$\mathsf{NEEXP} \subseteq \mathsf{MIP}^*$

Anand Natarajan*
California Institute of Technology

John Wright†
Massachusetts Institute of Technology

May 23, 2019

Abstract

We study multiprover interactive proof systems. The power of classical multiprover interactive proof systems, in which the provers do not share entanglement, was characterized in a famous work by Babai, Fortnow, and Lund (Computational Complexity 1991), whose main result was the equality $\mathsf{MIP} = \mathsf{NEXP}$. The power of quantum multiprover interactive proof systems, in which the provers are allowed to share entanglement, has proven to be much more difficult to characterize. The best known lower-bound on MIP^* is $\mathsf{NEXP} \subseteq \mathsf{MIP}^*$ due to Ito and Vidick (FOCS 2012). As for upper bounds, MIP^* could be as large as RE, the class of recursively enumerable languages.

The main result of this work is the inclusion $\mathsf{NEEXP} = \mathsf{NTIME}[2^{2^{\mathrm{poly}(n)}}] \subseteq \mathsf{MIP}^*$. This is an exponential improvement over the prior lower bound and shows that proof systems with entangled provers are at least exponentially more powerful than classical provers. In our protocol the verifier delegates a classical, exponentially large MIP protocol for NEEXP to two entangled provers: the provers obtain their exponentially large questions by measuring their shared state, and use a classical PCP to certify the correctness of their exponentially-long answers. For the soundness of our protocol, it is crucial that each player should not only sample its own question correctly but also avoid performing measurements that would reveal the *other* player's sampled question. We ensure this by commanding the players to perform a complementary measurement, relying on the Heisenberg uncertainty principle to prevent the forbidden measurements from being performed.

*anandn@caltech.edu
†jswright@mit.edu

Contents

Part I
Introduction

1 Introduction

This paper is about the complexity class MIP^* of multiprover interactive proof systems with entangled quantum provers—the quantum version of the classical class MIP. Classically, the study of MIP has had far-reaching implications in theoretical computer science. In complexity theory, the proof by Babai, Fortnow, and Lund [BFL91] that $\mathsf{MIP} = \mathsf{NEXP}$ was the direct antecedent of the PCP theorem [ALM+98, AS98], a seminal result which is the foundation of the modern theory of hardness of approximation. In cryptography, the MIP model was introduced to allow for information-theoretic zero-knowledge proofs [BOGKW88], and more recently MIP protocols have become essential building blocks in designing delegated computation schemes (see e.g. [KRR14]). These implications alone would be a sufficient motivation for considering the quantum class MIP^*, but remarkably, the study of MIP^* is also deeply related to long-standing questions in the foundations of quantum mechanics regarding the nature of quantum entanglement. Indeed, the MIP^* model itself was anticipated by the *nonlocal games* or *Bell tests* introduced in the work of John Bell [Bel64], who was in turn inspired by the thought experiment proposed by Einstein, Podolsky, and Rosen [EPR35]. These nonlocal games have had applications to quantum cryptography [Eke91, MY98, Col06], delegated quantum computation [RUV13], and more.

Even though the class MIP is now well-understood, it has proven difficult to determine the computational power of MIP^*. A priori, it is not even clear that MIP^* contains MIP, since adding entanglement could increase or decrease the power of the proof system. This is because the added resource of entanglement can make it easier for dishonest provers to cheat the verifier. Indeed, Cleve et al. [CHTW04] showed that for proof systems based on so-called XOR games (where the verifier's decision can only depend on the XOR of the provers' answer bits), the quantum class $\oplus\mathsf{MIP}^* \subseteq \mathsf{EXP}$, whereas classically $\oplus\mathsf{MIP} = \mathsf{NEXP}$. In particular, this result implied that the classical $\oplus\mathsf{MIP}$ protocol for NEXP of Håstad [Hås97] could not be sound against entangled provers. In spite of this, Ito and Vidick [IV12, Vid16] were able to show that $\mathsf{NEXP} \subseteq \mathsf{MIP}^*$, by proving that a different classical protocol *is* sound against entanglement. Note that the protocol of [Vid16] is *identical* to a protocol shown to be unsound by Cleve et al., except in that it uses 3 provers rather than 2 (the protocol is played by choosing a random subset of 2 provers from the 3). This illustrates the subtleties of dealing with entangled provers.

With the lower bound $\mathsf{NEXP} \subseteq \mathsf{MIP}^*$ established, a natural follow-up question is whether MIP^* is *strictly* more powerful than MIP. Indeed, it was long known that some MIP^* protocols possess a uniquely quantum property called *self-testing*, which has no direct analog in the classical setting. Roughly speaking, an MIP^* protocol is a self-test for a particular entangled state $|\psi\rangle$ if only provers using states close to $|\psi\rangle$ can achieve close to optimal success in the protocol. In such a protocol, observing that the provers succeed with nearly optimal probability *certifies* that they share a state close to the target state $|\psi\rangle$. The germ of this idea came from the work of Bell [Bel64], who studied the types of bipartite correlations that could be obtained from measuring an entangled state called the EPR state, which had been introduced by Einstein, Podolsky, and Rosen [EPR35]. Bell gave a protocol where provers using the EPR state could succeed with a greater probability than purely classical provers, and subsequent works of Tsirelson [Tsi80], and Summers and Werner [SW88] showed that (a variant of) Bell's protocol certifies the EPR state in the sense of self-testing.

In order to prove stronger lower bonds on MIP^*, the post-Ito-Vidick phase of MIP^* research aimed to use this self-testing property to design protocols for problems in Hamiltonian complexity,

the quantum analog of the theory of NP-completeness. In Hamiltonian complexity, the complexity class QMA plays the role of NP; it is the set of problems for which there exists a quantum witness state that can be efficiently checked by a polynomial-time quantum verifier. Problems in QMA seemed like a natural match for the powers of MIP* as one could potentially construct a protocol for QMA by designing a self-test for accepting witness states of some QMA-complete problem. The connection between MIP* and QMA was also well motivated from the point of view of the "quantum PCP" research program, which strives to find quantum analogues of the classical PCP theorem. In the classical setting, the PCP theorem can be viewed as a scaled-down version of MIP* = NEXP, showing that there exists an MIP* protocol for 3SAT (and thus for all of NP) with $O(\log(n))$-sized messages. Drawing inspiration from this, Fitzsimons and Vidick [FV15] stated a "quantum games PCP conjecture": that there should exist an MIP* protocol with $\log(n)$-sized messages for the local Hamiltonian problem, and thus for the class QMA. This was proved by Natarajan and Vidick [NV18a] in 2018 with a 7-prover protocol. Along the way to achieving this goal, [NV18a] developed a highly efficient self-test for high-dimensional entangled states: their "quantum low-degree test" is a self-test for n EPR pairs with only $O(\log(n))$ communication.

Already, the result of [NV18a] is strong evidence that MIP* $\neq$ MIP, since it is believed that QMA $\neq$ NP. But, at the same time, several other works showed that even larger separations were possible in the regime of subconstant soundness gaps. Here there are results in two settings. For MIP* with a soundness gap scaling inverse-exponentially (i.e. $1/\exp(n)$) in the instance size, Ji [Ji17] showed a protocol for NEEXP: nondeterministic *doubly*-exponential time, and a subsequent work by Fitzsimons, Ji, Vidick, and Yuen [FJVY19] showed protocols for non-deterministic iterated exponential time (e.g. NTIME(2^{2^n})) with a correspondingly small soundness gap (e.g. $2^{-C\cdot 2^n}$). In the "gapless" case, Slofstra [Slo16, Slo19] showed that given a description of an MIP* protocol, determining whether there exists an entangled strategy that succeeds with probability exactly 1 is undecidable by any Turing machine.

These results hint at the full power of MIP* but are not conclusive, as it is not unusual for quantum complexity classes to increase significantly in power when a numerical precision parameter is allowed to shrink. For instance, QIP (quantum interactive proofs with a single prover) with an exponentially small gap is equal to EXP [IKW12], while QIP with a polynomial gap is equal to IP = PSPACE. Likewise, QMA with exponentially small gap (known as PreciseQMA) is known to be equal to PSPACE [FL18], while QMA is contained PP, and QMA(k) (QMA with multiple unentangled Merlins) with exponentially small gap is equal to NEXP [Per12], whereas in the constant-gap regime the best known lower bound is that QMA(k) $\supseteq$ QMA. Moreover, even the QMA lower bound for MIP$^*_{\log}$ obtained by [NV18b] holds for 7 provers only; with 2 provers, the best known lower bound for MIP$^*_{\log}$ is NP = MIP$_{\log}$ [NV18a]. Could it be that 2-prover MIP* is equal to MIP, with entanglement providing no advantage at all?

This paper conclusively answers this question in the negative. Our main result (Theorem 1.1) is to show that MIP* contains NEEXP, with only two provers and with a constant completeness-soundness gap. This is establishes the first known unconditional separation between MIP* and MIP in the constant-gap regime: previously, such a separation was known only assuming QMA $\neq$ NP, and only in the scaled-down setting of logarithmic-sized messages.

Theorem 1.1 (Theorem 17.12 in the body)**.** *There is a two-prover, one-round* MIP* *protocol for the* NEEXP*-complete problem* Succinct-Succinct-3Sat *with completeness* 1*, soundness* 1/2*, and question and answer length* poly(n)*.*

As a corollary of Theorem 1.1, we obtain a lower bound on the hardness of approximation for the entangled value ω^* of a nonlocal game.

Corollary 1.2. *There exists a constant $c < 1$ such that given a two-prover nonlocal game $\mathcal{G}$ of size N, the problem of deciding whether $\omega^*(\mathcal{G}) = 1$ or $\omega^*(\mathcal{G}) \leq 1/2$, promised one of the two holds, is* $\mathsf{NTIME}(2^{N^{\log^{-c} N}})$*-hard.*

For two-player games, the best prior lower bound was NP [NV18b]. The lower bound achieved in Corollary 1.2 is stronger as for any $c < 1$, the function $2^{N^{\log^{-c} N}}$ is superpolynomial.

Techniques. Our construction, inspired by [Ji17] and [FJVY19], involves "compression": we show how to take an MIP protocol for NEEXP with exponentially-long questions and answers (the "big" protocol), and simulate it by an MIP^* protocol with polynomial-sized messages (the "small" protocol). However, the techniques we use to achieve our compression are quite different. We eschew the Hamiltonian-complexity ideas that were used in previous works, and in particular the use of history states. In our protocol, honest provers need only share a quantum resource state of (exponentially many) EPR pairs, together with a *classical* assignment to the NEEXP instance being tested. The use of history states was the main barrier preventing previous works from applying to the case of two provers.

We divide compression into two steps: *question compression* and *answer compression*. We achieve question compression by a technique which we call *introspection*, in which we command the provers to perform measurements on their shared EPR pairs whose outcomes are pairs of questions from the "big" protocol. To force the provers to sample their questions honestly, we use a variant of the quantum low-degree test from [NV18a], which certifies Pauli measurements on exponentially many EPR pairs using messages of only polynomial size. A crucial challenge is to prevent each prover from learning the other prover's sampled question, since this would destroy the soundness of the "big" protocol. To achieve this, we use the *"data-hiding"* properties of quantum measurements in incompatible bases: if a set of qubits is measured in the Pauli X-basis, this "erases" all information about Z-basis measurements. This means that if Alice samples her question by measuring her half of a block of EPR pairs in the Z-basis, then her question can be hidden from Bob by forcing him (via self-testing) to measure his half of the EPR pairs in the X-basis. Interestingly, our data-hiding scheme does *not* operate in a black-box way on the "big" protocol, but rather makes essential use of its structure. In particular, we start with a "big" protocol based on a scaled-up version of a PCP construction using the low-degree test, where the question distribution consists of pairs of random points in a vector space and affine subspaces containing them. The linear structure of the vector space is essential for our data-hiding procedure to work.

Our approach to answer compression is more standard, essentially using composition with a classical PCP of proximity. Here, the verifier asks the provers to compute a PCP proof that their "big" answers satisfy the success conditions of the protocol, and verifies this PCP proof by reading an exponentially smaller number of bits. Care is needed to deal with entanglement between the provers. The first, fundamental challenge we face is that the success condition of the "big" protocol is a function of *both* provers' answers. Thus, to compute a PCP proof that the condition is satisfied, one of the provers must have access to both provers' answers. Classically, this is achieved using the technique of oracularization, in which one prover receives *both* provers' questions and is checked for consistency against the other prover, which only receives a single question. In the entangled setting, this oracularization procedure is sound, but not necessarily complete. This is because oracularization requires that each prover, if given the *other* prover's question, could predict its answer with certainty, even though this answer is obtained from a nondeterministic quantum measurement. In our protocol, we are able to use oracularization because honest provers always use a maximally entangled state, which they measure with projective measurements that pairwise commute for every pair of questions asked in the game. While this commutation requirement is

restrictive, it still permits non-trivial quantum behavior; indeed, the linear system games used by Slofstra [Slo19] involve similar commutation conditions.

The second challenge is to ensure that the PCP of proximity we use for composition is itself sound against entanglement. We achieve this by performing a further step of composition: we ask the provers to encode their PCP proof in the low-degree code and verify it with the low-degree test, which is known to be sound even against entangled provers [NV18b]. This technique was introduced in the QMA protocol of [NV18a] in order to perform energy measurements on the provers' state.

Implications and future work We believe that our work opens up several exciting directions for further progress. For the complexity theorist, the most obvious future direction is to obtain even stronger lower bounds on MIP* by iterating our protocol, as in [FJVY19]. At the most basic level, we could imagine taking our MIP* protocol for NEEXP and performing a further layer of question compression and answer compression on it, thus obtaining an MIP* protocol with logarithmic message size for NEEXP, or, scaling up, an MIP* protocol with polynomial message size for $\mathsf{NTIME}(2^{2^{2^{\mathrm{poly}(n)}}})$. By further iterating question reduction and answer reduction k times, we could obtain potentially obtain lower bounds of $\mathsf{NTIME}(\underbrace{2^{\cdot^{\cdot^{n}}}}_{k})$ on MIP* while retaining a constant completeness-soundness gap. The main obstacle to achieving such results is that the question compression procedure developed in this paper is tailored to a special distribution of questions (that of the $\mathsf{MIP}_{\mathrm{exp}}$ protocol for NEEXP), whereas our answer compression procedure produces protocols whose question distribution is not of this form.

Assuming that this obstacle can be surmounted, we could aspire to a more ambitious goal: a general "gap-preserving compression procedure" for some subclass of MIP* protocols, which we may label "compressible" protocols. Such a procedure would consist of a Turing machine that takes as input any compressible MIP* protocol $\mathscr{G}$, and generates a new compressible protocol $\mathscr{G}'$ with exponentially smaller message size, but approximately the same entangled value. It was shown by [FJVY19] that the existence of such a compression procedure for the set of *all* MIP* protocols would imply that MIP* contains the set of all computable languages, and moreover that there exists an undecidable language in MIP*. These consequences would continue to hold as long as the set of compressible protocols contains a family of protocols solving problems in $\mathsf{NTIME}(f(n))$, where $f(n)$ is a growing function of n.

Showing that MIP* contains undecidable languages would be significant not just for complexity theory but also for the foundations of quantum mechanics, as it would resolve a long-standing open problem known as *Tsirelson's problem*. Tsirelson's problem asks whether two notions of quantum nonlocality are equivalent: the *tensor-product model*, in which two parties Alice and Bob each act on their respective factor of a tensor-product Hilbert space $\mathcal{H}_{\mathsf{Alice}} \otimes \mathcal{H}_{\mathsf{Bob}}$, and the *commuting-operator model*, in which both parties act on a common Hilbert space $\mathcal{H}$, but the algebra of Alice's measurement operators must commute with Bob's, and vice versa. It was shown by Slofstra [Slo16] that in the "zero-error" setting, these two models differ: there are quantum correlations which can be *exactly* achieved in the commuting-operator model but not in the tensor product model. Surprisingly, showing that MIP* contains undecidable languages would imply that the two models are separated even in the bounded-error setting: it would imply that there exist correlations that can be achieved in the commuting-operator model that cannot even be approximated (up to constant precision) in the tensor-product model. The reason for this implication is that if the two models are indistinguishable up to bounded error, then there exists a Turing machine that can decide any language in MIP* and is guaranteed to halt. This observation, which is folklore in the community, follows from the completeness of the non-commutative sum of squares hierarchy for the commuting-

operator model, as documented in [FJVY19]. Showing a separation between the two models would have significant mathematical consequences as well, as it would yield a negative answer to the long-standing Connes' embedding problem.

In addition to these connections to complexity and mathematical physics, we hope that our results will have applications in other areas such as to delegated computation or quantum cryptography. In particular, our use of introspection is reminiscent of ideas used in quantum randomness expansion, where randomness generated by measuring EPR pairs is used to generate questions for a nonlocal game. Could our results improve on the infinite randomness expansion protocol of Coudron and Yuen [CY14]?

Acknowledgements We thank Henry Yuen for many useful conversations about the idea of "introspecting" interactive proof protocols, which inspired us to start this project. AN is also grateful to the Simons Institute for the hospitable environment of the Summer Cluster on Challenges in Quantum Computation during which these conversations where held. We thank Thomas Vidick for his guidance and advice. We thank Ryan O'Donnell and Ryan Williams for a succinct review of the literature on the complexity of succinct (succinct) 3Sat and NE(E)XP. We are also grateful to Zhengfeng Ji for several useful discussions, especially regarding the consequences of recursively composing our protocol with itself.

We acknowledge funding provided by the Institute for Quantum Information and Matter, an NSF Physics Frontiers Center (NSF Grant PHY-1733907).

2 Overview of our proof

In this section we give a more detailed overview of the technical parts of the paper.

2.1 Basic quantum notation and qudits

While the main body of the paper contains a more complete set of quantum preliminaries in Section 4, for the purposes of this introduction we define some basic notation, aimed at the reader who is familiar with the standard quantum computing formalism over qubits but is less familiar with *qudits*: quantum systems of dimension not equal to 2. In this paper, we make extensive use of such qudits: in particular, for a finite field $\mathbb{F}_Q$, we will consider qudits of dimension Q, with a basis state $|i\rangle$ for every element $i \in \mathbb{F}_Q$. Under tensor product, we obtain a basis for the space of M qudits of dimension Q where each basis state $|x\rangle$ corresponds to a vector $x \in \mathbb{F}_Q^M$.

The basic resource state used in our protocols will be the EPR state over $2M$ qudits of dimension Q. The qudits are split into two registers of M qudits each, held by the two provers Alice and Bob, respectively.

$$|\mathrm{EPR}_Q^M\rangle = \frac{1}{\sqrt{Q^M}} \sum_{x \in \mathbb{F}_Q^M} |x\rangle_{\mathsf{Alice}} \otimes |x\rangle_{\mathsf{Bob}} .$$

This state is a *maximally entangled* state between Alice and Bob.

Acting on this state, we will ask the provers to perform measurements from a special class called *Pauli basis measurements*. To define these over a general field $\mathbb{F}_Q$ requires the introduction of some finite field technology, in particular the finite field trace function. For simplicity, in this overview we will imagine that Q is prime, allowing the addition in $\mathbb{F}_Q$ to be identified with the additive group $\mathbb{Z}_Q$, and simplifying the definition of the Paulis; in the main body of the paper, we will work with Q a power of 2. For a single qudit of dimension Q, the Pauli X and Z bases are the

sets $\{|\tau_u^X\rangle\}_{u\in\mathbb{F}_Q}$ and $\{|\tau_u^Z\rangle\}_{u\in\mathbb{F}_Q}$ of vectors

$$|\tau_u^X\rangle = \frac{1}{\sqrt{Q}} \sum_{x\in\mathbb{F}_Q} \omega^{xu} |x\rangle\,, \qquad |\tau_u^Z\rangle = |u\rangle\,,$$

where $\omega = \exp(2\pi i/Q)$ is the Q-th root of unity. We denote the projectors onto these basis states by τ_u^X and τ_u^Z, respectively. For a system of M qudits, the Pauli X and Z observables are a set of *generalized observables* indexed by elements of $\mathbb{F}_Q^M$: a generalized observable is a Hermitian matrix with eigenvalues that are Q-th roots of unity. They are given by

$$X(v) = \sum_{u\in\mathbb{F}_Q^M} \omega^{u\cdot v} \tau_{u_1}^X \otimes \ldots \otimes \tau_{u_M}^Z, \qquad Z(v) = \sum_{u\in\mathbb{F}_Q^M} \omega^{u\cdot v} \tau_{u_1}^Z \otimes \ldots \otimes \tau_{u_M}^Z,$$

where $u_1, \ldots, u_M$ are the components of the vector u, and $u\cdot v$ is the dot product $\sum_{i=1}^M u_i \cdot v_i$. Measuring a generalized observable means performing a projective measurement onto the eigenvectors of the observable, with the outcome a corresponding to the eigenvector with eigenvalue ω^a.

2.2 Our starting point: a classical interactive proof for NEEXP

We start with a classical multiprover interactive proof protocol for NEEXP. The equality MIP = NEXP was originally shown by Babai, Fortnow, and Lund [BFL91] using a protocol based on the *multilinearity test*: the idea is that an exponentially-long witness for a problem in NEXP is encoded in the truth-table of a multivariate polynomial function over a finite field, which is linear in each of the variables individually. The verifier is able to verify the witness by evaluating the multilinear polynomial over appropriately chosen points and subspaces. To scale up to NEEXP, we use a much more efficient version of the same idea, replacing the multilinearity test with the *low-degree test*, which works with multivariate polynomials of low total degree. This more efficient construction comes from the PCP literature. We give a relatively self-contained presentation of the protocol in Section 11. For the purposes of this overview, it is sufficient to know the following: any problem in NEEXP can be reduced to satisfiability for a doubly exponentially long 3Sat formula, succinctly encoded by a polynomial-sized circuit. (We refer to this problem as Succinct-Succinct-3Sat). Given a 3Sat formula ψ, we would like the provers to prove to us that they have a satisfying assignment a to this formula. Instead of reading the assignment directly, we will ask the provers to encode their assignment as a multivariate polynomial $g_a : \mathbb{F}_Q^M \to \mathbb{F}_Q$, where the number of variables M and the finite field size Q are appropriately chosen parameters, and return evaluations of this polynomial. To check that a satisfies ψ, the verifier first uses a technique called arithmetization to convert the formula ψ into a multivariate polynomial $g_\psi : \mathbb{F}_Q^{3M+k} \to \mathbb{F}_Q$. The polynomial g_ψ is chosen such that the assignment a satisfies ψ if and only if the expression

$$\mathrm{sat}_{\psi,a}(x,b,w) := g_\psi(x,b,w) \cdot (g_a(x_1) - b_1)(g_a(x_2) - b_2)(g_a(x_3) - b_3)$$

is equal to 0 at every point in a particular subset $H \subseteq \mathbb{F}_Q^{3M+k}$. Our classical protocol for NEEXP checks this condition:

Informal Theorem 2.1 (Section 11 in the body)**.** *There exists a protocol $\mathscr{G}_0$ for* Succinct-Succinct-3Sat *(and hence* NEEXP*), where the verifier's questions to the provers are constant-dimension subspaces of $\mathbb{F}_Q^M$, and the provers' responses are evaluations of degree-D M-variate polynomials on these subspaces. The parameters M, Q, D are all chosen to be* $\exp(n)$*, and hence the question and answer lengths as well as the runtime of the verifier in this protocol are* $\exp(n)$*.*

The distribution over subspaces sent to the provers in $\mathscr{G}_0$ is relatively simple, and in fact is independent of the instance of Succinct-Succinct-3Sat being tested. For the purposes of this overview, the reader can take the distribution over pairs of questions to be the *plane-point distribution* $\mathcal{D}$. A pair $(\boldsymbol{s}, \boldsymbol{u}) \sim \mathcal{D}$ consists of a uniformly random affine plane $\boldsymbol{s} \subseteq \mathbb{F}_Q^M$, which is sent to Alice, and a uniformly random point $\boldsymbol{u} \in \boldsymbol{s}$ which is sent to Bob. The full distribution over questions in $\mathscr{G}_0$ is more complicated than this but the essential ideas of our protocol will be illustrated by restricting to this case.

2.3 Restricting the strategies: registers and compilers

One of the main challenges in working with entangled provers is showing soundness against general entangled strategies. An important technique in this area is to force the provers to use a particular state and class of measurements by playing a type of game known as a *self-test*.

Informal Definition 2.2. A game $\mathscr{G}_{\text{test}}$ is a *self-test* for a state $|\psi\rangle$ and measurements M^x if any strategy that succeeds in $\mathscr{G}_{\text{test}}$ with probability $1-\epsilon$ must use a state $|\psi'\rangle$ and measurements $(M')^x$ that are $\delta(\epsilon)$-close, in the appropriate metric, to $|\psi\rangle$ and M^x.

Some of the earliest self-tests include the famous CHSH game, which self-tests the Pauli X and Z operators on a single EPR pair (of qubits). Self-testing technology has greatly advanced over the years, and in this paper we design a highly efficient self-test based on the low-degree test of [NV18a].

Informal Theorem 2.3 (Theorem 6.2 in the body)**.** *The Pauli basis test* $\text{Pauli}(n, q)$ *is a self-test for the state* $|\text{EPR}_q^n\rangle$ *and the Pauli X and Z basis measurements. This test sends the players questions of length* $O(\log(n))$ *and receives answers of length* $O(\text{poly}(n))$*.*

The Pauli X and Z measurements are "complete" measurements, and as a consequence, there is no nontrivial measurement on a set n qudits that can be measured jointly with both the Pauli X and Z measurements on those qudits. Using this property, we design a game called the *data-hiding game*, which certifies that a prover's measurements act trivially on a specified set of qudits.

Informal Theorem 2.4 (Theorem 8.3 in the body)**.** *The data-hiding game* $\mathscr{G}_{\text{hide}}$ *is a self test for states* $|\psi\rangle = |\text{EPR}_q^n\rangle \otimes |\text{aux}\rangle$ *and measurements* M^x *of the form* $M^x = I \otimes (M')^x_{\mathsf{aux}}$*. It has questions of length* $O(\log(n))$ *and answers of length* $O(\text{poly}(n))$*.*

Together, the Pauli basis test and the data-hiding game allow us to restrict our analysis of our protocols to a class of strategies we call *register strategies*: strategies for which the shared state is a collection of ℓ registers, each in an EPR state, together with some auxiliary register:

$$|\psi\rangle = |\text{EPR}_{q_1}^{n_1}\rangle \otimes \ldots \otimes |\text{EPR}_{q_\ell}^{n_\ell}\rangle \otimes |\text{aux}\rangle\,,$$

and where the provers can be commanded to perform either (1) Pauli basis measurements on specified subsets of the registers, or (2) measurements that do *not* act on specified subset of the EPR registers (but act on the auxiliary register or the remaining EPR registers). We formalize this by designing a *compiler*, which takes in a protocol $\mathscr{G}$ that is complete and sound for register strategies, and produces a new protocol $\mathscr{G}'$ which is complete and sound over all strategies.

Informal Theorem 2.5 (Theorem 7.2 and Theorem 9.2 in the body)**.** *Suppose $\mathscr{G}$ is a protocol for a computation problem for which completeness and soundness hold for register strategies, with* $O(1)$ *many registers of size n. (That is, for YES instances of the problem, there exists a register*

strategy achieving value 1*, and for NO instances, no register strategy achieves value greater than* 1/2*). Let the questions in* $\mathscr{G}$ *be of length* Q *and the answers be of length* A*. Then there exists a protocol* $\mathscr{G}'$ *which is complete and sound for general strategies, and for which the question length is* $Q + \log(n)$ *and the answer length is* $A + \mathrm{poly}(n)$*.*

The compiled protocol $\mathscr{G}'$ either runs the original protocol $\mathscr{G}$, or, with some probability, runs the Pauli basis test, the data-hiding game, or a consistency test.

2.4 Question reduction through introspection

With our compiler in place, we have now given the verifier the power to command the provers to perform Pauli basis measurements on a set of EPR pairs. We would like to use this to reduce the question size of the classical protocol $\mathscr{G}_0$ for NEEXP described above from $\exp(n)$ to $\mathrm{poly}(n)$. We will do so by forcing the provers, rather than the verifier, to sample the protocol's $\exp(n)$-length questions, a technique we call "introspection". That is, we would like to force the provers to sample pairs $(\boldsymbol{s}, \boldsymbol{u})$ from the plane-vs-point distribution $\mathcal{D}$, where $\boldsymbol{s}$ is a uniformly random affine plane in $\mathbb{F}_Q^M$, and $\boldsymbol{u}$ a uniformly random point on $\boldsymbol{s}$.

To design a scheme to sample from this distribution, let us first fix a representation of affine planes. We will represent an affine plane by an *intercept* $u \in \mathbb{F}_Q^M$ and two *slopes* $v_1, v_2 \in \mathbb{F}_Q^M$. The plane given by u, v_1, v_2 is the set $s_u^v = \{u + \lambda_1 v_1 + \lambda_2 v_2 : \lambda_1, \lambda_2 \in \mathbb{F}_Q\}$. As a first attempt, we may try the following scheme:

1. Alice and Bob share three registers, each of which contains an EPR state, so their shared state is
$$|\psi_0\rangle = |\mathrm{EPR}_Q^M\rangle_{R_0} \otimes |\mathrm{EPR}_Q^M\rangle_{R_1} \otimes |\mathrm{EPR}_Q^M\rangle_{R_2}.$$
2. Alice first measures her half of registers R_1 and R_2 in the Pauli Z-basis, to obtain uniformly random outcomes $\boldsymbol{v}_1, \boldsymbol{v}_2$. The shared state is now
$$|\psi_1\rangle = |\mathrm{EPR}_Q^M\rangle_{R_0} \otimes (|\boldsymbol{v}_1\rangle_{\mathsf{Alice}} \otimes |\boldsymbol{v}_1\rangle_{\mathsf{Bob}})_{R_1} \otimes (|\boldsymbol{v}_2\rangle_{\mathsf{Alice}} \otimes |\boldsymbol{v}_2\rangle_{\mathsf{Bob}})_{R_2}.$$
3. Now, Alice and Bob both measure register R_0 in the Pauli Z-basis, both obtaining the same outcome $\boldsymbol{u}$. The shared state is now
$$|\psi_2\rangle = (|\boldsymbol{u}\rangle_{\mathsf{Alice}} \otimes |\boldsymbol{u}\rangle_{\mathsf{Bob}})_{R_0} \otimes (|\boldsymbol{v}_1\rangle_{\mathsf{Alice}} \otimes |\boldsymbol{v}_1\rangle_{\mathsf{Bob}})_{R_1} \otimes (|\boldsymbol{v}_2\rangle_{\mathsf{Alice}} \otimes |\boldsymbol{v}_2\rangle_{\mathsf{Bob}})_{R_2}.$$
Alice sets her plane $\boldsymbol{s}$ to be $s_{\boldsymbol{u}}^{\boldsymbol{v}}$ and Bob sets his point to be $\boldsymbol{u}$.

Indeed, the pair $(\boldsymbol{s}, \boldsymbol{u})$ generated by this procedure is distributed according to $\mathcal{D}$. However, there is a problem: through her measurement, Alice obtains additional side information, specifically the value of Bob's point $\boldsymbol{u}$. Can we command Alice to erase the side information? In fact, we can, using the *Heisenberg uncertainty principle*: if two observables anticommute, then measuring one completely destroys information about the other. Using this idea, we modify our protocol as follows:

1. As above.
2. As above. At this point, applying the definition of $|\mathrm{EPR}_Q^M\rangle$, we can write the shared state as
$$|\psi_1\rangle \propto \sum_{u \in \mathbb{F}_Q^M} (|u\rangle_{\mathsf{Alice}} \otimes |u\rangle_{\mathsf{Bob}})_{R_0} \otimes (|\boldsymbol{v}_1\rangle_{\mathsf{Alice}} \otimes |\boldsymbol{v}_1\rangle_{\mathsf{Bob}})_{R_1} \otimes (|\boldsymbol{v}_2\rangle_{\mathsf{Alice}} \otimes |\boldsymbol{v}_2\rangle_{\mathsf{Bob}})_{R_2}.$$

3. **New:** Intuitively, we would like Alice to be *prevented* from measuring the component of the intercept along the directions $\boldsymbol{v}_1, \boldsymbol{v}_2$. This information would be obtained by measuring the observables[1] $Z(\boldsymbol{v}_1), Z(\boldsymbol{v}_2)$. To destroy it, we will ask Alice to measure the *complementary* Pauli observables $X(\boldsymbol{v}_1), X(\boldsymbol{v}_2)$ on register R_0, obtaining outcomes $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2 \in \mathbb{F}_Q$. The shared state is now

$$|\psi_2'\rangle \propto \sum_u \sum_{\lambda,\mu} \left(\omega^{\boldsymbol{\alpha}_1\lambda + \boldsymbol{\alpha}_2\mu} \underbrace{|u + \lambda\boldsymbol{v}_1 + \mu\boldsymbol{v}_2\rangle}_{u'}_{\mathsf{Alice}} |u\rangle_{\mathsf{Bob}} \right)_{R_0} (|\boldsymbol{v}_1\rangle_{\mathsf{Alice}} \otimes |\boldsymbol{v}_1\rangle_{\mathsf{Bob}})_{R_1} \otimes (|\boldsymbol{v}_2\rangle_{\mathsf{Alice}} \otimes |\boldsymbol{v}_2\rangle_{\mathsf{Bob}})_{R_2}.$$

   where, as above, $\omega = \exp(2\pi i/Q)$ is a Q-th root of unity. Alice and Bob's state on R_0 is now a uniform superposition over pairs u, u' of points lying on the same affine subspace with slopes $\boldsymbol{v}_1, \boldsymbol{v}_2$.

4. Alice and Bob both measure register R_0 in the Z basis, obtaining outcomes $\boldsymbol{u}$ and $\boldsymbol{u}'$, respectively. The shared state is now

$$|\psi_3'\rangle = (|\boldsymbol{u}\rangle_{\mathsf{Alice}} \otimes |\boldsymbol{u}'\rangle_{\mathsf{Bob}})_{R_0} \otimes (|\boldsymbol{v}_1\rangle_{\mathsf{Alice}} \otimes |\boldsymbol{v}_1\rangle_{\mathsf{Bob}})_{R_1} \otimes (|\boldsymbol{v}_2\rangle_{\mathsf{Alice}} \otimes |\boldsymbol{v}_2\rangle_{\mathsf{Bob}})_{R_2}.$$

   Alice sets her plane to be $s^{\boldsymbol{v}}_{\boldsymbol{u}}$ and Bob sets his point to be $\boldsymbol{u}'$.

Now, from the calculation performed above, it's clear that Bob's point $\boldsymbol{u}'$ is uncorrelated with Alice's intercept $\boldsymbol{u}$, apart from lying in the plane $s^{\boldsymbol{v}}_{\boldsymbol{u}}$, and hence there is no further information about Bob's point that Alice can learn by measuring her portion of the final state $|\psi_3'\rangle$. But Alice still obtains some additional information from her measurements along the way, in particular the outcomes α_1, α_2 of the X measurements. And moreover, how can we certify that the X measurements were performed correctly, since they are not Pauli basis measurements as given to us by the compiler? To answer these questions, we define a new game called the *partial data-hiding game* (Theorem 10.4), which certifies that Alice and Bob perform the steps described above and that no extra information is leaked. Building on this game, we can now design a protocol for NEEXP with small question size:

Informal Theorem 2.6 (Theorem 15.8 in the body)**.** *There is an* MIP* *protocol* $\mathscr{G}_1$ *for* NEEXP *with questions of length* poly(n)*, and answers of length* exp(n)*. The verifier can generate the questions in* poly(n) *time but needs* exp(n) *time to verify the answers.*

2.5 Answer reduction through PCP composition

We have succeeded in obtaining a game with short questions, but the answers are now exponentially long. In the last step, we will use composition with a classical probabilistically checkable proof (PCP) to delegate verification of the answers to the provers.

Schematically, the protocol $\mathscr{G}_1$ consists of the following steps:

1. The verifier sends Alice a question $\boldsymbol{x}$ and Bob a question $\boldsymbol{y}$.
2. Alice returns an (exponentially-long) answer $\boldsymbol{A}$ and Bob an exponetially-long answer $\boldsymbol{B}$.
3. The verifier computes a verification predicate $V(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{A}, \boldsymbol{B})$ in exponential time.

[1] Strictly speaking, this is only true when $\boldsymbol{v}_1 \cdot \boldsymbol{v}_1 \neq 0$ and $\boldsymbol{v}_2 \cdot \boldsymbol{v}_2 \neq 0$. A more rigorous treatment of this is given in Section 10.

We would like to delegate the last step to the provers by asking them to compute a PCP proof that $V(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{A}, \boldsymbol{B}) = 1$, which the verifier can check by communicating only polynomially many bits with the provers. However, we face an obstacle: Alice cannot know $\boldsymbol{y}$ and $\boldsymbol{B}$, and neither can Bob know $\boldsymbol{x}$ and $\boldsymbol{A}$, and distributed PCPs (where neither party knows the entire assignment) are known to be impossible [ARW17]. To proceed, we will first have to modify $\mathscr{G}_1$ by *oracularizing* it:

1. The verifier sends Alice the questions $\boldsymbol{x}, \boldsymbol{y}$, and Bob either $\boldsymbol{x}$ or $\boldsymbol{y}$, chosen uniformly at random.
2. Alice returns exponentially-long answers $\boldsymbol{A}, \boldsymbol{B}$, and Bob returns an answer $\boldsymbol{C}$.
3. The verifier computes a verification predicate $V(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{A}, \boldsymbol{B})$ on Alice's questions and answers, and further checks that $\boldsymbol{A} = \boldsymbol{C}$, if Bob received $\boldsymbol{x}$, or that $\boldsymbol{B} = \boldsymbol{C}$, if Bob received $\boldsymbol{y}$.

