  Example: graph isomorphism

Generalizing the results
Interactive proof systems

- A statement \((x)\) is True iff \(x \in L\) for some fixed language \(L\)
  - Example: \(L\) is the language of all pairs of graphs that are isomorphic, \(x\) is the pair \((G_0, G_1)\)

- Proving as an interactive procedure
  - Prover \((P)\) convinces Verifier \((V)\) of the truth of some statement \((x)\) by giving a proof/certificate/witness \((w)\)

- (Optional) restrictions:
  - Verifier is modelled by a Turing machine
  - Verifier runs in polynomial time
    - Size of the proof \(|w|\) is polynomial
  - Verifier might only accept with probability \(\geq 2/3\)
In general, a proof system is concerned about two things:
- Completeness: if both parties play honest, will V accept?
- Soundness: if P cheats, will V reject?

Zero-knowledge handles the case in which the V cheats:
- Zero-knowledge: the protocol asserts nothing but $x \in L$
- Leakage includes:
  - V cannot convince a third party of $x \in L$
  - V cannot convince a third party of “P knows that $x \in L$”
  - V cannot convince a third party that any conversation between P and V took place at all

How to prove “Zero-knowledgeness”?
Zero-knowledge

- In general, a proof system is concerned about two things:
  - Completeness: if both parties play honest, will $V$ accept?
  - Soundness: if $P$ cheats, will $V$ reject?
- Zero-knowledge handles the case in which the $V$ cheats:
  - Zero-knowledge: the protocol asserts nothing but $x \in L$
  - Leakage includes:
    - $V$ cannot convince a third party of $x \in L$
    - $V$ cannot convince a third party of “$P$ knows that $x \in L$”
    - $V$ cannot convince a third party that any conversation between $P$ and $V$ took place at all

- How to prove “Zero-knowledgeness”?
In general, a proof system is concerned about two things:
- Completeness: if both parties play honest, will V accept?
- Soundness: if P cheats, will V reject?

Zero-knowledge handles the case in which the V cheats:
- Zero-knowledge: the protocol asserts nothing but \( x \in L \)
- Leakage includes:
  - V cannot convince a third party of \( x \in L \)
  - V cannot convince a third party of “P knows that \( x \in L \)”
  - V cannot convince a third party that any conversation between P and V took place at all

How to prove “Zero-knowledgeness”?
Zero-knowledge protocols

- An interactive protocol between $P$ and $V$ is zero-knowledge on $L$ if for all possible (cheating) verifiers ($V'$):

  $$\text{View}_{P,V'} \text{ is approximable on } L' = \{(x, H) | x \in L \land |H| = |x|^c\}$$

- View is all $V'$ sees
  - Random bits
  - Messages from $P$
  - Additional helper data $H$

- A View is approximable if there exists an efficient Turing machine $S$ that creates a distribution that is indistinguishable from the View.
  - $S$ is called the simulator
An interactive protocol between $P$ and $V$ is zero-knowledge on $L$ if for all possible (cheating) verifiers ($V'$):

$$\text{View}_{P, V'} \text{ is approximable on } L' = \{(x, H) | x \in L \land |H| = |x|^c\}$$

- View is all $V'$ sees
  - Random bits
  - Messages from $P$
  - Additional helper data $H$

- A View is approximable if there exists an efficient Turing machine $S$ that creates a distribution that is indistinguishable from the View.
  - $S$ is called the simulator.
An interactive protocol between $P$ and $V$ is zero-knowledge on $L$ if for all possible (cheating) verifiers ($V'$):

$$\text{View}_{P, V'} \text{ is approximable on } L' = \{(x, H) | x \in L \land |H| = |x|^c\}$$

- **View** is all $V'$ sees
  - Random bits
  - Messages from $P$
  - Additional helper data $H$

- A View is **approximable** if there exists an efficient Turing machine $S$ that creates a distribution that is indistinguishable from the View.
  - $S$ is called the **simulator**
Defining indistinguishability

- Families of random variables $U : \{U(x)\}$ and $V : \{V(x)\}$
  - If there is no judge $J$ that can tell if the sample came from $U(x)$ or $V(x)$, $U$ and $V$ are indistinguishable
- Types of indistinguishability:
  - **Perfect**: $J$ gets arbitrary many samples
    - $U = V$
  - **Statistical**: $J$ gets only polynomial many samples
    - Statistical difference between $U$ and $V$ is negligible
  - **Computational**: $J$ gets only polynomial many samples and has only polynomial time to distinguish them
    - $U$ and $V$ cannot be distinguished by an efficient algorithm
Applications of zero-knowledge

