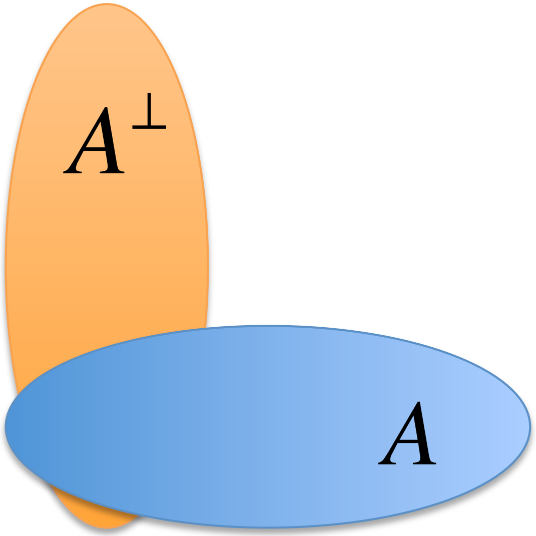
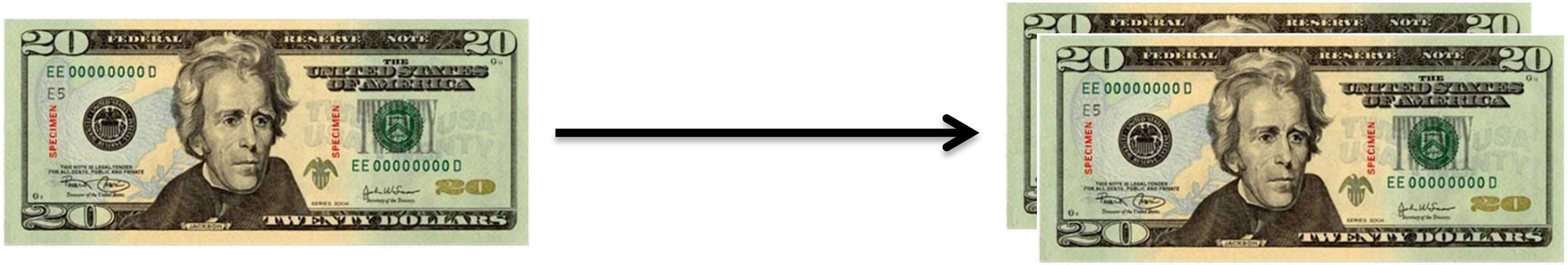
Quantum Money from Hidden Subspaces



Scott Aaronson and Paul Christiano

As long as there has been money, there have been people trying to copy it.



Problem: whatever a bank can do to print money, a forger can do to copy it.

$$x \longrightarrow (x, x)$$

Classically, we need a trusted third party to prevent double-spending…

The No-Cloning Theorem

$$|\psi\rangle \;\not\longrightarrow\; |\psi\rangle|\psi\rangle$$

There is *no* procedure which duplicates a general quantum state.

Can we use “uncloneable” quantum states as unforgeable currency?

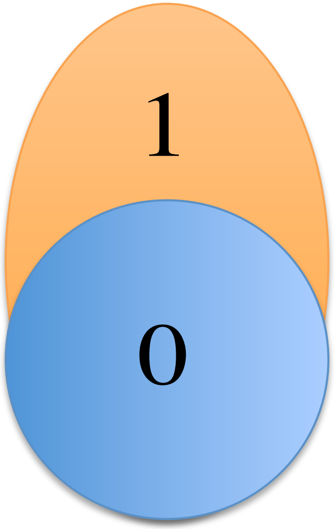
A simple solution inspired by Wiesner [1969]:

If I randomly give you one of the two pure states…

$$|0\rangle + |1\rangle$$

or

$$|0\rangle$$

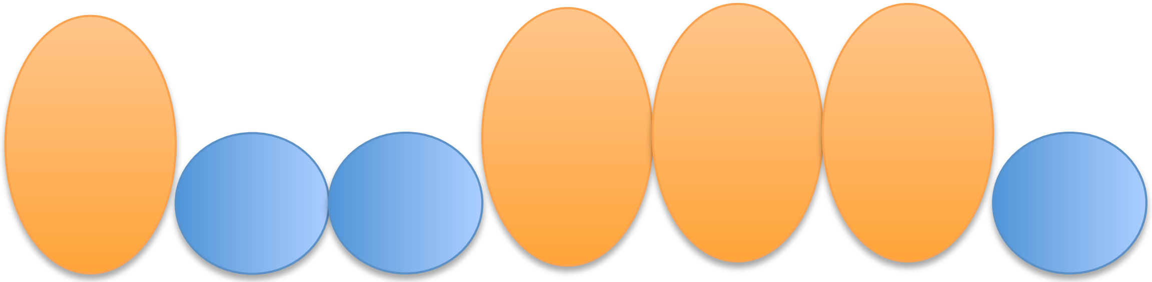


…you can't guess which I gave you with probability more than (3/4)…

…and you can't faithfully copy it.

