Quantum Proof Systems for Iterated Exponential Time, and Beyond*

Joseph Fitzsimons
Horizon Quantum Computing
Singapore

Zhengfeng Ji
University of Technology Sydney
Australia

Thomas Vidick
California Institute of Technology
USA

Henry Yuen
University of Toronto
Canada

ABSTRACT

We show that any language solvable in nondeterministic time $\exp(\exp(\cdots \exp(n)))$, where the number of iterated exponentials is an arbitrary function $R(n)$, can be decided by a multiprover interactive proof system with a classical polynomial-time verifier and a constant number of quantum entangled provers, with completeness 1 and soundness $1 - \exp(-C \exp(\cdots \exp(n)))$, where the number of iterated exponentials is $R(n) - 1$ and $C > 0$ is a universal constant. The result was previously known for $R = 1$ and $R = 2$; we obtain it for any time-constructible function R.

The result is based on a compression technique for interactive proof systems with entangled provers that significantly simplifies and strengthens a protocol compression result of Ji (STOC'17). As a separate consequence of this technique we obtain a different proof of Slofstra's recent result on the uncomputability of the entangled value of multiprover games (Forum of Mathematics, Pi 2019).

Finally, we show that even minor improvements to our compression result would yield remarkable consequences in computational complexity theory and the foundations of quantum mechanics: first, it would imply that the class MIP* contains all computable languages; second, it would provide a negative resolution to a multipartite version of Tsirelson's problem on the relation between the commuting operator and tensor product models for quantum correlations.

CCS CONCEPTS

• **Theory of computation → Quantum complexity theory**.

KEYWORDS

Quantum multiprover interactive proofs, quantum entanglement, quantum correlations, self-testing

*The full version of this paper can be found at [10].

STOC '19, June 23–26, 2019, Phoenix, AZ, USA

ACM ISBN 978-1-4503-6705-9/19/06…$15.00
https://doi.org/10.1145/3313276.3316343

ACM Reference Format:
Joseph Fitzsimons, Zhengfeng Ji, Thomas Vidick, and Henry Yuen. 2019. Quantum Proof Systems for Iterated Exponential Time, and Beyond. In *Proceedings of the 51st Annual ACM SIGACT Symposium on the Theory of Computing (STOC '19), June 23–26, 2019, Phoenix, AZ, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3313276.3316343

1 INTRODUCTION

The combined study of interactive proof systems and quantum entanglement has led to multiple discoveries at the intersection of theoretical computer science and quantum physics. On the one hand, the study has revealed that quantum entanglement, a fundamental physical phenomenon, can be harnessed in interactive protocols to accomplish an array of novel computing and cryptographic tasks, ranging from the certified generation of random numbers to improved protocols for multi-party cryptography and classically-verifiable quantum computation. On the other hand, interactive proof systems, a cornerstone of modern complexity theory and cryptography, have provided a powerful lens through which to examine the counter-intuitive properties of quantum entanglement. This lens has enabled researchers to develop sophisticated ways of exploring phenomena such as the monogamy of entanglement, embezzlement of quantum states, and more.

We investigate a central question in this area: what is the *computational complexity* of interactive proof systems with multiple quantum entangled provers? The starting point for this question dates back to the seminal result of Babai, Fortnow and Lund, who showed that the set of languages that can be decided by a (classical) multiprover interactive proof system, denoted by MIP, equals the set of languages that can be decided in nondeterministic exponential time (denoted by NEXP) [3]. It is not difficult to show that MIP $\subseteq$ NEXP, but the reverse containment is nontrivial and the work of [3] was an influential stepping stone towards the PCP Theorem [1, 2].

A long line of work, starting with that of Cleve et al. [6], has explored the setting of interactive proof systems where a classical polynomial-time verifier interacts with provers that are *quantum* and may share *entanglement*. This gives rise to the complexity class MIP*, which is the set of all languages decidable by such proof systems.[1] Quantum entanglement is a resource that allows isolated parties to generate correlations that cannot be reproduced by (classical) shared randomness alone; however, entanglement does not allow for instantaneous communication. A central question

[1]The * in MIP* refers to the entanglement.

raised by [6] is whether MIP* = MIP, or equivalently, whether MIP* = NEXP.

A richer set of correlations gives additional power to provers in an interactive proof system, making the relationship between MIP* and MIP non-obvious. On the one hand, a multiprover interactive proof system that is sound against "cheating" classical provers may no longer be sound against "cheating" entangled provers; this prevents one from automatically concluding that MIP ⊆ MIP*. On the other hand, a proof system may require "honest provers" to use quantum entanglement in order to satisfy the completeness property. Entanglement thus allows one to consider a broader set of protocols, putting in question the inclusion MIP* ⊆ MIP.

The quest to pin down the computational power of proof systems with entangled provers has led to a number of surprising discoveries. The best lower bound that is currently known is that NEXP = MIP ⊆ MIP*, a nontrivial result that follows from a more general technique of "immunization" of classical proof systems against malicious entangled provers [14, 24]. There is strong evidence that this lower bound can be improved; a recent result [23] shows that languages in QMAEXP – the exponential time version of QMA – have entangled proof systems under randomized Karp reductions. If the same were true under deterministic reductions, this would imply that QMAEXP ⊆ MIP*. Assuming that the exponential time versions of NP and QMA are different, this would unconditionally separate MIP from MIP*.

