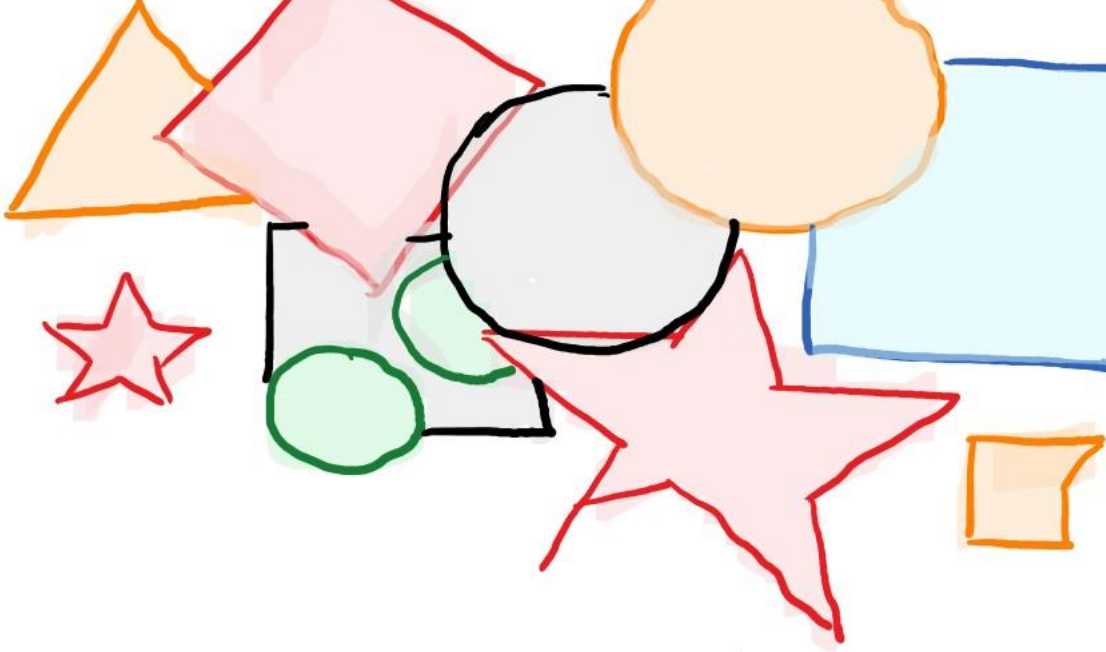
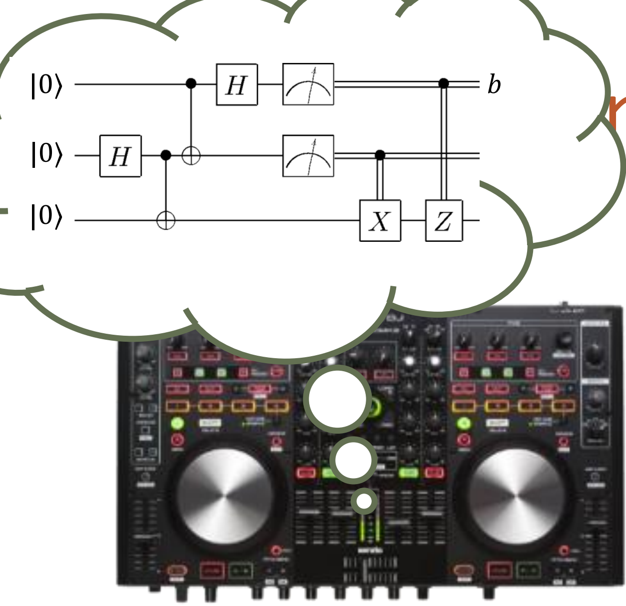


Delegating quantum computations

1. *Problem formulation*
2. *Overview of existing approaches*
3. *An open question*

Problem formulation

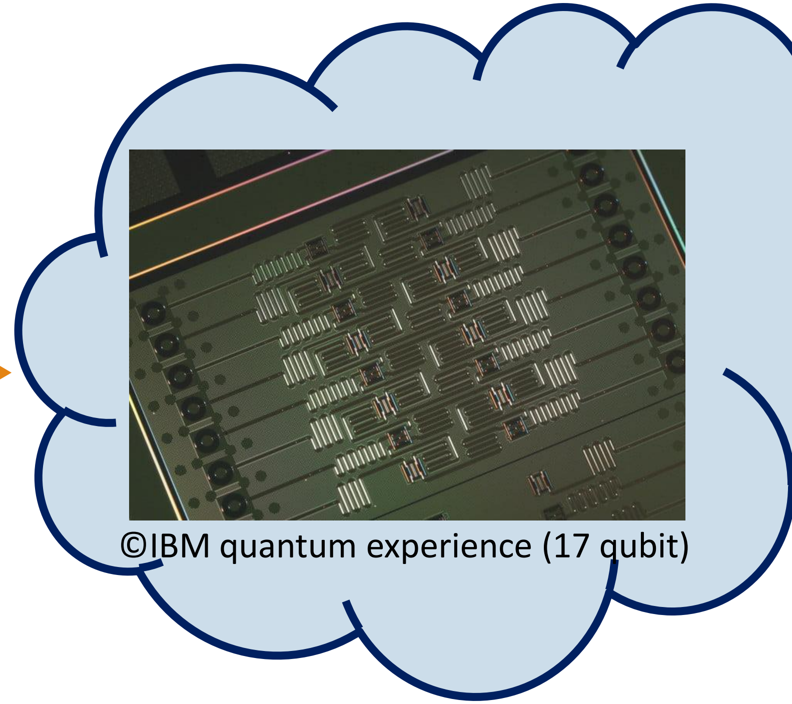
mulation



user

classical or quantum

communication



©IBM quantum experience (17 qubit)

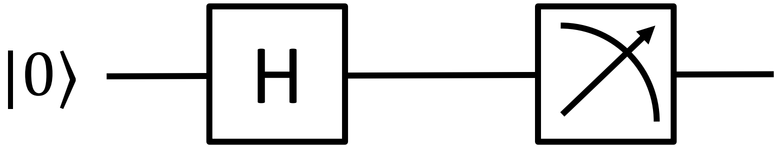
device

$\rightarrow (\boldsymbol{flag}, \boldsymbol{b})$

- Verifier has quantum computation C
- Multiple rounds of interaction with quantum device
- Verifier returns $(flag, b)$ s.t. $flag \in \{acc, rej\}$ and $b \in \{0,1\}$
- Goal: Whenever $\Pr(flag = acc)$ is non-negligible,

$$\Pr(\, b = 1 \,|\, flag = acc) \quad \approx \quad \Pr(\, C \text{ returns 1 on input } |0^n\rangle \quad)$$

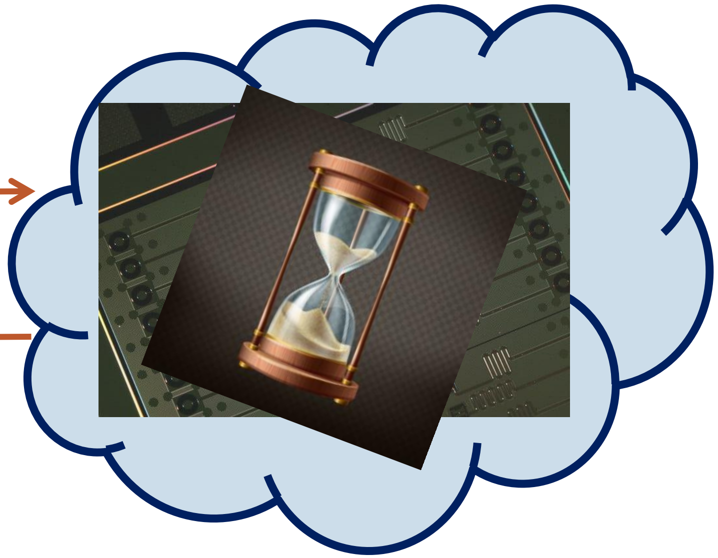
An example



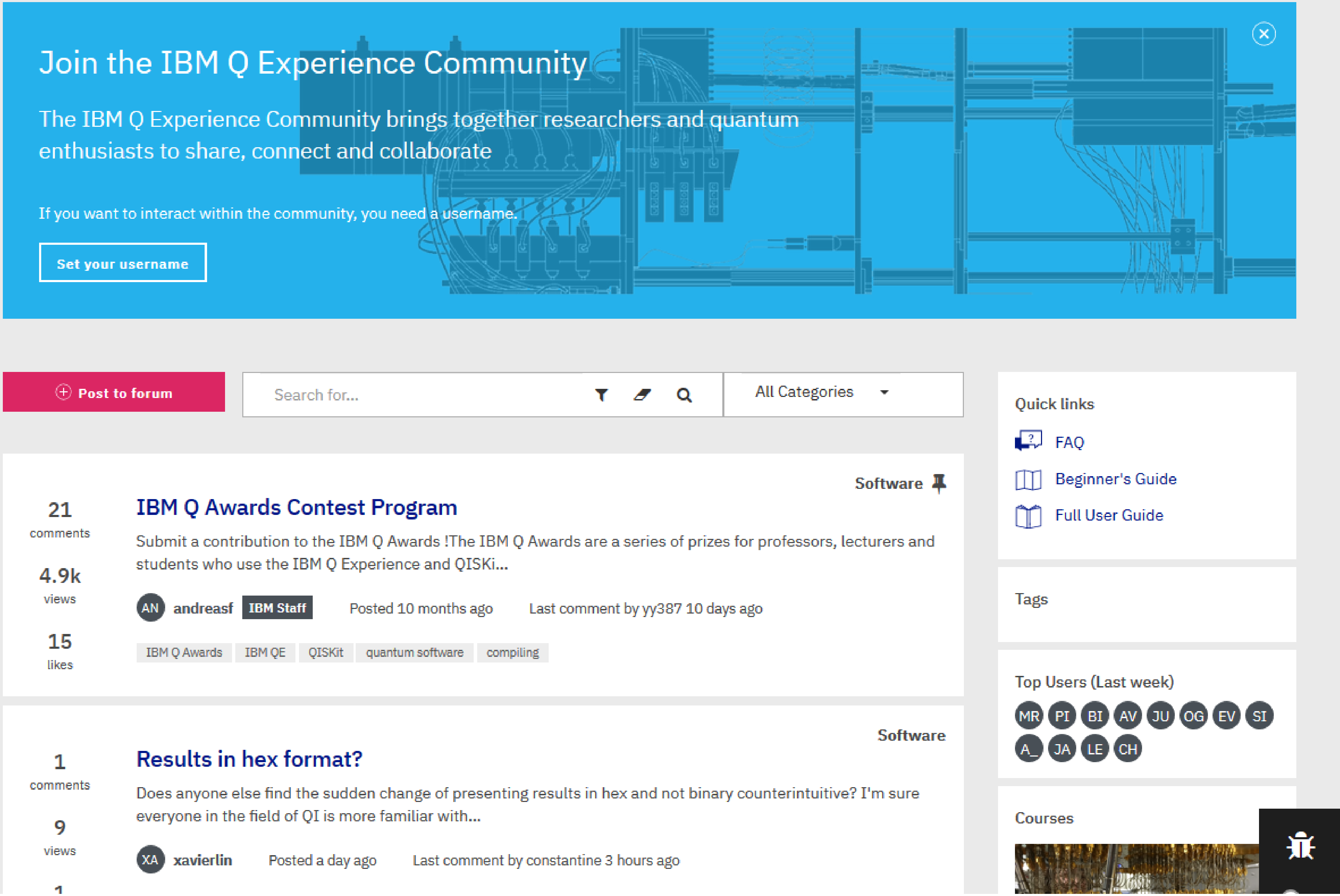
$$\begin{cases} b = 0 \text{ w.p. } 50\% \\ b = 1 \text{ w.p. } 50\% \end{cases}$$

“description of circuit C”

“I got $b = 0$”



Really??



Join the IBM Q Experience Community

The IBM Q Experience Community brings together researchers and quantum enthusiasts to share, connect and collaborate

If you want to interact within the community, you need a username.

Set your username

Post to forum

Search for...

All Categories

Quick links

FAQ

Beginner's Guide

Full User Guide

Tags

Top Users (Last week)

MR PI BI AV JU OG EV SI A_ JA LE CH

Courses

Software

21 comments

4.9k views

15 likes

IBM Q Awards Contest Program

Submit a contribution to the IBM Q Awards !The IBM Q Awards are a series of prizes for professors, lecturers and students who use the IBM Q Experience and QISKi...

