SIAM J. Comput.
Vol. 49, No. 2, pp. 245–283

ZERO-KNOWLEDGE PROOF SYSTEMS FOR QMA[*]

ANNE BROADBENT[†], ZHENGFENG JI[‡], FANG SONG[§], and JOHN WATROUS[¶]

Abstract. Prior work has established that all problems in NP admit classical zero-knowledge proof systems, and under reasonable hardness assumptions for quantum computations, these proof systems can be made secure against quantum attacks. We prove a result representing a further quantum generalization of this fact, which is that every problem in the complexity class QMA has a quantum zero-knowledge proof system. More specifically, assuming the existence of an unconditionally binding and quantum computationally concealing commitment scheme, we prove that every problem in the complexity class QMA has a quantum interactive proof system that is zero-knowledge with respect to efficient quantum computations. Our QMA proof system is sound against arbitrary quantum provers, but only requires an honest prover to perform polynomial-time quantum computations, provided that it holds a quantum witness for a given instance of the QMA problem under consideration. The proof system relies on a new variant of the QMA-complete local Hamiltonian problem in which the local terms are described by Clifford operations and standard basis measurements. We believe that the QMA-completeness of this problem may have other uses in quantum complexity.

Key words. QMA, local-Hamiltonian problem, zero-knowledge, quantum computation

AMS subject classifications. 81P45, 81P68, 81P94

DOI. 10.1137/18M1193530

1. Introduction.

Zero-knowledge proof systems, which were first introduced by Goldwasser, Micali, and Rackoff [28], are interactive protocols that allow a prover to convince a verifier of the validity of a statement while revealing no additional information beyond the statement's validity. As paradoxical as it may seem upon a first consideration, several problems that are not known to be efficiently computable, such as the quadratic non-residuosity, graph isomorphism, and graph non-isomorphism problems, admit zero-knowledge proof systems [26, 28]. Under reasonable intractability assumptions, Goldreich, Micali, and Wigderson [26] gave a zero-knowledge protocol for the graph 3-coloring problem and, because of its NP-completeness, for all NP problems. This line of work was further extended in [7], which showed that all problems in IP have zero-knowledge proof systems.

Since the invention of this concept, zero-knowledge proof systems have become a cornerstone of modern theoretical cryptography. In addition to the conceptual inno-

[*] Received by the editors June 15, 2018; accepted for publication (in revised form) November 5, 2019; published electronically March 10, 2020. A preliminary version of this paper appeared in Proceedings of the 57th Annual Symposium on Foundations of Computer Science (FOCS 2016), IEEE, Piscataway, NJ, 2016, pp. 31–40.
https://doi.org/10.1137/18M1193530

Funding: The first, third, and fourth authors were are supported in part by Canada's NSERC. The third author was supported in part by Cryptoworks21, CIFAR, and the U.S. National Science Foundation.

[†] Department of Mathematics and Statistics, University of Ottawa, Ottawa, Canada (abroadbe@uottawa.ca).

[‡] Centre for Quantum Software and Information, School of Computer Science, University of Technology Sydney, Sydney, Australia, and State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing, China (jizhengfeng@gmail.com).

[§] Computer Science and Engineering Department, Texas A&M University, College Station, TX 77843 (fang.song@tamu.edu).

[¶] Institute for Quantum Computing and School of Computer Science, University of Waterloo, Waterloo, ON N2L 3G1, Canada, and Canadian Institute for Advanced Research, Toronto, ON MSG 1M1, Canada (watrous@uwaterloo.ca).

vation of a complexity-theoretic notion of knowledge, zero-knowledge proof systems are essential building blocks in a host of cryptographic constructions. One notable example is the design of secure two-party and multiparty computation protocols [25].

The extensive works on zero-knowledge largely reside in a classical world. The development of quantum information science and technology has urged another look at the landscape of zero-knowledge proof systems in a *quantum* world. Namely, both honest users and adversaries may potentially possess the capability to exchange and process quantum information. There are, of course, zero-knowledge protocols that immediately become insecure in the presence of quantum attacks due to efficient quantum algorithms that break the intractability assumptions upon which these protocols rely. For instance, Shor's quantum algorithms for factoring and computing discrete logarithms [49] invalidate the use of these problems, generally conjectured to be classically hard, as a basis for the security of zero-knowledge protocols against quantum attacks. Even with computational assumptions against quantum adversaries, however, it is still highly nontrivial to establish the security of classical zero-knowledge proof systems in the presence of malicious quantum verifiers because of a technical reason that we now briefly explain.

The zero-knowledge property of a proof system for a fixed input string is concerned with the computations that may be realized through an interaction between a (possibly malicious) verifier and the prover. That is, the malicious verifier may take an arbitrary additional input (usually called the *auxiliary input* to distinguish it from the input string to the proof system under consideration), interact with the prover in any way it sees fit, and produce an output that is representative of what it has learned through the interaction. Roughly speaking, the prover is said to be *zero-knowledge* on the fixed input string if any computation of the sort just described can be efficiently approximated[1] by a *simulator* operating entirely on its own—meaning that it does not interact with the prover, and in the case of an NP problem it does not possess a witness for the fixed problem instance being considered. The proof system is then said to be zero-knowledge when this zero-knowledge property holds for all yes-instances of the problem under consideration.

Classically, the zero-knowledge property is typically established through a technique known as *rewinding*. In essence, the simulator can store a copy of the auxiliary input, and it can make guesses and store intermediate states representing a hypothetical prover/verifier interaction—and if it makes a bad guess or otherwise experiences bad luck when simulating this hypothetical interaction, it simply reverts to an earlier stage (or back to the beginning) of the simulation and tries again. Indeed, it is generally the simulator's freedom to disregard the temporal restrictions of the actual prover/verifier interaction in a way such as this that makes it possible to succeed.

However, rewinding a quantum simulation is more problematic; the *no-cloning theorem* [60] forbids one from copying quantum information, making it impossible to store a copy of the input or of an intermediate state, and measurements generally have an irreversible effect [21] that may partially destroy quantum information. Such difficulties were first observed by van de Graaf [54] and further studied in [14, 55]. Later, a *quantum rewinding* technique was found [58] to establish that several interactive proof systems, including the Goldreich–Micali–Wigderson graph 3-coloring proof system [26], remain zero-knowledge against malicious quantum verifiers (un-

[1] Different notions of approximations are considered, including *statistical* approximations and *computational* approximations, which require that the simulator's computation is either statistically (or information-theoretically) indistinguishable or computationally indistinguishable from the malicious verifier's computation. This paper is primarily concerned with the computational variant.

der appropriate quantum intractability assumptions in some cases). It follows that all NP problems have zero-knowledge proof systems even against quantum malicious verifiers, provided that a quantum analogue of the intractability assumption required by the Goldreich–Micali–Wigderson graph 3-coloring proof system are in place.

This work studies the quantum analogue of NP, known as QMA, in the context of zero-knowledge. This is the class of problems having succinct quantum witnesses satisfying similar completeness and soundness conditions to NP (or its randomized variant MA). Quantum witnesses and verification are conjectured to be more powerful than their classical counterparts: there are problems that admit short quantum witnesses, whereas there is no known method for verification using a polynomial-sized classical witness. In other words, NP $\subseteq$ QMA holds trivially, and the containment is typically conjectured to be proper. The question we address in this paper is, *does every problem in* QMA *have a zero-knowledge quantum interactive proof system?* In more philosophical terms, viewing quantum witnesses as precious sources of knowledge, *can one always devise a proof system that reveals nothing about a quantum witness beyond its validity?*

1.1. Our contributions.

We answer the above question positively by constructing a quantum interactive proof system for any problem in QMA. It is zero-knowledge against any polynomial-time quantum adversary, under a reasonable quantum intractability assumption.

THEOREM 1.1. *Assuming the existence of an unconditionally binding and quantum computationally concealing bit commitment scheme, every problem in* QMA *has a quantum computational zero-knowledge proof system.*

A few of the desirable features of our proof system are as follows:

1. Our proof system has a simple structure, similar to the graph 3-coloring proof system of Goldreich–Micali–Wigderson (and to so-called Σ-protocols more generally). It can be viewed as a three-phase process: the prover commits to a quantum witness, the verifier makes a random challenge, and finally the prover responds to the challenge by partial opening of the committed information that suffices to certify the validity.
2. All communications in our proof system are classical except for the first commitment message, and the verifier can measure the quantum message immediately upon its arrival (which has a strong technological appeal).
3. Our protocol is based on plausible computational assumptions. The sort of bit commitment scheme it requires can be implemented, for instance, under the existence of injective one-way functions that are hard to invert in quantum polynomial time.
4. Our protocol is prover-efficient: given a valid quantum witness, an honest prover only needs to perform efficient quantum computations in order to convince the verifier to accept with high probability. Moreover, as has already been suggested, aside from the preparation of the first quantum message, all of the remaining computations performed by the honest prover are classical polynomial-time computations. No computational assumptions on the prover are required in the soundness case; the protocol is sound against arbitrary quantum provers.

As a key ingredient of our zero-knowledge proof system, we introduce a new variant of the k-local Hamiltonian problem and prove that it remains QMA-complete (with respect to Karp reductions). The k-local Hamiltonian problem asks if the minimum eigenvalue (or ground state energy in physics parlance) of an n-qubit Hamiltonian $H = \sum_j H_j$, where each H_j is k-local (i.e., acts trivially on all but k of the n qubits), is below a particular threshold value. This problem was introduced and proved to be QMA-complete by Kitaev, Shen, and Vyalyi [40] for the case $k = 5$, and subsequently improved to $k = 2$ [37]. We show (for the case $k = 5$) that each H_j can be restricted to be realized by a Clifford operation, followed by a standard basis measurement, and the QMA-completeness is preserved. Beyond its use in this paper, this fact has the potential to provide other insights into the study of quantum Hamiltonian complexity. For an arbitrary problem $A \in$ QMA, we can reduce an instance of A efficiently to an instance of the k-local Clifford Hamiltonian problem, and a valid witness for A can also be transformed into a witness for the corresponding k-local Clifford Hamiltonian problem instance by an efficient quantum procedure. As a result, A has a zero-knowledge proof system by composing this reduction with our zero-knowledge proof system for the k-local Clifford Hamiltonian

Our proof system also employs a new encoding scheme for quantum states, which we construct by extending the *trap scheme* proposed in [11]. While our new scheme can be seen as a *quantum authentication scheme* (cf. [2, 5, 6]), it in addition allows performing arbitrary constant-qubit Clifford circuits and measuring in the computational basis directly on authenticated data without the need for auxiliary states. The only previously known scheme supporting this feature requires high-dimensional quantum systems (i.e., qudits rather than qubits) [6], which make it inconvenient in our setting where all quantum operations are on qubits.

1.2. Overview of protocol and techniques.

A natural approach to constructing zero-knowledge proofs for QMA is to consider a quantum analogue of the Goldreich–Micali–Wigderson proof system for graph 3-coloring (which we will hereafter refer to as the GMW 3-coloring proof system), in which the prover commits to a 3-coloring of the input graph and reveals only the colors of the vertices corresponding to an edge randomly selected by the verifier. Let us focus in particular on the local Hamiltonian problem, and consider a proof system in which the prover holds a quantum witness state for an instance of this problem, commits to this witness, and receives the challenge from the verifier (which, let us say, is a randomly chosen term of the local Hamiltonian). The prover might then open the commitments of the set of qubits on which the term acts nontrivially so that the verifier can measure the local energy for this term and determine acceptance accordingly.

There is a major difficulty when one attempts to carry out such an approach for QMA. The zero-knowledge property of the GMW 3-coloring proof system depends crucially on a structural property of the problem: the honest prover is free to randomize the three colors used in its coloring, and when the commitments to the colors of two neighboring vertices are revealed, the verifier will see just a uniform mixture over all pairs of different colors. This uniformity of the coloring marginals is important in achieving the zero-knowledge property of the proof system. Unlike the case of 3-coloring, however, none of the known QMA-complete problems under Karp reductions has such desirable properties. For example, if we use local Hamiltonian problems directly in a GMW-type proof system, of the sort suggested above, information about the reduced state of the quantum witness will be leaked to the verifier, possibly violating the zero-knowledge requirement.

To overcome the difficulty suggested above, we employ several ideas that enable the prover to "partially" open the commitments, revealing only the fact that the committed state is supported on certain subspaces. Our first technique simplifies the verification circuit for QMA-complete problems through the introduction of the local Clifford–Hamiltonian problem that was already described. More specifically, our formulation of this problem requires every Hamiltonian term to take the form $C^*|0^k\rangle\langle 0^k|C$ for some Clifford operation C. Because the local Clifford–Hamiltonian problem remains QMA-complete, it implies a random Clifford verification procedure for problems in QMA: intuitively, the verification of a quantum witness has been simplified to a Clifford measurement followed by a classical verification.

The Clifford verification procedure works in harmony with the encryption of quantum data via the quantum one-time pad and other derived hybrid schemes that are used by our proof system. This has the important effect of transforming statements about quantum states into statements about the classical keys of the quantum one-time pad, which naturally leads to our second main idea: the use of zero-knowledge proofs for NP against quantum attacks to simplify the construction of zero-knowledge proof systems for QMA. In our protocol, the verifier measures the encrypted quantum data and asks the prover to prove, using a zero-knowledge protocol for NP, that the decryption of this result is consistent with the verifier accepting.

In fact, if the verifier measures the quantum data according to the specifications of the protocol, the combination of the Clifford verification and the use of zero-knowledge proofs for NP suffices. A problem arises, however, if the verifier does not perform the honest measurement. Our third technique, inspired by work on quantum authentication [2, 6, 11, 17], employs a new scheme for encoding quantum states. Roughly speaking, if the prover encodes a witness state under our encoding scheme, then the verifier is essentially forced to perform the measurement honestly—any attempt to fake a “logically different” measurement result will succeed with negligible probability. In our proof system, we adapt the trap scheme proposed in [11] so that we can perform any constant-sized Clifford operations on authenticated quantum data followed by computational basis measurements, benefiting along the way from ideas concerning quantum computation on authenticated quantum data.

The resulting zero-knowledge proof system for QMA has a similar overall structure to the GMW 3-coloring protocol: the prover encodes the quantum witness state using a quantum authentication scheme, and sends the encoded quantum data together with a commitment to the secret keys of the authentication to the verifier. The verifier randomly samples a term $C^*|0^k\rangle\langle 0^k|C$ in the local Clifford–Hamiltonian problem, applies the operation C transversally on the encoded quantum data, and measures all qubits corresponding to the k qubits of the selected term in the computational basis, and sends the measurement outcomes to the prover. The prover and verifier then invoke a quantum-secure zero-knowledge proof for the NP statement that the commitment correctly encodes an authentication key and, under this key, the verifier's measurement outcomes do not decode to 0^k.

1.3. Comparisons to related work.

There has been other work on quantum complexity and theoretical cryptography, some of which is discussed below, that allows one to conclude statements having some similarity to our results. We will argue, however, that with respect to the problem of devising zero-knowledge quantum interactive proof systems for QMA, our main result is stronger in almost all respects. In addition, we believe that our proof system is appealing both because it is conceptually simple and represents a natural extension of well-known classical methods.

1. *Zero-knowledge proofs for all of* IP. Hallgren et al. [32] proved that, under certain technical conditions, any classical zero-knowledge proof system can be made secure against malicious quantum verifiers. A well-known result

of Ben-Or et al. [7] establishes that any problem in IP has a classical zero-knowledge protocol under a suitable cryptographic assumption. Although we have not verified that this zero-knowledge protocol for IP satisfies the technical conditions required by Hallgren et al. [32], we suspect that this is the case, assuming the existence of a quantum computationally hiding commitment scheme. If indeed this is so, it implies the existence of a classical protocol that is zero-knowledge against malicious quantum verifiers for all IP and, hence, for QMA because QMA is contained in IP. However, this generic protocol requires a computationally *unbounded* prover to carry out the honest protocol, and it is unlikely to allow for a reduced round complexity (e.g., constant round with constant soundness error) without causing unexpected consequences in complexity theory [27, 29, 56].

2. *Secure two-party computations.* One alternative approach to constructing zero-knowledge proof systems for QMA is to apply the general tool of secure two-party quantum computation [6, 16, 17]. In particular, we may imagine two parties, a prover and a verifier, jointly evaluating the verification circuit of a QMA problem, with the prover holding a quantum witness as its private input. In principle, one can design a two-party computation protocol so that the verifier learns the validity of the statement but nothing more about the prover's private input. While we believe that a careful analysis could make this approach work, it comes at a steep cost. First, we need to make significantly stronger computational assumptions, as secure quantum two-party computation relies on (at least) secure computations of classical functions against quantum adversaries. The best-known quantum-secure protocols for classical two-party computation assume quantum-secure dense public-key encryption [33] or similar primitives [42], in contrast to the existence of a quantum computationally hiding commitment scheme.[2] Second, the protocol obtained this way is an *argument* system. That is, the protocol is sound only against computationally bounded dishonest provers. Moreover, the generic quantum two-party computation protocol evaluates the verification circuit gate by gate and, in particular, interactions are unavoidable for some (non-Clifford) gates. This causes the round complexity to grow in proportion to the size of the verification circuit. In addition, the communications are inherently quantum, which makes the protocol much more demanding from a technological viewpoint.