Surprisingly, there are no meaningful upper bounds known for MIP*. In a striking result, Slofstra gave evidence that the complexity of MIP* might be very different from its classical counterpart: he proved that it is *undecidable* to determine whether an interactive proof system with two provers has an entangled strategy that is accepted with probability 1 (in other words, whether there is a *perfect* entangled strategy) [27, 28]. In contrast, the complexity of determining whether such a proof system has a perfect *classical* strategy is exactly equal to NEXP. Another recent result of Ji [17] points in the same direction: Ji showed that any language in nondeterministic doubly-exponential time can be decided by a classical polynomial-time verifier interacting with $k = 11$ provers, with completeness 1 and soundness that is exponentially close to 1.[2]

In this work we explore the expanse of complexity-space that entangled-prover interactive proof systems can reach. We focus on the "small gap" regime: we consider the problem of distinguishing between the cases when a multiprover proof system has a perfect entangled strategy, or when all entangled provers are rejected with probability at least ε, where ε is a quantity that may go to 0 quickly with the size of the verifier in the proof system. Our results smoothly interpolate between the hardness result of [14, 17, 24] and Slofstra's undecidability result. For clarity we restrict our attention to *hyper-exponential* time functions, i.e. time-constructible functions of the form $t(n) = \Lambda_R(n)$, where $\Lambda_0(n) = n$ and for any integer-valued function $R = R(n) \geq 0$, $\Lambda_{R+1}(n) = 2^{\Lambda_R(n)}$. For a multiprover game $\mathcal{G}$, the *entangled value* $\omega^*(\mathcal{G})$ is the maximum success probability of quantum provers sharing entanglement in the game.

[2] Due to the vanishing gaps neither Slofstra's nor Ji's result directly separates MIP* from MIP, though they do separate the zero-error and exponentially-small error variants respectively: MIP = NEXP for all gaps. Furthermore, since the provers in an MIP protocol are assumed to be deterministic, the error cannot be smaller than inverse exponential.

Theorem 1.1. *Let $k \geq 15$ be an integer. Let $t : \mathbb{N} \to \mathbb{N}$ be a hyper-exponential function. There are universal constants $C, c > 0$ such that given the description of polynomial-size circuits for the verifier in a k-prover game $\mathcal{G}$, the problem of distinguishing between*

$$\omega^*(\mathcal{G}) = 1 \qquad or \qquad \omega^*(\mathcal{G}) \leq 1 - \frac{C}{(t(n))^c}$$

is hard for nondeterministic $2^{t(n)}$ time.

The "base case" for Theorem 1.1, corresponding to $R = 0$ and $t(n) = n$, is the result that NEXP ⊆ MIP* [14, 24], where MIP* is the class of languages that can be decided using an entangled-prover interactive proof system, with completeness $\frac{2}{3}$ and soundness $\frac{1}{3}$ (the completeness-soundness gap can be amplified from inverse polynomial to constant using hardness amplification techniques [4]). The first step, $R = 1$ and $t(n) = 2^n$, follows from Ji's result [17] mentioned earlier, albeit using a game with $k = 11$ provers.

A corollary of both our and Ji's earlier result is that the "honest strategy" for the provers (i.e. those satisfying the completeness property) in the games constructed through the reduction from Theorem 1.1 provably require the provers to share entanglement. Moreover, it is often possible to obtain lower bounds on the dimension of entanglement required to achieve close to optimal success probability; this is the case for our result, as described below.

The proof of Theorem 1.1 is based on a compression technique that significantly simplifies and extends the approach pioneered in [17]. Our generalized compression result can be recursively composed with itself in order to obtain the statement of Theorem 1.1 for any integer-valued $R(n) \geq 1$.

The starting point of the compression approach of [17] is to extend the notion of a *history state*. The concept of a history state was first introduced by Kitaev in order to efficiently encode any polynomial-time quantum computation as the ground state of a local Hamiltonian, in a way that is also efficiently verifiable [22]. The compression result of [17] as well as the one in this paper constructs a game to verify history states that encode the execution of a (different) multiprover game, including the actions of the provers (which in general are not efficiently computable). The verification is performed by executing a "games" version of the traditional verification procedure for history states, that consists in randomly sampling a local Hamiltonian term and measuring its energy.

There are two key ideas behind our generalized compression technique. The first is to ensure that the game $\mathcal{G}$ that verifies the history state of a multiprover game $\mathcal{G}'$ can be executed using a circuit that is logarithmic in the size of $\mathcal{G}'$, provided that $\mathcal{G}'$ is specified in a sufficiently uniform and succinct manner. The second idea is to compose the first idea with itself, i.e. consider the history state for the computation performed by the history state verification procedure. At this point there are a number of delicate issues to consider, including identifying the right model for specifying verifiers, verifiers of verifiers, etc.; we give more details in Section 2.

On a more informal note, we observe that the kind of compression achieved here may be thought of as a "bootstrapping" of Kitaev's history state technique, in a similar sense to the composition technique from the PCP literature that "bootstraps" an efficient

PCP into a super-efficient one.[3] The fact that history states are ground states of local Hamiltonians is a statement about the local verifiability of arbitrary quantum computation. Our result goes further by making the following observations. First, not only is the verification procedure local, it is also exceedingly efficient — it can be executed in time logarithmic in the size of the original computation. Second, it is possible to consider a history state for the verification procedure itself. Third, and most strikingly, the latter history state can be verified with the same complexity as the verification procedure, without reference to the size of the original computation. This last step crucially relies on *rigidity* properties of entanglement which acts as a "leash" on quantum systems. It is sufficient to only control the leash-holder: if the leash-holder manages to hold the dog tightly enough, then there is no longer any reason to worry about the (hyper-exponential-size) dog itself.

It is worth noting that such "PCP composition on steroids" has no classical analogue. A classical PCP verifier runs in polynomial time and uses polynomially many random bits to verify an exponentially long proof. Encoding the computation performed by such a verifier in a way that can be verified using, say, a classical multiprover interactive proof system, again requires a polynomial-sized verifier flipping polynomially many bits. This is because the only way to "verify the verification procedure" is to, at least with some probability, access some of the original proof bits. In the quantum case, it is possible to leverage entanglement between provers to avoid the need for the "inner" verifier (to borrow some terminology from the PCP literature) to make any query at all to the original proof qubits.