Wiesner's Quantum Money

If I concatenate *k* of these states to produce

$|\$\rangle =$ 

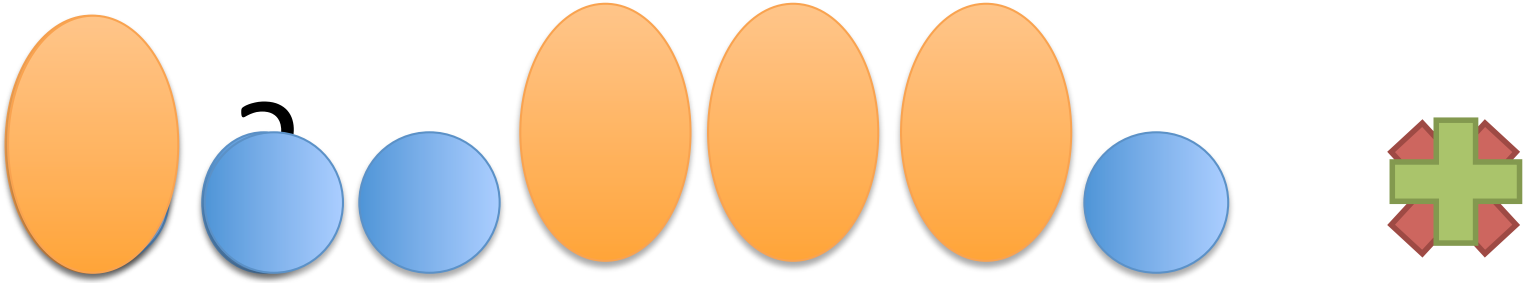
I can recognize $|\$\rangle$ by measuring each bit in an appropriate basis…

…but you can't copy $|\$\rangle$ except with exponentially small success probability.

Problems with Wiesner’s Scheme

Only the bank that minted it can recognize money.

In fact, the money becomes insecure as soon as we give the users a verification oracle.



…

Modern goal: secure quantum money that anyone can verify

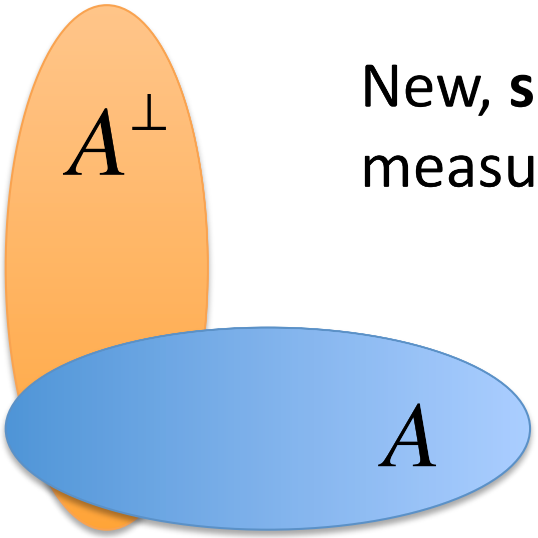
Prior Art

Aaronson, CCC' 2009: Showed there is no generic counterfeiting strategy using the verification procedure as a black box.

Aaronson, CCC' 2009: Proposed an explicit quantum money scheme, which was broken in **Lutomirski et al. 2010.**

Farhi et al., ITCS' 2012: Proposed a new money scheme based on knot diagrams. A significant advance, but its security is poorly understood. (Even when the knot diagrams are replaced by black-box idealizations.)

Our Results



New, **simple** scheme: verification consists of measuring in just two complementary bases.

Security based on a **purely classical** assumption about the hardness of an algebraic problem.

A "black-box" version of our scheme, in which the bank provides perfectly obfuscated subspace membership oracles, is **unconditionally secure**.

The same construction gives the first "**private-key**" money scheme which remains secure given interaction with the bank.

$$\text{KeyGen}\left(0^k\right)=\left(k_{public},k_{private}\right)$$

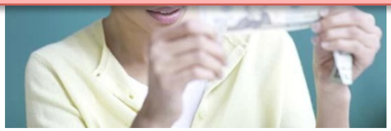
$k_{private}$

Completeness: Ver accepts valid notes w.h.p.

k_{public}

$|\$\rangle$

Soundness: If a counterfeiter starts with *n* notes and outputs *n+1*, Ver rejects one w.h.p.



$$\text{Ver}\left(k_{public},|\$\rangle\right)$$

$$C\left(k_{public},|\$_1\rangle,\cdots,|\$_n\rangle\right)=$$
$$|¢_1,¢_2\cdots,¢_{n+1}\rangle$$

Quantum Money “Mini-scheme”

Simplified scheme in which mint produces only one banknote.