The idea is that the new Alice simulates both Alice and Bob from the original protocol, and the new Bob certifies that the new Alice does not take advantage of her access to both questions to cheat. It is well-known that oracularization does not harm the soundness of interactive protocols, be they classical or quantum. However, in the quantum world, it is not necessarily the case that the oracularized protocol retains *completeness*. This is because Alice and Bob may have been asked to perform non-compatible measurements in the original protocol, rendering it impossible for the new Alice to simulate both the original Alice and Bob. Fortunately for us, the honest strategy for protocol $\mathscr{G}_1$ is such that completeness under oracularization.

Now that a single prover is in possession of all inputs to the verification predicate V, we can implement our idea of using a PCP proof. Classically, this idea is known as PCP *composition*, and is extensively used in the PCP literature. In the quantum case, the requirement to maintain soundness against entanglement makes composition technically difficult, and we defer the details to Part V of the paper. Once the composition is performed, we reach our main result.

Informal Theorem 2.7 (Theorem 17.12 in the body)**.** *There is an* MIP^* *protocol* $\mathscr{G}_2$ *for* $\mathsf{Succinct\text{-}Succinct\text{-}3Sat}$ *(and hence for* NEEXP*) with question size, answer, and verifier runtime* $\mathrm{poly}(n)$*.*

2.6 Organization

The paper is organized into five parts. The first part is the introduction and this overview. The remaining parts are organized as follows.

- Part II contains two sections of preliminaries, one containing the classical background and another the quantum background.
- Part III contains the register compiler, i.e. the proof of Informal Theorem 2.5. This involves designing the Pauli basis test (Section 6) and the data hiding test (Section 8). Section 5 serves as an introduction to this part and contains more details on the organization.
- Part IV contains the "introspection" question reduction step, i.e. the proof of Informal Theorem 2.6. To begin, we sketch the classical MIP protocol for $\mathsf{Succinct\text{-}3Sat}$ in Section 11. Then we give the introspected, i.e. "big", low-degree test in Section 13, and finish by giving the entire small-question NEEXP protocol in Section 15. Section 14 contains a test necessary for the protocol called the "intersecting lines test". It allows us carry over the results of the low-degree test from one register to another.
- Part V contains the answer reduction, i.e. the proof of Informal Theorem 2.7. The construction involves composing PCP protocols with error-correcting codes, and so Section 16 surveys the properties we need of an error-correcting code. Finally, Section 17 contains the actual proof of the answer reduction step.

Part II
Preliminaries

3 Classical preliminaries

3.1 Finite fields and polynomials

In this section we review some basic facts about finite fields and polynomials over them. These facts can be found in a standard reference such as [MBG+13]. Let p be a prime and $q = p^t$ be a prime power. We denote by $\mathbb{F}_p$ and $\mathbb{F}_q$ the finite fields with p and q elements, respectively. The field $\mathbb{F}_p$ is called the *base field* or *prime subfield* of $\mathbb{F}_q$. The larger field $\mathbb{F}_q$ can be viewed as a t-dimensional vector space $\mathbb{F}_p^t$ over the smaller field. We define the trace $\mathrm{tr} : \mathbb{F}_q \to \mathbb{F}_p$ by

$$\mathrm{tr}[a] = \sum_{\ell=0}^{t-1} a^{p^\ell}.$$

The trace is a linear map under linear combinations with coefficients drawn from $\mathbb{F}_p$.

A *basis* for $\mathbb{F}_q$ over $\mathbb{F}_p$ consists of k elements $\{\alpha_1, \ldots, \alpha_k\}$, such that any element $u \in \mathbb{F}_q$ can be written as a linear combination

$$u = \sum_{i=1}^{k} c_i \alpha_i,$$

where the coefficients $c_i \in \mathbb{F}_p$. Two bases $\{\alpha_i\}$ and $\{\beta_i\}$ are *dual* bases if $\mathrm{tr}[\alpha_i \beta_j] = \delta_{ij}$, where δ_{ij} is the Kronecker delta function.

Fields of characteristic 2 : When $p = 2$ (i.e. q is even), several useful properties hold. Most importantly for us, the field $\mathbb{F}_q$ has a *self-dual* basis: that is, there exists a basis $\{\alpha_i\}$ such that $\mathrm{tr}[\alpha_i \alpha_j] = \delta_{ij}$ [MBG+13, Theorem 1.9]. This means that given a field element $u = \sum_i c_i \alpha_i$, we can recover the coefficient c_j by the expression $c_j = \mathrm{tr}[u\alpha_j]$.

Fourier identities Below, we give two useful identities for simplifying Fourier sums over finite fields. We set $\omega = e^{2\pi i/p}$ to be a p-th root of unity.

Fact 3.1. $\mathbf{E}_{\boldsymbol{u} \in \mathbb{F}_q} \omega^{\mathrm{tr}[\boldsymbol{u} \cdot a]} = 0$ *if* a *is nonzero.*

Proof. If $a \neq 0$, then there must exist some nonzero $y \in \mathbb{F}_q$ such that $\mathrm{tr}[ay] \neq 0$. Let the value of the expectation we want to compute be denoted by σ. Then we have

$$\begin{aligned}
\sigma &= \mathop{\mathbf{E}}_{\boldsymbol{u} \in \mathbb{F}_q} \omega^{\mathrm{tr}[a\boldsymbol{u}]} \\
&= \mathop{\mathbf{E}}_{\boldsymbol{u} \in \mathbb{F}_q} \omega^{\mathrm{tr}[a(\boldsymbol{u}+y)]} \\
&= \omega^{\mathrm{tr}[ay]} \mathop{\mathbf{E}}_{\boldsymbol{u} \in \mathbb{F}_q} \omega^{\mathrm{tr}[a\boldsymbol{u}]} \\
&= \omega^{\mathrm{tr}[ay]} \sigma,
\end{aligned}$$

and thus $\sigma = 0$. □

Fact 3.2. *Let* V *be a subspace of* $\mathbb{F}_q^n$. *Then* $\mathbf{E}_{\boldsymbol{u} \sim V} \omega^{\mathrm{tr}[\langle \boldsymbol{u}, a \rangle]} = 0$ *if* $a \notin V^\perp$.

Proof. The idea is the same as the proof of the previous fact. Suppose $a \notin V^\perp$. Then there exists some nonzero $y \in \mathbb{F}_q^n$ such that $\mathrm{tr}[\langle a, y\rangle] \neq 0$. Letting the value of the expectation we wish to compute be denoted σ, we have

$$\begin{aligned}\sigma &= \mathop{\mathbf{E}}_{\boldsymbol{u}\sim V} \omega^{\mathrm{tr}[\langle \boldsymbol{u}, a\rangle]} \\ &= \mathop{\mathbf{E}}_{\boldsymbol{u}\sim V} \omega^{\mathrm{tr}[\langle \boldsymbol{u}+y, a\rangle]} \\ &= \omega^{\mathrm{tr}[\langle a, y\rangle]}\sigma,\end{aligned}$$

and hence σ must vanish. □

3.2 Two-player one-round games and MIP

A two-prover nonlocal game is an interaction between a verifier and two noncommunicating provers, in which the verifier samples a pair of random questions and sends them to the provers, receives a pair of answers, and decides whether to accept or reject based on the questions and answers. In the literature, a game is usually taken to be described by the verifier's distribution over question pairs, together with a table describing the verifier's behavior for all possible choices of questions and answers. For our purposes, it will be more convenient to work with *uniformly generated* families of games, which are specified by Turing machines that sample the questions and decide whether to accept or reject given the questions and answers.

Definition 3.3 (Two-player one-round uniform game family)**.** A *two-prover one-round game uniform game family* $\mathscr{G}$ is an interaction between a verifier and two provers, Alice and Bob. The verifier $V = (\mathrm{Alg_Q}, \mathrm{Alg_A})$ consists of a "question" randomized Turing machine $\mathrm{Alg_Q}$ and an "answer" deterministic Turing machine $\mathrm{Alg_A}$. Given an input string input, the verifier samples two questions $(\boldsymbol{x}_0, \boldsymbol{x}_1) \sim \mathrm{Alg_Q}(\mathsf{input})$ and distributes $\boldsymbol{x}_0$ to Alice and $\boldsymbol{x}_1$ to Bob. They reply with answers $\boldsymbol{a}_0$ and $\boldsymbol{a}_1$, respectively, and the verifier accepts if $\mathrm{Alg_A}(\mathsf{input}, \boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{a}_0, \boldsymbol{a}_1) = 1$. A strategy for Alice and Bob is said to be *classical* if they are allowed shared randomness but no shared quantum resources. The *value* of Alice and Bob's strategy is simply the probability that the verifier accepts, and the *classical value* of the game is the maximum value of any classical strategy. We write $\mathsf{Q\text{-}length}(\mathscr{G})$ for the maximum bit length of the questions as a function of the input input, and similarly $\mathsf{A\text{-}length}(\mathscr{G})$ for the maximum bit length of the answers, $\mathsf{Q\text{-}time}(\mathscr{G})$ for the maximum running time of $\mathrm{Alg_Q}$, and $\mathsf{A\text{-}time}(\mathscr{G})$ for the maximum running time of $\mathrm{Alg_A}$. Often we will not explicitly write the dependence of these quantities on input.

Definition 3.4 (Multiprover interactive proofs)**.** A *2-player 1-round multiprover interactive proof protocol* is a uniform game family $\mathscr{G}$ as in Definition 3.3. For parameters $0 < s < c \leq 1$, we say that the protocol $\mathscr{G}$ decides the language L with completeness c and soundness s if the following three conditions are true.

- (Completeness) Suppose $\mathsf{input} \in L$. Then there is a classical strategy for $\mathscr{G}$ with value at least c.
- (Soundness) Suppose $\mathsf{input} \notin L$. Then every classical strategy for $\mathscr{G}$ has value at most s.
- All of $\mathsf{Q\text{-}length}(\mathscr{G})$, $\mathsf{A\text{-}length}(\mathscr{G})$, $\mathsf{Q\text{-}time}(\mathscr{G})$, and $\mathsf{A\text{-}time}(\mathscr{G})$ are $\mathrm{poly}(n)$ where n is the bit length of input.

The class $\mathsf{MIP}_{c,s}$ is the set of all languages that can be decided by multiprover interactive proof protocols with the parameters c, s.

If $c - s$ is a constant, then we will suppress the dependence on them when writing MIP and just say that $L \in \mathsf{MIP}$. Here, "c" is referred to as the *completeness* and "s" is referred to as the *soundness*. We will typically deal with the case when $c = 1$ and $s = 1 - \epsilon$, where $\epsilon > 0$ is a small constant.

In this definition of MIP, the parameters $\mathsf{Q\text{-}length}(\mathscr{G}), \mathsf{A\text{-}length}(\mathscr{G}), \mathsf{Q\text{-}time}(\mathscr{G}), \mathsf{A\text{-}time}(\mathscr{G})$ are required to be polynomial in the input length n. However, in this paper, several of the intermediate results we achieve are protocols where these parameters scale superpolynomially (indeed, even exponentially or worse) in n. In these cases, we will explicitly indicate the dependence of these parameters on n.

3.3 Low-degree code

Let q be a prime power and $h \leq q$ be an integer. Let H be a subset of $\mathbb{F}_q$ of size h. For $n \geq 0$, let $x \in H^n$. The *indicator function of x over H^n* is the polynomial with inputs $y \in \mathbb{F}_q^m$ defined as

$$\mathrm{ind}_{H,x}(y) := \frac{\prod_{i=1}^{m} \prod_{b \in H, b \neq x_i} (y_i - b)}{\prod_{i=1}^{m} \prod_{b \in H, b \neq x_i} (x_i - b)}.$$

There are two properties of this polynomial that we will need:

(i) that it is low-degree, i.e. a degree-$m(h-1)$ polynomial,

(ii) that for any $x, y \in H^m$, $\mathrm{ind}_{H,x}(y) = 1$ if and only if $x = y$, and otherwise $\mathrm{ind}_{H,x}(y) = 0$.

Using this, we can define the low-degree code.

Definition 3.5 (Low-degree encoding)**.** Let $|\mathcal{S}| \leq h^m$, and let $\pi : \mathcal{S} \to H^m$ be an injection. Then the *low-degree encoding* (sometimes also called the *Reed-Muller encoding*) of a string $a \in \mathbb{F}_q^{\mathcal{S}}$ is the polynomial $g_a : \mathbb{F}_q^m \to \mathbb{F}_q$ defined as

$$g_a(x) := \sum_{i \in \mathcal{S}} a_i \cdot \mathrm{ind}_{H,\pi(i)}(x).$$

By the properties of the indicator function above, (i) g_a is a degree-$m(h-1)$ polynomial, and (ii) $g_a(\pi(i)) = a_i$ for all $i \in \mathcal{S}$. We will typically, though not always, take $\mathcal{S} = [n]$. Given an error-correcting code, there are two key properties we care about: the rate and the distance. The rate of the low-degree code is n/q^m. As for the distance, we can estimate it with the following lemma.

Lemma 3.6 (Schwartz-Zippel lemma [Sch80, Zip79])**.** *Let f, g be two unequal m-variate degree-d polynomials over $\mathbb{F}_q$. Then*

$$\Pr_{\boldsymbol{x} \sim \mathbb{F}_q^m}[f(\boldsymbol{x}) = g(\boldsymbol{x})] \leq d/q.$$

As a result, the low-degree encoding has relative distance $m(h-1)/q$. In a typical application, we would like a code with large rate and distance. To achieve this, we will often use the following "rule of thumb" setting of parameters:

$$h = \Theta(\log(n)), \qquad m = \Theta\left(\frac{\log(n)}{\log\log(n)}\right), \qquad q = \mathrm{polylog}(n). \tag{1}$$

This gives a code with rate $1/\mathrm{poly}(n)$ and distance $o(1)$. The polynomials involved are degree $d = \Theta(\log(n)^2/\log\log(n))$.

3.4 A canonical low-degree encoding

The low-degree encoding affords us some flexibility when choosing the parameters and the injection; however, for our application we will have to choose these with care, because each of our uses of the low-degree code requires that the injection π be efficiently computable. In this section, we give a simple, canonical choice for the subset H and the injection π so that this is true.

Definition 3.7. We say that n, $h = 2^{t_1}$, $q = 2^{t_2}$, and m are *admissible parameters* if $t_1 \leq t_2$ and $h^m \geq n$.

The following definition gives the canonical encoding.

Definition 3.8 (Canonical low-degree encoding)**.** Let n, $h = 2^{t_1}$, $q = 2^{t_2}$, and m be admissible parameters. Set $\ell = t_1 \cdot m$. The *canonical low-degree code* is defined as follows.

(i) Let $e_1, \ldots, e_{t_2}$ be a self-dual basis for $\mathbb{F}_q$ over $\mathbb{F}_2$. Then we set H to be the subset

$$H := H_{t_1,t_2} = \{b_1 \cdot e_1 + \cdots + b_{t_1} \cdot e_{t_1} \mid b_1, \ldots, b_{t_1} \in \mathbb{F}_2\}.$$

As desired, $|H| = h$.

(ii) Let $\sigma := \sigma_{t_1,t_2} : \{0,1\}^{t_1} \to H_{t_1,t_2}$ be the bijection $\sigma(b_1, \ldots, b_{t_1}) = b_1 \cdot e_1 + \cdots + b_{t_1} \cdot e_{t_1}$. From this, we can construct a bijection $\sigma_{\ell,t_1,t_2} : \{0,1\}^{\ell} \to H^m$ by setting

$$\sigma_{\ell,t_1,t_2}(b_1, \ldots, b_\ell) = (\sigma(b_1, \ldots, b_{t_1}), \sigma(b_{t_1+1}, \ldots, b_{2t}), \ldots, \sigma(b_{\ell-t_1+1}, \ldots, b_\ell)).$$

(iii) Given an index $i \in [n]$, write $\mathrm{bin}_\ell(i)$ for its ℓ-digit binary encoding. Then we define the injection $\pi := \pi_{\ell,t_1,t_2} : [n] \to H^m$ as $\pi(i) = \sigma_{\ell,t_1,t_2}(\mathrm{bin}_\ell(i))$.

The following proposition gives the time complexity of the canonical low-degree encoding.

Proposition 3.9. *The bijection σ_{ℓ,t_1,t_2} and the injection $\pi := \pi_{\ell,t_1,t_2}$ are both computable in time $m \cdot \mathrm{polylog}(q)$. As a result, given a string $a \in \mathbb{F}_q^n$ and a point $x \in \mathbb{F}_q^m$, the value $g_a(x)$ takes time $\mathrm{poly}(n, m, q)$ to compute.*

3.5 Low-degree testing

Definition 3.10 (Surface-versus-point test)**.** The *surface-versus-point low-degree test with parameters m, d, q* (a prime power), and k, denoted $\mathscr{G}_{\mathrm{Surface}}(m, d, q, k)$, is defined as follows. Let $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k$ be k uniformly random vectors in $\mathbb{F}_q^m$, and let $\boldsymbol{s}$ be a uniformly random affine subspace parallel to $\mathrm{span}\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k\}$ (that is, $\boldsymbol{s}$ is the set $\{\boldsymbol{w} + \lambda_1\boldsymbol{v}_1 + \cdots + \lambda_k\boldsymbol{v}_k : \lambda_1, \ldots, \lambda_k \in \mathbb{F}_q\}$ for a uniformly random $\boldsymbol{w}$), and let $\boldsymbol{u}$ be a uniformly random point on $\boldsymbol{s}$. Given these, the test is performed as follows.

- The vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k$ and the surface $\boldsymbol{s}$ are given to Alice, who responds with a degree-d polynomial $\boldsymbol{f} : \boldsymbol{s} \to \mathbb{F}_q$.
- The point $\boldsymbol{u}$ is given to Bob, who responds with a number $\boldsymbol{b} \in \mathbb{F}_q$.

Alice and Bob pass the test if $\boldsymbol{f}(\boldsymbol{u}) = \boldsymbol{b}$.

Remark 3.11. Let us remark briefly on the encodings used in this test. A surface s with directions $v_1, \ldots, v_k$ is encoded by the string $(u, w_1, \ldots, w_k) \in \mathbb{F}_q^{(k+1)n}$. Here, u is the lexicographically minimum point in s, and $w_1, \ldots, w_k$ are the rows of the matrix produced by taking the matrix with rows $v_1, \ldots, v_k$ and transforming it to reduced row echelon form. We note that given $v_1, \ldots, v_k$ and a point $u \in s$, this encoding can be produced in time $\mathrm{poly}(n, k, \log(q))$.

A function $f : s \to \mathbb{F}_q$ is *a degree-d polynomial on s* if there exists a degree-d k-variate polynomial $f' : \mathbb{F}_q^k \to \mathbb{F}_q$ such that $f'(\lambda_1, \ldots, \lambda_k) = f(u + \lambda_1 w_1 + \cdots + \lambda_k w_k)$. When s is already known, we can encode f by specifying f', which involves writing out its $d[k] := \binom{d+k}{k}$ coefficients in some arbitrary but fixed order.

We note that this definition of the surface-versus-point test differs slightly from the standard definition of the surface-versus-point test in two respects. First, we do not require that the vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k$ be linearly independent or even nonzero, which implies that there is a

$$1 - \left(1 - \frac{1}{q^m}\right)\left(1 - \frac{q}{q^m}\right) \cdots \left(1 - \frac{q^{k-1}}{q^m}\right) \leq \frac{q^k}{q^m}$$

chance that $\boldsymbol{s}$ is less than k-dimensional. Second, we send the vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k$ to Alice in addition to the description of the surface $\boldsymbol{s}$. It is not hard to see that these two modifications do not asymptotically harm the soundness guarantee obtained for the standard plane-versus-point test shown by Raz and Safra [RS97], which we restate here.

Theorem 3.12 ([RS97])**.** *There exist absolute constants $c, c' > 0$ such that the following holds. Suppose Alice and Bob pass $\mathscr{G}_{\mathrm{Surface}}(m, d, q, 2)$ with probability at least μ. Then there exists a degree-d polynomial $g : \mathbb{F}_q^m \to \mathbb{F}_q$ such that*

$$\Pr_{(\boldsymbol{s}, \boldsymbol{u})}[g(\boldsymbol{u}) = \boldsymbol{b}] \geq \mu - c \cdot m(d/q)^{c'}.$$

Explicit values for c, c' have been derived by Moshkovitz and Raz [MR08], albeit for the weaker guarantee that g be a degree-md polynomial, which is still sufficient for most applications.

Communication cost. We can compute the communication cost of this test as follows.

- **Question length:** We encode a plane in $\mathbb{F}_q^m$ with a string $(u, v_1, v_2) \in \mathbb{F}_q^{3m}$. This requires $3m \log(q)$ bits to communicate.
- **Answer length:** A degree-d bivariate polynomial on $\mathbb{F}_q$ can be described with $\binom{d+2}{2} \leq (d+1)^2$ coefficients in $\mathbb{F}_q$. These require $(d+1)^2 \log(q)$ bits to communicate.

Recalling Equation (1), a typical setting of parameters gives questions of length $\Theta(\log(n))$, and answers of length $\Theta(\log(n)^4 / \log\log(n))$.

3.6 Simultaneous low-degree testing

Definition 3.13 (Simultaneous surface-versus-point test)**.** The *simultaneous surface-versus-point low-degree test with parameters m, d, q* (a prime power), *k*, and *ℓ*, denoted $\mathscr{G}^{\ell}_{\mathrm{Surface}}(m, d, q, k)$, is defined as follows. A draw $(\boldsymbol{s}, \boldsymbol{u})$ is sampled as in $\mathscr{G}_{\mathrm{Surface}}(m, d, q, k)$. Given this, the test is performed as follows.

- The surface $\boldsymbol{s}$ is given to Alice, who responds with ℓ degree-d polynomials $\boldsymbol{f}_1, \ldots, \boldsymbol{f}_\ell : \boldsymbol{s} \to \mathbb{F}_q$.

- The point $\boldsymbol{u}$ is given to Bob, who responds with ℓ numbers $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_\ell \in \mathbb{F}_q$.

Alice and Bob pass the test if $\boldsymbol{f}_1(\boldsymbol{u}) = \boldsymbol{b}_1, \ldots, \boldsymbol{f}_\ell(\boldsymbol{u}) = \boldsymbol{b}_\ell$.

Classically, the $k = 2$ case of this test can be reduced to a slight generalization of Theorem 3.12 using a simple and standard union-bound argument. Quantumly, however, a corresponding entanglement-sound analogue of this generalization is not known to hold. Instead, we use a slightly more involved reduction in which the ℓ outputs of Alice and Bob are "combined" to create a strategy for the "standard" plane-versus-point test. (This technique is standard and was also used in the proof of Lemma 4.6 in [NV18a] for the case $\ell = 2$.) In this section, we will introduce the notation needed for this reduction and carry out the proof of the classical soundness of the simultaneous low-degree test as a warm-up for our proof of quantum soundness later. We begin by showing how to combine ℓ functions by introducing ℓ "indexing" variables.

Notation 3.14. Let $g_1, \ldots, g_\ell : s \to \mathbb{F}_q$ be functions, where s is a subset of $\mathbb{F}_q^m$. Then we define the new function $\text{combine}_g(x, y) : \mathbb{F}_q^\ell \otimes s \to \mathbb{F}_q$ as follows:

$$\text{combine}_g(x, y) = x_1 \cdot g_1(y) + \cdots + x_\ell \cdot g_\ell(y).$$

We will typically apply this with $s = \mathbb{F}_q^m$ or s a dimension-k subspace of $\mathbb{F}_q^m$.

If the g_i's are degree-d polynomials on $\mathbb{F}_q^m$, this produces a degree-$(d+1)$ polynomial on $\mathbb{F}_q^{\ell+m}$. First, we show that given a surface-versus-point query from this $(\ell+m)$-dimensional space, we can produce a surface-versus-point query from the m-dimensional space.

Proposition 3.15. *Given a subset $s \subseteq \mathbb{F}_q^{\ell+m}$, let $s_{\text{proj}} = \{y \mid (x, y) \in \mathbb{F}_q^\ell \otimes \mathbb{F}_q^m\}$.*

- *If s is a dimension-k subspace of $\mathbb{F}_q^{\ell+m}$, then s_{proj} is a dimension-k' subspace of $\mathbb{F}_q^m$, for $k' \leq k$.*

Define $\boldsymbol{s}' \sim_k s_{\text{proj}}$ to be a uniformly random dimension-k subspace of $\mathbb{F}_q^m$ containing s_{proj}.

- *If $\boldsymbol{s}$ and $(\boldsymbol{x}, \boldsymbol{y})$ are distributed as $\mathscr{D}_{\text{Surface}}(\ell+m, q, k)$, then $\boldsymbol{s}' \sim \boldsymbol{s}_{\text{proj}}$ and $\boldsymbol{y}$ are distributed as $\mathscr{D}_{\text{Surface}}(m, q, k)$.*

Proof. The first bullet follows because if $\{(x_1, y_1), \ldots, (x_k, y_k)\}$ is a set of k linearly independent vectors which span s, then $\{y_1, \ldots, y_k\}$ is a set of k vectors which span s_{span}, though they may no longer be linearly independent. The second bullet follows by symmetry. □

Next, we show that answers to the queries on the m-dimensional space can be used to produce answers to the queries on the $(\ell + m)$-dimensional space.

Proposition 3.16. *Let s be a dimension-k subspace of $\mathbb{F}_q^{\ell+m}$, and let $s' \subseteq \mathbb{F}_q^m$ be a subspace which contains s_{proj}. Then $\mathbb{F}_q^\ell \otimes s'$ is a subspace, and it contains s. In particular, if $f_1, \ldots, f_\ell$ are degree-d functions on s', then combine_f is a degree-$(d+1)$ function on $\mathbb{F}_q^\ell \otimes s'$, and it can be restricted to a degree-$(d+1)$ function on s.*

Proof. Consider a point $(x, y) \in s$. Then $y \in s_{\text{proj}} \subseteq s'$, and so $(x, y) \in \mathbb{F}_q^\ell \otimes s$. The statement about combine_f follows immediately. □

Finally, we need a technical result: that nonlinear low-degree polynomials rarely become linear after restricting variables.

Definition 3.17. Let $n \geq 0$. A function $f : \mathbb{F}_q^{\ell+n} \to \mathbb{F}_q$ is *exactly linear in x* if it can be written as

$$f(x, y) = x_1 \cdot f_1(y) + \cdot + x_\ell \cdot f_\ell(y).$$

(We do not allow constant terms.) Note that when $n = 0$, such a function can be written as $c_1 \cdot x_1 + \cdots + c_\ell \cdot x_\ell$, where each $c_i \in \mathbb{F}_q$, in which case we simply call it "exactly linear". Given a function $f(x, y)$ and a string $y \in \mathbb{F}_q^m$, we will also write $f|_y$ for the function defined as $f_y(x) = f(x, y)$.

Proposition 3.18. *Suppose $f(x, y) : \mathbb{F}_q^{\ell+m} \to \mathbb{F}_q$ is a degree-d polynomial which is not exactly linear in x. Then the probability that $f|_{\boldsymbol{y}}$ is exactly linear, over a uniformly random $\boldsymbol{y} \sim \mathbb{F}_q^m$, is at most d/q.*

Proof. Because f is not exactly linear in x, it contains some non-linear x-monomial $x^i = x_1^{i_1} \cdots x_\ell^{i_\ell}$ in which $i_1 + \cdots + i_\ell$ is either zero or at least two. Thus, f can be written as $f(x, y) = x^i \cdot g_i(y) + f'(x, y)$, where $g_i(y)$ is degree-d and f' contains no x^i terms. For $f|_{\boldsymbol{y}}$ to be exactly linear, this term must vanish, which means $g_i(\boldsymbol{y}) = 0$. But by Schwartz-Zippel (Lemma 3.6), this happens with probability at most d/q. □

We are now ready to prove soundness of the simultaneous low-degree test in the $k = 2$ case.

Theorem 3.19. *There exists absolute constants $c, c' > 0$ such that the following holds. Suppose Alice and Bob pass $\mathscr{G}^{\ell}_{\text{Surface}}(m, d, q, 2)$ with probability at least μ. Then there exist degree-d polynomials $g_1, \ldots, g_\ell : \mathbb{F}_q^m \to \mathbb{F}_q$ such that*

$$\Pr_{(\boldsymbol{s}, \boldsymbol{u})} [g_1(\boldsymbol{u}) = \boldsymbol{b}_1, \ldots, g_\ell(\boldsymbol{u}) = \boldsymbol{b}_\ell] \geq \mu - c \cdot (m + \ell)(d/q)^{c'}.$$

Proof. Let $c, c' > 0$ be as in Theorem 3.12. We pick the constants in this theorem, say $\hat{c}, \hat{c}'$ so that

$$\mu - c \cdot (m + \ell)((d + 1)/q)^{c'} - 2(d + 1)/q \geq \mu - \hat{c} \cdot (m + \ell)(d/q)^{\hat{c}'}.$$

Note that this means that the theorem is trivial when $2(d+1)/q \geq \mu - c \cdot (m+\ell)((d+1)/q)^{c'}$. As such, we will assume below that

$$2(d + 1)/q < \mu - c \cdot (m + \ell)((d + 1)/q)^{c'}. \tag{2}$$

Suppose Alice and Bob pass $\mathscr{G}^{\ell}_{\text{Surface}}(m, d, q, 2)$ with probability at least μ. We will use them to simulate two provers, "Combined Alice" and "Combined Bob", who pass the single-function low-degree test $\mathscr{G}_{\text{Surface}}(\ell + m, d + 1, q, 2)$ with probability at least μ. They are specified as follows:

- **Combined Alice:** Given $\boldsymbol{s} \subseteq \mathbb{F}_q^{\ell+m}$, draw $\boldsymbol{s}' \sim_2 \boldsymbol{s}_{\text{proj}}$. Give it to Alice, who responds with $\boldsymbol{f}_1, \ldots, \boldsymbol{f}_\ell : \boldsymbol{s}' \to \mathbb{F}_q$. Output the function $\text{combine}_{\boldsymbol{f}|_{\boldsymbol{s}}}$.
- **Combined Bob:** Given $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{F}_q^{\ell+m}$, compute $\boldsymbol{y} \in \mathbb{F}_q^m$. Give it to Bob, who responds with $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_\ell \in \mathbb{F}_q$. Return $\text{combine}_{\boldsymbol{b}}(\boldsymbol{x}) \in \mathbb{F}_q$.

By Proposition 3.15, $\boldsymbol{s}'$ and $\boldsymbol{y}$ are distributed as the questions in $\mathscr{G}^{\ell}_{\text{Surface}}(m, d, q, 2)$. Using our assumption on Alice and Bob, this means that $\boldsymbol{f}_1(\boldsymbol{y}) = \boldsymbol{b}_1, \ldots, \boldsymbol{f}_\ell(\boldsymbol{y}) = \boldsymbol{b}_\ell$ with probability at least μ. As a result, $(\text{combine}_{\boldsymbol{f}|_{\boldsymbol{s}}})(\boldsymbol{x}, \boldsymbol{y}) = \text{combine}_{\boldsymbol{b}}(\boldsymbol{y})$ with probability at least μ. By Proposition 3.16, $\text{combine}_{\boldsymbol{f}|_{\boldsymbol{s}}}$ is a degree-$(d+1)$ function on $\boldsymbol{s}$, and so it is a valid response to subspace queries. This means Combined Alice and Bob pass $\mathscr{G}_{\text{Surface}}(\ell + m, d + 1, q, 2)$ with probability at least μ.

Thus, we can apply Theorem 3.12. It gives a degree-$(d+1)$ function $g : \mathbb{F}_q^{\ell+m} \to \mathbb{F}_q$ such that

$$\Pr_{\boldsymbol{x},\boldsymbol{y}}[g(\boldsymbol{x},\boldsymbol{y}) = \text{combine}_{\boldsymbol{b}}(\boldsymbol{x})] \geq \mu - c \cdot (\ell+m) \cdot ((d+1)/q)^{c'}. \tag{3}$$

We would like to show that g is exactly linear in x. Assume for the sake of contradiction that this is not the case. Because $\boldsymbol{b}$ depends only on $\boldsymbol{y}$ (and Bob's internal randomness), we can consider varying these two variables independently of $\boldsymbol{x}$. By Proposition 3.18, the probability that $g_{\boldsymbol{y}}$ is not exactly linear is at least $1-(d+1)/q$. In this case, because $\text{combine}_b(\boldsymbol{x})$ is always exactly linear, the probability that $g|_{\boldsymbol{y}}(\boldsymbol{x}) = \text{combine}_{\boldsymbol{b}}(\boldsymbol{x})$ is at most $(d+1)/q$ by Schwartz-Zippel (Lemma 3.6). As a result, the probability that $g(\boldsymbol{x},\boldsymbol{y}) = \text{combine}_{\boldsymbol{b}}(\boldsymbol{x})$ is at most $(d+1)/q + (d+1)/q$, which contradicts Equations (2) and (3). Thus, we may conclude that g is exactly linear in x.

This implies that we can write $g(x,y) = \sum_i x_i \cdot g_i(y)$, where each g_i is a degree-d polynomial. Now, for any fixed b and y, if it is not the case that $g_1(y) = b_1, \ldots, g_\ell(y) = b_\ell$, then the probability that $g(\boldsymbol{x},y) = \text{combine}_b(\boldsymbol{x})$ over a random $\boldsymbol{x}$ is at most $1/q$ by Schwartz-Zippel since both are exactly linear functions. Thus, if η is the probability that $g_1(\boldsymbol{y}) = \boldsymbol{b}_1, \ldots, g_\ell(\boldsymbol{y}) = \boldsymbol{b}_\ell$, then the probability that $g(\boldsymbol{x},\boldsymbol{y}) = \text{combine}_{\boldsymbol{b}}(\boldsymbol{x})$ is at most $\eta + (1-\eta)/q \leq \eta + 1/q$. Combined with Equation (3), this implies the theorem. □

3.7 NEXP, NEEXP, and complete problems for them

Definition 3.20. The class NEEXP (respectively, NEEXP) is the class of all problems that can be solved in exponential (respectively, doubly-exponential) time by a nondeterministic Turing machine. Formally,

$$\mathsf{NEXP} = \bigcup_{c\in\mathbb{N}} \mathsf{NTIME}(2^{n^c}), \qquad \mathsf{NEEXP} = \bigcup_{c\in\mathbb{N}} \mathsf{NTIME}(2^{2^{n^c}}).$$

A standard way of generating NEXP-complete problems is by considering "succinct" versions of NP-complete problems, in which an exponential-sized input is encoded by a polynomial-sized circuit. The canonical complete problem is a succinct version of 3Sat, but there is considerable freedom in choosing the succinct encoding used. We choose the following encoding.

Definition 3.21. Succinct-3Sat is the following problem.

- **Input:** a circuit $\mathcal{C}$ with $3n+3$ input bits and size $\text{poly}(n)$. It encodes the 3-Sat instance $\psi_{\mathcal{C}}$ with variable set x_u for $u \in \{0,1\}^n$ which includes the constraint $(x_{u_1}^{b_1} \vee x_{u_2}^{b_2} \vee x_{u_3}^{b_3})$ whenever

  $$\mathcal{C}(u_1,u_2,u_3,b_1,b_2,b_3) = 1.$$

  (Here, x_i^1 refers to the literal x_i and x_i^0 refers to the negated literal $\overline{x_i}$.)

- **Output:** accept if $\psi_{\mathcal{C}}$ is satisfiable and reject otherwise.

A proof that Succinct-3Sat is NEXP complete can be found in [Pap94, Chapter 20], albeit with a different encoding. Below, we show this implies NEXP-completeness for our encoding as well.

Proposition 3.22. Succinct-3Sat *is* NEXP*-complete.*

Proof. Papadimitriou [Pap94] considers circuits $\mathcal{C}_{\text{Pap}}$ which encode 3Sat formulas ϕ with n variables and m clauses as follows: $\mathcal{C}_{\text{Pap}}$ takes as input a string (b,u,k), where $b,k \in \{0,1\}^2$ are interpreted as integers in $\{0,1,2,3\}$ and $u \in \{0,1\}^{\log(m)}$ is interpreted either as a vertex $1 \leq u \leq n$ or a clause $1 \leq u \leq m$. If $1 \leq u \leq n$ and $0 \leq k \leq 2$, then on input $(0,u,k)$, $\mathcal{C}_{\text{Pap}}$ outputs the index of the clause

where $\overline{x_u}$ appears for the k-th time, and on input $(1, u, k)$, it outputs the index of the clause where x_u appears for the k-th time. (In addition, if $1 \le u \le m$ and $0 \le k \le 3$, then on input $(2, u, k)$, $\mathcal{C}_{\text{Pap}}$ outputs the k-th literal of the u-th clause in ϕ. We state this for completeness, though we will not need it for the proof.) Such a 3Sat formula ψ has $2n$ literals, each occurring 3 times, and so $m = 2n$. By [Pap94, Chapter 20], this succinct encoding of 3Sat is NEEXP-complete. Using this, we can generate an instance of the Succinct-3Sat problem $\mathcal{C}$ such that $\phi_{\mathcal{C}} = \phi$ as follows: given input $(u_1, u_2, u_3, b_1, b_2, b_3)$, we simply evaluate $\mathcal{C}_{\text{Pap}}$ on (b_i, u_i, k), for each $1 \le i \le 3$ and $0 \le k \le 2$, and output 1 if there is any clause containing all three literals. □

The complete problem for NEEXP is, appropriately enough, a succinct version of Succinct-3Sat. To define it precisely, it helps to fix a notion of a Boolean circuit. Following Section 4.3 of [Pap94], we consider Boolean circuits in which each gate can be one of six types: input, true, false, $\wedge$, $\vee$, or $\neg$. These gates have 0, 0, 0, 2, 2, or 1 inputs, respectively. A succinct representation of a circuit $\mathcal{C}_1$ is a circuit $\mathcal{C}_2$ that, given an index i, outputs the type of gate i as well as the indices j_1, j_2 of its inputs (one or both of these indices may be the null index $\varnothing$ depending on the type of the gate i).