Before proceeding we formulate another consequence of compression that highlights the versatility of our approach. As already mentioned, it was recently shown by Slofstra that the problem of determining whether a given multiprover game has a perfect entangled strategy is undecidable. Slofstra's result proceeds by an ingenious (and intricate) reduction to the word problem in finitely presented groups, which is known to be undecidable. The proof of the latter itself involves a sophisticated embedding of the computation of an arbitrary Turing Machine (in fact, a Minsky machine) in an instance of the word problem in a suitable finitely presented group [5, 21, 25].

We give a different proof of Slofstra's undecidability result, by directly constructing an interactive proof system from a Turing machine. Arguably, our result provides an intuitive reason for *why* the problem is undecidable, showing in a precise sense how smaller and smaller gaps can be leveraged to verify that the provers are performing an increasingly complex computation. More precisely, the main idea for our proof is to design a family of games $\{\mathcal{G}_n\}_{n\geq 1}$ such that for any $n \geq 1$ the verifier in the game $\mathcal{G}_n$ verifies if a Turing machine provided as input halts within n steps, and if it does not, executes a game with the provers that verifies that, either the provers hold a quantum proof that the Turing machine halts within 2^n steps, or they hold a history state for the verification of a quantum proof that either the Turing machine halts within 2^{2^n} steps, or... Somewhat more formally, we obtain the following (see the full version of the paper in [10] for a more complete statement):

Theorem 1.2. *For all deterministic Turing machines M, there exists a multiprover game $\mathcal{G}_M$ (that can be computed from the description of M) such that if M halts in finite time then $\omega^*(\mathcal{G}_M) < 1$, whereas if M does not halt then $\omega^*(\mathcal{G}_M) = 1$. Furthermore, there exists a universal constant $\eta > 0$ such that for any non-halting M, any strategy for the provers that succeeds with probability at least $1 - \varepsilon$ in $\mathcal{G}_M$, for some $\varepsilon \geq 0$, requires the use of an entangled state of local dimension at least $2^{\Omega(\varepsilon^{-\eta})}$.*

The game $\mathcal{G}_M$ in Theorem 1.2 is a game with 15 provers that can be efficiently computed from M; the undecidability result follows immediately. In addition, as stated in the theorem our game can be used as a form of dimension test for the strategies of the provers. Up to the value of the constant η the bound $2^{\Omega(\varepsilon^{-\eta})}$ matches the best bound known, for a three-prover game considered in [18].

2 PROOF OVERVIEW

We provide a detailed overview for the proof of Theorem 1.1. In Section 2.1 we sketch our main "compression" result and expand on the compression technique from [17]. The following sections sketch the proof of the compression theorem. We start by describing a method to succinctly describe the actions of a verifier in a multiprover game in Section 2.2. In Section 2.3 we describe the main steps of the proof: (1) design a history state associated with the execution of a multiprover game, (2) design a game that verifies the history state with the help of an additional trusted prover, and finally (3) design a game in which the honest prover has been merged into existing provers. This last step, prover merging, is described in more detail in Section 2.4. In Section 2.5 we sketch how the compression theorem can be applied recursively to show Theorem 1.1 and Theorem 1.2.

2.1 Protocol Compression

The main workhorse of this paper is a compression theorem for quantum multiprover interactive protocols that simplifies and strengthens the compression result of [17]. To state the result, we first review the notion of k-prover "extended nonlocal (ENL) game", which is a type of quantum multiprover game introduced in [19]. A k-prover ENL game is a three-turn interaction between a quantum verifier and k quantum provers sharing entanglement. The game (or "protocol") proceeds in three stages. First, the provers send a quantum register C to the verifier. Second, the verifier measures the register C to obtain an outcome t. [4] The verifier then computes a classical query $Q = (q_1, \ldots, q_k)$ that it distributes to the provers. Third, the provers respond with classical answers $a = (a_1, \ldots, a_k)$ to their respective questions. In general, each prover's answer is determined by performing a measurement on the prover's share of a quantum state that may be entangled with C. Finally, the verifier makes an accept/reject decision based on the outcome t, its internal randomness, and the provers' answers. The maximum acceptance probability of an ENL game $\mathcal{G}$ is denoted $\omega^*(\mathcal{G})$, and is also called the (entangled) *value* of $\mathcal{G}$.

[3] The analogy only goes so far: composition in PCPs reduces the answer size; here, we reduce the query size.

[4] Our definition of ENL game is slightly more general than that in [19], where the sampling of questions is classical and does not depend on C.

The whole interaction between verifier and provers in an ENL game can be represented as a quantum circuit of a special form that we call a *protocol circuit*, as depicted in Figure 1. A protocol circuit starts with the application of a quantum circuit C_Q on registers C (which holds the provers' first message), V (the verifier's private workspace), and M (which holds the messages exchanged between the verifier and provers). The circuit C_Q implements the verifier's measurement on register C, and the verifier's choice of questions to the provers. The circuit C_Q is followed by an arbitrary unitary transformation for each prover i, applied on the component M_i of the message register that the prover has access to, as well as its private workspace P_i (that contains the prover's part of shared entangled state). Finally, the last step in the protocol circuit is the application of a circuit C_A that acts on C, V and M and computes the verifier's decision in the game, that is written on a specially designated "output qubit".

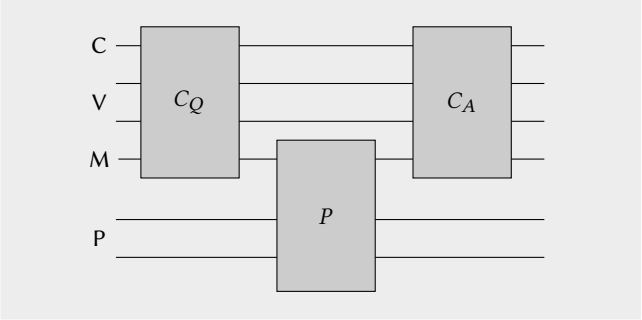

Figure 1: The protocol circuit of an extended nonlocal game.