Complet … s output of MintOne w.h.p.

Public-Key Signature Scheme

Soundness: For any counterfeiter C, if

Full Quantum Money Scheme

th … $\mathrm{rOne}\left(s,|¢_2\rangle\right)$ rejects.

$\mathrm{VerOne}\left(s,|\$\rangle\right)$

$C\left(s,|\$_1\rangle\right)=|¢_1,¢_2\rangle$

Run KeyGen for a public key signature scheme

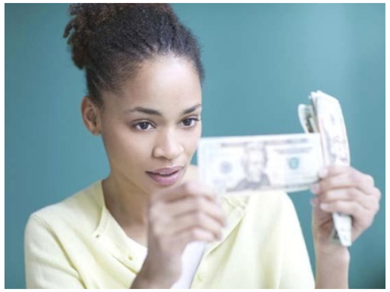
$k_{private}$

$$\mathrm{MintOne}\left(0^{k}\right)=\left(s,\left|\$\right\rangle\right)$$

$$\mathrm{Sign}_{k_{private}}\left(s\right)=\sigma\left(s\right)$$

k_{public}

$$\left(\sigma(s),\left|\$\right\rangle\right)$$

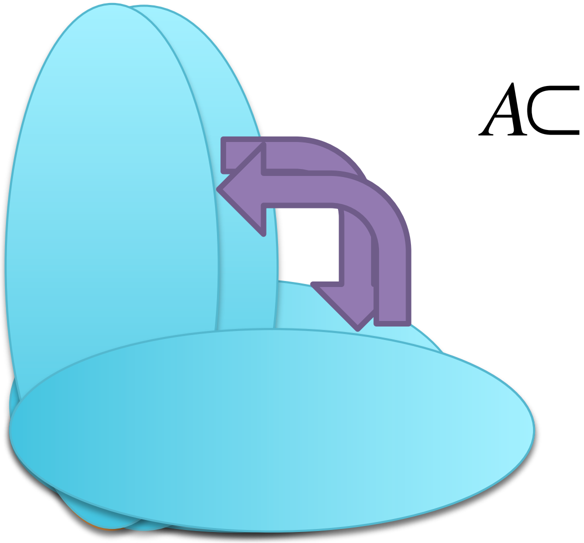


$$\mathrm{VerOne}\left(s,\left|\$\right\rangle\right)$$

$$\mathrm{Ver}_{k_{public}}\left(\sigma\left(s\right)\right)$$

Must either break signature scheme, or break mini-scheme.

The Hidden Subspace Scheme



$A \subset_R F_2^k \qquad \dim(A) = \frac{k}{2}$

$$|\$\rangle = |A\rangle = \frac{1}{2^{k/4}} \sum_{v \in A} |v\rangle$$

s is some data (TBD) which lets the user test membership in A and $A^\perp$.

$\mathrm{Ver}(|\$\rangle, s)$:

Apply membership test for A

Hadamard transform

Apply membership test for $A^\perp$ $\quad = |A\rangle\langle A|$

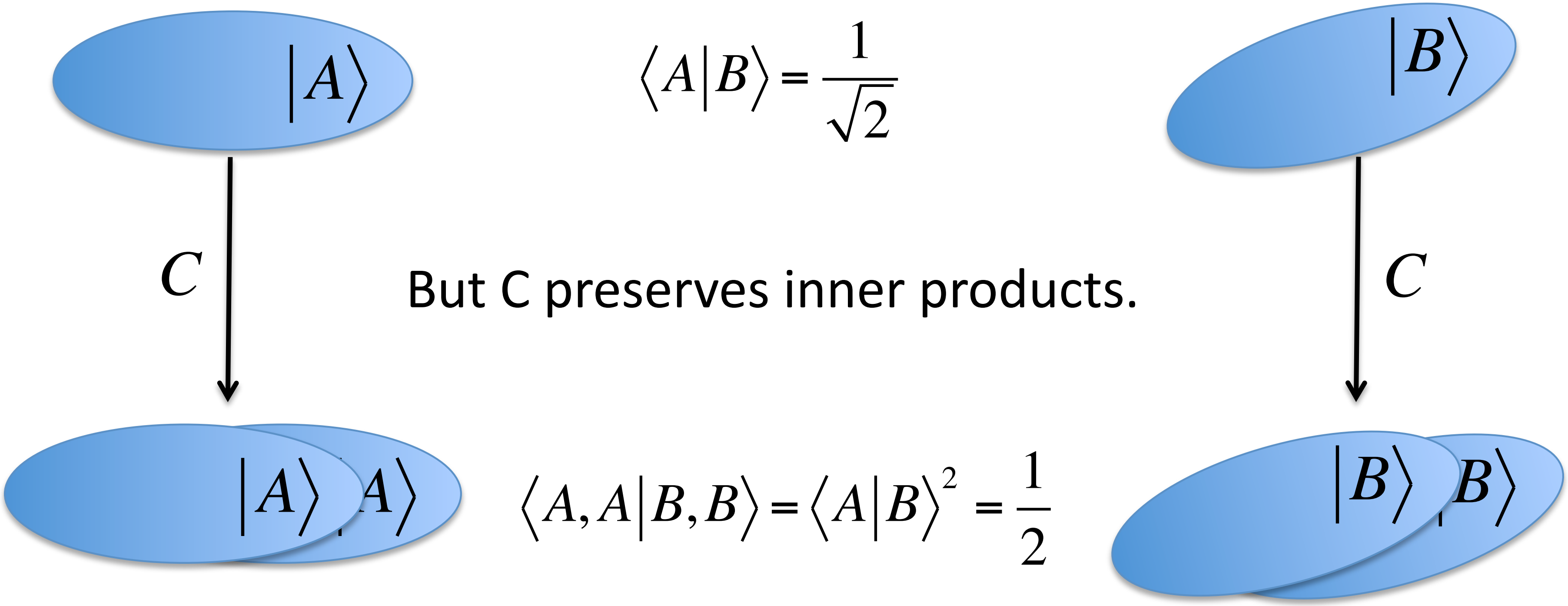
Hadamard transform $\quad \mathrm{Probability(Accept)} = \langle \$ | A \rangle^2$

Accept if both tests accept

Proof of “Black-Box” Security

Warm-up: Consider a counterfeiter C who doesn’t make use of *s* at all.

Let A and B be maximally overlapping subspaces.



$$\langle A|B\rangle = \frac{1}{\sqrt{2}}$$

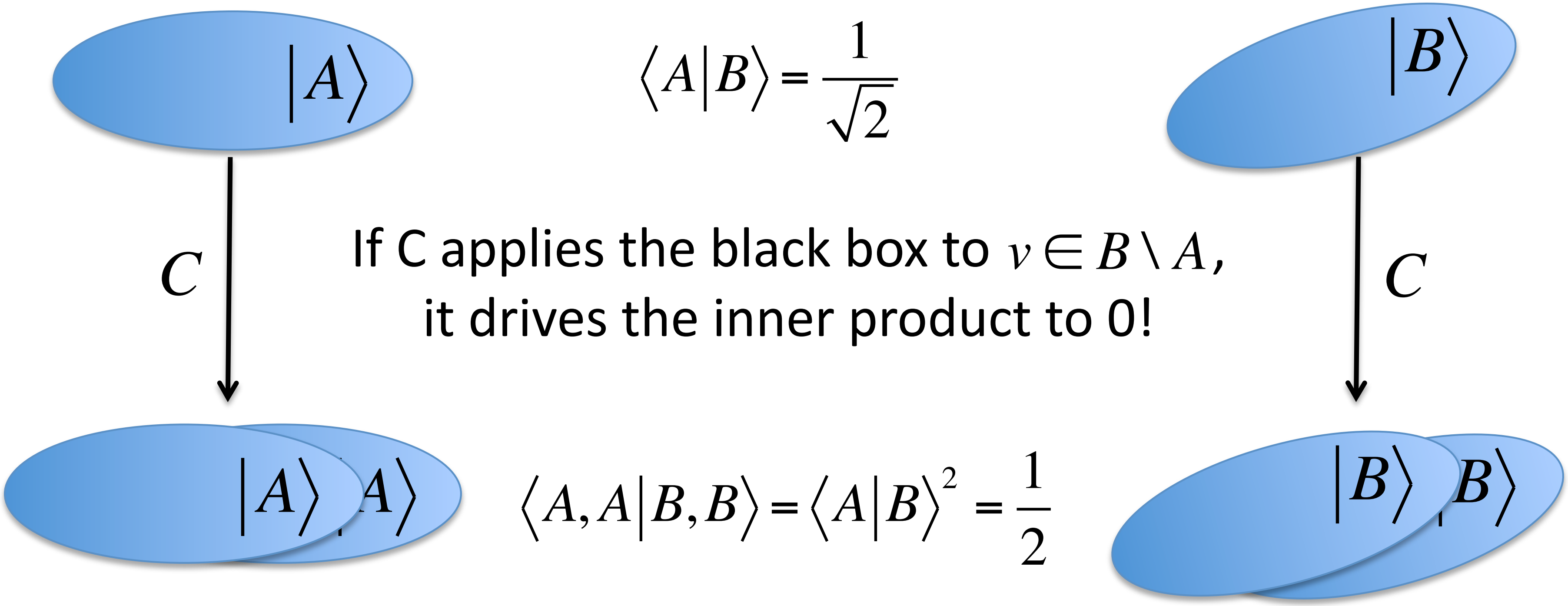
But C preserves inner products.