Definition 3.23. Succinct-Succinct-3Sat is the following problem.

- **Input:** a circuit $\mathcal{C}$ with size $\text{poly}(n)$, which is a succinct representation of a circuit $\mathcal{C}'$, which is itself an instance of Succinct-3Sat with instance size $N = 2^{\text{poly}(n)}$.
- **Output:** accept if $\psi_{\mathcal{C}'}$ (the 3Sat formula on $2^N = 2^{2^{\text{poly}(n)}}$ variables generated by the circuit $\mathcal{C}'$) is satisfiable and reject otherwise.

Fact 3.24. *Let M be a deterministic Turing machine which takes two inputs x_1, x_2. Then for any input x_1 of size n_1 and for any size parameter n_2 and time $T > n_1 + n_2$, there exists a circuit C_{M,T,x_1} of size $N = O(T^2)$ which, on an input x_2 of size n_2, computes M run for T steps on the input pair x_1, x_2. Moreover, there exists a Turing machine M' that given x_1, n_2, and an index $i \in \{1, \dots, N\}$ in binary, outputs in polynomial time the type of the ith gate of C_{M,T,x_1} and the indices j_1, j_2' of the inputs to this gate.*

Proof. The construction in the proof of Theorem 8.1 of [Pap94] yields a circuit of the desired size. This circuit consists of $O(T^2)$ copies of a constant-sized circuit C_M that depends only on M. □

Theorem 3.25 (Cook-Levin)**.** *Let L be a language in $\mathsf{NTIME}(T(n))$. Then the following properties hold:*

1. *For every string x of length n, there exists a 3Sat formula Φ_x on $n' = \text{poly}(T(n))$ variables $z_1, \dots, z_{n'}$, such that $x \in L$ iff Φ_x is satisfiable.*
2. *There exists a Turing machine R that given an input x of length n, three indices $u_1, u_2, u_3 \in \{1, \dots, n'\}$ in binary, and three bits $b_1, b_2, b_3 \in \{0, 1\}$, runs in $\text{poly}\log(n') = \text{poly}\log(T(n))$ time and outputs 1 iff the clause $(z_{u_1}^{b_1}, z_{u_2}^{b_2}, z_{u_3}^{b_3})$ is included in Φ_x.*

Proof. We follow the proof of Theorem 8.2 of [Pap94] to obtain the 3Sat instance Φ_x. □

Theorem 3.26. Succinct-Succinct-3Sat *is complete for* NEEXP *under polynomial time mapping reductions. That is, for any language L in* NEEXP*, there exists a Turing machine R which takes as input a string $x \in \{0,1\}^n$, and in time $\text{poly}(n)$ outputs an instance $\mathcal{C}_x$ of* Succinct-Succinct-3Sat*, such that $\mathcal{C}_x$ is satisfiable iff $x \in L$.*

Proof. Suppose we start with a language $L \in \mathsf{NEEXP}$. This means there is a nondeterministic Turing machine M which decides L in time $T_0 = 2^{2^{n^c}}$. By Theorem 3.25, there exists a Turing machine R_1 which runs in time $T_1 = \text{poly}\log(T_0) = 2^{O(n^c)}$ such that given input x and clause indices $i = (u_1, u_2, u_3, b_1, b_2, b_3)$, represented as a binary string of length $|i| = \log(\text{poly}(T_0)) = \log(2^{O(2^{n^c})}) = O(2^{n^c})$, runs in time polynomial in ℓ and outputs 1 iff the corresponding clause exists in a $\mathsf{3Sat}$ formula Φ_x such that $x \in L$ iff Φ_x is satisfiable.

Now, if we apply Fact 3.24 to R_1, with x playing the role of the first input x_1 and i the role of the second input, we obtain that for every x there exists a circuit $C_{R_1,T_1,x}$ of size $O(T_1^2) = 2^{O(n^c)}$ which takes as input a tuple of indices i, and runs R_1 for time T_1 on this time to output whether clause i is present in the formula Φ_x. Moreover, there exists a Turing machine R_2 that, given x, the size parameter $|i| = O(2^{n^c})$, represented in binary as a string of $O(n^c)$ bits, and an index j, represented as a string of $O(n^c)$ bits, outputs the jth gate of $C_{R_1,T_1,x}$ in time $T_2 = \text{poly}(n)$. Note that R_2 is a Turing machine which takes in input of size $\text{poly}(n)$ and runs in time $\text{poly}(n)$.

We are now almost where we need to be. In the final step, we once again apply Fact 3.24 to R_2, obtaining a third Turing machine R_3 that takes as input x and the size parameters, and an index k, and generates the kth gate of the circuit $C_{R_2,T_2,x}$ corresponding to running R_2 for T_2 steps. Finally, by fixing the dependence of the size parameters on the size of x, and iterating through all possible values of the index parameter, we obtain a Turing machine R_3' that takes as input x and runs in time $\text{poly}(n)$, and outputs the complete description of a $\mathsf{Succinct\text{-}Succinct\text{-}3Sat}$ instance $\mathcal{C}_x$ with the desired properties. □

3.8 The Tseitin transformation

In this section, we introduce the Tseitin transformation, which is a simple method of converting a Boolean circuit into a Boolean formula.

Definition 3.27 (Tseitin transformation)**.** Let $\mathcal{C}$ be a Boolean circuit with n input variables $x_1, \ldots, x_n$ and s gates. Then the *Tseitin transformation of* $\mathcal{C}$, denoted $\mathcal{F} := \text{Tseitin}(\mathcal{C})$, is the Boolean formula defined as follows.

(i) Introduce new variables $w_1, \ldots, w_s$ corresponding to the output wires of the gates in $\mathcal{C}$. Then the input variables to $\mathcal{F}$ consist of $x_1, \ldots, x_n$ along with $w_1, \ldots, w_s$.

(ii) Each gate in $\mathcal{C}$ operates on one or two variables in $\{x_1, \ldots, x_n, w_1, \ldots, w_s\}$. Write $g_i(x, w)$ for the function computed by the i-th gate. Then $\mathcal{F}$ computes the intermediate expression

$$z_i := (g_i(x, w) \wedge w_i) \vee (\overline{g_i(x, w)} \wedge \overline{w_i}).$$

The final output of $\mathcal{F}$ is $z_1 \wedge (z_2 \wedge (\cdots \wedge z_s))$.

By construction, $\mathcal{C}(x) = 1$ if and only if there exists a w such that $\mathcal{F}(x, w) = 1$ (in particular, w is taken to be the wire values of $\mathcal{C}$ on input x). In addition, $\mathcal{F}$ contains exactly $7s + (s - 1)$ gates, meaning that it has size $O(s)$.

Next, we show how to convert Boolean formulas into functions over $\mathbb{F}_q$.

Definition 3.28 (Arithmetization)**.** Let $\mathcal{F}$ be a Boolean formula of n variables and size s. The *arithmetization of* $\mathcal{F}$ *over* $\mathbb{F}_q$, denoted $\text{arith}_q(\mathcal{F})$, is the formula produced by the following two-step process.

(i) Transform $\mathcal{F}$ by replacing all $\vee$ gates with appropriate $\wedge$ and $\neg$ gates.

(ii) Transform each Boolean gate into an $\mathbb{F}_q$ gate as follows: Replace each $\wedge$ gate in $\mathcal{F}$ with a $\times$ gate. Replace each $\neg$ gate with a $\times -1$ gate followed by a $+1$ gate (enacting the transformation $b \in \mathbb{F}_q \mapsto 1 - b$). Call the resulting formula $\mathrm{arith}_q(\mathcal{F})$.

Set $\mathcal{F}_{\mathrm{arith}} := \mathrm{arith}_q(\mathcal{F})$. On inputs $x \in \{0,1\}^n$, $\mathcal{F}_{\mathrm{arith}}(x) = \mathcal{F}(x)$. On general inputs $x \in \mathbb{F}_q^n$, $\mathcal{F}_{\mathrm{arith}}(x)$ is computable in time $\mathrm{poly}(s, q)$.

The following proposition shows that small Boolean formulas have low-degree arithmetizations.

Proposition 3.29 (Low-degree arithmetization)**.** *Let $\mathcal{F}$ be a Boolean formula of n variables, size s, and m gates. Then $\mathrm{arith}_q(\mathcal{F})$ is a degree-s polynomial over $\mathbb{F}_q$.*

Proof. By induction on the number of gates, the base case ($m = 0$) being trivial. For the induction hypothesis, assume the proposition holds for Boolean formulas which have fewer than m gates. Either the gate at the root of $\mathcal{F}$ is a $\neg$ gate or an $\{\vee, \wedge\}$-gate. In the former case, $\mathcal{F} = \neg\mathcal{F}'$ for some Boolean formula with $m - 1$ gates, and so $\mathrm{arith}_q(\mathcal{F}) = 1 - \mathrm{arith}_q(\mathcal{F}')$ by construction. But these have the same degree, and so $\mathrm{arith}_q(\mathcal{F})$ is degree s by the induction hypothesis. In the latter case, assume without loss of generality that it is an $\wedge$-gate. Then $\mathcal{F} = \mathcal{F}_{\mathrm{left}} \wedge \mathcal{F}_{\mathrm{right}}$ for two formulas of size $s_{\mathrm{left}} + s_{\mathrm{right}} = s$ and fewer than m gates. By the induction hypothesis, $\mathrm{arith}_q(\mathcal{F}_{\mathrm{left}})$ has degree-s_{left} and $\mathrm{arith}_q(\mathcal{F}_{\mathrm{right}})$ has degree-s_{right}, and so $\mathrm{arith}_q(\mathcal{F}) = \mathrm{arith}_q(\mathcal{F}_{\mathrm{left}}) \times \mathrm{arith}_q(\mathcal{F}_{\mathrm{right}})$ has degree s. □

The arithmetization procedure describe in Definition 3.28 can also be applied to general Boolean circuits $\mathcal{C}$, not just Boolean formulas. But Proposition 3.29 does not apply to general circuits; in fact, the arithmetization of a Boolean circuit can have very high degree, even if that circuit is small. This motivates using the Tseitin transformation: it allows us to convert a small circuit into a small formula, which has a low-degree arithmetization.

4 Quantum preliminaries

4.1 Quantum measurements

The most general notion of a quantum measurement is a POVM measurement, which consists of a set of Hermitian operators $\{M_a\}_{a \in S}$ indexed by outcomes a from a set S. These satisfy the conditions

$$\forall a,\ M_a \succeq 0, \qquad \sum_a M_a = I.$$

To refer to the measurement as a whole we will use the letter M, without the subscript indicating the outcome. For a state $|\psi\rangle$, the probability that the measurement M returns outcome a is

$$\mathbf{Pr}[\boldsymbol{a}] = \langle\psi| M_{\boldsymbol{a}} |\psi\rangle .$$

A POVM is said to be *projective* if each element M_a is an orthogonal projector, i.e. $M_a^2 = M_a$. Note that this implies that $M_a M_b = 0$ for any $a \neq b$, i.e. that the projectors are pairwise orthogonal. Naimark's theorem says that any POVM measurement can be simulated by a projective measurement on an enlarged space.

Theorem 4.1 (Naimark)**.** *Suppose $\{M_a\}$ is a POVM acting on a Hilbert space $\mathcal{H}$. Then there exists a projective measurement $\{M'_a\}$ acting on the space $\mathcal{H} \otimes \mathcal{H}_{\mathrm{aux}}$ together with a state $|\mathrm{aux}\rangle$*

such that for all states $|\psi\rangle \in \mathcal{H}$ *and all outcomes* a*, the post-measurement state after applying* M *and* M' *is the same:*

$$\sqrt{M_a}\,|\psi\rangle\,\langle\psi|\,\sqrt{M_a} = \mathrm{tr}_{\mathsf{aux}}(M'_a(|\psi\rangle\,\langle\psi| \otimes |\mathrm{aux}\rangle\,\langle\mathrm{aux}|)M'_a). \tag{4}$$

As a consequence, M *and* N *induce the same distribution over outcome probabilities:*

$$\langle\psi|\,M_a\,|\psi\rangle = (\langle\psi| \otimes \langle\mathrm{aux}|)N_a(|\psi\rangle \otimes |\mathrm{aux}\rangle).$$

Moreover, given any upper-bound n *on the number of outcomes of* M_a*, there is a universal choice of the state* $|\mathrm{aux}\rangle$ *that works for all POVMs* M_a *with at most* n *outcomes. The projective measurement* M'_a *and state* $|\mathrm{aux}\rangle$ *together constitute a* Naimark dilation *of the POVM* M_a*.*

Theorem 4.2 (Partial Naimark)**.** *Suppose* $\{M_{a_1,a_2}\}$ *is a POVM acting on a tensor product Hilbert space* $\mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2$ *of the form* $M_{a_1,a_2} = \Pi_{a_1} \otimes A^{a_1}_{a_2}$*, where the operators* $\{\Pi_{a_1}\}_{a_1}$ *is a projective measurement. Then there is a Naimark dilation* M'_{a_1,a_2} *and a state* $|\mathrm{aux}\rangle$ *as above, with the property that* $M'_{a_1,a_2} = \Pi_{a_1} \otimes A'^{a_1}_{a_2}$ *for projectors* $A'^{a_1}_{a_2}$ *acting on* $\mathcal{H}_2 \otimes \mathcal{H}_{\mathrm{aux}}$*.*

Proof. The condition that $\{\Pi_{a_1}\}$ forms a projective measurement implies that for each a_1, $\{A^{a_1}_{a_2}\}$ is a POVM. By Theorem 4.1, for each a_1 there exists a POVM $A'^{a_1}_{a_2}$ dilating $A^{a_1}_{a_2}$, and all of these POVMs act on the same universal auxiliary state $|\mathrm{aux}\rangle$. Now, define $M'_{a_1,a_1} = \Pi_{a_1} \otimes A'^{a_1}_{a_2}$. For every state $|\psi\rangle$, we have that

$$\begin{aligned}
\mathrm{tr}_{\mathsf{aux}}(M'_{a_1,a_2}(|\psi\rangle\,\langle\psi|)M'_{a_1,a_2}) &= \mathrm{tr}_{\mathsf{aux}}(A'^{a_1}_{a_2}(\Pi_{a_1}\,|\psi\rangle\,\langle\psi|\,\Pi_{a_1} \otimes |\mathrm{aux}\rangle\,\langle\mathrm{aux}|)A'^{a_1}_{a_2}) \\
&= \sqrt{A^{a_1}_{a_2}}(\Pi_{a_1}\,|\psi\rangle\,\langle\psi|\,\Pi_{a_1})\sqrt{A^{a_1}_{a_2}} \\
&= \sqrt{M_{a_1,a_2}}\,|\psi\rangle\,\langle\psi|\,\sqrt{M_{a_1,a_2}},
\end{aligned}$$

where in going from the first to the second line we used Theorem 4.1. □

For the purposes of this paper, we will need to specialize the POVM notation introduced above in several ways. First, we will often work with families of POVM measurements indexed by questions in an interactive proof protocol. These will be denoted $\{M^q_a\}$, where q indexes the question and a the outcome. (We note that the reverse convention "$\{M^a_q\}$", which we will *not* use, is also common in the literature.) In many cases, the outcomes will consist of tuples of elements, some of which we may wish to discard. We use the convention that if an outcome element is not written, it is understood to be *summed* over. Thus, if $\{M^x_{a,b}\}$ is a family of POVMs, we would have

$$M^x_a := \sum_b M^x_{a,b}. \tag{5}$$

Notation 4.3. We will also often consider situations where some of the information in a measurement outcome is discarded. In particular, given a POVM $\{M_f\}$ whose outcomes are functions $f : U \to V$ over some domain U, and given a point $x \in U$, we will denote by $\{M_{f(x)=y}\}_{y\in V}$ the measurement corresponding to applying M to obtain a function f, and returning the value of f at x. Formally, the POVM elements of this measurement are given by

$$M_{[f(x)=a]} = \sum_{f:f(x)=a} M_f.$$

(We note that Equation (5) can be viewed as a special case in which the "discarding function" f simply removes the second coordinate. For this case, it is simpler to use the convenient notation in Equation (5) than the more cumbersome bracket notation given here.)

The following lemma contains a useful fact about marginalized projective measurements.

Lemma 4.4. *Let $M_{a,b}$ be a projective measurement on a tensor product Hilbert space $\mathcal{H}_1 \otimes \mathcal{H}_2$, and suppose that for all a, $M_a = A_a \otimes B_a$ where A_a is a rank-one matrix on $\mathcal{H}_1$. Then for all a, b, $M_{a,b} = A_a \otimes C_{a,b}$ with $C_{a,b}$ projectors.*

Proof. By the Schmidt decomposition, we can write $M_{a,b}$ as

$$\begin{aligned} M_{a,b} &= \sum_j |\psi_{a,b,j}\rangle \langle \psi_{a,b,j}| \\ &= \sum_{j,k} \sigma_{jk} |u_{a,b,j,k}\rangle \langle u_{a,b,j,k}| \otimes |v_{a,b,j,k}\rangle \langle v_{a,b,jk}| . \end{aligned}$$

Write the rank-one matrix A_a as an outer product $|\psi_a\rangle \langle \psi_a|$. Then we have

$$\sum_b M_{a,b} = \sum_{j,k,b} \sigma_{jk} |u_{a,b,j,k}\rangle \langle u_{a,b,j,k}| \otimes |v_{a,b,j,k}\rangle \langle v_{a,b,jk}| = |\psi_a\rangle \langle \psi_a| \otimes B_a.$$

Taking the partial trace on the B system, we have

$$\mathrm{tr}_B(\sum_b M_{a,b}) = \sum_{j,k,b} \sigma_{jk} |u_{a,b,j,k}\rangle \langle u_{a,b,j,k}| = |\psi_a\rangle \langle \psi_a| .$$

Suppose we multiply on the left by $\langle v|$ and on the right by $|v\rangle$, for $|v\rangle$ orthogonal to $|\psi_a\rangle$. Then the RHS is 0 while the LHS is a sum $\sum_{j,k,b} \sigma_{jk} |\langle v|u_{a,b,j,k}\rangle|^2$ of nonnegative terms. Hence, each of these terms must be zero. Thus, the equation can only hold if all the vectors $|u_{a,b,j,k}\rangle$ are multiples of $|\psi_a\rangle$. This implies that

$$M_{a,b} = |\psi_a\rangle \langle \psi_a| \otimes C_{a,b}$$

for some $C_{a,b}$ as desired. □

4.2 Nonlocal games and MIP*

Now, we augment Definitions 3.3 and 3.4 to allow for provers to share quantum resources.

Definition 4.5. Given a game $\mathscr{G}$, a *quantum strategy* is one in which Alice and Bob are allowed to share entanglement but not to communicate. We can model their behavior with the *strategy* $\mathcal{S} = (\rho, A, B)$. Here,

- Write $\mathcal{H}_A$ for Alice's local Hilbert space and $\mathcal{H}_B$ for Bob's. Then ρ is a (possibly entangled) state in $\mathcal{L}(\mathcal{H}_A \otimes \mathcal{H}_B)$.
- The set A contains a matrix A_a^x for each question x and answer a, with the guarantee that for each question x, $A^x := \{A_a^x\}_a$ is a POVM. (Likewise for B.)

Alice and Bob perform their strategy as follows: given question x, Alice performs the POVM $\{A_a^x\}_a$ and returns her measurement outcome to the verifier. Bob plays similarly. The *value* of their strategy, denoted $\mathrm{val}_{\mathscr{G}}(\mathcal{S})$, is the probability that they pass the test, over the randomness in $\mathscr{G}$ and in their measurement outcomes.

$$\begin{aligned} \mathrm{val}_{\mathscr{G}}(\mathcal{S}) &= \mathop{\mathbf{E}}_{(\boldsymbol{x}_0,\boldsymbol{x}_1)\sim \mathrm{Alg_Q}} \mathop{\mathbf{Pr}}_{\boldsymbol{a}_0,\boldsymbol{a}_1} [\mathrm{Alg_A}(\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{a}_0, \boldsymbol{a}_1) = 1] \\ &= \mathop{\mathbf{E}}_{(\boldsymbol{x}_0,\boldsymbol{x}_1)\sim \mathrm{Alg_Q}} \sum_{\substack{a_0,a_1, \\ \mathrm{Alg_A}(\boldsymbol{x}_0,\boldsymbol{x}_1,a_0,a_1)=1}} \mathrm{tr}(A_{a_0}^{\boldsymbol{x}_0} \otimes A_{a_1}^{\boldsymbol{x}_1} \cdot \rho), \end{aligned}$$

where in the first line, $(\boldsymbol{a}_0, \boldsymbol{a}_1)$ is the distribution on answers given questions $\boldsymbol{x}_0, \boldsymbol{x}_1$. We write $\mathrm{val}(\mathscr{G})$ for the infimum of $\mathrm{val}_{\mathscr{G}}(\mathcal{S})$ over all strategies $\mathcal{S}$. We define value analogously for interactive proofs.

We say that $L \in \mathsf{MIP}^*_{c,s}$ if there is an quantum interactive proof $\mathscr{G}$ that decides it. This means that the following three conditions are true.

- (Completeness) Suppose $\mathsf{input} \in L$. Then there is a quantum strategy for $\mathscr{G}$ with value at least c.
- (Soundness) Suppose $\mathsf{input} \notin L$. Then every quantum strategy for $\mathscr{G}$ has value at most s.
- All of $\mathsf{Q\text{-}length}(\mathscr{G})$, $\mathsf{A\text{-}length}(\mathscr{G})$, $\mathsf{Q\text{-}time}(\mathscr{G})$, and $\mathsf{A\text{-}time}(\mathscr{G})$ are $\mathrm{poly}(n)$.

If $c - s$ is a constant, then we will suppress the dependence on them and just say that $L \in \mathsf{MIP}^*$.

Remark 4.6. A game $\mathscr{G}$ is *symmetric* if its distribution on questions treats Alice and Bob symmetrically. In this case, we may assume without loss of generality that Alice and Bob's strategies are also symmetric, i.e. that $A^x_a = B^x_a$ for all questions x and answers a. This allows us to represent their measurements by a single set of matrices M (for which $M^x_a = A^x_a = B^x_a$). As a further simplification, by applying Naimark's dilation theorem to Alice and Bob's strategy we can assume that their shared state ψ is pure and their measurements are projectors.

Occasionally, it will be useful to speak of the distribution over measurement outcomes induced by a strategy independently of any particular game. For this, we introduce the notion of a bipartite correlation

Definition 4.7. Given a strategy $\mathcal{S} = (\rho, A, B)$, the bipartite correlation produced by it is the function $P(a, b|x, y) = \mathrm{tr}(A^x_a \otimes B^x_y \cdot \rho)$.

If two strategies produce the same bipartite correlation, they have the same value for any game they are used for. Naimark's theorem (Theorem 4.1) implies for any strategy $\mathcal{S}$, there exists a strategy $\mathcal{S}'$ using only projective measurements that produces the same correlation:

Corollary 4.8 (Naimark's theorem for strategies)**.** *Suppose $\{M_a\}$ and $\{N_b\}$ are two POVMs acting on the A and B factors of a tensor Hilbert space $\mathcal{H}_A \otimes \mathcal{H}_B$, respectively. Then for any Naimark dilation of $\{M_a\}$ given by projectors M'_a and an auxiliary state $|\mathrm{aux}_A\rangle \in \mathcal{H}_{\mathrm{aux}_A}$, and any Naimark dilation of $\{N_b\}$ given by projectors $\mathbb{N}'_b$ and an auxiliary state $|\mathrm{aux}_B\rangle \in \mathcal{H}_{\mathrm{aux}_B}$, it holds that for any bipartite state $|\psi\rangle \in \mathcal{H}_A \otimes \mathcal{H}_b$, the post-measurement state after applying $M \otimes N$ to $|\psi\rangle$ and $M' \otimes N'$ to $|\psi\rangle \otimes |\mathrm{aux}_A\rangle \otimes |\mathrm{aux}_B\rangle$ is the same:*

$$\sqrt{M_a} \otimes \sqrt{N_b} \, |\psi\rangle \langle\psi| \, \sqrt{M_a} \otimes \sqrt{N_b} = \mathrm{tr}_{\mathrm{aux}}[(M'_a \otimes N'_b)(|\psi\rangle \langle\psi| \otimes |\mathrm{aux}_A\rangle \langle \mathrm{aux}_A| \otimes |\mathrm{aux}_B\rangle \langle \mathrm{aux}_B|)(M'_a \otimes N'_b)].$$

Moreover, such dilations exist by Theorem 4.1. As a consequence, M, N and M', N' induce the same joint distribution over outcome probabilities:

$$\langle\psi| \, M_a \otimes N_b \, |\psi\rangle = (\langle\psi| \otimes |\mathrm{aux}_A\rangle \otimes |\mathrm{aux}_B\rangle) M'_a \otimes N'_b (|\psi\rangle \otimes |\mathrm{aux}_A\rangle \otimes |\mathrm{aux}_B\rangle).$$

Proof. The existence of such dilations follows immediately from Theorem 4.1. To deduce the equality of post-measurement states, we apply Equation (4) twice, and use the fact that the partial trace composes, i.e. that $\mathrm{tr}_{\mathrm{aux}}[\cdot] = \mathrm{tr}_{\mathrm{aux}_B}[\mathrm{tr}_{\mathrm{aux}_A}[\cdot]]$.

$$\begin{aligned}
&\mathrm{tr}_{\mathrm{aux}}[(M'_a \otimes N'_b)(|\psi\rangle \langle\psi| \otimes |\mathrm{aux}_A\rangle \langle \mathrm{aux}_A| \otimes |\mathrm{aux}_B\rangle \langle \mathrm{aux}_B|)(M'_a \otimes N'_b)] \\
&\quad = \mathrm{tr}_{\mathrm{aux}_B}[\sqrt{M_a \otimes I}((I \otimes N'_b) \, |\psi\rangle \, |\mathrm{aux}_A\rangle \langle \mathrm{aux}_A| \, \langle\psi| \, (I \otimes N'_b))\sqrt{M_a \otimes I}] \\
&\quad = \sqrt{I \otimes N_b}\sqrt{M_a \otimes I} \, |\psi\rangle \langle\psi| \, \sqrt{M_a \otimes I}\sqrt{I \otimes N_b}.
\end{aligned}$$

□

4.3 Pauli matrices and the EPR state

Over a finite field $\mathbb{F}_q$ with order $q = p^t$ for prime p, the single-qudit Pauli matrices are a set of unitary matrices acting on $\mathbb{C}^q$. Every Pauli matrix can be uniquely written as a product $\omega^a X(x)Z(z)$, where ω is the p-th root $\omega = e^{2\pi/p}$, and $X(x)$ and $Z(z)$ are the matrices

$$X(x) = \sum_{j\in\mathbb{F}_q} |j+x\rangle\langle j|, \qquad Z(z) = \sum_{j\in\mathbb{F}_q} \omega^{\mathrm{tr}[zj]} |j\rangle\langle j|, \tag{6}$$

where the arguments x, z are in $\mathbb{F}_q$, $\mathrm{tr} : \mathbb{F}_q \to \mathbb{F}_p$ is the finite field trace. The set of all Pauli matrices form a group, known as the Pauli group or the Weyl-Heisenberg group. For the most part, in this paper, it will suffice to consider only the group elements of the form $\omega^a X(x)$ ("X-type" Paulis) and $Z(z)$ ("Z-type" Paulis). Elements of the form $\omega^a X(x)Z(z)$ for $x, z \neq 0$ are sometimes called "Y-type".

The eigenvalues of $X(x)$ and $Z(z)$ for all x, z are powers of ω, as can be seen from the facts $X(x)^p = Z(z)^p = I$. Any unitary with this property is known as a *(generalized) observable*. Every generalized observable U induces a projective measurement with p outcomes, corresponding to the p possible eigenvalues of U. As a convenient shorthand, we will refer to performing this projective measurement as "measuring U." In the case of of the X and Z operators, the eigenvectors $|\tau_u^X\rangle$ and $|\tau_u^Z\rangle$ of $X(1)$ and $Z(1)$ are indexed by elements u of $\mathbb{F}_q$, with eigenvalue $\mathrm{tr}[u]$; thus, each eigenvalue occurs with multiplicity q/p. Explicitly, they are given by

$$|\tau_u^X\rangle = \frac{1}{\sqrt{q}} \sum_{v\in\mathbb{F}_q} \omega^{-\mathrm{tr}[uv]} |v\rangle, \qquad |\tau_u^Z\rangle = |u\rangle. \tag{7}$$

We denote the projectors onto these eigenvectors by τ_u^X and τ_u^Z, respectively. These eigenvectors are also the eigenvectors of the remaining $X(x), Z(z)$ observables, as shown by the following fact.

Fact 4.9. *For $W \in \{X, Z\}$, the observables $W(v)$ are related to the projectors τ_u^W by*

$$W(v) = \sum_u \omega^{\mathrm{tr}[u\cdot v]} \tau_u^W \tag{8}$$

$$\tau_u^W = \mathop{\mathbf{E}}_{\boldsymbol{v}} \omega^{-\mathrm{tr}[u\cdot \boldsymbol{v}]} W(\boldsymbol{v}). \tag{9}$$

Proof. We start with Equation (8). For $W = Z$, the relation follows immediately from the definitions. For $W = X$, by calculation we have:

$$\begin{aligned}
\sum_u \omega^{\mathrm{tr}[u\cdot x]} \tau_u^X &= \frac{1}{q} \sum_{u,v,v'} \omega^{\mathrm{tr}[u\cdot x]} \omega^{\mathrm{tr}[u(v-v')]} |v'\rangle\langle v| \\
&= \sum_{v,v'} \mathop{\mathbf{E}}_{\boldsymbol{u}} \omega^{\mathrm{tr}[\boldsymbol{u}\cdot(x+v-v')]} |v'\rangle\langle v| \\
&= \sum_v |v+x\rangle\langle v| \\
&= X(x),
\end{aligned}$$

where we have applied Fact 3.1 in passing from the second to the third line. Now we show Equation (9):

$$\mathop{\mathbf{E}}_{\boldsymbol{v}} \omega^{-\mathrm{tr}[u\cdot\boldsymbol{v}]} W(\boldsymbol{v}) = \mathop{\mathbf{E}}_{\boldsymbol{v}} \sum_a \omega^{-\mathrm{tr}[u\cdot\boldsymbol{v}]} \omega^{\mathrm{tr}[\boldsymbol{v}\cdot a]} \tau_a^W = \mathop{\mathbf{E}}_{\boldsymbol{v}} \sum_a \omega^{\mathrm{tr}[(a-u)\cdot\boldsymbol{v}]} \tau_a^W = \tau_u^W,$$

where we first applied Equation (8) in the first equality, and then used Fact 3.1 to perform the expectation over $\boldsymbol{v}$. □

The Pauli matrices obey the commutation relation

$$X(x)Z(z) = \omega^{-\mathrm{tr}[xz]} Z(z)X(x). \tag{10}$$

This follows directly from Equation (6). It follows from this that all of the Pauli matrices (including the Y-type matrices) are generalized observables.

The maximally entangled state, or EPR state, over qudits of dimension q is the state

$$|\mathrm{EPR}_q\rangle = \frac{1}{\sqrt{q}} \sum_{u \in \mathbb{F}_q} |u\rangle \otimes |u\rangle .$$

We will write $|\mathrm{EPR}_q^n\rangle$ for $|\mathrm{EPR}_q\rangle^{\otimes n}$. This state obeys the stabilizer relations

$$X(x) \otimes X(x) |\mathrm{EPR}_q\rangle = Z(z) \otimes Z(-z) |\mathrm{EPR}_q\rangle = |\mathrm{EPR}_q\rangle \tag{11}$$

$$\tau_u^X \otimes I |\mathrm{EPR}_q\rangle = I \otimes \tau_{-u}^X |\mathrm{EPR}_q\rangle \tag{12}$$

$$\tau_u^Z \otimes I |\mathrm{EPR}_q\rangle = I \otimes \tau_u^Z |\mathrm{EPR}_q\rangle . \tag{13}$$

Relations (12) and (13) imply that measuring $X(x)$ on both halves of an EPR state will yield two outcomes a, b satisfying $a = -b$, and measuring $Z(z)$ on both halves will yield two outcomes a, b that are equal.

In the important special case of finite fields with characteristic 2 (i.e. $\mathbb{F}_q$ for even q), $u = -u$ for all $u \in \mathbb{F}_q$, and thus measuring any of the X and Z operators on both sides of the state will always yield the same outcome.

4.4 State dependent distances

In this section, we introduce two state-dependent distances. To motivate them, we first define the consistency game, perhaps the simplest nontrivial two-player game.

Definition 4.10. The *consistency game with question x*, denoted $\mathscr{G}_{\mathrm{con}}(x)$ is defined as follows. The question x is given to Alice and Bob, who respond with answers $\boldsymbol{a}$ and $\boldsymbol{a}'$, respectively. The verifier accepts if $\boldsymbol{a} = \boldsymbol{a}'$.

We will typically play the consistency game when $\boldsymbol{x}$, rather than being a fixed question, is drawn from some distribution. Our first state-dependent distance quantifies the players' success probability in this case.

Definition 4.11. Let $\{A_a^x\}$ and $\{B_a^x\}$ be sets of matrices in $\mathcal{L}(\mathcal{H}_A)$ and $\mathcal{L}(\mathcal{H}_B)$, respectively. Let $\mathcal{D}$ be a distribution on questions x and $|\psi\rangle$ be a state in $\mathcal{H}_A \otimes \mathcal{H}_B$. Consider the game in which the verifier selects $\boldsymbol{x} \sim \mathcal{D}$ and then plays $\mathscr{G}_{\mathrm{con}}(\boldsymbol{x})$. We say that

$$A_a^x \otimes I_{\mathsf{Bob}} \simeq_\delta I_{\mathsf{Alice}} \otimes B_a^x$$

on state $|\psi\rangle$ and distribution $\mathcal{D}$ if Alice and Bob win with probability $1 - O(\delta)$ using the measurements A and B, respectively.

We will sometimes leave the state or distribution unspecified, as they are often clear from context. This distance has a clear operational interpretation. Our second state-dependent distance, defined next, is more analytic.

Definition 4.12. Let $\{Q_a^x\}$ and $\{R_a^x\}$ be sets of matrices in $\mathcal{L}(\mathcal{H})$. Let $\mathcal{D}$ be a distribution on the variables x and $|\psi\rangle$ be a state in $\mathcal{H}$. Then we say that $Q_a^x \approx_\delta R_a^x$ *on state* $|\psi\rangle$ *and distribution* $\mathcal{D}$ if

$$\mathop{\mathbf{E}}_{\boldsymbol{x}\sim\mathcal{D}} \sum_a \|(Q_a^{\boldsymbol{x}} - R_a^{\boldsymbol{x}})\,|\psi\rangle\,\|^2 = O(\delta).$$

As above, we will sometimes leave the state or distribution unspecified when clear from context. This is sometimes referred to as *the* state-dependence distance, whereas our first distance measure is often referred to as the "consistency". A typical setting of parameters is $\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_B$, $Q_a^x := A_a^x \otimes I_{\mathsf{Bob}}$, and $R_a^x := I_{\mathsf{Alice}} \otimes B_a^x$. In this case, we have the following relationship between the two state-dependent distances.

Fact 4.13. *Let* $\{A_a^x\}$ *and* $\{B_a^x\}$ *be POVM measurements. The following two facts hold.*

1. *If* $A_a^x \otimes I_{\mathsf{Bob}} \simeq_\delta I_{\mathsf{Alice}} \otimes B_a^x$ *then* $A_a^x \otimes I_{\mathsf{Bob}} \approx_\delta I_{\mathsf{Alice}} \otimes B_a^x$.
2. *If* $A_a^x \otimes I_{\mathsf{Bob}} \approx_\delta I_{\mathsf{Alice}} \otimes B_a^x$ and $\{A_a^x\}$ *and* $\{B_a^x\}$ *are projective measurements, then* $A_a^x \otimes I_{\mathsf{Bob}} \simeq_\delta I_{\mathsf{Alice}} \otimes B_a^x$.

