THEORY OF COMPUTING, Volume 14 (11), 2018, pp. 1–37
www.theoryofcomputing.org

How to Verify a Quantum Computation

Anne Broadbent*

Received October 3, 2016; Revised October 27, 2017; Published June 11, 2018

To my daughter Émily, on the occasion of her first birthday.

Abstract: We give a new theoretical solution to a leading-edge experimental challenge, namely to the verification of quantum computations in the regime of high computational complexity. Our results are given in the language of quantum interactive proof systems. Specifically, we show that any language in BQP has a quantum interactive proof system with a polynomial-time classical verifier (who can also prepare random single-qubit pure states), and a quantum polynomial-time prover. Here, soundness is unconditional—i. e., it holds even for computationally unbounded provers. Compared to prior work achieving similar results, our technique does not require the encoding of the input or of the computation; instead, we rely on encryption of the input (together with a method to perform computations on encrypted inputs), and show that the random choice between three types of input (defining a *computational run*, versus two types of *test runs*) suffices. Because the overhead is very low for each run (it is linear in the size of the circuit), this shows that verification could be achieved at minimal cost compared to performing the computation. As a proof technique, we use a reduction to an entanglement-based protocol; to the best of our knowledge, this is the first time this technique has been used in the context of verification of quantum computations, and it enables a relatively straightforward analysis.

ACM Classification: F.1.3

AMS Classification: 68Q15, 81P68

Key words and phrases: complexity theory, cryptography, interactive proofs, quantum computing, quantum interactive proofs, quantum cryptography

*This material is based upon work supported by the Air Force Office of Scientific Research under award number FA9550-17-1-0083, Canada's NSERC, and the University of Ottawa's Research Chairs program.

 DOI: 10.4086/toc.2018.v014a011

1 Introduction

Feynman [24] was the first to point out that quantum computers, if built, would be able to perform quantum simulations (i. e., to compute the predictions of quantum mechanics; which is widely believed to be classically intractable). But this immediately begs the question: if the output of a quantum computation cannot be predicted, how do we know that it is correct? Conventional wisdom would tell us that we can rely on testing *parts* (or scaled-down versions) of a quantum computer—conclusive results would then extrapolate to the larger system. But this is somewhat unsatisfactory, since we may not rule out the hypothesis that, at a large scale, quantum computers behave unexpectedly. A different approach to the verification of a quantum computation would be to construct a number of quantum computers based on different technologies (e. g., with ionic, photonic, superconducting and/or solid state systems), and to accept the computed predictions if the experimental results agree. Again, this is still somewhat unsatisfactory, as a positive outcome does not confirm the correctness of the output, but instead confirms that the various large-scale devices behave similarly on the given instances.

This problem, though theoretical in nature [6], is already appearing as a major experimental challenge. One of the outstanding applications for the verification of quantum systems is in quantum chemistry, where the current state-of-the-art is that the inability to verify quantum simulations is much more the norm than the exception [29]. Any theoretical advance in this area could have dramatic consequences on applications of quantum chemistry simulations, including the potential to revolutionize drug discovery. Another case where experimental techniques are reaching the limits of classical verifiability is in the Boson Sampling problem [1], where the process of verification has been raised as a fundamental objection to the viability of experiments [30] (fortunately, these claims are refuted [2], and progress was made in the experimental verification [45]).

As mere classical probabilistic polynomial-time[1] individuals, we appear to be in an impasse: how can we validate the output of a quantum computation?[2] For some problems of interest in quantum computing (such as factoring and search), a claimed solution can be efficiently verified by a classical computer. However, current techniques do not give us such an efficient verification procedure for the *hardest* problems that can be solved by quantum computers (such problems are known as BQP-complete, and include the problem of approximating the Jones polynomial [5]). Here, we propose a solution based on *interaction*, viewing an experiment not in the traditional, static, predict-and-verify framework, but as an interaction between an experimentalist and a quantum device. In the context of theoretical computer science, it has been established for quite some time that interaction between a probabilistic polynomial-time *verifier* and a computationally unbounded *prover* allows the verification of a class of problems *much* wider than what static proofs allow.[3]

Interactive proof systems traditionally model the prover as being all-powerful (i. e., computationally unbounded).[4] For our purposes, we restrict the prover to being a "realistic" quantum device, i. e., we model the prover as a quantum polynomial-time machine. Our approach equates the verifier with a

[1] I. e., assuming humans can flip coins and execute classical computations that take time polynomial in n to solve on inputs of size n.

[2] Assuming the widely-held belief that $\mathsf{BQP} \neq \mathsf{BPP}$, i. e., that quantum computers are indeed more powerful than classical computers.

[3] This is the famous $\mathsf{IP} = \mathsf{PSPACE}$ result [40, 43].

[4] Notable exceptions include [20, 31].

classical polynomial-time machine, augmented with *extremely* rudimentary quantum operations, namely of being able to prepare random single pure-state qubits (chosen among a specific set, see Section 3). Our verifier does not require any quantum memory or quantum processing power. Without loss of generality, the random quantum bits can be sent in the first round, the balance of the interaction and verifier's computation being classical. Formally, we present our results in terms of an *interactive proof system*, showing that in our model, it is possible to devise a *quantum-prover interactive proof system* for all problems solvable (with bounded error) in quantum polynomial time.

1.1 Related work

The complexity class QPIP, corresponding to quantum-prover interactive proof systems, was originally defined by Aharonov, Ben-Or and Eban [3], who, using techniques from [11], showed that $\mathsf{BQP} = \mathsf{QPIP}$ for a verifier with the capacity to perform quantum computations on a constant-sized quantum register (together with polynomial-time classical computation). The main idea of [3] is to encode the input into a *quantum authentication code* [9], and to use interactive techniques for *quantum computing on authenticated data* in order to enable verification of a quantum computation. This result was revisited in light of foundations of physics in [6], and the protocol was also shown secure in a scenario of *composition* [16].

In a different line of research, Kashefi and Fitzsimons [27] consider a measurement-based approach to the problem, giving a scheme that requires the verifier to prepare only random single qubits: the main idea is to encode the computation into a larger one which includes a verification mechanism, and to execute the resulting computation using blind quantum computing [15]. Thus, success of the encoded computation can be used to deduce the correctness of actual computation. A small-scale version of this protocol was implemented in quantum optics [10]. Further work by Kapourniotis, Dunjko and Kashefi [37] shows how to combine the [3] and [27] protocols in order to reduce the quantum communication overhead; Kashefi and Wallden [38] also show how to reduce the overhead of [27].

To the best of our knowledge, the proof techniques in these prior works appear as sketches only, or are cumbersome. In particular, the approach that uses quantum authentication codes [3] is based on [11]. However, the full proof of security for [11] never appeared. Although [3] makes significant progress towards closing this gap, it provides only a sketch of how the soundness is established in the interactive case (see, however, the very recent [4]). A full proof of soundness for [3] follows from [16], however the proof is very elaborate and phrased in terms of a rather different cryptographic task (called "quantum one-time programs"). In terms of the measurement-based approach, note that a proposed protocol for verification in [15] was deemed incomplete [27], but any gaps were addressed in [27]. In this case, however, the protocol (and proof) are very elaborate, and to the best of our knowledge, remain unpublished.[5] Note, however that follow-up work has appeared in peer-reviewed form [37, 38], and that these works consider the more general problem of verification for *quantum* inputs and outputs.

A related line of research also studies the problem of verification with a client that can perform only single-qubit measurements [35]; the case of untrusted devices is also considered in [34]. In sharp contrast to these approaches, Reichardt, Unger and Vazirani [42] show that it is possible to make the verifier *completely* classical, as long as we postulate *two* non-communicating entangled provers. (This could be

[5] This work has been published as [28] during the review period of the current paper.

enforced, for instance, by space-like separation such that communication between the provers would be forbidden by the limit on the speed of light.) The main technique used is a *rigidity theorem* which, provided that the provers pass a certain number of tests, gives the verifier a tight classical control on the quantum provers. Very recently, Coladangelo, Grilo, Jeffery, and Vidick [19] have used the techniques described here to achieve efficient schemes for verifying quantum computations in the model of a classical verifier and two entangled provers.

1.2 Contributions

Our main contributions are a new, simple quantum-prover interactive proof system for BQP, with a verifier whose quantum power is limited to the random preparation of single-qubit pure states, together with a new proof technique.

New protocol. All prior approaches to the verification of quantum computations required some type of encoding (either of the input or of the computation), or otherwise had the verifier perform part of the computation. In contrast, our protocol achieves soundness via the verifier's random choice of different types of runs. This is a typical construction in interactive proofs, and in some sense it is surprising that it is used here for the first time in the context of verifying quantum computations. According to the new protocol, the overhead required for verification can be reduced to repetition of a very simple protocol (with overhead at most linear compared to performing the original computation), and thus may lead to implementations sooner than expected (in general, it is much easier to repeat an experiment using different initial settings, than to run a single, more complex experiment!).

New proof technique. In order to prove soundness, we use the proof technique of a reduction to an "entanglement-based" protocol. This proof technique originates from Shor and Preskill [44] and has been used in a number of quantum cryptographic scenarios, e. g., [21,23,25]. To the best of our knowledge, this is the first time that this technique is used in the context of the verification of quantum computations; we show how the technique provides a much-needed succinct and convincing method to prove soundness. In particular, it allows us to reduce the analysis of an interactive protocol to the analysis of a non-interactive one, and to formally delay the verifier's choice of run until *after* the interaction with the prover.

Furthermore, this work unifies the two distinct approaches given above, (one based on quantum authentication codes and the other on measurement-based quantum computing). Indeed, one can view our protocol as performing a very basic type of quantum computing on authenticated data [16]; with hidden gates being executed via a computation-by-teleportation process [33] that is reminiscent of measurement-based quantum computation, and thus of blind quantum computation [15].

On the conceptual front, this work focuses on the *simplest possible* way to achieve a quantum-prover interactive proof system. Via this process, we have further emphasized links between various concepts.

1. **A link between input privacy and verification.** Prior results [3, 15, 16, 27] all happened to provide both *privacy* of a quantum computation and its *verification* (one notable exception being the recent [26]). Here, we make this link explicit, essentially starting from input privacy and constructing a verifiable scheme (this was also done, to a certain extent in [15,27]).

2. **A link between fault-tolerant quantum computation and cryptography.** Prior results [3,11,16] used constructions inspired by fault-tolerant quantum computation. Here, we make the link even more explicit by using single-qubit gate gadgets that are adaptations of the gate gadgets used in fault-tolerant quantum computation. Furthermore, our results also emphasize how the ubiquitous technique of "tracking the Pauli frame" from fault-tolerant quantum computation can be re-phrased in terms of keeping track of an encryption key.
3. **A link between entanglement and parallelization.** It is known that entanglement can reduce the number of rounds in quantum interactive proof systems [39]; a consequence of our entanglement-based protocol is that we can parallelize our interactive proof system to a single round, as long as we are willing to allow the prover to share entanglement with the verifier, and to perform adaptive measurements.

1.3 Overview of techniques

The main idea for our quantum-prover interactive proof system is that the verifier chooses randomly to interact with the prover in one of three runs. Among these runs, one is the *computation* run, while the two others are *test* runs. In an honest interaction, the output of the computation run is the result (a single bit) of evaluating the given quantum circuit. The test runs are used to detect a deviating prover; there are two types of test runs: an X-*test* and a Z-*test*. Intuitively (and formally proved in Section 7.1), we see that the prover cannot distinguish between all three runs. Thus, his strategy must be invariant over the different runs. It should be clear now how this work links *input privacy* with verification: by varying the input to the computation, the verifier differentiates between test and computation runs; by input privacy, however, the prover cannot identify the type of run and thus any deviation from the prescribed protocol has a chance of being detected.

In more details, the runs have the following properties (from the point of view of the verifier)

- **Computation run.** In a computation run, the prover executes the target circuit on input $|0\rangle^{\otimes n}$.
- X**-test run.** In an X-test run, the prover executes the identity circuit on input $|0\rangle^{\otimes n}$. At the end of the computation, the verifier verifies that the result is 0. This test also contains internal checks for cheating within the protocol.
- Z**-test run.** In a Z-test run, the prover executes the identity circuit on input $|+\rangle^{\otimes n}$. This test run is used only as an internal check for cheating within the protocol.

In order for the prover to execute the above computations without being able to distinguish between the runs, we use a technique inspired by *quantum computing on encrypted data (QCED)* [14,25]: the input qubits are encrypted with a random Pauli, as are auxiliary qubits that are used to drive the computation.

Viewing the target computation as a sequence of gates in the universal set of gates $\{\mathsf{X},\mathsf{Z},\mathsf{H},\mathsf{CNOT},\mathsf{T}\}$ (see Section 2.1 for notation), the task we face is, in the computation run, to perform these logical gates on encrypted quantum data. Furthermore, the X- and Z-test runs should (up to an encryption key), leave the quantum wires in the $|0\rangle^{\otimes n}$ or $|+\rangle^{\otimes n}$ state, respectively. Performing Pauli gates in this fashion is straightforward, as this can be done by adjusting the encryption key (in the computation run only). As we show in Section 4.2, the CNOT gate can be executed directly (since it does not have any effect on the wires for the test runs). The T-gate (Section 4.3) is performed using a construction ("gate gadget") inspired both by QCED and fault-tolerant quantum computation [12] (see also [17]); the T-gate gadget involves the use of an auxiliary qubit and classical interaction. The H is performed thanks to an identity involving the H and P (Section 4.4). Note that P can be accomplished as T^2.

In order to prove soundness, we consider any general deviation of the prover, and show that such deviation can be mapped to an attack on the measured wires only, corresponding to an honest run of the protocol (without loss of generality, we can also delay all measurements until the end of the protocol). Furthermore, because the computation is performed on encrypted data, by the *Pauli twirl* [22], this attack can be described as a convex combination of Pauli attacks on the measured qubits. Since all measurements are performed in the computational basis, Z attacks are obliterated, and thus the only family of attacks of concern consists in X- and Y-gates applied to various measured qubits; these act as bit flips on the corresponding classical output. We show that the combined effect of test runs is to detect *all* such attacks; this allows us to bound the probability that the verifier accepts a *no*-instance. Since only X and Y attacks require detection, one may wonder why we use also a Z-test run. The answer to this question lies in the implementation of the H-gate: while its net effect is to apply the identity in the test runs, its internal workings temporarily *swap* the roles of the X- and Z-test runs; thus the Z-test runs are also used to detect X and Y errors.

Finally, some words on showing indistinguishability between the test and computation runs. This is done by showing that the verifier can delay her choice of run (computation, X- or Z-test) until *after* the interaction with the prover is complete. This is accomplished via an entanglement-based protocol, where the verifier's messages to the prover consist in only half-EPR pairs, as well as classical random bits. These messages are identical in both the test and computation runs; as the verifier decides on the type of run only *after* having the interacted with the prover. Depending on this choice, the verifier performs measurements on the system returned by the prover, resulting in the desired effect.

1.4 Open problems

The main outstanding open problem is the verifiability of a quantum computation with a *classical* verifier, interacting with a *single* quantum polynomial-time prover. In this context, we make a few observations.

- If the prover is unbounded, there exists a quantum interactive proof system for BQP, since $\mathsf{QIP}(=\mathsf{PSPACE})=\mathsf{IP}$.[6]
- If $\mathsf{P}=\mathsf{BQP}$, there is a trivial quantum interactive proof system.
- One possible approach would be to relax the definition to require only *computational* soundness (following the lines of Brassard, Chaum and Crépeau [13], this would lead to a quantum interactive *argument*). This approach seems promising, especially if we consider a computational assumption that is *post-quantum* secure. If, via its interaction with the prover, a classical verifier accepts, then we can conclude that either the verifier performed the correct computation *or* the prover has broken the computational assumption.

1.5 Organization

The remainder of this paper is organized as follows. Section 2 presents some preliminary notation and background. Section 3 defines quantum-prover interactive proofs and states our main theorem. Section 4

[6] $\mathsf{QIP}=\mathsf{PSPACE}$ is due to [36].

describes the interactive proof system, for which we show completeness (Section 6), and soundness (Section 7).