$$\langle A,A|B,B\rangle = \langle A|B\rangle^2 = \frac{1}{2}$$

Proof of “Black-Box” Security

Now consider a counterfeiting algorithm *C*
which uses *s* as a “black box”:

C has access to a different black box on different inputs.



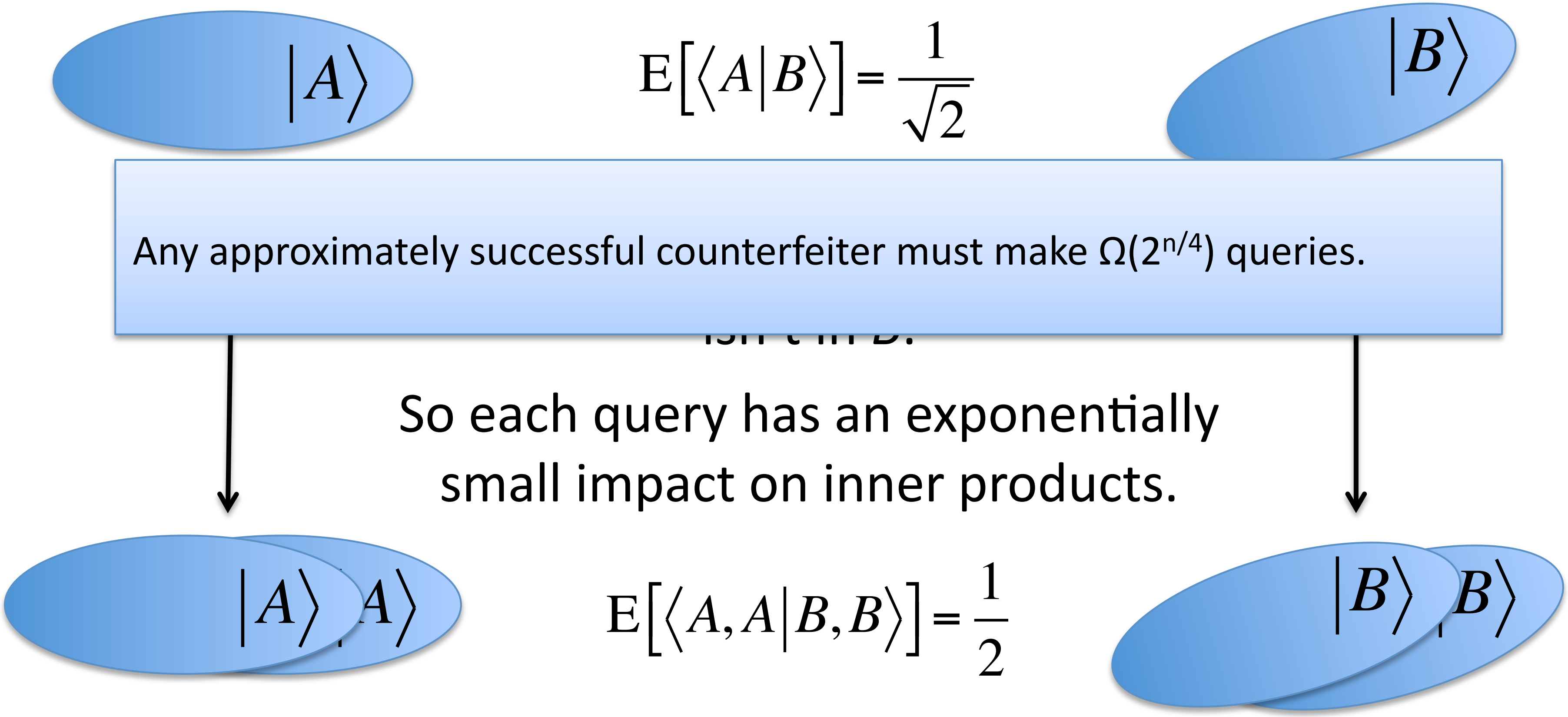
$$\langle A|B\rangle = \frac{1}{\sqrt{2}}$$

If C applies the black box to $v \in B \setminus A$,
it drives the inner product to 0!

$$\langle A, A|B, B\rangle = \langle A|B\rangle^2 = \frac{1}{2}$$

Inner-Product Adversary Method

Idea: Pick a uniformly random pair of (maximally overlapping) subspaces. Bound the *expected* inner product.



$$\mathrm{E}\left[\langle A|B\rangle\right]=\frac{1}{\sqrt{2}}$$

Any approximately successful counterfeiter must make $\Omega(2^{n/4})$ queries.

