Overview of Mahadev's Protocol for Classical Verification of Quantum Computation

PhD Interview - Technical Talk

Alexander Kulpe

2023-08-29

MAX PLANCK INSTITUTE FOR SECURITY AND PRIVACY

RUHR UNIVERSITÄT BOCHUM

Table of Contents

Table of Contents

Quantum Computations

Quantum computation give an advantage over classical computation in

- Simulating Quantum Systems
- Optimization
- Factorization / Discrete Logarithms
- ...

Quantum Computations

Quantum computation give an advantage over classical computation in

- Simulating Quantum Systems
- Optimization
- Factorization / Discrete Logarithms
- ...

Verifying quantum computations is important:

- Errors in computations / Verifying integrity
- Validation of Quantum Algorithms
- Verifying Quantum Supremacy
- Building Trust
- ...

Quantum Computations

Quantum computation give an advantage over classical computation in

- Simulating Quantum Systems
- Optimization
- Factorization / Discrete Logarithms
- ...

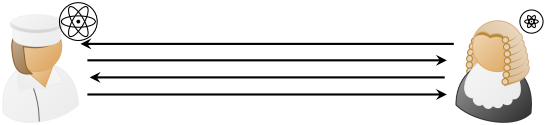
Verifying quantum computations is important:

- Errors in computations / Verifying integrity
- Validation of Quantum Algorithms
- Verifying Quantum Supremacy
- Building Trust
- ...

Verifying quantum computations classically is not feasible
$\Rightarrow$ Mahadev's Protocol: Use cryptography and interact classically with the quantum computer

Short History Lesson

- 2004: Question whether a classical computer can verify the result of a quantum computation through interaction is raised.
- BQP $\subseteq$ PSPACE $=$ IP, but powerful prover
- What if the Prover has to be efficient?
- Approach 1: Verifier has access to small quantum computer (error-correcting codes)



- Approach 2: Play multiple provers against each other (CHSH)
- Can we verify by only interacting with one prover without small quantum computer?

Table of Contents

Verifying quantum computation with trusted measurement device

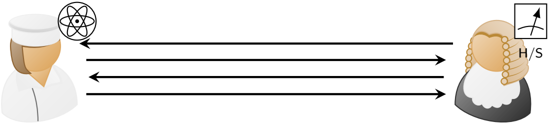
- [KSV02]: k-LOCAL HAMILTONIAN is QMA-complete (quantum analogue of NP). An eigenstate with sufficiently low energy is witness.

Verifying quantum computation with trusted measurement device

- [KSV02]: k-LOCAL HAMILTONIAN is QMA-complete (quantum analogue of NP). An eigenstate with sufficiently low energy is witness.
- [BL08]: Energy can be estimated by standard/Hadamard basis measurements.

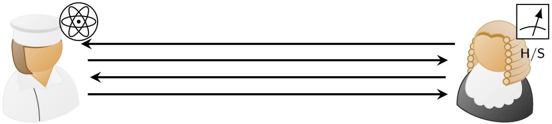
Verifying quantum computation with trusted measurement device

- [KSV02]: k-LOCAL HAMILTONIAN is QMA-complete (quantum analogue of NP). An eigenstate with sufficiently low energy is witness.
- [BL08]: Energy can be estimated by standard/Hadamard basis measurements.
- [FHM18]: Protocol with trusted measurement device:
  1. Verifier reduces x to local Hamiltonian H_x
  2. Verifier requests state from prover
  3. Verifier checks if received state has low energy with respect to H_x. If energy is low, Verifier accepts.



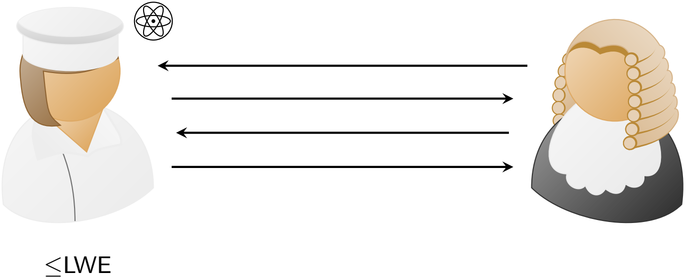
Verifying quantum computation with trusted measurement device

- [KSV02]: k-LOCAL HAMILTONIAN is QMA-complete (quantum analogue of NP). An eigenstate with sufficiently low energy is witness.
- [BL08]: Energy can be estimated by standard/Hadamard basis measurements.
- [FHM18]: Protocol with trusted measurement device:
  1. Verifier reduces x to local Hamiltonian H_x
  2. Verifier requests state from prover
  3. Verifier checks if received state has low energy with respect to H_x. If energy is low, Verifier accepts.