2 Preliminaries

2.1 Notation

We assume the reader is familiar with the basics of quantum information [41]. We use the following well-known qubit gates

$$\mathsf{X} : |j\rangle \mapsto |j \oplus 1\rangle\,, \tag{2.1}$$

$$\mathsf{Z} : |j\rangle \mapsto (-1)^j |j\rangle\,, \tag{2.2}$$

$$\text{Hadamard} \quad \mathsf{H} : |j\rangle \mapsto \frac{1}{\sqrt{2}}(|0\rangle + (-1)^j |1\rangle)\,, \tag{2.3}$$

$$\text{phase gate} \quad \mathsf{P} : |j\rangle \mapsto i^j |j\rangle\,, \tag{2.4}$$

$$\pi/8 \text{ rotation} \quad \mathsf{T} : |j\rangle \mapsto e^{(i\pi/4)^j} |j\rangle)\,, \qquad \text{and the} \tag{2.5}$$

$$\text{two-qubit controlled-not} \quad \mathsf{CNOT} : |j\rangle|k\rangle \mapsto |j\rangle|j \oplus k\rangle\,. \tag{2.6}$$

Let $\mathsf{Y} = i\mathsf{XZ}$. We denote by $\mathbb{P}_n$ the set of n-qubit Pauli operators, where $P \in \mathbb{P}_n$ is given by $P = P_1 \otimes P_2 \otimes \cdots \otimes P_n$ where $P_i \in \{I, \mathsf{X}, \mathsf{Y}, \mathsf{Z}\}$; we also denote an *EPR pair*

$$\left|\Phi^+\right\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)\,. \tag{2.7}$$

2.2 Quantum encryption and the Pauli twirl

The quantum one-time pad encryption maps a single-qubit system ρ to

$$\frac{1}{4} \sum_{a,b \in \{0,1\}} \mathsf{X}^a \mathsf{Z}^b \rho \mathsf{Z}^b \mathsf{X}^a = \frac{I}{2}\,; \tag{2.8}$$

its generalization to n-qubit systems is straightforward [7]. Here, we take (a,b) to be the classical private *encryption key*. Clearly, this scheme provides information-theoretic security, while allowing decryption, given knowledge of the key. A useful observation is that if we have an *a priori* knowledge of the quantum operator ρ, then it may not be necessary to encrypt it with a full quantum one-time pad (e. g., if the state corresponds to a pure state of the form $(1/\sqrt{2})(|0\rangle + e^{i\theta}|1\rangle)$, it can be encrypted with a random Z), although there is no loss of generality in encrypting it with the full random Pauli. We use the two interpretations interchangeably.

Consider for a moment the classical one-time pad (that encrypts a plaintext message by XORing it with a random bit-string of the same length). It is intuitively clear that if an adversary (who does not know the encryption key) has access to the ciphertext only, and is allowed to modify it, then the effect of any adversarial attack (after decryption) is to probabilistically introduce bit flips in target locations. The quantum analogue of this is given by the *Pauli twirl* [22].

Lemma 2.1 (Pauli Twirl)**.** *Let $P, P' \in \mathbb{P}_n$. Then*

$$\frac{1}{|\mathbb{P}_n|} \sum_{Q \in \mathbb{P}_n} Q^* P Q \rho Q^* P'^* Q = \begin{cases} 0, & P \neq P', \\ P \rho P^*, & \textit{otherwise}. \end{cases} \tag{2.9}$$

We also obtain the classical case for the single-qubit Pauli twirl, alluded to above, as the following.

Lemma 2.2 (Classical Pauli Twirl)**.** *Let $c, i \in \{0,1\}$ and $P, P' \in \mathbb{P}_1$. Then*

$$\frac{1}{2} \sum_{Q \in \{I, \mathsf{X}\}} \langle i | Q^* P Q | c \rangle \langle c | Q^* P' Q | i \rangle = \begin{cases} 0, & P \neq P', \\ \langle i | P | c \rangle \langle c | P | i \rangle, & \textit{otherwise}. \end{cases} \tag{2.10}$$

Proof. The proof is a simple application of Lemma 2.1, together with the observation that $|0\rangle, |1\rangle$ are eigenstates of Z:

$$\frac{1}{2} \sum_{Q \in \{I, \mathsf{X}\}} \langle i | Q^* P Q | c \rangle \langle c | Q^* P' Q | i \rangle = \frac{1}{4} \sum_{Q \in \{I, \mathsf{X}, \mathsf{Y}, \mathsf{Z}\}} \langle i | Q^* P Q | c \rangle \langle c | Q^* P' Q | i \rangle \tag{2.11}$$

$$= \begin{cases} 0, & P \neq P', \\ \langle i | P | c \rangle \langle c | P | i \rangle, & \textit{otherwise}. \end{cases} \tag{2.12}$$

□

Working in the basis $\{|+\rangle, |-\rangle\}$, we also obtain the following.

Lemma 2.3. *Let $c, i \in \{0,1\}$ and $P, P' \in \mathbb{P}_1$. Then*

$$\frac{1}{2} \sum_{Q \in \{I, \mathsf{X}\}} \langle i | Q^* H P H Q | c \rangle \langle c | Q^* H P' H Q | i \rangle = \begin{cases} 0, & P \neq P', \\ \langle i | HPH | c \rangle \langle c | HPH | i \rangle, & \textit{otherwise}. \end{cases} \tag{2.13}$$

3 Definitions and statement of results

Interactive proof systems were introduced by Babai [8] and Goldwasser, Micali, and Rackoff [32]. An interactive proof system consists of an interaction between a computationally unbounded prover and a computationally bounded probabilistic verifier. For a language L and a string x, the prover attempts to convince the verifier that $x \in L$, while the verifier tries to determine the validity of this "proof." Thus, a language L is said to have an interactive proof system if there exists a polynomial-time verifier V with the following properties.

- (Completeness) if $x \in L$, there exists a prover (called an honest prover) such that the verifier accepts with probability $p \geq 2/3$;
- (Soundness) if $x \notin L$, no prover can convince V to accept with probability $p \geq 1/3$.

The class of languages having interactive proof systems is denoted IP.

Watrous [46] defined QIP as the quantum analogue of IP, i. e., as the class of languages having a *quantum* interactive proof system, which consists in a quantum interaction between a computationally unbounded quantum prover and a computationally bounded quantum verifier, with the analogous completeness and soundness conditions as given above.

For our results, we are interested in the scenario of a polynomial-time prover (in the honest case), as well as an *almost-classical* verifier; that is, a verifier with the power to generate random qubits as specified by a parameter $\mathcal{S}$ (Definition 3.1). Furthermore, as a technicality, instead of considering languages, we consider promise problems. A promise problem $\Pi = (\Pi_Y, \Pi_N)$ is a pair of disjoint sets of strings, corresponding to YES and NO instances, respectively. For a formal treatment of the model (which we specialize here to our scenario), see [46].

Definition 3.1. Let $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_\ell\}$ where $\mathcal{S}_i = \{\rho_1, \ldots, \rho_{\ell_i}\}$ $(i = 1, \ldots, \ell)$ is a set of density operators. A $\mathcal{S}$-*quantum-prover Interactive Proof System for a promise problem* $\Pi = (\Pi_Y, \Pi_N)$ is an interactive proof system with a verifier V that runs in classical probabilistic polynomial time, augmented with the capacity to randomly generate states in each of $\mathcal{S}_1, \ldots, \mathcal{S}_\ell$ (upon generation, these states are immediately sent to the prover, with the index $i \in \{1, \ldots, \ell\}$ known to the verifier and prover, and the index $j \in \{1, \ldots, \ell_i\}$ known to the verifier only). The interaction of the verifier V and the prover P satisfies the following conditions.

- (Completeness) if $x \in \Pi_Y$, there exists a quantum polynomial-time prover (called an honest prover) such that the verifier accepts with probability $p \geq 2/3$;
- (Soundness) if $x \in \Pi_N$, no prover (even unbounded) can convince V to accept with probability $p \geq 1/3$.

The class of promise problems having an $\mathcal{S}$-quantum interactive proof systems is denoted $\mathsf{QPIP}_{\mathcal{S}}$. Note that by standard amplification, the class $\mathsf{QPIP}_{\mathcal{S}}$ is unchanged if we replace the completeness parameter c and soundness parameter s by any values, as long as $c - s > 1/\mathsf{poly}(n)$.

Comparing our definition of $\mathsf{QPIP}_{\mathcal{S}}$ to the class of quantum-prover interactive proof systems (QPIP) as given in [3], we note that we have made some modifications and clarifications, namely that the verifier in $\mathsf{QPIP}_{\mathcal{S}}$ does not have any quantum memory and does not perform any gates (QPIP allows a verifier that stores and operates on a quantum register of a constant number of qubits), and that soundness holds against unbounded provers.

Finally, we use the canonical BQP-complete problem [3], defined as follows.

Definition 3.2. The input to the promise problem $\mathsf{Q\text{-}CIRCUIT}$ consists of a quantum circuit made of a sequence of gates, $U = U_T, \ldots, U_1$ acting on n input qubits. (We take these circuits to be given in the universal gateset $\{\mathsf{X}, \mathsf{Z}, \mathsf{H}, \mathsf{CNOT}, \mathsf{T}\}$.) Let

$$p(U) = \| |0\rangle\langle 0| \otimes \mathbb{I}_{n-1} U |0^n\rangle \|^2 \tag{3.1}$$

be the probability of observing "0" as a result of a computational basis measurement of the n^{th} output qubit, obtained by evaluating U on input $|0^n\rangle$.

Then define Q-CIRCUIT= $\{\mathsf{Q-CIRCUIT}_{\mathsf{YES}}, \mathsf{Q-CIRCUIT}_{\mathsf{NO}}\}$ with

$$\mathsf{Q-CIRCUIT}_{\mathsf{YES}} : p(U) \geq 2/3\,, \tag{3.2}$$

$$\mathsf{Q-CIRCUIT}_{\mathsf{NO}} : p(U) \leq 1/3\,. \tag{3.3}$$

We can now formally state our main theorem.

Theorem 3.3 (Main Theorem)**.** *Let*

$$\mathcal{S} = \big\{\{|0\rangle,|1\rangle\},\{|+\rangle,|-\rangle\},\{\mathsf{P}|+\rangle,\mathsf{P}|-\rangle\},\{\mathsf{T}|+\rangle,\mathsf{T}|-\rangle,\mathsf{PT}|+\rangle,\mathsf{PT}|-\rangle\}\big\}. \tag{3.4}$$

Then $\mathsf{BQP} = \mathsf{QPIP}_{\mathcal{S}}$.

4 Quantum-prover interactive proof system

In order to prove Theorem 3.3, we give an interactive proof system (see **Interactive Proof System 1**). This protocol uses the various gate gadgets as described in Sections 4.1–4.4. Completeness is studied in Section 6 and soundness is proved in Section 7.

4.1 X- and Z-gate gadget

In order to apply an X on a qubit register i encrypted with key (a_i, b_i), the verifier updates the key according to $a_i \leftarrow a_i \oplus 1$ (b_i is unchanged). In order to apply an Z on a qubit register i encrypted with key (a_i, b_i), the verifier updates the key according to $b_i \leftarrow b_i \oplus 1$ (a_i is unchanged). This operation is performed only in the computation run.

4.2 CNOT-gate gadget

In order to apply a CNOT gate on the encrypted registers (say with register i being the control and register j the target), the prover simply applies the CNOT gate on the respective registers. The verifier updates the encryption keys according to $a_i \leftarrow a_i; b_i \leftarrow b_i \oplus b_j; a_j \leftarrow a_i \oplus a_j$; and $b_j \leftarrow b_j$. We mention that $\mathsf{CNOT}(|0\rangle|0\rangle) = |0\rangle|0\rangle$ and $\mathsf{CNOT}(|+\rangle|+\rangle) = |+\rangle|+\rangle$; thus in the X- and Z-test runs, the underlying data is unchanged.

4.3 T-gate gadget

Here, we show how the T is performed on encrypted data. This is accomplished using an auxiliary qubit, as well as classical interaction. For the computation run, we use a combination of a protocol inspired from [14, 25], as well as fault-tolerant quantum computation [12] (see also [17]). This is given in Figure 1. In the case of an X and Z test runs, as usual, we want the identity map to be applied. This is done as in Figures 2 and 3, respectively. Correctness of Figures 1–3 is proven in Section 5. Note that we show in Section 6.1 that the set of auxiliary quantum states required by the verifier can be reduced via a simple re-labeling, in order to match the resources required in Theorem 3.3.

Also, note that in this work, we have slightly sacrificed efficiency for clarity in the proof, namely that we could have defined a P-gadget using one simple auxiliary qubit instead of two auxiliary qubits that are used by implementing the P as T^2. Furthermore, we suspect that the P^y gate is unnecessary in Figure 1 and thus that we can simplify the set $\mathcal{S}$ (however, the proof in this case appears to be more elaborate, so once more we choose clarity of the proof over efficiency).

Interactive Proof System 1 Verifiable quantum computation with trusted auxiliary states

Let $\mathcal{C}$ be given as an n-qubit quantum circuit in the universal gateset $\mathsf{X}, \mathsf{Z}, \mathsf{CNOT}, \mathsf{H}, \mathsf{T}$.

1. The verifier randomly chooses to execute one of the following three runs (but does not inform the prover of this choice).
   A. **Computation Run**
      A.1. The verifier encrypts input $|0\rangle^{\otimes n}$ and sends the input to P.
      A.2. The verifier sends auxiliary qubits required for the T-gate gadgets for the computation run as given in Sections 4.4 and 4.3.
      A.3. For each gate G in $\mathcal{C}$: X, Z and CNOT are performed without any auxiliary qubits or interaction as given in Sections 4.1 and 4.2, while the H- and T-gate gadgets are performed using the auxiliary qubits from Step A.2 and the interaction as given in Sections 4.4 and 4.3, respectively.
      A.4. P measures the output qubit and returns the result to V.
      A.5. V decrypts the answer; let the result be c_{comp}. V accepts if $c_{\text{comp}} = 0$; otherwise reject.
   B. X**-test Run**
      B.1. The verifier encrypts input $|0\rangle^{\otimes n}$ and sends the input to P.
      B.2. The verifier sends auxiliary qubits required for the T-gate gadgets for the X-test run as given in Sections 4.4 and 4.3.
      B.3. For each gate G in $\mathcal{C}$: X, Z and CNOT are performed without any auxiliary qubits or interaction as given in Sections 4.1 and 4.2, while the H- and T-gate gadgets are performed using the auxiliary qubits from Step B.2 and the interaction as given in Sections 4.4 and 4.3, respectively.
      B.4. P measures the output qubit and returns the result to V.
      B.5. V decrypts the answer; let the result be c_{test}. V accepts if no errors were detected in Step B.3 *and* if $c_{\text{test}} = 0$; otherwise reject.
   C. Z**-test Run**
      C.1. The verifier encrypts input $|+\rangle^{\otimes n}$ and sends the input to P.
      C.2. The verifier sends auxiliary qubits required for the T-gate gadgets for the Z-test run as given in Sections 4.4 and 4.3.
      C.3. For each gate G in $\mathcal{C}$: X, Z and CNOT are performed without any auxiliary qubits or interaction as given in Sections 4.1 and 4.2, while the H- and T-gate gadgets are performed using the auxiliary qubits from Step C.2 and the interaction as given in Sections 4.4 and 4.3, respectively.
      C.4. P measures the output qubit and returns the result to V.
      C.5. V disregards the output. V accepts if no errors were detected in Step C.3; otherwise reject.

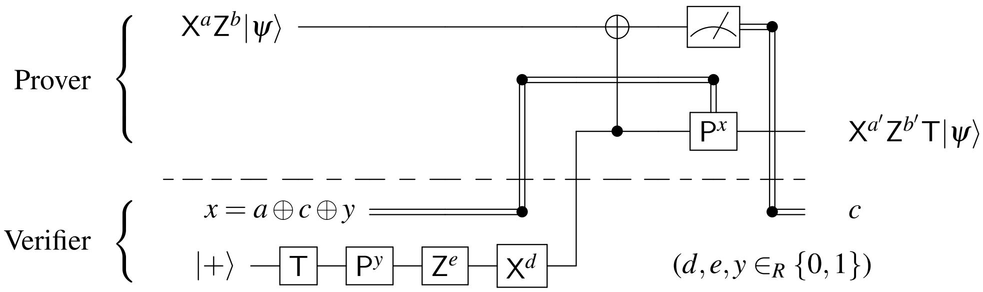

Figure 1: T-gate gadget for a computation run. Here, an auxiliary qubit is prepared by the verifier in the state $\mathsf{X}^d\mathsf{Z}^e\mathsf{P}^y\mathsf{T}|+\rangle$ and sent to the prover. The prover performs a CNOT between the auxiliary register and the data register; and then measures the data register. Given the measurement result, c, the verifier sends a classical message, $x = a\oplus c\oplus y$ to the prover, who applies the conditional gate P^x to the remaining register (which we now re-label as the data register). The verifier's key update rule is given by $a' = a\oplus c$ and $b' = (a\oplus c)\cdot(d\oplus y)\oplus a\oplus b\oplus c\oplus e\oplus y$.

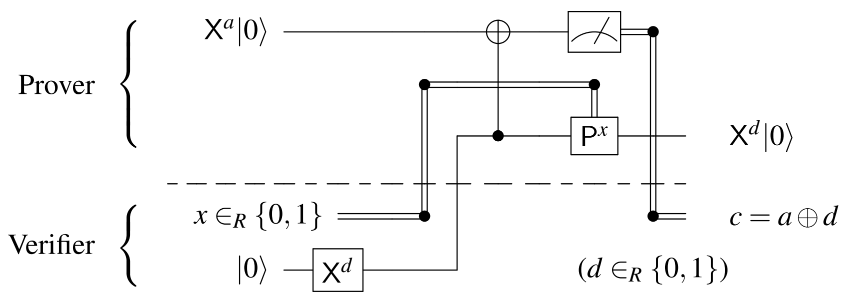

Figure 2: T-gate gadget for an X-gate test run. The goal here is to mimic the interaction established in Figure 1, but to perform the identity operation on the input state $|0\rangle$ (up to encryptions). Here, we include an additional *verification* that $c = a\oplus d$.