So each query has an exponentially small impact on inner products.

$$\mathrm{E}\left[\langle A,A|B,B\rangle\right]=\frac{1}{2}$$

Hiding Subspaces

Need to provide classical data which allows a user to test membership in A and $A^{\perp}$ without revealing them.

One solution: Represent A as a uniformly random system:

$$\begin{array}{c} p_1(x_1, x_2, \ldots, x_k) \\ p_2(x_1, x_2, \ldots, x_k) \\ \vdots \\ p_k(x_1, x_2, \ldots, x_k) \end{array} \quad \text{with} \quad \begin{array}{l} p_i(x_1, x_2, \ldots, x_k) = 0 \\ \forall (x_1, x_2, \ldots, x_k) \in A \end{array}$$

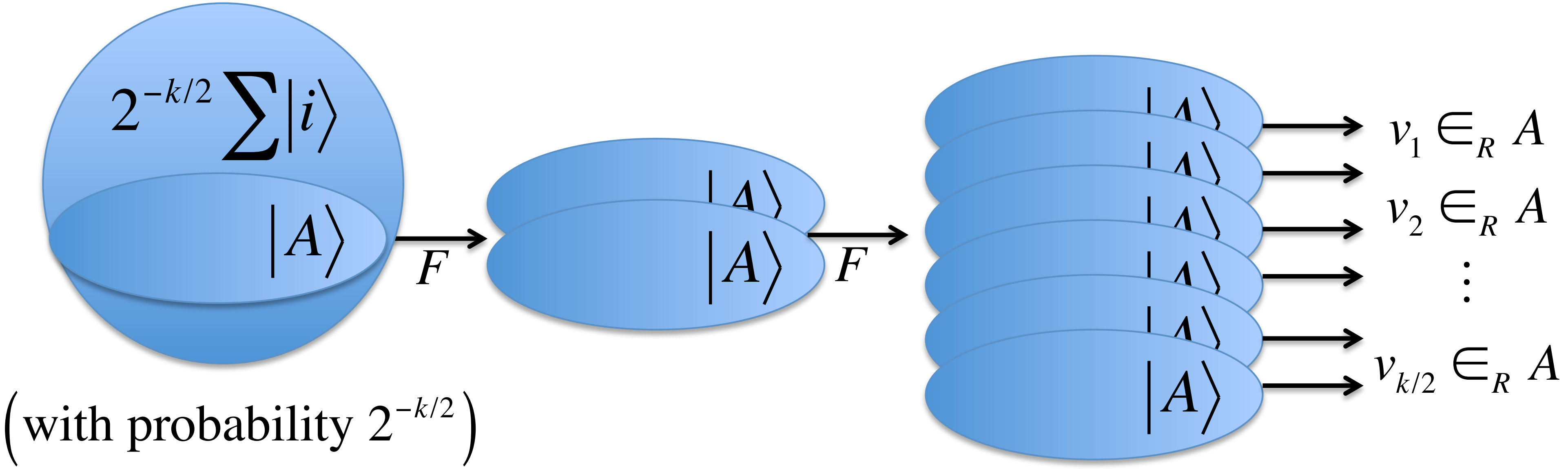
We can add any constant amount of noise.

To generate: sample polynomials which vanish when $x_1 = x_2 = \cdots = x_{k/2}$, then apply a change of basis.

Proof of Security

Conjecture: Given our obfuscations of A and $A^{\perp}$, no efficient quantum algorithm recovers a basis for A with probability $\Omega\left(2^{-k/2}\right)$.

Suppose there were an efficient forging algorithm F. Then we can violate the conjecture:



Status of Hardness Assumption

If $d = 1$, recovering A given noisy polynomials that vanish on A is equivalent to learning a noisy parity…

…but we can use a membership oracle for $A^{\perp}$ to remove the noise.

If $d \geq 2$, recovering A from a single polynomial is related to the *Polynomial Isomorphism* problem.

For $d = 2$ this is easy.

For $d = 3$, the problem can be solved with a single hint from A, which can be obtained with probability $2^{-k/2}$.