Proof. Suppose that $A_a^x \otimes I_{\mathsf{Bob}} \simeq_\delta I_{\mathsf{Alice}} \otimes B_a^x$. This is equivalent to the statement

$$\mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \langle\psi|\, A_a^{\boldsymbol{x}} \otimes B_a^{\boldsymbol{x}}\, |\psi\rangle \geq 1 - O(\delta). \tag{14}$$

As a result, using the fact that A and B are POVMs,

$$\begin{aligned}
\mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \|(A_a^{\boldsymbol{x}} \otimes I - I \otimes B_a^{\boldsymbol{x}})\,|\psi\rangle\,\|^2 &= \mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \langle\psi|\,((A_a^{\boldsymbol{x}})^2 \otimes I + I \otimes (B_a^{\boldsymbol{x}})^2 - 2A_a^{\boldsymbol{x}} \otimes B_a^{\boldsymbol{x}})\,|\psi\rangle \\
&\leq \mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \langle\psi|\,(A_a^{\boldsymbol{x}} \otimes I + I \otimes B_a^{\boldsymbol{x}} - 2A_a^{\boldsymbol{x}} \otimes B_a^{\boldsymbol{x}})\,|\psi\rangle \qquad (15) \\
&= 2 - 2\mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \langle\psi|\, A_a^{\boldsymbol{x}} \otimes B_a^{\boldsymbol{x}}\, |\psi\rangle\,.
\end{aligned}$$

By Equation (14), this is $O(\delta)$. As a result, $A_a^x \otimes I_{\mathsf{Bob}} \approx_\delta I_{\mathsf{Alice}} \otimes B_a^x$. The reverse statement holds when A and B are projective measurements because Equation (15) is an equality in this case. □

The following fact shows that we can derive a weaker converse in the case when only one of A or B is projective.

Fact 4.14. *Suppose* $\{A_a^x\}$ *and* $\{B_a^x\}$ *are two measurements such that* $A_a^x \otimes I_{\mathsf{Bob}} \approx_\delta I_{\mathsf{Alice}} \otimes B_a^x$. *Suppose further that either* A *or* B *is a projective measurement (and the other is a POVM measurement). Then* $A_a^x \otimes I_{\mathsf{Bob}} \simeq_{\delta^{1/2}} I_{\mathsf{Alice}} \otimes B_a^x$.

Proof. Our goal is to upper bound the expression

$$1 - \mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \langle\psi|\, A_a^{\boldsymbol{x}} \otimes B_a^{\boldsymbol{x}}\, |\psi\rangle\,. \tag{16}$$

We begin by rewriting the number 1. Here we use the fact that because A is projective, $(A_a^x)^2 = A_a^x$, and so $\sum_a (A_a^x)^2 = I$. This gives us:

$$\begin{aligned}
(16) &= \mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \langle\psi|\,(A_a^{\boldsymbol{x}})^2 \otimes I_{\mathsf{Bob}}\,|\psi\rangle - \mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \langle\psi|\, A_a^{\boldsymbol{x}} \otimes B_a^{\boldsymbol{x}}\,|\psi\rangle \\
&= \mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \langle\psi|\,(A_a^{\boldsymbol{x}} \otimes I_{\mathsf{Bob}}) \cdot (A_a^{\boldsymbol{x}} \otimes I_{\mathsf{Bob}} - I_{\mathsf{Alice}} \otimes B_a^{\boldsymbol{x}})\,|\psi\rangle \\
&\leq \mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \|A_a^{\boldsymbol{x}} \otimes I_{\mathsf{Bob}}\,|\psi\rangle\,\| \cdot \|(A_a^{\boldsymbol{x}} \otimes I_{\mathsf{Bob}} - I_{\mathsf{Alice}} \otimes B_a^{\boldsymbol{x}})\,|\psi\rangle\,\| \qquad (17)
\end{aligned}$$

Now we apply Cauchy-Schwarz and then Jensen's inequality:

$$(17) \leq \sqrt{\mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \| A_a^{\boldsymbol{x}} \otimes I_{\mathsf{Bob}} |\psi\rangle \|^2 \cdot \sum_a \| (A_a^{\boldsymbol{x}} \otimes I_{\mathsf{Bob}} - I_{\mathsf{Alice}} \otimes B_a^{\boldsymbol{x}}) |\psi\rangle \|^2}$$

The first of these terms we bound by 1, and the second is $O(\delta)$ by assumption. □

Remark 4.15. We note that the requirement in Fact 4.14 that one of the two measurements be projective is necessary. Consider the measurements $\{A_a^x\}$ and $\{B_a^x\}$ with m separate outcomes a in which $A_a^x = B_a^x = I/m$ for all x and a. Then $A_a^x \otimes I_{\mathsf{Bob}} \approx_0 I_{\mathsf{Alice}} \otimes B_a^x$, but as $m \to \infty$,

$$1 - \mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \langle\psi| A_a^{\boldsymbol{x}} \otimes B_a^{\boldsymbol{x}} |\psi\rangle \to 1.$$

Thus, when one of the measurements is projective, the "$\approx_\delta$" distance is roughly equivalent to the "$\simeq_\delta$" distance, up to a polynomial factor (which we can tolerate losing in our proofs). More generally, however, the "$\approx_\delta$" distance can be viewed as a weakening of the "$\simeq_\delta$" distance. In spite of this, we will spend much of the paper dealing with the "$\approx_\delta$" distance, as it is easier to manipulate but still strong enough to reach our desired consequences. (See [Vid11, Section 2.3.1] for a further defense of this distance.) We note that even when $|\psi\rangle$ is a bipartite state, the "$\approx_\delta$" distance is defined for matrices Q and R which are not necessarily tensor products over the bipartition. Such matrices will often be useful to pass through during intermediate steps of our proofs.

A common use case for these distances is when the verifier (i) samples a pair of questions $\boldsymbol{x} = (\boldsymbol{x}_0, \boldsymbol{x}_1)$, (ii) hands $\boldsymbol{x}_0$ to Alice and $\boldsymbol{x}_1$ second to Bob, (iii) receives their answers $\boldsymbol{a}_0$ and $\boldsymbol{a}_1$ and (iv) accepts if $f(\boldsymbol{x}, \boldsymbol{a}_0) = g(\boldsymbol{x}, \boldsymbol{a}_1)$ for some functions f and g. Write $\{A_{a_0}^{x_0}\}_{a_0}$ and $\{B_{a_1}^{x_1}\}_{a_1}$ for Alice and Bob's measurements, respectively. We can view these as measurements which receive the pair $(\boldsymbol{x}_0, \boldsymbol{x}_1)$ and simply ignore one coordinate. Suppose the verifier accepts with probability $1 - \delta$. Using Notation 4.3, we can view this as performing the consistency game between the measurements $A_{[f(x,a_0)=b]}^{x_0}$ and $B_{[f(x,a_1)=b]}^{x_1}$. Hence, we can derive the following two facts:

$$A_{[f(x,a_0)=b]}^{x_0} \otimes I_{\mathsf{Bob}} \simeq_\delta I_{\mathsf{Alice}} \otimes B_{[f(x,a_1)=b]}^{x_1}, \quad \text{and} \quad A_{[f(x,a_0)=b]}^{x_0} \otimes I_{\mathsf{Bob}} \approx_\delta I_{\mathsf{Alice}} \otimes B_{[f(x,a_1)=b]}^{x_1}.$$

This generic format will be the most common use of these notations.

Remark 4.15 highlights the importance of *projective* measurements when dealing with the "$\approx_\delta$" distance. This, and several other key facts about the "$\approx_\delta$" distance, are true only for projective measurements. As a result, we will sometimes apply Naimark's theorem (Theorem 4.1) during our proofs to "round" POVM measurements into projective measurements. However, there is a subtlety in doing so, namely that because Naimark's theorem preserves measurement outcomes, any "$\simeq_\delta$" statements we have derived about our measurement operators will remain true, but Naimark's theorem is *not* guaranteed to preserve "$\approx_\delta$" statements. In this work, we will be able to dispense with this subtlety and assume all "$\approx_\delta$" statements *are* preserved, because *all "$\approx_\delta$" statements in our proofs will be derived from "$\simeq_\delta$" statements*, and so they will remain true after performing Naimark's theorem, since we could simply rederive them.

4.5 Miscellaneous properties of the state-dependent distances

In this section, we record some facts about the "$\approx_\delta$" notation which we will use repeatedly throughout the paper. A good rule of thumb is that everything one expects to be true about the "$\approx_\delta$" notation actually *is* true, except for those things which are not. As a result, we will be overly pedantic in this section in order to call attention to these cases.

4.5.1 Simple state-dependent distance facts

Fact 4.16. *For two vectors* $|\psi_1\rangle, |\psi_2\rangle$, $\| |\psi_1\rangle + |\psi_2\rangle \|^2 \leq 2\| |\psi_1\rangle \|^2 + 2\| |\psi_2\rangle \|^2$.

Fact 4.17. *Let* $\{A_a\}$ *be a measurement. Then*

$$\sum_a \|A_a |\psi\rangle\|^2 \leq \| |\psi\rangle \|^2.$$

Proof. If $\{A_a\}$ is a measurement, then

$$\sum_a \|A_a |\psi\rangle\|^2 = \sum_a \mathrm{tr}(A_a A_a |\psi\rangle\langle\psi|) \leq \mathrm{tr}(I \cdot |\psi\rangle\langle\psi|) = \| |\psi\rangle \|^2. \qquad \square$$

Fact 4.18. *Let* $\{A_a\}$, $\{B_a\}$ *be measurements. Then for any state* $|\psi\rangle$,

$$\sum_a \|(A_a - B_a) |\psi\rangle\|^2 \leq 4 \cdot \| |\psi\rangle \|^2.$$

Proof. By Fact 4.16,

$$\sum_a \|(A_a - B_a) |\psi\rangle\|^2 \leq 2\sum_a \|A_a |\psi\rangle\|^2 + 2\sum_a \|B_a |\psi\rangle\|^2.$$

The fact now follows from Fact 4.17. $\square$

Fact 4.19. *Let* $\{A_a^x\}$ *and* $\{B_a^x\}$ *be POVM measurements. Then* $A_a^x \approx_1 B_a^x$.

Fact 4.20. *Let* $\{A_a^x\}$ *and* $\{B_a^x\}$ *be matrices. Let* $\{C_b^y\}$ *be matrices such that* $\sum_b (C_b^y)^\dagger C_b^y \leq I$ *for all* y. *(This includes the case when* $\{C_b^y\}$ *form projective or POVM measurements.) Then*

$$A_a^x \approx_\delta B_a^x \text{ implies } C_b^y A_a^x \approx_\delta C_b^y B_a^x.$$

Proof. Fix questions x, y and answers a. Because of our property on $\{C_b^y\}_b$,

$$\begin{aligned}\sum_b \|(C_b^y A_a^x - C_b^y B_a^x) |\psi\rangle\|^2 &= \sum_b \langle\psi| (A_a^x - B_a^x)^\dagger (C_b^y)^\dagger (C_b^y)(A_a^x - B_a^x) |\psi\rangle \\ &\leq \langle\psi| (A_a^x - B_a^x)^\dagger (A_a^x - B_a^x) |\psi\rangle = \|(A_a^x - B_a^x) |\psi\rangle\|^2\end{aligned}$$

We can therefore derive our desired conclusion:

$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}} \sum_{a,b} \|(C_b^{\boldsymbol{y}} A_a^{\boldsymbol{x}} - C_b^{\boldsymbol{y}} B_a^{\boldsymbol{x}}) |\psi\rangle\|^2 \leq \mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}} \sum_a \|(A_a^{\boldsymbol{x}} - B_a^{\boldsymbol{x}}) |\psi\rangle\|^2 = \delta. \qquad \square$$

Fact 4.21. *Let* $\mathcal{D}, \mathcal{D}'$ *be two distributions such that* $d_{\mathrm{TV}}(\mathcal{D}, \mathcal{D}') \leq \epsilon$. *Let* $\{A_a^x\}$ *and* $\{B_a^x\}$ *be measurements, and suppose* $A_a^x \approx_\delta B_a^x$ *with respect to* $\mathcal{D}$. *Then* $A_a^x \approx_{\delta+\epsilon} B_a^x$ *with respect to* $\mathcal{D}'$.

Proof. By the definition of total variation distance, for any set of numbers $\{\nu_x\}$ satisfying $0 \leq \nu_x \leq c$, the expectations under the two distributions are similar:

$$|\mathop{\mathbf{E}}_{\boldsymbol{x}\sim\mathcal{D}}[\nu_{\boldsymbol{x}}] - \mathop{\mathbf{E}}_{\boldsymbol{x}\sim\mathcal{D}'}[\nu_{\boldsymbol{x}}]| \leq c \cdot \epsilon.$$

We will take for our numbers $\nu_x = \sum_a \|(A_a^x - B_a^x) |\psi\rangle\|^2$, which is always less than 4 by Fact 4.18. As a result,

$$\mathop{\mathbf{E}}_{\boldsymbol{x}\sim\mathcal{D}'} \nu_{\boldsymbol{x}} \leq \mathop{\mathbf{E}}_{\boldsymbol{x}\sim\mathcal{D}} \nu_{\boldsymbol{x}} + 4\epsilon \leq \delta + 4\epsilon.$$

This is $O(\delta + \epsilon)$, which proves the fact. $\square$

Fact 4.22. *Suppose $A_a^x \approx_\delta B_a^x$ on state $|\psi\rangle$, and suppose $\| |\psi\rangle - |\overline{\psi}\rangle \|^2 \leq \epsilon$. Then $A_a^x \approx_{\delta+\epsilon} B_a^x$ on state $|\overline{\psi}\rangle$.*

Proof. Applying Fact 4.16 to $(A_a^x - B_a^x)|\psi\rangle$ and $(A_a^x - B_a^x)(|\overline{\psi}\rangle - |\psi\rangle)$,

$$\mathop{\mathbf{E}}_{\boldsymbol{x}\sim\mathcal{D}} \sum_a \|(A_a^{\boldsymbol{x}} - B_a^{\boldsymbol{x}})|\overline{\psi}\rangle\|^2 \leq \mathop{\mathbf{E}}_{\boldsymbol{x}\sim\mathcal{D}} \sum_a \|(A_a^{\boldsymbol{x}} - B_a^{\boldsymbol{x}})|\psi\rangle\|^2 + \mathop{\mathbf{E}}_{\boldsymbol{x}\sim\mathcal{D}} \sum_a \|(A_a^{\boldsymbol{x}} - B_a^{\boldsymbol{x}})(|\overline{\psi}\rangle - |\psi\rangle)\|^2.$$

The first of these is bounded by $O(\delta)$ by the assumption, and the second of these is bounded by 4ϵ by the assumption and Fact 4.18. □

Fact 4.23. *Suppose $A_a^{x_1} \approx_\delta B_a^{x_1}$ with respect to a distribution $\mathcal{D}_{\text{margin}}$ on x_1. Let $\mathcal{D}$ be a distribution on (x_1, x_2) such that the marginal distribution on x_1 is $\mathcal{D}_{\text{margin}}$. Then $A_a^{x_1} \approx_\delta B_a^{x_1}$ with respect to $\mathcal{D}$.*

Proof. This is a simple calculation involving Notation 4.3. □

Fact 4.24. *Let k be a constant, and consider distributions over questions $\mathcal{D}_1, \ldots, \mathcal{D}_k$. Let $\mathcal{D}$ be a mixture of these distributions, meaning that there is a probability distribution $p = (p_1, \ldots, p_k)$ such that a draw from $\mathcal{D}$ can be simulated as follows: draw $\boldsymbol{i} \sim p$ and output $\boldsymbol{x}$ sampled from $\mathcal{D}_{\boldsymbol{i}}$. Suppose $A_a^x \approx_\delta B_a^x$ with respect to $\mathcal{D}_i$, for all $i \in [k]$. Then $A_a^x \approx_\delta B_a^x$ with respect to $\mathcal{D}$.*

Proof. By definition, for each $i \in [k]$ there is some constant C_i such that

$$\mathop{\mathbf{E}}_{\boldsymbol{x}\sim\mathcal{D}_i} \sum_a \|(A_a^{\boldsymbol{x}} - B_b^{\boldsymbol{x}})|\psi\rangle\|^2 \leq C_k \cdot \delta.$$

Then we can bound the mixture with

$$\mathop{\mathbf{E}}_{\boldsymbol{x}\sim\mathcal{D}} \sum_a \|(A_a^{\boldsymbol{x}} - B_b^{\boldsymbol{x}})|\psi\rangle\|^2 = \mathop{\mathbf{E}}_{\boldsymbol{i}\sim p} \mathop{\mathbf{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{i}}} \sum_a \|(A_a^{\boldsymbol{x}} - B_b^{\boldsymbol{x}})|\psi\rangle\|^2 \leq \mathop{\mathbf{E}}_{\boldsymbol{i}\sim p} C_{\boldsymbol{i}} \cdot \delta \leq \max_i\{C_i\} \cdot \delta = O(\delta). \quad \square$$

Fact 4.25. *Suppose $\{A_a^x\}$ is a projective measurement and $\{B_a^x\}$ is a set of matrices such that each B_a^x is positive semidefinite and $\sum_a B_a^x \preceq I$. Define C_a^x such that for each x, there exists an a such that $C_a^x := B_a^x + (I - \sum_{a'} B_{a'}^x)$ and for all other $a' \neq a$, $C_{a'}^x := B_{a'}^x$. Thus, C^x is a POVM for each x. If $A_a^x \approx_\epsilon B_a^x$ then $A_a^x \approx_{\epsilon^{1/2}} C_a^x$.*

Proof. By Fact 4.16,

$$\mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \|(A_a^{\boldsymbol{x}} - C_a^{\boldsymbol{x}})|\psi\rangle\|^2 \leq 2 \mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \|(A_a^{\boldsymbol{x}} - B_a^{\boldsymbol{x}})|\psi\rangle\|^2 + 2 \mathop{\mathbf{E}}_{\boldsymbol{x}} \|(I - \sum_a B_a^{\boldsymbol{x}})|\psi\rangle\|^2.$$

The first of these terms we can bound by $O(\epsilon)$. As for the second,

$$\begin{aligned}\mathop{\mathbf{E}}_{\boldsymbol{x}} \|(I - \sum_a B_a^{\boldsymbol{x}})|\psi\rangle\|^2 = \mathop{\mathbf{E}}_{\boldsymbol{x}} \langle\psi|(I - \sum_a B_a^{\boldsymbol{x}})^2|\psi\rangle &\leq \mathop{\mathbf{E}}_{\boldsymbol{x}} \langle\psi|(I - \sum_a B_a^{\boldsymbol{x}})|\psi\rangle \\ &= 1 - \mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \langle\psi| B_a^{\boldsymbol{x}} |\psi\rangle \leq 1 - \mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \langle\psi|(B_a^{\boldsymbol{x}})^2|\psi\rangle.\end{aligned}$$

Now, we write $1 = \mathbf{E}_{\boldsymbol{x}} \sum_a \langle\psi|(A_a^{\boldsymbol{x}})^2|\psi\rangle$, which holds because A is a projective measurement. We bound the result as follows.

$$\begin{aligned}\mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \langle\psi|((A_a^{\boldsymbol{x}})^2 - (B_a^{\boldsymbol{x}})^2)|\psi\rangle &= \Re\left(\mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \langle\psi|(A_a^{\boldsymbol{x}} + B_a^{\boldsymbol{x}})(A_a^{\boldsymbol{x}} - B_a^{\boldsymbol{x}})|\psi\rangle\right) \\ &\leq \mathop{\mathbf{E}}_{\boldsymbol{x}} \sqrt{\sum_a \|(A_a^{\boldsymbol{x}} + B_a^{\boldsymbol{x}})|\psi\rangle\|^2} \cdot \sqrt{\sum_a \|(A_a^{\boldsymbol{x}} - B_a^{\boldsymbol{x}})|\psi\rangle\|^2}.\end{aligned}$$

For each $\boldsymbol{x}$, we can bound the first square root by $O(1)$ due to Fact 4.16 and Fact 4.17. Having done so, we can move the expectation into the second square root by Jensen's inequality. The result is $O(\epsilon^{1/2})$ by assumption. This proves the fact. □

4.5.2 Data processing

In this section, we show a simple data processing inequality for the "$\simeq_\delta$" distance. We also observe that one does *not* hold for the "$\approx_\delta$" distance.

Fact 4.26. *Suppose that* $A^x_a \otimes I_{\mathsf{Bob}} \simeq_\delta I_{\mathsf{Alice}} \otimes B^x_a$. *Then* $A^x_{[f(a)=b]} \otimes I_{\mathsf{Bob}} \simeq_\delta I_{\mathsf{Alice}} \otimes B^x_{[f(a)=b]}$.

Proof. Given question $\boldsymbol{x}$, if Alice and Bob return $\boldsymbol{a}$ and $\boldsymbol{a}'$ in which $\boldsymbol{a} = \boldsymbol{a}'$, then $f(\boldsymbol{a}) = f(\boldsymbol{a}')$. As a result, applying f to their answers cannot decrease the probability they agree. □

Remark 4.27. We note that the same fact is *not* true for the "$\approx_\delta$" distance. Consider answers of the form $a = (b, i)$, where $b \in \{0, 1\}$ and $i \in [m]$. Suppose $A^x_{b,i} = I/(2m)$ for all a, whereas $B^x_{0,i} = I/m$ and $B^x_{1,i} = 0$ for all i. Consider the function $f(b, i) = b$. It can be checked that in this case, $A^x_a \otimes I_{\mathsf{Bob}} \approx_{1/2m} I_{\mathsf{Alice}} \otimes B^x_a$ but $A^x_{[f(a)=b]} \otimes I_{\mathsf{Bob}} \approx_{1/2} I_{\mathsf{Alice}} \otimes B^x_{[f(a)=b]}$.

4.5.3 Triangle inequalities

In this section, we give two triangle inequalities. Our first is for the state-dependent distance.

Fact 4.28 (Triangle inequality)**.** *Suppose* $A^x_a \approx_\delta B^x_a$ *and* $B^x_a \approx_\epsilon C^x_a$. *Then* $A^x_a \approx_{\delta+\epsilon} C^x_a$.

Proof. Applying Fact 4.16 to $(A^x_a - B^x_a)\,|\psi\rangle$ and $(B^x_a - C^x_a)\,|\psi\rangle$,

$$\begin{aligned}\mathop{\mathbf{E}}_{\boldsymbol{x}\sim\mathcal{D}} \sum_a \|(A^{\boldsymbol{x}}_a - C^{\boldsymbol{x}}_a)\,|\psi\rangle\,\|^2 &\leq 2 \mathop{\mathbf{E}}_{\boldsymbol{x}\sim\mathcal{D}} \sum_a \|(A^{\boldsymbol{x}}_a - B^{\boldsymbol{x}}_a)\,|\psi\rangle\,\|^2 + 2 \mathop{\mathbf{E}}_{\boldsymbol{x}\sim\mathcal{D}} \sum_a \|(B^{\boldsymbol{x}}_a - C^{\boldsymbol{x}}_a)\,|\psi\rangle\,\|^2 \\ &\leq 2(\delta + \epsilon).\end{aligned}$$

□

Note that this does *not* show that if

$$A^x_a \otimes I_{\mathsf{Bob}} \approx_\delta I_{\mathsf{Alice}} \otimes B^x_a \quad \text{and} \quad B^x_a \otimes I_{\mathsf{Bob}} \approx_\delta I_{\mathsf{Alice}} \otimes C^x_a$$

then $A^x_a \otimes I_{\mathsf{Bob}} \approx_\delta I_{\mathsf{Alice}} \otimes C^x_a$. This would only follow if, for example, we also knew that $D^x_a \otimes I_{\mathsf{Bob}} \approx_\delta I_{\mathsf{Alice}} \otimes D^x_a$, for D equal to one of A, B, or C. We do, however, always have the following triangle-like inequalities.

Fact 4.29 (Triangle-like inequalities)**.** *The following two facts are true.*

1. *Suppose* $A^x_a \otimes I_{\mathsf{Bob}} \simeq_\delta I_{\mathsf{Alice}} \otimes B^x_a$, $B^x_a \otimes I_{\mathsf{Bob}} \simeq_\delta I_{\mathsf{Alice}} \otimes C^x_a$, *and* $C^x_a \otimes I_{\mathsf{Bob}} \simeq_\delta I_{\mathsf{Alice}} \otimes D^x_a$. *Then* $A^x_a \otimes I_{\mathsf{Bob}} \simeq_\delta I_{\mathsf{Alice}} \otimes D^x_a$.
2. *Suppose* $A^x_a \otimes I_{\mathsf{Bob}} \approx_\delta I_{\mathsf{Alice}} \otimes B^x_a$, $B^x_a \otimes I_{\mathsf{Bob}} \approx_\delta I_{\mathsf{Alice}} \otimes C^x_a$, *and* $C^x_a \otimes I_{\mathsf{Bob}} \approx_\delta I_{\mathsf{Alice}} \otimes D^x_a$. *Then* $A^x_a \otimes I_{\mathsf{Bob}} \approx_\delta I_{\mathsf{Alice}} \otimes D^x_a$.

Before proving this, we need the following fact from linear algebra.

Fact 4.30. *Suppose* $0 \preceq A, B, C, D \preceq I$. *Then*

$$1 - \langle\psi|\,A \otimes D\,|\psi\rangle \leq (1 - \langle\psi|\,A \otimes B\,|\psi\rangle) + (1 - \langle\psi|\,B \otimes C\,|\psi\rangle) + (1 - \langle\psi|\,C \otimes D\,|\psi\rangle).$$

Proof. Rearranging, we want to show that

$$\langle\psi| A \otimes B |\psi\rangle + \langle\psi| B \otimes C |\psi\rangle + \langle\psi| C \otimes D |\psi\rangle - \langle\psi| A \otimes D |\psi\rangle \leq 2.$$

Or, equivalently

$$\mathrm{tr}(|\psi\rangle\langle\psi| \cdot (A \otimes B + B \otimes C + C \otimes D - A \otimes D)) \leq 2.$$

The left-hand side is at most the maximum eigenvalue of $A \otimes B + B \otimes C + C \otimes D - A \otimes D$. To bound this maximum eigenvalue, we note that $A \otimes B \preceq A \otimes I$, $B \otimes C \preceq I \otimes I$, and $C \otimes D \preceq I \otimes D$. As a result,

$$A \otimes B + B \otimes C + C \otimes D - A \otimes D \preceq A \otimes I + I \otimes I + I \otimes D - A \otimes D.$$

Next, $I \otimes D - A \otimes D = (I - A) \otimes D \preceq (I - A) \otimes I$ because $A \preceq I$. Thus,

$$A \otimes I + I \otimes I + I \otimes D - A \otimes D \preceq A \otimes I + I \otimes I + (I - A) \otimes I = 2 \cdot I \otimes I.$$

But the maximum eigenvalue of this is 2. □

Now we prove Fact 4.29.

Proof of Fact 4.29. The second fact follows from several applications of Fact 4.28. As for the first fact, we can write the consistency as

$$\mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a (1 - \langle\psi| A_a^{\boldsymbol{x}} \otimes D_a^{\boldsymbol{x}} |\psi\rangle)$$

Applying Fact 4.30, this is at most

$$\mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a (1 - \langle\psi| A_a^{\boldsymbol{x}} \otimes B_a^{\boldsymbol{x}} |\psi\rangle) + (1 - \langle\psi| B_a^{\boldsymbol{x}} \otimes C_a^{\boldsymbol{x}} |\psi\rangle) + (1 - \langle\psi| C_a^{\boldsymbol{x}} \otimes D_a^{\boldsymbol{x}} |\psi\rangle)$$

Averaging over questions and summing over answers, each of these terms is at most δ, by assumption. □

4.5.4 Close strategies have close game values

In this section, we will show that two strategies which are close in state-dependent distance are also close in value for any game $\mathscr{G}$. We note crucially that one of the two strategies must be *projective* to apply this fact.

Fact 4.31. *Let $\mathcal{D}$ be a distribution on questions x, and for each x let $\mathrm{acc}(x)$ be a set of "accepting" answers. Given a state ψ and a strategy $\{A_a^x\}$ define*

$$\mathrm{val}(A) = \mathop{\mathbf{E}}_{\boldsymbol{x} \sim \mathcal{D}} \sum_{a \in \mathrm{acc}(x)} \langle\psi| A_a^{\boldsymbol{x}} |\psi\rangle .$$

Suppose $\{A_a^x\}$ and $\{B_a^x\}$ are two strategies such that $A_a^x \approx_\delta B_a^x$ on state ψ and distribution $\mathcal{D}$. Suppose further that either A or B is a projective measurement (and the other is a POVM measurement). Then

$$\mathrm{val}(A) - O(\delta^{1/2}) \leq \mathrm{val}(B) \leq \mathrm{val}(A) + O(\delta^{1/2}).$$

Proof. Assume without loss of generality that A is a projective measurement and B is a POVM measurement. We will prove the fact by showing the following stronger statement: for each x, let $S(x)$ be any set of answers a, and define

$$\mathrm{val}(A, S) := \mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_{a \in S(x)} \langle \psi | A_a^{\boldsymbol{x}} | \psi \rangle .$$

Then $\mathrm{val}(A, S) \leq \mathrm{val}(B, S) + O(\delta^{1/2})$. By taking $S(x) := \mathrm{acc}(x)$ this implies the lower bound $\mathrm{val}(A) - O(\delta^{1/2}) \leq \mathrm{val}(B)$, and by taking $S(x) := \mathrm{rej}(x)$, defined to be the set of answers *not* in $\mathrm{acc}(x)$, then this implies the upper bound $\mathrm{val}(B) \leq \mathrm{val}(A) + O(\delta^{1/2})$.

If we write $|u_a^x\rangle = A_a^x |\psi\rangle$ and $|w_a^x\rangle = (B_a^x - A_a^x)|\psi\rangle$, then

$$\| B_a^x |\psi\rangle \|^2 = \| |u_a^x\rangle + |v_a^x\rangle \|^2 = \| |u_a^x\rangle \|^2 + \| |w_a^x\rangle \|^2 + \langle u_a^x \mid w_a^x \rangle + \langle w_a^x \mid u_a^x \rangle .$$

By definition,

$$\begin{aligned}
\mathrm{val}(B) &= \mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_{a \in S(\boldsymbol{x})} \langle \psi | B_a^{\boldsymbol{x}} | \psi \rangle \\
&\geq \mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_{a \in S(\boldsymbol{x})} \langle \psi | (B_a^{\boldsymbol{x}})^2 | \psi \rangle && \text{(because } B \text{ is a POVM)} \\
&= \mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_{a \in S(\boldsymbol{x})} \| B_a^{\boldsymbol{x}} |\psi\rangle \|^2 \\
&= \mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_{a \in S(\boldsymbol{x})} \| |u_a^{\boldsymbol{x}}\rangle \|^2 + \| |w_a^{\boldsymbol{x}}\rangle \|^2 + \langle u_a^{\boldsymbol{x}} \mid w_a^{\boldsymbol{x}} \rangle + \langle w_a^{\boldsymbol{x}} \mid u_a^{\boldsymbol{x}} \rangle .
\end{aligned}$$

Averaging over questions and summing over answers, the first term is exactly $\mathrm{val}(A)$ because A is projective. The second term is always nonnegative, so we lower bound it by zero. As for the last two terms,

$$\langle u_a^x \mid w_a^x \rangle + \langle w_a^x \mid u_a^x \rangle \geq -2 \cdot | \langle u_a^x \mid w_a^x \rangle | \geq -2 \cdot \|u_a^x\| \cdot \|w_a^x\|. \tag{18}$$

Applying Cauchy-Schwarz, Jensen's inequality, and Fact 4.17,

$$\mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_{a \in S(\boldsymbol{x})} \|u_a^{\boldsymbol{x}}\| \cdot \|w_a^{\boldsymbol{x}}\| \leq \mathop{\mathbf{E}}_{\boldsymbol{x}} \sqrt{\sum_{a \in S(\boldsymbol{x})} \|u_a^{\boldsymbol{x}}\|^2 \cdot \sum_{a \in S(\boldsymbol{x})} \|w_a^{\boldsymbol{x}}\|^2} \leq \sqrt{\mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_{a \in S(\boldsymbol{x})} \|w_a^{\boldsymbol{x}}\|^2}. \tag{19}$$

But the expectation inside the root is at most $O(\delta)$ because $A_a^x \approx_\delta B_a^x$. Combining Equations (18) and (19) completes the proof. □

We will typically, though not always, apply Fact 4.31 in the following special case.

Fact 4.32. *Let $\mathscr{G}$ be a game whose questions $(\boldsymbol{x}_1, \boldsymbol{x}_2) \sim \mathscr{G}$ have marginal distribution $\boldsymbol{x}_1 \sim \mathcal{D}$. Suppose $\{A_a^x\}$ and $\{B_a^x\}$ are measurements such that $A_a^x \otimes I \approx_\delta B_a^x \otimes I$ on state ψ and distribution $\mathcal{D}$. Consider the strategies $\mathcal{S}_A = \{\psi, A\}$ and $\mathcal{S}_B = \{\psi, B\}$. If either A or B is a projective measurement (and the other is a POVM measurement), then*

$$\mathrm{val}_{\mathscr{G}}(\mathcal{S}_A) - O(\delta^{1/2}) \leq \mathrm{val}_{\mathscr{G}}(\mathcal{S}_B) \leq \mathrm{val}_{\mathscr{G}}(\mathcal{S}_A) + O(\delta^{1/2}).$$

Proof. First, we observe that

$$A_{a_1}^{x_1} \otimes A_{a_2}^{x_2} \approx_\delta A_{a_1}^{x_1} \otimes B_{a_2}^{x_2} \approx_\delta B_{a_1}^{x_1} \otimes B_{a_2}^{x_2}$$

by Fact 4.20. The result follows by applying Fact 4.32 with "A" set to $A_{a_1}^{x_1} \otimes A_{a_2}^{x_2}$, "$B$" set to $B_{a_1}^{x_1} \otimes B_{a_2}^{x_2}$, and "$\mathcal{D}$" set to the distribution on $(\boldsymbol{x}_1, \boldsymbol{x}_2)$. We note that "$\mathrm{val}(A)$" there is equal to $\mathrm{val}_{\mathscr{G}}(\mathcal{S}_A)$ here and "$\mathrm{val}(B)$" there is equal to $\mathrm{val}_{\mathscr{G}}(\mathcal{S}_B)$ here. □

4.5.5 Generating new measurements

In this section, we show how to combine multiple measurements into a single measurement by "sandwiching" them together.

Fact 4.33. *Let $k \geq 0$ be a constant. Let $\{A^x_{a_1,\ldots,a_k}\}$ be a projective measurement. For each $1 \leq i \leq k$, let $\{(B_i)^x_{a_i}\}$ be a projective measurement, and suppose that*

$$(A^x_{a_i})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_\delta I_{\mathsf{Alice}} \otimes ((B_i)^x_{a_i})_{\mathsf{Bob}}. \tag{20}$$

Define the POVM measurement $\{J^x_{g_1,\ldots,g_k}\}$ as

$$J^x_{a_1,\ldots,a_k} := (B_k)^x_{a_k} \cdots (B_2)^x_{a_2} \cdot (B_1)^x_{a_1} \cdot (B_2)^x_{a_2} \cdots (B_k)^x_{a_k}.$$

Then

$$(A^x_{a_1,\ldots,a_k})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_{\delta^{1/2}} I_{\mathsf{Alice}} \otimes (J^x_{a_1,\ldots,a_k})_{\mathsf{Bob}}.$$

Proof. For each $1 \leq i \leq k$, Equation (20) implies that

$$(A^x_{a_i})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_\delta I_{\mathsf{Alice}} \otimes ((B_i)^x_{a_i})_{\mathsf{Bob}}.$$

Now, we repeatedly apply this using Fact 4.20:

$$\begin{aligned}
(A^x_{a_1,\ldots,a_k})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} &= (A^x_{a_k} \cdots A^x_{a_2} \cdot A^x_{a_1} \cdot A^x_{a_2} \cdots A^x_{a_k})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \\
&\approx_\delta (A^x_{a_k} \cdots A^x_{a_2} \cdot A^x_{a_1} \cdot A^x_{a_2} \cdots A^x_{a_{k-1}})_{\mathsf{Alice}} \otimes ((B_k)^x_{a_k})_{\mathsf{Bob}} \\
&\cdots \\
&\approx_\delta I_{\mathsf{Alice}} \otimes ((B_k)^x_{a_k} \cdots (B_2)^x_{a_2} \cdot (B_1)^x_{a_1} \cdot (B_2)^x_{a_2} \cdots (B_k)^x_{a_k})_{\mathsf{Bob}} \\
&= I_{\mathsf{Alice}} \otimes (J^x_{a_1,\ldots,a_k})_{\mathsf{Bob}}.
\end{aligned}$$

The fact now follows from Fact 4.14 and the fact that A is a projective measurement. □

Next, we extend Fact 4.33 to the case of polynomial measurements (see Section 4.7 below). These are structured measurements in which the prover returns the evaluation of a function sampled independently from their input. The goal is to retain this structure even after "sandwiching" them together.