   On the positive side, through this approach, it is possible to achieve a negligible soundness error using just one copy of the witness state. In contrast, our proof system directly inherits the soundness error of the most natural and direct verification for the local Clifford–Hamiltonian problem (i.e., randomly select a Hamiltonian term and measure). If one reduces an arbitrary QMA-verification procedure to an instance of this problem, the resulting soundness guarantee will generally be weakened by this reduction.

3. *Zero-knowledge proofs for density matrix consistency.* It was pointed out by Liu [41] that the density matrix consistency problem, which asks if there exists a global state of n qubits that is consistent with a collection of k-qubit density matrix marginals, should admit a simple zero-knowledge proof system following the GMW 3-coloring approach. (See also [13] for further details

[2] Roughly speaking, this distinction is analogous to "cryptomania" vs "minicrypt" according to Impagliazzo's five-world paradigm [35].

regarding this claim.) While it approaches our main result, it does not necessarily admit a zero-knowledge proof system for all problems in QMA, as the density matrix consistency problem is only known to be hard for QMA with respect to Cook reductions.

4. *Verification for* QMA*, nonlocal games, and follow-up work.* We note that Clifford verification with classical postprocessing of QMA was considered in [44] using magic states as ancillary resources. Our construction is arguably simpler, uses only constant-size Clifford operations and, most importantly, does not require any resource states. This helps one to avoid checking the correctness of resource states in the final zero-knowledge protocol. Our Clifford–Hamiltonian verification also finds applications in offering an alternative proof of the single-qubit measurement verification for QMA in [45], as well as in the study of nonlocal games [20, 36]. Moreover, following a previous version of this work [12], Vidick and Zhang [52] showed that our techniques can be applied to the conceptually simple QMA-complete "XY Hamiltonian problem" [19]. They obtain a *classical* zero-knowledge *argument* system for QMA, at the cost of sacrificing perfect completeness, and assuming the prover is given polynomially many copies of the witness state.

Organization. Section 2 summarizes notation, definitions, and primitives that are used for the construction of our zero-knowledge proof system. Section 3 describes the variant of the local Hamiltonian problem mentioned above. We present our zero-knowledge proof system for QMA in section 4 and prove its completeness and soundness in section 5 and zero-knowledge property in section 6. We conclude with some remarks and future directions in section 7.

2. Preliminaries.

This section summarizes some of the notation, definitions, and known facts concerning quantum information and computation, cryptography, and other topics that are used throughout the paper. We refer to [40, 48, 57] for further details on the theory of quantum information and computation. Further information on classical zero-knowledge and cryptography can be found in [22, 23].

2.1. Basic terminology.

Throughout the paper we let $\Sigma = \{0, 1\}$ denote the binary alphabet, and only consider strings, promise problems, and complexity classes over this alphabet. For a string $x \in \Sigma^*$, $|x|$ denotes its length. A function $g : \mathbb{N} \to \mathbb{N}$ is a *polynomially bounded function* if there exists a deterministic polynomial-time Turing machine M_g that outputs $1^{g(n)}$ on input 1^n for every nonnegative integer n. A function $f : \mathbb{N} \to [0, \infty)$ is said to be *negligible* if, for every polynomially bounded function g, it holds that $f(n) < 1/g(n)$ for all but finitely many values of n.

2.2. Quantum information basics.

When we refer to a *quantum register* in this paper, we simply mean a collection of qubits that we wish to view as a single unit and to which we give some name. Names of registers will always be uppercase letters in a *sans serif* font, such as X, Y, and Z. The finite-dimensional complex Hilbert spaces associated with registers will be denoted by capital script letters such as $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{Z}$, using the same letter in the two different fonts to denote a quantum register and its corresponding space for convenience. Dirac notation is used to express vectors in Hilbert spaces and linear mappings between them in a standard way.

For a given space $\mathcal{X}$, we let $\mathrm{L}(\mathcal{X})$ denote the set of all linear mappings (or *operators*) from $\mathcal{X}$ to itself. The identity element of $\mathrm{L}(\mathcal{X})$ is denoted $\mathbb{1}_{\mathcal{X}}$, or just as $\mathbb{1}$ when $\mathcal{X}$ can be taken as implicit. The inner product between operators A and B is defined as $\langle A, B\rangle = \mathrm{Tr}(A^* B)$.

Quantum states are represented by density operators, which are positive semidefinite operators having unit trace. Under the assumption that $\mathcal{X}$ corresponds to n qubits, a linear map $\Phi : \mathrm{L}(\mathcal{X}) \to \mathrm{L}(\mathcal{Y})$ is *completely positive* if and only if its Choi operator

$$J(\Phi) = \sum_{x,y \in \Sigma^n} \Phi\big(|x\rangle\langle y|\big) \otimes |x\rangle\langle y| \tag{2.1}$$

is positive semidefinite, and Φ is said to be a *channel* if it is both completely positive and preserves trace. Channels are mappings from density operators to density operators that, in principle, represent physically realizable operations. A *measurement* is described by a collection of positive semidefinite operators $\{M_j\}$ such that $\sum_j M_j = \mathbb{1}$, with the probability that the measurement on state ρ results in outcome j being given by $\langle M_j, \rho\rangle$.

We review a few definitions of norms on operators, which are used to discuss the distinguishability of quantum states and channels. The *trace norm* of an operator $X \in \mathrm{L}(\mathcal{X})$ is defined as $\|X\|_1 = \mathrm{Tr}\sqrt{X^*X}$. For any linear map $\Phi : \mathrm{L}(\mathcal{X}) \to \mathrm{L}(\mathcal{Y})$, the *diamond norm* (or completely bounded trace norm) [3, 39, 40] is defined as

$$\|\Phi\|_\diamond = \max\left\{\|(\Phi \otimes \mathbb{1}_{\mathrm{L}(\mathcal{W})})(X)\|_1 \,:\, X \in \mathrm{L}(\mathcal{X} \otimes \mathcal{W})\,,\ \|X\|_1 \le 1\right\},$$

where $\mathcal{W}$ is any space with dimension equal to that of $\mathcal{X}$. (The value remains the same for any choice of $\mathcal{W}$, provided its dimension is at least that of $\mathcal{X}$.)

Quantum gates and circuits. A *quantum circuit* is an acyclic network of quantum gates connected by wires. The quantum gates represent quantum channels while the wires represent qubits on which the channels act.

We will refer to two types of quantum circuits in this paper: *unitary* quantum circuits and *general* quantum circuits. By unitary quantum circuits we mean circuits composed of unitary gates (such as the ones described below) chosen from some finite gate set. General quantum circuits are composed of gates that may correspond to channels that are not necessarily unitary. It is sufficient for the purposes of this paper that we consider just two simple nonunitary gates: *ancillary gates*, which input nothing and output a qubit in the $|0\rangle$ state; and *erasure gates*, which input one qubit and output nothing (and correspond to the channel described by the trace mapping). As is described elsewhere [3, 59], arbitrary channels mapping one register to another can always be approximated arbitrarily closely by quantum circuits whose gates include a universal collection of unitary gates together with ancillary and erasure gates. The *size* of a quantum circuit is the number of gates in the circuit plus the number of qubits on which it acts. We will refer specifically to the following well-known single-qubit unitary gates:

1. *Pauli gates:*

$$X : |a\rangle \mapsto |1-a\rangle \qquad \text{and} \qquad Z : |a\rangle \mapsto (-1)^a|a\rangle \tag{2.2}$$

for each $a \in \Sigma$, as well as $Y = iXZ$.

2. *Hadamard gate:*

$$H : |a\rangle \mapsto \frac{1}{\sqrt{2}}|0\rangle + \frac{(-1)^a}{\sqrt{2}}|1\rangle \tag{2.3}$$

for each $a \in \Sigma$.

3. *Phase gate:*

$$P : |a\rangle \mapsto i^a|a\rangle \tag{2.4}$$

for each $a \in \Sigma$.

In addition, for any k-qubit unitary quantum gate U we define the *controlled-U* gate as

$$\Lambda(U) : |a\rangle|x\rangle \mapsto |a\rangle U^a|x\rangle \tag{2.5}$$

for each $a \in \Sigma$ and $x \in \Sigma^k$.

The k-qubit *Pauli group* is the group containing all unitary operators of the form

$$\alpha U_1 \otimes \cdots \otimes U_k, \tag{2.6}$$

where $\alpha \in \{1, i, -1, -i\}$ and $U_1, \ldots, U_k \in \{\mathbb{1}, X, Y, Z\}$, where $\mathbb{1}$ denotes the single-qubit identity operation. Elements of this group are also referred to as *Pauli operations.* If $a, b \in \Sigma^k$ are binary strings of length k, then we write

$$X^a = X^{a_1} \otimes \cdots \otimes X^{a_k} \quad \text{and} \quad Z^b = Z^{b_1} \otimes \cdots \otimes Z^{b_k} \tag{2.7}$$

to denote the Pauli operations obtained from these strings as indicated.

Channels that can be expressed as convex combinations of unitary channels that correspond to Pauli operations are called *Pauli channels*. An example of a Pauli channel that is relevant to this paper is the *completely depolarizing* channel

$$\Omega(\rho) = \frac{1}{4}\sum_{a,b\in\Sigma} \big(X^a Z^b\big)\rho\big(X^a Z^b\big)^* = \frac{\mathbb{1}}{2} \tag{2.8}$$

for any single-qubit density operator ρ. We thus see that the effect of Ω is to completely randomize the state of a single-qubit system. By treating a random choice of a pair (a, b) as a secret key, we obtain a quantum generalization of the one-time pad, known as the *quantum one-time pad* [4]. When the channel is performed independently on k qubits, the effect is given by

$$\Omega^{\otimes k}(\rho) = 2^{-k}\, \mathbb{1} \otimes \cdots \otimes \mathbb{1} \tag{2.9}$$

for every k-qubit density operator ρ. The quantum one-time pad generalizes naturally to any choice of the number k.

Sometimes it will be convenient to consider quantum circuits that implement measurements. When we refer to a *measurement circuit*, we mean any general quantum circuit, followed by a measurement of all of its output qubits with respect to the standard basis. If Q is a measurement circuit that is applied to a collection of qubits in the state ρ, then $Q(\rho)$ is interpreted as a string-valued random variable describing the resulting measurement. We will only need to refer to measurement circuits outputting a single bit in this paper.

A k-qubit *Clifford circuit* is any unitary quantum circuit on k qubits whose gates are drawn from the set $\{H, P, \Lambda(X)\}$ containing Hadamard, phase, and controlled-not gates. (It is common that one also allows Pauli gates to be included in this set for convenience. Given that $X = HPPH$ and $Z = PP$, there is no generality lost in using the smaller gate set in the definition.) The set of all unitary operators that can be described by k-qubit Clifford circuits forms a finite group known as the *Clifford group.*

Up to scalar multiples, the k-qubit Clifford group is the normalizer of the k-qubit Pauli group: if U is a k-qubit unitary operator for which it holds that UVU^* is an element of the k-qubit Pauli group for every k-qubit Pauli group element V, then $U = \alpha C$ for $\alpha \in \mathbb{C}$ satisfying $|\alpha| = 1$ and C being a k-qubit Clifford group element. Given the description of a k-qubit Pauli group element V and a k-qubit Clifford circuit C, one can efficiently compute a description of the k-qubit Pauli group element CVC^* [31].

Clifford circuits are not universal for quantum computation. Two examples (among other known examples) of universal gate sets are the following:

1. Hadamard, phase, and Toffoli gates: $\{H, P, \Lambda(\Lambda(X))\}$.
2. Hadamard and controlled-phase gates: $\{H, \Lambda(P)\}$.

The first of these choices is sometimes easier to work with, but we will make use of the fact that the second gate set is universal in this work.

2.3. Polynomial-time generated families of quantum circuits and QMA.

Any quantum circuit with gates drawn from a fixed, finite gate set can be encoded as a binary string, with respect to a variety of possible encoding schemes. The specific details of such encoding schemes are not important within the context of this paper, so we will leave it to the reader to imagine that a sensible and efficient encoding scheme for quantum circuits has been selected, relative to whatever gate set is under consideration. It should be assumed, of course, that a circuit's size and its encoding length are polynomially related.

For any infinite set of binary strings $S \subseteq \Sigma^*$, a collection $\{V_x : x \in S\}$ of quantum circuits is said to be *polynomial-time generated* if there exists a deterministic polynomial-time Turing machine that, on input $x \in S$, outputs an encoding of V_x. The assumptions on encoding schemes suggested above imply that, if $\{V_x : x \in S\}$ is a polynomial-time generated collection, then V_x must have size polynomial in $|x|$.

Next we will define the complexity class QMA, which is commonly viewed as the most natural quantum generalization of NP.

Definition 2.1. *A promise problem $A = (A_{yes}, A_{no})$ is contained in the complexity class* $\mathrm{QMA}_{\alpha,\beta}$ *if there exists a polynomial-time generated collection*

$$\big\{V_x : x \in A_{yes} \cup A_{no}\big\} \tag{2.10}$$

of quantum circuits and a polynomially bounded function p possessing the following properties:

1. *For every string $x \in A_{yes} \cup A_{no}$, one has that V_x is a measurement circuit taking $p(|x|)$ input qubits and outputting a single bit.*
2. Completeness. *For all $x \in A_{yes}$, there exists a $p(|x|)$-qubit state ρ such that* $\Pr(V_x(\rho) = 1) \geq \alpha$.
3. Soundness. *For all $x \in A_{no}$ and all $p(|x|)$-qubit states ρ,* $\Pr(V_x(\rho) = 1) \leq \beta$.

In this definition, $\alpha, \beta \in [0, 1]$ may be constant values or functions of the length of the input string x. When they are omitted, it is to be assumed that they are $\alpha = 2/3$ and $\beta = 1/3$. Known error reduction methods [40, 43, 46] imply that a wide range of selections of α and β give rise to the same complexity class. In particular, QMA coincides with $\mathrm{QMA}_{\alpha,\beta}$ for $\alpha = 1 - 2^{-q(|x|)}$ and $\beta = 2^{-q(|x|)}$ for any polynomially bounded function q.

2.4. Quantum computational indistinguishability and zero-knowledge.

Next we review notions of quantum state and channel discrimination, as well as zero-knowledge in a quantum setting (as defined in [58]).

We first specify what it means for two collections of quantum states to be quantum computationally indistinguishable. The definition that follows may be viewed as being a nonuniform notion of quantum computational indistinguishability, as it places no uniformity conditions on quantum circuits and allows for an *auxiliary* quantum state σ to assist in the task of state discrimination.

DEFINITION 2.2 (quantum computationally indistinguishable states). *Let S be an infinite set of binary strings, let r be a polynomially bounded function, and let ρ_x and ξ_x be states on $r(|x|)$ qubits for each $x \in S$. The collections $\{\rho_x : x \in S\}$ and $\{\xi_x : x \in S\}$ are* quantum computationally indistinguishable *if, for every choice of polynomially bounded functions s and k, there exists a negligible function ε such that the following property holds for every string $x \in S$: for every $k(|x|)$-qubit state σ and every measurement circuit Q on $r(|x|) + k(|x|)$ qubits having size $s(|x|)$, it is the case that*

$$|\Pr[Q(\rho_x \otimes \sigma) = 1] - \Pr[Q(\xi_x \otimes \sigma) = 1]| \leq \varepsilon(|x|). \tag{2.11}$$

This notion extends naturally to distinguishing collections of channels, as the following definition makes precise.

DEFINITION 2.3 (quantum computationally indistinguishable channels). *Let S be an infinite set of binary strings, let q and r be polynomially bounded functions, and let Φ_x and Ψ_x be channels from $q(|x|)$ qubits to $r(|x|)$ qubits for each $x \in S$. The collections $\{\Phi_x : x \in S\}$ and $\{\Psi_x : x \in S\}$ are quantum computationally indistinguishable if, for every choice of polynomially bounded functions s and k, there exists a negligible function ε such that the following property holds for every string $x \in S$: for every state σ on $q(|x|) + k(|x|)$ qubits and every measurement circuit Q on $r(|x|) + k(|x|)$ qubits having size $s(|x|)$, it is the case that*

$$|\Pr[Q((\Phi_x \otimes \mathbb{1})(\sigma)) = 1] - \Pr[Q((\Psi_x \otimes \mathbb{1})(\sigma)) = 1]| \leq \varepsilon(|x|). \tag{2.12}$$

We will also make use of statistical notions of indistinguishability for states and channels, which are defined as follows.

DEFINITION 2.4 (statistically indistinguishable states). *Let S be an infinite set of binary strings, let r be a polynomially bounded function, and let ρ_x and ξ_x be states on $r(|x|)$ qubits for each $x \in S$. The collections $\{\rho_x : x \in S\}$ and $\{\xi_x : x \in S\}$ are* statistically indistinguishable *if there exists a negligible function ε such that, for all $x \in S$,*

$$\frac{1}{2}\| \rho_x - \xi_x \|_1 \leq \varepsilon(|x|). \tag{2.13}$$

DEFINITION 2.5 (statistically indistinguishable channels). *Let S be an infinite set of binary strings, let q and r be polynomially bounded functions, and let Φ_x and Ψ_x be channels from $q(|x|)$ qubits to $r(|x|)$ qubits for each $x \in S$. The collections $\{\Phi_x : x \in S\}$ and $\{\Psi_x : x \in S\}$ are statistically indistinguishable if there exists a negligible function ε such that, for all $x \in S$,*

$$\frac{1}{2}\| \Phi_x - \Psi_x \|_\diamond \leq \varepsilon(|x|). \tag{2.14}$$

Next we review the definition of quantum computational zero-knowledge proof systems as defined in [58]. Let (P, V) be a quantum or classical interactive proof system for a promise problem A. An arbitrary (possibly malicious) verifier V' is any quantum computational process that interacts with P according to the structural specification of (P, V). Similarly to the classical notion of auxiliary input zero-knowledge, a verifier V' will take, in addition to the input string x, an auxiliary input, and produce some output. This is crucial for the composition of zero-knowledge proof systems. The most general situation allowed by quantum information theory is that both the auxiliary input and the output are quantum, meaning that the verifier operates on quantum registers whose initial state is arbitrary and may be entangled with some external system. Also similarly to the classical case, we will assume that for any given polynomial-time verifier V' there exist polynomially bounded functions q and r that determine the number of auxiliary input qubits and output qubits of V'. To say that V' is a polynomial-time verifier means that the entire action of V' must be described by some polynomial-time generated family of quantum circuits.