AN andreasf IBM Staff Posted 10 months ago Last comment by yy387 10 days ago

IBM Q Awards · IBM QE · QISKit · quantum software · compiling

---

Software

1 comments

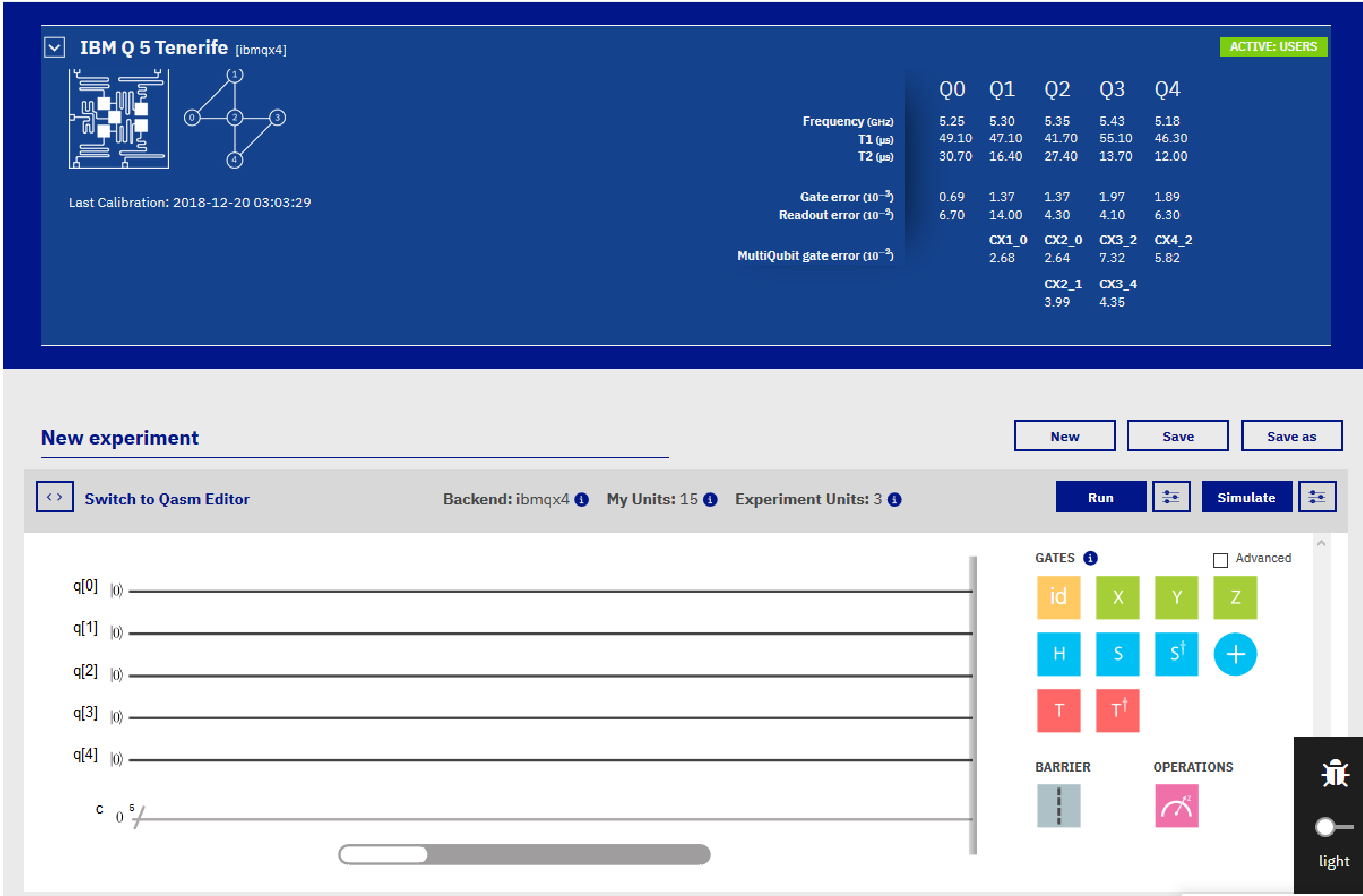
9 views

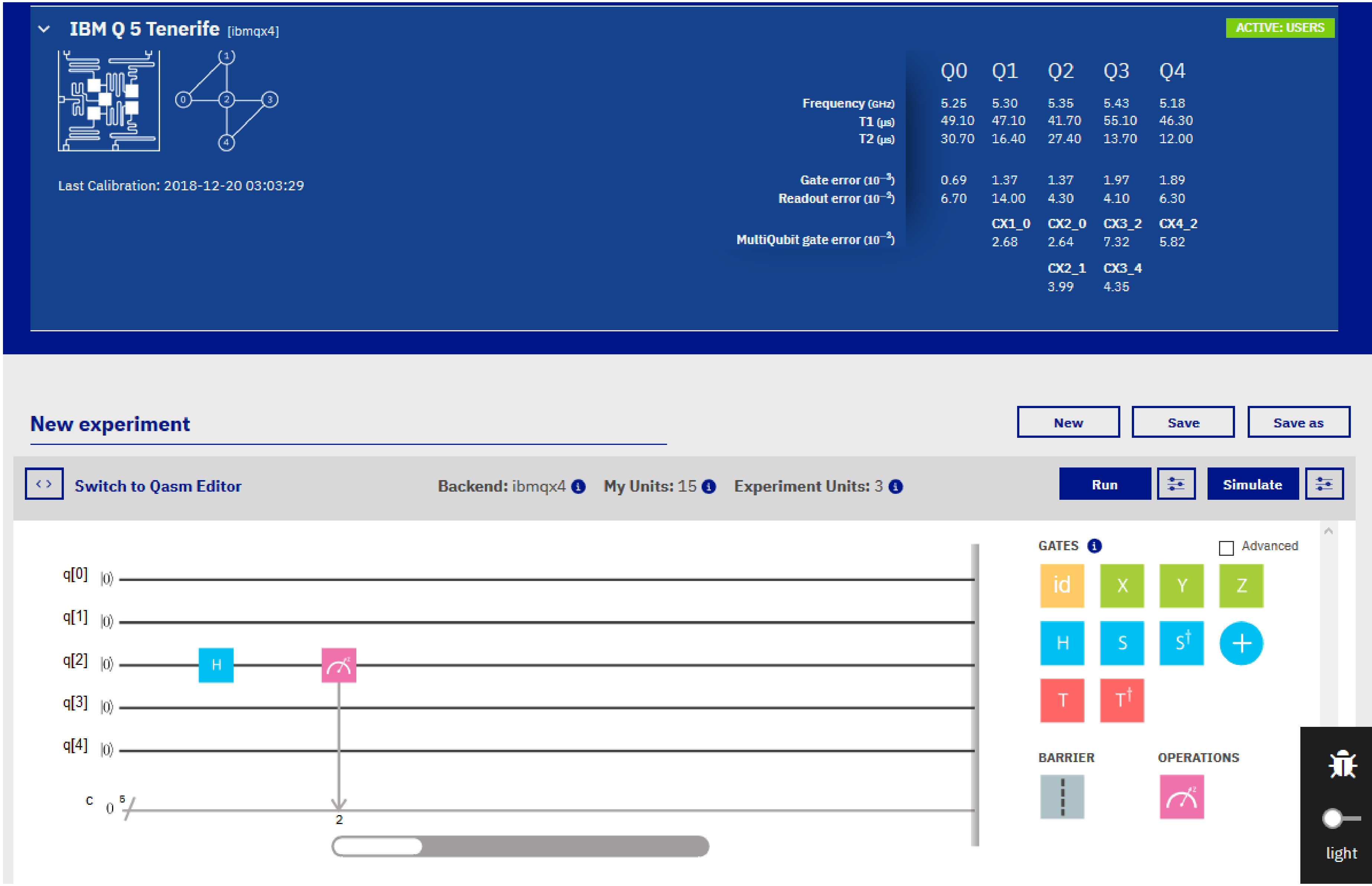
1

Results in hex format?

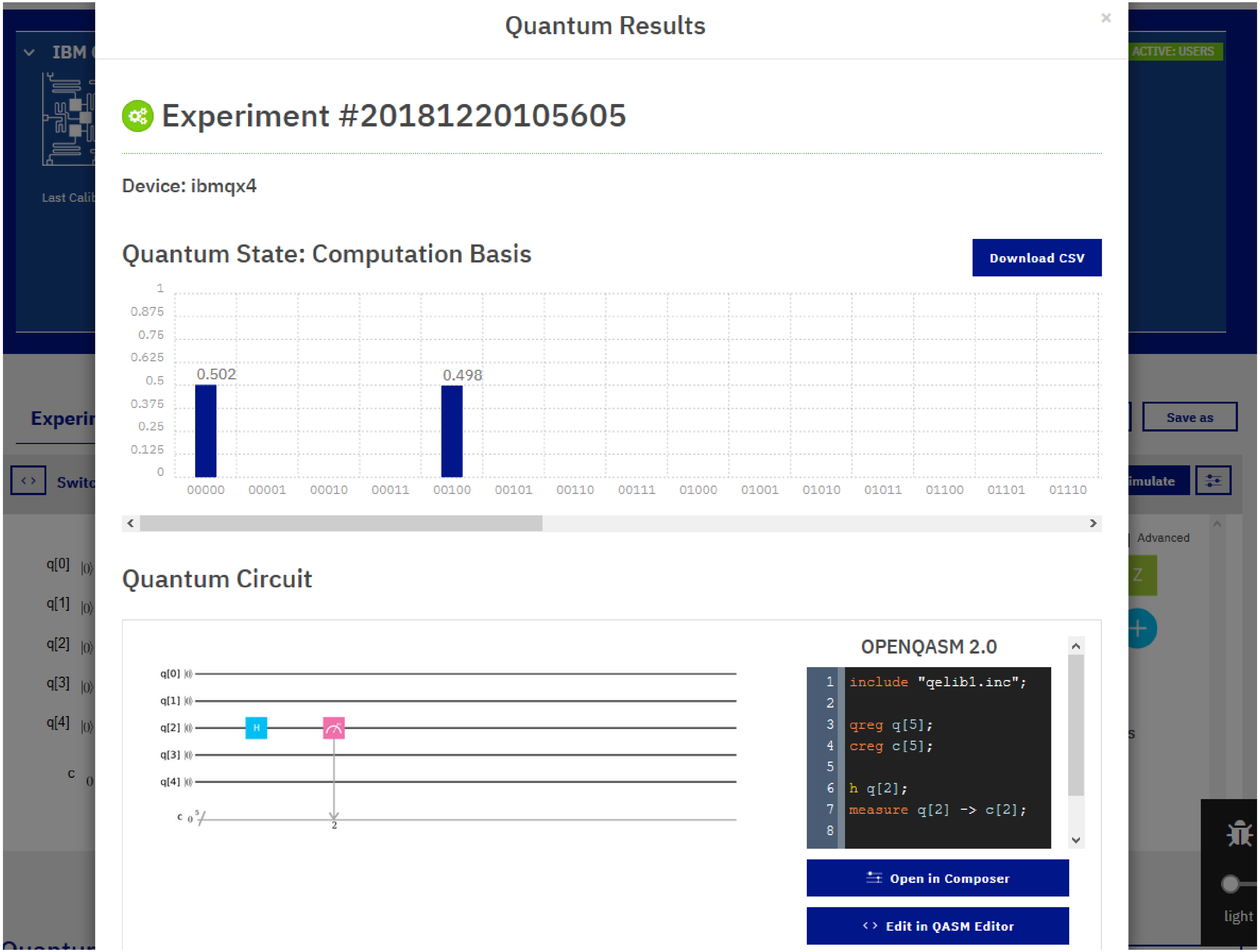
Does anyone else find the sudden change of presenting results in hex and not binary counterintuitive? I'm sure everyone in the field of QI is more familiar with...