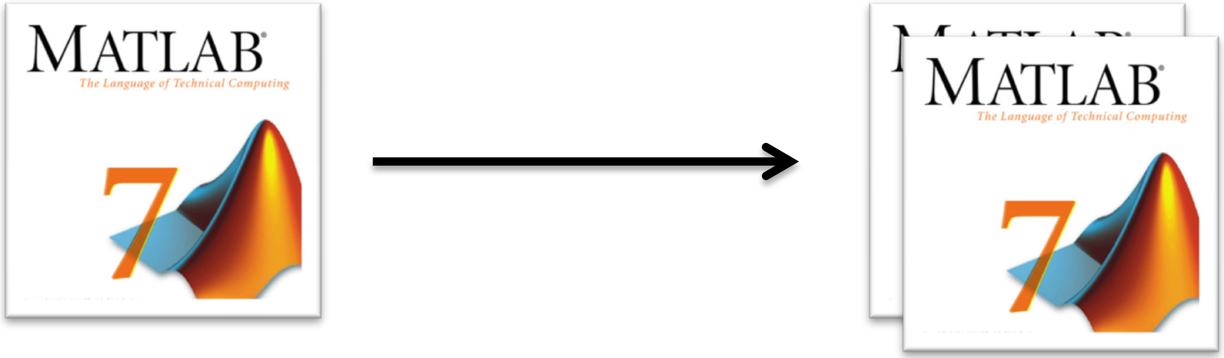
For $d \geq 4$, known techniques don't seem to work.

Quantum + Hardness Assumptions

- Most quantum cryptography tries to eliminate cryptographic assumptions.
- But quantum money requires both:
  - If an adversary keeps randomly generating forgeries, eventually they'll get lucky.
- Combining hardness assumptions with the uncertainty principle may make new primitives possible.
  - Money
  - Copy-protection
  - Obfuscation?
  - …?

Software Copy-Protection

Classical software can be freely copied.



To prevent copying, a vendor must interact with the user on every execution.

Can we design quantum "copy-protected" software?

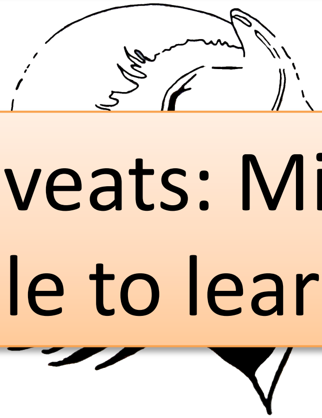
$|\psi\rangle$

Completeness: $\text{Eval}\left(|\psi\rangle, x\right) = C(x)$ w.h.p.

$\text{Eval}\left(|\psi\rangle, x\right) = C(x)$

Soundness: A pirate can't output two states either of which can be used to evaluate $C(x)$.

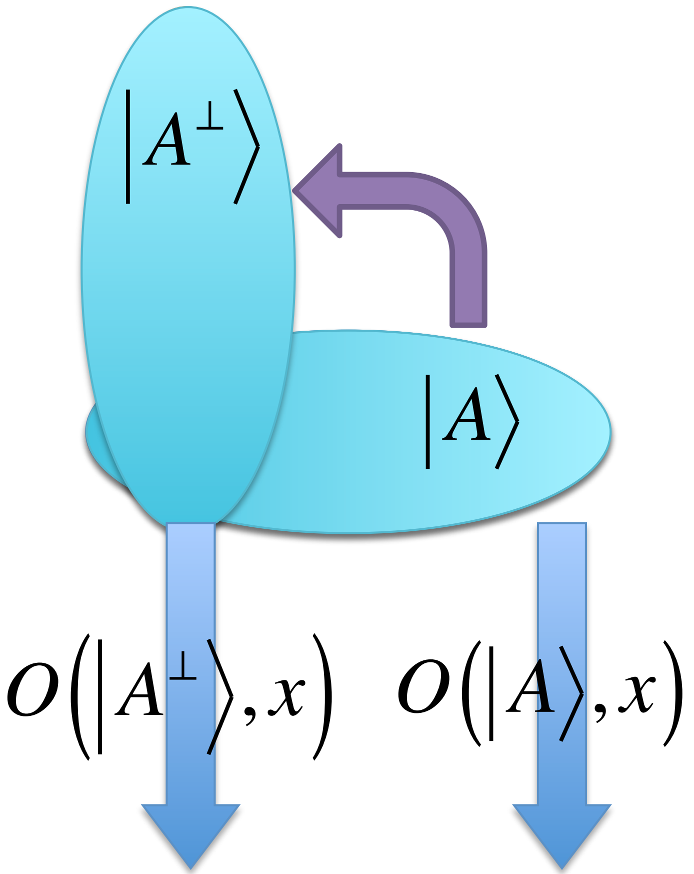
Caveats: Might be able to guess $C(x)$, might be able to learn an approximation to C…