The interaction of a verifier V' with P on input x induces some channel from the verifier's $q(|x|)$ auxiliary input qubits to $r(|x|)$ output qubits. Let $\mathcal{W}$ denote the vector space corresponding to the auxiliary input qubits, let $\mathcal{Z}$ denote the space corresponding to the output qubits, and let $\Phi_x : \mathrm{L}(\mathcal{W}) \to \mathrm{L}(\mathcal{Z})$ denote the resulting channel induced by the interaction of V' with P on input x. A simulator S for a given verifier V' is described by a polynomial-time generated family of general quantum circuits that agrees with V' on the functions q and r representing the number of auxiliary input qubits and output qubits, respectively. Such a simulator does not interact with P, but simply induces a channel that we will denote by $\Psi_x : \mathrm{L}(\mathcal{W}) \to \mathrm{L}(\mathcal{Z})$ on each input x.

Definition 2.6 (quantum computational zero-knowledge). *An interactive proof system (P, V) for a promise problem A is* quantum computational zero-knowledge *if, for every polynomial-time generated quantum verifier V', there exists a polynomial-time generated quantum simulator S that satisfies the following requirements:*

1. *The verifier V' and simulator S agree on the polynomially bounded functions q and r that specify the number of auxiliary input qubits and output qubits, respectively.*
2. *Let Φ_x be the channel that results from the interaction between V' and P on input x, and let Ψ_x be the channel induced by the simulator S on input x, both as described above. Then the collections $\{\Phi_x : x \in A_{yes}\}$ and $\{\Psi_x : x \in A_{yes}\}$ are quantum computationally indistinguishable.*

2.5. Cryptographic tools.

In this section we introduce cryptographic building blocks that are useful in our proof system. We emphasize that, as is typical in the classical setting, we formulate all computational security properties (e.g., concealing in a commitment scheme) with respect to nonuniform quantum adversaries, which provides more stringent security requirements and is crucial in many security proofs.

Commitment schemes. Our definition for quantum computationally secure commitment schemes is as follows. We note explicitly that this is a noninteractive definition: all messages are from a sender to a receiver.

Definition 2.7 (quantum computationally secure commitments). *A* quantum computationally secure commitment scheme *for an alphabet Γ is a collection of polynomial-time computable functions $\{f_n : n \in \mathbb{N}\}$ taking the form*

$$f_n : \Gamma \times \Sigma^{p(n)} \to \Sigma^{q(n)} \qquad (2.15)$$

for polynomially bounded functions p and q, such that the following conditions hold:

1. Unconditionally binding property. *For every choice of* $n \in \mathbb{N}$, $a, b \in \Gamma$, *and* $r, s \in \Sigma^{p(n)}$, *one has that* $f_n(a,r) = f_n(b,s)$ *implies* $a = b$.
2. Quantum computationally concealing property. *For every* $a \in \Gamma$ *and* $n \in \mathbb{N}$, *define*

$$\rho_{a,n} = \frac{1}{2^{p(n)}} \sum_{r \in \Sigma^{p(n)}} |f_n(a,r)\rangle\langle f_n(a,r)|. \tag{2.16}$$

For every choice of $a, b \in \Gamma$ *the ensembles* $\{\rho_{a,n} : n \in \mathbb{N}\}$ *and* $\{\rho_{b,n} : n \in \mathbb{N}\}$ *are quantum computationally indistinguishable.*

Such a bit commitment scheme (i.e., $\Gamma = \{0,1\}$) can be constructed based on certain quantum intractability assumptions. As shown in [1], it suffices to have quantum-resistant one-way *permutations*, which are permutations that can be computed efficiently on a classical computer but are hard to invert for both classical and quantum polynomial-time algorithms. The same commitment scheme remains quantum-secure based on a slightly weaker assumption of quantum-resistant *injective* one-way functions. To commit to a string, one can independently use the commitment described above bit by bit.

Based on a quantum-secure commitment scheme, we can obtain the other two essential cryptographic building blocks in our protocol: a zero-knowledge proof system for NP and a coin-flipping protocol, both secure against quantum adversaries.

Zero-knowledge proof for NP. In [58] it was proved that the GMW 3-coloring protocol [26] remains zero-knowledge in the presence of quantum verifiers, assuming the existence of a quantum computationally secure commitment scheme. This means that we have a classical zero-knowledge proof system for any NP language that is secure against arbitrary polynomial-time quantum verifiers.

Coin flipping. A coin-flipping protocol is an interactive process that allows two parties to jointly toss random coins. It is not necessary for us to consider this notion generally, as we only make use of one specific coin-flipping protocol, namely, Blum's coin-flipping protocol [8] in which an honest prover commits to a random $y \in \Sigma$, the honest verifier selects $z \in \Sigma$ at random, the prover reveals y, and the two participants agree that the random bit generated is $r = y \oplus z$.

Damgård and Lunemann [15] proved, assuming the existence of a quantum-secure commitment scheme, that Blum's coin-flipping protocol is quantum-secure. This protocol generates one random coin, and we will need to flip logarithmically many random bits. A simple way of achieving this is by sequential repetition, but more effectively it is possible to extend the analysis of Damgård and Lunemann and show that parallel repetition of Blum's protocol logarithmically many times remains quantum-secure.

2.6. Concatenated Steane codes. The last topic to be discussed in this section concerns the existence of quantum error-correcting codes having certain properties that are important to the functioning of our zero-knowledge proof system for QMA. There are multiple choices of codes that satisfy our requirements, but in the interest of simplicity we will describe just one specific family of codes in this category.

These codes are based on the 7-*qubit Steane code* [51], in which one qubit is encoded into 7 qubits by the following action on standard basis states:

$$|0\rangle \mapsto \frac{1}{\sqrt{8}} \sum_{x \in \mathcal{D}_7^0} |x\rangle \qquad \text{and} \qquad |1\rangle \mapsto \frac{1}{\sqrt{8}} \sum_{x \in \mathcal{D}_7^1} |x\rangle, \tag{2.17}$$

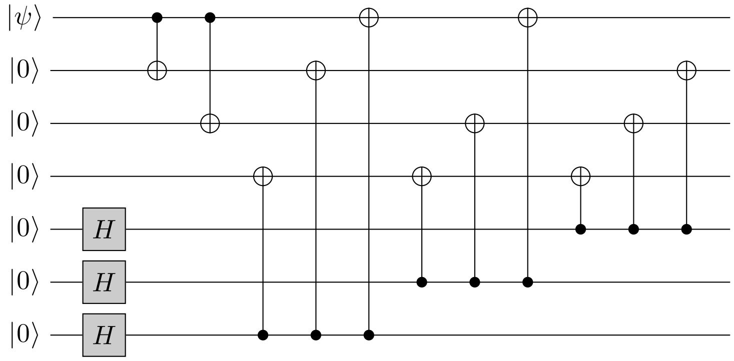


FIG. 2.1. *A Clifford circuit encoder for the 7-qubit Steane code. Hereafter we will write U_7 to refer to the unitary operator on 7 qubits described by this circuit.*

where

$$\mathcal{D}_7^0 = \{0000000, 0001111, 0110011, 0111100, 1010101, 1011010, 1100110, 1101001\},$$
$$\mathcal{D}_7^1 = \{0010110, 0011001, 0100101, 0101010, 1000011, 1001100, 1110000, 1111111\}.$$

It is the case that $\mathcal{D}_7^0$ is a $[7, 4]$-Hamming code, while

$$\mathcal{D}_7 = \mathcal{D}_7^0 \cup \mathcal{D}_7^1 \tag{2.18}$$

is the dual code to $\mathcal{D}_7^0$ (i.e., it is the code consisting of all binary strings of length 7 whose inner product with any codeword in $\mathcal{D}_7^0$ is even). This is an example of a *CSS code* [48], and it is capable of correcting single-qubit errors. The standard error-correcting procedure, which we do not actually need in this paper, is to first reversibly correct errors in the standard basis, with respect to the code $\mathcal{D}_7$, and then to do the same with respect to the diagonal basis. The 7-qubit Clifford circuit depicted in Figure 2.1 encodes one qubit into 7 with respect to this code, assuming 6 qubits in the $|0\rangle$ state are made available.

One of the properties of the 7-qubit Steane code that is important from the viewpoint of this paper is that it admits a *transversal* application of Clifford operations, in the sense that is explained in Figure 2.2.

Note that by concatenating the 7-qubit Steane code with itself, one obtains a code having similar properties to the 7-qubit code and, in addition, having a large minimum distance for the underlying code. More specifically, suppose that $N = 7^t$ for t being an even positive integer. (We take t to be even for convenience, as this eliminates the entrywise complex conjugation on Clifford operations induced by their transversal application.) By concatenating the 7-qubit Steane code to itself t times, one obtains a quantum error-correcting code in which one qubit is encoded into N qubits in the following way:

$$|0\rangle \mapsto \frac{1}{\sqrt{8^t}} \sum_{x \in \mathcal{D}_N^0} |x\rangle \quad \text{and} \quad |1\rangle \mapsto \frac{1}{\sqrt{8^t}} \sum_{x \in \mathcal{D}_N^1} |x\rangle, \tag{2.19}$$

where $\mathcal{D}_N^0, \mathcal{D}_N^1 \subseteq \Sigma^N$ are related in a way that generalizes the case $N = 7$. In particular, $\mathcal{D}_N^0$ is a binary linear code having 8^t elements, and whose dual code takes the form $\mathcal{D}_N = \mathcal{D}_N^0 \cup \mathcal{D}_N^1$ for $\mathcal{D}_N^1 \subseteq \Sigma^N$ being a coset of $\mathcal{D}_N^0$.

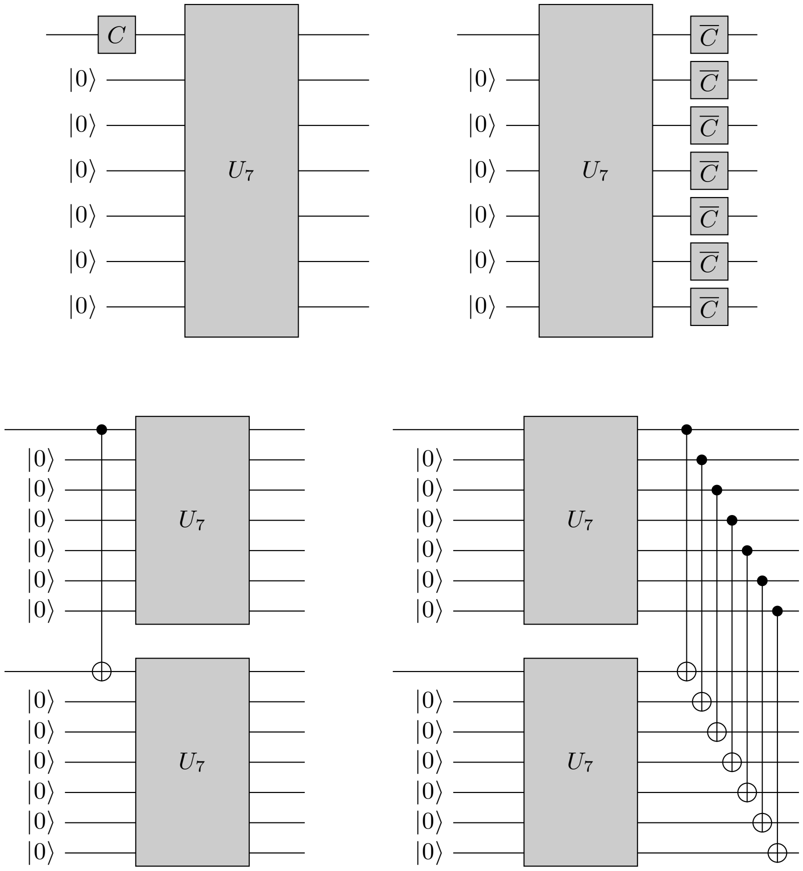


FIG. 2.2. *The 7-qubit Steane code allows for the transversal application of Clifford operations. That is, the circuits on the left are equivalent to the corresponding circuits on the right. In general, the application of any Clifford operation on k qubits prior to being encoded is equivalent to the entrywise complex conjugate of that Clifford operation being applied 7 times to the $7k$ qubits that encode the original k qubits.*

The t-fold concatenation of the 7-qubit Steane code inherits the properties of the 7-qubit Steane code mentioned above. A Clifford circuit U_N acting on N qubits, $N-1$ of which are to be initialized in the $|0\rangle$ state, performs the encoding. This circuit is obtained by creating a tree from multiple copies of the circuit U_7 in the natural way. The code allows for Clifford operations to be applied transversally.

An added feature of the concatenated versions of the 7-qubit Steane code is that it corrects more errors than the ordinary 7-qubit code. In particular, we will make use of the fact that the code $\mathcal{D}_N$, for $N = 7^t$, has minimum Hamming weight 3^t for a nonzero code word. This allows one to obtain a polynomial-length code for any polynomial lower bound on the minimum nonzero Hamming weight of a code word.

3. The local Clifford–Hamiltonian problem.

The local Hamiltonian problem is a well-known example of a complete problem for QMA, provided that certain assumptions are in place regarding the gap between the ground state energy (i.e., the smallest eigenvalue) of input Hamiltonians for yes- and no-inputs. A general and somewhat imprecise formulation of the local Hamiltonian problem is as follows.

The k-local Hamiltonian (k-LH) problem.

Input: A collection $H_1, \dots, H_m$ of k-local Hamiltonian operators, each acting on n qubits and satisfying $0 \leq H_j \leq \mathbb{1}$ for $j = 1, \dots, m$, along with real numbers α and β satisfying $\alpha < \beta$.

Yes: There exists an n-qubit state ρ such that $\langle \rho, H_1 + \cdots + H_m \rangle \leq \alpha$.

No: For every n-qubit state ρ, it holds that $\langle \rho, H_1 + \cdots + H_m \rangle \geq \beta$.

This problem statement is imprecise in the sense that it does not specify how α and β are to be represented or what requirements are placed on the gap $\beta - \alpha$ mentioned above. We will be more precise about these issues when formulating a restricted version of this problem below, but it is appropriate that we first summarize what is already known.

It is known that k-LH is complete for QMA (with respect to Karp reductions) provided α and β are input in a reasonable way and separated by an inverse polynomial gap; this was first proved by Kitaev, Shen, and Vyalyi [40] for the case $k = 5$, then by Kempe and Regev [38] for $k = 3$ and Kempe, Kitaev, and Regev [37] for $k = 2$. If one adds the additional requirement that α is exponentially small, which will be important in the context of this paper, then QMA-completeness for $k = 5$ still follows from Kitaev's proof, but the proofs of Kempe and Regev and Kempe, Kitaev, and Regev do not imply the same for $k = 3$ and $k = 2$. On the other hand, the works of Bravyi [9] and Gosset and Nagaj [30] do establish QMA-completeness for exponentially small α for $k = 4$ and $k = 3$, respectively.

The restricted version of the local Hamiltonian we introduce is one in which each Hamiltonian term H_j is not only k-local and satisfies $0 \leq H_j \leq \mathbb{1}$ but, furthermore, on the k qubits on which it acts nontrivially, its action must be given by a rank 1 projection operator of the form

$$C_j^* |0^k\rangle\langle 0^k| C_j \tag{3.1}$$

for some choice of a k-qubit Clifford operation C_j. For brevity, we will refer to any such operator as a *k-local Clifford–Hamiltonian projection.* The precise statement of our problem variant is as follows.

The k-local Clifford–Hamiltonian (k-LCH) problem.

Input: A collection $H_1, \dots, H_m$ of k-local Clifford–Hamiltonian projections, along with positive integers p and q expressed in unary notation (i.e., as strings 1^p and 1^q) and satisfying $2^p > q$.

Yes: There exists an n-qubit state ρ such that $\langle \rho, H_1 + \cdots + H_m \rangle \leq 2^{-p}$.

No: For every n-qubit state ρ, it holds that $\langle \rho, H_1 + \cdots + H_m \rangle \geq 1/q$.

It may be noted that, by the particular way we have stated this problem, we are focusing on a variant of the local Hamiltonian problem in which the parameter α may be exponentially small and the gap $\beta - \alpha$ is at least inverse polynomial.

THEOREM 3.1. *The* 5*-local Clifford–Hamiltonian problem is* QMA*-complete with respect to Karp reductions. Moreover, for any choice of a promise problem $A \in$* QMA *and a polynomially bounded function p, there exists a Karp reduction f from A to*

5-LCH having the form

$$f(x) = \left\langle H_1, \ldots, H_m, 1^{p(|x|)}, 1^q \right\rangle \tag{3.2}$$

for every $x \in A_{yes} \cup A_{no}$.

