# Simple Tests of Quantumness Also Certify Qubits


Zvika Brakerski (Weizmann Institute of Science)

Alexandru Gherghiu (Chalmers University of Technology)

Gregory D Kahanamoku-Meyer (University Of California, Berkeley)

**Eitan Porat (Weizmann Institute of Science)**

Thomas Vidick (Weizmann Institute of Science)



Crypto 2023

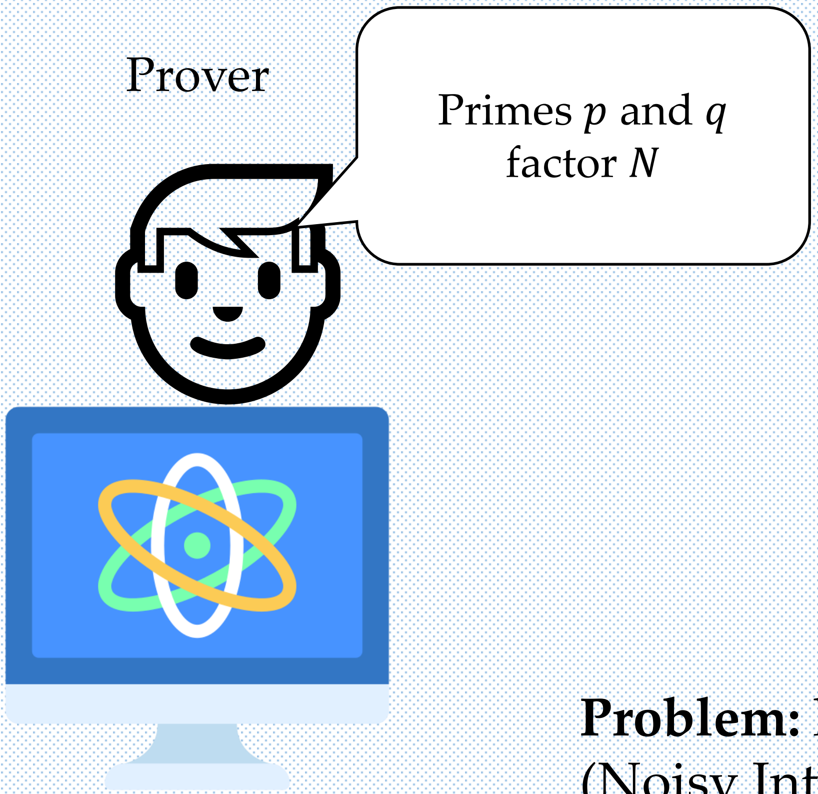
# Quantum Supremacy (Test of Quantumness)

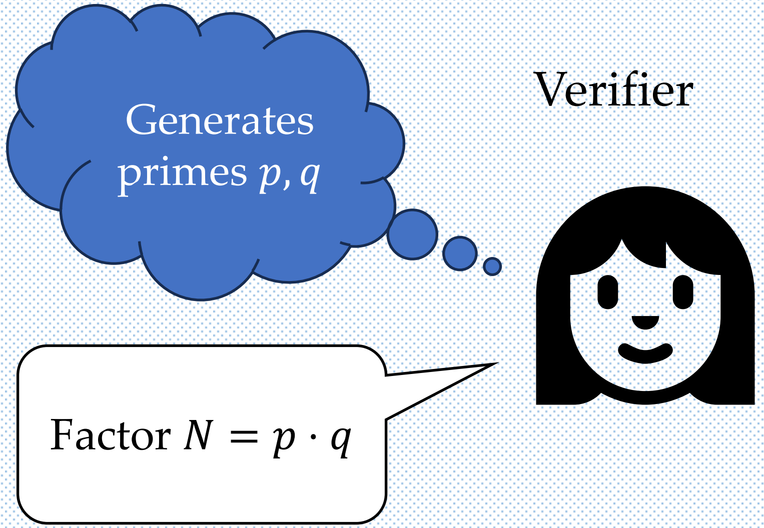
- Perform computations that outperforms classical computers.
- A need for efficiently-verifiable quantum advantage.