- What if we don't have access to trusted measurement device?

Mahadev's Protocol - Overview



- Measurement protocol: Classical verifier (BPP) using q. prover (BQP) as trusted measurement device
- Forces Prover to:
  - construct n qubit state of her choice
  - measure each qubit in Hadamard or Standard basis
  - report measurement result to verifier
- Soundness enforced based on LWE assumption: If verifier accepts, there exists a quantum state underlying the measurement result that is independent of the verifier's measurement choice

Commitment Phase

Definition (TCF+)

A function family $\mathcal{F} = \{f_{i,0}, f_{i,1} : \mathcal{X} \to \mathcal{D}\}$ is called TCF+ if

- there exists ppt $\mathsf{Gen}_{\mathcal{F}}$: $(i, \mathsf{td}_i) \leftarrow \mathsf{Gen}_{\mathcal{F}}(1^\lambda)$
- $f_{i,0}, f_{i,1}$ injective with same image
- there exists ppt Inv that given $i, \mathsf{td}_i, y \in \mathcal{D}$, finds both preimages: $(x_0, x_1) \leftarrow \mathsf{Inv}(i, \mathsf{td}_i, y)$
- adaptive Hardcore bit: $\forall d \neq 0 \forall$ claws (x_0, x_1) is is hard to compute both $d \cdot (x_0 \oplus x_1)$ and a preimage x_0 or x_1; $\exists\ d$ s.t. $\forall$ claws (x_0, x_1) the bit $d \cdot (x_0 \oplus x_1)$ is the same and indistinguishable from uniform

approximate TCF+ can be built from LWE

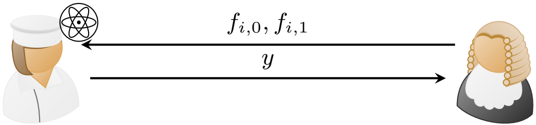
Commitment Phase

Definition (TCF+)

A function family $\mathcal{F} = \{f_{i,0}, f_{i,1} : \mathcal{X} \to \mathcal{D}\}$ is called TCF+ if

- there exists ppt $\mathsf{Gen}_{\mathcal{F}}$: $(i, \mathsf{td}_i) \leftarrow \mathsf{Gen}_{\mathcal{F}}(1^\lambda)$
- $f_{i,0}, f_{i,1}$ injective with same image
- there exists ppt Inv that given $i, \mathsf{td}_i, y \in \mathcal{D}$, finds both preimages: $(x_0, x_1) \leftarrow \mathsf{Inv}(i, \mathsf{td}_i, y)$
- adaptive Hardcore bit: $\forall d \neq 0 \forall$ claws (x_0, x_1) is is hard to compute both $d \cdot (x_0 \oplus x_1)$ and a preimage x_0 or x_1; $\exists\ d$ s.t. $\forall$ claws (x_0, x_1) the bit $d \cdot (x_0 \oplus x_1)$ is the same and indistinguishable from uniform

approximate TCF+ can be built from LWE



$\leq$LWE

- The Verifier samples TCF+ functions and sends $f_{i,0}, f_{i,1}$ to the Prover.
- Prover entangles a quantum state of his choice with a claw $y = f_{i,0}(x_0) = f_{i,1}(x_1)$ and sends y to the verifier

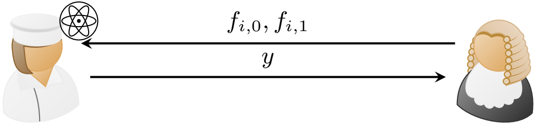
Commitment Phase

Definition (TCF+)

A function family $\mathcal{F} = \{f_{i,0}, f_{i,1} : \mathcal{X} \to \mathcal{D}\}$ is called TCF+ if