Fact 4.34. *Let $k \geq 0$ be a constant. Let $\mathcal{D}$ be a distribution on questions $x \in \mathcal{X}$. For each $1 \leq i \leq k$, let $\mathcal{G}_i$ be a set of functions $g_i : \mathcal{X} \to \mathcal{R}_i$. and let $\{G^i_g\}$ be a projective measurement with outcomes from this set. Suppose that the set $\mathcal{G}_i$ has the following distance property: for any two nonequal $g_i, g'_i \in \mathcal{G}_i$, the probability that $g_i(\boldsymbol{x}) = g'_i(\boldsymbol{x})$, over a random $\boldsymbol{x} \sim \mathcal{D}$, is at most ϵ.*

Let $\{A_{g_1,\ldots,g_k}\}$ be a projective measurement with outcomes $g_i \in \mathcal{F}_i$. For each $1 \leq i \leq k$, suppose that

$$(A_{[g_i(x)=a_i]})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_\delta I_{\mathsf{Alice}} \otimes (G^i_{[g_i(x)=a_i]})_{\mathsf{Bob}}. \tag{21}$$

Define the POVM measurement $\{J_{g_1,\ldots,g_k}\}$ as

$$J_{g_1,\ldots,g_k} := G^k_{g_k} \cdots G^2_{g_2} \cdot G^1_{g_1} \cdot G^2_{g_2} \cdots G^k_{g_k}.$$

Then

$$(A_{[g_1(x),\ldots,g_k(x)=a_1,\ldots,a_k]})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_{(\delta+\epsilon)^{1/2}} I_{\mathsf{Alice}} \otimes (J_{[g_1(x),\ldots,g_k(x)=a_1,\ldots,a_k]})_{\mathsf{Bob}}.$$

Proof. Let $1 \leq i \leq k$. By Equation (21), if Alice measures with A, producing $\boldsymbol{g}_i$, and Bob measures with G^i, producing $\boldsymbol{g}'_i$, then the probability that $\boldsymbol{g}_i(\boldsymbol{x}) \neq \boldsymbol{g}'_i(\boldsymbol{x})$ is $O(\delta)$. Write η for the probability that $\boldsymbol{g}_i \neq \boldsymbol{g}'_i$. Then we have the expression $\eta \cdot (1-\epsilon) \leq O(\delta)$ or, equivalently, $\eta \leq O(\delta/(1-\epsilon))$. When $\epsilon < 1/2$, this gives the bound $\eta \leq O(\delta)$, and when $\epsilon \geq 1/2$, we have the trivial bound $\eta \leq O(\epsilon)$. As a result, $\eta = O(\delta + \epsilon)$.

In conclusion,

$$(A_{g_i})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_{\delta+\epsilon} I_{\mathsf{Alice}} \otimes (G^i_{g_i})_{\mathsf{Bob}}.$$

We can now apply Fact 4.33 to $A_{g_1,\ldots,g_k}$ and the $G^i_{g_i}$ measurements. It implies that

$$(A_{g_1,\ldots,g_k})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_{(\delta+\epsilon)^{1/2}} I_{\mathsf{Alice}} \otimes (J_{g_1,\ldots,g_k})_{\mathsf{Bob}}.$$

The fact now follows from the data processing inequality Fact 4.26. □

In our next fact, we show that Fact 4.34 holds even when we drop the structured assumption on the A matrix. The tradeoff is that we must now assume that the k different measurements act on different parts of the input string. In this case, the distance condition becomes slightly more cumbersome to state.

Fact 4.35. *Let $k \geq 0$ be a constant. Let $\mathcal{D}$ be a distribution on questions $(x, y_1, \ldots, y_k)$, where each $y_i \in \mathcal{Y}_i$. For each $1 \leq i \leq k$, let $\mathcal{G}_i$ be a set of functions $g_i : \mathcal{Y}_i \to \mathcal{R}_i$. and let $\{(G_i)^x_g\}$ be a projective measurement with outcomes from this set. (For the $i = 1$ case, we also allow this measurement to be a POVM.) Suppose that the set $\mathcal{G}_i$ has the following distance property: fix a question $z = (x, y_1, \ldots, y_{i-1}, y_{i+1}, \ldots, y_k)$, and let $\mathcal{D}_z$ be the distribution on y_i conditioned on the other outcomes z. Then for any two nonequal $g_i, g'_i \in \mathcal{G}_i$, the probability that $g_i(\boldsymbol{y}_i) = g'_i(\boldsymbol{y}_i)$, over a random $\boldsymbol{y}_i \sim \mathcal{D}_z$, is at most ϵ.*

Let $\{A^{x,y_1,\ldots,y_k}_{a_1,\ldots,a_k}\}$ be a projective measurement with outcomes $g_i \in \mathcal{F}_i$. For each $1 \leq i \leq k$, suppose that

$$(A^{x,y_1,\ldots,y_k}_{a_i})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_{\delta} I_{\mathsf{Alice}} \otimes ((G_i)^x_{[g_i(y_i)=a_i]})_{\mathsf{Bob}}. \tag{22}$$

Suppose also that

$$(A^{x,y_1,\ldots,y_k}_{a_1,\ldots,a_k})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_{\delta} I_{\mathsf{Alice}} \otimes (A^{x,y_1,\ldots,y_k}_{a_1,\ldots,a_k})_{\mathsf{Bob}}. \tag{23}$$

Define the POVM measurement $\{J^x_{g_1,\ldots,g_k}\}$ as

$$J^x_{g_1,\ldots,g_k} := (G_k)^x_{g_k} \cdots (G_2)^x_{g_2} \cdot (G_1)^x_{g_1} \cdot (G_2)^x_{g_2} \cdots (G_k)^x_{g_k}.$$

Then

$$(A^{x,y_1,\ldots,y_k}_{a_1,\ldots,a_k})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_{\mathrm{poly}(\delta,\epsilon)} I_{\mathsf{Alice}} \otimes (J^x_{[g_1(y_1),\ldots,g_k(y_k)=a_1,\ldots,a_k]})_{\mathsf{Bob}}.$$

Proof. First, we show how to reduce this to the $k = 2$ case. Then we prove it for that case. Assume the fact holds when $k = 2$. Define the POVM measurement $\{(J_i)^x_{g_1,\ldots,g_i}\}$ as

$$(J_i)^x_{g_1,\ldots,g_i} := (G_i)^x_{g_i} \cdots (G_2)^x_{g_2} \cdot (G_1)^x_{g_1} \cdot (G_2)^x_{g_2} \cdots (G_i)^x_{g_i}.$$

We will show by induction that

$$(A^{x,y_1,\ldots,y_k}_{a_1,\ldots,a_i})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_{\mathrm{poly}(\delta,\epsilon)} I_{\mathsf{Alice}} \otimes ((J_i)^x_{[g_1(y_1),\ldots,g_i(y_i)=a_1,\ldots,a_i]})_{\mathsf{Bob}}, \tag{24}$$

the base case being trivial. Assume this holds for i. We apply the $k = 2$ case as follows: consider the question tuple $(y_1, \ldots, y_i)$ as a single question and consider functions of the form $(y_1, \ldots, y_i) \mapsto (g_1(y_1), \ldots, g_i(y_i))$. Then the first POVM measurement is J_i, which satisfies Equation (22) due to

Equation (24). The second measurement is the projector G_{i+1}. Then the $k = 2$ case immediately implies the $i + 1$ case of Equation (24).

Now we prove the $k = 2$ case. Our goal is to show that

$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2} \sum_{a_1,g_2} \langle\psi| (A^{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2}_{a_1,g_2(\boldsymbol{y}_2)})_{\mathsf{Alice}} \otimes ((G_2)^{\boldsymbol{x}}_{g_2} \cdot (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \cdot (G_2)^{\boldsymbol{x}}_{g_2})_{\mathsf{Bob}} |\psi\rangle \tag{25}$$

is at least $1 - \mathrm{poly}(\delta, \epsilon)$. We will do this by showing that

$$((G_1)^{x}_{[g_1(y_1)=a_1]} \cdot (G_2)^{x}_{g_2})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_{\mathrm{poly}(\delta,\epsilon)} ((G_2)^{x}_{g_2} \cdot (G_1)^{x}_{[g_1(y_1)=a_1]})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}. \tag{26}$$

is at most $\mathrm{poly}(\delta, \epsilon)$. To see that this is sufficient, note that the related expression

$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2} \sum_{a_1,g_2} \langle\psi| (A^{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2}_{a_1,g_2(\boldsymbol{y}_2)})_{\mathsf{Alice}} \otimes ((G_2)^{\boldsymbol{x}}_{g_2} \cdot (G_2)^{\boldsymbol{x}}_{g_2} \cdot (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]})_{\mathsf{Bob}} |\psi\rangle$$

is exactly equal to 1 because G_2 is a projector. Taking the difference between this and Equation (25), we get

$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2} \sum_{a_1,g_2} \langle\psi| (A^{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2}_{a_1,g_2(\boldsymbol{y}_2)})_{\mathsf{Alice}} \otimes ((G_2)^{\boldsymbol{x}}_{g_2} \cdot ((G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \cdot (G_2)^{\boldsymbol{x}}_{g_2} - (G_2)^{\boldsymbol{x}}_{g_2} \cdot (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}))_{\mathsf{Bob}} |\psi\rangle .$$

Cauchy-Schwarz allows us to bound this by

$$\leq \mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2} \sqrt{\sum_{a_1,g_2} \| (A^{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2}_{a_1,g_2(\boldsymbol{y}_2)})_{\mathsf{Alice}} \otimes ((G_2)^{\boldsymbol{x}}_{g_2})_{\mathsf{Bob}} |\psi\rangle \|^2} \cdot \sqrt{\sum_{a_1,g_2} \| I_{\mathsf{Alice}} \otimes ((G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \cdot (G_2)^{\boldsymbol{x}}_{g_2} - (G_2)^{\boldsymbol{x}}_{g_2} \cdot (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]})_{\mathsf{Bob}} |\psi\rangle \|^2}.$$

The expression inside the first square root is always at most 1. This allows us to bring the expectation into the second square root by Jensen's inequality, and the resulting expression we can bound due to Equation (26).

Now we bound Equation (26). Showing this is small is equivalent to showing

$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2} \sum_{a_1,g_2} \| ((G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \cdot (G_2)^{\boldsymbol{x}}_{g_2} - (G_2)^{\boldsymbol{x}}_{g_2} \cdot (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} |\psi\rangle \|^2$$

is small. Expanding this, we get

$$\begin{aligned}\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2} \sum_{a_1,g_2} \langle\psi| \big(&(G_2)^{\boldsymbol{x}}_{g_2} \cdot (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \cdot (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \cdot (G_2)^{\boldsymbol{x}}_{g_2} \otimes I_{\mathsf{Bob}} \\ &+ (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \cdot (G_2)^{\boldsymbol{x}}_{g_2} \cdot (G_2)^{\boldsymbol{x}}_{g_2} \cdot (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \otimes I_{\mathsf{Bob}} \\ &- (G_2)^{\boldsymbol{x}}_{g_2} \cdot (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \cdot (G_2)^{\boldsymbol{x}}_{g_2} \cdot (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \otimes I_{\mathsf{Bob}} \\ &- (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \cdot (G_2)^{\boldsymbol{x}}_{g_2} \cdot (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \cdot (G_2)^{\boldsymbol{x}}_{g_2} \otimes I_{\mathsf{Bob}} \big) |\psi\rangle .\end{aligned} \tag{27}$$

We *do* know that G_1 and G_2 satisfy some form of commutation. Because they satisfy Equation (22) and A is a projector, we know that

$$((G_1)^{x}_{[g_1(y_1)=a_1]} \cdot (G_2)^{x}_{[g_2(y_2)=a_2]})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_{\delta} ((G_2)^{x}_{[g_2(y_2)=a_2]} \cdot (G_1)^{x}_{[g_1(y_1)=a_1]})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}.$$

Expanding this as above, we can bound the following expression by δ:

$$\begin{aligned}\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2}\sum_{a_1,a_2}\langle\psi|\,\big((G_2)^{\boldsymbol{x}}_{[g_2(\boldsymbol{y}_2)=a_2]}\cdot(G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\cdot(G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\cdot(G_2)^{\boldsymbol{x}}_{[g_2(\boldsymbol{y}_2)=a_2]}\otimes I_{\mathsf{Bob}}\\ +(G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\cdot(G_2)^{\boldsymbol{x}}_{[g_2(\boldsymbol{y}_2)=a_2]}\cdot(G_2)^{\boldsymbol{x}}_{[g_2(\boldsymbol{y}_2)=a_2]}\cdot(G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\otimes I_{\mathsf{Bob}}\\ -(G_2)^{\boldsymbol{x}}_{[g_2(\boldsymbol{y}_2)=a_2]}\cdot(G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\cdot(G_2)^{\boldsymbol{x}}_{[g_2(\boldsymbol{y}_2)=a_2]}\cdot(G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\otimes I_{\mathsf{Bob}}\\ -(G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\cdot(G_2)^{\boldsymbol{x}}_{[g_2(\boldsymbol{y}_2)=a_2]}\cdot(G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\cdot(G_2)^{\boldsymbol{x}}_{[g_2(\boldsymbol{y}_2)=a_2]}\otimes I_{\mathsf{Bob}}\big)\,|\psi\rangle\,. \qquad (28)\end{aligned}$$

We can therefore show Equation (27) is small by upper-bounding (Equation (27)−Equation (28)). There are four terms in this difference; write Δ_i for the i-th term in Equation (27) minus the i-th term in Equation (28). We will bound each Δ_i one-by-one.

The first term in the difference, Δ_1, is

$$\begin{aligned}\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2}\sum_{a_1}\sum_{g_2}\langle\psi|\,(G_2)^{\boldsymbol{x}}_{g_2}\cdot(G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\cdot(G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\cdot(G_2)^{\boldsymbol{x}}_{g_2}\otimes I_{\mathsf{Bob}}\,|\psi\rangle\\ -\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2}\sum_{a_1,a_2}\langle\psi|\,(G_2)^{\boldsymbol{x}}_{[g_2(\boldsymbol{y}_2)=a_2]}\cdot(G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\cdot(G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\cdot(G_2)^{\boldsymbol{x}}_{[g_2(\boldsymbol{y}_2)=a_2]}\otimes I_{\mathsf{Bob}}\,|\psi\rangle\,.\end{aligned}$$

The first of these terms is at most 1, and so we just have to show that the second term is close to 1 as well. Note that by repeated applications of Equation (22), we have that

$$(G_2)^{\boldsymbol{x}}_{[g_2(\boldsymbol{y}_2)=a_2]}\cdot(G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\cdot(G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\cdot(G_2)^{\boldsymbol{x}}_{[g_2(\boldsymbol{y}_2)=a_2]}\otimes I_{\mathsf{Bob}}\approx_\delta I_{\mathsf{Alice}}\otimes A^{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2}_{a_1,a_2}.$$

But then by Fact 4.31, the expression we want to lower-bound is $O(\delta^{1/2})$-close to

$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2}\sum_{a_1,a_2}\langle\psi|\,I_{\mathsf{Alice}}\otimes A^{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2}_{a_1,a_2}\,|\psi\rangle\,,$$

which is exactly 1. As a result, Δ_1 is at most $O(\delta^{1/2})$.

The second term in the difference, Δ_2, can be written as

$$-\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2}\sum_{a_1}\sum_{\substack{g_2\neq g_2',\\ g_2(\boldsymbol{y}_2)=g_2'(\boldsymbol{y}_2)}}\langle\psi|\,\big((G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\cdot(G_2)^{\boldsymbol{x}}_{g_2}\cdot(G_2)^{\boldsymbol{x}}_{g_2'}\cdot(G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\otimes I_{\mathsf{Bob}}\big)\,|\psi\rangle\,.$$

This is zero because G_2 is a projector.

The third and fourth terms in Equation (27) are complex conjugates of each other, as are the third and fourth terms in Equation (28). As a result, it suffices to bound the magnitude of Δ_4, and this will serve to bound Δ_3 as well. We begin by manipulating the fourth term in Equation (27); specifically, we will show that it is close to

$$-\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2}\sum_{a_1,g_2}\langle\psi|\,(G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\cdot(G_2)^{\boldsymbol{x}}_{g_2}\cdot(G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\otimes(G_2)^{\boldsymbol{x}}_{g_2}\,|\psi\rangle\,. \qquad (29)$$

To do so, we take their difference:

$$\begin{aligned}\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2}\sum_{a_1,g_2}\langle\psi|\,\big((G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\cdot(G_2)^{\boldsymbol{x}}_{g_2}\cdot(G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]}\otimes I_{\mathsf{Bob}}\big)\\ \cdot\big(I_{\mathsf{Alice}}\otimes(G_2)^{\boldsymbol{x}}_{g_2}-(G_2)^{\boldsymbol{x}}_{g_2}\otimes I_{\mathsf{Bob}}\big)\,|\psi\rangle\,.\end{aligned}$$

To bound the magnitude, we apply Cauchy-Schwarz:

$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2} \sqrt{\sum_{a_1,g_1} \left\| ((G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \cdot (G_2)^{\boldsymbol{x}}_{g_2} \cdot (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \otimes I_{\mathsf{Bob}}) \left|\psi\right\rangle \right\|^2} \\ \cdot \sqrt{\sum_{a_1,g_1} \left\| ((G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \otimes I_{\mathsf{Bob}}) \cdot (I_{\mathsf{Alice}} \otimes (G_2)^{\boldsymbol{x}}_{g_2} - (G_2)^{\boldsymbol{x}}_{g_2} \otimes I_{\mathsf{Bob}}) \left|\psi\right\rangle \right\|^2}.$$

The expression inside the first square root is always at most 1. This allows us to bring the expectation into the second square root by Jensen's inequality. Because G_1 is a POVM, we can bound the resulting expectation by

$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2} \sum_{g_1} \left\| (I_{\mathsf{Alice}} \otimes (G_2)^{\boldsymbol{x}}_{g_2} - (G_2)^{\boldsymbol{x}}_{g_2} \otimes I_{\mathsf{Bob}}) \left|\psi\right\rangle \right\|^2. \tag{30}$$

To bound this, we note that Equations (22) and (23) along with Fact 4.29 imply that

$$(G_2)^{x}_{[g_2(y_2)=a_2]} \otimes I_{\mathsf{Bob}} \simeq_{\delta} I_{\mathsf{Bob}} \otimes (G_2)^{x}_{[g_2(y_2)=a_2]}.$$

Using the distance properties of $\mathcal{G}_2$, this implies that

$$(G_2)^{x}_{g_2} \otimes I_{\mathsf{Bob}} \simeq_{\delta+\epsilon} I_{\mathsf{Bob}} \otimes (G_2)^{x}_{g_2}.$$

Hence, Equation (30) is at most $O((\delta+\epsilon)^{1/2})$. A similar argument shows that the fourth term in Equation (28) is $O(\delta^{1/2})$-close to

$$- \mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2} \sum_{a_1,a_2} \left\langle\psi\right| (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \cdot (G_2)^{\boldsymbol{x}}_{[g_2(\boldsymbol{y}_2)=a_2]} \cdot (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \otimes (G_2)^{\boldsymbol{x}}_{[g_2(\boldsymbol{y}_2)=a_2]} \left|\psi\right\rangle. \tag{31}$$

Now, we compute Equation (29) minus Equation (31):

$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2} \sum_{a_1} \sum_{\substack{g_2,g_2' \\ g_2(\boldsymbol{y}_2)\neq g_2'(\boldsymbol{y}_2)}} \left\langle\psi\right| (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \cdot (G_2)^{\boldsymbol{x}}_{g_2} \cdot (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \otimes (G_2)^{\boldsymbol{x}}_{g_2'} \left|\psi\right\rangle$$
$$= \mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2} \sum_{a_1} \sum_{g_2,g_2'} \left\langle\psi\right| (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \cdot (G_2)^{\boldsymbol{x}}_{g_2} \cdot (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \otimes (G_2)^{\boldsymbol{x}}_{g_2'} \left|\psi\right\rangle \cdot \mathbf{1}(g_2,g_2',\boldsymbol{y}_2),$$

where $\mathbf{1}(g_2,g_2',\boldsymbol{y}_2)$ is the indicator that $g_2 \neq g_2'$ but $g_2(\boldsymbol{y}_2) = g_2'(\boldsymbol{y}_2)$. This is the only part of the expression that depends on $\boldsymbol{y}_2$, and by our distance assumption it is at most ϵ in expectation. Since the rest of the expression is guaranteed to be positive, we can upper-bound this by

$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}_1} \sum_{a_1} \sum_{g_2,g_2'} \left\langle\psi\right| (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \cdot (G_2)^{\boldsymbol{x}}_{g_2} \cdot (G_1)^{\boldsymbol{x}}_{[g_1(\boldsymbol{y}_1)=a_1]} \otimes (G_2)^{\boldsymbol{x}}_{g_2'} \left|\psi\right\rangle \cdot \epsilon.$$

But the remaining part of the expression is at most 1, and so in total we can upper-bound it by ϵ. This completes the proof. □

4.6 Commuting EPR strategies

In this section, we introduce a class of strategies important for our proof.

Definition 4.36. A strategy $\mathcal{S} = (\psi, M)$ is called an *EPR strategy* if it satisfies the following two properties. First, there is an integer k and powers of two $q_1, \ldots, q_k$ such that

$$|\psi\rangle = |\mathrm{EPR}_{q_1}\rangle \otimes |\mathrm{EPR}_{q_2}\rangle \otimes \cdots \otimes |\mathrm{EPR}_{q_k}\rangle .$$

Second, for each question x, M^x is a projective measurement. If for all questions x and answers a, M_a^x is a real-valued matrix, we say that the strategy is *real*

In addition, given a game $\mathscr{G}$, we say that a real EPR strategy $\mathcal{S}$ is a *real commuting EPR strategy (with respect to $\mathscr{G}$)* if for every (x_1, x_2) in the support of $\mathcal{S}$ and every a_1, a_2, $M_{a_1}^{x_1}$ commutes with $M_{a_2}^{x_2}$. We denote the set of real commuting EPR strategies by $\mathrm{ComEPR}(\mathscr{G})$.

Real commuting EPR strategies are motivated by the *completeness* cases that arise in this work. We give a series of transformations which modify games to make them sound against increasingly broader sets of strategies. Unfortunately, these transformations are not complete for all strategies, in the sense that value-1 strategies may be mapped to value-less-than-1 strategies, but we will be careful to ensure that they *are* complete for all commuting EPR strategies. For the majority of the paper, the one property of commuting EPR strategies that we will use, not shared by all value-1 strategies, is the following.

Fact 4.37. *Let (ψ, M) be a real EPR strategy. Then $M_a^x \otimes I_{\mathsf{Bob}} \simeq_0 I_{\mathsf{Alice}} \otimes M_a^x$ for every distribution on x.*

Proof. From the definition of EPR strategies, we know that $|\psi\rangle = |\mathrm{EPR}_{q_1}\rangle \otimes \ldots \otimes |\mathrm{EPR}_{q_k}\rangle \in (\mathbb{C}^{q_1 \cdot q_2 \cdot \cdots \cdot q_k})^{\otimes 2}$. We may choose a basis $\{|i\rangle : 1 \leq i \leq q_1 \cdot q_2 \cdot \cdots \cdot q_k\}$ for $\mathbb{C}^{q_1 \cdot \cdots \cdot q_k}$, so that

$$|\psi\rangle = \sum_i |i\rangle_{\mathsf{Alice}} \otimes |i\rangle_{\mathsf{Bob}} .$$

Let us denote the components of M_a^x by the notation $(M_a^x)_{ij}$, so that $M_a^x = \sum_{ij} (M_a^x)_{ij} |i\rangle \langle j|$. Now, for an arbitrary pair x, a, we can compute the post-measurement states from applying M_a^x on Alice's and Bob's systems.

$$\begin{aligned} M_a^x \otimes I_{\mathsf{Bob}} |\psi\rangle &= (M_a^x \otimes I_{\mathsf{Bob}}) \sum_i |i\rangle_{\mathsf{Alice}} \otimes |i\rangle_{\mathsf{Bob}} \\ &= \sum_{ij} (M_a^x)_{ij} |i\rangle \otimes |j\rangle \\ &= \sum_{ij} (M_a^x)_{ji} |i\rangle \otimes |j\rangle \\ &= (I_{\mathsf{Alice}} \otimes M_a^x) |\psi\rangle , \end{aligned}$$

where in going from the second to the third line, we have used the fact that M_a^x is real and Hermitian, and thus symmetric. □

The following fact is a useful special case.

Fact 4.38. *Let $n > 0$, $q = 2^t$, and $W \in \{X, Z\}$. Then $\tau_u^W \otimes I \approx_0 I \otimes \tau_u^W$ on the state $|\mathrm{EPR}_q^n\rangle$.*

Proof. By Equation (7), we can write

$$|\tau_u^X\rangle = \frac{1}{\sqrt{q}} \sum_{v \in \mathbb{F}_q} \omega^{-\mathrm{tr}[uv]} |v\rangle\,, \qquad |\tau_u^Z\rangle = |u\rangle\,.$$

The second of these self-evidently has real-valued coefficients. As for the first, $q = 2^t$ implies that $p = 2$. This means that $\omega = -1$ and $\mathrm{tr}[uv] \in \{0, 1\}$ for all u, v. As a result, it too has real-valued coefficients. The fact then follows from Fact 4.37. □

This property of real commuting strategies is useful for answer reduction because it allows us to perform oracularization, giving one prover both questions x_1 and x_2 so that they may simulate the action of both provers by simultaneously measuring M^{x_1} and M^{x_2}. For more details, see Part V.

4.7 Quantum soundness of the classical low-degree test

An important tool for quantum protocols is a version of the Raz-Safra theorem (Theorem 3.12) in which the soundness of the low-degree test is extended to hold even in the case when the provers are allowed to share entanglement. For the plane-versus-point test, this was first developed by Vidick in [Vid16], but for technical reasons he could only show it for the case of three or more quantum provers. In [NV18b], this was improved to hold for the two-prover case, and this is the result we use in this work. We begin by defining the class of polynomial measurements.

Definition 4.39. Define $\mathrm{PolyMeas}(m, d, q)$ to be the set of POVM measurements whose outcomes correspond to degree-d, $\mathbb{F}_q$-valued polynomials. In other words, $G \in \mathrm{PolyMeas}(m, d, q)$ if $G = \{G_g\}_g$ with outcomes degree-d polynomials $g : \mathbb{F}_q^m \to \mathbb{F}_q$. More generally, we let $\mathrm{PolyMeas}(m, d, q, \ell)$ be the set of measurements $G = \{G_{g_1,\ldots,g_\ell}\}$ outputting ℓ degree-d polynomials $g_i : \mathbb{F}_q^m \to \mathbb{F}_q$.

The following theorem establishes the quantum soundness of the classical low-degree test in the $k = 2$ case.

Theorem 4.40 (Quantum soundness of the classical low-degree test [NV18b, Theorem 2])**.** *There exists a constant $c > 0$ and a function $\delta(\epsilon) = \mathrm{poly}(\epsilon, dm/q^c)$ such that the following holds. Suppose Alice and Bob are entangled provers who pass $\mathscr{G}_{\mathrm{Surface}}(m, d, q, 2)$ with probability at least $1 - \epsilon$ using the strategy (ψ, M), where M consists of projective measurements. Then there exists a POVM measurement $G \in \mathrm{PolyMeas}(m, d, q)$ such that*

$$M_b^w \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g(w)=b]}, \qquad G_g \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_g,$$

where the first is on the uniform distribution over $\mathbb{F}_q^m$.

Remark 4.41. The statement of Theorem 4.40 is modified from how it appears in [NV18b, Theorem 2] to better suit our needs. In this remark, we show how to derive our version from theirs, which is stated as follows.

- There exists a constant $c > 0$ and a function $\delta(\epsilon) = \mathrm{poly}(\epsilon)$ such that the following holds. Suppose $q \geq (dm/\epsilon)^c$. Then if Alice and Bob pass the surface-versus-point test with probability $1 - \epsilon$, there is a measurement $G \in \mathrm{PolyMeas}(m, d, q)$ such that

$$\mathop{\mathbf{E}}_{\boldsymbol{s}} \sum_g \sum_{f \neq g|_s} \langle\psi| M_f^{\boldsymbol{s}} \otimes G_g |\psi\rangle \leq \delta(\epsilon), \qquad \sum_g \langle\psi| G_g \otimes (I - G_g) |\psi\rangle \leq \delta(\epsilon). \tag{32}$$

These are equivalent to the statements

$$M_f^s \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g|_s=f]}, \qquad G_g \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_g,$$

where the first is on the uniform distribution over surface in $\mathbb{F}_q^m$. The second of these matches the corresponding statement above. Next, by Fact 4.26 we derive

$$M^s_{[f(w)=b]} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g(w)=b]} \qquad \text{and} \qquad G_{[g(w)=b]} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g(w)=b]}.$$

with respect to the distribution $(\boldsymbol{s}, \boldsymbol{w})$ from $\mathscr{G}_{\text{Surface}}(m, d, q, 2)$. On top of that, since the strategy passes the test with probability $1 - \epsilon$,

$$M_b^w \otimes I_{\mathsf{Bob}} \simeq_\epsilon I_{\mathsf{Alice}} \otimes M^s_{[f(w)=b]}$$

As a result, if we use Fact 4.13 to switch these to "$\approx_\delta$" statements, then

$$M_b^w \otimes I_{\mathsf{Bob}} \approx_\epsilon I_{\mathsf{Alice}} \otimes M^s_{[f(w)=b]} \approx_{\delta(\epsilon)} G_{[g(w)=b]} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g(w)=b]}.$$

The result now follows from the triangle inequality (Fact 4.28) followed by Fact 4.14 and the fact that M was assumed to be projective.

Finally, we remove the condition on q using a trick from [NV18a]. If $q < (dm/\epsilon)^c$, then we select $\epsilon' > \epsilon$ such that $q = (dm/\epsilon')^c$. Alice and Bob also pass the plane-versus-point test with probability $1 - \epsilon'$ because $1 - \epsilon' < 1 - \epsilon$, and so we can apply the theorem with these parameters, giving a robustness of $\delta(\epsilon') = \delta(dm/q^{1/c})$. (In the case when $\epsilon' > 1$, which is not allowed, this bound trivially still holds because $dm/q^{1/c} > 1$.) In general, then, we can remove the condition on q so long as we replace the robustness of $\text{poly}(\epsilon)$ with $\text{poly}(\epsilon, dm/q^{1/c})$, which holds in both cases.

We will use the following proposition about polynomial measurements several times.

Proposition 4.42. *Let $d > 0$ be an integer. Consider a distribution $\mathcal{D}$ on pairs $(\boldsymbol{s}, \boldsymbol{u})$, where $\boldsymbol{s}$ is a subspace in $\mathbb{F}_q^m$ and $\boldsymbol{u}$ is a uniformly random point in $\boldsymbol{s}$. Let $\{M_f^s\}$ be a measurement whose outcomes are degree-d polynomials $f : s \to \mathbb{F}_q$, and let $G \in \text{PolyMeas}(m, d, q)$. Suppose that*

$$M^s_{[f(u)=b]} \otimes I_{\mathsf{Bob}} \simeq_\delta I_{\mathsf{Alice}} \otimes G_{[g(u)=b]}$$

with respect to $\mathcal{D}$. Then

$$M_f^s \otimes I_{\mathsf{Bob}} \simeq_{\delta + d/q} I_{\mathsf{Alice}} \otimes G_{[g|_s=f]}$$

with respect to $\mathcal{D}$.

Proof. Suppose the verifier (i) samples $(\boldsymbol{s}, \boldsymbol{u}) \sim \mathcal{D}$, (ii) gives Alice $\boldsymbol{s}$, who measures with $M^{\boldsymbol{s}}$ and returns her outcome $\boldsymbol{f} : \boldsymbol{s} \to \mathbb{F}_q$, (iii) receives $\boldsymbol{g} : \mathbb{F}_q^m \to \mathbb{F}_q$ from Bob, sampled via G, and (iv) accepts if $\boldsymbol{f}(\boldsymbol{u}) = \boldsymbol{g}(\boldsymbol{u})$. Then by assumption, the verifier accepts with probability at least $1 - O(\delta)$.

We can use this to bound the probability that $\boldsymbol{f}$ and $\boldsymbol{g}$ disagree on the subspace $\boldsymbol{s}$. By Schwartz-Zippel (Lemma 3.6), conditioned on $\boldsymbol{f}$ and $\boldsymbol{g}$ disagreeing, the probability they disagree on a random point $\boldsymbol{u} \sim \boldsymbol{s}$ is at least $1 - d/q$. This gives us the inequality $\mathbf{Pr}[\boldsymbol{f} \neq \boldsymbol{g}|_{\boldsymbol{s}}] \cdot (1 - d/q) \leq O(\delta)$. Now, assume first that $q \geq 2d$. Then this bound implies, via Fact 4.13, that

$$M_f^s \otimes I_{\mathsf{Bob}} \approx_{\delta + d/q} I_{\mathsf{Alice}} \otimes G_{[g|_s=f]}. \tag{33}$$

On the other hand, when $q \geq 2d$, then this bound is also true for trivial reasons. This is because we can pick $\delta(\cdot)$ such that $\delta(\epsilon) \geq 1$ in this case. □

4.8 Quantum soundness of the classical simultaneous low-degree test

We would now like to use Theorem 4.40 to show quantum soundness for the simultaneous classical low-degree test. This will be done using the same reduction presented in Section 3.6. The main result is the following.

Theorem 4.43 (Quantum soundness of the simultaneous classical low-degree test)**.** *There exists a constant $c > 0$ and a function $\delta(\epsilon) = \mathrm{poly}(\epsilon, d(m+\ell)/q^c)$ such that the following holds. Suppose Alice and Bob are entangled provers who pass $\mathscr{G}^{\ell}_{\mathrm{Surface}}(m, d, q, 2)$ with probability at least $1-\epsilon$ using the strategy (ψ, M), where M consists of projective measurements. Then there exists a measurement $G \in \mathrm{PolyMeas}(m, d, q, \ell)$ such that*

$$M^{w}_{b_1,\dots,b_\ell} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g_1(w),\dots,g_\ell(w)=b_1,\dots,b_\ell]}, \qquad G_{g_1,\dots,g_\ell} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{g_1,\dots,g_\ell},$$

where the first is on the uniform distribution over $\mathbb{F}_q^m$.

Proof. Suppose Alice and Bob pass $\mathscr{G}^{\ell}_{\mathrm{Surface}}(m, d, q, 2)$ with probability at least $1-\epsilon$. We will use them to simulate two provers, "Combined Alice" and "Combined Bob", who pass the single-function low-degree test $\mathscr{G}_{\mathrm{Surface}}(\ell+m, d+1, q, 2)$ with probability at least $1-\epsilon$. They are specified as follows:

- **Combined Alice:** Given $\boldsymbol{s} \subseteq \mathbb{F}_q^{\ell+m}$, draw $\boldsymbol{s}' \sim_2 \boldsymbol{s}_{\mathrm{proj}}$. Give it to Alice, who responds with $\boldsymbol{f}_1, \dots, \boldsymbol{f}_\ell : \boldsymbol{s}' \to \mathbb{F}_q$. Output the function $\mathrm{combine}_{\boldsymbol{f}}|_{\boldsymbol{s}}$.
- **Combined Bob:** Given $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{F}_q^{\ell+m}$, compute $\boldsymbol{y} \in \mathbb{F}_q^m$. Give it to Bob, who responds with $\boldsymbol{b}_1, \dots, \boldsymbol{b}_\ell \in \mathbb{F}_q$. Return $\mathrm{combine}_{\boldsymbol{b}}(\boldsymbol{x}) \in \mathbb{F}_q$.

By Proposition 3.15, $\boldsymbol{s}'$ and $\boldsymbol{y}$ are distributed as the questions in $\mathscr{G}^{\ell}_{\mathrm{Surface}}(m, d, q, 2)$. Using our assumption on Alice and Bob, this means that $\boldsymbol{f}_1(\boldsymbol{y}) = \boldsymbol{b}_1$, $\dots$, $\boldsymbol{f}_\ell(\boldsymbol{y}) = \boldsymbol{b}_\ell$ with probability at least $1-\epsilon$. As a result, $(\mathrm{combine}_{\boldsymbol{f}}|_{\boldsymbol{s}})(\boldsymbol{x}, \boldsymbol{y}) = \mathrm{combine}_{\boldsymbol{b}}(\boldsymbol{y})$ with probability at least $1-\epsilon$. By Proposition 3.16, $\mathrm{combine}_{\boldsymbol{f}}|_{\boldsymbol{s}}$ is a degree-$(d+1)$ function on $\boldsymbol{s}$, and so it is a valid response to subspace queries. This means Combined Alice and Bob pass $\mathscr{G}_{\mathrm{Surface}}(\ell+m, d+1, q, 2)$ with probability at least $1-\epsilon$.

Thus, we can apply Theorem 4.40. It gives a measurement $G \in \mathrm{PolyMeas}(\ell+m, d+1, q)$ such that

$$M^{y}_{[\mathrm{combine}_b(x)=\nu]} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g(x,y)=\nu]}, \qquad G_g \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_g, \tag{34}$$

where $\delta(\epsilon) = \mathrm{poly}(\epsilon, (d+1)(\ell+m)/q^c)$. This means that if we give Alice $\boldsymbol{y}$ and she returns $\boldsymbol{b}$, and Bob simply returns $\boldsymbol{g}$, then $\mathrm{combine}_{\boldsymbol{b}}(\boldsymbol{x}) = g(\boldsymbol{x}, \boldsymbol{y})$ with probability at least $1-\delta(\epsilon)$.