**Google Sycamore**

Image Rights: Forest Stearns, Google AI Quantum Artist in Residence

# Simple Example of Proof of Quantumness [Shor’94]




**Problem:** hard to implement on NISQ (Noisy Intermediate-Scale Quantum) Computers.

# Previous Works

- BCMVV'18[1] – Proof of Quantumness based on LWE with **adaptive-hardcore bit**.
  - Requires an aggressive setting of parameters for LWE which hampers practical implementation.
- YZ'22[2] – Proofs of Quantumness in the **random oracle model**.
- Recently, two proposals of protocols in the standard model with less computational assumptions KCVY'21[3] and KLVY'22[4].


1. A Cryptographic Test of Quantumness and Certifiable Randomness from a Single Quantum Device, Z. Brakerski, P. Christiano, U. Mahadev, U. Vazirani, T. Vidick, 2018
2. Verifiable Quantum Advantage without Structure, T. Yamakawa, M. Zhandry, 2022
3. Classically-Verifiable Quantum Advantage from a Computational Bell Test,  G. Kahanamoku-Meyer, S. Choi, U. Vazirani, N. Yao.
4. Quantum Advantage from Any Non-Local Game, Y. Tauman Kalai, A. Lombardi, V. Vaikuntanathan, L. Yang

# Beyond Quantum Supremacy

- Suppose we have a NISQ computer which achieves Quantum Supremacy
- Could we delegate computation to the Quantum computer?
- Could we make it generate certifiable randomness?
- Qubit Certification - a useful building block for quantum verification protocols.

# Qubit Certification

- Could we verify that the quantum computer has a qubit?
- What does it mean to “have” a qubit?

# Qubit Certification

- **Operational view of Qubits*:** the prover has a triplet $(|\psi\rangle, X, Z)$ where $X$ and $Z$ are binary operators which "approximately anti-commute" on $|\psi\rangle$.
- Could we use existing proofs of Quantumness as tests for qubits?
- Yes!

* Course FSMP, Fall'20: Interactions with Quantum Devices, Thomas Vidick, 2022

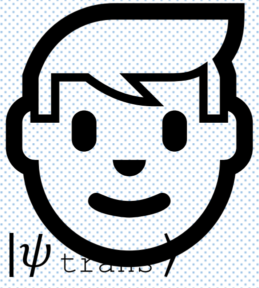
# Our Results

- For a specific class of protocols, we show:
  - A quantum soundness barrier against quantum cheating provers (vs classical soundness).
  - Provers that approach the quantum soundness barrier *must perform anti-commuting measurements* (a qubit test).
- NZ'23 show related results for the KLVY'22 protocol. Prove how it can be used to get a protocol *for delegation of quantum computation*.

# Our Protocol Template

Prover

Verifier



$|\psi_{\texttt{trans}}\rangle$

*quantum state*

$\texttt{rand}$

$\texttt{trans}$ *transcript of the protocol*

**PHASE 1**

⋮

$\texttt{flag} \in \{\texttt{acc, rej, cont}\}$

**if** $\texttt{flag = cont:}$

**PHASE 2**

$m \in \{0,1\}$ *challenge*

$b \in \{0,1\}$ *response*

verifier accepts if $(-1)^b = \hat{c}_m(\texttt{rand, trans})$

# Soundness for classical provers – Sketch

- Prove that it is hard (**classical)** to compute the parity of both challenges: $\hat{c}_0 \cdot \hat{c}_1$ with some noticeable advantage
- Show that a classical adversary that achieves $\frac{3}{4} + \varepsilon$ success probability can employ a rewinding startegy to compute the parity.

# Computing Parity in the Quantum World

- **Problem:** Quantum computers cannot perform rewinding…
- Could they somehow compute the parity with some noticeable advantage?

# Modeling Quantum Provers

- For each $m \in \{0,1\}$ (challenge bit) the prover performs a set projective measurement on its state $|\psi_{\texttt{trans}}\rangle$

$$\{\Pi_b^m\}$$

Challenge bit

Response bit

# Parity Algorithm

(*Algorithm* $\mathcal{A}_1$)

- Execute Phase 1 of the protocol template to obtain (trans, $|\psi_{\texttt{trans}}\rangle\rangle$ and a flag.

(*Algorithm* $\mathcal{A}_2$)

- $b_0 =$ measurement of $\mathcal{H}_P$ using $\{\Pi_0^0, \Pi_1^0\}$.
- $b_1 =$ measurement of $\mathcal{H}_P$ using $\{\Pi_0^1, \Pi_1^1\}$.
- Return $b_0 \oplus b_1$.