XA xavierlin Posted a day ago Last comment by constantine 3 hours ago

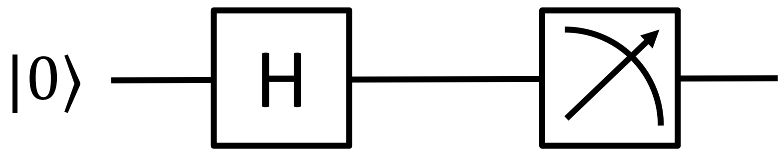

IBM Q 5 Tenerife [ibmqx4]
ACTIVE: USERS
Last Calibration: 2018-12-20 03:03:29
Q0 Q1 Q2 Q3 Q4
Frequency (GHz) 5.25 5.30 5.35 5.43 5.18
T1 (µs) 49.10 47.10 41.70 55.10 46.30
T2 (µs) 30.70 16.40 27.40 13.70 12.00
Gate error (10^-3) 0.69 1.37 1.37 1.97 1.89
Readout error (10^-2) 6.70 14.00 4.30 4.10 6.30
MultiQubit gate error (10^-2)
CX1_0 2.68
CX2_0 2.64
CX3_2 7.32
CX4_2 5.82
CX2_1 3.99
CX3_4 4.35
New experiment
New
Save
Save as
Switch to Qasm Editor
Backend: ibmqx4
My Units: 15
Experiment Units: 3
Run
Simulate
q[0] |0⟩
q[1] |0⟩
q[2] |0⟩
q[3] |0⟩
q[4] |0⟩
c 0 5
GATES
Advanced
id X Y Z
H S S† +
T T†
BARRIER
OPERATIONS
light



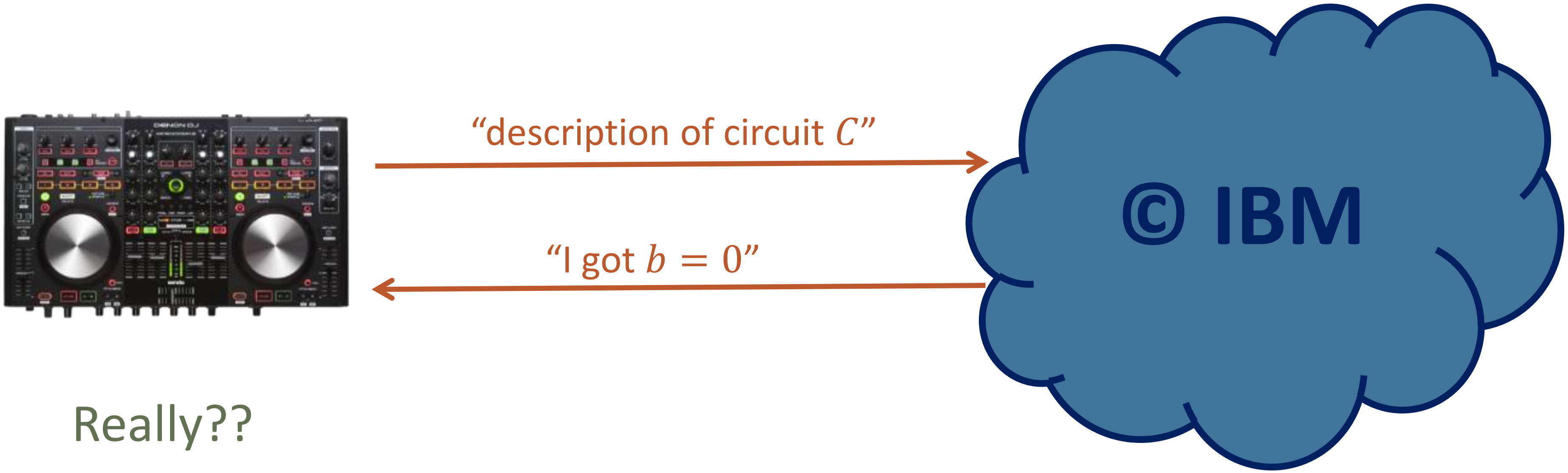
IBM Q 5 Tenerife [ibmqx4]
ACTIVE: USERS
Last Calibration: 2018-12-20 03:03:29
Q0 Q1 Q2 Q3 Q4
Frequency (GHz) 5.25 5.30 5.35 5.43 5.18
T1 (µs) 49.10 47.10 41.70 55.10 46.30
T2 (µs) 30.70 16.40 27.40 13.70 12.00
Gate error (10^{-3}) 0.69 1.37 1.37 1.97 1.89
Readout error (10^{-3}) 6.70 14.00 4.30 4.10 6.30
MultiQubit gate error (10^{-3})
CX1_0 2.68
CX2_0 2.64
CX3_2 7.32
CX4_2 5.82
CX2_1 3.99
CX3_4 4.35
New experiment
New
Save
Save as
Switch to Qasm Editor
Backend: ibmqx4
My Units: 15
Experiment Units: 3
Run
Simulate
q[0]
q[1]
q[2]
q[3]
q[4]
c
H
GATES
Advanced
id
X
Y
Z
H
S
S†
T
T†
BARRIER
OPERATIONS
light


Quantum Results
Experiment #20181220105605
Device: ibmqx4
Quantum State: Computation Basis
Download CSV
1
0.875
0.75
0.625
0.5
0.375
0.25
0.125
0
0.502
0.498
00000
00001
00010
00011
00100
00101
00110
00111
01000
01001
01010
01011
01100
01101
01110
Quantum Circuit
q[0] |0⟩
q[1] |0⟩
q[2] |0⟩
q[3] |0⟩
q[4] |0⟩
c 0
H
2
OPENQASM 2.0
include "qelib1.inc";
qreg q[5];
creg c[5];
h q[2];
measure q[2] -> c[2];
Open in Composer
Edit in QASM Editor
Save as
ACTIVE: USERS
Advanced
light

An example



$$\begin{cases} b = 0 \text{ w.p. } 50\% \\ b = 1 \text{ w.p. } 50\% \end{cases}$$

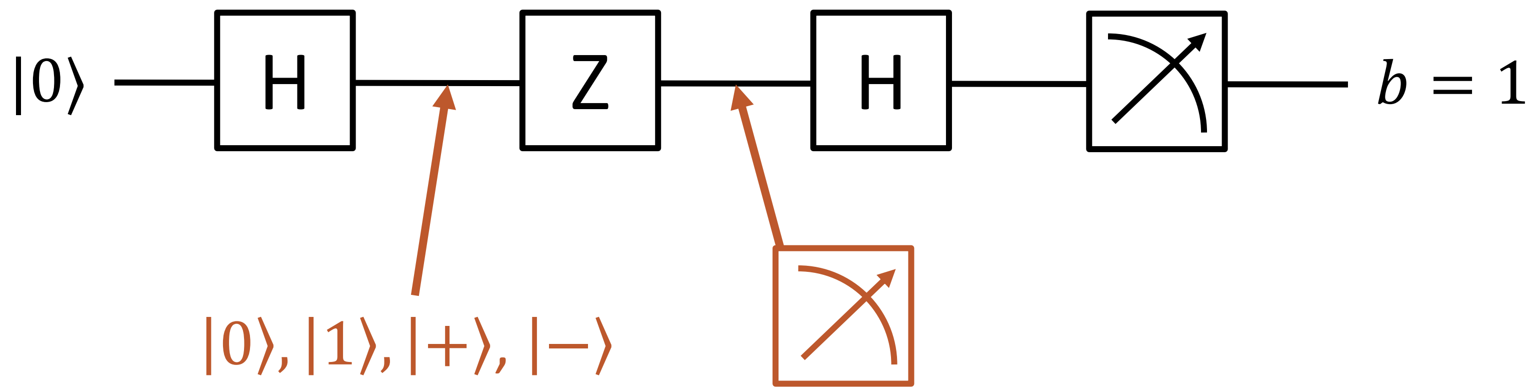


Really??

Repeat and collect statistics?

Run some tests?

Aside: benchmarking



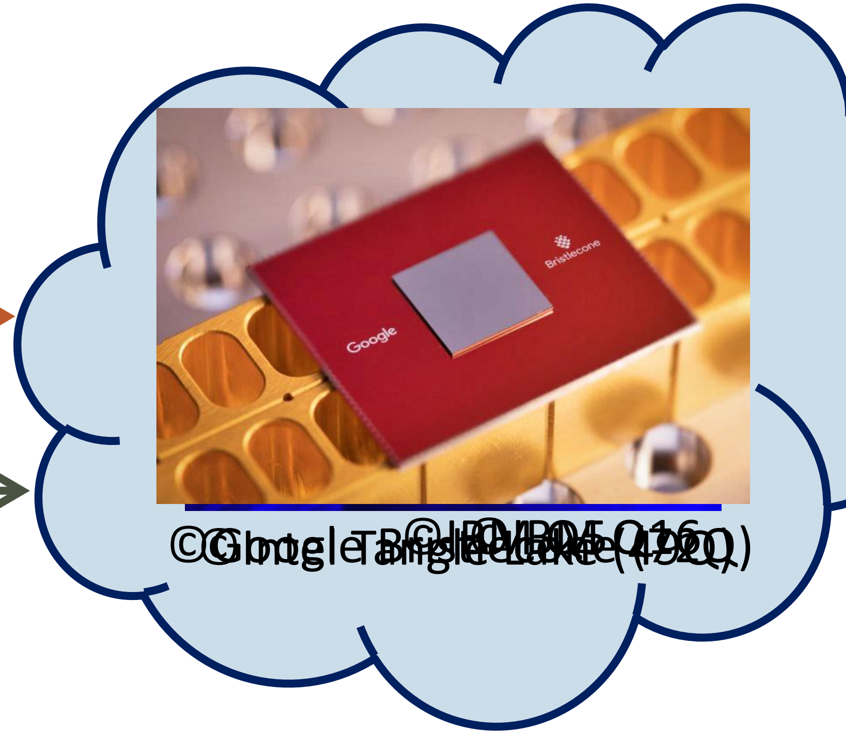
Sequentially test gate by injecting well-characterized states and collecting output statistics

- Requires access to inner workings of device
- Trusted state preparation and/or measurement
- Gates are not allowed to be "malicious", e.g. i.i.d. behavior is generally assumed
- Ineffective at large scales

Testing quantum mechanics at scale

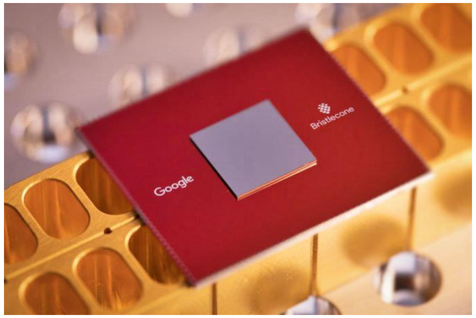
(q)

(c)



- Quantum mechanics untested at large scales
- Is there a limit to the exponential scaling of quantum devices?

Some other reasons to care



©Google Bristlecone

- Near-term demonstration of quantum advantage
  - Can verifiability be baked in current proposals?
- Cryptographic techniques
  - What modes of encryption allow transversal (homomorphic) computation?
  - Can they be combined with authentication?
- Models of computation & fault-tolerance
  - Do small nodes in a quantum network create fault-tolerance bottlenecks?
- Complexity theory
  - What is the expressive power of bounded-prover interactive proofs?
- Foundations
  - Are there analogues of the Bell inequalities without locality assumptions?

Prelude: Definitions

Semi-formal definition

A delegation protocol for quantum computations is:

A description of a (classical or quantum) polynomial-time **verifier,** that takes as input a **quantum circuit C** of size $|C| \leq n$, interacts with a **quantum prover**, and returns a pair $(flag, b)$ such that:

- *(Completeness)* There exists a (quantum, poly-time) prover P such that

  $$\Pr(flag = acc) \approx 1 \quad AND \quad \Pr(b = 1) \approx \Pr(\, C \; returns \; 1 \; on \; input \; |0^n\rangle \;)$$

- *(Soundness)* For any prover P^* such that $\Pr(flag = acc)$ is non-negligible,

  $$\Pr(\, b = 1 \,|flag = \mathrm{acc}) \quad \approx \quad \Pr(\, C \; returns \; 1 \; on \; input \; |0^n\rangle \;)$$

- *(Blindness)* For any prover P^*, $View_P(V_n(C) \leftrightarrow P^*)$ does not depend on C

Formal definition

"Stand-alone" definitions can fail! Example:

Protocol for testing if formula $\varphi = (x_1 \vee \overline{x_3} \vee x_5) \wedge (\cdots)$ *is satisfiable*

1. Prover sends assignment $x = (x_1, \dots, x_n)$
2. Verifier checks that x satisfies φ

This protocol is blind (prover learns nothing about φ) & verifiable

"Attack": Prover sends a uniformly random assignment

- Learns information about φ from verifier's accept/reject decision
- Protocol is not composable

Composable security: ideal-world/real-world paradigm

Parameters

Input size: n = number of qubits of circuit C

$|C|$ = number of gates

Completenes: Probability of accepting honest prover. This will always be ≈ 1

Soundness: Max. distinguishing ability between real-world/ideal-world.