The compression theorem applies to families of ENL games $\{\mathcal{G}_N\}$ that have *succinct descriptions*. By this we mean, not only that the protocol circuit associated with $\mathcal{G}_N$ has size polynomial in N, but moreover there exists a deterministic Turing machine G (called a *Gate Turing Machine* (GTM)) that on input (N, t), where N and t are two integers written in binary, runs in polynomial time and returns the description of the t-th gate of the protocol circuit associated with $\mathcal{G}_N$ (and a special symbol if t is larger than the circuit size). If the t-th gate is an action of the prover, the GTM returns another special symbol.

Theorem 2.1 (Compression Theorem). *Let $k \geq 7$ be an integer and let $\{\mathcal{G}_N\}$ be a succinctly described family of k-prover ENL games with GTM G. Then there exists a family of k-prover ENL games $\{\mathcal{G}_n^\sharp\}$ such that for all integer $n \geq 1$ and $N = 2^n$, it holds that*

$$\omega^*(\mathcal{G}_n^\sharp) \leq 1 - \frac{(1-\omega^*(\mathcal{G}_N))^\alpha}{\mathrm{poly}(N)}, \tag{1}$$

where $\alpha \geq 1$ is a universal constant, and if $\omega^*(\mathcal{G}_N) = 1$ then we have $\omega^*(\mathcal{G}_n^\sharp) = 1$. Moreover, there exists a Turing machine $A^\sharp$ that on input $(1^n, G)$ returns the description of $\mathcal{G}_n^\sharp$ in polynomial time.

The strength of the theorem lies in the exponential reduction in the size of the verifiers of the ENL game, from poly(N) (the size of $\mathcal{G}_N$) to poly(n) = poly(log N) (the size of $\mathcal{G}_n^\sharp$). The cost of this exponential compression of game size is that the value of the game gets "compressed" towards 1; nevertheless, games with value 1 (resp. < 1) are compressed to games with value 1 (resp. < 1). Theorem 2.1 differs from the results of [17] in two significant ways. First, the compression result in [17] does not yield a family $\{\mathcal{G}_n^\sharp\}$ that is as efficiently described as the games returned by our reduction.[5] The recourse to succinct descriptions via Gate Turing Machines is an essential ingredient for the recursive application of Theorem 2.1. Second, the compression result in [17] increases the number of provers, from k to $k + 8$. Our result does not require the use of additional provers; this is again essential in allowing a large (or even infinite) number of recursive applications of the theorem.

In the following subsections we sketch the proof of Theorem 2.1. The first step is to make the notion of "succinctly described" more concrete.

2.2 Succinct Descriptions of Verifiers

In the study of quantum interactive proof systems, families of games $\{\mathcal{G}_N\}$ are usually presented as a uniformly generated family of circuits for the verifier: there exists a polynomial-time deterministic Turing machine A that on input 1^N returns a circuit description of the verifier in $\mathcal{G}_N$. However, such uniform descriptions of verifier circuits are insufficient for our compression result: from a game $\mathcal{G}_N$ we aim to design a "compressed game" $\mathcal{G}_n^\sharp$ that has size poly(n), exponentially smaller than the size of $\mathcal{G}_N$. In particular, $\mathcal{G}_n^\sharp$ does not have nearly enough time to run A to get a circuit description of the verifier of $\mathcal{G}_N$. What we need is that the verifier of $\mathcal{G}_n^\sharp$ be granted some form of *implicit* description of the verifier of $\mathcal{G}_N$.

We achieve this via the notion of a *Gate Turing Machine* (GTM) for a family of ENL games $\{\mathcal{G}_N\}$. As mentioned before, it is a Turing machine G that on input (N, t) outputs in poly log(N) time the description of the t-th gate of the protocol circuit of $\mathcal{G}_N$ (which has size poly(N)).

Thus, our notion of "succinct description" for a family of ENL games $\{\mathcal{G}_N\}$ is that there is a GTM G for the family. With this notion in place, it remains to show the compression theorem: any succinctly described family of games $\{\mathcal{G}_N\}$ can be "compressed" to another family of ENL games $\{\mathcal{G}_n^\sharp\}$ with the properties described in Theorem 2.1. We sketch how this is done in the next sections.

2.3 Testing History States of Protocol Circuits

With the appropriate notion of succinct description in place, we describe the three main steps that go into the proof of Theorem 2.1.

The first step consists in considering the history state $|\Psi_{\mathcal{G}}(N)\rangle$ of the protocol circuit (Figure 1) associated with an execution of $\mathcal{G} = \mathcal{G}_N$, where $N = 2^n$. This state is defined on the registers CVMP, and may be extremely large, depending on the size of the provers' registers. In addition, the state has a component on a clock register C_{outer} of the same dimension as the total number of gates τ_N in the protocol circuit, which is polynomial in N; thus the register C_{outer} is over $O(n)$ qubits. Concretely, the state $|\Psi_{\mathcal{G}}(N)\rangle$

[5] Although the question lengths of the "compressed" game in [17] are $O(\log N)$, the verifier itself has size poly(N). The verifier for the game $\mathcal{G}_n^\sharp$, in contrast, has size poly(log N).

has the form

$$|\Psi_{\mathcal{G}}(N)\rangle \;=\; \frac{1}{\sqrt{\tau_N+1}}\sum_{t=0}^{\tau_N}|t\rangle_{\mathsf{C}_{outer}}\otimes U_t\cdots U_1|\psi_{\mathcal{G}}(0)\rangle_{\mathsf{CVMP}}\;. \quad (2)$$

Here $|\psi_{\mathcal{G}}(0)\rangle$ is the initial state of the verifier and the provers' registers in G, with C denoting the initial register received from the provers, V the private workspace for the verifier, $\mathsf{M}=\mathsf{M}_1,\ldots,\mathsf{M}_k$ the message registers, and $\mathsf{P}=\mathsf{P}_1,\ldots,\mathsf{P}_k$ the private spaces for the provers.