Proof. The containment of the 5-local Clifford–Hamiltonian problem in QMA follows from the fact that the 5-LH problem is in QMA for the same choice of the ground state energy bounds. It therefore remains to prove the statement concerning the QMA-hardness of the 5-LCH problem.

Let $A = (A_{\text{yes}}, A_{\text{no}})$ be any promise problem in QMA and let p be a polynomially bounded function. Using a standard error reduction procedure for QMA, one may conclude that there exists a polynomial-time generated collection

$$\{V_x \,:\, x \in A_{\text{yes}} \cup A_{\text{no}}\} \tag{3.3}$$

of measurement circuits having these properties:

1. If $x \in A_{\text{yes}}$, then there exists a state ρ such that $V_x(\rho) = 1$ with probability $1 - 2^{-p(|x|)}$.
2. If $x \in A_{\text{no}}$, then for all quantum states ρ representing valid inputs to V_x it holds that $V_x(\rho) = 1$ with probabilityh at most $1/2$.

It is known that $\{\Lambda(P), H\}$ is a universal gate set for quantum computation, so there is no loss of generality in assuming each V_x is a quantum circuit using gates from this set, together with a supply of ancillary qubits initialized to the state $|0\rangle$. For technical reasons (which are discussed later) we will assume something marginally stronger, which is that each V_x uses gates from the set $\{\Lambda(P), H \otimes H\}$. That is, every Hadamard gate appearing in V_x is paired with another Hadamard gate to be applied at the same time but on a different qubit. Note that for any circuit composed of gates from the set $\{\Lambda(P), H\}$, this stronger condition is easily met by adding to this circuit a number of additional Hadamard gates on an otherwise unused ancilla qubit.

Now consider the 5-local circuit-to-Hamiltonian construction of Kitaev, Shen, and Vyalyi [40], for a given choice of V_x. In this construction, the resulting Hamiltonians have the form

$$H_{\text{total}} = H_{\text{in}} + H_{\text{out}} + H_{\text{clock}} + H_{\text{prop}}, \tag{3.4}$$

where the terms check the initialization, readout, validity of unary clock, and propagation of computation, respectively. It follows from Kitaev's proof that, for $x \in A_{\text{yes}}$, the resulting Hamiltonian H_{total} has ground state energy at most $2^{-p(|x|)}$, and for $x \in A_{\text{no}}$ the ground state energy of H_{total} is at least $1/q(|x|)$, for some polynomially bounded function q. To complete the proof, it suffices to demonstrate that each of these terms can be expressed as a sum of Clifford–Hamiltonian projections.

The first three terms, H_{in}, H_{out}, and H_{clock}, can easily be expressed as sums of Clifford–Hamiltonian projections, as they are all projection operators that are diagonal in the standard basis. The propagation term has the form $H_{\text{prop}} = \sum_{t=1}^{T} H_{\text{prop},t}$, where each operator $H_{\text{prop},t}$ takes the form

$$\begin{aligned} H_{\text{prop},t} &= \frac{1}{2}\big[(|100\rangle\langle 100|_{t-1,t,t+1} + |110\rangle\langle 110|_{t-1,t,t+1}) \otimes \mathbb{1} \\ &\qquad - |110\rangle\langle 100|_{t-1,t,t+1} \otimes U_t - |100\rangle\langle 110|_{t-1,t,t+1} \otimes U_t^*\big] \\ &= |10\rangle\langle 10|_{t-1,t+1} \otimes \frac{1}{2}\big[\mathbb{1}_t \otimes \mathbb{1} - |1\rangle\langle 0|_t \otimes U_t - |0\rangle\langle 1|_t \otimes U_t^*\big]. \end{aligned} \tag{3.5}$$

Here, the first three qubits (indexed by $t-1$, t, and $t+1$) refer to qubits in a clock register and U_t represents the tth unitary gate in V_x. To prove that each propagation operator $H_{\text{prop},t}$ can be expressed as a sum of Clifford–Hamiltonian projections, it suffices to prove the same for every projection of the form

$$\frac{1}{2}\big[\mathbb{1}\otimes\mathbb{1} - |1\rangle\langle 0|\otimes U - |0\rangle\langle 1|\otimes U^*\big] \tag{3.6}$$

for U being either $\Lambda(P)$ or $H\otimes H$.

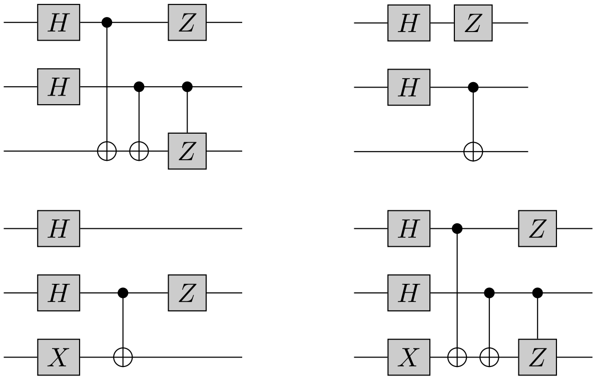
In the case that $U=\Lambda(P)$, one has that the projection (3.6) is the sum of the four Clifford–Hamiltonian projections corresponding to these vectors:

$$\begin{aligned}
|-\rangle|00\rangle &= (ZH\otimes\mathbb{1}\otimes\mathbb{1})|000\rangle,\\
|-\rangle|01\rangle &= (ZH\otimes\mathbb{1}\otimes X)|000\rangle,\\
|-\rangle|10\rangle &= (ZH\otimes X\otimes\mathbb{1})|000\rangle,\\
|\circlearrowright\rangle|11\rangle &= (P^*H\otimes X\otimes X)|000\rangle,
\end{aligned} \tag{3.7}$$

where $|\circlearrowright\rangle = (|0\rangle - i|1\rangle)/\sqrt{2}$. In the case that $U=H\otimes H$, one has that the projection (3.6) is the sum of the four Clifford–Hamiltonian projections corresponding to these vectors:

$$\begin{aligned}
|\psi_1\rangle &= \big(|000\rangle - |011\rangle - |101\rangle - |110\rangle\big)/2,\\
|\psi_2\rangle &= \big(|000\rangle + |011\rangle - |100\rangle - |111\rangle\big)/2,\\
|\psi_3\rangle &= \big(|001\rangle - |010\rangle + |101\rangle - |110\rangle\big)/2,\\
|\psi_4\rangle &= \big(|001\rangle + |010\rangle - |100\rangle + |111\rangle\big)/2.
\end{aligned} \tag{3.8}$$

All four of these vectors are obtained by a Clifford operation applied to the all-zero state. In particular, when the following Clifford circuits are applied to the state $|000\rangle$, the states $|\psi_1\rangle$, $|\psi_2\rangle$, $|\psi_3\rangle$, and $|\psi_4\rangle$ are obtained:



This completes the proof. □

Remark 3.2. If one is given a witness to a given QMA problem A, it is possible to efficiently compute a witness to the corresponding k-local Hamiltonian problem instance through Kitaev's reduction by preparing a superposition of clock states and then running a verification circuit for the corresponding number of steps. Our reduction also inherits this property.

Remark 3.3. There is no loss of generality in setting $q = 1$ in the statement of the k-LCH problem, meaning that Theorem 3.1 holds for this somewhat simplified problem statement. This may be proved by repeating each Hamiltonian term q times in a given problem instance and adjusting p as necessary.

Remark 3.4. States of the form $C|0^k\rangle$ for a Clifford operation C, are stabilizer states of k qubits. Theorem 3.1 therefore implies that there exists a QMA verification procedure in which the verifier randomly chooses a k-qubit stabilizer state and checks whether the quantum witness state is orthogonal to it.

Remark 3.5. If one takes $U = H$ in (3.6), the resulting projection operator projects onto the two-dimensional subspace spanned by $|-\rangle|\gamma_0\rangle$ and $|+\rangle|\gamma_1\rangle$, where

$$|\gamma_0\rangle = \cos(\pi/8)|0\rangle + \sin(\pi/8)|1\rangle \quad \text{and} \quad |\gamma_1\rangle = \sin(\pi/8)|0\rangle - \cos(\pi/8)|1\rangle \tag{3.9}$$

are eigenvectors of H. This projection cannot be expressed as a sum of Clifford–Hamiltonian projections, which explains why we needed to replace H with $H \otimes H$ in the proof above.

While considering this projection is not useful for proving Theorem 3.1, we do obtain from it a different result. In particular, we obtain an alternative proof of a result due to Morimae, Nagaj, and Schuch [45] establishing that single-qubit measurements and classical postprocessing are sufficient for QMA verification. Reference [45] actually provides two proofs of this fact, one based on measurement-based quantum computation and the other based on a local-Hamiltonian problem type of approach similar to what we propose. While their local-Hamiltonian approach does not work for one-sided error (or QMA$_1$) verifications, ours does (as does their measurement-based quantum computation proof).

4. Description of the proof system.

We now describe our zero-knowledge protocol for the local Clifford–Hamiltonian problem. The main steps of the proof system are described in the subsections that follow, and the entire proof system is summarized in Figure 4.1. Properties of the proof system, including completeness, soundness, and the zero-knowledge property, are discussed in later sections of the paper.

As was previously suggested, our proof system assumes the existence of a quantum computationally secure commitment scheme. Throughout this section it is to be assumed that an instance of the k-LCH problem has been selected. The instance describes Clifford–Hamiltonian projections $H_1, \ldots, H_m$, each given by $H_j = C_j^* |0^k\rangle\langle 0^k| C_j$ for k-qubit Clifford operations $C_1, \ldots, C_m$, along with a specification of which of the n qubits these projections act upon. The proof system does not refer to the parameters p and q in the description of the k-LCH problem, as these parameters are only relevant to the performance of the proof system and not its implementation. It must be assumed, however, that the completeness parameter 2^{-p} is a negligible function of the entire problem instance size in order for the proof system to be zero-knowledge, and we will make this assumption hereafter.

4.1. Prover's witness encoding.

Let $\mathsf{X} = (\mathsf{X}_1, \ldots, \mathsf{X}_n)$ be an n-tuple of single-qubit registers. These qubits are assumed to initially be in the prover's possession, and they store an n-qubit quantum state ρ representing a possible witness for the instance of the k-LCH problem under consideration.

The first step of the proof system requires the prover to encode the state of X, using a scheme that consists of four steps. Throughout the description of these steps it is to be assumed that N is a polynomially bounded function of the input size and is

Prover's encoding step:

The prover selects a tuple (t, π, a, b) uniformly at random, where $t = t_1 \cdots t_n$ for $t_1, \ldots, t_n \in \{0, +, \circlearrowright\}^N$, $\pi \in S_{2N}$, and $a = a_1 \cdots a_n$ and $b = b_1 \cdots b_n$ for $a_1, \ldots, a_n, b_1, \ldots, b_n \in \Sigma^{2N}$. The witness state contained in qubits $(\mathsf{X}_1, \ldots, \mathsf{X}_n)$ is encoded into qubit tuples

$$\left(\mathsf{Y}_1^1, \ldots, \mathsf{Y}_{2N}^1\right), \ldots, \left(\mathsf{Y}_1^n, \ldots, \mathsf{Y}_{2N}^n\right) \tag{4.1}$$

as described in the main text. These qubits are sent to the verifier, along with a commitment to the tuple (π, a, b).

Coin-flipping protocol:

The prover and verifier engage in a coin-flipping protocol, choosing a string r of a fixed length uniformly at random. This random string r determines a Hamiltonian term $H_r = C_r^* |0^k\rangle\langle 0^k| C_r$ that is to be tested.

Verifier's measurement:

The verifier applies the Clifford operation C_r transversally to the qubits

$$\left(\mathsf{Y}_1^{i_1}, \ldots, \mathsf{Y}_{2N}^{i_1}\right), \ldots, \left(\mathsf{Y}_1^{i_k}, \ldots, \mathsf{Y}_{2N}^{i_k}\right), \tag{4.2}$$

and measures all of these qubits in the standard basis for $(i_1, \ldots, i_k)$ being the indices of the qubits upon which the Hamiltonian term H_r acts nontrivially. The result of this measurement is sent to the prover.

Prover's verification and response:

The prover checks that the verifier's measurement results are consistent with the states of the trap qubits and the concatenated Steane code, aborting the proof system if not (causing the verifier to reject). In case the measurement results are consistent, the prover demonstrates that these measurement results are consistent with its prior commitment to (π, a, b) and with the Hamiltonian term H_r, through a classical zero-knowledge proof system for the corresponding NP statement described in the main text. The verifier accepts or rejects accordingly.

FIG. 4.1. *Summary of the zero-knowledge proof system for the LCH problem.*

an even positive integer power of 7. In effect, N acts as a security parameter (for the zero-knowledge property of the proof system), and we take it to be an even power of 7 so that it may be viewed as a number of qubits that could arise from a concatenated Steane code allowing for a transversal application of Clifford operations, as described in section 2.6. In particular, through an appropriate choice of N, one may guarantee that this code has any desired polynomial lower bound for the minimum nonzero Hamming weight of its underlying classical code.

1. For each $i = 1, \ldots, n$, the qubit X_i is encoded into qubits $(\mathsf{Y}_1^i, \ldots, \mathsf{Y}_N^i)$ by means of the concatenated Steane code. This results in the N-tuples

$$\left(\mathsf{Y}_1^1, \ldots, \mathsf{Y}_N^1\right), \ldots, \left(\mathsf{Y}_1^n, \ldots, \mathsf{Y}_N^n\right). \tag{4.3}$$

2. To each of the N-tuples in (4.3), the prover concatenates an additional N *trap qubits* with each trap qubit being initialized to one of the single qubit pure states $|0\rangle$, $|+\rangle$, or $|\circlearrowright\rangle$, selected independently and uniformly at random.

This results in qubits

$$\big(\mathsf{Y}_1^1, \ldots, \mathsf{Y}_{2N}^1\big), \ldots, \big(\mathsf{Y}_1^n, \ldots, \mathsf{Y}_{2N}^n\big). \tag{4.4}$$

The prover stores the string $t = t_1 \cdots t_n$, for $t_1, \ldots, t_n \in \{0, +, \circlearrowright\}^N$ representing the randomly chosen states of the trap qubits.

3. A random permutation $\pi \in S_{2N}$ is selected, and the qubits in each of the $2N$-tuples (4.4) are permuted according to π. (Note that it is a single permutation π that is selected and applied to all of the $2N$-tuples simultaneously.)
4. The quantum one-time pad is applied independently to each qubit in (4.4) (after they are permuted in step 3). That is, for $a_i, b_i \in \Sigma^{2N}$ chosen independently and uniformly at random, the unitary transformation $X^{a_i} Z^{b_i}$ is applied to $(\mathsf{Y}_1^i, \ldots, \mathsf{Y}_{2N}^i)$, and the strings a_i and b_i are stored by the prover for each $i = 1, \ldots, n$.

The randomness required by these encoding steps is described by a tuple (t, π, a, b), where t is the string representing the states of the trap qubits described in step 2, $\pi \in S_{2N}$ is the permutation applied in step 3, and $a = a_1 \cdots a_n$ and $b = b_1 \cdots b_n$ are binary strings representing the Pauli operators applied in the one-time pad in step 4. After performing the above encoding steps, the prover sends the resulting qubits,

$$\mathsf{Y} = \big(\big(\mathsf{Y}_1^1, \ldots, \mathsf{Y}_{2N}^1\big), \ldots, \big(\mathsf{Y}_1^n, \ldots, \mathsf{Y}_{2N}^n\big)\big), \tag{4.5}$$

along with a commitment

$$z = \operatorname{commit}((\pi, a, b), s) \tag{4.6}$$

to the tuple $\big(\pi, a, b\big)$, to the verifier. Here we assume that s is a random string chosen by the prover that allows for this commitment. (It is not necessary for the prover to commit to the selection of the trap qubit states indicated by t, although it would not affect the properties of the proof system if it were modified so that the prover also committed to the trap qubit state selections.)

4.2. Verifier's random challenge. Upon receiving the prover's encoded witness and commitment, the verifier issues a challenge: for a randomly selected index $j \in \{1, \ldots, m\}$, the verifier will check that the jth Hamiltonian term

$$H_j = C_j^* |0^k\rangle\langle 0^k| C_j \tag{4.7}$$

is not violated. Generally speaking, the verifier's actions in issuing this challenge are as follows: for a certain collection of qubits, the verifier applies the Clifford operation C_j transversally to those qubits, performs a measurement with respect to the standard basis, sends the outcomes to the prover, and then expects the prover to demonstrate that the obtained outcomes are valid (in the sense to be described later).

The randomly selected Hamiltonian term is to be determined by a binary string r, of a fixed length $\lceil \log m \rceil$, that should be viewed as being chosen uniformly at random. (In a moment we will discuss the random choice of r, which will be given by the output of a coin-flipping protocol that happens to be uniform for honest participants.) It is not important exactly how the binary strings of length $\lceil \log m \rceil$ are mapped to the indices $\{1, \ldots, m\}$, so long as every index is represented by at least one string—so that for a uniformly chosen string r, each Hamiltonian term j is selected with a nonnegligible probability. We will write H_r and C_r in place of H_j and C_j, and refer to the Hamiltonian term determined by r, when it is convenient to do this.