Ideally, exponentially small in n.

Verifier complexity: Ideally, classical polynomial-time.

Limited quantum capability may be acceptable.

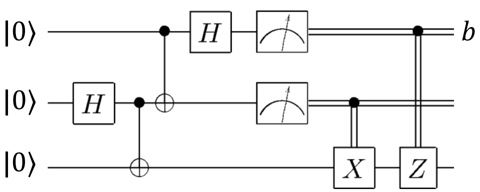
Prover complexity: Quantum polynomial-time. Ideally $\approx$ runtime(C).

Interaction: Minimize number of rounds + total communication

Overview of existing approaches

Models of computation

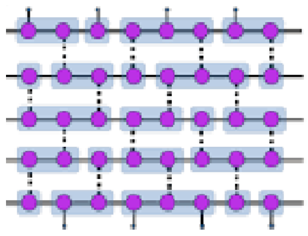
Circuit model



Input: circuit = sequence of gates acting on n qubits

Goal: determine value of output qubit, on input $|0\rangle$

Measurement-based



Input: adaptive sequence of single-qubit measurements

on resource state (e.g. “cluster state”)

Goal: determine value of output qubit

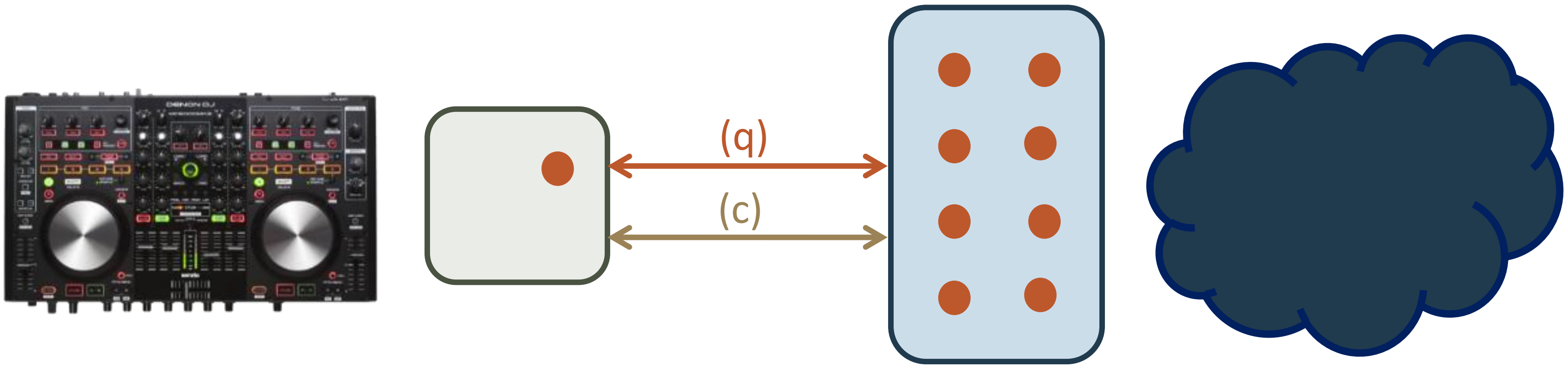
Hamiltonian model

$$H = H_{in} + H_{clock} + H_{prop} + H_{out}$$

Input: local Hamiltonian w. efficiently preparable ground state

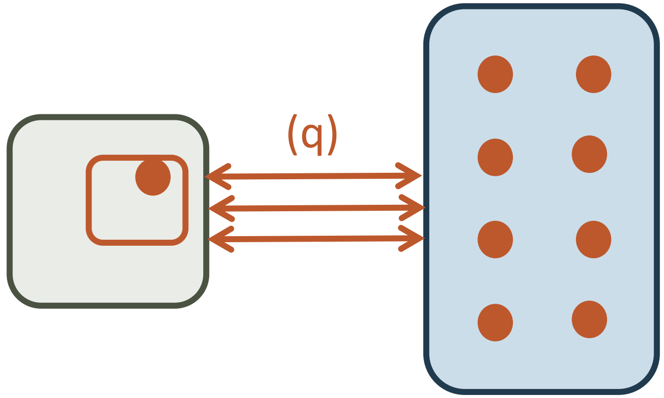
Goal: estimate ground state energy

Models for black-box verification



Challenge: Use minimal resources to verify complex quantum computation

Models for black-box verification

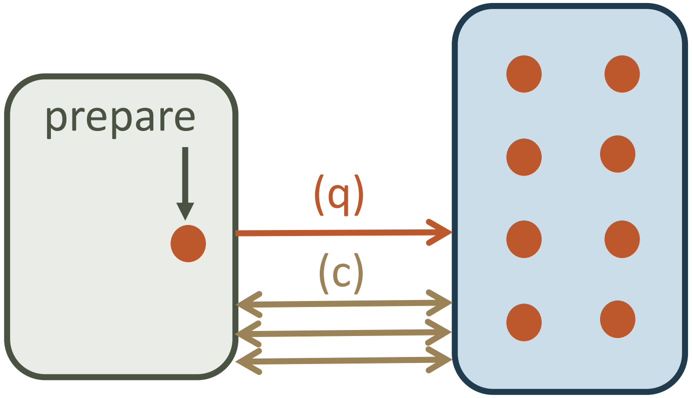


[Childs'05] Blind delegation

- Verifier has constant-size quantum computer and can only perform single-qubit Pauli gates
- Many-round quantum interaction
- Blind but not verifiable

Where are the qubits? *Honest-but-curious model*

Models for black-box verification



[Aharonov-Ben-Or-Eban'08, Aharonov-Ben-Or-Eban-Mahadev'18]

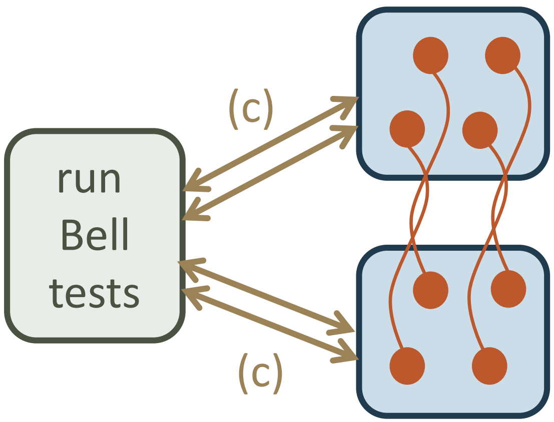
[Broadbent-Fitzsimons-Kashefi'09,Fitzsimons-Kashefi'16]

"Prepare-and-send" protocols:

- Verifier has ability to prepare & send O(1) qubits at a time
- Many-round classical interaction
  - [ABOE] *Circuit model*, uses authentication codes
  - [BFK] *Measurement-based model*, uses traps
- Both protocols are blind + verifiable

Where are the qubits? *The verifier authenticates them*

Models for black-box verification



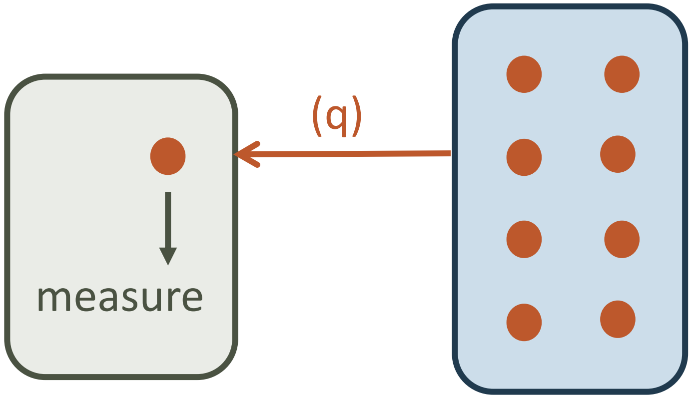
[Reichardt-Unger-Vazirani'12]

Two-prover protocols:

- Verifier is classical
- Many-round classical interaction with two isolated provers
- Verifier uses Bell tests to do state & process tomography
- Protocol is blind + verifiable

Where are the qubits? *Bell tests* $\rightarrow$ *EPR pairs* $\rightarrow$ *qubits*

Models for black-box verification



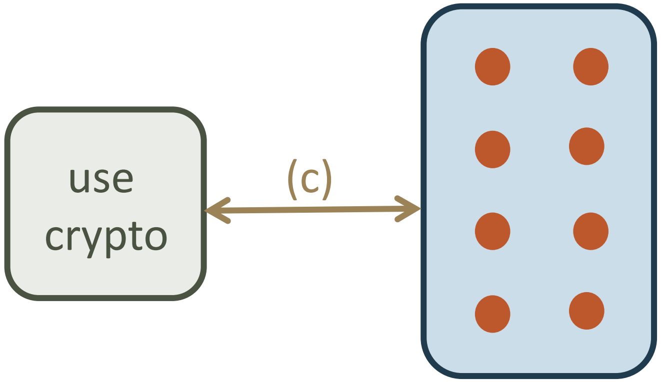
[Morimae-Fuji’13, Morimae-Fitzsimons’16]

“Receive & measure” protocols:

- Verifier has ability to receive & measure constant qubits
- [MNS’16] *Measurement-based model*, protocol is blind & verifiable
- [MF’16] *Hamiltonian model*, protocol is verifiable but not blind

Where are the qubits? *The verifier measures them*

Models for black-box verification

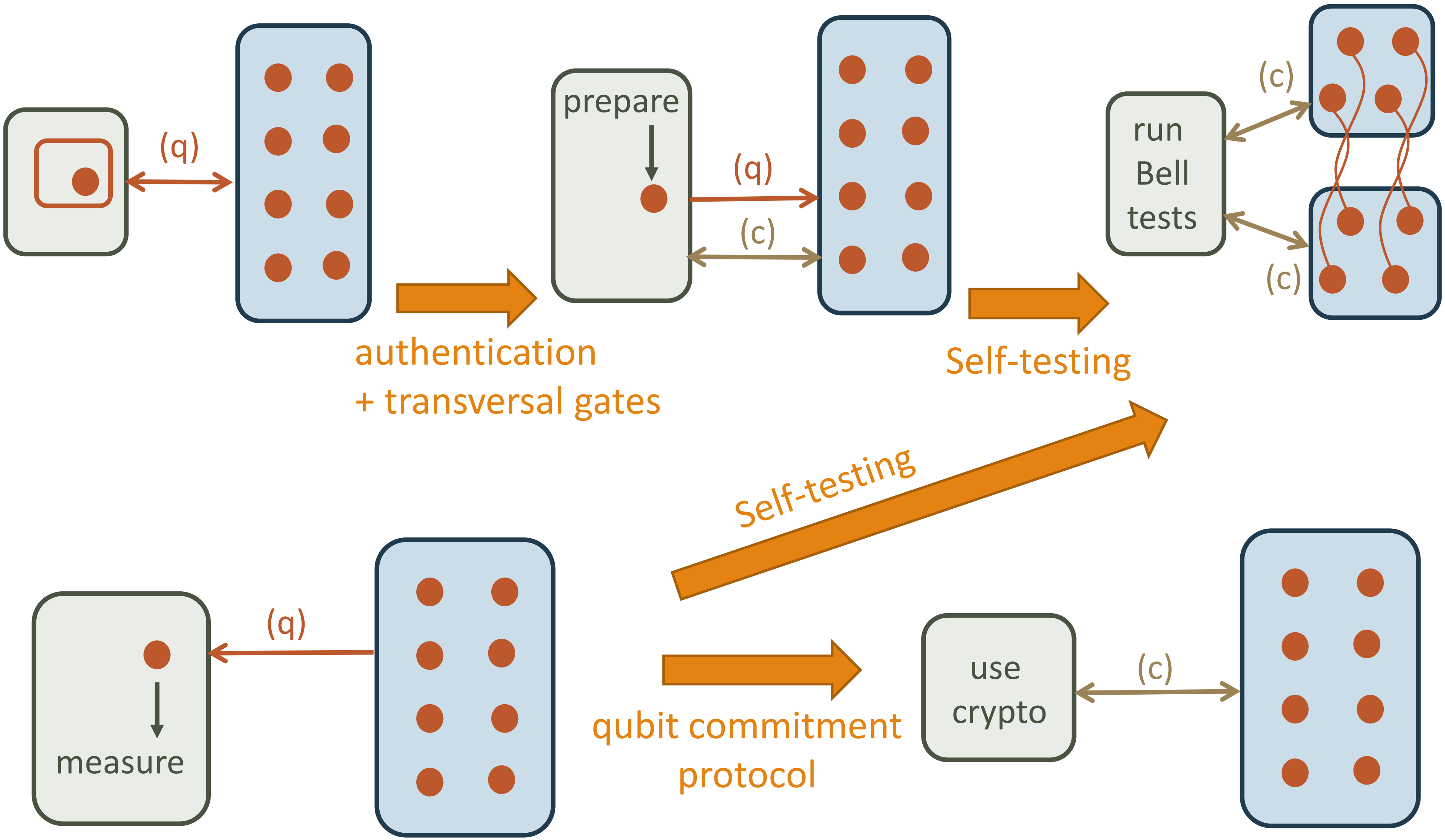


[Mahadev'18] "Commit & Reveal" protocols:

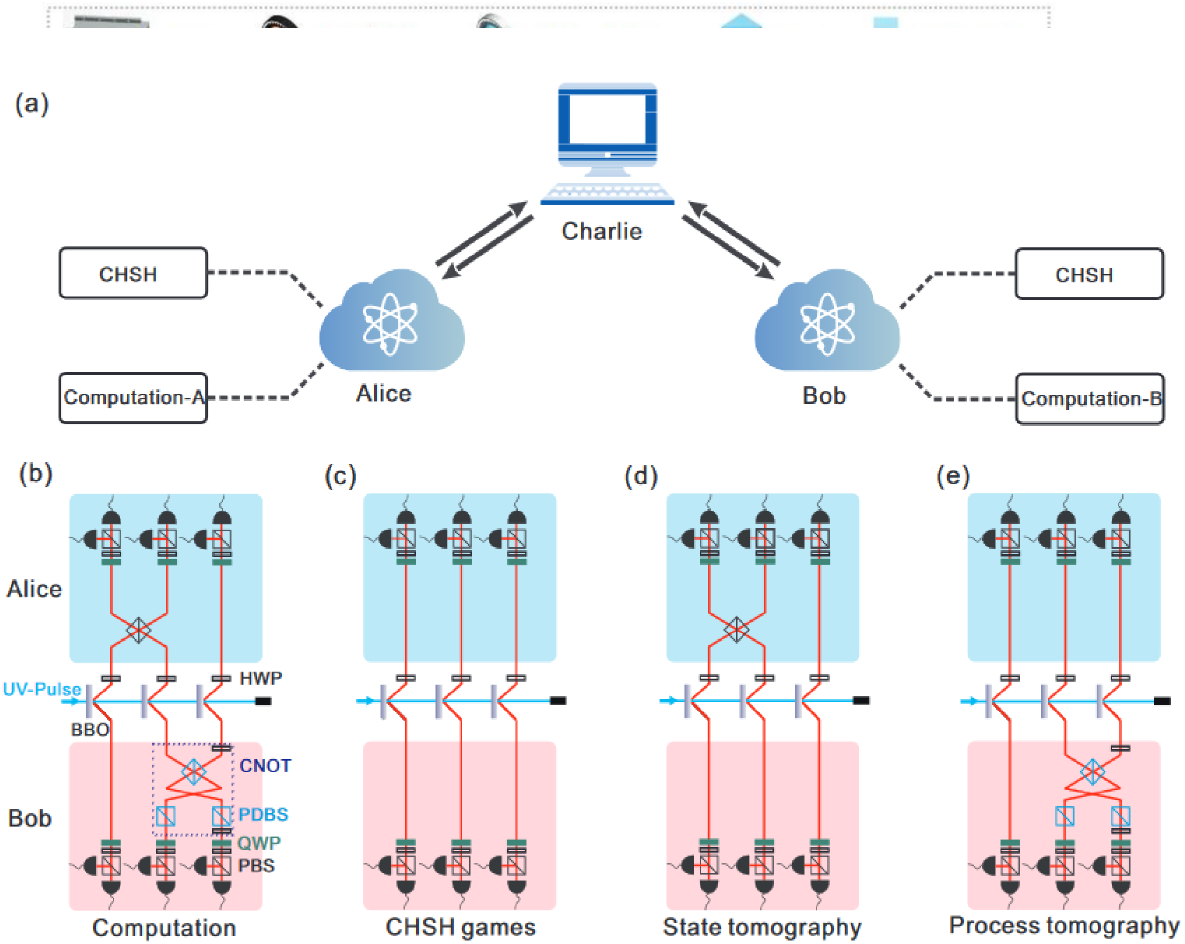
- Verifier is classical
- *Hamiltonian model*: protocol is not blind
- Verifiability assumes prover does not break post-quantum crypto

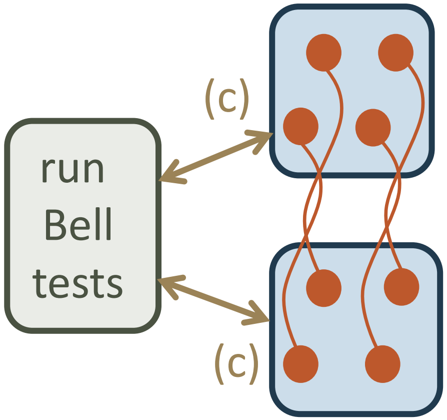
Where are the qubits? *Encoded using the crypto*

Building up


(q)
prepare
(q)
(c)
run
Bell
tests
(c)
(c)
authentication
+ transversal gates
Self-testing
Self-testing
measure
(q)
qubit commitment
protocol
use
crypto
(c)

Some experiments

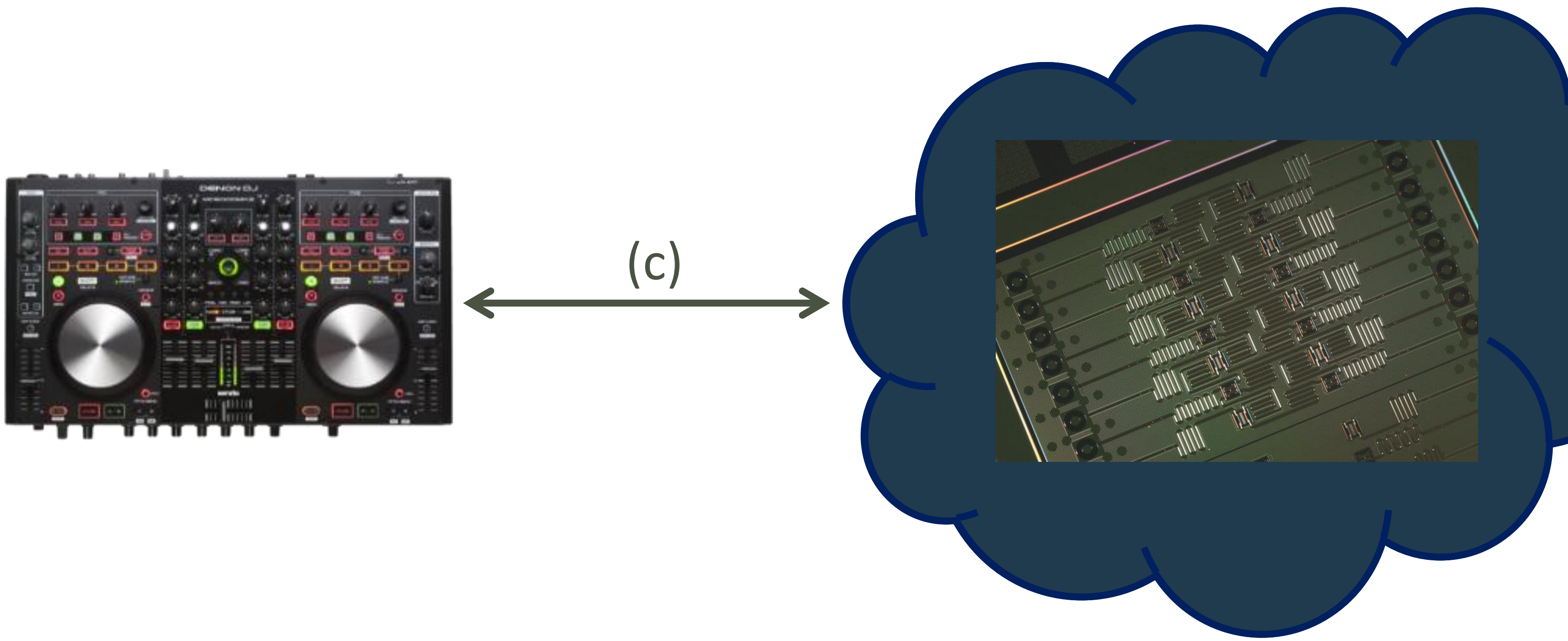




[Huang et al. 2017]
Thousands of Bell tests certify factorization of number 15

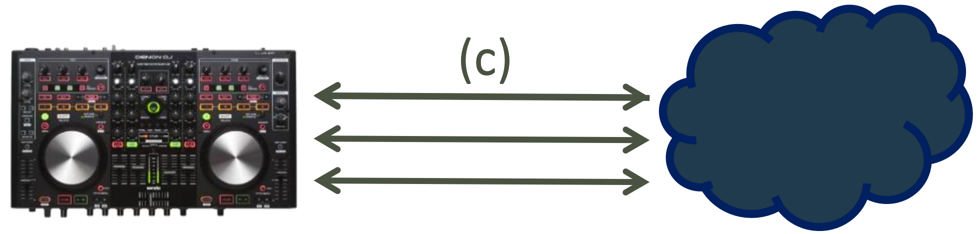
An open question

An open question

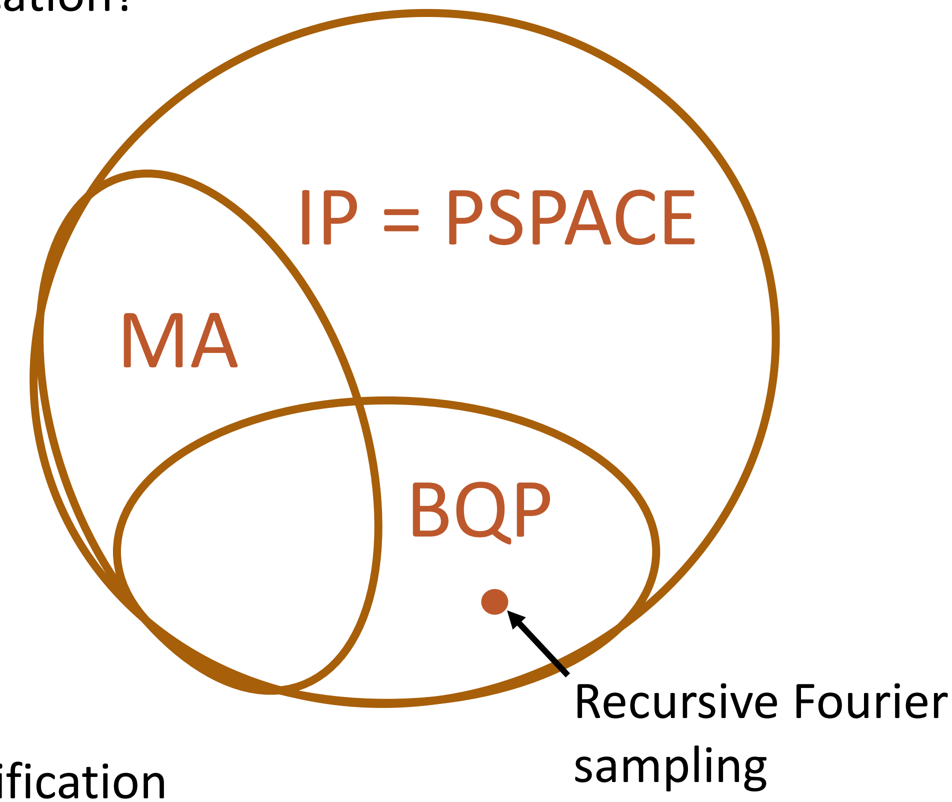


- Verifier is classical polynomial-time
- Communication channel is classical
- Verifier wants to determine $\Pr(C|0\rangle = 1)$