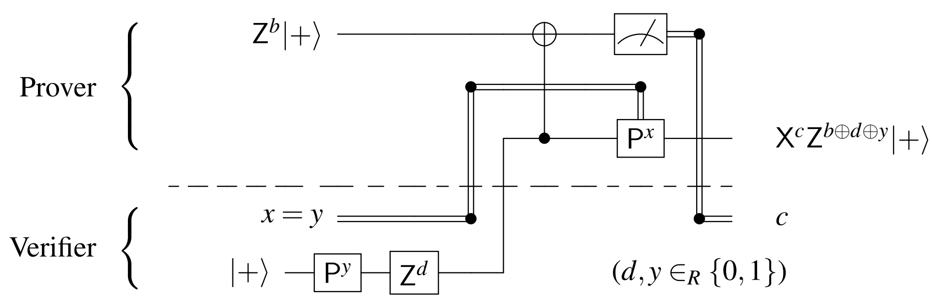

Figure 3: T-gate gadget for a Z-gate test run. The goal here is to mimic the interaction established in Figure 1, but to perform the identity operation on the input state $|+\rangle$ (up to encryptions).

4.4 H-gate gadget

Performing a H gate has the effect of locally swapping between the X- and Z-test runs (as well as swapping the role of the X and Z encryption keys in the computation run). While this is alright if done in isolation, it does not work if the H-gate is performed as part of a larger computation (for instance, a CNOT-gate could no longer be performed as given above as the inputs would not, in general, be of the form $|0\rangle|0\rangle$ (for the X-test run) or $|+\rangle|+\rangle$ (for the Z-test run)). Our solution is to use the following two identities.

$$\mathsf{HPHPHPH} = \mathsf{H}\,, \tag{4.1}$$

$$\mathsf{HHHH} = \mathbb{I}\,. \tag{4.2}$$

Thus we build the gadget so that the prover starts by applying an H at the start. By doing this, we locally swap the roles of the X- and Z-tests we also cause a key update which swaps the role of the X- and Z-encryption keys. For the following P, we apply twice the gadgets from Section 4.3 (taking in to account the swapped role for the test runs). The result is that a P is applied in the computation run, while the identity is applied in the test runs. Now an H is applied, which reverts the roles of the X- and Z-tests. We apply the P again. Continuing in this fashion, we observe the following effect.

1. In the computation run (using twice the gadget of Figure 1 for each P-gate), the effect is to apply H on the input qubit (by Equation (4.1)).
2. In the X-test run (using (twice each time) the gadgets of Figures 3, 2, 3 for the first, second and third P-gate), the effect is to apply the identity.
3. In the Z-test run (using (twice each time) gadgets of Figures 2, 3, 2 for the first, second and third P-gate), the effect is to apply the identity.

5 Correctness of the T-gate protocol

We give below a step-by-step proof of the correctness of the T-gate protocol as given in Figure 1 (Section 4.3). In Section 5.1, we show how similar techniques are used to show corrections of the T-gate protocol for the test runs, as given in Figures 2 and 3. The basic building block is the circuit identity for an X-teleportation from [47]. Also of relevance to this work are the techniques developed by Childs, Leung, and Nielsen [18] to manipulate circuits that produce an output that is correct *up to known Pauli corrections*.

We will make use of the following identities which all hold up to an irrelevant global phase: $\mathsf{XZ} = \mathsf{ZX}$, $\mathsf{PZ} = \mathsf{ZP}$, $\mathsf{PX} = \mathsf{XZP}$, $\mathsf{TZ} = \mathsf{ZT}$, $\mathsf{TX} = \mathsf{XZPT}$, $\mathsf{P}^2 = \mathsf{Z}$ and $\mathsf{P}^{a\oplus b} = \mathsf{Z}^{a\cdot b}\mathsf{P}^{a+b}$ (for $a, b \in \{0,1\}$).

1. We start with the "X-teleportation" of [47], which is easy to verify (Figure 4).

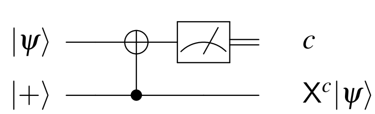

Figure 4: Circuit identity: "X-teleportation."

2. Then we substitute the input $\mathsf{X}^a\mathsf{Z}^b|\psi\rangle$ for the top wire. We add the gate sequence $\mathsf{T}, \mathsf{P}^y, \mathsf{Z}^e, \mathsf{X}^d$, $\mathsf{P}^{a\oplus c\oplus y}$ to the output (Figure 5). By Figure 4, the outcome is given by $\mathsf{P}^{a\oplus c\oplus y}\mathsf{X}^d\mathsf{Z}^e\mathsf{P}^y\mathsf{T}\mathsf{X}^{a\oplus c}\mathsf{Z}^b|\psi\rangle$. We apply the identities given above to simplify this to a Pauli correction (on T) as follows.

$$
\begin{aligned}
\mathsf{P}^{a\oplus c\oplus y}\mathsf{X}^d\mathsf{Z}^e\mathsf{P}^y\mathsf{T}\mathsf{X}^{a\oplus c}\mathsf{Z}^b &= \mathsf{P}^{a\oplus c\oplus y}\mathsf{X}^d\mathsf{Z}^e\mathsf{P}^y\mathsf{X}^{a\oplus c}\mathsf{P}^{a\oplus c}\mathsf{Z}^{b\oplus a\oplus c}\mathsf{T} && (5.1)\\
&= \mathsf{P}^{a\oplus c\oplus y}\mathsf{X}^d\mathsf{Z}^e\mathsf{P}^y\mathsf{P}^{a\oplus c}\mathsf{X}^{a\oplus c}\mathsf{Z}^b\mathsf{T} && (5.2)\\
&= \mathsf{P}^{a\oplus c\oplus y}\mathsf{P}^y\mathsf{X}^d\mathsf{Z}^{d\cdot y\oplus e}\mathsf{P}^{a\oplus c}\mathsf{X}^{a\oplus c}\mathsf{Z}^b\mathsf{T} && (5.3)\\
&= \mathsf{P}^{a\oplus c\oplus y}\mathsf{P}^y\mathsf{P}^{a\oplus c}\mathsf{X}^d\mathsf{Z}^{d\cdot(a\oplus c)}\mathsf{Z}^{d\cdot y\oplus e}\mathsf{X}^{a\oplus c}\mathsf{Z}^b\mathsf{T} && (5.4)\\
&= \mathsf{P}^{(a\oplus c)\oplus y}\mathsf{P}^y\mathsf{P}^{a\oplus c}\mathsf{X}^{a\oplus c\oplus d}\mathsf{Z}^{d(a\oplus c\oplus y)\oplus b\oplus e}\mathsf{T} && (5.5)\\
&= \mathsf{Z}^{y\cdot(a\oplus c)}\mathsf{P}^{a\oplus c}\mathsf{P}^y\mathsf{P}^y\mathsf{P}^{a\oplus c}\mathsf{X}^{a\oplus c\oplus d}\mathsf{Z}^{d(a\oplus c\oplus y)\oplus b\oplus e}\mathsf{T} && (5.6)\\
&= \mathsf{X}^{a\oplus c\oplus d}\mathsf{Z}^{(a\oplus c\oplus d)\cdot(d\oplus y)\oplus a\oplus b\oplus c\oplus d\oplus e\oplus y}\mathsf{T} && (5.7)\\
&= \mathsf{X}^{a\oplus c'}\mathsf{Z}^{(a\oplus c')\cdot(d\oplus y)\oplus a\oplus b\oplus c'\oplus e\oplus y}\mathsf{T}\,, && (5.8)
\end{aligned}
$$

where above we let $c' \leftarrow c\oplus d$.

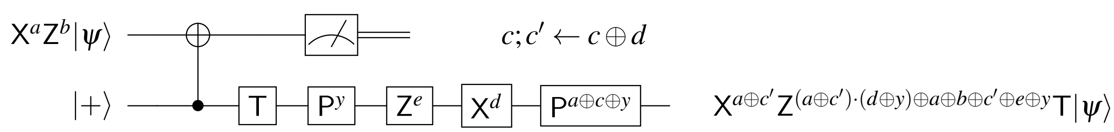

Figure 5

3. Next, we note that, because diagonal gates commute with control, the circuit of Figure 5 is equivalent to the one in Figure 6.

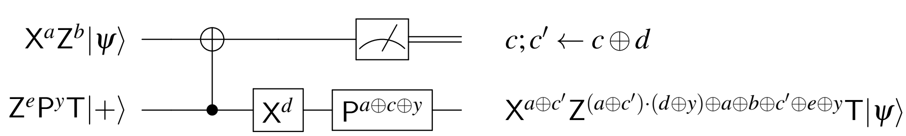

Figure 6

4. We note that the X^d on the bottom wire *after* the CNOT can be moved to the bottom wire *before* the CNOT, as long as we add an X^d to the top wire after the CNOT. (Figure 7.)

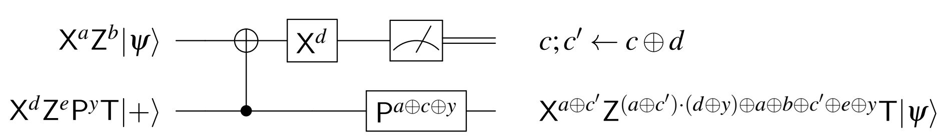

Figure 7

5. Finally, since $c' = c\oplus d$, yet the measurement result c undergoes an X^d, these two operations cancel out, and we obtain the final circuit as in Figure 8.

We note that a more direct proof of correctness for Figure 8 is possible, but that our intermediate Figure 7 is crucial in the proof of soundness.

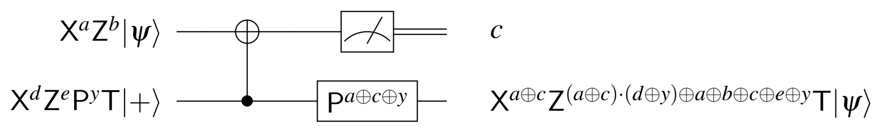

Figure 8

5.1 Correctness of the T-gate gadget in the test runs

The correctness of the T-gate gadget in the X-test run of Figure 2 is straightforward: the CNOT flips the bit in the top wire if and only if $d = 1$, while the P^x has no effect on the computational basis states. The correctness of Figure 3 is derived from the X-teleportation of Figure 4. Since the diagonal gates Z and P commute with control, they can be seen as acting on the output qubit. Furthermore, using $\mathsf{P}^2 = \mathsf{Z}$ and the fact that X has no effect on $|+\rangle$, we get the final circuit in Figure 3.

6 Completeness

Suppose $\mathcal{C}$ is a yes-instance of Q-CIRCUIT. Suppose P follows the protocol honestly. Then we have the following.

1. In the case of a computation run, the output bit, c_{comp} has the same distribution as the output bit of $C(|0^n\rangle)$, thus V accepts with probability at least $2/3$.
2. In the case of an X-test run and in the case of a Z-test run (by the identities and observations from the previous sections), V accepts with probability 1.

Given that each run happens with probability $1/3$, we get that V accepts with probability at least $2/3 + (1/3)\cdot(2/3) = 8/9$.

6.1 Auxiliary qubits for the T-gate gadget

In the protocol for the T-gate gadget (Figure 1), we assume the verifier can produce auxiliary qubits of the form $\mathsf{X}^d\mathsf{Z}^e\mathsf{P}^y\mathsf{T}|+\rangle$. We now show that this is equivalent to requiring the verifier to generate auxiliary qubits of the form $\mathsf{Z}^e\mathsf{P}^y\mathsf{T}|+\rangle$, as claimed in Theorem 3.3. This can be seen by the following equation, which holds up to global phase.

$$\mathsf{X}^d\mathsf{Z}^e\mathsf{P}^y\mathsf{T}|+\rangle = \mathsf{Z}^{e\oplus d}\mathsf{P}^{y\oplus d}\mathsf{T}|+\rangle\,. \tag{6.1}$$

The above can be seen easily since, up to global phase, $\mathsf{XT}|+\rangle = \mathsf{ZPT}|+\rangle$, and $\mathsf{XP} = \mathsf{ZPX}$. The analysis for the case $d = 1$, follows.

$$\begin{aligned}
\mathsf{X}\mathsf{Z}^e\mathsf{P}^y\mathsf{T}|+\rangle &= \mathsf{Z}^e\mathsf{X}\mathsf{P}^y\mathsf{T}|+\rangle && (6.2)\\
&= \mathsf{Z}^e\mathsf{Z}^y\mathsf{P}^y\mathsf{X}\mathsf{T}|+\rangle && (6.3)\\
&= \mathsf{Z}^{e\oplus y}\mathsf{P}^y\mathsf{Z}\mathsf{P}\mathsf{T}|+\rangle && (6.4)\\
&= \mathsf{Z}^{e\oplus y\oplus 1}\mathsf{P}^{y+1}\mathsf{T}|+\rangle && (6.5)\\
&= \mathsf{Z}^{e\oplus y\oplus 1}\mathsf{Z}^y\mathsf{P}^{y\oplus 1}\mathsf{T}|+\rangle && (6.6)\\
&= \mathsf{Z}^{e\oplus 1}\mathsf{P}^{y\oplus 1}\mathsf{T}|+\rangle\,. && (6.7)
\end{aligned}$$

Thus, the verifier chooses a classical x uniformly at random, and if $x = 1$, the verifier re-labels the auxiliary qubits according to Equation (6.1).

7 Soundness

As discussed in Section 1.3, the main idea to prove soundness is to analyze an entanglement-based version of the Interactive Proof System 1. We present the EPR-based version (Section 7.1), and argue why the completeness and soundness parameters are the same. Then, we analyze a general deviating prover P^* in the EPR-based version and show how to simplify an attack (Section 7.2). We then analyze the case of a test run (Section 7.4) and of a computation run (Sections 7.5). In Section 7.6, we show how this completes the proof of our main theorem (Theorem 3.3).

An interesting consequence of the analysis in this section is that it implies that, if we are willing to have the prover and the verifier share entanglement, then the protocol reduces to a single round. (However, in this case, the work of the verifier becomes more important; one can wonder if the verifier is still "almost-classical".) Another interesting observation is that sequential repetition is not required (parallel repetition suffices), due to the fact that the analysis makes use of the Pauli twirl (see Section 7.2), which would also be applicable to the scenario of parallel repetition.

7.1 EPR-based protocol

In this version of the quantum-prover interactive proof system (**Interactive Proof System 2**), *all* quantum inputs sent by the verifier are half-EPR pairs, and *all* classical messages sent by the verifier are random bits. The actions related to choosing between test and computation runs are done *after* the interaction with the server. For the T-gate, this can be done as shown in Figures 9, 10 and 11.

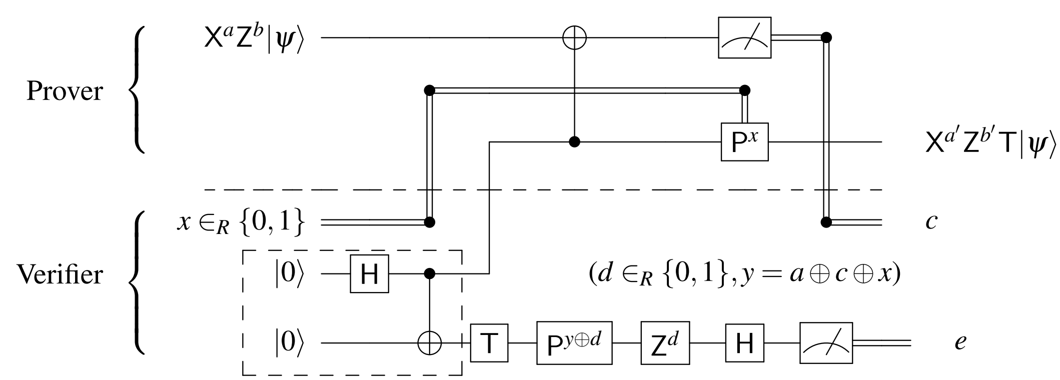

Figure 9: Entanglement-based protocol for a T-gate (computation run). This protocol performs the same computation as the protocol in Figure 1. The output is obtained from the output of Figure 1 by using $y = a \oplus c \oplus x$. The circuit in the dashed box prepares an EPR-pair. Here, $a' = a \oplus c$ and $b' = (a \oplus c) \cdot (d \oplus y) \oplus a \oplus b \oplus c \oplus e \oplus y$.