It would be natural to allow the verifier to randomly determine which Hamiltonian term is to be tested—but, as suggested above, we will assume that the challenge is determined through a *coin-flipping protocol* rather than leaving the choice to the verifier. More specifically, throughout the present subsection, it should be assumed that the random choice of the string r that determines which challenge is issued is the result of independent iterations of a commitment-based coin-flipping protocol (i.e., the honest prover commits to a random $y_i \in \Sigma$, the honest verifier selects $z_i \in \Sigma$ at random, the prover reveals y_i, and the two participants agree that the ith random bit of r is $r_i = y_i \oplus z_i$). This guarantees (assuming the security of the commitment protocol) that the choices are truly random, and greatly simplifies the analysis of the zero-knowledge property of the proof system. The use of such a protocol might not actually be necessary for the security of the proof system, but we leave the investigation of whether it is necessary to future work.

Now, let $(i_1, \dots, i_k)$ denote the indices of the qubits upon which the Hamiltonian term determined by the random string r acts nontrivially. The verifier applies the Clifford operation C_r independently to each of the k-qubit tuples

$$\big(\mathsf{Y}_1^{i_1}, \dots, \mathsf{Y}_1^{i_k}\big), \dots, \big(\mathsf{Y}_{2N}^{i_1}, \dots, \mathsf{Y}_{2N}^{i_k}\big), \tag{4.8}$$

which is equivalent to saying that C_r is applied transversally to the tuples

$$\big(\mathsf{Y}_1^{i_1}, \dots, \mathsf{Y}_{2N}^{i_1}\big), \; \dots, \; \big(\mathsf{Y}_1^{i_k}, \dots, \mathsf{Y}_{2N}^{i_k}\big) \tag{4.9}$$

that encode the qubits on which the Hamiltonian term H_r acts nontrivially. The qubits (4.9) are then measured with respect to the standard basis, and the results are sent to the prover. We will let $u_{i_1}, \dots, u_{i_k} \in \Sigma^{2N}$ denote the binary strings representing the verifier's standard basis measurement outcomes (or claimed outcomes) corresponding to the measurements of the tuples (4.9).

4.3. Prover's check and response. Upon receiving the verifier's claimed measurement outcomes corresponding to the randomly selected Hamiltonian term, the prover first checks to see that these outcomes could indeed have come from the measurements specified above, and then tries to convince the verifier that these measurement outcomes are consistent with the selected term.

In more detail, suppose that the Hamiltonian term determined by r has been challenged. As above, we assume that this term acts nontrivially on the k qubits indexed by the k-tuple $(i_1, \dots, i_k)$, and we will write $u = u_{i_1} \cdots u_{i_k} \in \Sigma^{2kN}$ to denote the verifier's claimed standard basis measurement outcomes.

To define the prover's check for this string, it will be helpful to first define a predicate R_r, which is a function of t, π, and u, and essentially represents the prover's check *after* it has made an adjustment to the verifier's response to account for the one-time pad. For each $i \in \{i_1, \dots, i_k\}$, define strings $y_i, z_i \in \Sigma^N$ so that

$$\pi(y_i z_i) = u_i. \tag{4.10}$$

The predicate R_r takes the value 1 if and only if these two conditions are met:

1. $y_i \in \mathcal{D}_N$ for every $i \in \{i_1, \dots, i_k\}$, and $y_i \in \mathcal{D}_N^1$ for at least one index $i \in \{i_1, \dots, i_k\}$.
2. $\big\langle z_{i_1} \cdots z_{i_k} \,\big|\, C_r^{\otimes N} \,\big|\, t_{i_1} \cdots t_{i_k} \big\rangle \neq 0$.

(Here we have written $|t_{i_1} \cdots t_{i_k}\rangle$ to denote the pure state of kN qubits obtained by tensoring the states $|0\rangle$, $|+\rangle$, and $|\circlearrowright\rangle$ in the most natural way.) The first condition concerns measurement outcomes corresponding to nontrap qubits, and reflects the condition that these measurement outcomes are proper encodings of binary values—but not all of which encode 0. The second condition concerns the consistency of the verifier's measurements with the trap qubits.

Next, we will define a predicate Q_r, which is a function of the variables t, π, a, b, and u, where t, π, and u are as above and $a, b \in \Sigma^{2nN}$ refer to the strings used for the one-time pad. The predicate Q_r represents the prover's actual check, in the case that the Hamiltonian term determined by r has been selected, including an adjustment to account for the one-time pad. Let $c_1, \ldots, c_n, d_1, \ldots, d_n \in \Sigma^{2N}$ be the unique strings for which the equation

$$C_r^{\otimes 2N}\big(X^{a_1}Z^{b_1} \otimes \cdots \otimes X^{a_n}Z^{b_n}\big) = \alpha\big(X^{c_1}Z^{d_1} \otimes \cdots \otimes X^{c_n}Z^{d_n}\big)C_r^{\otimes 2N} \tag{4.11}$$

holds for some choice of $\alpha \in \{1, i, -1, -i\}$. The Clifford operation C_r acts trivially on those qubits indexed by strings outside of the set $\{i_1, \ldots, i_k\}$, so it must be the case that $c_i = a_i$ and $d_i = b_i$ for $i \notin \{i_1, \ldots, i_k\}$, but for those indices $i \in \{i_1, \ldots, i_k\}$ it may be the case that $c_i \neq a_i$ and $d_i \neq b_i$. We will also write $c = c_1 \cdots c_n$ and $d = d_1 \cdots d_n$ for the sake of convenience. Given a description of the Clifford operation C_r it is possible to efficiently compute c and d from a and b. Having defined c and d, we may now express the predicate Q_r as follows:

$$Q_r(t, \pi, u, a, b) = R_r\big(t, \pi, u \oplus c_{i_1} \cdots c_{i_k}\big). \tag{4.12}$$

In essence, the predicate Q_r checks the validity of the verifier's claimed measurement results by first adjusting for the one-time pad, then referring to R_r.

The prover evaluates the predicate Q_r, and aborts the proof system if the predicate evaluates to 0 (as this is indicative of a dishonest verifier). Otherwise, the prover aims to convince the verifier that the measurement outcomes u are consistent with the prover's encoding, and also that they are not in violation of the Hamiltonian term H_r. It does this specifically by engaging in a classical zero-knowledge proof system for the following NP statement: there exists a random string s and an encoding key (t, π, a, b) such that (i) $\text{commit}((\pi, a, b), s)$ matches the prover's initial commitment z, and (ii) $Q_r(t, \pi, u, a, b) = 1$.

It will be convenient later, in the analysis of the proof system, to sometimes view r as being an input to the predicates defined above. Specifically, we define predicates

$$Q(r, t, \pi, a, b, u) = Q_r(t, \pi, a, b, u) \quad \text{and} \quad R(r, t, \pi, u) = R_r(t, \pi, u) \tag{4.13}$$

for this purpose. We also note explicitly that these predicates are polynomial-time computable.

5. Completeness and soundness of the proof system.

It is evident that the proof system described in the previous section is complete. For a given instance of the local Clifford–Hamiltonian problem, if the prover and verifier both behave honestly, as suggested in the description of the proof system, the verifier will accept with precisely the same probability that would be obtained by randomly selecting a Hamiltonian term, measuring the original n-qubit witness state against the corresponding projection, and accepting or rejecting accordingly. For a positive problem instance, this acceptance probability is at least $1 - 2^{-p}$ (for every choice of a random string r).

Next we will consider the soundness of the proof system. We will prove that on a negative instance of the problem, the honest verifier must reject with nonnegligible probability. The prover initially sends to the verifier the qubits

$$\big(\mathsf{Y}_1^1, \ldots, \mathsf{Y}_{2N}^1\big), \ldots, \big(\mathsf{Y}_1^n, \ldots, \mathsf{Y}_{2N}^n\big), \tag{5.1}$$

along with a commitment $z = \text{commit}((\pi, a, b), s)$ to a tuple (π, a, b). We have assumed that the commitment is perfectly binding, so there is a well-defined tuple (π, a, b) that is determined by the prover's commitment z. We may assume without loss of generality that this tuple has the proper form (meaning that $\pi \in S_{2N}$ is a permutation and a and b are binary strings of length $2nN$, as specified in the description of the proof system), as a commitment to a string not of this form must lead to rejection with high probability in all cases. Let ξ be the state of the qubits

$$\big(\mathsf{Y}_1^1, \ldots, \mathsf{Y}_N^1\big), \ldots, \big(\mathsf{Y}_1^n, \ldots, \mathsf{Y}_N^n\big) \tag{5.2}$$

that is obtained by inverting the quantum one-time pad with respect to the strings a and b, inverting the permutation of each of the tuples (5.1) with respect to the permutation π, and discarding the last N qubits within each tuple (i.e., the trap qubits). For an honest prover, the state ξ would be the state obtained by encoding the original witness state using the concatenated Steane code—although in general it cannot be assumed that ξ arises in this way. Although the verifier is not capable of recovering the state ξ on its own, because it does not know (π, a, b), it will nevertheless be helpful to refer to the state ξ for the purposes of establishing the soundness condition of the proof system.

We will define a collection of N-qubit projections operators and a channel from N qubits to one that will be useful for establishing soundness. First, let

$$\Pi_0 = \sum_{x \in \mathcal{D}_N^0} |x\rangle\langle x| \qquad \text{and} \qquad \Pi_1 = \sum_{x \in \mathcal{D}_N^1} |x\rangle\langle x|, \tag{5.3}$$

where $\mathcal{D}_N^0$ and $\mathcal{D}_N^1$ are subsets of Σ^N representing classical code words of the concatenated Steane code. A standard basis measurement of any qubit encoded using this code will necessarily yield an outcome in one of these two sets: an encoded $|0\rangle$ state yields an outcome in $\mathcal{D}_N^0$, and an encoded $|1\rangle$ state yields an outcome in $\mathcal{D}_N^1$. The projections Π_0 and Π_1 therefore correspond to these two possibilities, while the projection operator $\mathbb{1} - (\Pi_0 + \Pi_1)$ corresponds to the situation in which a standard basis measurement has yielded a result outside of the classical code space $\mathcal{D}_N = \mathcal{D}_N^0 \cup \mathcal{D}_N^1$. Also define projections

$$\Delta_0 = \frac{\mathbb{1}^{\otimes N} + Z^{\otimes N}}{2} \qquad \text{and} \qquad \Delta_1 = \frac{\mathbb{1}^{\otimes N} - Z^{\otimes N}}{2}, \tag{5.4}$$

which are the projections onto the spaces spanned by all even- and odd-parity standard basis states, respectively. It holds that $\Pi_0 \leq \Delta_0$ and $\Pi_1 \leq \Delta_1$, as the codewords in $\mathcal{D}_N^0$ all have even parity and the codewords in $\mathcal{D}_N^1$ all have odd parity. Finally, define a channel Ξ_N, mapping N qubits to 1 qubit, as follows:

$$\Xi_N(\sigma) = \frac{\langle \mathbb{1}^{\otimes N}, \sigma\rangle \mathbb{1} + \langle X^{\otimes N}, \sigma\rangle X + \langle Y^{\otimes N}, \sigma\rangle Y + \langle Z^{\otimes N}, \sigma\rangle Z}{2} \tag{5.5}$$

for every N-qubit operator σ. It is evident that this mapping preserves trace, and is completely positive when $N \equiv 1 \pmod 4$, which holds because N is an even power of 7. The complete positivity of Ξ_N when $N \equiv 1 \pmod 4$ may be verified by establishing that its Choi operator is positive semidefinite, which is a routine verification:

$$\begin{aligned} J(\Xi_N) &= \frac{1}{2}\big(\mathbb{1}^{\otimes(N+1)} + X^{\otimes(N+1)} - Y^{\otimes(N+1)} + Z^{\otimes(N+1)}\big) \\ &= \frac{1}{8}\big(\mathbb{1}^{\otimes(N+1)} + X^{\otimes(N+1)} - Y^{\otimes(N+1)} + Z^{\otimes(N+1)}\big)^2. \end{aligned} \tag{5.6}$$

One may observe that the adjoint mapping to Ξ_N is given by

$$\Xi_N^*(\tau) = \frac{\langle \mathbb{1}, \tau\rangle \mathbb{1}^{\otimes N} + \langle X, \tau\rangle X^{\otimes N} + \langle Y, \tau\rangle Y^{\otimes N} + \langle Z, \tau\rangle Z^{\otimes N}}{2}, \tag{5.7}$$

and satisfies

$$\Xi_N^*(|0\rangle\langle 0|) = \Delta_0 \qquad \text{and} \qquad \Xi_N^*(|1\rangle\langle 1|) = \Delta_1. \tag{5.8}$$

Now, consider the state $\rho = \Xi_N^{\otimes n}(\xi)$ of the qubits $(\mathsf{X}_1, \ldots, \mathsf{X}_n)$ that is obtained from ξ when Ξ_N is applied independently to each of the N-tuples of qubits in (5.2). We will prove that the verifier must reject with nonnegligible probability for a given choice of r provided that ρ violates the corresponding Hamiltonian term H_r. Because every n-qubit state creates a nonnegligible violation in at least one Hamiltonian term for a negative problem instance, this will suffice to prove the soundness of the proof system.

For each random string r generated by the coin-flipping procedure, one may define a measurement on the state ξ that corresponds to the verifier's actions and final decision to accept or reject given this choice of r, assuming the prover behaves optimally after the coin flipping and the verifier's measurement take place. Specifically, corresponding to the Hamiltonian term $H_r = C_r^*|0^k\rangle\langle 0^k|C_r$, acceptance is represented by a projection operator Λ_r on the qubits

$$\big(\mathsf{Y}_1^{i_1}, \ldots, \mathsf{Y}_N^{i_1}\big), \ldots, \big(\mathsf{Y}_1^{i_k}, \ldots, \mathsf{Y}_N^{i_k}\big) \tag{5.9}$$

defined as follows:

$$\Lambda_r = \sum_{\substack{z\in\Sigma^k \\ z\neq 0^k}} \big(C_r^{\otimes N}\big)^* \big(\Pi_{z_1} \otimes \cdots \otimes \Pi_{z_k}\big)\big(C_r^{\otimes N}\big). \tag{5.10}$$

The probability that the verifier rejects, for a given choice of r, is therefore at least $1 - \langle \Lambda_r, \xi\rangle$. Because $\Pi_0 \leq \Delta_0$ and $\Pi_1 \leq \Delta_1$, the probability of rejection is therefore at least

$$\begin{aligned} 1 - \sum_{\substack{z\in\Sigma^k \\ z\neq 0^k}} &\big\langle \big(C_r^{\otimes N}\big)^* \big(\Delta_{z_1} \otimes \cdots \otimes \Delta_{z_k}\big)\big(C_r^{\otimes N}\big), \xi \big\rangle \\ &= \big\langle \big(C_r^{\otimes N}\big)^* \big(\Delta_0 \otimes \cdots \otimes \Delta_0\big)\big(C_r^{\otimes N}\big), \xi\big\rangle. \end{aligned} \tag{5.11}$$

By considering properties of the channel Ξ_N, we conclude that the verifier rejects with probability at least

$$\begin{aligned} &\big\langle \big(C_r^{\otimes N}\big)^* \big(\Xi_N^*(|0\rangle\langle 0|) \otimes \cdots \otimes \Xi_N^*(|0\rangle\langle 0|)\big) C_r^{\otimes N}, \xi\big\rangle \\ &\quad = \big\langle 0^k \big| \Xi_N^{\otimes k}\big(C_r^{\otimes N} \xi \big(C_r^{\otimes N}\big)^*\big)\big|0^k\big\rangle = \big\langle C_r^*|0^k\rangle\langle 0^k|C_r, \Xi_N^{\otimes k}(\xi)\big\rangle = \big\langle H_r, \rho\big\rangle. \end{aligned} \tag{5.12}$$

Here we have used the observation that

$$\Xi_N^{\otimes k}\big(C^{\otimes N}\sigma\big(C^{\otimes N}\big)^*\big) = C\,\Xi_N^{\otimes k}(\sigma) C^* \tag{5.13}$$

for every k-qubit Clifford operation C and every kN-qubit state σ, which may be verified by considering the action of Ξ_N on Hadamard, phase, and controlled-not gates.

Intuitively speaking, the argument above shows that whatever state a malicious prover sends in the first message, one can essentially decode that state with respect to a highly simplified variant of the encoding scheme (after peeling off the quantum one-time pad and discarding the trap qubits), recovering a state that would pass the Hamiltonian energy test with at least the same probability as the verifier's acceptance probability in our zero-knowledge proof system. Because this probability must be bounded away from 1 on average for any no-instance of the problem, we obtain a soundness guarantee for the proof system.

6. Zero-knowledge property of the proof system. We now prove that the proof system described in section 4 is zero-knowledge in the quantum computational sense, assuming that the commitment scheme used in the proof system is unconditionally binding and quantum computationally concealing. The proof has several steps, to be presented below, but first we will summarize the main technical goal of the proof.

Figure 6.1 shows a diagram of the interaction between the honest participants in the proof system. A cheating verifier aiming to extract knowledge from the prover might, of course, not follow the prescribed actions of the honest verifier. In particular, the cheating verifier may take a quantum register as input, store quantum information in-between its actions, and output a quantum register. Figure 6.2 illustrates such a cheating verifier interacting with the honest prover. The goal of the proof is to demonstrate that, for any cheating verifier of the form suggested by Figure 6.2, there exists an efficient simulator that implements a channel that is computationally indistinguishable from the channel implemented by the cheating verifier and prover interaction. In particular, the simulator does not have access to the witness state ρ. This will be done, through a hybrid-style argument, over the course of several steps.