- Online authentication scheme
  - Client proves (in zero-knowledge) to a web-server that they know the password
  - (Note: this is not how the internet actually works: usually you just send a plaintext password)

- CASH
Example: graph isomorphism

Common input \( x = (G_0, G_1) \):

\[ G_0 \]

\[ G_1 \]

To prove: \( G_0 \simeq G_1 \)
Example: graph isomorphism

Prover knows permutation $\sigma$ such that $\sigma(G_1) = G_0$:

\[ G_0 = \sigma(G_1) \]

$\sigma$ can only exist if $G_0 \cong G_1$
Example: graph isomorphism

Prover chooses random permutation $\pi$ and computes $H = \pi(G_0)$:

$G_0 = \sigma(G_1)$

$H = \pi(G_0) = \pi\sigma(G_1)$

Prover sends $H$ to Verifier
Example: graph isomorphism

Repeat until $V$ is satisfied

To be a Zero-knowledge proof of $G_0 \cong G_1$, we need to prove:

- Completeness: $G_0 \cong G_1 \Rightarrow \Pr[\text{accept}] = 1$
- Soundness: $G_0 \not\cong G_1 \Rightarrow \Pr[\text{reject}] = 1/2$
- Zero-knowledge: does a simulator exist?
Example: graph isomorphism

Repeat until $V$ is satisfied

- To be a Zero-knowledge proof of $G_0 \simeq G_1$, we need to prove:
  - Completeness: $G_0 \simeq G_1 \Rightarrow \Pr[\text{accept}] = 1$
  - Soundness: $G_0 \not\simeq G_1 \Rightarrow \Pr[\text{reject}] = 1/2$
  - Zero-knowledge: does a simulator exist?
Example: graph isomorphism

Repeat until $V$ is satisfied

To be a Zero-knowledge proof of $G_0 \simeq G_1$, we need to prove:

- Completeness: $G_0 \simeq G_1 \Rightarrow \Pr[\text{accept}] = 1$
- Soundness: $G_0 \not\simeq G_1 \Rightarrow \Pr[\text{reject}] = 1/2$
- Zero-knowledge: does a simulator exist?
Example: graph isomorphism

Repeat until $V$ is satisfied

To be a Zero-knowledge proof of $G_0 \cong G_1$, we need to prove:

- Completeness: $G_0 \cong G_1 \Rightarrow \Pr[\text{accept}] = 1$
- Soundness: $G_0 \not\cong G_1 \Rightarrow \Pr[\text{reject}] = 1/2$
- Zero-knowledge: does a simulator exist?
Example: graph isomorphism

Repeat until \( V \) is satisfied

To be a Zero-knowledge proof of \( G_0 \simeq G_1 \), we need to prove:

1. **Completeness:** \( G_0 \simeq G_1 \Rightarrow \Pr[\text{accept}] = 1 \)
2. **Soundness:** \( G_0 \not\simeq G_1 \Rightarrow \Pr[\text{reject}] = \frac{1}{2} \)
3. **Zero-knowledge:** does a simulator exist?
Example: graph isomorphism

\begin{itemize}
  \item Repeat until \( V \) is satisfied
  \item To be a Zero-knowledge proof of \( G_0 \simeq G_1 \), we need to prove:
    \begin{itemize}
      \item Completeness: \( G_0 \simeq G_1 \Rightarrow \Pr[\text{accept}] = 1 \)
      \item Soundness: \( G_0 \not\simeq G_1 \Rightarrow \Pr[\text{reject}] = 1/2 \)
    \end{itemize}
    \item Zero-knowledge: does a simulator exist?
\end{itemize}
Example: graph isomorphism

For a graph \( G_0 \sim G_1 \), we need to prove:

- **Completeness:** \( G_0 \sim G_1 \Rightarrow \Pr[\text{accept}] = 1 \)
- **Soundness:** \( G_0 \not\sim G_1 \Rightarrow \Pr[\text{reject}] = 1/2 \)
- **Zero-knowledge:** does a simulator exist?
Q: Does a simulator exist?
A: Sure, just take out your time machine!
Define a simulator $S^{V'}$ that uses $V'$ as a subroutine

1. Pick a random permutation $\tau$ and bit $a'$
2. Send $H = \tau(G_{a'})$ to $V'$
3. $V'$ replies with $a$
4. if $a' = a$: send $\tau$
   else: go back in time (rewind $V'$) and try again!