# Soundness for quantum provers - sketch

- Prove that it is hard (**quantum)** to compute the parity of both challenges: $\hat{c}_0 \cdot \hat{c}_1$
- **Quantum Analogue:** Show that a quantum adversary that achieves $\cos^2\left(\frac{\pi}{8}\right) + \varepsilon$ success probability, using the parity algorithm can compute parities

# Parity Hardness → Quantum Soundness

No classical (quantum) polynomial time algorithm **guesses** $\hat{\boldsymbol{c}}_{\mathbf{0}} \cdot \hat{\boldsymbol{c}}_{\mathbf{1}}$ with non-negligible advantage

→

Then no classical (quantum) polynomial-time prover **succeeds in the protocol template** with probability larger than 75% (resp. $\cos^2(\pi/8) \approx 85\%$) by more than a negligible amount

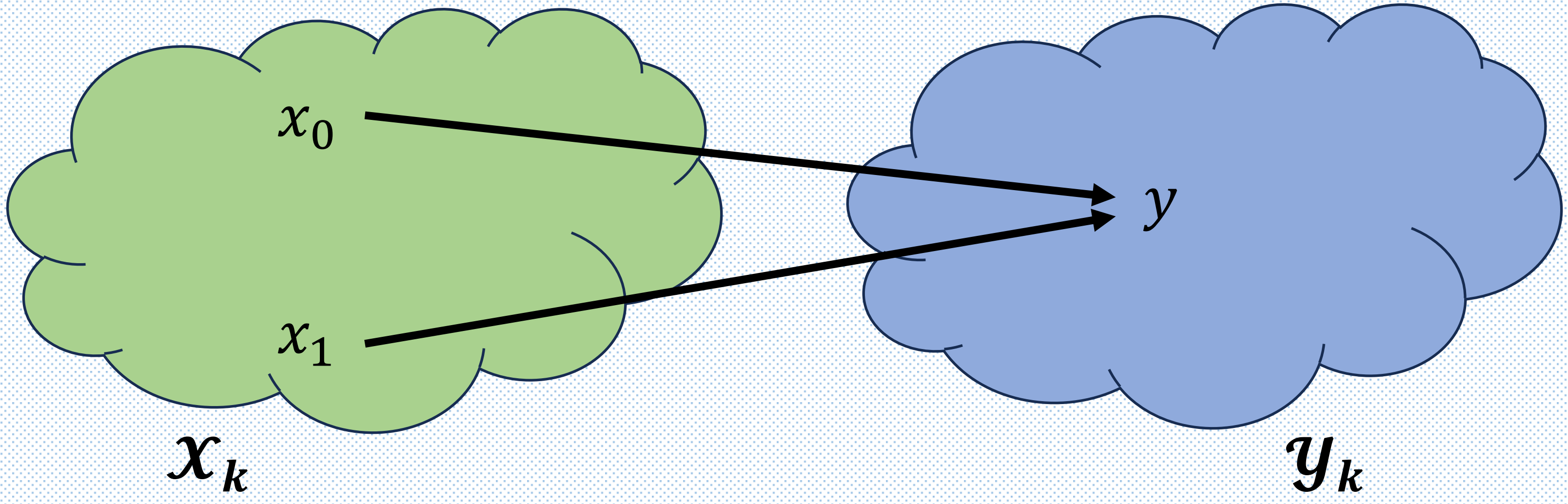
# Qubit Test

- The quantum soundness result gives us a qubit test
- If a prover approaches the soundness barrier, then the measurements $Q_0 = \Pi^0_{\hat{c}_0}$ and $Q_1 = \Pi^1_{\hat{c}_1}$ must be close to anti-commuting

# Example: KCVY Protocol

# Trapdoor claw-free functions

- Keyed functions $f_k\colon \mathcal{X}_k \to \mathcal{Y}_k$ with trapdoor $t_k$




- **Hard (quantum)** to find a **claw** $(x_0, x_1)$ such that $f_k(x_0) = f_k(x_1)$
- Given trapdoor $t_k$, for each $y$ easy to find $f_k(x_0) = f_k(x_1) = y$