Note that in (2), almost all unitaries are gates applied by the verifier, except k of them, one for each prover, that can be considered "wild cards". The important property is that, if $\omega^*(\mathcal{G})=1$ then there exists a state of the form (2), for some choice of $|\psi_{\mathcal{G}}(0)\rangle$, and some choice of unitaries to apply in the "wildcard" locations, that is a ground state (energy 0) of the local Hamiltonian $H_{\mathcal{G}}(N)$ that verifies the history state (this is entirely analogous to Kitaev's circuit-to-Hamiltonian construction, but for the use of the prover gates which may induce large non-local Hamiltonian terms to verify their propagation). Conversely, if $\omega^*(\mathcal{G}_N)\neq 1$ then no such state exists, irrespective of the choice of the "wildcard" unitaries.

The next step is to design an intermediate ENL game $\mathcal{G}_H$ that has one additional prover, called the "Pauli Prover" PV. We call the verifier in $\mathcal{G}_H$ the *outer verifier*. The goal of the outer verifier is to verify that the provers share the state $|\Psi_{\mathcal{G}}(N)\rangle$, where registers associated with the verifier in $\mathcal{G}$ (that we call the *inner verifier*), i.e. C, V and M, are given to PV, while the clock register C_{outer} is the prover's first message in the ENL game $\mathcal{G}$. As already mentioned, this initial message has length $O(n)$ qubits.

Informally, to achieve this verification task the outer verifier and the Pauli Prover collaborate to implement a family of tests that are game-like versions of the tests implemented by the local Hamiltonian $H_{\mathcal{G}}(N)$. This includes an "input check" (the state $|\psi_{\mathcal{G}}(0)\rangle$ is well-formatted), a "gate check" (each time step corresponds to the application of a unitary, and unitaries associated with the inner verifier are the right ones, as specified in the circuits C_Q and C_A), and an "output check" (the final decision made by the inner verifier is to accept). Each of these checks involves not only the verifier and PV, but also the other provers, that are required to apply their prover gate when the corresponding propagation check is performed.

In designing $\mathcal{G}_H$, we take advantage of the fact that the Pauli Prover is considered "honest": it always implements the observable that it is asked by the outer verifier. However, for reasons that will soon become clear the Pauli Prover can only be asked to implement single- or two-qubit Pauli observables.[6] This means that all tests performed by the outer verifier can only require such observables on the registers CVM.

The crucial point here is that the complexity of the verifier in the game $\mathcal{G}_H$ is exponentially smaller than the complexity of the verifier in $\mathcal{G}$. The reason this is possible is that in order for the verifier in $\mathcal{G}_H$ to check that the entangled state shared by the provers is a valid history state for the protocol circuit associated with $\mathcal{G}$ it is enough to select a random time step in that circuit, and implement the associated check. Both of these can be performed in time poly $\log(N)$; the first trivially so, and the second thanks to our assumption that $\mathcal{G}$ is specified through a "succinct description", provided by the verifier $\mathcal{V}$ and GTM G associated with $\{\mathcal{G}_N\}$, as described in Section 2.2.

In the last step we convert the Single Pauli Prover game $\mathcal{G}_H$ into a new ENL game $\mathcal{G}^\sharp=\mathcal{G}^\sharp_n$, with the same number of provers as in the original ENL $\mathcal{G}$, but with drastically reduced question length — it is now $O(n)$, when questions in $\mathcal{G}$ might have been poly(N) bits long. For this we need to remove the "honest" assumption on PV, and moreover we need to "merge" PV with existing provers. This step of prover merging is explained in the next subsection.

2.4 Prover Merging

Prover merging is performed in two steps. The first step uses somewhat standard techniques, similar to those employed in [17], that originate in the self-testing literature. The main idea is to require the honest Pauli prover PV in $\mathcal{P}$ to implement the observable it is asked to measure transversally, on an error-encoded version of his share of the state (this is the main motivation for restricting the prover to Pauli observables), and then to split PV into as many provers as the error-correcting code requires. It is then possible, using self-testing technique, to test the "split" PV so as to ensure that any deviation from the honest actions is detected by the verifier.

The second step is the actual merging step. This step is somewhat delicate: we take the split provers, and merge them into existing provers from $\mathcal{G}$. Since each prover P now simultaneously receives two questions — its question in $\mathcal{G}$, as well as the share of the question to PV that would have been sent to the split prover that got merged into P — soundness is non-obvious.

To show that this step does not compromise soundness, we leverage the fact that, by construction, the prover that is to be merged only has to perform very simple operations: Pauli σ_X and σ_Z observables, on a constant number of qubits at a time. These kinds of operations can be tested, indeed "commanded", in a very rigid way by using self-testing results. Therefore, we can embed these actions into any prover. It is then straightforward to enforce that a prover performs the right action on a Pauli observable. However, its action on the real question may depend on the Pauli question. To get around this we once again leverage the structure of the Pauli Prover game as well as the quantum error-correcting code. More details on this part are given in the full version of the paper in [10].

2.5 Recursive Compression

Ultimately, we use our compression theorem (Theorem 2.1) in a recursive fashion to prove Theorem 1.1. To illustrate the essential idea behind the recursive compression approach, we give an informal overview of the proof of the statement that any language computable in deterministic time $t(n)$ has a quantum interactive proof system with completeness-soundness gap that scales as an inverse polynomial in $t(n)$.