Step 1: Simulating the coin-flipping protocol. By the results of [15], there must exist an efficient simulator S_1 for the interaction of V_1' with P_1. To be more precise, for S_1 being given an input of the same form as V_1', along with a uniformly chosen random string r of the length required by our proof system, the resulting action is quantum computationally indistinguishable from V_1' interacting with P_1.

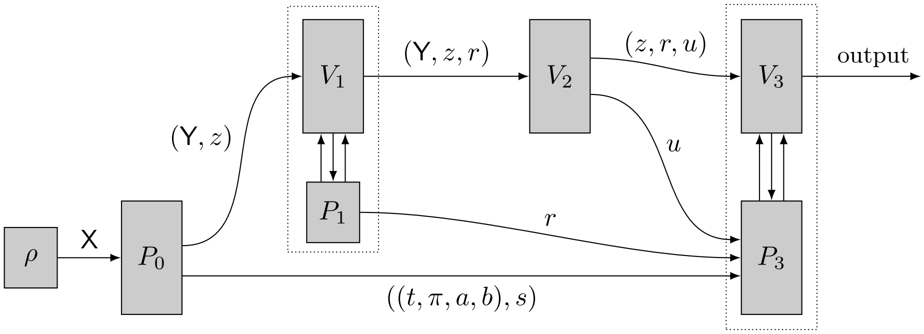


Fig. 6.1. *The interaction between honest participants. The prover's quantum witness ρ is encoded into Y together with the encoding key (t, π, a, b) by the prover's action P_0. The string z represents the prover's commitment to (π, a, b) and the string s represents random bits used by the prover to implement this commitment. The string r represents the random bits generated by the coin-flipping protocol, which is depicted within the dotted rectangle on the left. The string u represents the verifier's standard basis measurements for a subset of the qubits of Y determined by the challenge corresponding to the random string r. The classical zero-knowledge protocol is depicted within the dotted rectangle on the right.*

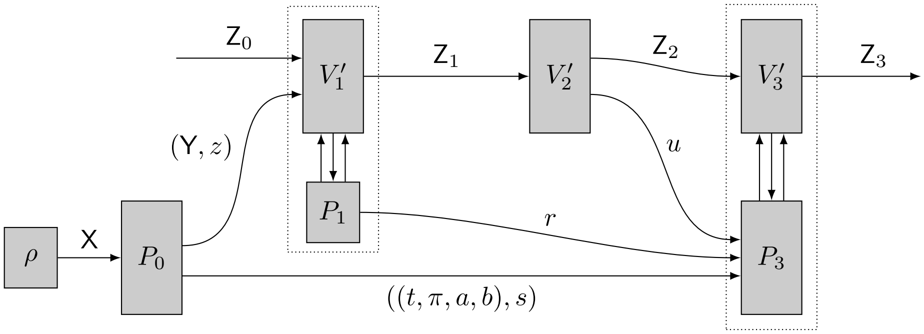


FIG. 6.2. *A potentially dishonest verifier takes an auxiliary quantum register Z_0 as input, may store quantum information (represented by registers Z_1 and Z_2), and outputs quantum information stored in register Z_3.*

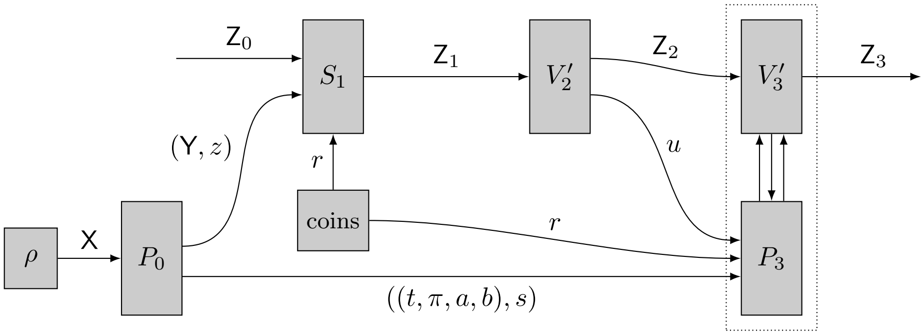


FIG. 6.3. *The interaction corresponding to the execution of the coin-flipping protocol has been replaced by a simulator S_1 along with a true random string generator (labeled* coins*).*

Figure 6.3 illustrates the process that is obtained by performing this substitution. As the simulator S_1 together with the true random string generator is computationally indistinguishable from the interaction between V'_1 and P_1, the process illustrated in Figure 6.3 is computationally indistinguishable from the process illustrated in Figure 6.2. It therefore suffices for us to prove that the process illustrated in Figure 6.3 can be efficiently simulated (without access to the witness state ρ).

Step 2: Simulating the classical zero-knowledge protocol. In the next step of the proof, we replace the interaction between a cheating verifier V'_3 and the prover P_3 in the classical zero-knowledge protocol by an efficient simulator S_3 together with the predicate Q, as is illustrated in Figure 6.4.

To describe this step in greater detail, we first observe that the prover holds an encoding key (t, π, a, b) along with the random string s it has used to commit to the tuple (π, a, b). The commitment $z = \text{commit}((\pi, a, b), s)$ is sent to the verifier, together with the encoding register Y, in the first step of the proof system. The verifier then sends a string u that, in the honest case, represents the output of a measurement of some subset of the qubits of Y with respect to the standard basis, after the transversal application of a Clifford operation depending on the random choice of r. The statement that the honest prover aims to prove in the classical zero-knowledge protocol is that there exists an encoding key (t, π, a, b) along with a string s such that $z = \text{commit}((\pi, a, b), s)$ and $Q(r, t, \pi, a, b, u) = 1$.

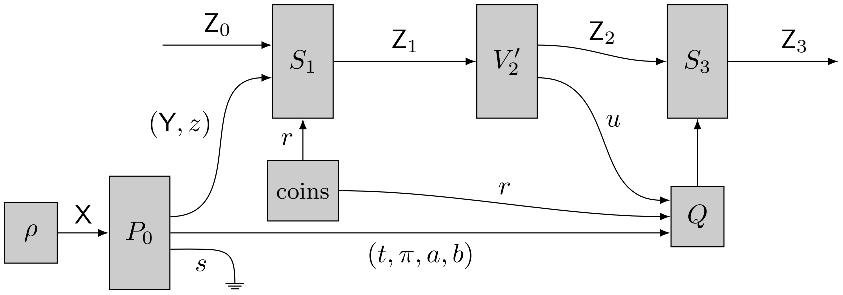


FIG. 6.4. *The interaction corresponding to the execution of the classical zero-knowledge protocol has been replaced by a simulator S_3 along with the predicate Q. It is assumed that when the output of Q is 0, the simulator S_3 behaves as the cheating verifier V_3' would when the prover aborts the proof system. The string s produced by P_0 in forming the commitment to (π, a, b) is discarded.*

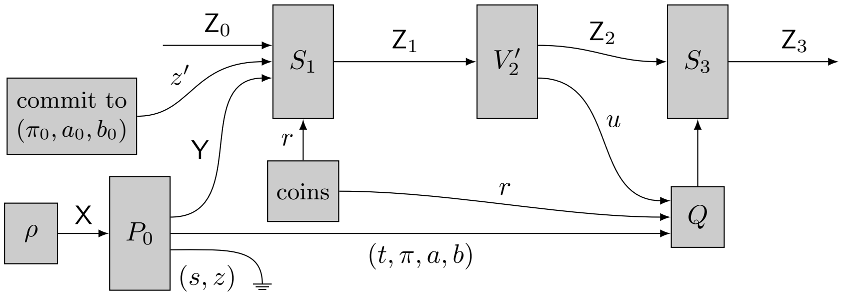


FIG. 6.5. *The commitment z given as input to S_1 has been replaced by a dummy commitment z' to a fixed tuple (π_0, a_0, b_0). Having been replaced by z', the original commitment z computed by P_0 may be considered to be discarded along with the random string s used to form that commitment.*

The honest prover always holds an encoding key (t, π, a, b) and a binary string s for which $z = \text{commit}((\pi, a, b), s)$, so we need not concern ourselves with the case that this is not so. The case that $Q(r, t, \pi, a, b, u) = 1$ therefore corresponds to a yes-instance of the classical zero-knowledge protocol, and by the assumption that the classical zero-knowledge protocol is indeed computational zero-knowledge against quantum attacks (q.v. Definition 2.6), there must therefore exist an efficient simulator S_3 that computes a transformation from Z_2 to Z_3 that is computationally indistinguishable from the one induced by the interaction between V_3' and P_3 in this case (which is signaled to S_3 when it receives a 1 input from the predicate Q). We have assumed that the prover aborts in the case $Q(r, t, \pi, a, b, u) = 0$, and so we define S_3 so that when it receives a 0 input from the predicate Q, it directly mimics whatever V_3' does in the situation that the prover aborts. It follows that the process described in Figure 6.4 is computationally indistinguishable from the one described by Figure 6.3. We observe that the string s used by P_0 to form the commitment $z = \text{commit}((\pi, a, b), s)$ can safely be discarded immediately after P_0 is run, as it is never again used in Figure 6.4.

Step 3: Eliminating the commitment. The next step is to eliminate the commitment. To this end, we consider the process described in Figure 6.5, which is identical to Figure 6.4 except that the commitment z to the tuple (π, a, b) given as input to S_1 has been replaced by a dummy commitment z' to a fixed tuple (π_0, a_0, b_0). Specifically, we take π_0 to be the identity permutation and a_0 and b_0 to be all-zero

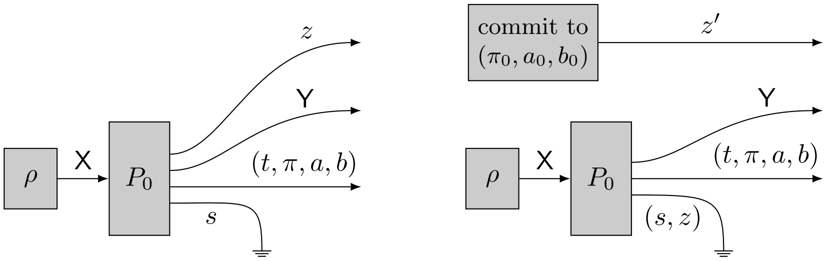


FIG. 6.6. *The processes described in Figures* 6.4 *and* 6.5 *differ only in the initial portions depicted. As the subsequent steps are quantum polynomial-time computable and identical for two processes, we find that the processes described in Figures* 6.4 *and* 6.5 *are quantum computationally indistinguishable provided that the states generated by the processes depicted are quantum computationally indistinguishable.*

strings of length $2nN$. For the sake of clarity, we note explicitly that we do not replace (π, a, b) with (π_0, a_0, b_0) as an input to the predicate Q, it is only the commitment to S_1 that is changed from Figure 6.4 to Figure 6.5. We claim that the processes described in Figures 6.4 and 6.5 are quantum computationally indistinguishable.

To verify this claim, observe first that S_1, V_2', S_3, Q, and the generation of the random coin flips r are all polynomial-time computable quantum processes. Therefore, if the processes described in Figures 6.4 and 6.5 were computationally distinguishable, the simpler processes described in Figure 6.6, which simply generate states, would also necessarily be computationally distinguishable. This is because the processes described in Figures 6.4 and 6.5 are obtained by composing the processes depicted in Figure 6.6 with exactly the same polynomial-time computable quantum process obtained from S_1, V_2', S_3, Q, and the generation of the random coin flips r.

To justify the claim made above, it therefore suffices to prove that the processes shown in Figure 6.6 are quantum computationally indistinguishable. Observe that the states generated by these two processes can be expressed as

$$\sum_z p(z)|z\rangle\langle z| \otimes \tau_z \quad \text{and} \quad \sum_z p(z)|z'\rangle\langle z'| \otimes \tau_z \tag{6.1}$$

for some choice of a distribution p and a collection of states $\{\tau_z\}$ representing both Y and (t, π, a, b). If these two states were quantum computationally distinguishable, then by convexity the states

$$|z\rangle\langle z| \otimes \tau_z \quad \text{and} \quad |z'\rangle\langle z'| \otimes \tau_z \tag{6.2}$$

would also be quantum computationally distinguishable for at least one choice of z, which directly contradicts the concealing property of the commitment scheme. We have therefore proved that the processes described in Figures 6.4 and 6.5 are computationally indistinguishable. For clarification, notice that when we replace z by z', the input to the classical zero-knowledge proof could become a negative instance. However, S_3 cannot distinguish between a yes- or no-instance here, for otherwise the hiding property of the commitment would be broken.

Before proceeding to the next step of the proof, it will be convenient to simplify the description of the process illustrated in Figure 6.5 without making any changes to the process itself. First, recall that P_0 is obtained by first performing the encoding steps described in section 4.1, followed by the formation of the commitment

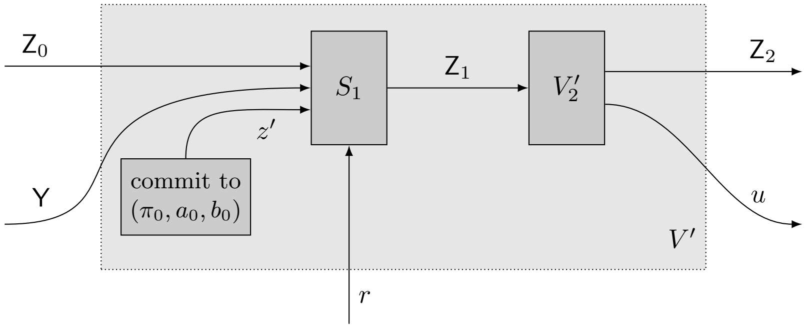


FIG. 6.7. *The commitment z' to a fixed tuple (π_0, a_0, b_0), the simulator S_1, and the dishonest verifier action V_2' may be merged into a single efficiently implementable action V' that represents an attack against the encoding scheme.*

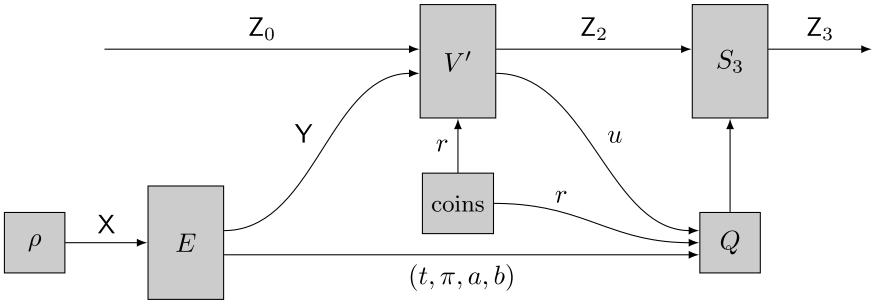


FIG. 6.8. *After making the simplifications described in the text, a process identical to the one described in Figure 6.5 is obtained. The boxed labeled E represents the encoding step performed by the prover, as described in section 4.1, and the box labeled V' denotes the merger of S_1, V_2', and the formation of the dummy commitment.*

$z = \text{commit}((\pi, a, b), s)$ (along with the random string s used to form this commitment). However, given that the commitment z and the random string s are discarded in the process described in Figure 6.5, we may as well replace P_0 with the process that performs just the encoding steps alone, without the formation of the commitment. We will name this process E, and in the interest of clarity let us state explicitly that E is the process that takes X as input and outputs Y along with (t, π, a, b), as described in section 4.1. Second, we may merge the commitment to the fixed tuple (π_0, a_0, b_0), the simulator S_1, and the cheating verifier action V_2' to form the single, efficiently implementable action V' as suggested by Figure 6.7. The process resulting from these simplifications is illustrated in Figure 6.8.

Step 4: Simulating an attack on the encoding scheme. It remains to prove that, for any efficiently implementable actions V' and S_3, the channel implemented by the process described by Figure 6.8 can be efficiently simulated. In fact, it will be possible to efficiently simulate this channel with statistical accuracy, not just in a computationally indistinguishable sense. This is not surprising: we have claimed that the computational zero-knowledge property of our proof system is based on a computationally concealing commitment scheme, and the uses of the commitment scheme have all been eliminated from consideration by the steps above.