An open question



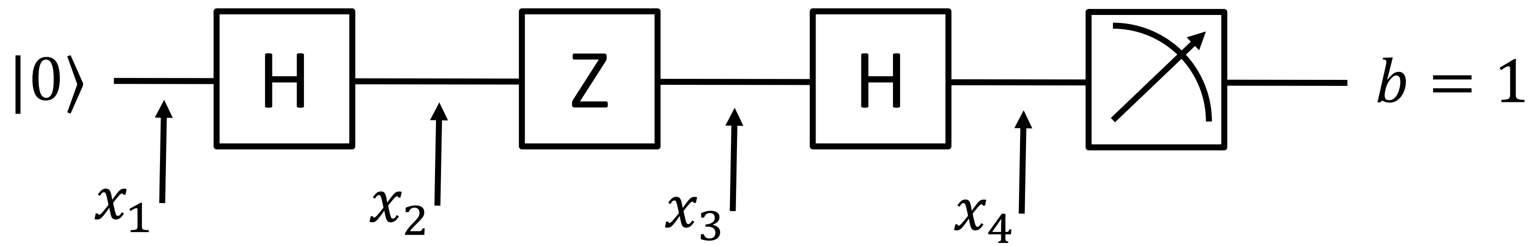
- Problems with efficient classical verification?
- MA = class of problems with efficient (probabilistic) verification
- Any problem in MA ∩ BQP has an efficiently verifiable solution
- Factoring, Graph Isomorphism



- IP = class of problems with efficient (probabilistic, interactive) verification
- IP Prover may not be efficient! Needs to compute exponentially large sums

Interactive proofs for BQP

- Feynman path integral: $\Pr(C|0\rangle = 1)$ is (square of) $\sum_{path=(x_1,\ldots,x_T)} amplitude(x_1, \ldots, x_T)$ summation over exponentially many paths



- Amplitude of individual path is easy to compute

$$amplitude(0,1,1,0) = 1 \cdot \frac{1}{\sqrt{2}} \cdot (-1) \cdot \frac{1}{\sqrt{2}} = -\frac{1}{2}$$

- Amplitude is multilinear polynomial in $x_1, \ldots, x_T$

Interactive proofs for BQP

- Given $P \in \mathbb{F}_q[X_1, \ldots, X_T]$ multilinear, compute $\sum_{x_1,\ldots,x_T \in \{0,1\}} P(x_1, \ldots, x_T)$

$S = \Sigma\ P(x_1, \ldots, x_T)$ $\longleftarrow$

$p_T(z) = \Sigma\ P(x_1, \ldots, x_{T-1}, z)$ $\longleftarrow$

$\Sigma_z p_T(z) = S$?

$\widetilde{z_T} \leftarrow_R \mathbb{F}_q$ $\xrightarrow{\widetilde{z_T}}$

$p_{T-1}(z) = \Sigma\ P(x_1, \ldots, z, \widetilde{z_T})$ $\longleftarrow$

$\Sigma_z p_{T-1}(z) = p_T(\widetilde{z_T})$?

$\widetilde{z_{T-1}} \leftarrow_R \mathbb{F}_q$ $\xrightarrow{\widetilde{z_{T-1}}}$

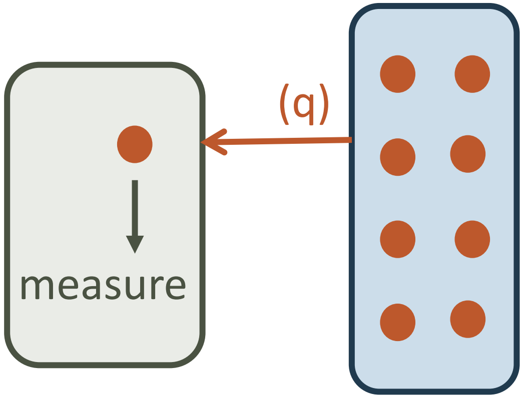
$\vdots$

$p_0 = P(\widetilde{z_1}, \ldots, \widetilde{z_T})$ $\longleftarrow$

$p_0 = P(\widetilde{z_1}, \ldots, \widetilde{z_T})$?

Receive & Measure Protocols

Receive & Measure protocols

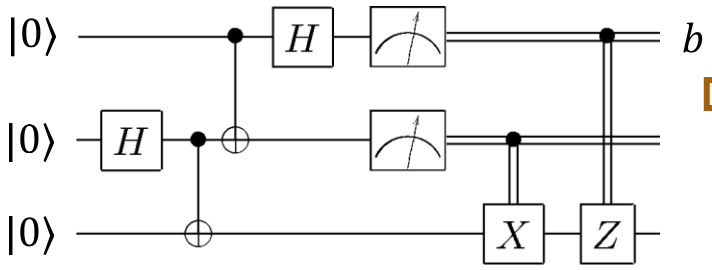


- <u>MBQC model</u>:
  - Prover prepares resource state (e.g. cluster state)
  - Verifier either (i) checks stabilizers of resource state

    (ii) implements computation
  - Only needs single-qubit measurements in small number of bases
- <u>Post-hoc model</u>:
  - Prover prepares history state of Kitaev Hamiltonian associated with circuit
  - Verifier measures randomly chosen term in Hamiltonian
  - Only needs single-qubit measurements in two bases, but protocol not blind

Circuit-to-Hamiltonian

[Kitaev'99]



$$H = H_{in} + H_{clock} + H_{prop} + H_{out}$$

$$\Pr(C|0\rangle = 1) \geq 2/3 \;\; \Longrightarrow \;\; \lambda_{min}(H) \leq a$$

$$\Pr(C|0\rangle = 1) \leq 1/3 \;\; \Longrightarrow \;\; \lambda_{min}(H) \geq a + \delta$$

- Hamiltonian can be expressed in "XX/ZZ form":

  H is weighted sum of local terms of the form $X_i X_j$ or $Z_i Z_j$
- Gap δ scales as $1/|C|^2$
- Complexity of preparing ground state of H scales as complexity of C

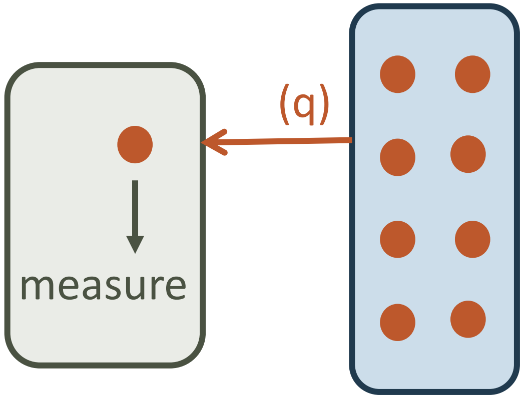
  (but may require higher depth)

Post-hoc verifiable delegation
[MF'16]

$$H = H_{in} + H_{clock} + H_{prop} + H_{out}$$

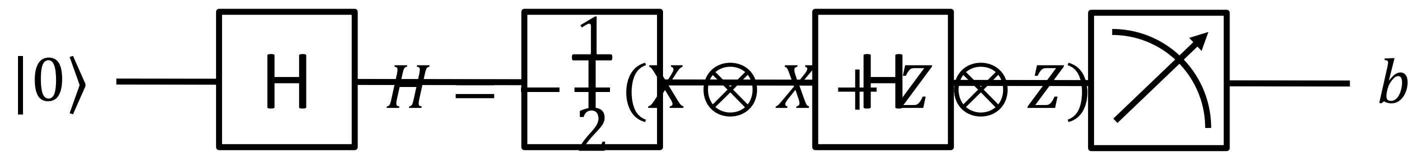
$$\Pr(C|0\rangle = 1) \geq 2/3 \implies \lambda_{min}(H) \leq a$$

$$\Pr(C|0\rangle = 1) \leq 1/3 \implies \lambda_{min}(H) \geq a + \delta$$



- Verifier computes H from C, sends to prover
- Prover prepares ground state of H
- Sends to verifier one qubit at a time
- Verifier secretly selects random local term $h_j = \mathrm{X}_{\mathrm{j}_1}\mathrm{X}_{\mathrm{j}_2}$ or $h_j = \mathrm{Z}_{\mathrm{j}_1}\mathrm{Z}_{\mathrm{j}_2}$
- Measures qubits j_1 and j_2 in required basis
- Repeat $1/\delta^2$ times to estimate energy

Running example



$H = \frac{1}{2}(X \otimes X + Z \otimes Z)$

$\xrightarrow{H}$

flip coin $W \in \{X, Z\}$

prepare $|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$

$\xleftarrow{\text{first qubit}}$

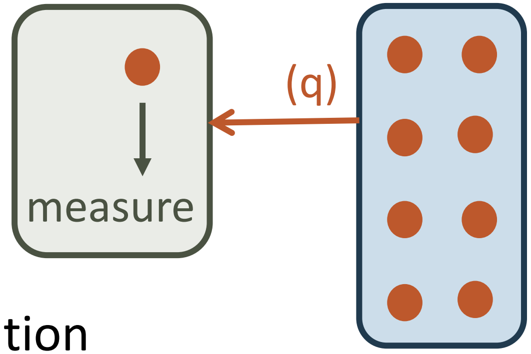
Measure in basis W $\rightarrow b_1$

$\xleftarrow{\text{second qubit}}$

Measure in basis W $\rightarrow b_2$

Check: $b_1 b_2 = +1$

Receive & Measure protocols: summary



- One-way quantum communication

- Hamiltonian model requires repetition for gap amplification

  MBQC model requires repetition for resource state testing

  Total communication at least $\sim |C|^3$

  Open: protocol with linear communication complexity

- Blind protocols only in MBQC model

- Protocols vulnerable to noise at the verifier

  [GHK'18] give fault-tolerant protocol in Hamiltonian model; not blind

  Open: receive & measure fault-tolerant blind delegation

Part II(c): Commit & Reveal

Models for black-box verification

- Verifier "delegates" X and Z measurements to server
- Hurdle: Certify that reported measurement outcomes are obtained from a single underlying n-qubit state
- Idea: Use cryptography to "commit" prover to fixed n-qubit state

Committing to a bit

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

$r \in_R \{0,1\}^n$

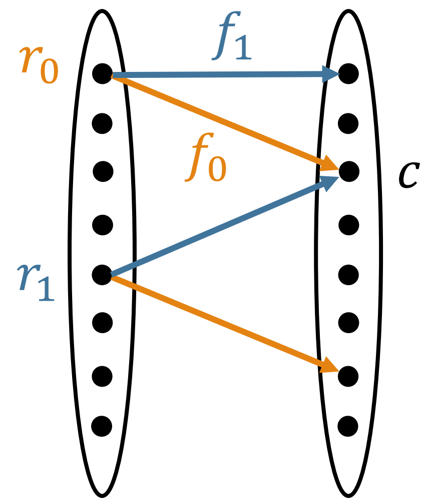
$c = com(b, r)$

$d = reveal(b, r)$

Return $(flag, b^*)$

- Hiding: c reveals no information about b $\quad c_{|b=0} \approx c_{|b=1}$
- Binding: For any efficient Bob, and any c such that $\Pr(flag = acc) \geq 0.01$, there is a b such that $\Pr(b^* = b |\, flag = acc) \geq 0.95$

Claw-free functions



$f_0, f_1 : \{0,1\}^n \to \{0,1\}^n$ a *claw-free* pair:

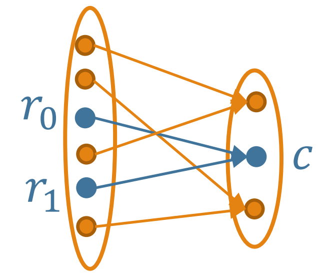
- Both f_0 and f_1 are bijections
- For every c in the range, there is a unique claw:
  a pair (r_0, r_1) such that $f_0(r_0) = f_1(r_1) = c$
- Claws are hard to find: no efficient procedure returns (r_0, r_1, c)

- Can construct based on "Learning with Errors" (LWE) problem
- f_0, f_1 are noisy multiplication by matrix A:

$$f_0(x) \approx A\,x + e, \quad f_1(x) \approx A(x - s) + e' \qquad \to \qquad r_1 \approx r_0 - s$$