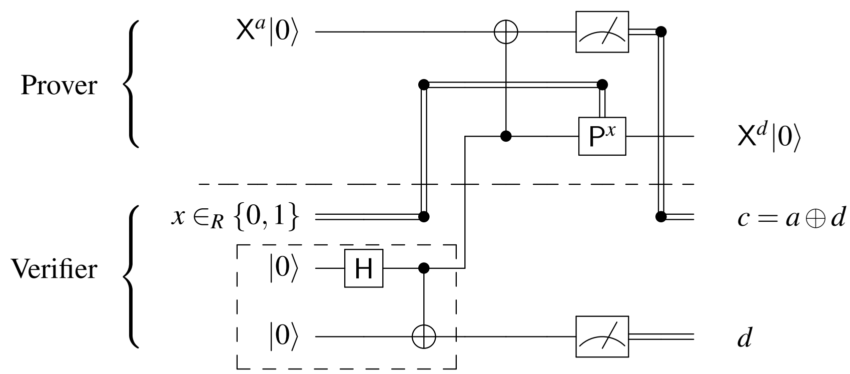

Figure 10: Entanglement-based protocol for a T-gate (X-test run). This protocol performs the same computation as the protocol in Figure 2. The circuit in the dashed box prepares an EPR-pair. As in Figure 2, we include an additional *verification* that $c = a \oplus d$.

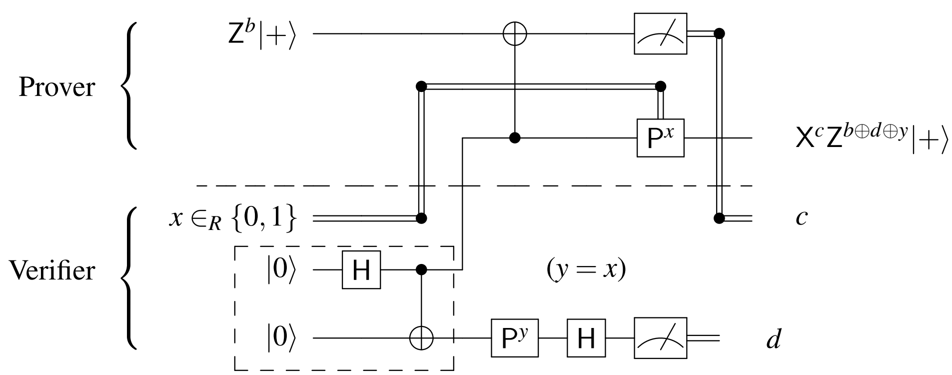

Figure 11: Entanglement-based protocol for a T-gate (Z-test run). This protocol performs the same computation as the protocol in Figure 3. The circuit in the dashed box prepares an EPR-pair.

We therefore define V_{EPR} as a verifier that delays all choices until after the Prover has returned all messages (i. e., as the verifier in Interactive Proof System 2). That the soundness parameter is the same for Interactive Proof Systems 1 and 2 follows from the following series of transformations, each of which preserves the probability of accept/reject.

1. The quantum communication from the verifier to the prover in Interactive Proof System 1 can be generated instead by preparing EPR pairs, sending half and then immediately measuring in the appropriate basis to obtain the required qubit. Call this Interactive Proof System 1.1.
2. Starting from Interactive Proof System 1.1, by a change of variable, we can specify the classical message x to be random in each of the T-gate gadgets. This determines y, and the operation P^y, that is conditioned on y (for the computation and Z-test run) can be applied immediately before the verifier's measurement. Call this Interactive Proof System 1.2.
3. In Interactive Proof System 1.2, the prover and verifier operate on different subsystems. Their operations therefore commute and we can specify that the Verifier delays his operations until *after* the interaction with the Prover. The result of this transformation is Interactive Proof System 2.

Since each transformation above preserves the soundness, and since the completeness parameter is unchanged, from now on, we can focus on establishing the soundness parameter for Interactive Proof System 2.

7.2 Simplifying a general attack

In this section, we derive a simplified expression for a general deviating prover for our Interactive Proof System. We show that without loss of generality, we can rewrite the actions of any deviating prover as the honest prover's actions, followed by an arbitrary cheating map. But first, as a technicality, we consider Interactive Proof System 3, which is closely related to Interactive Proof System 2, but where the prover is unitary and show (Lemma 7.2) that a bound on the soundness of Interactive Proof System 3 implies a bound on the soundness of Interactive Proof System 2.

Definition 7.1. We define *Interactive Proof System 3* as Interactive Proof System 2, but with a *unitary* prover. To be more precise, the prover (say, P^q) in Interactive Proof System 3 performs the same operations as the prover in Interactive Proof System 2, but does not perform any measurements: instead, P^q sends qubits to the verifier, say V^q_{EPR}, who immediately measures them in place of the prover, and then continues according to the Interactive Proof System 2.

Lemma 7.2. *Interactive Proof Systems 2 and 3 have the same completeness, and furthermore, an upper bound on the soundness parameter for Interactive Proof System 3 is an upper bound on the soundness parameter for Interactive Proof System 2.*

Proof. It follows immediately by definition that Interactive Proof Systems 2 and 3 have the same completeness parameter. For soundness, suppose that in Interactive Proof System 3, the probability that V^q_{EPR} accepts while interacting with *any* P^{q*} is at most s. Suppose for a contradiction that there is a P^* for Interactive Proof System 2, such that V_{EPR} accepts with $p > s$.

Using polynomial overhead, we obtain $\widetilde{P^*_{\mathrm{meas}}}$ from P^* via purification (all actions of $\widetilde{P^*_{\mathrm{meas}}}$ are unitary, except a one-qubit register is measured each time the protocol requires a classical message to be sent to the verifier). The probability p of acceptance is unchanged.

Furthermore, starting with P^*, we define $\widetilde{P^*}$ as a prover for Interactive Proof System 3, which behaves like $\widetilde{P^*_{\mathrm{meas}}}$, but instead of measuring messages to be sent to the verifier, it sends qubits, which are immediately measured by V^q_{EPR}, as per Interactive Proof System 3. The probability p that the verifier in Interactive Proof System 3 accepts, when interacting with $\widetilde{P^*}$ is the same as the probability that the verifier in Interactive Proof System 2 accepts, when interacting with P^*. This contradicts $p > s$, and proves the claim. □

Next, we show that, in Interactive Proof System 3, without loss of generality, we can assume that the prover's actions are the honest unitary ones, followed by a general attack. In order to see this, for a t-round protocol (involving t rounds of classical interaction), define a cheating prover's actions at round i by $\Phi_i H_i$, where H_i acts on the qubits used in the computation, as well as the classical bit received in round i, and is the honest application of the prover's unitary circuit, while Φ_i is a general deviating map acting on the classical bit received in round i, the output registers of H_i as well as a private memory register. (Recall that there are no measurements at this point—V does the measurement.)

Interactive Proof System 2 Verifiable quantum computation with trusted auxiliary states- EPR version

Let $\mathcal{C}$ be given as an n-qubit quantum circuit in the universal set of gates $\mathsf{X}, \mathsf{Z}, \mathsf{CNOT}, \mathsf{H}, \mathsf{T}$.

1. The verifier prepares $|\Phi^+\rangle^{\otimes n}$ and sends half of each pair to the prover. These registers are identified with the *input* registers.
2. For each auxiliary qubit required in the H- and T-gate gadgets, the verifier prepares $|\Phi^+\rangle$ and sends half of each pair to the prover.
3. The prover executes the gate gadgets. The verifier records the classical communication and responds with random classical bits (when required).
4. The prover returns a single bit of output, c to the verifier.
5. The verifier randomly chooses to execute one of the following three runs (but does not inform the prover of this choice).
   A. **Computation Run**
      A.1. Measure the remaining input register halves in the computational basis. Take the initial X-encryption key to be the measurement outcomes (set the Z-key to 0).
      A.2. For each gate G in $\mathcal{C}$: perform the key updates for the X, Z , CNOT and H gates. For the T gadget, taking into account the classical messages received and sent in Step 3, perform the measurement and key update rules for the T-gadget (Figure 9).
      A.3. V decrypts the output bit c; let the result be c_{comp}. V accepts if $c_{\text{comp}} = 0$; otherwise reject.
   B. **X-test Run**
      B.1. Measure the remaining input register halves in the computational basis. Take the initial X-encryption key to be the measurement outcomes (set the Z-key to 0).
      B.2. For each gate G in $\mathcal{C}$: perform the key updates for the X, Z, CNOT and H gates. For the T gadget, taking into account the classical messages received and sent in Step 3, perform the measurement, key update rules and tests for the T-gadget (Figure 10).
      B.3. V decrypts the output bit c; let the result be c_{comp}. V accepts if $c_{\text{comp}} = 0$ *and* if no errors were detected in Step B.2; otherwise reject.
   C. **Z-test Run**
      C.1. Measure the remaining input register halves in the Hadamard basis. Take the initial Z-encryption key to be the measurement outcomes (set the X-key to 0).
      C.2. For each gate G in $\mathcal{C}$: perform the key updates for the X, Z, CNOT and H gates. For the T gadget, taking into account the classical messages received and sent in Step 3, perform the measurement, key update rules and tests for the T-gadget (Figure 11)
      C.3. V accepts if no errors were detected in Step C.2; otherwise reject.

Thus the actions of a general prover P^* are given as the follows.

$$\Phi_t H_t \ldots \Phi_1 H_1 \Phi_0 H_0 \,. \tag{7.1}$$

Since $H_0, \ldots, H_t$ are unitary, we can rewrite Equation (7.1).

$$\begin{aligned}
\Phi_t H_t \ldots \Phi_3 H_3 \Phi_2 H_2 \Phi_1 H_1 \Phi_0 H_0 &= \Phi_t H_t \ldots \Phi_3 H_3 \Phi_2 H_2 (\Phi_1 H_1 \Phi_0 H_1{}^*) H_1 H_0 && (7.2)\\
&= \Phi_t H_t \ldots \Phi_3 H_3 (\Phi_2 H_2 \Phi_1 H_1 \Phi_0 H_1{}^* H_2{}^*) H_2 H_1 H_0 && (7.3)\\
&= \Phi_t H_t \ldots (\Phi_3 H_3 \Phi_2 H_2 \Phi_1 H_1 \Phi_0 H_1{}^* H_2{}^* H_3^*) H_3 H_2 H_1 H_0 && (7.4)\\
&= (\Phi_t H_t \ldots \Phi_3 H_3 \Phi_2 H_2 \Phi_1 H_1 \Phi_0 H_1{}^* H_2{}^* H_3^* \ldots H_t^*) H_t \ldots H_3 H_2 H_1 H_0 \,. && (7.5)
\end{aligned}$$

Thus, by denoting a general attack by

$$\Phi = \Phi_t H_t \ldots \Phi_3 H_3 \Phi_2 H_2 \Phi_1 H_1 \Phi_0 H_1{}^* H_2{}^* H_3^* \ldots H_t^* \,, \tag{7.6}$$

and the map corresponding to the honest prover as $H = H_t \ldots H_3 H_2 H_1 H_0$, we get that without loss of generality, we can assume that the prover's actions are the honest ones, followed by a general attack:

$$\Phi H \,. \tag{7.7}$$

Taking $\{E_k\}$ to be Kraus terms associated with Φ, and supposing a total of m qubit registers are involved, we get that the system after interaction with P^*, where the initial state is

$$\left|\Phi^+\right\rangle\!\left\langle\Phi^+\right|^{\otimes m} = \frac{1}{2^m}\sum_{i,j=0}^{2^m-1} |ii\rangle\langle jj| \tag{7.8}$$

(here, we include the classical random bits, as they are uniformly random and therefore we can represent them as maximally entangled states), and where the verifier has not yet performed the gates and measurements, can be described as

$$\frac{1}{2^m}\sum_k \sum_{i,j} (I \otimes E_k H)|ii\rangle\langle jj|(I \otimes H^* E_k^*) \,. \tag{7.9}$$

For a fixed k, we write E_k and E_k^* in the Pauli basis:

$$E_k = \sum_Q \alpha_{k,Q} Q \qquad \text{and} \qquad E_k^* = \sum_{Q'} \alpha_{k,Q'}^* Q' \,. \tag{7.10}$$

(To simplify notation, we assume throughout that Q, Q' ranges over $\mathbb{P}_m$.) By the completeness relation, we have:

$$\sum_k \sum_Q |\alpha_{k,Q}|^2 = 1 \,. \tag{7.11}$$

When it is clear from context, we drop the "k" subscript, thus denoting

$$E_k = \sum_Q \alpha_Q Q \qquad \text{and} \qquad E_k^* = \sum_{Q'} \alpha_{Q'}^* Q' \,. \tag{7.12}$$

In the following sections, we analyze the probability of acceptance, as a function of the type of run and of the prover's attack. By Lemma 7.2, a bound on the acceptance probability gives a bound on the acceptance probability in Interactive Proof System 2 (which is a bound on the acceptance probability of Interactive Proof System 1).

7.3 Conventions and definitions

In addition to the convention of representing an attack as in Equation (7.9), in the following Sections 7.4–7.5, we use the following conventions.

1. The circuit C that we consider is already "compiled" in terms of the H gates as in Section 4.4 (the identity $\mathsf{H} = \mathsf{HPHPHPH}$ is already applied).
2. The number of T-gate gadgets is t (each such gadget uses two auxiliary qubits—one representing an auxiliary quantum bit, and one representing a classical bit x), and the number of qubits in the computation is n. Thus we have $m = 2t + n$.
3. In the T-gate gadget, the auxiliary wire is swapped with the measured wire immediately before the measurement. This way, we may picture that only auxiliary qubits are measured as part of the computation, and that the data registers for the input represent the computation wires throughout.
4. Given the system as in Equation (7.9), we suppose that the first T-gate gadget uses the first EPR pair as auxiliary quantum bit, and the second EPR pair as a qubit representing the classical bit (and so on for the following T-gadgets). The last n EPR pairs are the data qubits, and we suppose that at the end of the protocol, the last data qubit is the one that is measured, representing the output.
5. Normalization constants are omitted when they are clear from context.

Finally, we define *benign* and *non-benign* Pauli attacks, based on their effect on the protocol. As we will see, benign attacks have no effect on the acceptance probability (because all qubits are either traced-out or measured in the computational basis). However, non-benign attacks may influence the acceptance probability.

Definition 7.3. For a fixed Pauli $P \in \mathbb{P}_m$, we call it *benign* if $P \in B_{t,n}$, where $B_{t,n}$ is the set of Paulis acting on $m = 2t + n$ qubits, such that the *measured* qubits in the protocol are acted on only by a gate in $\{I, \mathsf{Z}\}$. Using the above conventions, this means that $B_{t,n} = \{\{\{I, \mathsf{Z}\} \otimes \mathbb{P}\}^{\otimes t} \otimes \mathbb{P}_{n-1} \otimes \{I, \mathsf{Z}\}\}$. A Pauli P is called *non-benign* if at least one *measured* qubit in the protocol is acted on only by a gate in $\{\mathsf{X}, \mathsf{Y}\}$. In analogy to the set of benign Paulis, we denote the set of non-benign Paulis acting on $m = 2t + n$ qubits as $B'_{t,n}$.

7.4 In the case of a test run

Based on the preliminaries of Section 7.2 and Section 7.3, we now bound the probability of acceptance of the test runs, by describing the effect of the attack on the entire system, and considering which attacks are detected by the test runs (essentially, we show in Lemma 7.4 that all non-benign Pauli attacks are detected by one of the test runs).

Lemma 7.4. *Consider the Interactive Proof System 2 for a circuit C on n qubits and with t T-gate gadgets, with attack $\{E_k\}$ (with each $E_k = \sum_Q \alpha_{k,Q} Q$), and suppose a test run is applied. Let $B'_{t,n}$ be the set of non-benign attacks. Then with the following probability, the verifier rejects*

$$\frac{1}{2} \sum_k \sum_{Q \in B'_{t,n}} |\alpha_{k,Q}|^2. \tag{7.13}$$

Proof. As a first step towards proving Lemma 7.4, we derive an expression for the system after the application of the honest circuit (and before any attack). Let $h_i \in \{0, 1\}$ $(i = 1 \dots t)$ be a bit that indicates if the auxiliary qubit i is an encrypted version of $|0\rangle$ ($h_i = 0$) or $|+\rangle$ ($h_i = 1$). We note that, by the properties of the protocol, $h_i = 0$ in an X-test run if and only if $h_i = 1$ in a Z-test run.

In the case of an X-test, the system that we obtain after the verifier has performed measurements that prepare the encrypted auxiliary qubits and the computation wire is

$$\begin{aligned}\sum_{\substack{d_1\dots d_t\in\{0,1\}\\ x_1\dots x_t\in\{0,1\}}} \Big(\mathsf{P}^{h_1\cdot x_1}\mathsf{H}^{h_1}\mathsf{X}^{d_1}|0\rangle\langle 0|\mathsf{X}^{d_1}\mathsf{H}^{h_1}\mathsf{P}^{*h_1\cdot x_1}\otimes \mathsf{X}^{x_1}|0\rangle\langle 0|\mathsf{X}^{x_1}\Big)\otimes\cdots\\ \otimes\Big(\mathsf{P}^{h_t\cdot x_t}\mathsf{H}^{h_t}\mathsf{X}^{d_t}|0\rangle\langle 0|\mathsf{X}^{d_t}\mathsf{H}^{h_t}\mathsf{P}^{*h_t\cdot x_t}\otimes \mathsf{X}^{x_t}|0\rangle\langle 0|\mathsf{X}^{x_t}\Big)\otimes\\ \sum_{a_1\dots a_n\in\{0,1\}}\mathsf{X}^{a_1}|0\rangle\langle 0|\mathsf{X}^{a_1}\otimes\cdots\otimes\mathsf{X}^{a_n}|0\rangle\langle 0|\mathsf{X}^{a_n}\\ \otimes|d_1\dots d_t,x_1\dots x_t,a_1\dots a_n\rangle\langle d_1\dots d_t,x_1\dots x_t,a_1\dots a_n|\,.\end{aligned}\tag{7.14}$$

Note that we have appended a $2t+n$ qubit register, which is held by the verifier and that contains a classical basis state representing the key.