Efficient: expected to rewind once

$\text{View}_{P,V'} = \text{View}_{S^{V'},V'}$
Example: graph isomorphism

Q: Does a simulator exist?
A: Sure, just take out your time machine!

Define a simulator $S^{V'}$ that uses $V'$ as a subroutine

1. Pick a random permutation $\tau$ and bit $a'$
2. Send $H = \tau(G_{a'})$ to $V'$
3. $V'$ replies with $a$
4. If $a' = a$: send $\tau$
   - else: go back in time (rewind $V'$) and try again!

Efficient: expected to rewind once

$\text{View}_{P,V'} = \text{View}_{S^{V'},V'}$
Example: graph isomorphism

- Q: Does a simulator exist?
- A: Sure, just take out your time machine!
- Define a simulator $S^{V'}$ that uses $V'$ as a subroutine

1. Pick a random permutation $\tau$ and bit $a'$
2. Send $H = \tau(G_{a'})$ to $V'$
3. $V'$ replies with $a$
4. if $a' = a$: send $\tau$
   else: go back in time (rewind $V'$) and try again!

- Efficient: expected to rewind once
- $\text{View}_{P,V'} = \text{View}_{S^{V'},V'}$
Example: graph isomorphism

Q: Does a simulator exist?
A: Sure, just take out your time machine!

Define a simulator $S^{V'}$ that uses $V'$ as a subroutine

1. Pick a random permutation $\tau$ and bit $a'$
2. Send $H = \tau(G_{a'})$ to $V'$
3. $V'$ replies with $a$
4. if $a' = a$: send $\tau$
   else: go back in time (rewind $V'$) and try again!

Efficient: expected to rewind once

$View_{P,V'} = View_{S^{V'},V'}$
Example: graph isomorphism

▶ It’s all software: we don’t need a physical time machine
▶ $S^{V'}$ sets up $V'$ in a virtual machine
▶ Before every iteration of the protocol: take a snapshot
▶ if $a' = a$: success
   else: restart from the snapshot and try again

▶ What have we achieved?
   ▶ Any transcript that $V'$ gives to $J$ could have been created by $S^{V'}$, who does not have any knowledge
   ▶ So no transcript can leak any knowledge
What if cheating verifier $V'$ has a quantum computer?

- $V'$ could have auxiliary input entangled with qubits not accessible to $V'$ or $S^{V'}$, but available to the judge
- Rewinding cannot be applied *generally*
  - Quantum information cannot be copied
  - Running $V'$ might involve a irreversible measurement
  - Determining if a simulation was succesful requires a measurement
Quantum Zero-Knowledge

- Need to refine our notion of indistinguishability
- Instead of the View, we take a look at possible quantum channels that the cheating verifier can implement
- We use the Kitaev diamond norm distance between two channels $\Phi_0$ and $\Phi_1$:
  - $\frac{1}{2} \| \Phi_0 - \Phi_1 \|^\diamond$
  - Describes the maximum bias with which a physical process can distinguish them
  - Covers distinguishing with entangled states
  - This is analogous to the trace distance between quantum states
Quantum rewinding lemma

We can’t rewind in general. But we can if:

▶ Given a circuit $Q$ of the form:

$$|\psi\rangle|0^k\rangle = \sqrt{p(\psi)}|0\rangle|\phi_0(\psi)\rangle + \sqrt{1-p(\psi)}|1\rangle|\phi_1(\psi)\rangle$$

▶ We can *rewind* if $p = p(\psi)$ is constant (independent of $\psi$).

▶ Goal: to get $|\phi_0(\psi)\rangle$ with probability arbitrary close to 1
Quantum rewinding lemma

Getting $|\phi_0(\psi)\rangle$ from $|\psi\rangle$:
1. Apply $Q$
2. Repeat:
3. Measure first output register
4. If outcome is 1:
5. Apply $Q^{-1}$
6. Apply $U = 2 |0^m\rangle\langle 0^m| - I$ to ancilla
7. Apply $Q$
8. Output $|\phi\rangle$

After applying $Q$, we get in state
\[
Q |\psi\rangle |0^k\rangle = \sqrt{p} |0\rangle |\phi_0(\psi)\rangle + \sqrt{1-p} |1\rangle |\phi_1(\psi)\rangle
\]