Committing to a bit

$(f_0, f_1): \{0,1\}^n \to \{0,1\}^n$ a *claw-free* pair



$c = f_b(r)$

$d = (b, r)$

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

$r \in_R \{0,1\}^n$

Check $f_b(r) = c$

Return b

- <u>Perfectly hiding</u>: Any c has exactly one preimage under each function
- <u>Computationally binding</u>:

  If $\Pr(b^* = 0 | flag = acc) > 0.05$ and $\Pr(b^* = 1 | flag = acc) > 0.05$

  then run Bob 100 times on c to find a claw

Committing to a *qubit*

$c = com(|\psi\rangle, |R\rangle)$

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

$$|R\rangle = \frac{1}{\sqrt{2^n}} \sum_{r \in \{0,1\}^n} |r\rangle$$

$d_Z = Z\text{–}reveal(b, |R\rangle)$

Return $(flag, a_Z)$

$d_X = X\text{–}reveal(b, |R\rangle)$

Return $(flag, a_X)$

- Hiding: c reveals no information about $|\psi\rangle$
- Binding: For any efficient Bob and c such that $\Pr(flag = acc) \geq 0.01$ there is a ρ such that $a_Z \approx \mathrm{Tr}(Z\rho)$ and $a_X \approx \mathrm{Tr}(X\rho)$

Committing to a *qubit*

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

$$c = com(|\psi\rangle, |R\rangle)$$

$$|R\rangle = \frac{1}{\sqrt{2^n}} \sum_{r\in\{0,1\}^n} |r\rangle$$

$$|\psi\rangle \otimes |R\rangle \otimes |0^n\rangle \quad = \quad (\alpha|0\rangle + \beta|1\rangle) \otimes \frac{1}{\sqrt{2^n}} \sum_{r\in\{0,1\}^n} |r\rangle \otimes |0^n\rangle$$

$$\xrightarrow{\text{CTL-}f} \quad \frac{\alpha}{\sqrt{2^n}} \sum_{r\in\{0,1\}^n} |0\rangle|r\rangle|f_0(r)\rangle + \frac{\beta}{\sqrt{2^n}} \sum_{r\in\{0,1\}^n} |1\rangle|r\rangle|f_1(r)\rangle$$

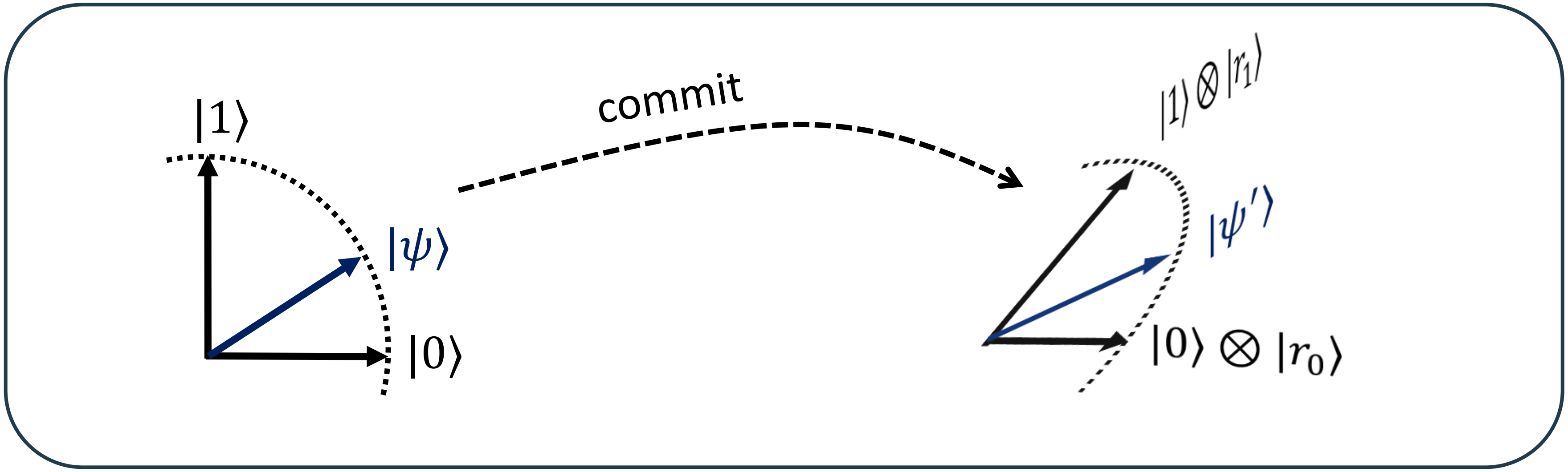
$$\xrightarrow{\text{meas. last register}} \quad (\alpha|0\rangle|r_0\rangle + \beta|1\rangle|r_1\rangle) \otimes |c\rangle$$

Committing to a *qubit*

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

$$c = com(|\psi\rangle, |R\rangle)$$

$$|R\rangle = \frac{1}{\sqrt{2^n}} \sum_{r\in\{0,1\}^n} |r\rangle$$



Committing to a *qubit*

$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$

$c = com(|\psi\rangle, |R\rangle)$

$$|R\rangle = \frac{1}{\sqrt{2^n}} \sum_{r\in\{0,1\}^n} |r\rangle$$

$d_Z = Z\text{–}reveal(b, |R\rangle)$

$$|\psi\rangle \otimes |R\rangle \otimes |0^n\rangle \quad = \quad (\alpha|0\rangle + \beta|1\rangle) \otimes \frac{1}{\sqrt{2^n}} \sum_{r\in\{0,1\}^n} |r\rangle \otimes |0^n\rangle$$

$$\xrightarrow{\text{CTL-}f} \quad \frac{\alpha}{\sqrt{2^n}} \sum_{r\in\{0,1\}^n} |0\rangle|r\rangle|f_0(r)\rangle + \frac{\beta}{\sqrt{2^n}} \sum_{r\in\{0,1\}^n} |1\rangle|r\rangle|f_1(r)\rangle$$

$$\xrightarrow{\text{meas. last register}} \quad (\alpha|0\rangle|r_0\rangle + \beta|1\rangle|r_1\rangle) \otimes |c\rangle$$

- Hiding: c reveals no information about $|\psi\rangle$ ✔
- *Z*-reveal: Bob measures in computational basis and returns $d_Z = (b, r_b)$

  Alice checks $f_b(r_b) = c$ and returns "decoded bit" $a_Z = b$

Committing to a *qubit*

$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$

$c = com(|\psi\rangle, |R\rangle)$

$d_X = \mathrm{X}{-}reveal(b, |R\rangle)$

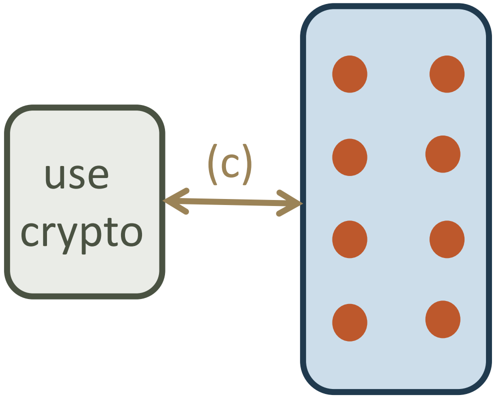
$|R\rangle = \frac{1}{\sqrt{2^n}} \sum_{r\in\{0,1\}^n} |r\rangle$

$$(\alpha|0\rangle|r_0\rangle + \beta|1\rangle|r_1\rangle) \xrightarrow{I \otimes H^{\otimes n}} \frac{1}{\sqrt{2^n}} \sum_{t\in\{0,1\}^n} (\alpha(-1)^{t\cdot r_0}|0\rangle + \beta(-1)^{t\cdot r_1}|1\rangle) \otimes |t\rangle$$

$$= \frac{1}{\sqrt{2^n}} \sum_{t\in\{0,1\}^n} (-1)^{t\cdot r_0} \; Z^{t\cdot r_0 \oplus t\cdot r_1} \; |\psi\rangle \otimes |t\rangle$$

- X-reveal: Bob measures in Hadamard basis and returns $d_X = (u, t)$
  Alice returns "decoded bit" $a_X = u \oplus (t \cdot r_0 \oplus t \cdot r_1)$

Commit & Reveal protocol
[Mahadev'18]



$$H = \sum_{(j_1,j_2)} \alpha_{j_1 j_2}(X_{j_1}X_{j_2} + Z_{j_1}Z_{j_2})$$

$$\Pr(C|0\rangle = 1) \geq 2/3 \implies \lambda_{min}(H) \leq a$$

$$\Pr(C|0\rangle = 1) \leq 1/3 \implies \lambda_{min}(H) \geq a + \delta$$

- Verifier computes H from C, sends to prover
- Prover prepares ground state of H

same as post-hoc protocol

- Prover individually commits to each qubit by sending $c_1, \ldots, c_n$
- Verifier secretly selects random local term $h_j = X_{j_1}X_{j_2}$ $(Z_{j_1}Z_{j_2})$
- Executes $X(Z)$-reveal phase with prover
- Records decoded outcomes $a_{X_{j_1}}a_{X_{j_2}}$ $(a_{Z_{j_1}}a_{Z_{j_2}})$
- Repeat $1/\delta^2$ times to estimate energy

Running example

$$H = -\frac{1}{2}(X \otimes X + Z \otimes Z)$$

H and f_0, f_1 and f'_0, f'_1

prepare

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

commitments c, c'

run commitment procedure:

$$\frac{1}{\sqrt{2}}(|0, r_0\rangle|0, r'_0\rangle + |1, r_1\rangle|1, r'_1\rangle)$$

flip coin $W \in \{X, Z\}$

X-reveal?

Measure X

$b, r_b, b', r'_{b'}$

u, t, u', t'

Check: $f_b(r_b) = c$

$f'_{b'}(r'_{b'}) = c'$

Set $a_X = u \oplus (t \cdot r_0 \oplus t \cdot r_1)$

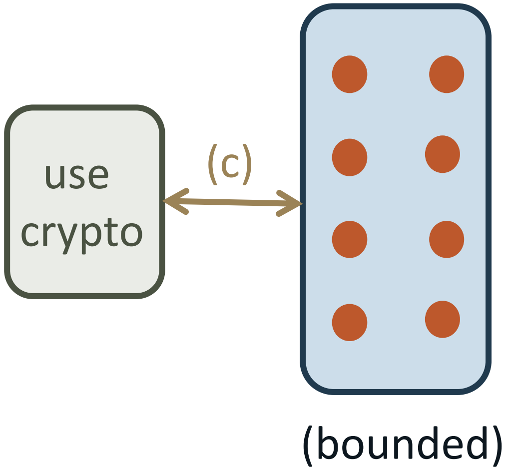
$a'_X = u' \oplus (t \cdot r'_0 \oplus t \cdot r'_1)$

Record b, b'

Record a_X, a'_X

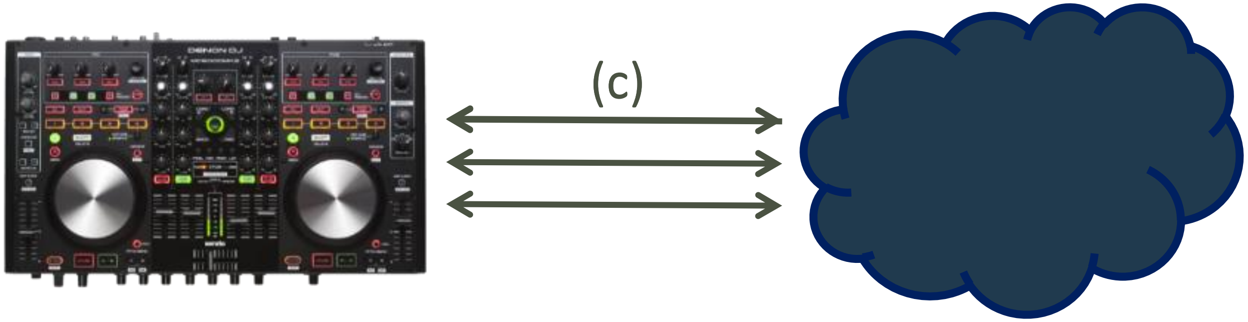
Repeat $1/\delta^2$ times to estimate energy

Commit & Reveal protocol: summary



- Hamiltonian model: protocol is not blind, but can be made blind by combining with quantum FHE
  Open: blind protocol in circuit or MBQC models?
- Complexity: cubic overhead due to Hamiltonian model
  Crypto overhead linear in security parameter
- Soundness guarantee: there *exists* a state that gives *computationally indistinguishable* measurement outcomes
  Open: computational assumption, information-theoretic guarantee?
- Claw-free function instantiated from learning with errors assumption (LWE)
  Open: more generic construction (e.g. quantum-secure OWF)?