We would like to show that $\boldsymbol{g}$ is exactly linear in x with high probability, over the randomness in the measurement G. Let us condition on a $\boldsymbol{g}$ which is not exactly linear. By Proposition 3.18, the probability that $\boldsymbol{g}|_{\boldsymbol{y}}$ is not exactly linear is at least $1-(d+1)/q$. On the other hand, because $\mathrm{combine}_{\boldsymbol{b}}(\boldsymbol{x})$ is always exactly linear by construction, the probability that $\boldsymbol{g}|_{\boldsymbol{y}}(\boldsymbol{x}) = \mathrm{combine}_{\boldsymbol{b}}(\boldsymbol{x})$ is at most $(d+1)/q$ by Schwartz-Zippel (Lemma 3.6). As a result, the probability that $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{y}) = \mathrm{combine}_{\boldsymbol{b}}(\boldsymbol{x})$ is at most $(d+1)/q + (d+1)/q$. Thus, if we write μ_{linear} for the probability that $\boldsymbol{g}$ is exactly linear, we have equality at most $\mu_{\mathrm{linear}} + 2(d+1)/q$ fraction of the time. Rearranging, $\boldsymbol{g}$ is exactly linear with probability

$$\mu_{\mathrm{linear}} \geq 1 - 2\delta(\epsilon) - 2(d+1)/q.$$

Define a new measurement $\{H_{g_1,\dots,g_\ell}\} \in \mathrm{PolyMeas}(m, d, q, \ell)$ operationally as follows: first, measure G and receive $\boldsymbol{g}$. If it is exactly linear, it can be written as $\sum_i x_i \cdot \boldsymbol{g}_i(y)$, and so output

$\boldsymbol{g}_1, \ldots, \boldsymbol{g}_\ell$. If $\boldsymbol{g}$ is *not* exactly linear, output any arbitrary degree-d polynomials instead. When $\boldsymbol{g}$ is exactly linear, we have $\text{combine}_{\boldsymbol{g}_1,\ldots,\boldsymbol{g}_\ell}(x,y) = \boldsymbol{g}(x,y)$. Since this happens with probability at least $1-\delta(\epsilon)$, we can replace G with H in Equation (34), yielding

$$M^y_{[\text{combine}_b(x)=\nu]} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes H_{[\text{combine}_g(x,y)=\nu]}, \tag{35}$$

On the other hand, because H is just G with data processing applied to its output, we can apply Fact 4.26 to Equation (34). This produces the equation

$$H_{g_1,\ldots,g_\ell} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes H_{g_1,\ldots,g_\ell}.$$

Consider $\boldsymbol{g}_1, \ldots, \boldsymbol{g}_\ell$ drawn by Bob using H. For any fixed b and y, if it is not the case that $g_1(y) = b_1, \ldots, g_\ell(y) = b_\ell$, then the probability that $\text{combine}_{\boldsymbol{g}}(\boldsymbol{x}, y) = \text{combine}_b(\boldsymbol{x})$ over a random $\boldsymbol{x}$ is at most $1/q$ by Schwartz-Zippel, since both are exactly linear functions. Thus, if Alice draws $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_\ell$ given $\boldsymbol{y}$, and we write η for the probability that $\boldsymbol{g}_1(\boldsymbol{y}) = \boldsymbol{b}_1, \ldots, \boldsymbol{g}_\ell(\boldsymbol{y}) = \boldsymbol{b}_\ell$, then the probability that $\text{combine}_{\boldsymbol{g}}(\boldsymbol{x}, \boldsymbol{y}) = \text{combine}_{\boldsymbol{b}}(\boldsymbol{x})$ is at most $\eta + (1-\eta)\cdot 1/q$. Combined with Equation (35), this implies that

$$\mathbf{Pr}[\boldsymbol{g}_1(\boldsymbol{y}) = \boldsymbol{b}_1, \ldots, \boldsymbol{g}_\ell(\boldsymbol{y}) = \boldsymbol{b}_\ell] \geq 1 - \delta(\epsilon) - 1/q.$$

Or, equivalently,

$$M^y_{b_1,\ldots,b_\ell} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g_1(y),\ldots,g_\ell(y)=b_1,\ldots,b_\ell]}.$$ □

4.9 Self-testing

The games presented in Sections 4.7 and 4.8 might be referred to as "measurement testers": if a strategy passes them with high probability, then we can extract some property on its measurements. In this section, we will introduce a significantly stronger notion of testing called *self-testing*. A self-tester is a game in which if a prover passes with high probability, then not only do we do exactly which measurements the prover must be performing, we also know which exactly state it must be performing them on (up to local isometry). (We note that some works use "self-testing" to refer both to "measurement testing" and what we refer to as "self-testing" [NV18a]. In this work, we will reserve the term exclusively for the latter.) We begin with a definition.

Definition 4.44. We say that $\mathcal{S} = (\psi, M)$ is a *partial strategy* for $\mathscr{G}$ if M contains the POVM M^x for only a subset of the questions in $\mathscr{G}$. We call this set of questions $\mathcal{S}$'s *question set*. A strategy $\mathcal{S}' = (\psi, M')$ *extends* $\mathcal{S}$ if $(M')^x = M^x$ for every x in $\mathcal{S}$'s question set.

Next, we define self-testing.

Definition 4.45 (Self-testing)**.** Let $\mathcal{S} = (\psi, G)$ be a partial strategy and $\mathcal{D}$ be a distribution over its question set. A game $\mathscr{G}$ is a *self-test for* $\mathcal{S}$ *over* $\mathcal{D}$ *with robustness* $\delta(\epsilon)$ if it satisfies the following two conditions.

- **Completeness:** There exists a (full) strategy $\mathcal{S}_{\text{full}}$ consistent with $\mathcal{S}$ which passes $\mathscr{G}$ with probability 1.
- **Soundness:** Let $\overline{\mathcal{S}} = (\overline{\psi}, \overline{M})$ be a strategy which passes $\mathscr{G}$ with probability $1-\epsilon$. Then there exists a local isometry $\phi = \phi_{\text{local}} \otimes \phi_{\text{local}}$ and a state $|\text{aux}\rangle$ such that

  $$\|\phi\,|\overline{\psi}\rangle - |\psi\rangle\,|\text{aux}\rangle\,\|^2 \leq \delta(\epsilon).$$

  Furthermore, if we define the new matrices $M^x_a := \phi_{\text{local}} \cdot \overline{M}^x_a \cdot (\phi_{\text{local}})^\dagger$, then

  $$M^x_a \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} (G^x_a \otimes I_{\text{aux}}) \otimes I_{\mathsf{Bob}}, \tag{36}$$

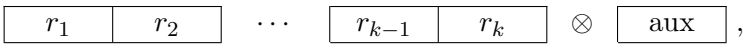
  on states $|\psi\rangle\,|\text{aux}\rangle$ and $|\psi'\rangle$ and distribution $\boldsymbol{x} \sim \mathcal{D}$.

We note that this definition of self-testing differs in several key places from the one given in [NV18a, Definition 2.5]. We will explain these differences in more detail when we cite the quantum low-degree test in Section 6.

Part III
Implementing the registers

5 Register overview

In this part, we implement the quantum registers. Our goal is force Alice and Bob to share a state of the following form:



in which each register r_i contains an EPR state, and aux is a symmetric auxiliary state. In addition, we want the verifier to be able to (i) force the provers to perform Pauli basis queries on some of these registers and report back the outcomes and (ii) "hide" the remaining registers from the provers so that they do not measure them at all.

5.1 Definitions

In this section, we will begin by defining quantum registers for *nonuniform* games. Defining registers for uniform games $\mathscr{G}$ is a little more complicated because we allow the number and size of registers for $\mathscr{G}(\mathsf{input})$ to depend on input. We detail this below in Section 5.3.

Definition 5.1. Let $k \geq 0$ be an integer, and let $n = (n_1, \ldots, n_k)$ and $q = (q_1, \ldots, q_k)$ be k-tuples of integers. A (k, n, q)-*register game* $\mathscr{G}$ is defined as follows.

- Questions x are formatted into two blocks $x = (x_1, x_2)$. The first block contains a list of k Pauli basis queries $x_1 = (W_1, \ldots, W_k)$, where each $W_i \in \{X, Z, H, \perp\}$.
- Answers a are formatted into two blocks $a = (a_1, a_2)$. The first block contains a list of answers to the Pauli basis queries $a_1 = (u_1, \ldots, u_k)$. Here each $u_i \in \mathbb{F}_{q_i}^{n_i} \cup \{\varnothing\}$.

An (k, n, q)-*register strategy* $\mathcal{S}$ is defined as follows.

- Alice and Bob share a state

$$|\psi\rangle = |r_1\rangle \otimes \cdots \otimes |r_k\rangle \otimes |\mathrm{aux}\rangle\,.$$

  Here, $|r_i\rangle = |\mathrm{EPR}_{q_i}^{n_i}\rangle$ for each i, and $|\mathrm{aux}\rangle$ is an arbitrary symmetric shared state.
- Given a question $x = (x_1, x_2)$ with first block $x_1 = (W_1, \ldots, W_k)$, Alice and Bob act as follows. Let $i \in [k]$.
  - If $W_i \in \{X, Z\}$, they measure τ^W on the i-th EPR register and set u_i to be the outcome.
  - If $W_i \in \{H, \perp\}$, they set $u_i = \varnothing$.

  Introduce the notation $\tau_\varnothing^W = I$ for $W \in \{H, \perp\}$. We can write their measurement as

$$M_{a_1}^{x_1,x_2} = \tau_{u_1}^{W_1} \otimes \cdots \otimes \tau_{u_k}^{W_k} \otimes I_{\mathsf{aux}}. \tag{37}$$

To produce the second part of their answer a_2, Alice and Bob can measure any part of their state except the EPR registers which have been "hidden". This entails the following: let $S = \{i \mid W_i = H\}$. Then for any answer a, the corresponding POVM acts as follows:

$$M_a^x = M_{\overline{S}} \otimes I_S. \tag{38}$$

Here, I_S is the identity matrix on the EPR registers in S, whereas $M_{\overline{S}}$ is a POVM acting on the EPR registers in $\overline{S}$ as well as the state $|\text{aux}\rangle$.

We define $\text{val}_{k,n,q}(\mathscr{G})$ to be the maximum over $\text{val}_{\mathscr{G}}(\mathcal{S})$, where $\mathcal{S}$ is any (k,n,q)-register strategy.

The X and Z questions specify the corresponding Pauli basis measurement, and the H question specifies that the register is to be hidden. The $\perp$ question is a "no-op" and does not restrict Alice and Bob at all, other than making them respond with the "no-op" answer $\varnothing$. Thus, unlike with the data hiding question, they are allowed to measure the register as they see fit. This will be useful later when we want Alice and Bob to measure both X *and* Z observables on the same register.

In designing our compiler, it will be convenient to define a set of strategies called "semiregister strategies". These will be strategies which are intermediate between $(k-1)$-register strategies and k-register strategies in the sense that they have Pauli basis queries implemented on the final (k-th) register but not data hiding queries. These are defined as follows.

Definition 5.2. A (k,n,q)-*semiregister strategy* is defined just as a (k,n,q)-register strategy, with the following modification: the set S used in Equation (38) is changed to be $S = \{i \neq k \mid W_i = H\}$. We define $\text{val}^{\text{semi}}_{k,n,q}(\mathscr{G})$ to be the maximum over $\text{val}_{\mathscr{G}}(\mathcal{S})$, where $\mathcal{S}$ is any (k,n,q)-semiregister strategy.

Thus, querying the k-th register of a semiregister strategy with a H is the same as querying it with a $\perp$.

The following lemma shows that we can restrict to *projective* register strategies without loss of generality.

Lemma 5.3. *Let $\mathcal{S}$ be a (k,n,q)-register strategy. Then there exists a (k,n,q)-register strategy $\mathcal{S}'$ in which all measurements are projective, and which produces the same bipartite correlation as $\mathcal{S}$.*

Proof. Start with the strategy $\mathcal{S}$, and let the measurements be denoted $M^{x_1,x_2}_{a_1,a_2}$. From the definition of register strategies, we know that for every set of questions x_1, x_2, the corresponding measurement can be written as a product

$$M^{x_1,x_2}_{a_1,a_2} = (\tau^{W_{x_1}}_{a_1})_S \otimes (A^{x_1,x_2,a_1}_{a_2})_{\overline{S}},$$

where S is the set of registers which receive a Pauli basis query in the set X, Z, H, $\overline{S}$ is its complement, and the operators $\{A^{x_1,x_2,a_1}_{a_2}\}$ form valid POVMs with outcomes a_2 for every choice of x_1, x_2, a_1. We will apply Naimark's theorem Theorem 4.1 using the universal auxiliary state $|\text{aux}\rangle$ to the A operator to produce projectors $A'^{x_1,x_2,a_1}_{a_2}$. Using these, we define a projective measurement

$$M'^{x_1,x_2}_{a_1,a_2} = (\tau^{W_{x_1}}_{a_1})_S \otimes (A'^{x_1,x_2,a_1}_{a_2})_{\overline{S}}.$$

It is not hard to see that M' and $|\text{aux}\rangle$ form a valid Naimark dilation of M. Let $\mathcal{S}'$ be the strategy $\mathcal{S}$ with the shared state $|\psi\rangle$ replaced by $|\psi\rangle \otimes |\text{aux}_A\rangle \otimes |\text{aux}_B\rangle$ and the measurements M replaced by M'. By construction, $\mathcal{S}'$ is a projective strategy. Further, from Corollary 4.8, it follows that the bipartite correlations produced by the strategies $\mathcal{S}'$ and $\mathcal{S}$ are the same. □

5.2 Results

The key elements of our compiler are two new nonlocal games called the Pauli basis test and the data hiding game. The Pauli basis test ensures that the provers share an EPR state and honestly answer Pauli basis queries to this state. The data hiding game allows us to "hide" this state from the provers, ensuring that they do not use this register unless we ask them to.

Our compiler operates a register at a time and involves two subroutines, $\mathcal{C}_{k\to\text{semi}}$ and $\mathcal{C}_{\text{semi}\to k-1}$. Given a k-register game, $\mathcal{C}_{k\to\text{semi}}$ produces a k-semiregister game. To do so, it removes the guarantee that the provers data hide the k-th register and replaces it by playing the data hiding game on this register. Thus, although the provers are no longer forced to hide the k-th register, they will have to do so anyway if they want to pass the data hiding game. Similarly, given a k-semiregister game, $\mathcal{C}_{\text{semi}\to k-1}$ produces a $(k-1)$-register game. To do so, it removes the guarantee that the provers have a k-th EPR register and replaces it by playing the Pauli basis test. Thus, by alternating these two subroutines, we can compile a k-register game into a 0-register game, i.e. a general game.

Before giving the properties of the Pauli basis compiler, we will need two definitions.

Definition 5.4. Given a string $x = (x_1, \ldots, x_k)$ and an integer $0 \le \ell \le k$, write $x|_\ell := (x_1, \ldots, x_\ell)$. We extend this to register parameters $\tau = (k, n, q)$ by setting $\tau|_\ell := (\ell, n|_\ell, q|_\ell)$. Thus, $\tau|_\ell$ is the register parameters for the first ℓ registers of τ.

Definition 5.5. Let n and q be integers and η be a real number. We say they *satisfy the Pauli basis condition* if

$$q = 2^t, \qquad \frac{1}{\text{poly}(n)} \le \eta \le \frac{1}{2}, \qquad \frac{64\log(n)^2}{\eta^2} \le q \le \text{poly}(n).$$

The following theorem describes the Pauli basis compiler.

Theorem 5.6. *Let $\lambda = (k, n, q)$, and let n_k, q_k, and η satisfy the Pauli basis condition. Suppose $\mathcal{G}_{\text{semi}}$ is a λ-semiregister game, and consider the $\lambda|_{k-1}$-register game $\mathcal{G}_{k-1} = \mathcal{C}_{\text{semi}\to(k-1)}(\mathcal{G}_{\text{semi}})$.*

- ***Completeness:*** *Suppose there is a value-1 λ-semiregister strategy for $\mathcal{G}_{\text{semi}}$ which is also a real commuting EPR strategy. Then there is a value-1 $\lambda|_{k-1}$-register strategy for $\mathcal{G}_{k-1}$ which is also a real commuting EPR strategy.*
- ***Soundness:*** *If $\text{val}_{\lambda|_{k-1}}(\mathcal{G}_{k-1}) \ge 1-\epsilon$ then $\text{val}^{\text{semi}}_{\lambda}(\mathcal{G}_{\text{semi}}) \ge 1-\delta(\epsilon)$, where $\delta(\epsilon) = \text{poly}(\epsilon, \eta)$.*

Furthermore,

$$\begin{aligned}
\mathsf{Q\text{-}time}(\mathcal{G}_{k-1}) &= \mathsf{Q\text{-}time}(\mathcal{G}_{\text{semi}}) + O(\log(n_k)),\\
\mathsf{Q\text{-}length}(\mathcal{G}_{k-1}) &= \mathsf{Q\text{-}length}(\mathcal{G}_{\text{semi}}) + O(\log(n_k)),\\
\mathsf{A\text{-}time}(\mathcal{G}_{k-1}) &= \mathsf{A\text{-}time}(\mathcal{G}_{\text{semi}}) + \text{poly}(n_k),\\
\mathsf{A\text{-}length}(\mathcal{G}_{k-1}) &= \mathsf{A\text{-}length}(\mathcal{G}_{\text{semi}}) + O(n_k \cdot \log\log(n_k)).
\end{aligned}$$

The following theorem describes the data hiding compiler.

Theorem 5.7. *Suppose $\mathcal{G}_k$ is a (k, n, q)-register game, and consider the (k, n, q)-semiregister game $\mathcal{G}_{\text{semi}} = \mathcal{C}_{k\to\text{semi}}(\mathcal{G}_k)$.*

- ***Completeness:*** *Suppose there is a value-1 (k, n, q)-register strategy for $\mathcal{G}_k$ which is also a real commuting EPR strategy. Then there is a value-1 (k, n, q)-semiregister strategy for $\mathcal{G}_{\text{semi}}$ which is also a real commuting EPR strategy.*

- ***Soundness:*** *If* $\mathrm{val}^{\mathrm{semi}}_{k,n,q}(\mathscr{G}_{\mathrm{semi}}) \geq 1-\epsilon$ *then* $\mathrm{val}_{k,n,q}(\mathscr{G}_k) \geq 1-\delta(\epsilon)$*, where* $\delta(\epsilon) = \mathrm{poly}(\epsilon)$*.*

Furthermore,

$$\mathsf{Q\text{-}time}(\mathscr{G}_{\mathrm{semi}}) = O(\mathsf{Q\text{-}time}(\mathscr{G}_k)), \quad \mathsf{A\text{-}time}(\mathscr{G}_{\mathrm{semi}}) = O(\mathsf{A\text{-}time}(\mathscr{G}_k)),$$

$$\mathsf{Q\text{-}length}(\mathscr{G}_{\mathrm{semi}}) = O(\mathsf{Q\text{-}length}(\mathscr{G}_k)), \quad \mathsf{A\text{-}length}(\mathscr{G}_{\mathrm{semi}}) = O(\mathsf{A\text{-}length}(\mathscr{G}_k)).$$

Combining Theorems 5.6 and 5.7 gives us the main result of Part III, a compiler $\mathcal{C}$ which compiles k-register games into general games.

Theorem 5.8. *Let* $\mathscr{G}_k$ *be a* (k,n,q)*-register game. Let* $\eta = (\eta_1, \ldots, \eta_k)$*, and suppose* n_i*,* q_i*, and* η_i *pass the Pauli basis condition for all* $i \in [k]$*. Write*

$$\mathscr{G} = \mathcal{C}(\mathscr{G}_k) := \mathcal{C}_{\mathrm{semi}\to 0}(\mathcal{C}_{1\to\mathrm{semi}}(\ \cdots\ \mathcal{C}_{\mathrm{semi}\to k-1}(\mathcal{C}_{\mathrm{k}\to\mathrm{semi}}(\mathscr{G}_k)))).$$

- ***Completeness:*** *Suppose there is a value-*1 (k,n,q)*-register strategy for* $\mathscr{G}_k$ *which is also a real commuting EPR strategy. Then there is a real commuting EPR strategy for* $\mathscr{G}$ *with value* 1*.*

- ***Soundness:*** *If* $\mathrm{val}(\mathscr{G}) \geq 1-\epsilon$ *then* $\mathrm{val}_{(k,n,q)}(\mathscr{G}_k) \geq 1-\delta(\epsilon)$*, where* $\delta(\epsilon) = \mathrm{poly}(\epsilon, \eta_1, \ldots, \eta_k)$*.*

Furthermore,

$$\begin{aligned}
\mathsf{Q\text{-}time}(\mathscr{G}) &= \mathsf{Q\text{-}time}(\mathscr{G}_k) + O(\log(n_1)) + \cdots + O(\log(n_k)),\\
\mathsf{Q\text{-}length}(\mathscr{G}) &= \mathsf{Q\text{-}length}(\mathscr{G}_k) + O(\log(n_1)) + \cdots + O(\log(n_k)),\\
\mathsf{A\text{-}time}(\mathscr{G}) &= \mathsf{A\text{-}time}(\mathscr{G}_k) + \mathrm{poly}(n_1) + \cdots + \mathrm{poly}(n_k),\\
\mathsf{A\text{-}length}(\mathscr{G}) &= \mathsf{A\text{-}length}(\mathscr{G}_k) + O(n_1 \cdot \log\log(n_1)) + \cdots O(n_k \cdot \log\log(n_k)).
\end{aligned}$$

5.3 Registers for uniform games

In this section, we generalize the notion of registers to the case of uniform games, in which a different set of register parameters might be used for each input. To compile these games, we will need for the register parameters themselves to be uniformly generated.

Definition 5.9. Let M_{Params} be a Turing machine which, given an input input, outputs $\lambda = (k,n,q)$. Let $\mathscr{G}$ be a (nonuniform) game. Then we say M_{Params} *outputs the register parameters of* $\mathscr{G}$ if for every input, $\mathscr{G}(\mathsf{input})$ is a $M_{\mathrm{Params}}(\mathsf{input})$-register game.

Given this, our compiler for uniform games is given as follows.

Corollary 5.10. *Let* $\mathscr{G}(\cdot)$ *be a (uniform) game, and let* M_{Params} *be a Turing machine which outputs its register parameters. Then there exists a (uniform) game* $\mathscr{G}_{\mathrm{Compile}}(\cdot)$ *with the following properties. Given an input* input*, write* $\mathscr{G} := \mathscr{G}(\mathsf{input})$*,* $\mathscr{G}_{\mathrm{Compile}} := \mathscr{G}_{\mathrm{Compile}}(\mathsf{input})$*, and* $\lambda = (k,n,q) := M_{\mathrm{Params}}(\mathsf{input})$*.*

- ***Completeness:*** *Suppose there is a value-*1 (k,n,q)*-register strategy for* $\mathscr{G}$ *which is also a real commuting EPR strategy. Then there is a real commuting EPR strategy for* $\mathscr{G}_{\mathrm{Compile}}$ *with value* 1*.*

- ***Soundness:*** *Let* $\eta = (\eta_1, \ldots, \eta_k)$*, and suppose* n_i*,* q_i*, and* η_i *pass the Pauli basis condition for all* $i \in [k]$*. If* $\mathrm{val}(\mathscr{G}_{\mathrm{Compile}}) \geq 1-\epsilon$ *then* $\mathrm{val}_{\lambda}(\mathscr{G}) \geq 1-\delta(\epsilon)$*, where* $\delta(\epsilon) = \mathrm{poly}(\epsilon, \eta_1, \ldots, \eta_k)$*.*

Furthermore,

$$\begin{aligned}
\mathsf{Q\text{-}time}(\mathscr{G}) &= \mathsf{Q\text{-}time}(\mathscr{G}_k) + O(\log(n_1)) + \cdots + O(\log(n_k)) + \mathsf{time}(M_{\mathrm{Params}}(\mathsf{input})),\\
\mathsf{Q\text{-}length}(\mathscr{G}) &= \mathsf{Q\text{-}length}(\mathscr{G}_k) + O(\log(n_1)) + \cdots + O(\log(n_k)),\\
\mathsf{A\text{-}time}(\mathscr{G}) &= \mathsf{A\text{-}time}(\mathscr{G}_k) + \mathrm{poly}(n_1) + \cdots + \mathrm{poly}(n_k) + \mathsf{time}(M_{\mathrm{Params}}(\mathsf{input})),\\
\mathsf{A\text{-}length}(\mathscr{G}) &= \mathsf{A\text{-}length}(\mathscr{G}_k) + O(n_1 \cdot \log\log(n_1)) + \cdots O(n_k \cdot \log\log(n_k)).
\end{aligned}$$

Proof. We first compute $\lambda = M_{\mathrm{Params}}(\mathsf{input})$ in time $\mathsf{time}(M_{\mathrm{Params}}(\mathsf{input}))$. Then it can be checked that the compiled game $\mathcal{C}(\mathscr{G}(\mathsf{input}))$ from Theorem 5.8 can be efficiently simulated given the register parameters λ. □

5.4 Organization

The remainder of Part III is organized as follows.

- In Section 6, we introduce the Pauli basis self-test and prove its correctness.
- Section 7 implements the Pauli basis compiler.
- In Section 8, we introduce the data hiding game.
- Section 9 implements the data hiding compiler.
- Section 10 contains a generalization of the data hiding game which allows us to hide more general sets of Pauli observables. This is not needed to implement the quantum registers, but it will be needed in Part IV when designing the NEEXP protocol.

6 A self test for the Pauli basis

In this section, we give a self test for the Pauli basis measurement. Given $W \in \{X, Z\}$, this test compels the prover to measure an EPR register in the W basis and return the outcome to the verifier.

Definition 6.1. The *Pauli basis strategy with parameters n and q* (a prime power), denoted $\mathrm{Pauli}(n, q)$, is the partial strategy with the state $|\mathrm{EPR}_q^n\rangle$ and measurement matrices τ_u^W for each $W \in \{X, Z\}, u \in \mathbb{F}_q^n$.

The main result of this section is the following self-test for the case when q is a power of 2.

Theorem 6.2. *Let $\boldsymbol{W} \sim \{X, Z\}$ uniformly at random. Let n, q, η satisfy the Pauli basis condition. Then there is a self-test $\mathscr{G}_{\mathrm{basis}} := \mathscr{G}_{\mathrm{basis}}(n, q)$ for $\mathrm{Pauli}(n, q)$ over $\boldsymbol{W}$ with robustness $\delta(\epsilon) = \mathrm{poly}(\epsilon, \eta)$. Moreover, there is a value-1 real commuting EPR strategy with auxiliary state $|\mathrm{EPR}_2\rangle$. Finally,*

$$\mathsf{Q\text{-}length}(\mathscr{G}_{\mathrm{basis}}) = O\left(\log(n)\right), \quad \mathsf{A\text{-}length}(\mathscr{G}_{\mathrm{basis}}) = \mathrm{poly}(n),$$

$$\mathsf{Q\text{-}time}(\mathscr{G}_{\mathrm{basis}}) = O\left(\log(n)\right), \quad \mathsf{A\text{-}time}(\mathscr{G}_{\mathrm{basis}}) = \mathrm{poly}(n).$$

We prove this by a straightforward reduction to the quantum low-degree test of [NV18a].

6.1 The quantum low-degree test

The goal of the quantum low-degree test of [NV18a] is to force the provers to use a "compressed" version of the Pauli basis strategy. Given $\boldsymbol{W}$, they should measure their register in the $\boldsymbol{W}$ basis, receiving $\boldsymbol{u} \in \mathbb{F}_q^n$. However, $\boldsymbol{u}$, a length-n string, might be prohibitively expensive to communicate to the verifier, so they should instead compute the low degree encoding $g_{\boldsymbol{u}}$ and return its evaluation at a single point $\boldsymbol{w} \in \mathbb{F}_q^m$ of the verifier's choosing. (The point of this section is to "uncompress" their protocol.)

Definition 6.3. Fix parameters for the low-degree encoding $\mathsf{params} := (q = p^t, h, H, m, n, \pi)$ satisfying the "low-degree conditions" $h \leq q$, and $n \leq h^m$. For any string $u \in \mathbb{F}_q^n$, these parameters give a low-degree encoding $g_u : \mathbb{F}_q^m \to \mathbb{F}_q$.

The *low-degree Pauli strategy with parameters* params, denoted $\mathcal{LD}(\mathsf{params})$, is the partial strategy with state $|\mathrm{EPR}_q^n\rangle$ and measurement matrices

$$\tau_a^{W,w} := \tau^W_{[g_u(w)=a]} = \sum_{u:g_u(w)=a} \tau_u^W$$

for each $W \in \{X, Z\}, w \in \mathbb{F}_q^m, a \in \mathbb{F}_q$. Equivalently, this is the strategy where we perform the Pauli W-basis measurement and output the low-degree encoding of the outcome u evaluated at the point w, i.e. the value $g_u(w)$.

The main result of [NV18a] is the following.

Theorem 6.4 ([NV18a, Theorem 3.2])**.** *Fix low-degree parameters* params *with* $p = 2$ *(so that* $q = 2^t$*) and* $m \geq 2$*, and let* $\mathcal{D}$ *be the uniform distribution over* (W, w) *with* $W \in \{X, Z\}, w \in \mathbb{F}_q^m$*. Then there is a self-test* $\mathscr{G}_{\mathrm{Qlowdeg}} := \mathscr{G}_{\mathrm{Qlowdeg}}(\mathsf{params})$ *for* $\mathcal{LD}(\mathsf{params})$ *over* $\mathcal{D}$ *with robustness* $\delta(\epsilon) = \mathrm{poly}(\epsilon, md/q^c)$*, with* $c > 0$*. Moreover, there is a value-*1 *real commuting EPR strategy with auxiliary state* $|\mathrm{EPR}_2\rangle$*. Finally,*

$$\mathsf{Q\text{-}length}(\mathscr{G}_{\mathrm{Qlowdeg}}) = O(m \log q), \quad \mathsf{A\text{-}length}(\mathscr{G}_{\mathrm{Qlowdeg}}) = O(d^2 \log(q)),$$

$$\mathsf{Q\text{-}time}(\mathscr{G}_{\mathrm{Qlowdeg}}) = O(m \log q), \quad \mathsf{A\text{-}time}(\mathscr{G}_{\mathrm{Qlowdeg}}) = \mathrm{poly}(m, d, \log q).$$

(We note that this result is stated in [NV18a] for general primes p. However, the $p \neq 2$ case relied on a self-testing result for a generalization of the Magic Square game which was recently discovered to contain a bug. Fortunately, the $p = 2$ case needs only a self-testing result for the "traditional" binary Magic Square game, and this follows from [WBMS16].)

Remark 6.5. We note that the quantum low-degree test, as stated in [NV18a], does *not* have value-1 real commuting EPR strategies. This is because it uses as a subroutine the standard magic square game, and the magic square game does not have value-1 real commuting EPR strategies. Its value-1 strategies *are* EPR strategies, and they *are* real (all observables are either X or Z, with the sole exception of the $Y \otimes Y$ observable, which can be rewritten as $Y \otimes Y = -(X \otimes X) \cdot (Z \otimes Z)$, manifestly real). But they are not commuting, because each row and column have at least one pair of noncommuting observables.

Consider instead the following "oracularized" version of the magic square game: one player is given a random row or column (and is expected to play as in the normal magic square game), and the other player is given a random cell in that row or column, and the verifier simply checks that they agree on that cell. In addition, with some constant probability, both players are given the same cell and their answers are checked against each other. In this case, all observables measured

With probability $\frac{1}{2}$ each, perform one of the following two tests.

1. **Low-degree:** Perform $\mathscr{G}_{\text{Qlowdeg}}(\mathsf{params})$.
2. **Cross-check:** Draw $\boldsymbol{W} \sim \{X, Z\}$, $\boldsymbol{w} \sim \mathbb{F}_q^m$. Flip an unbiased coin $\boldsymbol{b} \sim \{0, 1\}$. Distribute the questions as follows:
   - Player $\boldsymbol{b}$: Give $\boldsymbol{W}$; receive $\boldsymbol{u} \in \mathbb{F}_q^n$.
   - Player $\overline{\boldsymbol{b}}$: give $(\boldsymbol{W}, \boldsymbol{w})$; receive $\boldsymbol{a}$.

   Accept if $g_{\boldsymbol{u}}(\boldsymbol{w}) = \boldsymbol{a}$.

Figure 1: The game $\mathscr{G}_{\text{basis}}(n, q)$.

are commuting, and so this variant has a value-1 real commuting EPR strategy. In addition, it certifies the same state and measurements as the normal magic square game, and so we can use it as a subroutine in the quantum low-degree test instead.

Remark 6.6. We note again that the soundness case in our definition of a self-test is quite different from the one given in [NV18a, Definition 2.5], and it is not clear that a self-test in their sense implies a self-test in our sense. However, for the quantum low-degree test, their soundness case *does* match ours. By [NV18a, Lemma 4.1], there is a local isometry $\phi = \phi_1 \otimes \phi_2$ such that

$$\|\phi\,|\overline{\psi}\rangle - |\psi\rangle\,|\text{aux}\rangle\,\|^2 \leq \delta(\epsilon). \tag{39}$$

and

$$\mathop{\mathbf{E}}_{(\boldsymbol{W},\boldsymbol{w})} \sum_a \|\phi \cdot (\overline{M}_a^{\boldsymbol{W},\boldsymbol{w}} \otimes I_{\mathsf{Bob}})\,|\overline{\psi}\rangle - (\tau_a^{\boldsymbol{W},\boldsymbol{w}} \otimes I_{\mathsf{aux}}) \otimes I_{\mathsf{Bob}}\,|\psi\rangle\,|\text{aux}\rangle\,\|^2 \leq \delta(\epsilon). \tag{40}$$

The key difference from our self-test definition is that, as stated, their local isometry need not be symmetric (i.e. $\phi_1 \neq \phi_2$), but their construction actually *does* give a symmetric isometry with $\phi_1 = \phi_2$. Then, from Equation (40) it is easy to derive Equation (36) using Equation (40) and the triangle inequality (Fact 4.28).

6.2 Proof of Theorem 6.2: the Pauli basis test

We now state the Pauli basis test.

Definition 6.7. Let n, q, η be as in Theorem 6.2. Fix the remaining low-degree parameters params as follows:

$$h = \lceil q^{1/2} \rceil, \qquad m = 2 \cdot \left\lceil \frac{\log(n)}{\log(q)} \right\rceil, \qquad d = m \cdot (h - 1).$$

Then the *Pauli basis game* $\mathscr{G}_{\text{basis}}(n, q)$ is given by Figure 1.

These parameters are chosen so that they are valid low-degree parameters (guaranteeing the existence of the low-degree code), which is necessary for the quantum low-degree test. In particular, these satisfy (i) $h \leq q$ and (ii) $n \leq h^m$. The first of these is immediate; as for the second,

$$h^m \geq (q^{1/2})^{2 \cdot \log(n)/\log(q)} = q^{\log(n)/\log(q)} = n.$$

In addition, the code has relative distance $d/q \leq mh/q \leq \eta$.

$$\frac{d}{q} = \frac{m \cdot (h-1)}{q} \leq \frac{mh}{q} \leq 8 \cdot \frac{\log(n)}{\log(q)} \cdot \frac{q^{1/2}}{q} \leq \frac{8\log(n)}{q^{1/2}} \leq \eta,$$

where the final step is because n, q, η satisfy the Pauli basis condition. Finally, we note that even if q is a large polynomial of n, m is always at least 2, which permits us to use the quantum low-degree test. We now prove Theorem 6.2.

Proof of Theorem 6.2. The question lengths and times of both the quantum low-degree test and the cross-check are given by

$$m \log(q) = 2 \cdot \left\lceil \frac{\log(n)}{\log(q)} \right\rceil \cdot \log(q) = O(\log(n)).$$

As for the answer lengths and times, these are bounded by $\mathrm{poly}(n)$ for both the quantum low-degree test and the cross-check. We now consider the completeness and soundness cases separately.

Completeness. Let (ψ, M) be the value-1 commuting EPR strategy for the quantum low-degree test guaranteed by Theorem 6.4. This has state $|\psi\rangle = |\mathrm{EPR}_q^n\rangle\,|\mathrm{EPR}_2\rangle$ and measurement matrices $M_a^{W,w} = \tau_a^{W,w} \otimes I_{\mathsf{aux}}$. If we add in the measurement matrices $M_u^W = \tau_u^W \otimes I_{\mathsf{aux}}$, then this strategy passes the cross-check with probability 1. This is because after Player $\boldsymbol{b}$ measures $\boldsymbol{u}$, the state collapses to $|\tau_{\boldsymbol{u}}^{\boldsymbol{W}}\rangle\,|\tau_{\boldsymbol{u}}^{\boldsymbol{W}}\rangle\,|\mathrm{EPR}_2\rangle$, and so Player $\overline{\boldsymbol{b}}$ will measure $\boldsymbol{a} = g_{\boldsymbol{u}}(\boldsymbol{w})$. As a result, this is a value-1 strategy. Furthermore, it is a commuting EPR strategy because the cross-check measurements M^W and $M^{W,w}$ commute. Finally, this strategy extends the Pauli basis strategy. This proves the completeness case.

Soundness. Throughout the soundness, we will use $\delta(\epsilon)$ to denote a function of the form $\mathrm{poly}(\epsilon, \eta)$ which may change from use to use. The $\delta(\epsilon)$ in Theorem 6.4 is of this form because $d/q \le \eta$.