- there exists ppt $\text{Gen}_{\mathcal{F}}$: $(i, \text{td}_i) \leftarrow \text{Gen}_{\mathcal{F}}(1^\lambda)$
- $f_{i,0}, f_{i,1}$ injective with same image
- there exists ppt Inv that given $i, \text{td}_i, y \in \mathcal{D}$, finds both preimages: $(x_0, x_1) \leftarrow \text{Inv}(i, \text{td}_i, y)$
- adaptive Hardcore bit: $\forall d \neq 0 \forall$ claws (x_0, x_1) is is hard to compute both $d \cdot (x_0 \oplus x_1)$ and a preimage x_0 or x_1; $\exists\ d$ s.t. $\forall$ claws (x_0, x_1) the bit $d \cdot (x_0 \oplus x_1)$ is the same and indistinguishable from uniform

approximate TCF+ can be built from LWE



$\leq$LWE

- The Verifier samples TCF+ functions and sends $f_{i,0}, f_{i,1}$ to the Prover.
- Prover entangles a quantum state of his choice with a claw $y = f_{i,0}(x_0) = f_{i,1}(x_1)$ and sends y to the verifier

$$|\psi\rangle = \sum_{b \in \{0,1\}} \alpha_b |b\rangle \to \sum_{x \in \mathcal{X}} \sum_{b \in \{0,1\}} \alpha_b |b\rangle |x\rangle |f_{i,b}(x)\rangle \xrightarrow{f_{i,b}(x) = y} \sum_{b \in \{0,1\}} \alpha_b |b\rangle |x_b\rangle = \text{Enc}(|\psi\rangle)$$

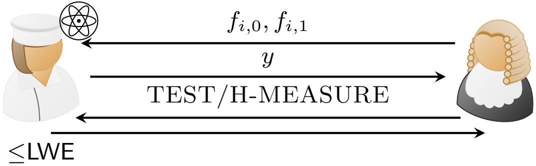
Challenge

TEST

- Verifier requests preimage (b, x_b) of y

$$\mathsf{Enc}(|\psi\rangle) = \sum_{b\in\{0,1\}} \alpha_b |b\rangle |x_b\rangle$$

$\Rightarrow$ Prover can measure in standard basis and sends result to the prover



Challenge

TEST

- Verifier requests preimage (b, x_b) of y

$$\mathsf{Enc}(|\psi\rangle) = \sum_{b\in\{0,1\}} \alpha_b |b\rangle |x_b\rangle$$

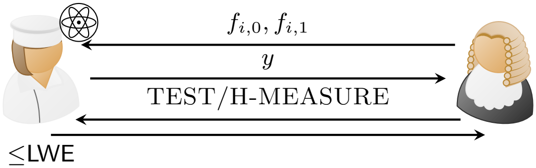
$\Rightarrow$ Prover can measure in standard basis and sends result to the prover

H-MEASURE

- Prover applies H to entire encoded state, measures second register and sends result r to the verifier

$$\mathsf{Enc}(|\psi\rangle) \xrightarrow{H} X^{d\cdot(x_0 \oplus x_1)} H |\psi\rangle$$

- Verifier decodes measurement by XORing $d \cdot (x_0 \oplus x_1)$ to r



Standard Basis Measurement

Definition (TIF+)

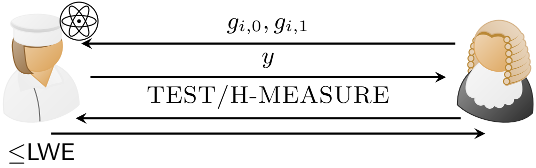
A function family $\mathcal{G} = \{g_{i,0}, g_{i,1} : \mathcal{X} \to \mathcal{D}\}$ is called TIF+ if

- there exists ppt $\mathsf{Gen}_{\mathcal{G}}$: $(i, \mathsf{td}_i) \leftarrow \mathsf{Gen}_{\mathcal{G}}^{(} 1^\lambda)$
- $g_{i,0}, g_{i,1}$ injective with distinct images
- there exists ppt $\mathsf{Inv}_{\mathcal{G}}$ that given, $i, \mathsf{td}_i, y \in \mathcal{D}$ finds preimage $x \leftarrow \mathsf{Inv}_{\mathcal{G}}(i, \mathsf{td}_i, y)$
- $(f_{i,0}, f_{i,1})$ computationally indistinguishable from $(g_{i,0}, g_{i,1})$

This acts as standard basis measurement:

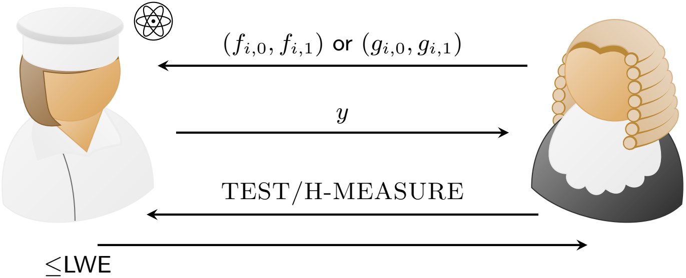
$$|\psi\rangle = \sum_{b\in\{0,1\}} \alpha_b |b\rangle \to \sum_{x\in\mathcal{X}} \sum_{b\in\{0,1\}} \alpha_b |b\rangle |x\rangle |g_{i,b}(x)\rangle$$

Given $y = g_{i,b}(x)$ the Verifier can reconstruct measurement result b using trapdoor



Verifier uses y to recover measurement result; ignores Hadamard measurement result

Protocol - Overview



Verifier chooses basis:

- Hadamard: send TCF+ $(f_{i,0}, f_{i,1})$
- Standard: send TIF+ $(g_{i,0}, g_{i,1})$

Verifier either:

- tests state structure or
- request measurement result

$\Rightarrow$ Apply this protocol for every qubit in parallel

Table of Contents

Completeness

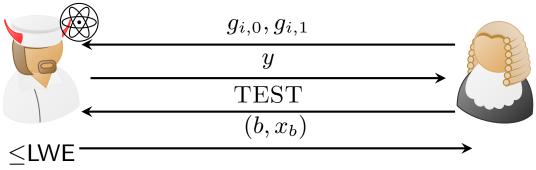
- Reduce problem to k-LOCAL-HAMILTONIAN
- Verifier chooses measurement basis
- Prover commits to ground state
- Prover measures honestly and sends measurement result
- Verifier can deduce that the commited state has low enough energy

Soundness

If verifier accepts, there exists a quantum state underlying the measurement result that is independent of the verifier's measurement choice

Soundness

If verifier accepts, there exists a quantum state underlying the measurement result that is independent of the verifier's measurement choice

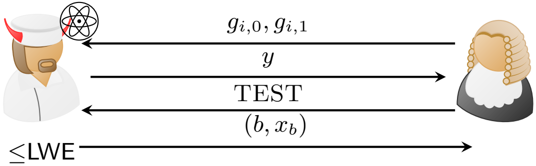


Definition (Pauli Twirl (Informal))

- Conjugation of unitary U by random Pauli
- $(X^x Z^z)^\dagger U (X^x Z^z)$
- averaging over random Paulis $\Rightarrow$ effect of Pauli

Soundness

If verifier accepts, there exists a quantum state underlying the measurement result that is independent of the verifier's measurement choice



- Prover's state must have been of the form:

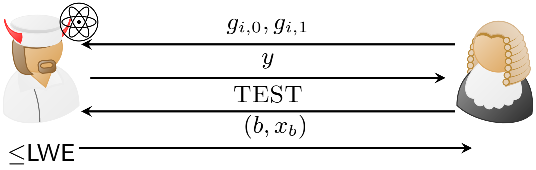
$$\sum_{b \in \{0,1\}} \alpha_b |b\rangle |x_b\rangle |\psi_{b,x_b}\rangle \text{ or } |b\rangle |x_b\rangle |\psi_{b,x_b}\rangle$$

Definition (Pauli Twirl (Informal))

- Conjugation of unitary U by random Pauli
- $(X^x Z^z)^\dagger U (X^x Z^z)$
- averaging over random Paulis $\Rightarrow$ effect of Pauli

Soundness

If verifier accepts, there exists a quantum state underlying the measurement result that is independent of the verifier's measurement choice



Definition (Pauli Twirl (Informal))

- Conjugation of unitary U by random Pauli
- $(X^x Z^z)^\dagger U (X^x Z^z)$
- averaging over random Paulis ⇒ effect of Pauli

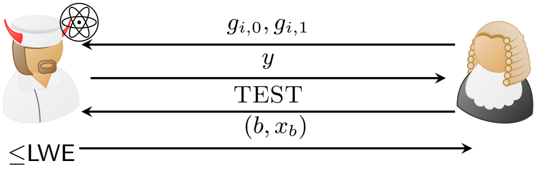
- Prover's state must have been of the form:

$$\sum_{b\in\{0,1\}} \alpha_b |b\rangle |x_b\rangle |\psi_{b,x_b}\rangle \text{ or } |b\rangle |x_b\rangle |\psi_{b,x_b}\rangle$$