Let L be such a language. Then there exists a deterministic Turing machine M that on input $x\in\{0,1\}^n$ decides whether $x\in L$ in time $t(n)$. For every $x\in\{0,1\}^n$ and integer $N\geq n$, we construct a verifier $\mathcal{V}_{x,N}$ for a 7-prover ENL game $\mathcal{G}_{x,N}$ that does the following. The verifier first runs M for N steps on input x. If M accepts in this time, then $\mathcal{V}_{x,N}$ accepts. If M rejects in this time, then $\mathcal{V}_{x,N}$ rejects. Otherwise, M has not halted. In this case $\mathcal{V}_{x,N}$ executes a

[6] In fact, triples of commuting two-qubit observables; we gloss over this for purposes of this overview.

compressed version of the protocol corresponding to $\mathcal{V}_{x,2^N}$, which is an exponentially larger version of itself. This compressed protocol is provided by Theorem 2.1. The recursion continues until at some point, M is run for a large enough "tower of exponential" number of steps that exceeds $t(n)$, in which case M either accepts or rejects input x. The following can then be shown by induction on R such that $t(n) \leq \Lambda_R(n)$. If $x \in L$ then the value of the game $\mathcal{G}_{x,t(n)}$ is 1, and therefore for all $N \leq t(n)$ the value of $\mathcal{G}_{x,N}$ is 1, which implies that $\mathcal{G}_{x,n}$ has value 1. Otherwise, if $x \notin L$, then using Theorem 2.1 we obtain that the value of $\mathcal{G}_{x,n}$ is at most $1 - \Omega\left(1/\mathrm{poly}(t(n))\right)$.

This nearly shows the desired conclusion, except that Theorem 2.1 requires that the family of games to be compressed have a succinct description in the manner described in Section 2.2. We thus need to argue that the family of games $\{\mathcal{G}_{x,n}\}$ has a GTM G associated with it. *A priori* it is unclear whether the verifiers $\{\mathcal{V}_{x,n}\}$ are structured enough so that any particular gate of the verifier circuits can be specified in polylogarithmic time. However, we show that as long as the verifiers $\{\mathcal{V}_{x,n}\}$ are *uniformly generated* (meaning that there is some polynomial time Turing machine A that on input $(1^n, x)$ returns the description of the verifier circuits of $\mathcal{V}_{x,n}$), there is an *equivalent* family of verifiers $\{\mathcal{V}'_{x,n}\}$ that has a *succinct description*. We prove this fact in the full version of the paper [10]; the proof relies on a concept from classical complexity theory known as *oblivious simulation* of Turing machines. Since the family of verifiers $\{\mathcal{V}_{x,N}\}$ is uniformly generated, we obtain that the verifiers have a succinct description via a GTM, which in turn allows us to apply the compression theorem as outlined above.

Finally, we address the issue that the games $\mathcal{G}_{x,N}$ described thus far are technically ENL games, meaning that the verifiers have some quantum capability, whereas the statement of Theorem 2.1 refers to standard nonlocal games, where the verifier is entirely classical. We can transform the ENL game $\mathcal{G}_{x,N}$ into a nonlocal game $\mathcal{G}'_{x,N}$ by delegating the measurements of the verifier $\mathcal{V}_{x,N}$ to an additional set of provers, by observing that the verifier's measurements are also simple Pauli measurements, and thus can be commanded by using the same self-testing techniques as used in the prover merging section.

Adapting this sketch to handle languages that are decided by *nondeterministic* Turing machines (as needed in Theorem 1.1), as well as reproving Slofstra's undecidability result (Theorem 1.2), requires additional care. We give additional details in the full version of the paper [10].

3 IMPROVING THE COMPRESSION THEOREM?

Theorem 2.1 offers the following tradeoff between "compression in size" and "compression of the gap": the former is scaled by an exponential factor, from polynomial in $N = 2^n$ to polynomial in n, while the latter is divided by a quantity that is polynomial in N, or equivalently, exponential in n.

Surprisingly, we show that *any* better tradeoff, i.e. one in which the gap gets reduced by a subexponential factor in n, would have far-reaching consequences in complexity theory and mathematics. The result provides a possible explanation for the absence of meaningful upper bounds on MIP* (provided an improved compression result does hold): not only would every computable language be decided by an MIP* proof system, it would show a negative resolution to a long standing open question in quantum information theory concerning the relationship between the commuting operator model and tensor product model of quantum correlations.

Theorem 3.1 (Consequences of an improved compression theorem). *Suppose an analogue of Theorem 2.1 holds, such that the factor* $\mathrm{poly}(N)$ *in the denominator on the right-hand side of* (1) *is replaced by a subexponential function of* $n = \log N$. *Then*

(1) MIP* *with constant gap contains all computable languages.*
(2) The commuting operator model of multipartite correlations is strictly more powerful than the tensor product model.

The idea behind the proof of Theorem 3.1 is that the tradeoff between a subexponential compression in gap and an exponential reduction in size can be "boosted" to a tradeoff where the gap does not get compressed at all, but the game size still gets compressed by a nontrivial amount. This uses *hardness amplification* techniques for multiprover entangled games [4], which employs a variant of parallel repetition to achieve this boosting. We give more details in the full version of the paper in [10].

We briefly explain what we mean by the second item in Theorem 3.1. In this paper, we define the entangled value of a nonlocal game as the supremum of the success probabilities over all "tensor product" strategies for the provers, which consist of a finite-dimensional Hilbert space for each prover, an entangled state in the tensor product of those Hilbert spaces, and a collection of measurement operators on each prover's space.

There is an alternate definition of the entangled value, which considers the supremum over so-called "commuting operator" strategies, for which there is a single (possibly infinite-dimensional) Hilbert space shared by all players, and the only restriction is that measurement operators applied by distinct provers commute with each other. Since tensor product strategies are also commuting operator strategies, the entangled value in the tensor product model is at most the entangled value in the commuting operator model. It is known that in the finite dimensional case, the two models are equivalent. Whether they coincide in general is a famous problem in quantum information known as "Tsirelson's problem" (see e.g. [12]).