In the case of a Z-test, the system that we start with is

$$\begin{aligned}\sum_{\substack{d_1\dots d_t\in\{0,1\}\\ x_1\dots x_t\in\{0,1\}}} \Big(\mathsf{P}^{h_1\cdot x_1}\mathsf{H}^{h_1}\mathsf{X}^{d_1}|0\rangle\langle 0|\mathsf{X}^{d_1}\mathsf{H}^{h_1}\mathsf{P}^{*h_1\cdot x_1}\otimes \mathsf{X}^{x_1}|0\rangle\langle 0|\mathsf{X}^{x_1}\Big)\otimes\cdots\\ \otimes\Big(\mathsf{P}^{h_t\cdot x_t}\mathsf{H}^{h_t}\mathsf{X}^{d_t}|0\rangle\langle 0|\mathsf{X}^{d_t}\mathsf{H}^{h_t}\mathsf{P}^{*h_t\cdot x_t}\otimes \mathsf{X}^{x_t}|0\rangle\langle 0|\mathsf{X}^{x_t}\Big)\otimes\\ \sum_{b_1\dots b_n\in\{0,1\}}\mathsf{Z}^{b_1}|+\rangle\langle +|\mathsf{Z}^{b_1}\otimes\cdots\otimes\mathsf{Z}^{b_n}|+\rangle\langle +|\mathsf{Z}^{b_n}\\ \otimes|d_1\dots d_t,x_1\dots x_t,b_1\dots b_n\rangle\langle d_1\dots d_t,x_1\dots x_t,b_1\dots b_n|\,.\end{aligned}\tag{7.15}$$

We claim that, replacing the P and P^* gates with the identity in Equation (7.14) and Equation (7.15), we obtain expressions for the system for each test run, respectively, at the *end* of the application of the honest unitary. This essentially follows by construction; for completeness we review the case of each gate gadget below.

CNOT-gate. As discussed in Section 4.2, in both the X-test and Z-test, a CNOT gate, when applied to the computation registers (the last n registers), will have no effect (up to a relabelling of the Paulis, as computed by the key update). Thus a simple change of variable reverts the system to an expression identical to its prior state.

H-gate. As given in Section 4.4, the application of the H gate to the computation registers will, up to a relabelling of the Paulis, cause $|0\rangle \mapsto |+\rangle$ and vice-versa. Since an even number of H gates are applied to each computation wire, the starting input state will not be changed by these gates.

T-gate as part of an X-test. Suppose a T-gate is applied in the X-test run, on qubit j, using an auxiliary qubit i $(i = 1 \dots t)$. Suppose furthermore that qubit j has undergone an *even* number of H gates, so that $h_i = 0$, and the system that the prover acts upon for the T-gate gadget, together with the relevant key register is

$$\sum_{d_i,x_i,a_j \in \{0,1\}} \mathsf{X}^{d_i}|0\rangle\langle 0|\mathsf{X}^{d_i} \otimes \mathsf{X}^{x_i}|0\rangle\langle 0|\mathsf{X}^{x_i} \otimes \mathsf{X}^{a_j}|0\rangle\langle 0|_j \mathsf{X}^{a_j} \otimes \left|d_i,x_i,a_j\right\rangle\left\langle d_i,x_i,a_j\right|. \tag{7.16}$$

Applying the P^x and CNOT as in the honest computation has very little effect on the system; it only causes a key update:

$$\sum_{d_i,x_i,a_j \in \{0,1\}} \mathsf{X}^{d_i}|0\rangle\langle 0|\mathsf{X}^{d_i} \otimes \mathsf{X}^{x_i}|0\rangle\langle 0|\mathsf{X}^{x_i} \otimes \mathsf{X}^{a_j\oplus d_i}|0\rangle\langle 0|_j \mathsf{X}^{a_j\oplus d_i} \otimes \left|d_i,x_i,a_j\oplus d_i\right\rangle\left\langle d_i,x_i,a_j\oplus d_i\right|. \tag{7.17}$$

As per our convention, we swap the first and last registers, so that the data wire remains in its position:

$$\sum_{d_i,x_i,a_j \in \{0,1\}} \mathsf{X}^{a_j\oplus d_i}|0\rangle\langle 0|\mathsf{X}^{a_j\oplus d_i} \otimes \mathsf{X}^{x_i}|0\rangle\langle 0|\mathsf{X}^{x_i} \otimes \mathsf{X}^{d_i}|0\rangle\langle 0|_j \mathsf{X}^{d_i} \otimes \left|a_j\oplus d_i,x_i,d_i\right\rangle\left\langle a_j\oplus d_i,x_i,d_i\right|. \tag{7.18}$$

Next, a change of variable shows that the expression is unchanged:

$$\sum_{d_i,x_i,a_j \in \{0,1\}} \mathsf{X}^{d_i}|0\rangle\langle 0|\mathsf{X}^{d_i} \otimes \mathsf{X}^{x_i}|0\rangle\langle 0|\mathsf{X}^{x_i} \otimes \mathsf{X}^{a_j}|0\rangle\langle 0|_j \mathsf{X}^{a_j} \otimes \left|d_i,x_i,a_j\right\rangle\left\langle d_i,x_i,a_j\right|. \tag{7.19}$$

T-gate as part of a Z-test. Suppose a T-gate is applied in the Z test run, on qubit j, using an auxiliary qubit i $(i = 1 \dots t)$. Suppose furthermore that qubit j has undergone an *even* number of H gates, so that $h_i = 1$, and the system that the prover acts upon for the T-gate gadget, together with the relevant key register is

$$\sum_{d_i,x_i,b_j \in \{0,1\}} \mathsf{P}^{x_i}\mathsf{Z}^{d_i}|+\rangle\langle +|\mathsf{Z}^{d_i}\mathsf{P}^{*x_i} \otimes \mathsf{X}^{x_i}|0\rangle\langle 0|\mathsf{X}^{x_i} \otimes \mathsf{Z}^{b_j}|+\rangle\langle +|_j \mathsf{Z}^{b_j} \otimes \left|d_i,x_i,b_j\right\rangle\left\langle d_i,x_i,b_j\right|. \tag{7.20}$$

Applying the P^x and CNOT as in the honest computation changes the system by canceling out the P_i^x and causing a key update:

$$\sum_{d_i,x_i,b_j \in \{0,1\}} \mathsf{Z}^{b_j\oplus d_i\oplus x_i}|+\rangle\langle +|\mathsf{Z}^{b_j\oplus d_i\oplus x_i} \otimes \mathsf{X}^{x_i}|0\rangle\langle 0|\mathsf{X}^{x_i} \otimes \mathsf{Z}^{b_j}|+\rangle\langle +|_j \mathsf{Z}^{b_j} \otimes \left|b_j\oplus d_i\oplus x_i,x_i,b_j\right\rangle\left\langle b_j\oplus d_i\oplus x_i,x_i,b_j\right|. \tag{7.21}$$

As per our convention, we swap the first and last registers, so that the data wire remains in the last position:

$$\sum_{d_i,x_i,b_j \in \{0,1\}} \mathsf{Z}^{b_j}|+\rangle\langle +|\mathsf{Z}^{b_j} \otimes \mathsf{X}^{x_i}|0\rangle\langle 0|\mathsf{X}^{x_i} \otimes \mathsf{Z}^{b_j\oplus d_i\oplus x_i}|+\rangle\langle +|_j \mathsf{Z}^{b_j\oplus d_i\oplus x_i} \otimes \left|b_j,x_i,b_j\oplus d_i\oplus x_i\right\rangle\left\langle b_j,x_i,b_j\oplus d_i\oplus x_i\right|. \tag{7.22}$$

Next, a change of variable shows that the expression is unchanged, except that the P and P^* gates are removed:

$$\sum_{d_i,x_i,b_j \in \{0,1\}} \mathsf{Z}^{d_i}|+\rangle\langle +|\mathsf{Z}^{d_i} \otimes \mathsf{X}^{x_i}|0\rangle\langle 0|\mathsf{X}^{x_i} \otimes \mathsf{Z}^{b_j}|+\rangle\langle +|_j \mathsf{Z}^{b_j} \otimes \left|d_i,x_i,b_j\right\rangle\left\langle d_i,x_i,b_j\right|. \tag{7.23}$$

Final expression before an attack. In the case that an *odd* number of H gates have been applied to a data wire, the protocol specifies that we should temporarily swap the roles of the X-test and Z-test runs for the T-gates that immediately follow. In this case, the data qubits and computation will be exactly those considered in the two cases above, but with the roles of the X-test and Z-test exchanged; the same analysis thus applies. For both the X-test and Z-test, we iteratively apply the various cases above (depending on the circuit). Since it is the case that all computation wires eventually have an *even* number of H-gates applied, we can write down an expression for the outcome for the X-test run:

$$\sum_{\substack{d_1 \dots d_t \in \{0,1\} \\ x_1 \dots x_t \in \{0,1\}}} \left(\mathsf{H}^{h_1} \mathsf{X}^{d_1} |0\rangle\langle 0| \mathsf{X}^{d_1} \mathsf{H}^{h_1} \otimes \mathsf{X}^{x_1} |0\rangle\langle 0| \mathsf{X}^{x_1} \right) \otimes \cdots \otimes \left(\mathsf{H}^{h_t} \mathsf{X}^{d_t} |0\rangle\langle 0| \mathsf{X}^{d_t} \mathsf{H}^{h_t} \otimes \mathsf{X}^{x_t} |0\rangle\langle 0| \mathsf{X}^{x_t} \right) \otimes \sum_{a_1 \dots a_n \in \{0,1\}} \mathsf{X}^{a_1} |0\rangle\langle 0| \mathsf{X}^{a_1} \otimes \cdots \otimes \mathsf{X}^{a_n} |0\rangle\langle 0| \mathsf{X}^{a_n} \otimes |d_1 \dots d_t, x_1 \dots x_t, a_1 \dots a_n\rangle\langle d_1 \dots d_t, x_1 \dots x_t, a_1 \dots a_n| . \tag{7.24}$$

In the case of a Z-test, an expression for the outcome is

$$\sum_{\substack{d_1 \dots d_t \in \{0,1\} \\ x_1 \dots x_t \in \{0,1\}}} \left(\mathsf{H}^{h_1} \mathsf{X}^{d_1} |0\rangle\langle 0| \mathsf{X}^{d_1} \mathsf{H}^{h_1} \otimes \mathsf{X}^{x_1} |0\rangle\langle 0| \mathsf{X}^{x_1} \right) \otimes \cdots \otimes \left(\mathsf{H}^{h_t} \mathsf{X}^{d_t} |0\rangle\langle 0| \mathsf{X}^{d_t} \mathsf{H}^{h_t} \otimes \mathsf{X}^{x_t} |0\rangle\langle 0| \mathsf{X}^{x_t} \right) \otimes \sum_{b_1 \dots b_n \in \{0,1\}} \mathsf{Z}^{b_1} |+\rangle\langle +| \mathsf{Z}^{b_1} \otimes \cdots \otimes \mathsf{Z}^{b_n} |+\rangle\langle +| \mathsf{Z}^{b_n} \otimes |d_1 \dots d_t, x_1 \dots x_t, b_1 \dots b_n\rangle\langle d_1 \dots d_t, x_1 \dots x_t, b_1 \dots b_n| . \tag{7.25}$$

Applying the attack, decryption and measurement. Next, we apply the attack for a fixed k, as given by

$$E_k = \sum_Q \alpha_Q Q \qquad \text{and} \qquad E_k^* = \sum_{Q'} \alpha_{Q'}^* Q' , \tag{7.26}$$

followed by the verifier's decryption, trace and measurement. For the registers that are traced out, we assume that they are decrypted and measured. Furthermore, since they are traced out, we can assume that the quantum auxiliary registers with $h_i = 1$ are measured in the diagonal basis. We let

$$Q = P_1 \otimes Q_1 \otimes P_2 \otimes Q_2 \otimes \cdots \otimes \mathsf{P}_t \otimes Q_t \otimes R_1 \otimes \cdots \otimes R_n , \tag{7.27}$$

with $P_i, Q_i, R_j \in \mathbb{P}_1$ ($i = 1 \dots t, j = 1 \dots n$) and similarly, let

$$Q' = P'_1 \otimes Q'_1 \otimes P'_2 \otimes Q'_2 \otimes \cdots \otimes P'_t \otimes Q'_t \otimes R'_1 \dots R'_n , \tag{7.28}$$

with $P'_i, Q'_i, R'_j \in \mathbb{P}_1$ ($i = 1 \dots t, j = 1 \dots n$).

For the X-test run, conditioned on outcomes i_ℓ (where $h_\ell = 0$), and $k_1,\ldots,k_n$ the system becomes

$$\sum_{\substack{(h_\ell=1\wedge\\ i_\ell\in\{0,1\}),\\ j_m\in\{0,1\}}} \sum_{Q,Q'} \alpha_Q\alpha^*_{Q'} \sum_{\substack{d_1\ldots d_t\in\{0,1\}\\ x_1\ldots x_t\in\{0,1\}}}$$
$$\Big(\langle i_1|\mathsf{X}^{d_1}\mathsf{H}^{h_1}P_1\mathsf{H}^{h_1}\mathsf{X}^{d_1}|0\rangle\langle 0|\mathsf{X}^{d_1}\mathsf{H}^{h_1}P'_1\mathsf{H}^{h_1}\mathsf{X}^{d_1}|i_1\rangle \otimes \langle j_1|\mathsf{X}^{x_1}Q_1\mathsf{X}^{x_1}|0\rangle\langle 0|\mathsf{X}^{x_1}Q'_1\mathsf{X}^{x_1}|j_1\rangle\Big)$$
$$\otimes\cdots\otimes$$
$$\Big(\langle i_t|\mathsf{X}^{d_t}\mathsf{H}^{h_t}P_t\mathsf{H}^{h_t}\mathsf{X}^{d_t}|0\rangle\langle 0|\mathsf{X}^{d_t}\mathsf{H}^{h_t}P'_t\mathsf{H}^{h_t}\mathsf{X}^{d_t}|i_t\rangle \otimes \langle j_t|\mathsf{X}^{x_t}Q_t\mathsf{X}^{x_t}|0\rangle\langle 0|\mathsf{X}^{x_t}Q'_t\mathsf{X}^{x_t}|j_t\rangle\Big)\otimes$$
$$\sum_{a_1\ldots a_n\in\{0,1\}} \langle k_1|\mathsf{X}^{a_1}R_1\mathsf{X}^{a_1}|0\rangle\langle 0|\mathsf{X}^{a_1}R'_1\mathsf{X}^{a_1}|k_1\rangle\otimes\cdots\otimes\langle k_n|\mathsf{X}^{a_n}R_n\mathsf{X}^{a_n}|0\rangle\langle 0|\mathsf{X}^{a_n}R'_n\mathsf{X}^{a_n}|k_n\rangle\,. \tag{7.29}$$

Applying the classical Pauli twirl (Lemmas 2.2 and 2.3), we obtain that the cross terms of the attack vanish, leaving as expression

$$\sum_{\substack{(h_\ell=1\wedge\\ i_\ell\in\{0,1\}),\\ j_m\in\{0,1\}}} \sum_{Q}|\alpha_Q|^2\Big(\langle i_1|\mathsf{H}^{h_1}P_1\mathsf{H}^{h_1}|0\rangle\langle 0|\mathsf{H}^{h_1}P_1\mathsf{H}^{h_1}|i_1\rangle\otimes\langle j_1|Q_1|0\rangle\langle 0|Q_1|j_1\rangle\Big)\otimes\cdots$$
$$\otimes\Big(\langle i_t|\mathsf{H}^{h_t}P_t\mathsf{H}^{h_t}|0\rangle\langle 0|\mathsf{X}^{d_t}\mathsf{H}^{h_t}P_t\mathsf{H}^{h_t}|i_t\rangle\otimes\langle j_t|Q_t|0\rangle\langle 0|Q_t|j_t\rangle\Big)\otimes$$
$$\langle k_1|R_1|0\rangle\langle 0|R_1|k_1\rangle\otimes\cdots\otimes\langle k_n|R_n|0\rangle\langle 0|R_n|k_n\rangle\,. \tag{7.30}$$

Recall that in an X-test run, the verifier rejects if a measurement result on an auxiliary qubit with $h_i = 0$ decrypts to the value 1, or if the output decrypts to the value 1. Thus, applying the above to all terms in $\{E_k\}$, we get that the probability that the verifier rejects is given by

$$\sum_k\sum_{Q\in B_1}|\alpha_{k,Q}|^2\,, \tag{7.31}$$

where B_1 is the set of $2t+n$-qubit Paulis with $P_i\in\{\mathsf{X},\mathsf{Y}\}$ $(i=1\ldots t)$ whenever $h_i=0$, or with $R_n\in\{\mathsf{X},\mathsf{Y}\}$.