Let $\overline{\mathcal{S}} = (\overline{\psi}, \overline{M})$ be a strategy with $\mathrm{val}_{\mathscr{G}_{\mathrm{basis}}}(\overline{\mathcal{S}}) = 1 - \epsilon$. Then this strategy must pass $\mathscr{G}_{\mathrm{Qlowdeg}}$ with probability at least $1 - 2\epsilon$. By Theorem 6.4 this gives us a local isometry $\phi = \phi_{\mathrm{local}} \otimes \phi_{\mathrm{local}}$ and a state $|\mathrm{aux}\rangle$ with the following properties: if we define the new strategy $\mathcal{S}$ in which $|\psi\rangle = \phi\,|\overline{\psi}\rangle$ and $M_a^x = \phi_{\mathrm{local}} \cdot \overline{M}_a^x \cdot \phi_{\mathrm{local}}^\dagger$, then

$$\| \,|\psi\rangle - |\mathrm{EPR}_q^n\rangle\,|\mathsf{aux}\rangle\,\|^2 \le \delta(\epsilon), \qquad M_a^{W,w} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} (\tau_a^{W,w} \otimes I_{\mathsf{aux}}) \otimes I_{\mathsf{Bob}}, \tag{41}$$

on state $|\psi\rangle$ and distribution $\mathcal{D}$. Because $\mathcal{S}$ is just a rotated version of $\overline{\mathcal{S}}$, it also passes $\mathscr{G}_{\mathrm{basis}}$ with probability $1 - \epsilon$. As a result, $\mathcal{S}$ passes the cross-check in Section 6.2 with probability at least $1 - 2\epsilon$. By Fact 4.13, we conclude that

$$M_{[g_u(w)=a]}^W \otimes I_{\mathsf{Bob}} \approx_\epsilon I_{\mathsf{Alice}} \otimes M_a^{W,w} \approx_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes (\tau_a^{W,w} \otimes I_{\mathsf{aux}}) = I_{\mathsf{Alice}} \otimes (\tau_{[g_v(w)=a]}^W \otimes I_{\mathsf{aux}}) \tag{42}$$

on state $|\psi\rangle$. By Fact 4.14 and the fact that the τ measurements are projective, this implies that

$$M_{[g_u(w)=a]}^W \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes (\tau_{[g_v(w)=a]}^W \otimes I_{\mathsf{aux}})$$

Now by Proposition 4.42 (where we let $\boldsymbol{s}$ be the singleton distribution on the "trivial" subspace $\boldsymbol{s} = \mathbb{F}_q^m$) and the fact that $d/q \le \eta$, we can conclude that

$$M_u^W \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes (\tau_u^W \otimes I_{\mathsf{aux}}).$$

Applying Fact 4.13 again, this yields

$$M_u^W \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes (\tau_u^W \otimes I_{\mathsf{aux}}) \approx_{\delta(\epsilon)} (\tau_u^W \otimes I_{\mathsf{aux}}) \otimes I_{\mathsf{Bob}} \tag{43}$$

on state $|\psi\rangle$, where the last step uses Fact 4.22 to combine Fact 4.38 with Equation (41). The analogous statement for the state $|\mathrm{EPR}_q^n\rangle$ follows from Fact 4.22. This establishes the theorem. □

Flip an unbiased coin $\boldsymbol{b} \sim \{0,1\}$. With probability $\frac{1}{4}$ each, perform one of the following four tests.

1. **Pauli basis:** Draw $(\boldsymbol{x}, \boldsymbol{x}') \sim \mathscr{G}_{\text{basis}}(n_k, q_k, \eta)$. Distribute the questions as follows:
   - Player $\boldsymbol{b}$: give $(H^{k-1}, \boldsymbol{x})$; receive $\boldsymbol{a} = (\boldsymbol{a}_1, \boldsymbol{a}_2)$.
   - Player $\overline{\boldsymbol{b}}$: give $(H^{k-1}, \boldsymbol{x}')$; receive $\boldsymbol{a}' = (\boldsymbol{a}'_1, \boldsymbol{a}'_2)$.

   Accept if $\boldsymbol{a}_2$ and $\boldsymbol{a}'_2$ are accepting answers to the Pauli basis test.
2. **Cross-check:** Draw $(\boldsymbol{x}, \boldsymbol{x}') \sim \mathscr{G}_{\text{semi}}$. Write $\boldsymbol{x} = (\boldsymbol{x}_1, \boldsymbol{x}_2)$ with $\boldsymbol{x}_1 = (\boldsymbol{W}_1, \ldots, \boldsymbol{W}_k)$. Distribute the questions as follows:
   - Player $\boldsymbol{b}$: give $\boldsymbol{x}$; receive $\boldsymbol{a} = (\boldsymbol{a}_1, \boldsymbol{a}_2)$, where $\boldsymbol{a}_1 = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_k)$.
   - Player $\overline{\boldsymbol{b}}$: give $(H^{k-1}, \boldsymbol{W}_k)$; receive strings $\boldsymbol{a}'_1 = (\boldsymbol{u}'_1, \ldots, \boldsymbol{u}'_k)$, $\boldsymbol{u}'_i \in \mathbb{F}_q^n$.

   If $\boldsymbol{W}_k \in \{X, Z\}$, accept if $\boldsymbol{u}_k = \boldsymbol{u}'_k$. Otherwise, accept if $\boldsymbol{u}_k = \varnothing$.
3. **Consistency check:** Draw $(\boldsymbol{x}, \boldsymbol{x}') \sim \mathscr{G}_{\text{semi}}$. Distribute the questions as follows:
   - Player $\boldsymbol{b}$: give $\boldsymbol{x}$; receive $\boldsymbol{a}$
   - Player $\overline{\boldsymbol{b}}$: give $\boldsymbol{x}$; receive $\boldsymbol{a}'$.

   Accept if $\boldsymbol{a} = \boldsymbol{a}'$.
4. **Play game:** Perform $\mathscr{G}_{\text{semi}}$.

Figure 2: The game $\mathcal{C}_{\text{semi}\to(k-1)}(\mathscr{G}_{\text{semi}})$.

7 Compiling games with the Pauli basis test

In this section, we show how to use the Pauli basis test to implement the compiler $\mathcal{C}_{\text{semi}\to(k-1)}$. Our construction is given in the following definition.

Definition 7.1. Let $\mathscr{G}_{\text{semi}}$ be a (k, n, q)-semiregister game. Then its compiled version is the game $\mathcal{C}_{\text{semi}\to(k-1)}(\mathscr{G}_{\text{semi}})$ defined in Figure 2.

In words, the provers might try to "trick" the verifier by using one of their $(k-1)$ existing EPR registers to answer queries meant for the new k-th register. To prevent this, the verifier performs the Pauli basis test with the first $k-1$ registers hidden, forcing the provers to introduce a new EPR register. It then cross-checks the provers' answers in the Pauli basis test with their answers in the game $\mathscr{G}_{\text{semi}}$. The performance of the compiler is given by the following theorem.

Theorem 7.2. *Let $\lambda = (k, n, q)$, and let n_k, q_k, and η satisfy the Pauli basis condition. Suppose $\mathscr{G}_{\text{semi}}$ is a λ-semiregister game, and consider the $\lambda|_{k-1}$-register game $\mathscr{G}_{k-1} = \mathcal{C}_{\text{semi}\to(k-1)}(\mathscr{G}_{\text{semi}})$.*

- ***Completeness:*** *Suppose there is a value-1 λ-semiregister strategy for $\mathscr{G}_{\text{semi}}$ which is also a real commuting EPR strategy. Then there is a value-1 $\lambda|_{k-1}$-register strategy for $\mathscr{G}_{k-1}$ which is also a real commuting EPR strategy.*
- ***Soundness:*** *If $\text{val}_{\lambda|_{k-1}}(\mathscr{G}_{k-1}) \geq 1 - \epsilon$ then $\text{val}^{\text{semi}}_{\lambda}(\mathscr{G}_{\text{semi}}) \geq 1 - \delta(\epsilon)$, where $\delta(\epsilon) = \text{poly}(\epsilon, \eta)$.*

Furthermore,

$$\begin{aligned}
\mathsf{Q\text{-}length}(\mathscr{G}_{k-1}) &= \mathsf{Q\text{-}length}(\mathscr{G}_{\text{semi}}) + O(\log(n)),\\
\mathsf{Q\text{-}time}(\mathscr{G}_{k-1}) &= \mathsf{Q\text{-}time}(\mathscr{G}_{\text{semi}}) + O(\log(n)),\\
\mathsf{A\text{-}length}(\mathscr{G}_{k-1}) &= \mathsf{A\text{-}length}(\mathscr{G}_{\text{semi}}) + \text{poly}(n),\\
\mathsf{A\text{-}time}(\mathscr{G}_{k-1}) &= \mathsf{A\text{-}time}(\mathscr{G}_{\text{semi}}) + \text{poly}(n).
\end{aligned}$$

Proof. The communication and time complexities are the result of combining the communication and time complexities from $\mathscr{G}_{\text{semi}}$ with the values for the Pauli basis test from Theorem 6.2.

Completeness. Let (ψ, M) be a value-1 λ-semiregister strategy for $\mathscr{G}_{\text{semi}}$ which is also a real commuting EPR strategy. Then $|\psi\rangle = |r_1\rangle \cdots |r_k\rangle\,|\text{aux}\rangle$, where each $|r_i\rangle = |\text{EPR}_{q_i}^{n_i}\rangle$ and $|\text{aux}\rangle$ is an EPR state. In addition, let (ψ', M') be the value-1 real commuting EPR strategy for $\mathscr{G}_{\text{basis}}$ guaranteed by Theorem 6.2. Then $|\psi'\rangle = |\text{EPR}_{q_k}^{n_k}\rangle\,|\text{aux}'\rangle$, where $|\text{aux}'\rangle$ is an EPR state.

Consider the following strategy for $\mathscr{G}_{k-1}$. For its state, it uses $|r_1\rangle \cdots |r_k\rangle\,|\text{aux}\rangle\,|\text{aux}'\rangle$. For inputs drawn from $\mathscr{G}_{\text{semi}}$, it uses the matrices in M applied to all but the $|\text{aux}'\rangle$ register. For inputs of the form (H^{k-1}, x), where x is sampled from $\mathscr{G}_{\text{basis}}$, it outputs $\varnothing^{k-1}$ along with the result of applying M' to $|r_k\rangle$ and $|\text{aux}'\rangle$. Finally, for inputs of the form (H^{k-1}, H) and $(H^{k-1}, \bot)$, it outputs $\varnothing^k$. This forms a valid $\lambda|_{k-1}$-register strategy for $\mathscr{G}_{k-1}$. In addition, its "auxiliary register" is $|r_k\rangle\,|\text{aux}\rangle\,|\text{aux}'\rangle$, which is an EPR state. Now we show that it has value 1.

By construction, this strategy passes the Pauli basis test and $\mathscr{G}_{\text{semi}}$ with probability 1. As for the cross-check, when $\boldsymbol{W}_k \in \{H, \bot\}$, the strategy always succeeds because (ψ, M) is a λ-semiregister strategy. On the other hand, when $\boldsymbol{W}_k \in \{X, Z\}$, this implies that $\boldsymbol{u}_k$ is the result of applying the $\tau^{\boldsymbol{W}_k}$ measurement to $|r_k\rangle$, putting it in state $|\tau_{\boldsymbol{u}_k}^{\boldsymbol{W}_k}\rangle\,|\tau_{\boldsymbol{u}_k}^{\boldsymbol{W}_k}\rangle$. But then because (ψ', M') implements the Pauli basis strategy on $|r_k\rangle$, the outcome $\boldsymbol{u}_k'$ is also the result of applying the $\tau^{\boldsymbol{W}_k}$ measurement to $|r_k\rangle$. As a result, $\boldsymbol{u}_k' = \boldsymbol{u}_k$.

Finally, it is clear that this forms an EPR strategy. As a result, by Fact 4.37, the consistency check passes with probability 1. Thus, the strategy passes the overall test with probability 1. Next, we show that this gives a *commuting* EPR strategy. For the questions that arise in the Pauli basis test, the consistency check, and $\mathscr{G}_{\text{semi}}$, commutation follows because M and M' are commuting. As for the cross-check, consider the case when $\boldsymbol{W}_k \in \{X, Z\}$. Then the first (i.e. Player $\boldsymbol{b}$'s) measurement is given by

$$(M_{a_1,a_2}^{x_1,x_2})_{1,\ldots,\mathsf{k},\mathsf{aux}} \otimes I_{\mathsf{aux}'} = \tau_{u_k}^{W_k} \otimes (A_{u_1,\ldots,u_{k-1},a_2}^{x_1,x_2})_{1,\ldots,\mathsf{k}-1,\mathsf{aux},\mathsf{aux}'},$$

where A is some measurement. This follows because M is a λ-semiregister strategy. Similarly, the second (i.e., Player $\overline{\boldsymbol{b}}$'s) measurement is given by

$$(\tau_{\varnothing}^{H} \otimes \cdots \otimes \tau_{\varnothing}^{H})_{1,\ldots,\mathsf{k}-1} \otimes (M_{u_k'}'^{W_k})_{\mathsf{k},\mathsf{aux}'} \otimes I_{\mathsf{aux}} = \tau_{u_k'}^{W_k} \otimes I_{1,\ldots,\mathsf{k}-1,\mathsf{aux},\mathsf{aux}'}.$$

By inspection, these two commute. On the other hand, when $\boldsymbol{W}_k \in \{H, \bot\}$, then Player $\overline{\boldsymbol{b}}$ always outputs $\varnothing^k$. Their measurement for this outcome is the matrix $I_{1,\ldots,\mathsf{k},\mathsf{aux},\mathsf{aux}'}$, and is the zero matrix for every other outcome. These clearly commute with any strategy for Player $\boldsymbol{b}$.

Finally, because M and M' are real strategies, this strategy is also real. As a result, this gives a value-1 real commuting EPR strategy.

Soundness. Suppose $\mathcal{S}_{\text{reg}} = (\psi_{\text{reg}}, M_{\text{reg}})$ is a $\lambda|_{k-1}$-register strategy for $\mathscr{G}_{k-1}$ with value $1 - \epsilon$. By Lemma 5.3, we can assume without loss of generality that M is projective. For $1 \le i \le k$, write $|r_i\rangle := |\text{EPR}_{q_i}^{n_i}\rangle$. By definition, $|\psi_{\text{reg}}\rangle = |r_1\rangle \otimes \cdots \otimes |r_{k-1}\rangle \otimes |\text{aux}_{\text{reg}}\rangle$. Our goal will be to decode $\mathcal{S}_{\text{reg}}$ into a λ-semiregister strategy $\mathcal{S}_{\text{semi}}$ for $\mathscr{G}_{\text{semi}}$ with nearly the same value.

Using the Pauli basis test. Passing the overall test with probability $1-\epsilon$ means that $\mathcal{S}_{\text{reg}}$ must pass the test in Item 1 with probability $1-4\epsilon$. This test only involves measurements of the form $\{(M_{\text{reg}})_{a_1,a_2}^{H,\dots,H,x}\}_{a_1,a_2}$. Because the first $k-1$ coordinates are hidden, Equation (38) allows us to write

$$(M_{\text{reg}})_{a_2}^{H,\dots,H,x} = I_{1,\dots,\mathsf{k}-1} \otimes (A_{a_2}^x)_{\mathsf{aux}},$$

where $\{A_a^x\}_x$ is some set of measurements. As a result, the state $|\text{aux}_{\text{reg}}\rangle$ and measurements $\{A_a^x\}_x$ form a strategy for the game $\mathscr{G}_{\text{basis}}(n_k, q_k, \eta)$ which succeeds with probability $1-4\epsilon$. By Theorem 6.2 this gives us a local isometry $\phi = \phi_{\text{local}} \otimes \phi_{\text{local}}$ and a state $|\text{aux}\rangle$ such that

$$\|\phi\,|\text{aux}_{\text{reg}}\rangle - |r_k\rangle\,|\text{aux}\rangle\,\|^2 \le \delta(\epsilon), \tag{44}$$

$$(\phi_{\text{local}} \cdot A_u^W \cdot \phi_{\text{local}}^\dagger)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} (\tau_u^W \otimes I_{\mathsf{aux}})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}, \tag{45}$$

on state $|r_k\rangle\,|\text{aux}\rangle$ and the uniform distribution on $\{X, Z\}$.

Define the new strategy $\mathcal{S}$ in which $|\psi\rangle = |r_1\rangle \otimes \cdots \otimes |r_{k-1}\rangle \otimes (\phi\,|\text{aux}_{\text{reg}}\rangle)$ and

$$M_a^x = (I_{1,\dots,\mathsf{k}-1} \otimes (\phi_{\text{local}})_{\mathsf{aux}}) \cdot (M_{\text{reg}})_a^x \cdot (I_{1,\dots,\mathsf{k}-1} \otimes (\phi_{\text{local}}^\dagger)_{\mathsf{aux}}).$$

Then Equations (44) and (45) implies that

$$\|\,|\psi\rangle - |r_1\rangle \otimes \cdots \otimes |r_k\rangle\,|\text{aux}\rangle\,\|^2 \le \delta(\epsilon), \tag{46}$$

$$(M_u^{H,\dots,H,W})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} (I_{1,\dots,\mathsf{k}-1} \otimes \tau_u^W \otimes I_{\mathsf{aux}})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}, \tag{47}$$

on state $|\psi\rangle$ and the uniform distribution on $\{X, Z\}$. Because $\mathcal{S}$ is just a rotated version of $\mathcal{S}_{\text{reg}}$, it also passes $\mathscr{G}_{k-1}$ with probability $1-\epsilon$. In addition, it is also a $\lambda|_{k-1}$-register strategy.

Performing the cross-check. To analyze the cross-check, we begin with a definition. Given $W \in \{X, Z, H, \perp\}$ and $u \in \mathbb{F}_{q_k}^{n_k} \cup \{\varnothing\}$, define $\text{null}_W(u) = u$ if $W \in \{X, Z\}$ and $\varnothing$ otherwise. The cross-check in Item 2 checks equality between $\boldsymbol{u}_k$ and $\text{null}_{\boldsymbol{W}_k}(\boldsymbol{u}'_k)$. As a result,

$$(M_{u_k}^x)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_\epsilon I_{\mathsf{Alice}} \otimes (M_{[\text{null}_{W_k}(u'_k)=u_k]}^{H,\dots,H,W_k})_{\mathsf{Bob}}.$$

Next, we note that when $W_k \in \{H, \perp\}$,

$$M_{[\text{null}_{W_k}(u'_k)=u_k]}^{H,\dots,H,W_k} = I_{1,\dots,\mathsf{k}-1} \otimes \tau_{u_k}^{W_k} \otimes I_{\mathsf{aux}},$$

because both sides are the identity when $u_k = \varnothing$ and zero otherwise. On the other hand, when $W_k \in \{X, Z\}$, these two are close due to Equation (47). Applying Fact 4.24 and Fact 4.28, we get

$$(M_{u_k}^x)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes (I_{1,\dots,\mathsf{k}-1} \otimes \tau_{u_k}^{W_k} \otimes I_{\mathsf{aux}})_{\mathsf{Bob}}, \tag{48}$$

where we have also applied Fact 4.13 to switch to the "$\simeq_{\delta(\epsilon)}$" notation.

Extracting a strategy. Now we use this to define a λ-semiregister strategy $\mathcal{S}_{\text{semi}}$ for $\mathscr{G}_{\text{semi}}$. This strategy will have state $|\psi_{\text{semi}}\rangle = |r_1\rangle \cdots |r_k\rangle\,|\text{aux}\rangle$. In addition, for each input $x = (x_1, x_2)$ and output $a = (a_1, a_2)$, it will have a matrix

$$\Lambda_{a_1,a_2}^{x_1,x_2} := (I_{1,\dots,\mathsf{k}-1} \otimes \tau_{u_k}^{W_k} \otimes I_{\mathsf{aux}}) \cdot M_{(u_1,\dots,u_{k-1}),a_2}^{x_1,x_2} \cdot (I_{1,\dots,\mathsf{k}-1} \otimes \tau_{u_k}^{W_k} \otimes I_{\mathsf{aux}}).$$

First, it follows from M being a $\lambda|_{k-1}$-strategy that this is indeed a λ-semiregister strategy. This is because

$$\begin{aligned}\Lambda_{a_1}^{x_1,x_2} &= (I_{1,\ldots,\mathsf{k}-1} \otimes \tau_{u_k}^{W_k} \otimes I_{\mathsf{aux}}) \cdot M_{u_1,\ldots,u_{k-1}}^{x_1,x_2} \cdot (I_{1,\ldots,\mathsf{k}-1} \otimes \tau_{u_k}^{W_k} \otimes I_{\mathsf{aux}}) \\ &= (I_{1,\ldots,\mathsf{k}-1} \otimes \tau_{u_k}^{W_k} \otimes I_{\mathsf{aux}}) \cdot (\tau_{u_1}^{W_1} \otimes \cdots \otimes \tau_{u_{k-1}}^{W_{k-1}} \otimes I_{\mathsf{k,aux}}) \cdot (I_{1,\ldots,\mathsf{k}-1} \otimes \tau_{u_k}^{W_k} \otimes I_{\mathsf{aux}}) \\ &= \tau_{u_1}^{W_1} \otimes \cdots \otimes \tau_{u_k}^{W_k} \otimes I_{\mathsf{aux}}.\end{aligned}$$

In addition, if $S = \{i \neq k \mid W_i = H\}$, then

$$\Lambda_{a_1,a_2}^{x_1,x_2} = (I_{1,\ldots,\mathsf{k}-1} \otimes \tau_{u_k}^{W_k} \otimes I_{\mathsf{aux}}) \cdot (I_S \otimes A_{\overline{S}}) \cdot (I_{1,\ldots,\mathsf{k}-1} \otimes \tau_{u_k}^{W_k} \otimes I_{\mathsf{aux}}) = I_S \otimes A'_{\overline{S}},$$

where A and A' are matrices acting on the registers not in S and on the auxiliary register.

Next, we show that this has good value. Write $\mathcal{D}$ for the marginal distribution of questions given to player 1 in $\mathscr{G}_{\mathrm{semi}}$. By the consistency check,

$$(M_{(u_1,\ldots,u_{k-1}),a_2}^{x_1,x_2})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes (M_{(u_1,\ldots,u_{k-1}),a_2}^{x_1,x_2})_{\mathsf{Bob}}$$

with respect to $\mathcal{D}$. As a result, Equation (48) and Fact 4.33 imply that

$$(\Lambda_a^x)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes (M_a^x)_{\mathsf{Bob}} \approx_{\delta(\epsilon)} (M_a^x)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}},$$

where the last step uses the self-consistency of M. Applying Fact 4.32, $\mathcal{S}_{\mathrm{semi}}$ passes $\mathscr{G}_{\mathrm{semi}}$ with probability at least $\mathrm{val}_{\mathscr{G}_{k-1}}(\mathcal{S}) - \delta(\epsilon)$. Thus, $\mathrm{val}_\lambda^{\mathrm{semi}}(\mathscr{G}_{\mathrm{semi}}) \geq 1 - \delta(\epsilon)$, and we are done. □

8 The data hiding game

In this section, we introduce a new, simple game called the *data hiding game*. This game assumes two (k, n, q)-semiregister provers with a shared state $|r_1\rangle \cdots |r_k\rangle\,|\mathrm{aux}\rangle$. The goal is to test that a given measurement $\{M_a^x\}_a$ acts as the identity on the k-th register.

Definition 8.1. Let $x = (x_1, x_2)$ with $x_1 = (W_1, \ldots, W_k)$, and suppose $W_k = H$. Then the *data hiding game* $\mathscr{G}_{\mathrm{hide}} := \mathscr{G}_{\mathrm{hide}}(x)$ is given by Figure 3. It has the following parameters:

$$\mathsf{Q\text{-}time}(\mathscr{G}_{\mathrm{hide}}), \mathsf{Q\text{-}length}(\mathscr{G}_{\mathrm{hide}}) = O(|x|), \quad \mathsf{A\text{-}time}(\mathscr{G}_{\mathrm{hide}}), \mathsf{A\text{-}length}(\mathscr{G}_{\mathrm{hide}}) = O(\textstyle\sum_i n_i \log(q_i) + \ell).$$

Here ℓ is the maximum of $|a_2|, |a_2'|$ over all answers a_2 and a_2' given by the provers.

Draw $\boldsymbol{W} \sim \{X, Z\}$. Set $\boldsymbol{x}' = (\boldsymbol{x}_1', x_2)$, where $\boldsymbol{x}_1' = (W_1, \ldots, W_{k-1}, \boldsymbol{W})$. Flip an unbiased coin $\boldsymbol{b} \sim \{0, 1\}$. Distribute the questions as follows:

- Player $\boldsymbol{b}$: give x; receive $(\boldsymbol{a}_1, \boldsymbol{a}_2)$.
- Player $\overline{\boldsymbol{b}}$: give $\boldsymbol{x}'$; receive $(\boldsymbol{a}_1', \boldsymbol{a}_2')$.

Accept if $\boldsymbol{a}_2 = \boldsymbol{a}_2'$.

Figure 3: The game $\mathscr{G}_{\mathrm{hide}}(x)$, with input $x = (x_1, x_2)$

For a measurement $\{M_a\}_a$ which operates on multiple subsystems, it will be convenient to define a version of the measurement in which one of the subsystems is "hidden".

Notation 8.2. Let M be a matrix which operates on $\mathcal{H}_1 \otimes \cdots \otimes \mathcal{H}_k \otimes \mathcal{H}_{\text{aux}}$, and let $i \in [k]$. Define the notation

$$\text{hide}_i(M) := \frac{1}{\text{tr}(I_i)} \cdot I_i \otimes \text{tr}_i(M).$$

If $\{M_a\}_a$ is a measurement, then so is $\{\text{hide}_k(M_a)\}_a$ (though it may not be projective, even if $\{M_a\}_a$ is). Our main result regarding the data hiding game is that passing it with high probability certifies that $\{M_a\}_a$ is close to $\{\text{hide}_k(M_a)\}$.

Theorem 8.3. *Let x be as in Definition 10.3.*

- ***Completeness:*** *Let $\mathcal{S}_{\text{partial}} = (\psi, M^x)$ be a partial (k, n, q)-register strategy which is also a real commuting EPR strategy. Then there is a (k, n, q)-register strategy $\mathcal{S}$ extending $\mathcal{S}_{\text{partial}}$ which is also a real commuting EPR strategy such that $\text{val}_{\mathcal{G}_{\text{hide}}}(\mathcal{S}) = 1$.*
- ***Soundness:*** *Let $\mathcal{S} = (\psi, M)$ be a projective (k, n, q)-semiregister strategy such that $\text{val}_{\mathcal{G}_{\text{hide}}}(\mathcal{S}) \geq 1 - \epsilon$. Then*

  $$(M_a^x)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_\epsilon (\text{hide}_k(M_a^x))_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}$$

  on the singleton distribution on input x.

This section is organized as follows: in Section 8.1 we introduce the *Pauli twirl*, and in Section 8.2 we use it to prove Theorem 8.3. Finally, in Section 9, we design our compiler from layer-two to layer-one. This last step is essentially standard and is included for completeness.

8.1 Some facts about the Pauli twirl

Definition 8.4. The *Pauli twirl* $\mathscr{T} : \mathcal{B}((\mathbb{C}^q)^{\otimes n}) \to \mathcal{B}((\mathbb{C}^q)^{\otimes n})$ is the linear operator

$$\mathscr{T}(A) := \mathop{\mathbf{E}}_{\boldsymbol{u}, \boldsymbol{u}' \sim \mathbb{F}_q^n} \left[X(\boldsymbol{u}) Z(\boldsymbol{u}') \cdot A \cdot Z(-\boldsymbol{u}') X(-\boldsymbol{u}) \right].$$

Proposition 8.5. *Let P be a Pauli matrix on n qudits of dimension q. Then $\mathscr{T}(P) = P$ if P is a multiple of the identity, and otherwise $\mathscr{T}(P) = 0$.*

Proof. The case when P is a multiple of the identity follows from the definition. Otherwise, by **??**, we can write $P = \omega^z X(a) Z(b)$, where at least one of a and b is nonzero. Then

$$\begin{aligned} \mathscr{T}(P) &= \mathop{\mathbf{E}}_{\boldsymbol{u}, \boldsymbol{u}'} \left[X(\boldsymbol{u}) Z(\boldsymbol{u}') \cdot P \cdot Z(-\boldsymbol{u}') X(-\boldsymbol{u}) \right] \\ &= \omega^z \mathop{\mathbf{E}}_{\boldsymbol{u}, \boldsymbol{u}'} \left[X(\boldsymbol{u}) Z(\boldsymbol{u}') \cdot X(a) Z(b) \cdot Z(-\boldsymbol{u}') X(-\boldsymbol{u}) \right]. \end{aligned}$$

By the Pauli X and Z commutation relations (Equation (10)), this rearranges to

$$\omega^z \mathop{\mathbf{E}}_{\boldsymbol{u}, \boldsymbol{u}'} \left[\omega^{\text{tr}[\boldsymbol{u}' \cdot a - \boldsymbol{u} \cdot b]} \right] \cdot X(a) Z(b) = \mathop{\mathbf{E}}_{\boldsymbol{u}, \boldsymbol{u}'} \left[\omega^{\text{tr}[\boldsymbol{u}' \cdot a - \boldsymbol{u} \cdot b]} \right] \cdot P = \mathop{\mathbf{E}}_{\boldsymbol{u}'} \left[\omega^{\text{tr}[\boldsymbol{u}' \cdot a]} \right] \cdot \mathop{\mathbf{E}}_{\boldsymbol{u}} \left[\omega^{-\text{tr}[\boldsymbol{u} \cdot b]} \right] \cdot P = 0.$$

Here the last step uses Fact 3.1 and the fact that at least one of a or b is nonzero. □

In the next couple of sections, we will consider the effects of applying the Pauli twirl to our measurements. For convenience, we will "group" our state into two parts: $|\psi_1\rangle = |r_k\rangle$ is the subsystem we want to hide, and $|\psi_2\rangle = |r_1\rangle \cdots |r_{k-1}\rangle |\text{aux}\rangle$ is the remaining part of this state. In this way, we can consider our measurements as operating on the bipartite state $|\psi_1\rangle |\psi_2\rangle$.

Proposition 8.6. *Let $\{M_a\}$ be a measurement on the state $|\psi\rangle = |\psi_1\rangle |\psi_2\rangle$. Then*

$$(\mathscr{T}_1 \otimes \mathrm{id}_2)[M_a] = \mathrm{hide}_1(M_a),$$

where id_2 is the identity superoperator applied to the second register.

Proof. Let P_J be the elements of the Pauli group on n qudits of dimension q, with $P_0 = I$. Because these form a basis for the set of matrices, we can write

$$M_a = \sum_J P_J \otimes M_{a,J},$$

where the $M_{a,J}$'s are matrices acting on the auxiliary register. Using Proposition 8.5,

$$(\mathscr{T}_1 \otimes \mathrm{id}_2)[M_a] = \sum_J \mathscr{T}(P_J) \otimes M_{a,J} = P_0 \otimes M_{a,0} = I \otimes M_{a,0}.$$

On the other hand, because P_J is traceless unless $J = 0$ (i.e. P_J is the identity),

$$\mathrm{hide}_1(M_a) = \sum_J \mathrm{hide}_1(P_J \otimes M_{a,J}) = \sum_J \frac{1}{q^n} \cdot I \otimes \mathrm{tr}_1(P_J \otimes M_{a,J}) = I \otimes M_{a,0}.$$

These two are equal, completing the proof. □

8.2 Hiding a single coordinate

In this section, we prove Theorem 8.3. Prior to doing so, we prove a couple of technical lemmas. The first shows that a measurement which approximately commutes with the Pauli measurements also approximately commutes with the Pauli observables.

Lemma 8.7. *Let $W \in \{X, Z\}$. Suppose $\{M_a\}$ is a measurement on the state $|\psi\rangle = |\mathrm{EPR}_q^n\rangle |\psi_2\rangle$ for which*

$$(M_a \cdot (\tau_u^W \otimes I_2))_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_\delta ((\tau_u^W \otimes I_2) \cdot M_a)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}.$$

Then the statement

$$(M_a \cdot (W(u) \otimes I_2))_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_\delta ((W(u) \otimes I_2) \cdot M_a)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}$$

holds with respect to the uniform distribution on $\boldsymbol{u} \in \mathbb{F}_q^n$.

Proof. Our goal is to bound

$$\mathop{\mathbf{E}}_{\boldsymbol{u}} \sum_a \| (M_a \cdot (W(\boldsymbol{u}) \otimes I_2) - (W(\boldsymbol{u}) \otimes I_2) \cdot M_a) \otimes I \, |\psi\rangle \, \|^2. \tag{49}$$

by δ. To do so, for a fixed u we introduce the notation

$$\begin{aligned} \Delta_a^u &:= M_a \cdot (W(u) \otimes I_2) - (W(u) \otimes I_2) \cdot M_a \\ &= \sum_{v \in \mathbb{F}_q^n} \omega^{\mathrm{tr}[u \cdot v]} \underbrace{(M_a \cdot (\tau_v^W \otimes I_2) - (\tau_v^W \otimes I_2) \cdot M_a)}_{\Delta_{a,v}}. \end{aligned} \tag{50}$$

We record the following identity, which follows from Equation (50):

$$\mathop{\mathbf{E}}_{\boldsymbol{u}} (\Delta_a^{\boldsymbol{u}})^\dagger \cdot \Delta_a^{\boldsymbol{u}} = \mathop{\mathbf{E}}_{\boldsymbol{u}} \sum_{v, v' \in \mathbb{F}_q^n} \omega^{\mathrm{tr}[\boldsymbol{u} \cdot (v' - v)]} (\Delta_{a,v})^\dagger \Delta_{a,v'} = \sum_{v \in \mathbb{F}_q^n} (\Delta_{a,v})^\dagger \Delta_{a,v}.$$

As a result,

$$(49) = \mathop{\mathbf{E}}_{\boldsymbol{u}} \sum_a \|(\Delta_a^{\boldsymbol{u}} \otimes I)\,|\psi\rangle\,\|^2 = \mathop{\mathbf{E}}_{\boldsymbol{u}} \sum_a \langle\psi|\,(\Delta_a^{\boldsymbol{u}})^\dagger \Delta_a^{\boldsymbol{u}} \otimes I\,|\psi\rangle$$
$$= \sum_a \sum_{v \in \mathbb{F}_q^n} \langle\psi|\,(\Delta_{a,v})^\dagger \Delta_{a,v} \otimes I\,|\psi\rangle$$
$$= \sum_{a,v} \|\Delta_{a,v} \otimes I\,|\psi\rangle\,\|^2.$$

But this is at most $O(\delta)$, by assumption. This completes the proof. □

The next technical lemma shows that a measurement which approximately commutes with products of X and Z observables is approximately equal to its own Pauli twirl.

Lemma 8.8. *Consider the distribution $\mathcal{D}$ on pairs $(\boldsymbol{u}, \boldsymbol{u}')$, where $\boldsymbol{u}, \boldsymbol{u}' \sim \mathbb{F}_q^n$. Suppose $\{M_a\}$ is a measurement on the state $|\psi\rangle = |\mathrm{EPR}_q^n\rangle\,|\psi_2\rangle$ for which*

$$((Z(u')X(u) \otimes I_2) \cdot M_a) \otimes I_{\mathsf{Bob}} \approx_\delta (M_a \cdot (Z(u')X(u) \otimes I_2)) \otimes I_{\mathsf{Bob}}.$$

on distribution $\mathcal{D}$. Then

$$M_a \otimes I_{\mathsf{Bob}} \approx_\delta (\mathscr{T}_1 \otimes \mathrm{id}_2)[M_a] \otimes I_{\mathsf{Bob}}.$$

Proof. By definition,

$$(\mathscr{T}_1 \otimes \mathrm{id}_2)[M_a] = \mathop{\mathbf{E}}_{\boldsymbol{u},\boldsymbol{u}'}[(X(\boldsymbol{u})Z(\boldsymbol{u}') \otimes I_2) \cdot M_a \cdot (Z(-\boldsymbol{u}')X(-\boldsymbol{u}) \otimes I_2)].$$

Similarly,

$$M_a = \mathop{\mathbf{E}}_{\boldsymbol{u},\boldsymbol{u}'}[(X(\boldsymbol{u})Z(\boldsymbol{u}') \otimes I_2) \cdot (Z(-\boldsymbol{u}')X(-\boldsymbol{u}) \otimes I_2) \cdot M_a].$$

As a result, if we set $A^{\boldsymbol{u},\boldsymbol{u}'} = X(\boldsymbol{u})Z(\boldsymbol{u}') \otimes I_2$, and

$$B_a^{\boldsymbol{u},\boldsymbol{u}'} = (Z(-\boldsymbol{u}')X(-\boldsymbol{u}) \otimes I_2) \cdot M_a - M_a \cdot (Z(-\boldsymbol{u}')X(-\boldsymbol{u}) \otimes I_2),$$

then

$$\Delta_a := M_a - (\mathscr{T}_1 \otimes \mathrm{id}_2)[M_a] = \mathop{\mathbf{E}}_{\boldsymbol{u},\boldsymbol{u}'}[A^{\boldsymbol{u},\boldsymbol{u}'} \cdot B_a^{\boldsymbol{u},\boldsymbol{u}'}].$$

We can therefore establish the lemma as follows:

$$\sum_a \|(\Delta_a)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}\,|\psi\rangle\,\|^2 = \sum_a \|\mathop{\mathbf{E}}_{\boldsymbol{u},\boldsymbol{u}'}[A^{\boldsymbol{u},\boldsymbol{u}'} \cdot B_a^{\boldsymbol{u},\boldsymbol{u}'}] \otimes I_{\mathsf{Bob}}\,|\psi\rangle\,\|^2$$
$$\leq \mathop{\mathbf{E}}_{\boldsymbol{u},\boldsymbol{u}'} \sum_a \|(A^{\boldsymbol{u},\boldsymbol{u}'} \cdot B_a^{\boldsymbol{u},\boldsymbol{u}'}) \otimes I_{\mathsf{Bob}}\,|\psi\rangle\,\|^2 \qquad \text{(Jensen's inequality)}$$
$$= \mathop{\mathbf{E}}_{\boldsymbol{u},\boldsymbol{u}'} \sum_a \|(B_a^{\boldsymbol{u},\boldsymbol{u}'}) \otimes I_{\mathsf{Bob}}\,|\psi\rangle\,\|^2 \qquad (A^{\boldsymbol{u},\boldsymbol{u}'} \text{ is unitary})$$

By assumption, this quantity is $O(\delta)$. This concludes the proof. □

Now we prove Theorem 8.3.