A positive resolution to Tsirelson's problem implies the existence of an algorithm to approximate the value of any nonlocal game. However, the first item of Theorem 3.1 shows that an improved compression theorem would refute the existence of such an algorithm, and thus would give a negative answer to (the multipartite version of) Tsirelson's problem.

It is known that Tsirelson's problem for two-prover games is essentially equivalent to Connes' Embedding Conjecture [7], a longstanding open problem in functional analysis (see [12, 20, 26]). In particular, a separation between the definitions of entangled value for games with *two* provers would refute Connes' Embedding Conjecture. We do not know if a separation for games with more than two provers (e.g., 15) would still refute Connes' Embedding Conjecture.

4 RELATED WORK

We were informed of a forthcoming paper [8] by Coudron and Slofstra that establishes a result similar (though strictly incomparable)

to Theorem 1.1, using completely different techniques. In particular, the authors show that distinguishing between entangled value 1 or $1 - 1/\mathrm{poly}(t(n))$ for games with *two* provers in the commuting operator model is hard for nondeterministic $t(n)$ time (whereas our result shows hardness for nondeterministic $2^{t(n)}$ time for games with 15 provers in the tensor product model). This result relies on the group-theoretic framework that was pioneered in [27, 28].

5 OUTLOOK

The most important structural properties of classical multiprover interactive proof systems have been established since the 90s. It is known that any multiprover interactive proof system can be parallelized to a single round of interaction, with two provers only; that completeness 1 can be achieved without loss of generality; that soundness can be amplified in parallel; finally, and most importantly, that the class MIP of languages that can be recognized by any multiprover interactive proof system, for any nontrivial choice of completeness and soundness parameters, is exactly NEXP. Here, by nontrivial we mean any (c, s) such that $\exp(-\mathrm{poly}(n)) \leq s < c \leq 1$, where $c - s$ is at least $\exp(-\mathrm{poly}(n))$. We use $\mathrm{MIP}_{c,s}(k, r)$ to denote the class of languages that can be decided by a polynomial-time verifier interacting with k provers through an r-round interaction, with completeness c and soundness s. Thus, $\mathrm{MIP}_{c,s}(2, 1) = \mathrm{NEXP}$ for all nontrivial values of (c, s). When we write MIP we mean the union of all $\mathrm{MIP}_{c,s}(k, r)$ for polynomially bounded functions k, r, and c, s such that $0 < s < c \leq 1$ and $(c - s)^{-1}$ is polynomially bounded.

In contrast, complexity-theoretic aspects of entangled-prover interactive proof systems remain, to put it mildly, an untamed wilderness. Prior to our work it was known that $\mathrm{NEXP} \subseteq \mathrm{MIP}^*$ [14, 24, 29] with completeness 1 and soundness $\frac{1}{2}$, and that if one allows the completeness-soundness gap to close exponentially fast with n, then the inclusion can be strengthened to NEEXP, or, in our notation, $\mathrm{NTIME}(\Lambda_2(n))$ [17]. Interestingly, a similar phenomenon had previously been observed for single-prover interactive proof systems, for which it is known that QIP = PSPACE with constant gap [15], but QIP contains EXP if one allows a doubly exponentially small gap [13]. Unlike MIP^*, however, the power of QIP does not grow arbitrarily when the gap goes to zero; for any positive gap the class is contained in EXPSPACE [13].

For the case of multiprover interactive proof systems with entangled provers, there is no compelling reason that a shrinking gap would be necessary for the verification of languages beyond NEXP. Indeed, no upper bounds are known on MIP^* with constant gap — it is not even known to be contained in the set of decidable languages. In fact, recent works provide indication that the class may be larger than NEXP: it is known that $\mathrm{QMA}_{\mathrm{EXP}}$, the "exponential-size proof" analogue of QMA, is such that $\mathrm{QMA}_{\mathrm{EXP}} \subseteq \mathrm{MIP}^*_{1,1-2^{-n}}(5, 1)$ [11, 16], and inclusion with a constant gap holds under randomized reductions [23]. It is therefore an interesting question to determine to what extent the exponentially small completeness-soundness gap that our technique requires is necessary. As mentioned earlier, significant consequences in complexity theory and mathematics would follow from even a small improvement in our compression theorem, Theorem 2.1.

Another major open question on entangled-prover interactive proof systems is the role of the number of provers. Currently, it is not known if e.g. 3 provers allow to determine more languages than 2 (for any setting of the completeness-soundness gap). Our proof of the compression theorem involves a "prover merging" step that reduces the number of provers, albeit for a very restricted type of interactive proof systems. We also note that our techniques restrict us to games with at least 7 provers. This could potentially be decreased to 5, or even 3, by replacing the use of the 7-qubit Steane code with, say, a qutrit error-detecting code. Achieving a result with two provers seems more challenging. Yet, the undecidability results in [28] apply to two-prover games; it would be interesting to investigate whether some improvements on our techniques could take us all the way to hardness results for two-prover games as well.

A number of problems in quantum information theory are known to be undecidable. One that bears superficial similarity with the problem considered in this paper, in the statement as well as in the techniques, is the undecidability of the spectral gap of an infinite translation-invariant Hamiltonian, shown in [9]. It would be interesting to determine whether there could be a direct reduction from a multiprover game to that problem.

ACKNOWLEDGMENTS

We thank the anonymous STOC 2019 referees for helpful comments that have improved the presentation of this paper. Joseph Fitzsimons acknowledges support from Singapore's Ministry of Education and National Research Foundation, and the US Air Force Office of Scientific Research under AOARD grant FA2386-15-1-4082. This material is based on research funded in part by the Singapore National Research Foundation under NRF Award NRF-NRFF2013-01. Thomas Vidick is supported by NSF CAREER Grant CCF-1553477, AFOSR YIP award number FA9550-16-1-0495, a CIFAR Azrieli Global Scholar award, and the IQIM, an NSF Physics Frontiers Center (NSF Grant PHY-1125565) with support of the Gordon and Betty Moore Foundation (GBMF-12500028). Henry Yuen conducted the research for this work as a postdoctoral fellow at the University of California, Berkeley.