A similar calculation shows that for the Z-test run, conditioned on outcomes i_ℓ (where $h_\ell = 1$), and $k_1,\ldots,k_n$ the system becomes

$$\sum_{\substack{(h_\ell=0\wedge\\ i_\ell\in\{0,1\}),\\ j_m\in\{0,1\}}} \sum_{Q}|\alpha_Q|^2\Big(\langle i_1|\mathsf{H}^{h_1}P_1\mathsf{H}^{h_1}|0\rangle\langle 0|\mathsf{H}^{h_1}P_1\mathsf{H}^{h_1}|i_1\rangle\otimes\langle j_1|Q_1|0\rangle\langle 0|Q_1|j_1\rangle\Big)\otimes\cdots$$
$$\otimes\Big(\langle i_t|\mathsf{H}^{h_t}P_t\mathsf{H}^{h_t}|0\rangle\langle 0|\mathsf{X}^{d_t}\mathsf{H}^{h_t}P_t\mathsf{H}^{h_t}|i_t\rangle\otimes\langle j_t|Q_t|0\rangle\langle 0|Q_t|j_t\rangle\Big)\otimes$$
$$\langle k_1|R_1|+\rangle\langle +|R_1|k_1\rangle\otimes\cdots\otimes\langle k_n|R_n|+\rangle\langle +|R_n|k_n\rangle\,. \tag{7.32}$$

Recall that in a Z-test run, the verifier rejects if a measurement result on an auxiliary qubit with $h_i = 0$ decrypts to the value 1. Thus, applying the above to all terms in $\{E_k\}$, we get that the probability that the verifier rejects is given by

$$\sum_k\sum_{Q\in B_2}|\alpha_{k,Q}|^2\,, \tag{7.33}$$

where B_2 is the set of $2t+n$-qubit Paulis with $P_i \in \{\mathsf{X},\mathsf{Y}\}$ $(i=1\dots t)$ whenever $h_i=0$.

Since each test is executed with probability $1/2$, and since $h_i=0$ in the X-test run if and only if $h_i=1$ in the Z-test run, we obtain that the probability that the verifier rejects is

$$\frac{1}{2}\sum_k \sum_{Q\in B'_{t,n}} |\alpha_{k,Q}|^2\,. \tag{7.34}$$

□

7.5 In the case of a computation run

Again using the preliminaries of Section 7.2 and Section 7.3, we now analyze soundness in the case of a computation run. First, we determine the effect of a bit flip on the measured qubit in the T-gate gadget (Section 7.5.1), then we do an analysis for the case that the computation run consists in a single T-gate gadget (Section 7.5.2). This is extended to full generality in Section 7.5.3, where we give a lower bound (as a function of the attack and of the underlying computation) on the probability that the verifier rejects in a computation run (see Lemma 7.8).

7.5.1 Effect of a bit flip on a measured qubit

In Lemma 7.5, we establish the effect of a bit flip on the measured qubit in the T-gate gadget.

Lemma 7.5. *The error induced by an* X*-gate on the measured qubit in the* T*-gate gadget in Figure 9 is to introduce an extra* XZP *on the output.*

Proof. An X-gate on the measured qubit in Figure 1 will cause the bottom wire to receive the correction $\mathsf{P}^{a\oplus c\oplus y\oplus 1}$ (instead of $\mathsf{P}^{a\oplus c\oplus y}$). Since $\mathsf{P}^{a\oplus c\oplus y\oplus 1}=\mathsf{P}\mathsf{Z}^{a\oplus c\oplus y}\mathsf{P}^{a\oplus c\oplus y}$, the following shows how we can use we use and revise the calculation from Equation (5.1)–Equation (5.8).

$$\begin{aligned}
\mathsf{P}^{a\oplus c\oplus y\oplus 1}\mathsf{X}^d\mathsf{Z}^e\mathsf{P}^y\mathsf{T}\mathsf{X}^{a\oplus c}\mathsf{Z}^b &= \mathsf{P}\mathsf{Z}^{a\oplus c\oplus y}\mathsf{P}^{a\oplus c\oplus y}\mathsf{X}^d\mathsf{Z}^e\mathsf{P}^y\mathsf{T}\mathsf{X}^{a\oplus c}\mathsf{Z}^b && (7.35)\\
&= \mathsf{P}\mathsf{Z}^{a\oplus c\oplus y}\mathsf{X}^{a\oplus c\oplus d}\mathsf{Z}^{(a\oplus c\oplus d)\cdot(d\oplus y)\oplus a\oplus b\oplus c\oplus d\oplus e\oplus y}\mathsf{T} && (7.36)\\
&= \mathsf{X}^{a\oplus c\oplus d}\mathsf{Z}^{(a\oplus c\oplus d)\cdot(d\oplus y)\oplus a\oplus b\oplus c\oplus d\oplus e\oplus y}\mathsf{Z}^{a\oplus c\oplus y}\mathsf{Z}^{a\oplus c\oplus d}\mathsf{P}\mathsf{T} && (7.37)\\
&= \mathsf{X}^{a\oplus c\oplus d}\mathsf{Z}^{(a\oplus c\oplus d)\cdot(d\oplus y)\oplus a\oplus b\oplus c\oplus e}\mathsf{P}\mathsf{T}\,. && (7.38)
\end{aligned}$$

We note furthermore that the X-gate on the measured qubit causes the Pauli key to be updated as $c\leftarrow c\oplus 1$. Starting with the right-hand side of Equation (7.38), we thus obtain

$$\mathsf{X}^{a\oplus c\oplus d\oplus 1}\mathsf{Z}^{(a\oplus c\oplus d\oplus 1)\cdot(d\oplus y)\oplus a\oplus b\oplus c\oplus e\oplus 1}\mathsf{P}\mathsf{T}=\mathsf{X}^{a\oplus c\oplus d}\mathsf{Z}^{(a\oplus c\oplus d)\cdot(d\oplus y)\oplus a\oplus b\oplus c\oplus d\oplus e\oplus y}(\mathsf{X}\mathsf{Z}\mathsf{P})\mathsf{T}\,. \tag{7.39}$$

Comparing with Equation (5.8), we thus see that the effect is to apply XZP. □

7.5.2 The T-gate protocol under attack

In this section, we analyze the effect of an attack in a single T-gate gadget: we show that the effect of the gadget on an encrypted qubit is to apply a T gate on the plaintext, while maintaining the encryption. Furthermore, if the auxiliary qubit undergoes an attack $Q_1\in\{\mathsf{X},\mathsf{Y}\}$, then an error $E=\mathsf{XZP}$ (by

Lemma 7.5) will be applied to the computation wire. The Pauli twirl plays again an important part in simplifying a general attack to a convex combination of Pauli attacks. This analysis (Lemma 7.6) is used as the inductive step in the proof of the general case as analysed in Section 7.5.3 (see Lemma 7.7).

Lemma 7.6. *In Interactive Proof System 2, consider a circuit consisting in a single* T*-gate, applied to a data wire which is initially an encryption of* ρ*. Consider a term in the attack* $\{E_k\}$*, given by* $Q = P_1 \otimes Q_1$*,* $Q' = P'_1 \otimes Q'_1$*, acting on the auxiliary qubits* ($P_1, Q_1, P'_1, Q'_1 \in \mathbb{P}_1$)*. Then in the case* $P_1 \otimes Q_1 \neq P'_1 \otimes Q'_1$*, this term simplifies to* 0*, whereas otherwise the effect is to apply* $E^{\delta_{P_1}}\mathsf{T}$ *on the data, while maintaining a uniform encryption, with the key held by the verifier. (Here,* $Q \in \mathbb{P}_1$*, and* $\delta_Q = 0$ *if* $Q \in \{I, \mathsf{Z}\}$ *and* $\delta_Q = 1$ *otherwise.)*

Proof. Suppose the honest circuit of the prover is applied, followed by an attack and the coherent correction of the verifier, which is then followed by a measurement of auxiliary qubits. We consider the effect of a Pauli attack $Q = P_1 \otimes Q_1$, $Q' = P'_1 \otimes Q'_1$, acting on the auxiliary qubits ($P_1, Q_1, P'_1, Q'_1 \in \mathbb{P}_1$). (Strictly speaking this is not a full attack, but instead, ignoring the coefficient, it corresponds to one term in the expansion of the full attack as given by $\{E_k\}$.) When a bit flip occurs on the top wire (i. e., when $P_1 \in \{\mathsf{X}, \mathsf{Y}\}$) in Figure 9, then the outcome undergoes an error $E = \mathsf{XZP}$ as given by Lemma 7.5. The above is summarized in Figure 12.

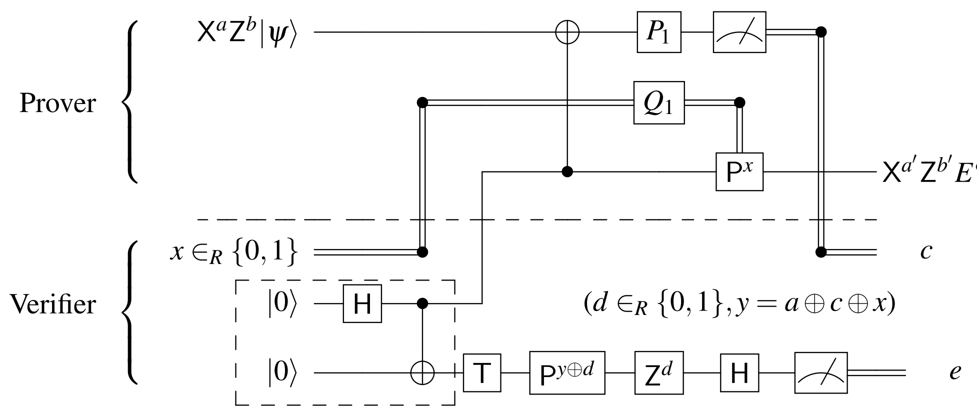

Figure 12: An attack $P_1 \otimes Q_1$ on a single T-gate gadget (computation run). Here, $a' = a \oplus c$ and $b' = (a \oplus c) \cdot (d \oplus y) \oplus a \oplus b \oplus c \oplus e \oplus y$.

In order to give a mathematical expression for Figure 12, we use the circuit identity in Figure 7 (according to which the measurement outcome c undergoes an encryption and decryption with d). First we consider the case $\delta_{P_1} = \delta_{P'_1}$. Applying $y = a \oplus c \oplus x$, and considering the trace of the auxiliary qubits, we get the following expression

$$\sum_{i,j \in \{0,1\}} \sum_{a,b,c,d,e,x \in \{0,1\}} \langle i|\mathsf{X}^d P_1 \mathsf{X}^d|c\rangle\langle c|\mathsf{X}^d P'_1 \mathsf{X}^d|i\rangle \otimes \langle j|\mathsf{X}^x Q_1 \mathsf{X}^x|0\rangle\langle 0|\mathsf{X}^x Q'_1 \mathsf{X}^x|j\rangle \otimes \\ \mathsf{X}^{a\oplus c}\mathsf{Z}^{(a\oplus c)\cdot(d\oplus x)\oplus a\oplus b\oplus c\oplus e\oplus x} E^{\delta_{P_1}} \mathsf{T}\rho\mathsf{T}^* {E^*}^{\delta_{P_1}} \mathsf{Z}^{(a\oplus c)\cdot(d\oplus x)\oplus a\oplus b\oplus c\oplus e\oplus x}\mathsf{X}^{a\oplus c} \\ \otimes |a\oplus c, (a\oplus c)\cdot(d\oplus x)\oplus a\oplus b\oplus c\oplus e\oplus x\rangle\langle a\oplus c, (a\oplus c)\cdot(d\oplus x)\oplus a\oplus b\oplus c\oplus e\oplus x| \,. \quad (7.40)$$

Note that above, we have included a key register for the computation wire, and have considered that the first register is traced out, thus since $\delta_{P_1} = \delta_{P'_1}$, cross terms of the form

$$\langle i|\mathsf{X}^d P_1 \mathsf{X}^d|c\rangle \langle c'|\mathsf{X}^d P'{}_1 \mathsf{X}^d|i\rangle \tag{7.41}$$

with $c \neq c'$ vanish and are therefore excluded. Furthermore, we consider without loss of generality that the second register is decrypted with X^x before being traced out.

Since a, b, and e appear only on the data register, and by a change of variable, we can rewrite this as

$$\sum_{i,j\in\{0,1\}} \sum_{c,d,x\in\{0,1\}} \langle i|\mathsf{X}^d P_1 \mathsf{X}^d|c\rangle\langle c|\mathsf{X}^d P'{}_1 \mathsf{X}^d|i\rangle \otimes \langle j|\mathsf{X}^x Q_1 \mathsf{X}^x|0\rangle\langle 0|\mathsf{X}^x Q'{}_1 \mathsf{X}^x|j\rangle \otimes \\ \sum_{a,b\in\{0,1\}} \mathsf{X}^a \mathsf{Z}^b E^{\delta_{P_1}} \mathsf{T}\rho \mathsf{T}^* E^{*\delta_{P_1}} \mathsf{Z}^b \mathsf{X}^a \otimes |a,b\rangle\langle a,b|\,. \tag{7.42}$$

Applying the classical Pauli twirl (Lemma 2.2), we get that for the case under consideration ($\delta_{P_1} = \delta_{P'_1}$), the system is 0 if $P_1 \otimes Q_1 \neq P'_1 \otimes Q'_1$, and otherwise is

$$\sum_{a,b\in\{0,1\}} \mathsf{X}^a \mathsf{Z}^b E^{\delta_{P_1}} \mathsf{T}\rho \mathsf{T}^* E^{*\delta_{P_1}} \mathsf{Z}^b \mathsf{X}^a \otimes |a,b\rangle\langle a,b|\,. \tag{7.43}$$

Next we consider the case $\delta_{P_1} \neq \delta_{P'_1}$. Again applying $y = a \oplus c \oplus x$, letting $c' = c \oplus 1$, and considering the trace of the auxiliary qubits, we get the following expression

$$\sum_{i,j\in\{0,1\}} \sum_{a,b,c,d,e,x\in\{0,1\}} \langle i|\mathsf{X}^d P_1 \mathsf{X}^d|c\rangle\langle c'|\mathsf{X}^d P'{}_1 \mathsf{X}^d|i\rangle \otimes \langle j|\mathsf{X}^x Q_1 \mathsf{X}^x|0\rangle\langle 0|\mathsf{X}^x Q'{}_1 \mathsf{X}^x|j\rangle \otimes \\ \xi \mathsf{X}^{a\oplus c} \mathsf{Z}^{(a\oplus c)\cdot(d\oplus x)\oplus a\oplus b\oplus c\oplus e\oplus x} E^{\delta_{P_1}} \mathsf{T}\rho \mathsf{T}^* E^{*\delta_{P'_1}} \mathsf{Z}^{(a\oplus c')\cdot(d\oplus x)\oplus a\oplus b\oplus c'\oplus e\oplus x} \mathsf{X}^{a\oplus c'} \\ \otimes |a\oplus c, (a\oplus c)\cdot(d\oplus x)\oplus a\oplus b\oplus c\oplus e\oplus x\rangle\langle a\oplus c', (a\oplus c')\cdot(d\oplus x)\oplus a\oplus b\oplus c'\oplus e\oplus x|\,. \tag{7.44}$$

Note that above, we have included a key register for the computation wire, and have considered that the first register is traced out. Thus terms of the form $\langle i|\mathsf{X}^d P_1 \mathsf{X}^d|c\rangle\langle c|\mathsf{X}^d P'{}_1 \mathsf{X}^d|i\rangle$ vanish and are therefore excluded. We again consider without loss of generality that the second register is decrypted with X^x before being traced out. Here, ξ represents a phase that depends on the key register and is due to the fact that $c \neq c'$.

Since $\delta_{P_1} = \delta_{P'_1} \oplus 1$, since a, b, and e appear only on the data register, and by the following change of variable,

$$a \leftarrow a \oplus c \tag{7.45}$$

$$b \leftarrow (a\oplus c)\cdot(d\oplus x)\oplus a\oplus b\oplus c\oplus e\oplus x \tag{7.46}$$

$$e \leftarrow (a\oplus c')\cdot(d\oplus x)\oplus a\oplus b\oplus c'\oplus e\oplus x\,, \tag{7.47}$$

we can rewrite this as

$$\sum_{i,j\in\{0,1\}} \sum_{c,d,x\in\{0,1\}} \langle i|\mathsf{X}^d P_1 \mathsf{X}^d|c\rangle\langle c|\mathsf{X}^d (\mathsf{X}P'{}_1) \mathsf{X}^d|i\rangle \otimes \langle j|\mathsf{X}^x Q_1 \mathsf{X}^x|0\rangle\langle 0|\mathsf{X}^x Q'{}_1 \mathsf{X}^x|j\rangle \otimes \\ \sum_{a,b,e\in\{0,1\}} \xi \mathsf{X}^a \mathsf{Z}^b E^{\delta_{P_1}} \mathsf{T}\rho \mathsf{T}^* E^{*\delta_{P_1}} \mathsf{Z}^e \mathsf{X}^{a\oplus 1} \otimes |a,b\rangle\langle a\oplus 1, e|\,. \tag{7.48}$$

Next, we apply the Pauli twirl to the first register, which yields 0 unless $P_1 = \mathsf{X}P'_1$, in which case we also get 0, since $\langle i|P_1|c\rangle\langle c|(\mathsf{X}P_1)|i\rangle = 0$.