If we measure 0, we are done! Else we apply
\[
Q(I \otimes U)Q^{-1} |1\rangle |\phi(\psi)\rangle = 2\sqrt{p(1-p)} |0\rangle |\phi_0(\psi)\rangle + (1 - 2p) |1\rangle |\phi_1(\psi)\rangle
\]
Quantum rewinding lemma

Getting $|\phi_0(\psi)\rangle$ from $|\psi\rangle$:

1. Apply $Q$
2. Repeat:
3. Measure first output register
4. If outcome is 1:
5. Apply $Q^{-1}$
6. Apply $U = 2|0^m\rangle\langle 0^m| - 1$ to ancilla
7. Apply $Q$
8. Output $|\phi\rangle$

After applying $Q$, we get in state

$$Q|\psi\rangle|0^k\rangle = \sqrt{p}|0\rangle|\phi_0(\psi)\rangle + \sqrt{1-p}|1\rangle|\phi_1(\psi)\rangle$$

If we measure 0, we are done! Else we apply

$$Q(I \otimes U)Q^{-1}|1\rangle|\phi(\psi)\rangle$$

$$= 2\sqrt{p(1-p)}|0\rangle|\phi_0(\psi)\rangle + (1-2p)|1\rangle|\phi_1(\psi)\rangle$$
Example: graph isomorphism

A cheating verifier has the following interaction:

\[ |\psi\rangle \]

\[ \rho \]

Where \( \rho = \frac{1}{n!} \sum_{\pi \in S_n} \sum_{a \in \{0,1\}} M_{\pi(G_0),a} |\psi\rangle \otimes \psi \langle M^\ast_{\pi(G_0),a} \]

The channel \( \Phi \) is then the tensor product of all registers in the view.
Example: graph isomorphism

We simulate this using the following $Q$:

\[
|\psi\rangle \rightarrow V_1' \rightarrow T
\]

Where $T$ creates a uniform superposition:

\[
\frac{1}{\sqrt{2n!}} \sum_{\tau \in S_n} \sum_{b \in \{0,1\}} |\tau(G_b)\rangle |b\rangle |\tau\rangle
\]
Example: graph isomorphism

- Works out to $p = 1/2$ with compatible states $\phi_0$ and $\phi_1$
- Applying the Quantum Rewinding lemma once to $|1\rangle |\phi_1(\psi)\rangle$
  
  $2\sqrt{p(1-p)} |0\rangle |\phi_0(\psi)\rangle + (1 - 2p) |1\rangle |\phi_1(\psi)\rangle$

  $= 2\sqrt{1/4} |0\rangle |\phi_0(\psi)\rangle + 0 |1\rangle |\phi_1(\psi)\rangle$

  $= |0\rangle |\phi_0(\psi)\rangle$

- For graph isomorphism, we need to rewind (at most) once
Generalizing the results

- Relax the assumption that $p$ is independent of $|\psi\rangle$
  - When $p(\psi)$ varies only by an exponentially small amount, we can still achieve statistical zero-knowledge

- The construction applies to all protocols of the form:
  1. $P$ sends a message to $V$
     - Message could even be some quantum state
  2. $V$ flips a fair coin and sends the result to $P$
  3. $P$ responds with a second message
     - Message could even be some quantum state

- All problems in HVQSZK can be expressed in this form
  - $\text{HVQSZK} = \text{QSZK}$
  - $\text{SZK} \subseteq \text{HVQSZK}$
    - Not known: are all classical proofs in SZK secure against quantum attacks?
Generalizing the results

- Relax the assumption that $p$ is independent of $|\psi\rangle$
  - When $p(\psi)$ varies only by an exponentially small amount, we can still achieve statistical zero-knowledge
- The construction applies to all protocols of the form:
  1. $P$ sends a message to $V$
     - Message could even be some quantum state
  2. $V$ flips a fair coin and sends the result to $P$
  3. $P$ responds with a second message
     - Message could even be some quantum state
- All problems in HVQSZK can be expressed in this form
  - $HVQSZK = QSZK$
  - $SZK \subseteq HVQSZK$
    - Not known: are all classical proofs in $SZK$ secure against quantum attacks?
Generalizing the results

- Relax the assumption that $p$ is independent of $|\psi\rangle$
  - When $p(\psi)$ varies only by an exponentially small amount, we can still achieve statistical zero-knowledge