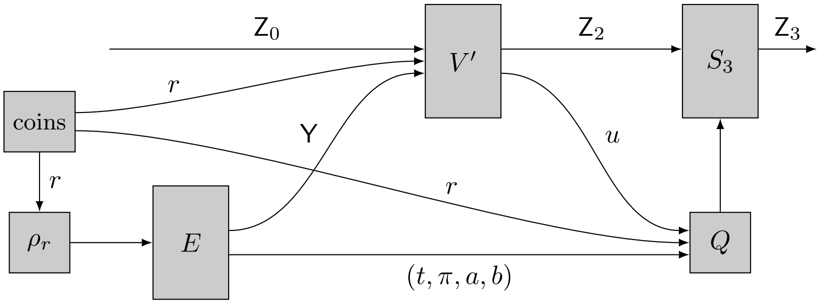


FIG. 6.9. *The simulation of the process shown in Figure* 6.8 *is nearly identical to that process, except that it uses the random string r to encode a state ρ_r that is guaranteed to pass the challenge corresponding to r, rather than encoding the witness state ρ.*

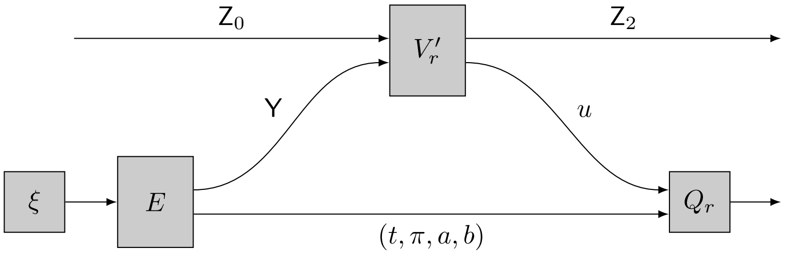


FIG. 6.10. *An arbitrary n-qubit state ξ is encoded, and the cheating verifier V' and predicate Q for a fixed choice of a string r interact as depicted. It will be proved that the channels obtained by substituting ρ and ρ_r for ξ are approximately equal.*

At this point we may describe the simulator directly: it is illustrated in Figure 6.9, and it represents the most straightforward approach to obtaining a simulator.

This simulator differs from the process described in Figure 6.8 in that it uses the output of the random string generator to choose a quantum state that, once encoded, passes the randomly selected Hamiltonian term challenge with certainty. It is trivial to efficiently prepare such a state given the string r. It remains to prove that the channel implemented by the simulator described in Figure 6.9 is indistinguishable from the channel implemented by the process described in Figure 6.8. By convexity it suffices to prove that this is so for every fixed choice of the string r. Moreover, it suffices to prove that the two processes obtained by removing S_3 from Figures 6.8 and 6.9, so that the outputs of the processes are Z_2 and the output bit of Q, are indistinguishable—for composing those two processes with the same action S_3 cannot make them more distinguishable.

With this goal in mind, consider the process described in Figure 6.10, in which an arbitrary state ξ is encoded (corresponding either to ρ or ρ_r in Figures 6.8 and 6.9), and the string r is fixed (which has been indicated by the substitution of V'_r and Q_r for V' and Q, respectively). We will prove that the channel implemented by any such process can have only a limited dependence on the state ξ.

More specifically, let us assume that ξ_0 and ξ_1 are arbitrary n-qubit states, let p_0 and p_1 denote the probabilities with which these two states would pass the challenge determined by r for an honest prover and verifier pair (i.e., $p_i = 1 - \langle \xi_i | H_r | \xi_i \rangle$, $i = 0, 1$). Let Ψ_0 and Ψ_1 denote the channels from Z_0 to Z_2 together with the output bit of the

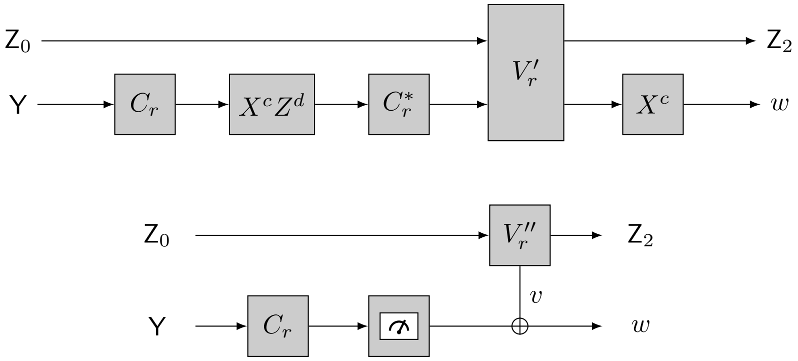


FIG. 6.11. *The prover's one-time pad merged with the cheating verifier operation V'_r. Averaging over random choices of c and d results in a process that can alternatively be described as illustrated in the lower diagram. In this process, V''_r represents a so-called* quantum instrument*, which transforms* Z_0 *into* Z_2 *and produces a classical measurement outcome. In this case, this classical measurement outcome is XORed onto the string produced by a standard basis measurement of those qubits that correspond to the Hamiltonian term given by r. (Here, one should interpret C_r and C_r^* as referring to the* transversal *application of the corresponding Clifford operation, and interpret the rightmost X^c operation in the top circuit as a classical XOR from the relevant bits of c onto the verifier's output string u.)*

predicate Q_r that are implemented by the process shown in Figure 6.10 when ξ_0 or ξ_1 is substituted for ξ, respectively.

We claim that if $|p_0 - p_1|$ is negligible, then the distance $\| \Psi_0 - \Psi_1 \|_\diamond$ is also negligible. The two steps that follow establish that this claim is true. By the assumption that the prover initially holds a witness state ρ that satisfies every Hamiltonian term with probability exponentially close to 1, this will complete the proof.

Step 5: Twirling the cheating verifier. To prove the fact suggested above regarding the channel implemented by Figure 6.10, we will naturally need to make use of the specific properties of the encoding scheme, which have not played an important role in the analysis thus far. The first step is to recognize that the effect of the prover's one time pad is to *twirl*[3] the verifier as Figure 6.11 illustrates.

More specifically, the last step of the encoding process is the quantum one-time pad: the prover independently chooses one of the Pauli operations $\mathbb{1}$, X, Z, or XZ for each qubit of Y and applies that operation, storing the randomly selected strings $a, b \in \Sigma^{2Nn}$. With respect to the Clifford operation C_r associated with the randomly selected challenge (determined by the string r), the prover computes the pair (c, d) for which it holds that

$$X^a Z^b = \big(C_r^{\otimes 2N}\big)^* X^c Z^d \big(C_r^{\otimes 2N}\big). \tag{6.3}$$

The first step in computing the predicate Q_r is the application of X^c to the string u, which is supposed to represent the outcome of a standard basis measurement of a subset of the qubits after the transversal application of C_r to the corresponding qubits in the register Y. The resulting string $w = u \oplus c_{i_1} \cdots c_{i_k}$ is then fed into the predicate R_r described previously. Merging the Clifford operation C_r^* with the cheating verifier

[3] The term twirl is commonly used in quantum information theory to describe a process whereby a symmetrization over a collection of randomly chosen unitary operations has a particular effect on a state or channel. Twirled states and channels often take on a significantly simpler form than the original state or channel prior to twirling. See examples in [2, 10].

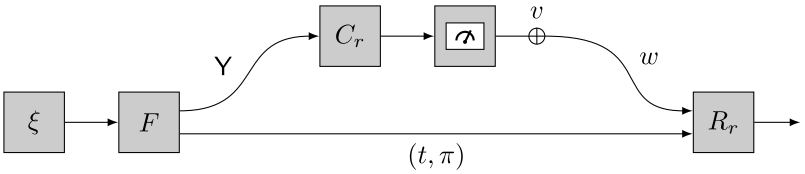


FIG. 6.12. *An XOR attack against the prover's encoding scheme without the one-time pad. The transformation F denotes the first three steps of the prover's encoding scheme.*

operation V_r', then averaging over c and d chosen uniformly at random (which is equivalent to averaging over a and b chosen uniformly at random), one obtains a process of the form illustrated in the lower diagram in Figure 6.11. In greater detail, the channel obtained by first performing Z^d on Y for d chosen uniformly at random, followed by the operation C_r^* performed on Y, followed by V_r' on $(\mathsf{Z}_0, \mathsf{Y})$, is a channel that effectively treats Y as if it were classical, so that it can be expressed as

$$\sum_{y,z} \Phi_{y,z} \otimes \Delta_{y,z}, \tag{6.4}$$

where each $\Phi_{y,z}$ is a completely positive map and $\Delta_{y,z}$ is defined as

$$\Delta_{y,z}(Y) = |z\rangle\langle y|Y|y\rangle\langle z| \tag{6.5}$$

for every $y \in \Sigma^{2nN}$ and $z \in \Sigma^{2kN}$. For a uniformly selected string c, composing this operation with the XOR operations represented by X^c yields the operation

$$\sum_{y,z} \Phi_{y,z} \otimes \sum_{c} \Delta_{y\oplus c, z\oplus c_{i_1}\cdots c_{i_k}} = \sum_{y,z} \Phi_{y,z} \otimes \sum_{c} \Delta_{c, c_{i_1}\cdots c_{i_k} \oplus y_{i_1}\cdots y_{i_k} \oplus z}, \tag{6.6}$$

which has the form suggested in Figure 6.11 (for $v = y_{i_1} \cdots y_{i_k} \oplus z$).

By the observation we have just made, it suffices to consider processes of the form described in Figure 6.12, in which an n-qubit state ξ is encoded as described by the first three steps in the prover's encoding procedure (but not including the one-time pad), the Clifford operation C_r (for a fixed choice of r) is applied transversally to the resulting register, and the qubits on which those transversal Clifford operations act are measured with respect to the standard basis. For some arbitrary but fixed string v, the XOR of the outcome of this measurement with v is fed into the predicate R_r. The process outputs a single bit, obtained by evaluating the predicate R_r.

Step 6: Encoding security under XOR attacks. Now let us return to the claim made previously, in which ξ_0 and ξ_1 represent n-qubit states, p_0 and p_1 denote the probabilities with which these two states would pass the challenge determined by r (for an honest prover and verifier pair), and Ψ_0 and Ψ_1 denote the channels implemented by the process shown in Figure 6.10 when ξ_0 or ξ_1 is substituted for ξ, respectively. If it is the case that the distributions of output bits obtained by substituting ξ_0 and ξ_1 for ξ in Figure 6.12 have negligible statistical difference, then it follows that the difference $\|\Psi_0 - \Psi_1\|_\diamond$ is also negligible. It therefore remains to argue that the distributions obtained by substituting ξ_0 and ξ_1 into Figure 6.12 have negligible statistical difference.

Before finishing off the last step of the analysis, it is helpful to consider the possible outcomes of the measurement, the definition of R_r, and the behavior of the procedure described in Figure 6.12 when $v = 0 \cdots 0$ is the all-zero string. For any choice of ξ, the measurement is guaranteed to yield a string of length $2kN$ taking the form $u_{i_1} \cdots u_{i_k}$, where $u_{i_1}, \ldots, u_{i_k} \in \Sigma^{2N}$ and $(i_1, \ldots, i_k)$ index the qubits on which C_r acts nontrivially. With respect to a particular choice of (t, π), if we define strings $y_i, z_i \in \Sigma^N$ for each $i \in \{i_1, \ldots, i_k\}$ so that $\pi(y_i z_i) = u_i$, then these two conditions will necessarily be met:

1. $y_i \in \mathcal{D}_N$ for every $i \in \{i_1, \ldots, i_k\}$, and
2. $\left\langle z_{i_1} \cdots z_{i_k} \,\middle|\, C_r^{\otimes N} \,\middle|\, t_{i_1} \cdots t_{i_k} \right\rangle \neq 0$.

Moreover, in the case that r determines a Hamiltonian term challenge, the event that $y_i \in \mathcal{D}_N^1$ for at least one index $i \in \{i_1, \ldots, i_k\}$ is equivalent to ξ passing this challenge. Thus, in the case that $v = 0 \cdots 0$, the process described in Figure 6.12 outputs the bit 1 with precisely the probability that an honest prover and verifier pair would result in acceptance, assuming the prover's initial state is ξ and r is selected as a random string determining the challenge.

Now let us assume that v is a nonzero string, and let us consider two cases: the first is that the Hamming weight $|v|_1$ of v satisfies $|v|_1 < K$ for K being the minimum Hamming weight of a nonzero codeword in $\mathcal{D}_N$, and the second case is that $|v|_1 \geq K$.

If it is the case that $|v|_1 < K$, then there are two possible ways that the value of the predicate R_r could change in comparison to the case $v = 0 \cdots 0$. In both cases, if there is a change, it must be from 1 to 0, caused by conditions 1 or 2 above being violated. The first case is that one or more bits in one of the codewords $y_{i_1}, \ldots, y_{i_k}$ are flipped, causing condition 1 to be violated. The second case is that a measurement outcome for the trap qubits is obtained that potentially violates condition 2. Note that it is not possible that condition 1 remains satisfied while the Hamiltonian term challenge condition that $y_i \in \mathcal{D}_N^1$ for at least one index $i \in \{i_1, \ldots, i_k\}$ changes, as such a change would require at least K bit-flips to cause a logical change in valid codewords. It is unimportant for the purposes of the analysis to determine the probability with which one of the two conditions becomes violated, except to observe that it is independent of ξ. (In somewhat more detail, the string v may be written as $v = v_{i_1} \cdots v_{i_k}$, and the probability that neither of the two conditions is affected is given by the probability that $\pi^{-1}(v_i)$ places no 1's within the first N bits or over a trap qubit left in a standard basis state within the second N bits, for a random choice of π and for each $i \in \{i_1, \ldots, i_k\}$.)

If it is the case that $|v|_1 \geq K$, then there is a possibility that, in comparison to the functioning of the process for $v = 0 \cdots 0$, the Hamiltonian term challenge condition that $y_i \in \mathcal{D}_N^1$ for at least one index $i \in \{i_1, \ldots, i_k\}$ could be affected. That is, v has enough Hamming weight to affect the logical values represented by the codewords $y_{i_1}, \ldots, y_{i_k}$. However, as we will show, the assumption that $|v|_1 \geq K$ necessarily leads to a negligible probability that the second condition remains satisfied—for a string v having Hamming weight K or higher, the probability that none of the traps is sprung is exponentially small. In order to argue that this is so, we require the following simple lemma.

Lemma 6.1. *Let k be a positive integer, let C be a Clifford operation on k qubits, and let $j \in \{1, \ldots, k\}$. There exists a string $t \in \{0, +, \circlearrowright\}^k$, a bit $a \in \Sigma$, and pure states $|\phi_0\rangle$ and $|\phi_1\rangle$ on $j-1$ qubits and $k-j$ qubits, respectively, so that*

$$C|t\rangle = |\phi_0\rangle|a\rangle|\phi_1\rangle. \tag{6.7}$$

Equivalently, there is a choice of t so that the jth qubit of $C|t\rangle$ is left in a standard basis state.

Proof. The lemma is equivalent to the existence of a string t so that $|t\rangle$ is an eigenvector of the operator

$$C^{*}\big(\mathbb{1}^{\otimes(j-1)} \otimes Z \otimes \mathbb{1}^{\otimes(k-j)}\big)C. \tag{6.8}$$

As the Clifford group normalizes the Pauli group, the operator (6.8) is a scalar multiple of a tensor product of Pauli operators and identity operators. The lemma follows from the observation that t may be chosen so that each $|t_1\rangle, \ldots, |t_k\rangle$ is an eigenvector of the Pauli operator in the corresponding position. ∎

By this lemma, one finds that for a random choice of $t \in \{0, +, \circlearrowleft\}^{kN}$, and for any k-qubit Clifford operation C applied transversally to $|t\rangle$, each qubit is left in a standard basis state with probability at least 3^{-k}, and for any choice of N or fewer qubits acted on by distinct Clifford operations these events are independent. In greater detail, if the qubits

$$\big(\mathsf{Z}_1^1, \ldots, \mathsf{Z}_1^k\big), \ldots, \big(\mathsf{Z}_N^1, \ldots, \mathsf{Z}_N^k\big) \tag{6.9}$$

are initialized to the state $|t\rangle$ for $t \in \{0, +, \circlearrowleft\}^{kN}$ chosen uniformly at random, and the k-qubit Clifford operation C is applied independently to each k-tuple of qubits, then each qubit is left in a standard basis state with probability at least 3^{-k}, and the states of the k-tuples of qubits are independent.

Now we return to the analysis for a string v of length $2kN$ having Hamming weight at least K. By virtue of the fact just mentioned, it is straightforward to obtain a negligible upper bound on the probability for the process described in Figure 6.12 to output 1. As this event requires that a random choice of the permutation π leaves none of the 1-bits of v in positions corresponding to trap qubits left in standard basis states by the transversal action of C_r, we find that the probability to output 1 is exponentially small in K. In particular, this probability is at most

$$\left(1 - \frac{1}{3^{k+1}}\right)^{K/k} = \exp(-\varepsilon(k)K), \tag{6.10}$$

where $\varepsilon(k)$ denotes a positive real number depending on k (which the reader will recall is constant and may be taken to be $k = 5$) but not K.

From a consideration of the two cases just presented, we may conclude the following. Suppose as before that ξ_0 and ξ_1 are n-qubit states that may be substituted for ξ in Figure 6.12, and that the probabilities p_0 and p_1 for these states to pass the challenge determined by a fixed choice of r have negligible difference. Let us write $q_0(v)$ and $q_1(v)$, respectively, to denote the probability that the process described in Figure 6.12 outputs 1. As noted before, it holds that $p_0 = q_0(0\cdots0)$ and $p_1 = q_1(0\cdots0)$. For any choice of v satisfying $|v|_1 < K$, we have that $q_0(v) = \beta(v)q_0(0\cdots0)$ and $q_1(v) = \beta(v)q_1(0\cdots0)$ for $\beta(v) \in (0,1)$ that is independent of ξ_0 and ξ_1. Finally, for any choice of v satisfying $|v|_1 \geq K$, we have that $q_0(v)$ and $q_1(v)$ are both negligible. It therefore follows that the difference $|q_0(v) - q_1(v)|$ is negligible in all cases, which completes the proof.