Thus, we conclude that the case $P_1 \otimes Q_1 \neq P'_1 \otimes Q'_1$ evaluates to 0, whereas otherwise (by Equation (7.43)) the effect of the T-gate gadget is to apply $E^{\delta_{P_1}}\mathsf{T}$ on the data, while maintaining a uniform encryption, with the key held by the verifier. □

7.5.3 General analysis for a computation run

In this section, we bound the acceptance probability in the case of a computation run. This result is presented in Lemma 7.8, the proof of which depends on the following Lemma

Lemma 7.7. *Consider Interactive Proof System 2 for a circuit C on n qubits and with t T-gate gadgets, with attack $\{E_k\}$ (with each $E_k = \sum_Q \alpha_{k,Q} Q$). Suppose the target circuit C is decomposed as*

$$C = \mathsf{T}_{\ell_t} C_t \dots \mathsf{T}_{\ell_2} C_2 \mathsf{T}_{\ell_1} C_1 \,, \tag{7.49}$$

where each C_i is a Clifford group circuit and $\ell_i \in \{1 \dots n\}$ indicates that the i^{th} T-gate acts on qubit ℓ_i ($i = 1 \dots t$). Let

$$Q = P_1 \otimes Q_1 \otimes P_2 \otimes Q_2 \otimes \dots \otimes \mathsf{P}_t \otimes Q_t \,, \tag{7.50}$$

with $P_i, Q_i \in \mathbb{P}_1$ ($i = 1 \dots t$) and similarly, let

$$Q' = P'_1 \otimes Q'_1 \otimes P'_2 \otimes Q'_2 \otimes \dots \otimes P'_t \otimes Q'_t \,, \tag{7.51}$$

with $P'_i, Q'_i \in \mathbb{P}_1$ ($i = 1 \dots t$). Let E be the error on the output induced by an X on the top wire in Figure 9. For a Pauli $P \in \mathbb{P}_1$, let $\delta_P = 0$ if $P \in \{I, \mathsf{Z}\}$ and $\delta_P = 1$ otherwise. Then we claim that after the honest computation, attack, the verifier's conditional corrections and tracing out of the auxiliary system, the term of the system corresponding to the attack (Q, Q') is 0 if

$$P_1 \otimes Q_1 \otimes P_2 \otimes Q_2 \otimes \dots \otimes \mathsf{P}_t \otimes Q_t \neq P'_1 \otimes Q'_1 \otimes P'_2 \otimes Q'_2 \otimes \dots \otimes \mathsf{P}'_t \otimes Q'_t \,, \tag{7.52}$$

and otherwise is

$$\begin{aligned} \sum_{\substack{a_1,b_1,\dots a_n,b_n \\ \in\{0,1\}}} &(\mathsf{X}^{a_1}\mathsf{Z}^{b_1} \otimes \dots \otimes \mathsf{X}^{a_n}\mathsf{Z}^{b_n}) E_{\ell_t}^{\delta_{P_t}} \mathsf{T}_{\ell_t} C_t \dots E^{\delta_{P_2}}\ell_2 \mathsf{T}_{\ell_2} C_2 E_{\ell_1}^{\delta_{P_1}} \mathsf{T}_{\ell_1} C_1 |0^n\rangle \\ &\langle 0^n| C_1{}^* \mathsf{T}^*_{\ell_1} E^*{}_{\ell_1}^{\delta_{P_1}} C_2{}^* \mathsf{T}^*_{\ell_2} E^*{}_{\ell_2}^{\delta_{P_2}} \dots C_t{}^* \mathsf{T}^*_{\ell_t} E^*{}_{\ell_t}^{\delta_{P_t}} (\mathsf{Z}^{b_1}\mathsf{X}^{a_1} \otimes \dots \mathsf{Z}^{b_n}\mathsf{X}^{a_n}) \\ &\otimes |a_1, b_1, \dots, a_n, b_n\rangle\langle a_1, b_1, \dots, a_n, b_n| \,. \end{aligned} \tag{7.53}$$

Note that in the statement of Lemma 7.7 above, we have that Q and Q' are applied only to auxiliary qubits—we have thus omitted the "R" portion of the attack. Furthermore, we refer the reader to Figure 12 for the $t = 1$ case.

Proof. Lemma 7.7 is proven by induction on t. The base case $t=0$ is verified, since the initial system is

$$\sum_{\substack{a_1,b_1,\dots,a_n,b_n\\ \in\{0,1\}}} \mathsf{X}^{a_1}\mathsf{Z}^{b_1}\otimes\cdots\otimes\mathsf{X}^{a_n}\mathsf{Z}^{b_n}|0^n\rangle\langle 0^n|\mathsf{Z}^{b_1}\mathsf{X}^{a_1}\otimes\cdots\otimes\mathsf{Z}^{b_n}\mathsf{X}^{a_n}\otimes|a_1,b_1,\dots,a_n,b_n\rangle\langle a_1,b_1,\dots,a_n,b_n|\,. \tag{7.54}$$

Next, suppose Equation (7.53) holds for $t=k$ and consider $t=k+1$. Now, because each Pauli acts on a different subsystem, an attack

$$Q=P_1\otimes Q_1\otimes P_2\otimes Q_2\otimes\cdots\otimes\mathsf{P}_k\otimes Q_k\otimes\mathsf{P}_{k+1}\otimes Q_{k+1} \tag{7.55}$$

can be decomposed as an attack on the first $2k$ auxiliary qubits, followed by an attack of the last two qubits (and similarly for Q'). Suppose $P_i\otimes Q_i=P'_i\otimes Q'_i$ $(i=1\dots k)$. By the hypothesis, the outcome will be the application of the computation corresponding to the gadgets for C_{k+1} and $T_{\ell_{k+1}}$, followed by attack

$$(Q_{k+1}\otimes P_{k+1},Q'_{k+1}\otimes P'_{k+1})\,, \tag{7.56}$$

on an encrypted system ρ given by:

$$\rho=\sum_{\substack{a_1,b_1,\dots a_n,b_n\\ \in\{0,1\}}}(\mathsf{X}^{a_1}\mathsf{Z}^{b_1}\otimes\cdots\otimes\mathsf{X}^{a_n}\mathsf{Z}^{b_n})E_{\ell_k}^{\delta_{P_k}}\mathsf{T}_{\ell_k}C_k\dots E^{\delta_{P_2}}\ell_2\mathsf{T}_{\ell_2}C_2E_{\ell_1}^{\delta_{P_1}}\mathsf{T}_{\ell_1}C_1|0^n\rangle$$
$$\langle 0^n|C_1^*\mathsf{T}_{\ell_1}^*E^{*\delta_{P_1}}_{\ell_1}C_2^*\mathsf{T}_{\ell_2}^*E^{*\delta_{P_2}}_{\ell_2}\dots C_k^*\mathsf{T}_{\ell_k}^*E^{*\delta_{P_t}}_{\ell_k}(\mathsf{Z}^{b_1}\mathsf{X}^{a_1}\otimes\cdots\mathsf{Z}^{b_n}\mathsf{X}^{a_n})\otimes$$
$$|a_1,b_1,\dots,a_n,b_n\rangle\langle a_1,b_1,\dots,a_n,b_n|\,. \tag{7.57}$$

First, the prover applies a Clifford circuit C_{k+1}. After a key update as given in Section 4, a change of variable will lead to ρ':

$$\rho'=\sum_{\substack{a_1,b_1,\dots a_n,b_n\\ \in\{0,1\}}}(\mathsf{X}^{a_1}\mathsf{Z}^{b_1}\otimes\cdots\otimes\mathsf{X}^{a_n}\mathsf{Z}^{b_n})C_{k+1}E_{\ell_k}^{\delta_{P_k}}\mathsf{T}_{\ell_k}C_k\dots E^{\delta_{P_2}}\ell_2\mathsf{T}_{\ell_2}C_2E_{\ell_1}^{\delta_{P_1}}\mathsf{T}_{\ell_1}C_1|0^n\rangle$$
$$\langle 0^n|C_1{}^*\mathsf{T}_{\ell_1}^*E^{*\delta_{P_1}}_{\ell_1}C_2^*\mathsf{T}_{\ell_2}^*E^{*\delta_{P_2}}_{\ell_2}\dots C_k^*\mathsf{T}_{\ell_k}^*E^{*\delta_{P_t}}_{\ell_k}C_{k+1}^*(\mathsf{Z}^{b_1}\mathsf{X}^{a_1}\otimes\cdots\mathsf{Z}^{b_n}\mathsf{X}^{a_n})\otimes$$
$$|a_1,b_1,\dots,a_n,b_n\rangle\langle a_1,b_1,\dots,a_n,b_n|\,. \tag{7.58}$$

Next, the auxiliary qubits for $\mathsf{T}_{\ell_{k+1}}$ are used; we apply Lemma 7.6 (only a single qubit, ℓ_{k+1} is affected by this part of the computation). Thus, if

$$Q_{k+1}\otimes P_{k+1}\neq Q'_{k+1}\otimes P'_{k+1}\,, \tag{7.59}$$

the term vanishes, and otherwise it becomes

$$\sum_{\substack{a_1,b_1,\dots a_n,b_n\\ \in\{0,1\}}}(\mathsf{X}^{a_1}\mathsf{Z}^{b_1}\otimes\cdots\otimes\mathsf{X}^{a_n}\mathsf{Z}^{b_n})E_{\ell_{k+1}}^{\delta_{P_{k+1}}}\mathsf{T}_{\ell_{k+1}}C_{k+1}E_{\ell_k}^{\delta_{P_k}}\mathsf{T}_{\ell_k}C_k\dots E^{\delta_{P_2}}\ell_2\mathsf{T}_{\ell_2}C_2E_{\ell_1}^{\delta_{P_1}}\mathsf{T}_{\ell_1}C_1|0^n\rangle$$
$$\langle 0^n|C_1^*\mathsf{T}_{\ell_1}^*E^{*\delta_{P_1}}_{\ell_1}C_2^*\mathsf{T}_{\ell_2}^*E^{*\delta_{P_2}}_{\ell_2}\dots C_k^*\mathsf{T}_{\ell_k}^*E^{*\delta_{P_t}}_{\ell_k}C_{k+1}^*\mathsf{T}^*{}_{\ell_{k+1}}E^{*\delta_{P_{k+1}}}_{\ell_{k+1}}(\mathsf{Z}^{b_1}\mathsf{X}^{a_1}\otimes\cdots\mathsf{Z}^{b_n}\mathsf{X}^{a_n})\otimes$$
$$|a_1,b_1,\dots,a_n,b_n\rangle\langle a_1,b_1,\dots,a_n,b_n|\,. \tag{7.60}$$

By the hypothesis, the term also vanishes if

$$P_i \otimes Q_i \neq P_i' \otimes Q_i' \qquad (i = 1 \dots k)\,. \tag{7.61}$$

Thus, Lemma 7.7 holds for the case $t = k+1$ and by induction, Lemma 7.7 holds in general. □

We now state and prove the main result of this Section.

Lemma 7.8. *Consider the Interactive Proof System 2 for a circuit C on n qubits and with t T-gate gadgets, with attack $\{E_k\}$ (with each $E_k = \sum_Q \alpha_{k,Q} Q$). Let $B_{t,n}$ be the set of benign attacks. Then the probability that the verifier rejects for a computation run is at least:*

$$(1-p)\sum_k \sum_{Q \in B_{t,n}} |\alpha_{k,Q}|^2\,, \tag{7.62}$$

where

$$p = \|(|0\rangle\langle 0| \otimes \mathbb{I}_{n-1})C|0^n\rangle\|^2 \tag{7.63}$$

is the probability that we observe "0" as a results of a computational basis measurement of the n^{th} output qubit, obtained by evaluating C on input $|0^n\rangle$.

Proof. Let the notation be as in Lemma 7.7. We apply Lemma 7.7 to the case of the attack $\{E_k\}$. We denote each

$$E_k = \sum_{Q \otimes R} \alpha_{k,Q\otimes R} Q \otimes R\,, \tag{7.64}$$

where

$$Q = P_1 \otimes Q_1 \otimes P_2 \otimes Q_2 \otimes \cdots \otimes \mathsf{P}_t \otimes Q_t \tag{7.65}$$

(as before) and $R \in \mathbb{P}_n$. By linearity, we obtain the following system after the honest computation, attack, the verifier's conditional corrections and tracing out of the auxiliary system

$$\begin{aligned}\sum_{\substack{a_1,b_1,\dots a_n,b_n\\ \in\{0,1\}}} \sum_k \sum_{R,R'} \sum_Q \alpha_{k,Q\otimes R}\alpha^*_{k,Q\otimes R'} R(\mathsf{X}^{a_1}\mathsf{Z}^{b_1} \otimes \cdots \otimes \mathsf{X}^{a_n}\mathsf{Z}^{b_n})&\\ E_{\ell_t}^{\delta_{P_t}} \mathsf{T}_{\ell_t} C_t \dots E^{\delta_{P_2}}\ell_2 \mathsf{T}_{\ell_2} C_2 E_{\ell_1}^{\delta_{P_1}} \mathsf{T}_{\ell_1} C_1 |0^n\rangle\langle 0^n| C_1^* \mathsf{T}^*_{\ell_1} {E^*}_{\ell_1}^{\delta_{P_1}} C_2^* \mathsf{T}^*_{\ell_2} {E^*}_{\ell_2}^{\delta_{P_2}} \dots&\\ \dots C_t^* \mathsf{T}^*_{\ell_t} {E^*}_{\ell_t}^{\delta_{P_t}} (\mathsf{Z}^{b_1}\mathsf{X}^{a_1} \otimes \cdots \mathsf{Z}^{b_n}\mathsf{X}^{a_n}) R' \otimes&\\ |a_1,b_1,\dots,a_n,b_n\rangle\langle a_1,b_1,\dots,a_n,b_n|\,.&\end{aligned} \tag{7.66}$$

We can assume that the verifier then decrypts the system; by the Pauli twirl (Lemma 2.1), the terms with $R \neq R'$ vanish, leaving as the computation registers

$$\sum_k \sum_{Q,R} |\alpha_{k,Q\otimes R}|^2 R E_{\ell_t}^{\delta_{P_t}} \mathsf{T}_{\ell_t} C_t \dots E^{\delta_{P_2}}\ell_2 \mathsf{T}_{\ell_2} C_2 E_{\ell_1}^{\delta_{P_1}} \mathsf{T}_{\ell_1} C_1 |0^n\rangle\langle 0^n| C_1^* \mathsf{T}^*_{\ell_1} {E^*}_{\ell_1}^{\delta_{P_1}} C_2^* \mathsf{T}^*_{\ell_2} {E^*}_{\ell_2}^{\delta_{P_2}} \dots C_t^* \mathsf{T}^*_{\ell_t} {E^*}_{\ell_t}^{\delta_{P_t}} R\,. \tag{7.67}$$

Denoting $R = R_1 \otimes \cdots \otimes R_n$ ($R_i \in \mathbb{P}_1$), we see that each term with $Q \otimes R$ being benign leads to the output bit having the same distribution as in the honest case (Because for all i, benign attacks have $\delta_{P_i} = 0$, and also $R_n \in \{I, Z\}$ will have no effect on the output qubit that is measured.) In the honest case, p is the probability of observing "0" (which leads to the verifier accepting). In the case of the Pauli $Q \otimes R$ being *not* benign, we have no bound on the acceptance probability. Thus, with probability at least

$$(1-p)\sum_k \sum_{Q \in B_{t,n}} |\alpha_{k,Q}|^2 , \tag{7.68}$$

the verifier rejects. □

7.6 Proof of soundness

In order to complete the proof of soundness, we combine Lemma 7.4 and Lemma 7.8. Consider the Interactive Proof System 2 for a circuit C on n qubits and with t T-gate gadgets, with attack $\{E_k\}$, where

$$E_k = \sum_Q \alpha_{k,Q} Q . \tag{7.69}$$

Let $B_{t,n}$ be the set of benign Pauli attacks, and $B'_{t,n}$ be the set of non-benign Pauli attacks, and let p be as given in Lemma 7.8. Then since a test run occurs with probability $2/3$ and a computation run occurs with probability $1/3$, the probability that the verifier rejects is at least

$$\frac{2}{3} \cdot \frac{1}{2} \sum_k \sum_{Q \in B'_{t,n}} |\alpha_{k,Q}|^2 + \frac{1}{3}(1-p) \sum_k \sum_{Q \in B_{t,n}} |\alpha_{k,Q}|^2 . \tag{7.70}$$

Suppose that the input corresponds to a *no* instance of $\mathsf{Q\text{-}CIRCUIT}$. Hence, $1-p \geq 2/3$ and the probability that the verifier rejects is at least $2/9$ since

$$\begin{aligned}
&\frac{1}{3} \sum_k \sum_{Q \in B'_{t,n}} |\alpha_{k,Q}|^2 + \frac{1-p}{3} \sum_k \sum_{Q \in B_{t,n}} |\alpha_{k,Q}|^2 && (7.71)\\
&\geq \frac{1}{3} \sum_k \sum_{Q \in B'_{t,n}} |\alpha_{k,Q}|^2 + \frac{2}{9} \sum_k \sum_{Q \in B_{t,n}} |\alpha_{k,Q}|^2 && (7.72)\\
&= \frac{2}{9} \sum_k \sum_{Q \in \mathbb{P}_{2t+n}} |\alpha_{k,Q}|^2 + \frac{1}{9} \sum_k \sum_{Q \in B'_{t,n}} |\alpha_{k,Q}|^2 && (7.73)\\
&\geq \frac{2}{9} . && (7.74)
\end{aligned}$$

In Section 6, we saw that, in the case that the prover is honest, the probability c of acceptance of a *yes*-instance of $\mathsf{Q\text{-}CIRCUIT}$ satisfies $c \geq 8/9$ (completeness). We have determined above that for every prover, the probability s of acceptance of a *no*-instance satisfies $s \leq 1 - 2/9 = 7/9$ (soundness). Thus we have $c - s \geq 1/9$, which, by standard amplification, completes the proof of Theorem 3.3.