- The construction applies to all protocols of the form:
  1. P sends a message to V
     - Message could even be some quantum state
  2. V flips a fair coin and sends the result to P
  3. P responds with a second message
     - Message could even be some quantum state

- All problems in HVQSZK can be expressed in this form
  - HVQSZK = QSZK
  - SZK $\subseteq$ HVQSZK
  - Not known: are all classical proofs in SZK secure against quantum attacks?
Generalizing the results

- Relax the assumption that $p$ is independent of $|\psi\rangle$
  - When $p(\psi)$ varies only by an exponentially small amount, we can still achieve statistical zero-knowledge
- The construction applies to all protocols of the form:
  1. $P$ sends a message to $V$
     - Message could even be some quantum state
  2. $V$ flips a fair coin and sends the result to $P$
  3. $P$ responds with a second message
     - Message could even be some quantum state
- All problems in HVQSZK can be expressed in this form
  - HVQSZK = QSZK
  - $SZK \subseteq HVQSZK$
    - Not known: are all classical proofs in $SZK$ secure against quantum attacks?
Generalizing the results

- Relax the assumption that $p$ is independent of $|\psi\rangle$
  - When $p(\psi)$ varies only by an exponentially small amount, we can still achieve statistical zero-knowledge

- The construction applies to all protocols of the form:
  1. $P$ sends a message to $V$
     - Message could even be some quantum state
  2. $V$ flips a fair coin and sends the result to $P$
  3. $P$ responds with a second message
     - Message could even be some quantum state

- All problems in HVQSZK can be expressed in this form
  - $HVQSZK = QSZK$
  - $SZK \subseteq HVQSZK$
    - Not known: are all classical proofs in $SZK$ secure against quantum attacks?
Generalizing the results

- Complexity class results
  - QSZK is closed under complement
  - The “close quantum states” problem is complete for QSZK
  - $\text{QSZK} \subseteq \text{QIP}(2)$
  - $\text{QSZK}_{a,b} = \text{QSZK}_{1,c}$ with $c$ polynomially small
  - Similar results for QZK and QPZK

- Non-interactive zero-knowledge proofs
  - Quantum non-interactive zero-knowledge proofs
Further reading

- Zero Knowledge Proofs: An illustrated primer – Matthew Green (2007) [link]
- Zero-Knowledge Against Quantum Attacks – John Watrous (2009) [link]
- Quantum Proofs – Thomas Vidick, John Watrous (2016) [link]
- Slides will be available at my website: [zeroknowledge.me]
Thank you
We can use the interactive games to define complexity classes

▶ **NP**: Non-deterministic Polynomial time
  - \( L \in \textbf{NP} \) if a short proof exists for an efficient verifier
    - Formally: there exist polynomials \( p, q \) and verifier \( V \) such that
      \[
      \forall x \in L : \exists w : |w| = q(|x|) \land V(x, w) = 1
      \]
      \[
      \forall x \not\in L : |w| = q(|x|) \Rightarrow V(x, w) = 0
      \]
      \[
      \forall x, w : V(x, w) \text{ runs in time } p(|x|)
      \]

▶ **P**: Polynomial time
  - \( L \in \textbf{P} \) if an efficient verifier exists
    - Formally: There exists a polynomial \( p \) and verifier \( V \) such that
      \[
      \forall x \in L : V(x, \emptyset) = 1
      \]
      \[
      \forall x \not\in L : V(x, \emptyset) = 0
      \]
      \[
      \forall x : V(x, \emptyset) \text{ runs in time } p(|x|)
      \]

▶ \( \textbf{P} \subseteq \textbf{NP} \)
**BPP**: Bounded-error Probabilistic Polynomial time

- $L \in \text{BPP}$ if an efficient *probabilistic* verifier exists
  \[
  \forall x \in L : \Pr[V(x, \emptyset) = 1] \geq 2/3
  \]
  \[
  \forall x \notin L : \Pr[V(x, \emptyset) = 0] \leq 1/3
  \]

**MA**: Merlin-Arthur

- Arthur is a **BPP** verifier

**AM**: Arthur-Merlin

- Arthur can send a message (challenge) to Merlin before Merlin provides a proof

**IP**: Interactive Proof systems

- Like **AM**, but allows many rounds interactions

$P \subseteq \text{BPP} \subseteq \text{MA} \subseteq \text{AM} \subseteq \text{IP}$