7. Conclusion.

This paper gives a zero-knowledge proof system for any problem in QMA assuming the existence of a quantum computationally concealing and unconditionally binding commitment scheme. Such a commitment scheme can be obtained assuming quantum-secure one-way permutations [1] (or injections more generally). It also appears feasible to use a commitment scheme with an *interactive* commit phase, such as Naor's two-message commitment scheme [47] based on a pseudorandom generator. This would reduce the zero-knowledge protocol to a quantum-secure one-way function [34, 50, 61], and we leave this for further verification. We conclude with a few open questions and future directions.

1. Our proof system inherits the soundness error of straightforward verification procedure for the local Clifford–Hamiltonian problem, which is to randomly select a Hamiltonian term and perform a measurement corresponding to it. When an arbitrary QMA problem is reduced to the local Hamiltonian problem, the resulting soundness error may potentially be large (polynomially bounded away from 1). Can a zero-knowledge proof system for any QMA problem be obtained with small soundness error while maintaining the other features of our proof system (e.g., constant round of communications)?

   We note that if a prover has polynomially many copies of a valid quantum witness, then a parallel repetition of our proof system may yield a constant round zero-knowledge proof system having small soundness error for any QMA problem—but this would require a parallel repetition result concerning zero-knowledge proof systems for NP secure against quantum attacks. Analogous results for zero-knowledge proofs for NP against classical attacks are known [18, 24], but they involve sophisticated rewinding arguments for which known quantum rewinding techniques do not seem to be applicable.
2. Are there natural formalizations of *proofs of quantum knowledge*? Roughly speaking, one would expect such a notion to require that whenever a prover is able to prove the validity of a statement, one could construct a knowledge extractor that can extract a quantum witness given access to such a prover. (Unruh [53] has formulated a notion of *quantum proofs of knowledge* that refers to the extraction of a classical witness from a possibly malicious quantum prover, but here we are referring to the extraction of a quantum witness.) It seems plausible that our proof system could be adapted to such a notion, although we have not investigated this in depth.
3. Finally, we make one further remark on an abstract view of our proof system. Classically speaking, one can imagine a "commit-and-open" primitive where a sender commits to a message m, and later opens sufficient information so that a receiver can test a property $\mathcal{P}(\cdot)$ on m, and nothing more. For example, $\mathcal{P}$ can be an NP-relation $R(x,\cdot)$ that checks if message m is a valid witness. This can be implemented easily by a standard commitment scheme and during the opening phase, the sender and receiver run a zero-knowledge proof of $R(x,m) = 1$ instead of the standard opening. Our proof system, which combines a commitment scheme and classical zero-knowledge proofs for NP, can be viewed as a quantum analogue. Namely, we commit to a witness state and open just enough information to verify that some reduced density of the witness state falls into a specific subspace. We can only deal with properties of a very special form, and it is an interesting direction for future work to generalize and find applications of this sort of primitive.

Acknowledgments. We thank Michael Beverland, Sevag Gharibian, David Gosset, Yi-Kai Liu, and Bei Zeng for helpful conversations.

REFERENCES

[1] M. Adcock and R. Cleve, *A quantum Goldreich-Levin theorem with cryptographic applications*, in Proceedings of the 19th International Symposium on Theoretical Aspects of Computer Science, Lecture Notes in Comput. Sci. 2285 , Springer, Berlin, 2002, pp. 323–334, https://doi.org/10.1007/3-540-45841-7_26.

[2] D. Aharonov, M. Ben-Or, and E. Eban, *Interactive proofs for quantum computations*, in Innovations in Computer Science, ACM, New York, 2010, pp. 453–469.

[3] D. Aharonov, A. Kitaev, and N. Nisan, *Quantum circuits with mixed states*, in Proceedings of the 30th Annual ACM Symposium on Theory of Computing, ACM, New York, 1998, pp. 20–30, https://doi.org/10.1145/276698.276708.

[4] A. Ambainis, M. Mosca, A. Tapp, and R. de Wolf, *Private quantum channels*, in Proceedings of the 41st Annual IEEE Symposium on Foundations of Computer Science, IEEE Computer Society, Los Alamitos, CA, 2000, pp. 547–553, https://doi.org/10.1109/SFCS.2000.892142.

[5] H. Barnum, C. Crépeau, D. Gottesman, A. Smith, and A. Tapp, *Authentication of quantum messages*, in Proceedings of the 43th Annual IEEE Symposium on Foundations of Computer Science, IEEE Computer Society, Los Alamitos, CA, 2002, pp. 449–458, https://doi.org/10.1109/SFCS.2002.1181969.

[6] M. Ben-Or, C. Crépeau, D. Gottesman, A. Hassidim, and A. Smith, *Secure multiparty quantum computation with (only) a strict honest majority*, in Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science, IEEE Computer Society, Los Alamitos, CA, 2006, pp. 249–260, https://doi.org/10.1109/FOCS.2006.68.

[7] M. Ben-Or, O. Goldreich, S. Goldwasser, J. Håstad, J. Kilian, S. Micali, and P. Rogaway, *Everything provable is provable in zero-knowledge*, in Advances in Cryptology – CRYPTO 1988, Lecture Notes in Comput. Sci. 403, Springer, Berlin, 1990, pp. 37–56, https://doi.org/10.1007/0-387-34799-2_4.

[8] M. Blum, *Coin flipping by telephone a protocol for solving impossible problems*, ACM SIGACT News, 15 (1983), pp. 23–27, https://doi.org/10.1145/1008908.1008911.

[9] S. Bravyi, *Efficient algorithms for a quantum analogue of* 2*-SAT*, Contemp. Math., 536 (2011), pp. 33–48, https://doi.org/10.1090/conm/536/10552.

[10] A. Broadbent, *How to verify a quantum computation*, Theory Comput., 14 (2018), pp. 1–37.

[11] A. Broadbent, G. Gutoski, and D. Stebila, *Quantum one-time programs*, in Advances in Cryptology – CRYPTO 2013, Lecture Notes in Comput. Sci. 8043, Springer, Berlin, 2013, pp. 344–360, https://doi.org/10.1007/978-3-642-40084-1_20.

[12] A. Broadbent, Z. Ji, F. Song, and J. Watrous, *Zero-knowledge proof systems for QMA*, in Proceedings of the 2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS), IEEE, Piscataway, NJ, 2016, pp. 31–40.

[13] A. Chailloux and I. Kerenidis, *Increasing the power of the verifier in quantum zero knowledge*, in IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, Wadern, Germany, Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2008.

[14] I. Damgård, S. Fehr, and L. Salvail, *Zero-knowledge proofs and string commitments withstanding quantum attacks*, in Advances in Cryptology – CRYPTO 2004, Lecture Notes in Comput. Sci. 3152, Springer, Berlin, 2004, pp. 254–272, https://doi.org/10.1007/978-3-540-28628-8_16.

[15] I. Damgård and C. Lunemann, *Quantum-secure coin-flipping and applications*, in Advances in Cryptology – ASIACRYPT 2009, Lecture Notes in Comput. Sci. 5912 , Springer, Berlin, 2009, pp. 52–69, https://doi.org/10.1007/978-3-642-10366-7_4.

[16] F. Dupuis, J. B. Nielsen, and L. Salvail, *Secure two-party quantum evaluation of unitaries against specious adversaries*, in Advances in Cryptology – CRYPTO 2010, Lecture Notes in Comput. Sci. 6223, Springer, Berlin, 2010, pp. 685–706, https://doi.org/10.1007/978-3-642-14623-7_37.

[17] F. Dupuis, J. B. Nielsen, and L. Salvail, *Actively secure two-party evaluation of any quantum operation*, in Advances in Cryptology – CRYPTO 2012, Lecture Notes in Comput. Sci. 7417, Springer, Berlin, 2012, pp. 794–811, https://doi.org/10.1007/978-3-642-32009-5_46.

[18] U. Feige and A. Shamir, *Zero knowledge proofs of knowledge in two rounds*, in Advances in Cryptology – CRYPTO 1989, Lecture Notes in Comput. Sci. 435, Springer, New York, 1990, pp. 526–544, https://doi.org/10.1007/0-387-34805-0_46.

[19] J. F. Fitzsimons, M. Hajdušek, and T. Morimae, *Post hoc verification with a single prover*, Phys. Rev. Lett., 120 (2018), 040501, https://doi.org/10.1103/PhysRevLett.120.040501.

[20] J. Fitzsimons, Z. Ji, T. Vidick, and H. Yuen, *Quantum proof systems for iterated exponential time, and beyond*, in Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, 2019, pp. 473–480, https://doi.org/10.1145/3313276.3316343.

[21] C. A. Fuchs and A. Peres, *Quantum-state disturbance versus information gain: Uncertainty relations for quantum information*, Phys. Rev. A(3), 53 (1996), pp. 2038–2045, https://doi.org/10.1103/PhysRevA.53.2038.

[22] O. Goldreich, *Foundations of Cryptography* I: *Basic Tools*, Cambridge University Press, Cambridge, 2001, https://doi.org/10.1017/CBO9780511546891.

[23] O. Goldreich, *Foundations of Cryptography* II: *Basic Applications*, Cambridge University Press, Cambridge, 2004, https://doi.org/10.1017/CBO9780511721656.

[24] O. Goldreich and A. Kahan, *How to construct constant-round zero-knowledge proof systems for NP*, J. Cryptology, 9 (1996), pp. 167–189, https://doi.org/10.1007/BF00208001.

[25] O. Goldreich, S. Micali, and A. Wigderson, *How to play ANY mental game*, in Proceedings of the 19th Annual ACM Symposium on Theory of Computing, 1987, pp. 218–229, https://doi.org/10.1145/28395.28420.

[26] O. Goldreich, S. Micali, and A. Wigderson, *Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems*, Journal of the ACM, 38 (1991), pp. 690–728, https://doi.org/10.1145/116825.116852.

[27] O. Goldreich and Y. Oren, *Definitions and properties of zero-knowledge proof systems*, J. Cryptology, 7 (1994), pp. 1–32, https://doi.org/10.1007/BF00195207.

[28] S. Goldwasser, S. Micali, and C. Rackoff, *The knowledge complexity of interactive proof systems*, SIAM J. Comput., 18 (1989), pp. 186–208, https://doi.org/10.1137/0218012.

[29] S. Goldwasser and M. Sipser, *Private coins versus public coins in interactive proof systems*, in Proceedings of the 18th Annual ACM Symposium on Theory of Computing, ACM, New York, 1986, pp. 59–68, https://doi.org/10.1145/12130.12137.

[30] D. Gosset and D. Nagaj, *Quantum 3-SAT is QMA$_1$-complete*, SIAM J. Comput., 45 (2016), pp. 1080–1128, https://doi.org/10.1137/140957056.

[31] D. Gottesman, *The Heisenberg representation of quantum computers*, in Group 22: Proceedings of the 22nd International Colloquium on Group Theoretical Methods in Physics, International Press, Cambridge, MA, 1998, pp. 32–43.

[32] S. Hallgren, A. Kolla, P. Sen, and S. Zhang, *Making classical honest verifier zero knowledge protocols secure against quantum attacks*, in Proceedings of the 35th International Colloquium on Automata, Languages and Programming, Part II, Lecture Notes in Comput. Sci. 5126, Springer, Berlin, 2008, pp. 592–603, https://doi.org/10.1007/978-3-540-70583-3_48.

[33] S. Hallgren, A. Smith, and F. Song, *Classical cryptographic protocols in a quantum world*, Int. J. Quantum Inf., 13 (2015), 1550028, https://doi.org/10.1142/S0219749915500288.

[34] J. Håstad, R. Impagliazzo, L. A. Levin, and M. Luby, *A pseudorandom generator from any one-way function*, SIAM J. Comput., 28 (1999), pp. 1364–1396, https://doi.org/10.1137/S0097539793244708.

[35] R. Impagliazzo, *A personal view of average-case complexity*, in Proceedings of 10th Annual IEEE Structure in Complexity Theory Conference, IEEE Compter Society, Los Alamitos, CA, 1995, pp. 134–147, https://doi.org/10.1109/SCT.1995.514853.

[36] Z. Ji, *Compression of quantum multi-prover interactive proofs*, in Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, ACM, New York, 2017, pp. 289–302, https://doi.org/10.1145/3055399.3055441.

[37] J. Kempe, A. Kitaev, and O. Regev, *The complexity of the local Hamiltonian problem*, SIAM J. Comput., 35 (2006), pp. 1070–1097, https://doi.org/10.1137/S0097539704445226.

[38] J. Kempe and O. Regev, 3-*local Hamiltonian is QMA-complete*, Quantum Inf. Comput., 3 (2003), pp. 258–264, http://portal.acm.org/citation.cfm?id=2011541.

[39] A. Y. Kitaev, *Quantum computations: Algorithms and error correction*, Russian Math. Surveys, 52 (1997), pp. 1191–1249, http://stacks.iop.org/0036-0279/52/i=6/a=R02.

[40] A. Y. Kitaev, A. H. Shen, and M. N. Vyalyi, *Classical and Quantum Computation*, Grad. Stud. Math. 47, Amer. Math. Soc., Providence, RI, 2002.

[41] Y.-K. Liu, *Consistency of local density matrices is QMA-complete*, in Proceedings of the 9th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems, APPROX 2006 and 10th International Workshop on Randomization and Computation, RANDOM 2006, Lecture Notes in Comput. Sci. 4110, Springer, Berlin, 2006, pp. 438–449, https://doi.org/10.1007/11830924_40.

[42] C. Lunemann and J. B. Nielsen, *Fully simulatable quantum-secure coin-flipping and applications*, in Progress in Cryptology – AFRICACRYPT 2011, Lecture Notes in Comput. Sci. 6737, Springer, Berlin, 2011, pp. 21–40, https://doi.org/10.1007/978-3-642-21969-6_2.

[43] C. Marriott and J. Watrous, *Quantum Arthur-Merlin games*, Comput. Complexity, 14 (2005), pp. 122–152, https://doi.org/10.1007/s00037-005-0194-x.

[44] T. Morimae, M. Hayashi, H. Nishimura, and K. Fujii, *Quantum Merlin-Arthur with Clifford Arthur*, Quantum Inf. Comput., 15 (2015), pp. 1420–1430.

[45] T. Morimae, D. Nagaj, and N. Schuch, *Quantum proofs can be verified using only single-qubit measurements*, Phys. Rev. A(3), 93 (2016), 022326, https://doi.org/10.1103/PhysRevA.93.022326.

[46] D. Nagaj, P. Wocjan, and Y. Zhang, *Fast amplification of QMA*, Quantum Inf. Comput., 9 (2009), pp. 1053–1068.

[47] M. Naor, *Bit commitment using pseudorandomness*, J. Cryptology, 4 (1991), pp. 151–158, https://doi.org/10.1007/BF00196774.

[48] M. Nielsen and I. Chuang, *Quantum Computation and Quantum Information*, Cambridge University Press, Cambridge, 2000.

[49] P. W. Shor, *Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer*, SIAM J. Comput., 26 (1997), pp. 1484–1509, https://doi.org/10.1137/S0097539795293172.

[50] F. Song, *A note on quantum security for post-quantum cryptography*, in Proceedings of the 6th International Workshop on Post-Quantum Cryptography, Lecture Notes in Comput. Sci. 8772, Springer, Cham, Switzerland, 2014, pp. 246–265, https://doi.org//10.1007/978-3-319-11659-4_15.

[51] A. Steane, *Multi-particle interference and quantum error correction*, Proc. Royal Soc. A, 452 (1996), pp. 2551–2577, https://doi.org/10.1098/rspa.1996.0136.

[52] T. Vidick and T. Zhang, *Classical Zero-Knowledge Arguments for Quantum Computations*, preprint, arXiv:1902.05217, 2019.

[53] D. Unruh, *Quantum proofs of knowledge*, in Advances in Cryptology – EUROCRYPT 2012, Lecture Notes in Comput. Sci. 7237, Springer, Heidelberg, Germany, 2012, pp. 135–152.

[54] J. van de Graaf, *Towards a Formal Definition of Security for Quantum Protocols*, Ph.D. thesis, Université de Montréal, Montreal, 1997.

[55] J. Watrous, *Limits on the power of quantum statistical zero-knowledge*, in Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science, IEEE Computer Society, 2002, pp. 459–468, https://doi.org/10.1109/SFCS.2002.1181970.

[56] J. Watrous, *PSPACE has constant-round quantum interactive proof systems*, Theoret. Comput. Sci., 292 (2003), pp. 575–588, https://doi.org/10.1016/S0304-3975(01)00375-9.

[57] J. Watrous, *Quantum computational complexity*, in Encyclopedia of Complexity and Systems Science, Springer, New York, 2009, pp. 7174–7201, https://doi.org/10.1007/978-0-387-30440-3_428.

[58] J. Watrous, *Zero-knowledge against quantum attacks*, SIAM J. Comput., 39 (2009), pp. 25–58, https://doi.org/10.1137/060670997.

[59] J. Watrous, *Guest column: An introduction to quantum information and quantum circuits*, ACM SIGACT News, 42 (2011), pp. 52–67, https://doi.org/10.1145/1998037.1998053.

[60] W. K. Wootters and W. H. Zurek, *A single quantum cannot be cloned*, Nature, 299 (1982), pp. 802–803, https://doi.org/10.1038/299802a0.

[61] M. Zhandry, *How to construct quantum random functions*, in Proceedings of the 53rd Annual IEEE Symposium on Foundations of Computer Science, IEEE, Piscataway, NJ, 2012, pp. 679–687, https://doi.org/10.1109/FOCS.2012.37.