One consequence of the proof in this section is to note that the identity attack (i. e., the prover's honest behaviour) yields

$$\sum_k \sum_{Q \in B'_{t,n}} |\alpha_{k,Q}|^2 = 0 \tag{7.75}$$

and

$$\frac{1-p}{3}\sum_k \sum_{Q\in B_{t,n}} |\alpha_{k,Q}|^2 = \frac{1-p}{3}\,, \tag{7.76}$$

from which we conclude (Equation (7.71)) that the identity attack is an attack that minimizes the probability of rejection of a *no* instance.

Acknowledgements

I am grateful to Urmila Mahadev for suggesting the relabelling described in Section 6.1, which simplifies the proof of soundness, and also for related discussions. It is a pleasure to thank Gus Gutoski for many deep conversations, from which this work originated, and Thomas Vidick for many related discussions. Furthermore, it is a pleasure to thank Jacob Krich for supplying background material on quantum simulations. I am also grateful to Harry Buhrman, Evelyn Wainewright and to the anonymous referees for feedback on an earlier version of this work.

References

[1] SCOTT AARONSON AND ALEX ARKHIPOV: The computational complexity of linear optics. *Theory of Computing*, 9(4):143–252, 2013. Preliminary version in STOC'11. [doi:10.4086/toc.2013.v009a004, arXiv:1011.3245] 2

[2] SCOTT AARONSON AND ALEX ARKHIPOV: BosonSampling is far from uniform. *Quantum Inf. Comput.*, 14(15-16):1383–1423, 2014. [arXiv:1309.7460, ECCC:TR13-135] 2

[3] DORIT AHARONOV, MICHAEL BEN-OR, AND ELAD EBAN: Interactive proofs for quantum computations. In *Proc. 1st Symp. Innovations in Computer Science (ICS'10)*, pp. 453–469, 2010. [arXiv:1704.04487] 3, 4, 5, 9

[4] DORIT AHARONOV, MICHAEL BEN-OR, ELAD EBAN, AND URMILA MAHADEV: Interactive proofs for quantum computations, 2017. [arXiv:1704.04487] 3

[5] DORIT AHARONOV, VAUGHAN JONES, AND ZEPH LANDAU: A polynomial quantum algorithm for approximating the Jones polynomial. *Algorithmica*, 55(3):395–421, 2009. Preliminary version in STOC'06. [doi:10.1007/s00453-008-9168-0, arXiv:quant-ph/0511096] 2

[6] DORIT AHARONOV AND UMESH VAZIRANI: Is Quantum Mechanics falsifiable? A computational perspective on the foundations of Quantum Mechanics. In *Computability: Turing, Gödel, Church, and Beyond*, pp. 329–349. MIT Press, 2013. [arXiv:1206.3686] 2, 3

[7] ANDRIS AMBAINIS, MICHELE MOSCA, ALAIN TAPP, AND RONALD DE WOLF: Private quantum channels. In *Proc. 41st FOCS*, pp. 547–553. IEEE Comp. Soc. Press, 2000. [doi:10.1109/SFCS.2000.892142, arXiv:quant-ph/0003101] 7

[8] LÁSZLÓ BABAI: Trading group theory for randomness. In *Proc. 17th STOC*, pp. 421–429. ACM Press, 1985. [doi:10.1145/22145.22192] 8

[9] HOWARD BARNUM, CLAUDE CRÉPEAU, DANIEL GOTTESMAN, ADAM SMITH, AND ALAIN TAPP: Authentication of quantum messages. In *Proc. 43rd FOCS*, pp. 449–458. IEEE Comp. Soc. Press, 2002. [doi:10.1109/SFCS.2002.1181969, arXiv:quant-ph/0205128] 3

[10] STEFANIE BARZ, JOSEPH F. FITZSIMONS, ELHAM KASHEFI, AND PHILIP WALTHER: Experimental verification of quantum computation. *Nature Physics*, 9(11):727–731, 2013. [doi:10.1038/nphys2763, arXiv:1309.0005] 3

[11] MICHAEL BEN-OR, CLAUDE CRÉPEAU, DANIEL GOTTESMAN, AVINATAN HASSIDIM, AND ADAM SMITH: Secure multiparty quantum computation with (only) a strict honest majority. In *Proc. 47th FOCS*, pp. 249–260. IEEE Comp. Soc. Press, 2006. [doi:10.1109/FOCS.2006.68, arXiv:0801.1544] 3, 5

[12] P. OSCAR BOYKIN, TAL MOR, MATTHEW PULVER, VWANI ROYCHOWDHURY, AND FARROKH VATAN: A new universal and fault-tolerant quantum basis. *Inform. Process. Lett.*, 75(3):101–107, 2000. [doi:10.1016/S0020-0190(00)00084-3, arXiv:quant-ph/9906054] 5, 10

[13] GILLES BRASSARD, DAVID CHAUM, AND CLAUDE CRÉPEAU: Minimum disclosure proofs of knowledge. *J. Comput. System Sci.*, 37(2):156–189, 1988. [doi:10.1016/0022-0000(88)90005-0] 6

[14] ANNE BROADBENT: Delegating private quantum computations. *Canad. J. Physics*, 93(9):941–946, 2015. [doi:10.1139/cjp-2015-0030, arXiv:1506.01328] 5, 10

[15] ANNE BROADBENT, JOSEPH F. FITZSIMONS, AND ELHAM KASHEFI: Universal blind quantum computation. In *Proc. 50th FOCS*, pp. 517–526. IEEE Comp. Soc. Press, 2009. [doi:10.1109/FOCS.2009.36, arXiv:0807.4154] 3, 4

[16] ANNE BROADBENT, GUS GUTOSKI, AND DOUGLAS STEBILA: Quantum one-time programs. In *Proc. 33rd Ann. Intern. Cryptology Conf. (CRYPTO'13)*, pp. 344–360, 2013. [doi:10.1007/978-3-642-40084-1_20, arXiv:1211.1080] 3, 4, 5

[17] ANNE BROADBENT AND STACEY JEFFERY: Quantum homomorphic encryption for circuits of low T-gate complexity. In *Proc. 35th Ann. Intern. Cryptology Conf. (CRYPTO'15)*, pp. 609–629, 2015. [doi:10.1007/978-3-662-48000-7_30, arXiv:1412.8766] 5, 10

[18] ANDREW M. CHILDS, DEBBIE W. LEUNG, AND MICHAEL A. NIELSEN: Unified derivations of measurement-based schemes for quantum computation. *Phys. Rev. A*, 71(3):032318, 2005. [doi:10.1103/PhysRevA.71.032318, arXiv:quant-ph/0404132] 13

[19] ANDREA COLADANGELO, ALEX GRILO, STACEY JEFFERY, AND THOMAS VIDICK: Verifier-on-a-leash: new schemes for verifiable delegated quantum computation, with quasilinear resources, 2017. [arXiv:1708.07359] 4

[20] ANNE CONDON AND RICHARD E. LADNER: Interactive proof systems with polynomially bounded strategies. *J. Comput. System Sci.*, 50(3):506–518, 1995. Preliminary version in SCT'92. [doi:10.1109/SCT.1992.215403] 2

[21] IVAN B. DAMGÅRD, SERGE FEHR, LOUIS SALVAIL, AND CHRISTIAN SCHAFFNER: Cryptography in the bounded-quantum-storage model. *SIAM J. Comput.*, 37(6):1865–1890, 2008. Preliminary version in FOCS'05. [doi:10.1137/060651343, arXiv:quant-ph/0508222] 4

[22] CHRISTOPH DANKERT, RICHARD CLEVE, JOSEPH EMERSON, AND ETERA LIVINE: Exact and approximate unitary 2-designs and their application to fidelity estimation. *Phys. Rev. A*, 80(1):012304, 2009. [doi:10.1103/PhysRevA.80.012304, arXiv:quant-ph/0606161] 6, 7

[23] VEDRAN DUNJKO, JOSEPH F. FITZSIMONS, CHRISTOPHER PORTMANN, AND RENATO RENNER: Composable security of delegated quantum computation. In *20th Internat. Conf. on the Theory and Appl. of Cryptology and Information Security (ASIACRYPT'14)*, pp. 406–425, 2014. [doi:10.1007/978-3-662-45608-8_22, arXiv:1301.3662] 4

[24] RICHARD P. FEYNMAN: Simulating physics with computers. *Internat. J. Theoretical Physics*, 21(6-7):467–488, 1982. [doi:10.1007/BF02650179] 2

[25] KENT A. G. FISHER, ANNE BROADBENT, LYNDEN K. SHALM, ZHENNYA YAN, JONATHAN LAVOIE, ROBERT PREVEDEL, THOMAS JENNEWEIN, AND KEVIN J. RESCH: Quantum computing on encrypted data. *Nature Communications*, 5:3074, 2014. [doi:10.1038/ncomms4074, arXiv:1309.2586] 4, 5, 10

[26] JOSEPH F. FITZSIMONS AND MICHAL HAJDUŠEK: Post hoc verification of quantum computation, 2015. [arXiv:1512.04375] 4

[27] JOSEPH F. FITZSIMONS AND ELHAM KASHEFI: Unconditionally verifiable blind computation, 2012. [arXiv:1203.5217] 3, 4

[28] JOSEPH F. FITZSIMONS AND ELHAM KASHEFI: Unconditionally verifiable blind quantum computation. *Phys. Rev. A*, 96(1):012303, 2017. [doi:10.1103/PhysRevA.96.012303] 3

[29] ZHENGTING GAN AND ROBERT J. HARRISON: Calibrating quantum chemistry: A multi-teraflop, parallel-vector, full-configuration interaction program for the Cray-X1. In *18th Ann. Conf. on Supercomputing (SC'05)*, pp. 22–22, 2005. [doi:10.1109/SC.2005.17] 2

[30] CHRISTIAN GOGOLIN, MARTIN KLIESCH, LEANDRO AOLITA, AND JENS EISERT: Boson-Sampling in the light of sample complexity, 2013. [arXiv:1306.3995] 2

[31] SHAFI GOLDWASSER, YAEL TAUMAN KALAI, AND GUY N. ROTHBLUM: Delegating computation: Interactive proofs for muggles. *J. ACM*, 62(4):27:1–27:64, 2015. Preliminary version in STOC'08. [doi:10.1145/2699436, ECCC:TR17-108] 2

[32] SHAFI GOLDWASSER, SILVIO MICALI, AND CHARLES RACKOFF: The knowledge complexity of interactive proof systems. *SIAM J. Comput.*, 18(1):186–208, 1989. Preliminary version in STOC'85. [doi:10.1137/0218012] 8

[33] DANIEL GOTTESMAN AND ISAAC L. CHUANG: Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations. *Nature*, 402(6760):390–393, 1999. [doi:10.1038/46503, arXiv:quant-ph/9908010] 4

[34] MASAHITO HAYASHI AND MICHAL HAJDUŠEK: Self-guaranteed measurement-based quantum computation, 2016. [arXiv:1603.02195] 3

[35] MASAHITO HAYASHI AND TOMOYUKI MORIMAE: Verifiable measurement-only blind quantum computing with stabilizer testing. *Phys. Rev. Lett.*, 115(22):220502, 2015. [doi:10.1103/PhysRevLett.115.220502, arXiv:1505.07535] 3

[36] RAHUL JAIN, ZHENGFENG JI, SARVAGYA UPADHYAY, AND JOHN WATROUS: QIP = PSPACE. *Comm. ACM*, 53(12):102–109, 2010. Preliminary version in STOC'10. [doi:10.1145/1859204.1859231, arXiv:0907.4737] 6

[37] THEODOROS KAPOURNIOTIS, VEDRAN DUNJKO, AND ELHAM KASHEFI: On optimising quantum communications in verifiable quantum computing. In *Asian Quantum Info. Sci. Conf. (AQIS'15)*, pp. 23–25, 2015. [arXiv:1506.06943] 3

[38] ELHAM KASHEFI AND PETROS WALLDEN: Optimised resource construction for verifiable quantum computation. *J. Physics A*, 50(14):145306, 2017. [doi:10.1088/1751-8121/aa5dac, arXiv:1510.07408] 3

[39] JULIA KEMPE, HIROTADA KOBAYASHI, KEIJI MATSUMOTO, AND THOMAS VIDICK: Using entanglement in quantum multi-prover interactive proofs. *Comput. Complexity*, 18(2):273–307, 2009. Preliminary version in CCC'08. [doi:10.1007/s00037-009-0275-3, arXiv:0711.3715] 5

[40] CARSTEN LUND, LANCE FORTNOW, HOWARD KARLOFF, AND NOAM NISAN: Algebraic methods for interactive proof systems. *J. ACM*, 39(4):859–868, 1992. Preliminary version in FOCS'90. [doi:10.1145/146585.146605] 2

[41] MICHAEL A. NIELSEN AND ISSAC L. CHUANG: *Quantum Computation and Quantum Information*. Cambridge University Press, 2000. [doi:10.1017/CBO9780511976667] 7

[42] BEN W. REICHARDT, FALK UNGER, AND UMESH VAZIRANI: Classical command of quantum systems. *Nature*, 496(7446):456–460, 2013. Preliminary version in ITCS'13. [doi:10.1038/nature12035, arXiv:1209.0448, arXiv:1209.0449] 3

[43] ADI SHAMIR: IP = PSPACE. *J. ACM*, 39(4):869–877, 1992. Preliminary version in FOCS'90. [doi:10.1145/146585.146609] 2

[44] PETER W. SHOR AND JOHN PRESKILL: Simple proof of security of the BB84 quantum key distribution protocol. *Phys. Rev. Lett.*, 85(2):441–444, 2000. [doi:10.1103/physrevlett.85.441, arXiv:quant-ph/0003004] 4

[45] NICOLÒ SPAGNOLO, CHIARA VITELLI, MARCO BENTIVEGNA, DANIEL J BROD, ANDREA CRESPI, FULVIO FLAMINI, SANDRO GIACOMINI, GIORGIO MILANI, ROBERTA RAMPONI, PAOLO MATALONI, ROBERTO OSELLAME, ERNESTO F. GALVÃO, AND FABIO SCIARRINO: Experimental validation of photonic boson sampling. *Nature Photonics*, 8(8):615–620, 2014. [doi:10.1038/nphoton.2014.135, arXiv:1311.1622] 2

[46] JOHN WATROUS: PSPACE has constant-round quantum interactive proof systems. *Theoret. Comput. Sci.*, 292(3):575–588, 2003. Preliminary version in FOCS'99. [doi:10.1016/S0304-3975(01)00375-9, arXiv:cs/9901015] 9

[47] XINLAN ZHOU, DEBBIE W. LEUNG, AND ISAAC L. CHUANG: Methodology for quantum logic gate construction. *Phys. Rev. A*, 62(5):052316, 2000. [doi:10.1103/PhysRevA.62.052316, arXiv:quant-ph/0002039] 13

AUTHOR

Anne Broadbent
Associate professor
University of Ottawa
Ottawa, ON, Canada
abroadbe@uottawa.ca
http://mysite.science.uottawa.ca/abroadbe/

ABOUT THE AUTHOR

ANNE BROADBENT holds an undergraduate degree (2002) in Combinatorics and Optimization from the University of Waterloo. Her Master's (2004) and Ph. D. (2008) work were supervised by Gilles Brassard and Alain Tapp at the Département d'informatique et de recherche opérationnelle of Université de Montréal. In her spare time, she likes researching quantum cryptography beyond quantum key distribution.