- Let U be the deviation from the protocol
- Verifier's decoding is $d \cdot (x_0 \oplus x_1)$
- Part of U acting on first register computationally randomized by initial state and Verifier's decoding

Soundness

If verifier accepts, there exists a quantum state underlying the measurement result that is independent of the verifier's measurement choice



Definition (Pauli Twirl (Informal))

- Conjugation of unitary U by random Pauli
- $(X^x Z^z)^\dagger U (X^x Z^z)$
- averaging over random Paulis $\Rightarrow$ effect of Pauli

- Prover's state must have been of the form:

$$\sum_{b\in\{0,1\}} \alpha_b |b\rangle |x_b\rangle |\psi_{b,x_b}\rangle \text{ or } |b\rangle |x_b\rangle |\psi_{b,x_b}\rangle$$

- Let U be the deviation from the protocol
- Verifier's decoding is $d \cdot (x_0 \oplus x_1)$
- Part of U acting on first register computationally randomized by initial state and Verifier's decoding
- $d \cdot (x_0 \oplus x_1)$ computationally indistinguishable from uniform $\Rightarrow$ Use Pauli Twirl and U is simplified to Pauli
- U commutes with standard basis measurement $\Rightarrow$ U could have been applied before the commitment $\Rightarrow$ measurement distribution equivalent to honest prover with commited state $|\psi'\rangle = U|\psi\rangle$

Table of Contents

Further Work

- ↯ Relies on hardcore-bit properties
- ↯ Polynomially many repetitions needed

Further Work

- $\lightning$ Relies on hardcore-bit properties
- $\lightning$ Polynomially many repetitions needed

[Ala+20] Non-interactive classical verification of quantum computation

- Make first message instance-independent in offline-step
- Use a parallel repetition theorem to run $3\text{poly}(\lambda)$ steps in 3 steps
- Fiat-Shamir $\Rightarrow$ Non-interactive (QROM)
- classical NIZK + classical FHE $\Rightarrow$ Zero-Knowledge (requires circuit-private FHE)

Further Work

- $\lightning$ Relies on hardcore-bit properties
- $\lightning$ Polynomially many repetitions needed

[Ala+20] Non-interactive classical verification of quantum computation

- Make first message instance-independent in offline-step
- Use a parallel repetition theorem to run $3\text{poly}(\lambda)$ steps in 3 steps
- Fiat-Shamir $\Rightarrow$ Non-interactive (QROM)
- classical NIZK + classical FHE $\Rightarrow$ Zero-Knowledge (requires circuit-private FHE)

[Bar+22] Succint Classical Verifcation of Quantum Computation

- Succint Key Generation based on iO / PPRF
- SNARGs in QROM

[Ala+20] Non-interactive classical verification of quantum computation

1. Make first message instance-independent in offline-step
    - Initial message depends on sequence of basis choices
    - Random choice correct with constant probability

    $\Rightarrow$ Increase copies of ground state by constant factor s.t. at least one copy with consistent assignment

[Ala+20] Non-interactive classical verification of quantum computation

1. Make first message instance-independent in offline-step
   - Initial message depends on sequence of basis choices
   - Random choice correct with constant probability

   $\Rightarrow$ Increase copies of ground state by constant factor s.t. at least one copy with consistent assignment
2. Parallel repetition

   $\lightning$ Private coin, rewinding (nested rejection sampling)
   - For NO instance: path of Verifier for two challenges correspond to nearly computational orthogonal projectors
   - k-fold parallel repetition: each pair of distinct challenge tuples correspond to nearly orthogonal projectors
   - Prover can only succeed in negligible fraction of challenge strings
   - $\delta \rightarrow \delta^k$

[Ala+20] Non-interactive classical verification of quantum computation

1. Make first message instance-independent in offline-step
    - Initial message depends on sequence of basis choices
    - Random choice correct with constant probability

    $\Rightarrow$ Increase copies of ground state by constant factor s.t. at least one copy with consistent assignment
2. Parallel repetition
    - $\lightning$ Private coin, rewinding (nested rejection sampling)
    - For NO instance: path of Verifier for two challenges correspond to nearly computational orthogonal projectors
    - k-fold parallel repetition: each pair of distinct challenge tuples correspond to nearly orthogonal projectors
    - Prover can only succeed in negligible fraction of challenge strings
    - $\delta \rightarrow \delta^k$
3. Zero-Knowledge
    - classical NIZK + FHE
    - encryption of key provided in setup Phase
    - $\lightning$ Assumption: setup by trusted third party