REFERENCES

[1] Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. 1998. Proof Verification and the Hardness of Approximation Problems. *J. ACM* 45, 3 (1998), 501–555.
[2] Sanjeev Arora and Shmuel Safra. 1998. Probabilistic Checking of Proofs: A New Characterization of NP. *J. ACM* 45, 1 (1998), 70–122.
[3] László Babai, Lance Fortnow, and Carsten Lund. 1991. Non-deterministic exponential time has two-prover interactive protocols. *Computational Complexity* 1 (1991), 3–40. Issue 1.
[4] Mohammad Bavarian, Thomas Vidick, and Henry Yuen. 2017. Hardness amplification for entangled games via anchoring. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*. ACM, 303–316.
[5] William W. Boone. 1958. THE WORD PROBLEM. *Proceedings of the National Academy of Sciences* 44, 10 (1958), 1061–1065. https://doi.org/10.1073/pnas.44.10.1061
[6] Richard Cleve, Peter Hoyer, Benjamin Toner, and John Watrous. 2004. Consequences and limits of nonlocal strategies. In *Computational Complexity, 2004. Proceedings. 19th IEEE Annual Conference on*. IEEE, 236–249.
[7] Alain Connes. 1976. Classification of injective factors Cases $II_1, II_\infty, III_\lambda, \lambda \neq 1$. *Annals of Mathematics* (1976), 73–115.
[8] Matthew Coudron and William Slofstra. 2018. Complexity Lower Bounds for Approximating Entangled Games to High Precision. (2018).

[9] Toby S Cubitt, David Perez-Garcia, and Michael M Wolf. 2015. Undecidability of the spectral gap. *Nature* 528, 7581 (2015), 207.
[10] Joseph Fitzsimons, Zhengfeng Ji, Thomas Vidick, and Henry Yuen. 2018. Quantum proof systems for iterated exponential time, and beyond. *arXiv preprint arXiv:1805.12166* (2018).
[11] Joseph Fitzsimons and Thomas Vidick. 2015. A multiprover interactive proof system for the local Hamiltonian problem. In *Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science*. ACM, 103–112.
[12] Tobias Fritz. 2012. Tsirelson's problem and Kirchberg's conjecture. *Reviews in Mathematical Physics* 24, 05 (2012), 1250012.
[13] Tsuyoshi Ito, Hirotada Kobayashi, and John Watrous. 2012. Quantum interactive proofs with weak error bounds. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*. ACM, 266–275.
[14] Tsuyoshi Ito and Thomas Vidick. 2012. A multi-prover interactive proof for NEXP sound against entangled provers. *Proc. 53rd FOCS* (2012), 243–252. arXiv:arXiv:1207.0550
[15] Rahul Jain, Zhengfeng Ji, Sarvagya Upadhyay, and John Watrous. 2010. QIP = PSPACE. *Commun. ACM* 53, 12 (2010), 102–109. arXiv:0907.4737
[16] Zhengfeng Ji. 2016. Classical verification of quantum proofs. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing*. ACM, 885–898.
[17] Zhengfeng Ji. 2017. Compression of quantum multi-prover interactive proofs. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, Hamed Hatami, Pierre McKenzie, and Valerie King (Eds.). ACM, 289–302. https://doi.org/10.1145/3055399.3055441
[18] Zhengfeng Ji, Debbie Leung, and Thomas Vidick. 2018. A three-player coherent state embezzlement game. *arXiv preprint arXiv:1802.04926* (2018).
[19] N. Johnston, R. Mittal, Russo V., and J. Watrous. 2016. Extended nonlocal games and monogamy-of-entanglement games. *Proceedings of the Royal Society A* 472 (2016), 20160003.
[20] Marius Junge, Miguel Navascues, Carlos Palazuelos, D Perez-Garcia, Volkher B Scholz, and Reinhard F Werner. 2011. Connes' embedding problem and Tsirelson's problem. *J. Math. Phys.* 52, 1 (2011), 012102.
[21] OG Karlampovič. 1982. A finitely presented solvable group with unsolvable word problem. *Mathematics of the USSR-Izvestiya* 19, 1 (1982), 151.
[22] Alexei Yu Kitaev, Alexander Shen, and Mikhail N Vyalyi. 2002. *Classical and quantum computation*. Number 47. American Mathematical Soc.
[23] Anand Natarajan and Thomas Vidick. 2018. Low-degree testing for quantum states, and a quantum entangled games PCP for QMA. In *Proceedings of Foundations of Computer Science (FOCS)*. arXiv:1801.03821
[24] Anand Natarajan and Thomas Vidick. 2018. Two-Player Entangled Games are NP-Hard. In *33rd Computational Complexity Conference, CCC 2018, June 22-24, 2018, San Diego, CA, USA (LIPIcs)*, Rocco A. Servedio (Ed.), Vol. 102. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 20:1–20:18. https://doi.org/10.4230/LIPIcs.CCC.2018.20
[25] P. S. Novikov. 1955. On the algorithmic unsolvability of the word problem in group theory. *Trudy Mat. Inst. Steklov.* 44 (1955), 3–143.
[26] Narutaka Ozawa. 2013. About the Connes embedding conjecture. *Japanese Journal of Mathematics* 8, 1 (2013), 147–183.
[27] William Slofstra. 2016. Tsirelson's problem and an embedding theorem for groups arising from non-local games. *arXiv preprint arXiv:1606.03140* (2016).
[28] William Slofstra. 2019. The set of quantum correlations is not closed. In *Forum of Mathematics, Pi*, Vol. 7. Cambridge University Press.
[29] Thomas Vidick. 2013. Three-player entangled XOR games are NP-hard to approximate. In *Proc. 54th FOCS*. arXiv:1302.1242