# Trapdoor claw-free functions – cntd.

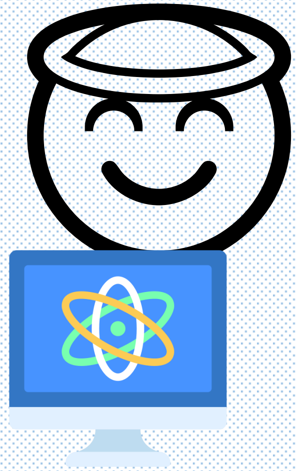
- Efficiently generate superposition

$$\frac{1}{\sqrt{|\mathcal{X}_k|}}\sum_x |x\rangle|f_k(x)\rangle$$

- Efficiently distinguish between the preimages $x_0$ and $x_1$

# KCVY Protocol - Simplified

Honest Prover

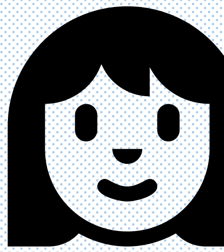


**PHASE 1**

1. Generates $\sum_x |x\rangle_{\mathcal{X}} |f_k(x)\rangle_{\mathcal{Y}}$
2. Measures $\mathcal{Y}$ register
   $(|x_0\rangle_{\mathcal{X}} + |x_1\rangle_{\mathcal{X}})|y\rangle_{\mathcal{Y}}$
3. Sends $y$ to the verifier.

Generates $k, t_k$ for a TCF

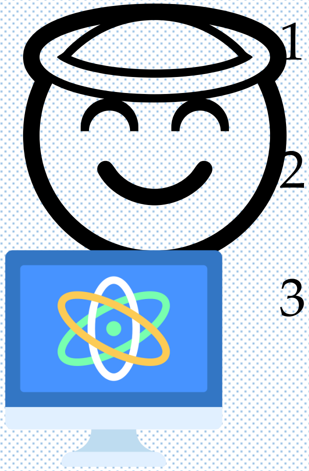
Verifier



$k$ - Key for TCF

# KCVY Protocol - Simplified

Honest Prover

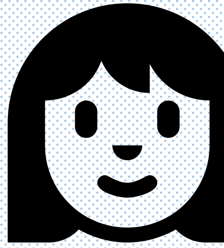


**PHASE 1**

1. Computes ancilla bit
$$|0\rangle|x_0\rangle_\mathcal{X} + |1\rangle|x_1\rangle_\mathcal{X}$$
2. Using ancilla
$$|0\rangle|x_0\rangle_\mathcal{X}|r_0 \cdot x_0\rangle_\mathcal{X} + |1\rangle|x_1\rangle_\mathcal{X}|r_1 \cdot x_1\rangle_\mathcal{X}$$
3. Uncomputes ancilla
$$|x_0\rangle_\mathcal{X}|r_0 \cdot x_0\rangle + |x_1\rangle_\mathcal{X}|r_1 \cdot x_1\rangle$$

Generates $k, t_k$ for a TCF

Verifier

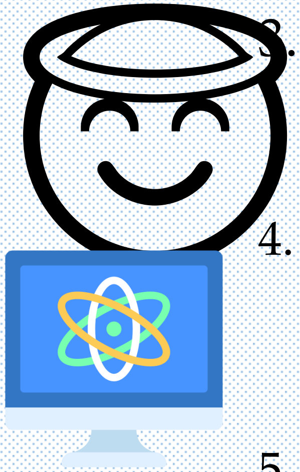


$k$ - Key for TCF

$r_0, r_1 \leftarrow_R \{0,1\}^n$

# KCVY Protocol - Simplified

Honest Prover



**PHASE 1**

3. Computes Hadamard on $\mathcal{X}$ register

$$\sum_d |d\rangle_{\mathcal{X}} \left( (-1)^{d\cdot x_0} |r_0 \cdot x_0\rangle + (-1)^{d \cdot x_1} |x_1\rangle_{\mathcal{X}} |r_1 \cdot x_1\rangle \right)$$

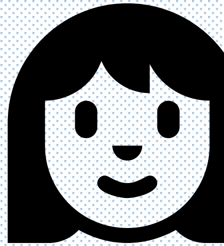
4. Measures $\mathcal{X}$ register

$$|d\rangle_{\mathcal{X}} \left( (-1)^{d\cdot x_0} |r_0 \cdot x_0\rangle + (-1)^{d \cdot x_1} |x_1\rangle_{\mathcal{X}} |r_1 \cdot x_1\rangle \right)$$