[Ala+20] Non-interactive classical verification of quantum computation

1. Make first message instance-independent in offline-step
   - Initial message depends on sequence of basis choices
   - Random choice correct with constant probability
   - $\Rightarrow$ Increase copies of ground state by constant factor s.t. at least one copy with consistent assignment
2. Parallel repetition
   - $\lightning$ Private coin, rewinding (nested rejection sampling)
   - For NO instance: path of Verifier for two challenges correspond to nearly computational orthogonal projectors
   - k-fold parallel repetition: each pair of distinct challenge tuples correspond to nearly orthogonal projectors
   - Prover can only succeed in negligible fraction of challenge strings
   - $\delta \to \delta^k$
3. Zero-Knowledge
   - classical NIZK + FHE
   - encryption of key provided in setup Phase
   - $\lightning$ Assumption: setup by trusted third party
4. Fiat-Shamir
   - $c = \mathcal{H}(H_x, \mathsf{pk}, y)$
   - QROM

Succint classical verification of quantum computation

1. "Succint batch key generation algorithm"
    - outputs short description of many $(\mathsf{pk}, \mathsf{sk})$ pairs
    - can be constructed from iO + PPRFs
    - compose succint key generation with TCF+
2. provides template for succint arguments for QMA

Summary

- Quantum computations have an advantage over classical computations

Summary

- Quantum computations have an advantage over classical computations
- Mahadev's Protocol
  - verifying quantum computations with quantum-secure cryptography and interaction
  - Verifier chooses basis measurement and sends TCF+ or TIF+
  - Prover commits classically to a claw
  - Verifier picks test challenge or Hadamard measurement
  - Prover measures and sends result to Verifier

Summary

- Quantum computations have an advantage over classical computations
- Mahadev's Protocol
  - verifying quantum computations with quantum-secure cryptography and interaction
  - Verifier chooses basis measurement and sends TCF+ or TIF+
  - Prover commits classically to a claw
  - Verifier picks test challenge or Hadamard measurement
  - Prover measures and sends result to Verifier
- Complete, Soundness (reduce perfect attackers to trivial attackers)

Summary

- Quantum computations have an advantage over classical computations
- Mahadev's Protocol
  - verifying quantum computations with quantum-secure cryptography and interaction
  - Verifier chooses basis measurement and sends TCF+ or TIF+
  - Prover commits classically to a claw
  - Verifier picks test challenge or Hadamard measurement
  - Prover measures and sends result to Verifier
- Complete, Soundness (reduce perfect attackers to trivial attackers)
- Application of parallel repetition, FS possible
- ZK possible
- Succint arguments with succint key generation based on iO / PPRF possible

References I

[Ala+20] Gorjan Alagic et al. "Non-interactive Classical Verification of Quantum Computation". In: *Theory of Cryptography*. Springer International Publishing, 2020, pp. 153–180. DOI: 10.1007/978-3-030-64381-2_6. URL: https://doi.org/10.1007%2F978-3-030-64381-2_6.

[Bar+22] James Bartusek et al. *Succinct Classical Verification of Quantum Computation*. 2022. arXiv: 2206.14929 [quant-ph].

[BL08] Jacob D. Biamonte and Peter J. Love. "Realizable Hamiltonians for universal adiabatic quantum computers". In: *Physical Review A* 78.1 (July 2008). DOI: 10.1103/physreva.78.012352. URL: https://doi.org/10.1103%2Fphysreva.78.012352.

[Bra+20] Zvika Brakerski et al. *Simpler Proofs of Quantumness*. 2020. arXiv: 2005.04826 [quant-ph].

[FHM18] Joseph F. Fitzsimons, Michal Hajduš ek, and Tomoyuki Morimae. "Post hoc verification with a single prover". In: *Physical Review Letters* 120.4 (Jan. 2018). DOI: 10.1103/physrevlett.120.040501. URL: https://doi.org/10.1103%2Fphysrevlett.120.040501.

References II

[KSV02] Alexei Y. Kitaev, A. H. Shen, and Mikhail N. Vyalyi. "Classical and Quantum Computation". In: *Graduate Studies in Mathematics*. 2002. URL: https://api.semanticscholar.org/CorpusID:119772104.

[Mah18] Urmila Mahadev. *Classical Verification of Quantum Computations*. 2018. arXiv: 1804.01082 [quant-ph].