Interactive proofs for BQP



- Any language in BQP has a classical-verifier interactive proof
- Prover needs to compute unphysical quantities
- Cannot be implemented using quantum computer
- [AG'17] give "quantum-inspired" variant of protocol
- Open: protocol with prover less powerful than PostBQP
- Challenge: allow prover to make statistical estimation errors

  while restricting capacity to cheat

Summary

Problem formulation

Ideal functionality for verifiable & blind delegation

circuit C →

← bit b

$$b = \begin{cases} C(|0\rangle) & \text{if } e \in \{0,1\} \\ err & \text{if } e = 2 \end{cases}$$

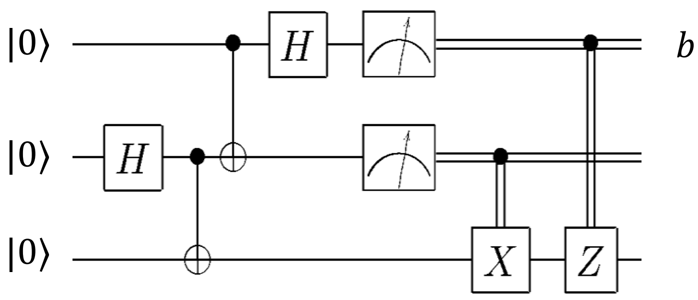
size T →

← trit e

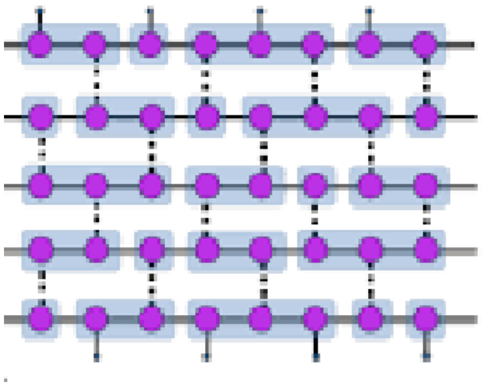
A protocol is verifiable & blind if no malicious party interacting with the honest party can distinguish from an interaction with the ideal functionality

Models of computation

Circuit model



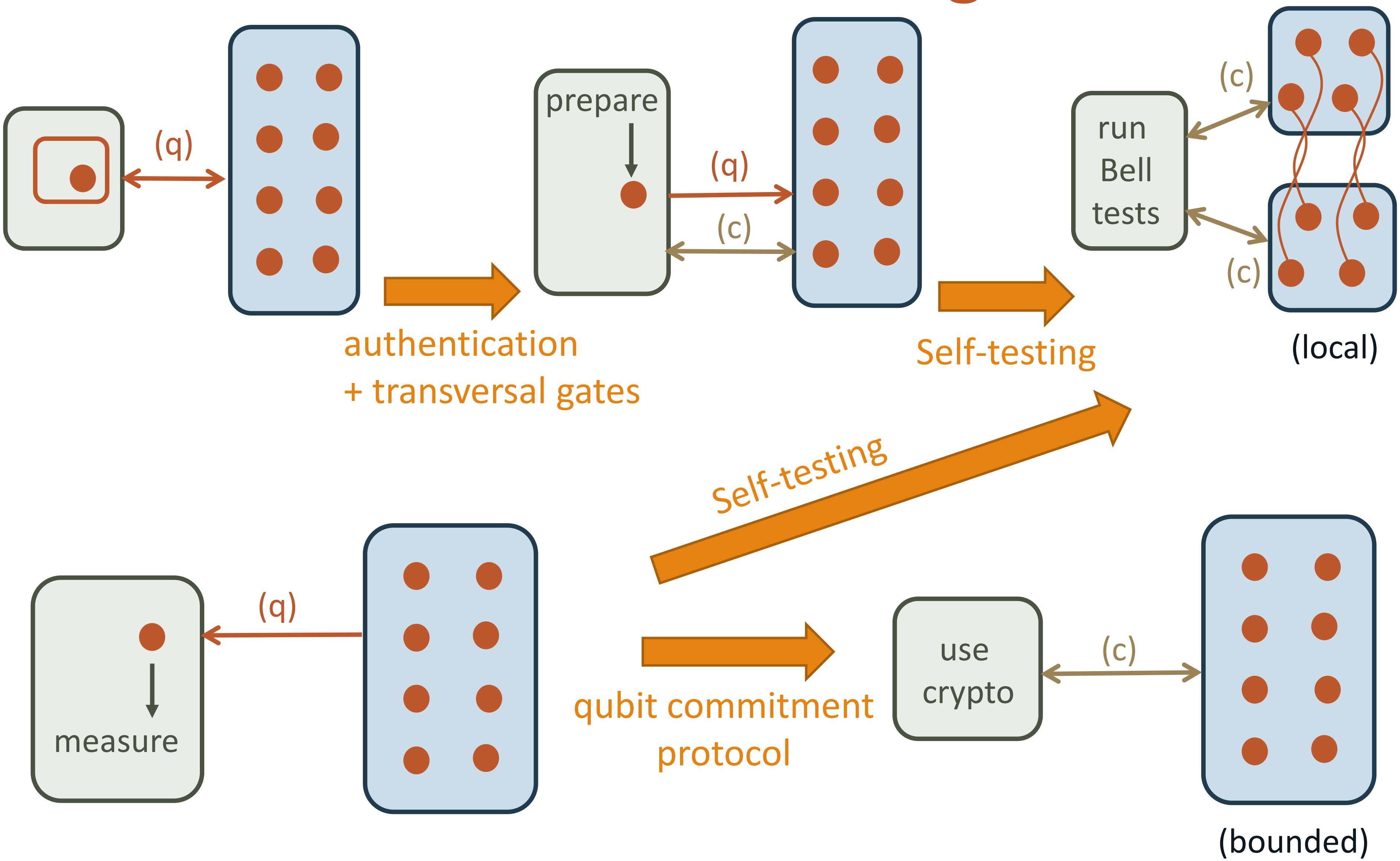
Measurement-based model



Hamiltonian model

$$H = H_{in} + H_{clock} + H_{prop} + H_{out}$$

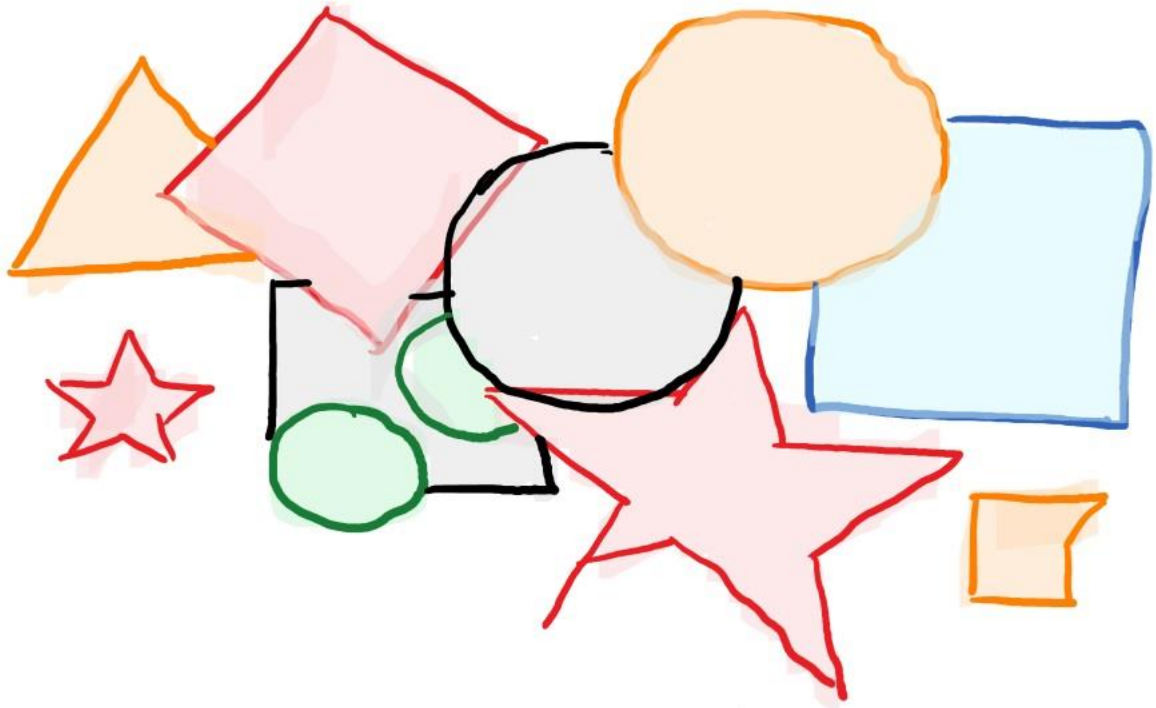
Protocols for verifiable delegation



Complexity considerations

Input: Circuit C, T gates, n qubits. eps: distance from ideal functionality

Protocol	Computation model	Verifier	Communication
Childs'05	Circuit	O(1)	O(T)
ABOE'08	Circuit	O(log 1/eps)	O(T log(1/eps))
BFK'09	MBQC	O(1)	O(T log(1/eps))
MF'13	MBQC	O(1)	O(T/eps^2)
MF'16	Hamiltonian	O(1)	O(T^3 log(1/eps))
CGJV'18	Circuit	classical	O(T/eps^c)
Mahadev'18	Hamiltonian	classical	O(T^3 log(1/eps)log(1/lambda))



Thank you

SLIDES:
HTTP://USERS.CMS.CALTECH.EDU/~VIDICK/VERIFICATION.{PPSX,PDF}

References

[ADSS'17] Alagic et al. "Quantum Fully Homomorphic Encryption With Verification." *arXiv:1708.09156*

[AG'17] Aharonov and Green. "A Quantum inspired proof of P^{\# P}\subseteq IP." *arXiv:1710.09078*

[BKB+'12] Barz et al. "Demonstration of blind quantum computing." *Science* 335.6066 (2012): 303-308.

[Broadbent'15] Broadbent. "How to verify a quantum computation." *arXiv:1509.09180*

[Childs'05] Childs. "Secure assisted quantum computation." *arXiv preprint quant-ph/0111046*

[DFPR'13] Dunjko et al. "Composable security of delegated quantum computation." *arXiv:1301.3662*

[GRB+'16] Greganti et al. "Demonstration of measurement-only blind quantum computing." *NJP* 18.1 (2016): 013020

[Grilo'18] Grilo. "Relativistic verifiable delegation of quantum computation." *arXiv:1711.09585*

[GHK'18] Gheorghiu et al. "A simple protocol for fault tolerant verification of quantum computation." *arXiv:1804.06105*

[GKK'17] Gheorghiu et al. "Verification of quantum computation: An overview of existing approaches." *arXiv:1709.06984*

[HZM+'17]Huang et al. "Experimental blind quantum computing for a classical client." *PRL* 119.5 (2017): 050503.

[Mahadev'18] Mahadev. "Classical verification of quantum computations." *arXiv:1804.01082*

[MNS'16] Morimae and Fujii. "Blind quantum computation protocol in which Alice only makes measurements." *arXiv:1201.3966*

[MF'16] Morimae, Nagaj and Schuch. "Quantum proofs can be verified using only single-qubit measurements." *arXiv:1510.06789*

[MY'05] Mayers and Yao. "Self testing quantum apparatus." *quant-ph/0307205*

[MYS'12] McKague et al. "Robust Self Testing of the Singlet." *arXiv:1203.2976*.

[RUV'12] Reichardt et al. "A classical leash for a quantum system." *arXiv:1209.0448*.

[WS'88] Summers and Werner. "Maximal violation of Bell's inequalities for algebras of observables in tangent spacetime regions." *Annales de l'Institut Henri Poincare Physique Theorique*. Vol. 49. No. 2. 1988