$\text{Pirate}\left(|\psi\rangle\right) = |\varphi_1, \varphi_2\rangle$

$\text{Eval}^*\left(|\varphi_2\rangle, x\right) =_? C(x)$

Black-Box Copy-Protection Scheme



$$H(x) \oplus \left(C(x) \oplus H(x)\right) = C(x)$$

$$|\psi\rangle = |A\rangle = \frac{1}{2^{k/4}} \sum_{v \in A} |v\rangle$$

$$O(v, x) = \begin{cases} C(x) \oplus H(x) & v \in A \\ H(x) & v \in A^{\perp} \\ 0 & \text{otherwise} \end{cases}$$

For a random function $H(x)$

Sketch of Security Proof

Goal: construct a simulator, which uses Pirate to learn C
OR find an element of A and an element of $A^{\perp}$

If we halt both, we recover elements of A and $A^{\perp}$, which is ruled out by the inner product adversary method.

(We can simulate Pirate

So one of them runs successfully without using the oracle.
Therefore C is learnable, and we can't hope to stop Pirate!

$\text{Eval}^{*}\left(|\varphi_1\rangle, x\right)$

$\text{Eval}^{*}\left(|\varphi_2\rangle, x\right)$

If $O(v,x)$ is queried for some $v \in A$, halt and record v.

If $O(v,x)$ is queried for some $v \in A^{\perp}$, halt and record v.

Key idea: To make meaningful use of the oracle, $\text{Eval}^{*}\left(|\varphi_i\rangle\right)$ must use both an element of A and an element of $A^{\perp}$.

Program Obfuscation?

- Challenge: Given C, produce Obfuscation(C), which allows the user to evaluate C but learn nothing else.
- Known to be impossible classically…
- …but the possibility of quantum obfuscation remains open (even of quantum circuits!)

$|\psi\rangle$

Completeness: $\mathrm{Eval}\left(|\psi\rangle, x\right) = C(x)$ w.h.p.

$\mathrm{Obfuscate}(C) = |\psi\rangle$

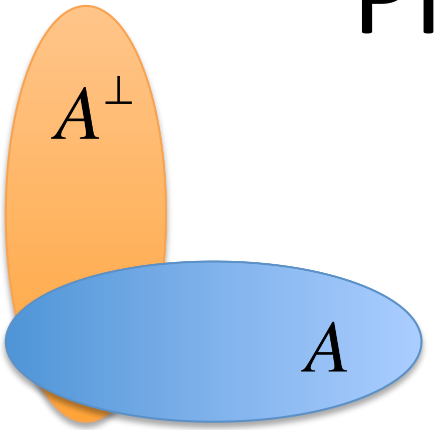
$\mathrm{Eval}\left(|\psi\rangle, x\right) = C(x)$

Soundness: any measurement can be simulated using only black-box access to C.

Makes an arbitrary measurement of $|\psi\rangle$

Simulated by simulator with black-box access to C

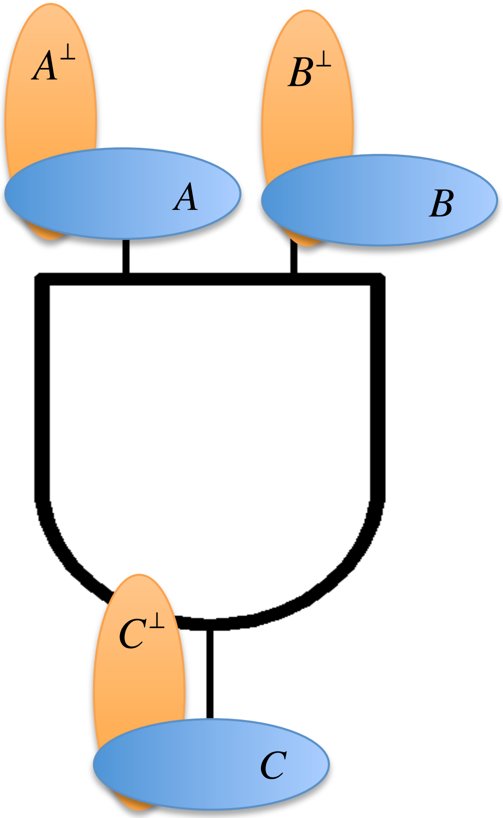
Program Obfuscation?



The state $|A\rangle$ acts like a non-interactive 1-of-2 oblivious transfer.

Q: Can we implement Yao's garbled circuits, with hidden subspaces as secrets instead of encryption keys?

A: Yes, but hard to determine security.



Open Questions

- Break our candidate money scheme based on multivariate polynomials (?)
- Come up with new implementations of hidden subspaces
- Copy-protection without an oracle
- Program obfuscation
- Given oracle access to a subspace, prove you can't find a basis with probability $\Omega\left(2^{-k/2}\right)$.

Questions?