5. Sends $d$ to the verifier

Generates $k, t_k$ for a TCF

Verifier



$k$ = Key for TCF

$r_0, r_1 \leftarrow_R \{0,1\}^n$

# KCVY Protocol - Simplified

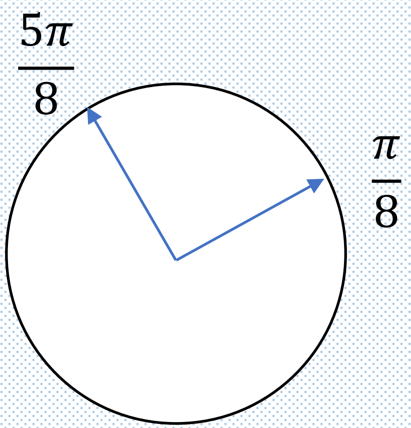
Honest Prover

**PHASE 2**

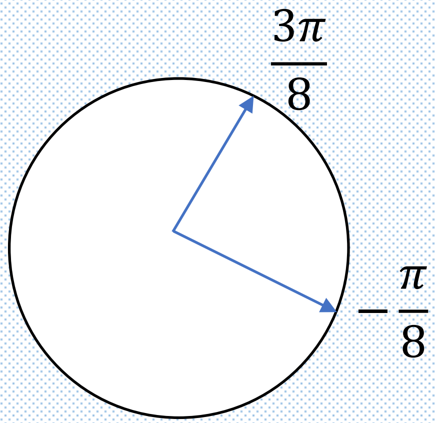
Holds the state

$$|\psi\rangle = |r_0 \cdot x_0\rangle + (-1)^{d\cdot(x_0 \oplus x_1)}|x_1\rangle_{\mathcal{X}}|r_1 \cdot x_1\rangle$$

$m = 0$




$m = 1$




Sends $b$ the outcome of the measurement.

Generates $k, t_k$ for a TCF

Verifier

$k$ = Key for TCF

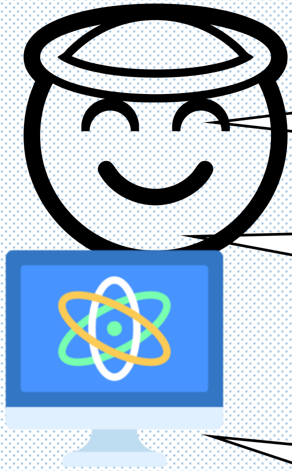
$r_0, r_1 \leftarrow_R \{0,1\}^n$

$m \in \{0,1\}$

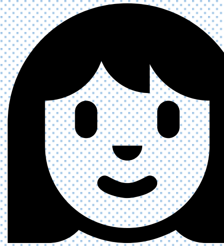
# KCVY Protocol - Simplified

Generates $k, t_k$ for a TCF

Honest Prover



Verifier



**PHASE 1**

$y \in \mathcal{Y}_k$

Generate claw and measure $y$

$k$ = Key for TCF

$d \in \{0,1\}^n$

Multiply by $r_0, r_1$
& Perform Hadamard measurement

$r_0, r_1 \leftarrow_R \{0,1\}^n$

**PHASE 2**

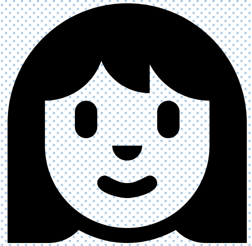
$b \in \{0,1\}$

Challenge-Response

$m \in \{0,1\}$

# KCVY Protocol - Simplified

Verifier



Accept if $b$ is the “expected” measurement outcome

Using trapdoor $t_k$ can find $x_0$ and $x_1$

Computes $\hat{c}_m(x_0, x_1, r_0, r_1, d)$

Accepts if $(-1)^b = \hat{c}_m$

# Post-Quantum TCF → Hardness of parity

- Easy to see that $\hat{c}_0 \cdot \hat{c}_1 = (-1)^{r_0 \cdot x_0 \oplus r_1 \cdot x_1} = (-1)^{r \cdot (x_0 || x_1)}$ where $r = r_0 || r_1$

Post-Quantum Trapdoor claw-free functions

*Quantum Goldreich-Levin**

→

Hardness of Computing Parity

* A quantum Goldreich-Levin theorem with cryptographic applications, Mark Adcock, Richard Cleve, 2002

# Open Questions

- Could we generalize our approach to the tests of quantumness in BCMVV'18 and the ones that operate in the random oracle model?
- A hierarchy of "capabilities"
  - What is the minimal basis for achieving these capabilities?