Proof of Theorem 8.3. We consider the completeness and soundness cases separately.

Completeness. Let $\mathcal{S}_{\text{partial}} = (\psi, M^x)$ be a partial (k,n,q)-register strategy which is also a real commuting EPR strategy. To this strategy we will add matrices for the questions $x' = (x'_1, x_2)$ with $x'_1 = (W_1, \ldots, W_{k-1}, W)$.

Let $a_1 = (u_1, \ldots, u_k)$, where $u_i = \varnothing$ if $W_i \in \{H, \perp\}$. Let $S = \{i \mid W_i \neq \perp\}$. By definition of a (k,n,q)-register strategy,

$$M^x_a = \bigotimes_{i \in S} \tau^{W_i}_{u_i} \otimes M^{x,a_1}_{a_2},$$

where $M^{x,a_1}_{a_2}$ acts on the auxiliary registers and the registers not in S. Next, set $a' = (a'_1, a_2)$ where $a'_1 = (u_1, \ldots, u_{k-1}, u'_k)$ and $u'_k \in \mathbb{F}_{q_k}^{n_k}$. Then we set

$$(M')^{x'_1,x_2}_{a'_1,a_2} = \bigotimes_{i \in S \setminus k} \tau^{W_i}_{u_i} \otimes \tau^{W}_{u'_k} \otimes M^{x,a_1}_{a_2}.$$

This is a (k,n,q)-register strategy for $\mathscr{G}_{\text{hide}}$ by design. To see that it is value 1, suppose on question x Player $\boldsymbol{b}$ measures $\boldsymbol{a}_1$. Then by Fact 4.38, Player $\overline{\boldsymbol{b}}$ will measure $\boldsymbol{a}'_1$ in which $\boldsymbol{u}'_i = \boldsymbol{u}_i$ for all $i < k$. As a result, to measure $\boldsymbol{a}_2$, Player $\boldsymbol{b}$ will measure $M^{x,\boldsymbol{a}_1}$ and Player $\overline{\boldsymbol{b}}$ will measure $M^{x,\boldsymbol{a}_1}$, both on state $|r_{\overline{S}}\rangle\,|\text{aux}\rangle$. As this is an EPR state, by Fact 4.37 the outcomes will always be the same, and so this strategy has value 1. The fact that this a real strategy follows from the assumption that the matrices M^x_a are real, and the fact that for $W \in \{X, Z\}$, τ^W_u is a real matrix. Finally, the fact that this is a commuting strategy follows from the fact that M^x_a and $(M')^{x'}_{a'}$ are commuting.

Soundness. We write x and $x' = (x'_1, x_2)$ with $x'_1 = (W_1, \ldots, W_{k-1}, W)$ as in the test. Because the test passes with probability $1 - \epsilon$, Fact 4.13 implies that

$$(M^x_{a_2})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_\epsilon I_{\mathsf{Alice}} \otimes (M^{x'}_{a_2})_{\mathsf{Bob}}.$$

Because $\mathcal{S}$ is a (k,n,q)-semiregister strategy, Equation (37) implies that $M^{x'}_{u_k} = \tau^W_{u_k} \otimes I_{\overline{k}}$, where we write $I_{\overline{k}} := I_{1,\ldots,\mathsf{k-1},\mathsf{aux}}$. Our next step is to show that the measurements approximately commute. This follows the analysis of the commutation test (cf. [CGJV18, Lemma 28]).

$$\begin{aligned}
M^{x'}_{u_k} M^x_{a_2} \otimes I_{\mathsf{Bob}} &\approx_\epsilon M^{x'}_{u_k} \otimes M^{x'}_{a_2} && \text{(Fact 4.20)}\\
&\approx_0 I_{\mathsf{Alice}} \otimes M^{x'}_{a_2} M^{x'}_{u_k} && \text{(Fact 4.38)}\\
&= I_{\mathsf{Alice}} \otimes M^{x'}_{u_k} M^{x'}_{a_2}\\
&\approx_\epsilon M^x_{a_2} \otimes M^{x'}_{u_k} && \text{(Fact 4.20)}\\
&\approx_0 M^x_{a_2} M^{x'}_{u_k} \otimes I_{\mathsf{Bob}}. && \text{(Fact 4.38)}
\end{aligned}$$

In summary,

$$((\tau^W_{u_k} \otimes I_{\overline{k}}) \cdot M^x_{a_2})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_\epsilon (M^x_{a_2} \cdot (\tau^W_{u_k} \otimes I_{\overline{k}}))_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}.$$

Recall this is with respect to the distribution $\boldsymbol{W}$ where $\boldsymbol{W} \sim \{X, Z\}$ is uniform. Therefore, it also holds with respect to the distribution where $\boldsymbol{W}$ is fixed to either X or Z. As a result, for a fixed $W \in \{X, Z\}$, by Lemma 8.7,

$$(M^x_{a_2} \cdot (W(u) \otimes I_{\overline{k}}))_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_\epsilon ((W(u) \otimes I_{\overline{k}}) \cdot M^x_{a_2})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}.$$

on distribution $\boldsymbol{u} \sim \mathbb{F}_q^n$. As a result, by Fact 4.38 and Fact 4.20,

$$\begin{aligned}
(M^x_{a_2} \cdot (Z(u')X(u) \otimes I_{\overline{k}}))_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} &\approx_0 (M^x_{a_2} \cdot (Z(u') \otimes I_{\overline{k}}))_{\mathsf{Alice}} \otimes (X(-u) \otimes I_{\overline{k}})_{\mathsf{Bob}}\\
&\approx_\epsilon ((Z(u') \otimes I_{\overline{k}}) \cdot M^x_{a_2})_{\mathsf{Alice}} \otimes (X(-u) \otimes I_{\overline{k}})_{\mathsf{Bob}}\\
&\approx_0 ((Z(u') \otimes I_{\overline{k}}) \cdot M^x_{a_2} \cdot (X(u) \otimes I_{\overline{k}}))_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}\\
&\approx_\epsilon ((Z(u')X(u) \otimes I_{\overline{k}}) \cdot M^x_{a_2})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}},
\end{aligned}$$

With probability $\frac{1}{2}$ each, perform one of the following three tests.

1. **Data hiding:** Draw $(\boldsymbol{x}, \boldsymbol{x}', \boldsymbol{C}) \sim \mathscr{G}_k$, where $\boldsymbol{x} = (\boldsymbol{x}_1, \boldsymbol{x}_2)$ and $\boldsymbol{x}_1 = (\boldsymbol{W}_1, \ldots, \boldsymbol{W}_k)$. If $\boldsymbol{W}_k = H$, play $\mathscr{G}_{\text{hide}}$ with question $\boldsymbol{x}$.
2. **Play game:** Perform $\mathscr{G}_k$.

Figure 4: The game $\mathcal{C}_{k\to\text{semi}}(\mathscr{G}_k)$.

on distribution $\boldsymbol{u}, \boldsymbol{u}' \sim \mathbb{F}_q^n$. Applying Lemma 8.8 and Proposition 8.6, we can therefore conclude

$$M^x_{a_2} \otimes I_{\mathsf{Bob}} \approx_\epsilon (\mathscr{T}_k \otimes \mathrm{id}_{\overline{k}})[M^x_{a_2}] \otimes I_{\mathsf{Bob}} = (\mathrm{hide}_k(M^x_{a_2})) \otimes I_{\mathsf{Bob}}. \qquad \square$$

9 Compiling games with the data hiding test

Now we can show how to compile games from the second layer to the first layer. Our construction is given in the following definition.

Definition 9.1. Let $\mathscr{G}_k$ be a (k, n, q)-register game. Then its compiled version is the game $\mathcal{C}_{k\to\text{semi}}(\mathscr{G}_k)$ defined in Figure 4.

Theorem 9.2. *Suppose $\mathscr{G}_k$ is a (k, n, q)-register game, and consider the (k, n, q)-semiregister game $\mathscr{G}_{\text{semi}} = \mathcal{C}_{k\to\text{semi}}(\mathscr{G}_k)$.*

- ***Completeness:*** *Suppose there is a value-1 (k, n, q)-register strategy for $\mathscr{G}_k$ which is also a real commuting EPR strategy. Then there is a value-1 (k, n, q)-semiregister strategy for $\mathscr{G}_{\text{semi}}$ which is also a real commuting EPR strategy.*
- ***Soundness:*** *If $\mathrm{val}^{\text{semi}}_{k,n,q}(\mathscr{G}_{\text{semi}}) \geq 1 - \epsilon$ then $\mathrm{val}_{k,n,q}(\mathscr{G}_k) \geq 1 - \delta(\epsilon)$, where $\delta(\epsilon) = \mathrm{poly}(\epsilon)$.*

Furthermore,

$$\mathsf{Q\text{-}length}(\mathscr{G}_{\text{semi}}) = O(\mathsf{Q\text{-}length}(\mathscr{G}_k)), \quad \mathsf{A\text{-}length}(\mathscr{G}_{\text{semi}}) = O(\mathsf{A\text{-}length}(\mathscr{G}_k)),$$

$$\mathsf{Q\text{-}time}(\mathscr{G}_{\text{semi}}) = O(\mathsf{Q\text{-}time}(\mathscr{G}_k)), \quad \mathsf{A\text{-}time}(\mathscr{G}_{\text{semi}}) = O(\mathsf{A\text{-}time}(\mathscr{G}_k)).$$

Proof of Theorem 9.2. The communication and time complexities are the result of combining the communication and time complexities from $\mathscr{G}_{\text{k}}$ with the values for the data hiding game from Definition 10.3.

Completeness. Let (ψ, M) be a value-1 (k, n, q)-register strategy for $\mathscr{G}_k$ which is also a commuting EPR strategy. Then for every $x = (x_1, x_2)$ where $x_1 = (W_1, \ldots, W_k)$ with $W_k = H$, by Theorem 8.3 we can extend this strategy to one that passes the data hiding game with question x with probability 1. Thus, this strategy has value 1 overall. In addition, Theorem 8.3 implies this strategy is a real commuting EPR strategy as well.

Soundness. Suppose $\mathcal{S} = (\psi, M)$ is a (k, n, q)-semiregister strategy for $\mathscr{G}_{\text{semi}}$ with value $1 - \epsilon$. By Lemma 5.3, we can assume without loss of generality that M is projective. Our goal will be to decode $\mathcal{S}$ into a (k, n, q)-register strategy $\mathcal{S}_k$ with nearly the same value.

Using the data hiding test. For a fixed question x, write ν_x for the probability that $\mathcal{S}$ passes the test in Item 1. Then on average, the probability that $\mathcal{S}$ passes this test is $\mathbf{E}_{\boldsymbol{x}} \nu_{\boldsymbol{x}}$, which is at least $1 - 2\epsilon$ because the overall test passes with probability at least $1 - \epsilon$. This implies that $\nu_{\boldsymbol{x}} \geq 1 - \epsilon^{1/2}$ with probability at least $1 - 2\epsilon^{1/2}$. Given a matrix M and a $W \in \{X, Z, H, \perp\}$, let us write $\mathrm{hide}_W(M) := \mathrm{hide}_k(M)$ if $W = H$ and $\mathrm{hide}_W(M) := M$ otherwise. For a question x, if $W_k \neq H$, then $\mathrm{hide}_{W_k}(M_a^x) = M_a^x$ trivially. On the other hand, suppose $W_k = H$. Then either $\nu_x \geq 1 - \epsilon^{1/2}$, in which case $M_a^x \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} \mathrm{hide}_{W_k}(M_a^x) \otimes I_{\mathsf{Bob}}$ by Theorem 8.3, or $\nu_x < 1 - \epsilon^{1/2}$, in which case we have the trivial bound $M_a^x \otimes I_{\mathsf{Bob}} \approx_1 \mathrm{hide}_{W_k}(M_a^x) \otimes I_{\mathsf{Bob}}$ from Fact 4.19. Since this latter case happens with probability at most $2\epsilon^{1/2}$, averaging over all x gives us

$$M_a^x \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} \mathrm{hide}_{W_k}(M_a^x) \otimes I_{\mathsf{Bob}}, \tag{51}$$

on the distribution $\mathcal{D}$.

Extracting a strategy. Define the strategy $\mathcal{S}_k = (\psi, \Lambda)$, in which $\Lambda_a^x := \mathrm{hide}_{W_k}(M_a^x)$. First, we show that $\mathcal{S}_k$ is a (k, n, q)-register strategy. To do so, fix $x = (x_1, x_2)$ with $x_1 = (W_1, \ldots, W_k)$ and $a = (a_1, a_2)$ with $a_1 = (u_1, \ldots, u_k)$. Then

$$\Lambda_{a_1}^x = \mathrm{hide}_{W_k}\left(\tau_{u_1}^{W_1} \otimes \cdots \otimes \tau_{u_k}^{W_k} \otimes I_{\mathsf{aux}}\right) = \tau_{u_1}^{W_1} \otimes \cdots \otimes \tau_{u_k}^{W_k} \otimes I_{\mathsf{aux}}.$$

The first equality is by definition of Λ and the fact that $\mathcal{S}$ is a (k, n, q)-quasiregister strategy. The second equality is trivial when $W_k \neq H$ and follows from the fact that $\tau_{\varnothing}^{W_k} = I$ when $W_k = H$. Next, define $S = \{i \neq k \mid W_i = H\}$. If $W_k \neq H$ then $\Lambda_a^x = M_a^x = M_{\overline{S}} \otimes I_S$ for some matrix M. Otherwise, if $W_k = H$, set $\mathsf{F} = \mathrm{tr}(I_k)$. Then

$$\Lambda_a^x = \mathrm{hide}_k(M_a^x) = \mathrm{hide}_k(M_{\overline{S}} \otimes I_S) = \frac{1}{\mathsf{F}} \cdot I_k \otimes \mathrm{tr}_k(M_{\overline{S}} \otimes I_S) = \frac{1}{\mathsf{F}} \cdot I_{S \cup k} \otimes \mathrm{tr}_k(M_{\overline{S}}).$$

The matrix $\mathrm{tr}_k(M_{\overline{S}}) \cdot \mathsf{F}^{-1}$ only acts on the registers in $\overline{S \cup k}$ and the auxiliary register, and as a result, this strategy satisfies data hiding. Thus, $\mathcal{S}_k$ is a (k, n, q)-register strategy.

It remains to show that $\mathcal{S}_k$ has good value. This follows by combining Equation (51) with Fact 4.32: $\mathrm{val}_{\mathscr{G}_k}(\mathcal{S}_k) \geq \mathrm{val}_{\mathscr{G}_{\mathrm{semi}}}(\mathcal{S}) - \delta(\epsilon)$, and so $\mathrm{val}_{k,n,q}(\mathscr{G}_k) \geq 1 - \delta(\epsilon)$. □

10 Partial data hiding

The data-hiding game presented above was used to show that the provers' measurement acts as identity on a subset of the provers' qudits, and thus the prover learns no information from those qudits. In particular, the measurement outcome of any X- or Z-observable measurement on the qubits in the subset is hidden from the prover. In this subsection, we generalize this idea to show how to certify that certain *partial* information about a register is hidden from a prover. This test is a crucial component in our technique of introspection, wherein two provers measure a shared EPR state to sample from the joint distribution over questions of a classical game. The partial data hiding test will prevent one prover from learning the question sampled by the other prover.

Notation 10.1. Given a set $v = \{v_1, \ldots, v_k\}$ of k vectors in $\mathbb{F}_q^n$, denote their span by $V = \mathrm{span}(\{v_1, \ldots, v_k\})$. The orthogonal complement of their span is the subspace $V^\perp = \{a : \forall i \in \{1, \ldots, k\}, \langle a, v_i \rangle = 0\}$. We denote by $\mathsf{Surfaces}_v$ the set of all affine subspaces parallel to V, i.e. sets of the form:

$$s = \{u + \lambda_1 v_1 + \cdots + \lambda_k v_k : \lambda_1, \ldots, \lambda_k \in \mathbb{F}_q\}.$$

For a subspace $s \in \mathsf{Surfaces}_v$, the subspace projector Π_s^v is the projector

$$\Pi_s^v = \sum_{w \in s} |w\rangle\langle w| .$$

Lemma 10.2. *Given a set of vectors $\{v_1, \ldots, v_k\}$, let*

$$\tau^X_{[\forall i, u \cdot v_i = a_i]} = \sum_{u : \forall i, u \cdot v_i = a_i} \tau^X_u .$$

Then $\tau^X_{[\forall i, u \cdot v_i = a_i]}$ commutes with Π_s^v for all $s \in \mathsf{Surfaces}_v$.

Proof. The proof is by calculation.

$$\begin{aligned}
\Pi_s^v \tau^X_{[\forall i, u \cdot v_i = a_i]} &= \sum_{w \in s} |w\rangle\langle w| \sum_{u : \forall i, u \cdot v_i = a_i} \tau^X_u \\
&= \sum_{w \in s} \sum_{u : \forall i, u \cdot v_i = a_i} \mathop{\mathbf{E}}_{\boldsymbol{b}} \omega^{-\mathrm{tr}[\boldsymbol{b} \cdot u]} |w\rangle\langle w| X(\boldsymbol{b}).
\end{aligned}$$

We note an important fact: for any two outcomes u, u' satisfying $u \cdot v_i = u' \cdot v_i = a_i$ for all i, the difference $u - u'$ must lie in $V^\perp$. Fixing some appropriate outcome vector u_0, we can then express the summation variable u as $u_0 + x$ where x runs over $V^\perp$:

$$\begin{aligned}
&= \sum_{w \in s} \sum_{x \in V^\perp} \mathop{\mathbf{E}}_{\boldsymbol{b}} \omega^{-\mathrm{tr}[\boldsymbol{b} \cdot (u_0 + x)]} |w\rangle\langle w| X(\boldsymbol{b}) \\
&= \sum_{w \in s} \sum_{x \in V^\perp} \mathop{\mathbf{E}}_{\boldsymbol{b}} \omega^{-\mathrm{tr}[\boldsymbol{b} \cdot (u_0 + x)]} |w\rangle \langle w - \boldsymbol{b}| .
\end{aligned}$$

Now, the summation over x vanishes unless $\boldsymbol{b} \in (V^\perp)^\perp = V$, by Fact 3.2. This happens with probability q^{k-n} which cancels out the factor of q^{n-k} from evaluating the sum over $x \in V^\perp$, yielding:

$$= \sum_{w \in s} \mathop{\mathbf{E}}_{\boldsymbol{b} \in V} \omega^{-\mathrm{tr}[\boldsymbol{b} \cdot u_0]} |w\rangle \langle w - \boldsymbol{b}| .$$

Now, since $\boldsymbol{b} \in V$, and the summation variable w runs over an affine subspace parallel to V, we can shift it from w to $w + \boldsymbol{b}$, yielding

$$= \sum_{w \in s} \mathop{\mathbf{E}}_{\boldsymbol{b} \in V} \omega^{-\mathrm{tr}[\boldsymbol{b} \cdot u_0]} |w + \boldsymbol{b}\rangle \langle w| .$$

Finally, we can perform the same manipulations in reverse:

$$\begin{aligned}
&= \ldots \\
&= \tau^X_{[\forall i, u \cdot v_i = a_i]} \Pi_s^v .
\end{aligned}$$

□

Definition 10.3. The *partial data-hiding game* is given by Figure 5.

Theorem 10.4. *Let S be any set of k-tuples of vectors in $\mathbb{F}_q^n$, and let x be an arbitrary query.*

Given a set S of k-tuples of linearly independent set of vectors $v_1, \ldots, v_k \in \mathbb{F}_q^n$ and a query string x. Sample $v = \{v_1, \ldots, v_k\}$ uniformly from S. Flip an unbiased coin $\boldsymbol{b} \sim \{0, 1\}$. Perform one of the following three tests with probability $1/3$ each.

1. Distribute the questions as follows:
   - Player $\boldsymbol{b}$: Give $(\perp, x, v)$; receive $(\varnothing, \boldsymbol{a}_2)$.
   - Player $\overline{\boldsymbol{b}}$: give (Z, x, v); receive $(\boldsymbol{a}'_1, \boldsymbol{a}'_2)$.

   Accept if $\boldsymbol{a}_2 = \boldsymbol{a}'_2$.
2. Distribute the questions as follows:
   - Player $\boldsymbol{b}$: Give $(\perp, x, v)$; receive $(\varnothing, \boldsymbol{a}_2)$.
   - Player $\overline{\boldsymbol{b}}$: give $(\perp, x, \{X, v\})$; receive $(\varnothing, \boldsymbol{a}'_2, \{\boldsymbol{a}'_{1,1}, \ldots \boldsymbol{a}'_{1,k}\})$.

   Accept if $\boldsymbol{a}_2 = \boldsymbol{a}'_2$.
3. Distribute the questions as follows:
   - Player $\boldsymbol{b}$: Give $(X, \cdot)$; receive $(\boldsymbol{a}_1, \cdot)$. (Here, "$\cdot$" is the empty string.)
   - Player $\overline{\boldsymbol{b}}$: give $(\perp, \perp, \{X, v\})$; receive $(\varnothing, \varnothing, \{\boldsymbol{a}'_{1,1}, \ldots \boldsymbol{a}'_{1,k}\})$.

   Accept if $\boldsymbol{a}'_{1,i} = v_i \cdot \boldsymbol{a}_1$ for all $i \in \{1, \ldots, k\}$.
4. Distribute the questions as follows:
   - Player $\boldsymbol{b}$: Give $(\perp, x, \{X, v\})$; receive $(\varnothing, \boldsymbol{a}_2, \{\boldsymbol{a}_{1,1}, \ldots, \boldsymbol{a}_{1,k}\})$.
   - Player $\overline{\boldsymbol{b}}$: give $(\perp, \perp, \{X, v\})$; receive $(\varnothing, \varnothing, \{\boldsymbol{a}'_{1,1}, \ldots, \boldsymbol{a}'_{1,k}\})$.

   Accept if $\boldsymbol{a}_{1,i} = \boldsymbol{a}'_{1,i}$ for all $i \in \{1, \ldots, k\}$.

Figure 5: The partial data-hiding game $\mathscr{G}_{\text{hide}}(S, x)$.

- ***Completeness:*** *Let $\mathcal{S}_{\text{partial}} = (\psi, M^{\perp,x,v})$ be a partial $(1,n,q)$-register strategy for $\mathscr{G}_{\text{hide}}(S,x)$, which is also a real commuting EPR strategy, and for which*

$$M_{a_2}^{\perp,x,v} = \sum_{s \in \mathsf{Surfaces}_v} \Pi_s^v \otimes A_{a_2}^{x,v,s},$$

for some measurement $A_{a_2}^{x,v,s}$ acting only on the aux *register. Then there is a $(1,n,q)$-register strategy $\mathcal{S}$ extending $\mathcal{S}_{\text{partial}}$ for which $\text{val}_{\mathscr{G}_{\text{hide}}}(\mathcal{S}) = 1$.*

- ***Soundness:*** *Let $\mathcal{S} = (\psi, M)$ be a projective $(1,n,q)$-register strategy such that $\text{val}_{\mathscr{G}_{\text{hide}}(S,x)}(\mathcal{S}) \geq 1-\epsilon$. Then there exists an ideal measurement $M_a'^{\perp,x,v}$ with the property that*

$$M_a'^{\perp,x,v} = \sum_{s \in \mathsf{Surfaces}_v} \Pi_s^v \otimes M_a^{s,x,v},$$

such that the measurement $M_a^{\perp,x,v}$ used by strategy $\mathcal{S}$ in response to the query x is close to $M_a'^{\perp,x,v}$:

$$(M_a^{\perp,x,v})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_\epsilon (M_a'^{\perp,x,v})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}.$$

To prove this theorem, we will start with some basic facts about the subspace projector measurements. Let us denote the linear subspace spanned by the vectors $v_1, \ldots, v_k$ by V.

Definition 10.5. For any distribution $\mathcal{U}$ over unitary matrices, the *twirl by* $\mathcal{U}$ is the linear operator $\mathscr{T}_{\mathcal{U}} : \mathcal{B}((\mathbb{C}^q)^{\otimes n}) \to \mathcal{B}((\mathbb{C}^q)^{\otimes n})$ defined by

$$\mathscr{T}_{\mathcal{U}}(A) := \mathop{\mathbf{E}}_{\boldsymbol{U} \sim \mathcal{U}} \left[\boldsymbol{U} A \boldsymbol{U}^\dagger \right].$$

Definition 10.6. Let $v = \{v_1, \ldots, v_k\}$ be a set of linearly independent vectors over $\mathbb{F}_q$. Further let $\mathcal{V}$ be the uniform distribution over the set $\{X(a) : a \in V\}$, $\mathcal{Z}$ be the uniform distribution over the set of all Pauli Z operators $\{Z(a) : a \in \mathbb{F}_q^n\}$, and $\mathcal{S}$ be the distribution over products $\boldsymbol{M}\boldsymbol{N}$ where $\boldsymbol{M}$ is drawn from $\mathcal{V}$ and $\boldsymbol{N}$ from $\mathcal{Z}$. Then the *v-subspace twirl* is the twirl over $\mathcal{S}$:

$$\mathscr{T}_{\mathcal{S}} = \mathscr{T}_{\mathcal{V}} \circ \mathscr{T}_{\mathcal{Z}}$$

Proposition 10.7. *Let A be a Hermitian matrix and v a set of k vectors over $\mathbb{F}_q$. Then the v-subspace twirl of A is a linear combination of projectors onto affine subspaces along v:*

$$(\mathscr{T}_{\mathcal{S}} \otimes \text{id}_{\mathsf{aux}})(A) = \sum_{s \in \mathsf{Surfaces}_v} \Pi_s^v \otimes (M_s)_{\mathsf{aux}},$$

for some choice of Hermitian matrices M_s indexed by subspaces s.

Proof. Start by decomposing A into a linear combination of Pauli matrices:

$$A = \sum_{u,u'} X(u)Z(u') \otimes (A_{u,u'})_{\mathsf{aux}}.$$

After the twirl over $\mathcal{Z}$, the only terms that survive are those with no X part, i.e.

$$A' = (\mathscr{T}_{\mathcal{Z}} \otimes \text{id}_{\text{aux}})(A) = \sum_u Z(u) \otimes (A_{0,u})_{\mathsf{aux}}$$

Now if we perform the twirl over $\mathcal{V}$, we get

$$\begin{aligned}
(\mathscr{T}_{\mathcal{V}} \otimes \mathrm{id}_{\mathsf{aux}})(A') &= \sum_u \mathop{\mathbf{E}}_{\boldsymbol{a} \in V} X(\boldsymbol{a}) Z(u) X(\boldsymbol{a})^\dagger \otimes (A_{0,u})_{\mathsf{aux}} \\
&= \sum_u \mathop{\mathbf{E}}_{\boldsymbol{a} \in V} \omega^{\mathrm{tr}[\langle \boldsymbol{a}, u \rangle]} Z(u) \otimes (A_{0,u})_{\mathsf{aux}} \\
&= \sum_{u \in V^\perp} Z(u) \otimes (A_{0,u})_{\mathsf{aux}} && \text{(Fact 3.2)} \\
&= \sum_{u \in V^\perp} \sum_w \omega^{\mathrm{tr}[\langle w, u \rangle]} |w\rangle \langle w| \otimes (A_{0,u})_{\mathsf{aux}} \\
&= \sum_w |w\rangle \langle w| \otimes \sum_{u \in V^\perp} \omega^{\mathrm{tr}[\langle w, u \rangle]} (A_{0,u})_{\mathsf{aux}}. && (52)
\end{aligned}$$

Now, consider a surface $s \in \mathsf{Surfaces}_v$. For some $x \in \mathbb{F}_q^n$, s is the set of points written $w = x + v$, where $v \in V$. Then for any $u \in V^\perp$, $\langle w, u \rangle = \langle x + v, u \rangle = \langle x, u \rangle$, a quantity which depends only on the subspace and not on the point w. Call this quantity $c_{s,u}$. As a result,

$$(52) = \sum_{s \in \mathsf{Surfaces}_v} \sum_{w \in s} |w\rangle \langle w| \otimes \sum_{u \in V^\perp} c_{s,u} (A_{0,u})_{\mathsf{aux}} = \sum_{s \in \mathsf{Surfaces}_v} \Pi_s^v \otimes (\hat{A}_s)_{\mathsf{aux}},$$

where $\hat{A}_s = \sum_{u \in V^\perp} c_{s,u} A_{0,u}$. □

Lemma 10.8. *Let $W \in \{X, Z\}$, and let $v = \{v_1, \ldots, v_k\}$ be a set of k linearly independent vectors in $\mathbb{F}_q^n$ and V be their span. Suppose $\{M_{a_2}\}$ is a measurement for which*

$$(M_{a_2} \cdot (\tau^W_{[\forall i, v_i \cdot a_1 = a_{1,i}]} \otimes I_{\mathsf{aux}})) \otimes I_{\mathsf{Bob}} \approx_\delta ((\tau^W_{[\forall i, v_i \cdot a_1 = a_{1,i}]} \otimes I_{\mathsf{aux}}) \cdot M_{a_2}) \otimes I_{\mathsf{Bob}}, \tag{53}$$

where

$$\tau^W_{[\forall i, v_i \cdot a_1 = a_{1,i}]} = \sum_{a_1 : \forall i, v_i \cdot a_1 = a_{1,i}} \tau^W_{a_1}.$$

Then

$$(M_{a_2} \cdot (W(u) \otimes I_{\mathsf{aux}})) \otimes I_{\mathsf{Bob}} \approx_\delta ((W(u) \otimes I_{\mathsf{aux}}) \cdot M_{a_2}) \otimes I_{\mathsf{Bob}},$$

for a uniformly random $\boldsymbol{u}$ drawn from V.

Proof. To start, given a set of outcomes $a_{1,1}, \ldots, a_{1,k}$, suppose u and u' are outcomes for a full W-basis measurement consistent with these outcomes, i.e. u and u' are vectors such that for all i, $u \cdot v_i = a_{1,i}$. Then it must hold that $u - u' \in V^\perp$. Using this, the bound in Equation (53) becomes

$$\sum_{a_2} \frac{1}{|V^\perp|} \sum_u \left\| \sum_{w \in V^\perp} (M_{a_2} \cdot (\tau^W_{u+w} \otimes I_{\mathsf{aux}}) - (\tau^W_{u+w} \otimes I_{\mathsf{aux}}) \cdot M_{a_2}) \otimes I_{\mathsf{Bob}} |\psi\rangle \right\|^2 \leq \delta, \tag{54}$$

where the factor of $1/|V^\perp|$ is because each outcome $a_{1,1}, \cdots, a_{1,k}$ corresponds to $|V^\perp|$ different choices of u.

Our goal is to bound

$$\mathop{\mathbf{E}}_{\boldsymbol{u} \sim V} \sum_{a_2} \| (M_{a_2} \cdot (W(\boldsymbol{u}) \otimes I_{\mathsf{aux}}) - (W(\boldsymbol{u}) \otimes I_{\mathsf{aux}}) \cdot M_{a_2}) \otimes I_{\mathsf{Bob}} |\psi\rangle \|^2. \tag{55}$$

by δ. To do so, for a fixed u we introduce the notation

$$\Delta^u_{a_2} := M_{a_2} \cdot (W(u) \otimes I_{\mathsf{aux}}) - (W(u) \otimes I_{\mathsf{aux}}) \cdot M_{a_2} \tag{56}$$

$$= \sum_{x \in \mathbb{F}_q^n} \omega^{\mathrm{tr}[u \cdot x]} \underbrace{(M_{a_2} \cdot (\tau^W_x \otimes I_{\mathsf{aux}}) - (\tau^W_x \otimes I_{\mathsf{aux}}) \cdot M_{a_2})}_{\Delta_{a_2,x}}. \tag{57}$$

We record the following identity, which follows from Equation (57) and Fact 3.2:

$$\mathop{\mathbf{E}}_{\boldsymbol{u} \sim V} (\Delta^{\boldsymbol{u}}_{a_2})^\dagger \cdot \Delta^{\boldsymbol{u}}_{a_2} = \mathop{\mathbf{E}}_{\boldsymbol{u} \sim V} \sum_{x,x' \in \mathbb{F}_q^n} \omega^{\mathrm{tr}[\boldsymbol{u} \cdot (x'-x)]} (\Delta_{a_2,x})^\dagger \Delta_{a_2,x'} = \sum_{x \in \mathbb{F}_q^n} \sum_{w \in V^\perp} (\Delta_{a_2,x})^\dagger \Delta_{a_2,x+w}.$$

As a result,

$$\begin{aligned}
(55) = \mathop{\mathbf{E}}_{\boldsymbol{u}} \sum_{a_2} \| (\Delta^{\boldsymbol{u}}_{a_2} \otimes I_{\mathsf{Bob}}) |\psi\rangle \|^2 &= \mathop{\mathbf{E}}_{\boldsymbol{u}} \sum_{a_2} \langle \psi | (\Delta^{\boldsymbol{u}}_{a_2})^\dagger \Delta^{\boldsymbol{u}}_{a_2} \otimes I_{\mathsf{Bob}} |\psi\rangle \\
&= \sum_{a_2} \sum_{x \in \mathbb{F}_q^n} \sum_{w \in V^\perp} \langle \psi | (\Delta_{a_2,x})^\dagger \Delta_{a_2,x+w} \otimes I_{\mathsf{Bob}} |\psi\rangle \\
&= \sum_{a_2,x} \frac{1}{|V^\perp|} \Big\| \sum_{w \in V^\perp} \Delta_{a_2,x+w} \otimes I_{\mathsf{Bob}} |\psi\rangle \Big\|^2,
\end{aligned}$$

where the factor of $1/|V^\perp|$ is again to deal with overcounting. But this is at most $O(\delta)$, by Equation (54). This completes the proof. □

Lemma 10.9. *Let $\{M_a\}$ be a measurement and $\mathcal{U}$ be a distribution over unitaries, and suppose that for U drawn uniformly from $\mathcal{U}$,*

$$((U^\dagger \otimes I_{\mathsf{aux}}) \cdot M_a) \otimes I_{\mathsf{Bob}} \approx_\delta (M_a \cdot (U^\dagger \otimes I_{\mathsf{aux}})) \otimes I_{\mathsf{Bob}},$$

where the distribution inherent in the $\approx_\delta$ notation is the uniform distribution over $\mathcal{U}$. Then

$$M_a \otimes I_{\mathsf{Bob}} \approx_\delta (\mathscr{T}_{\mathcal{U}} \otimes I_{\mathsf{aux}})[M_a] \otimes I_{\mathsf{Bob}}.$$

Proof. By definition,

$$(\mathscr{T}_1 \otimes I_{\mathsf{aux}})[M_a] = \mathop{\mathbf{E}}_{\boldsymbol{U} \sim \mathcal{U}} [(\boldsymbol{U} \otimes I_{\mathsf{aux}}) \cdot M_a \cdot (\boldsymbol{U}^\dagger \otimes I_{\mathsf{aux}})].$$

Similarly,

$$M_a = \mathop{\mathbf{E}}_{\boldsymbol{U} \sim \mathcal{U}} [(\boldsymbol{U} \otimes I_{\mathsf{aux}}) \cdot (\boldsymbol{U}^\dagger \otimes I_{\mathsf{aux}}) \cdot M_a].$$

As a result, if we set

$$B(\boldsymbol{U})_a = (\boldsymbol{U}^\dagger \otimes I_{\mathsf{aux}}) \cdot M_a - M_a \cdot (\boldsymbol{U}^\dagger \otimes I_{\mathsf{aux}}),$$

then

$$\Delta_a := M_a - (\mathscr{T}_1 \otimes I_{\mathsf{aux}})[M_a] = \mathop{\mathbf{E}}_{\boldsymbol{U} \sim \mathcal{U}} [\boldsymbol{U} \cdot B(\boldsymbol{U})_a].$$

We can therefore establish the lemma as follows:

$$\begin{aligned}
\sum_a \| \Delta_a \otimes I_{\mathsf{Bob}} |\psi\rangle \|^2 &= \sum_a \| \mathop{\mathbf{E}}_{\boldsymbol{U} \sim \mathcal{U}} [\boldsymbol{U} \cdot B(\boldsymbol{U})_a] \otimes I_{\mathsf{Bob}} |\psi\rangle \|^2 \\
&\leq \mathop{\mathbf{E}}_{\boldsymbol{U}} \sum_a \| (\boldsymbol{U} \cdot B(\boldsymbol{U})_a) \otimes I_{\mathsf{Bob}} |\psi\rangle \|^2 && \text{(Jensen's inequality)} \\
&= \mathop{\mathbf{E}}_{\boldsymbol{U}} \sum_a \| B(\boldsymbol{U})_a \otimes I_{\mathsf{Bob}} |\psi\rangle \|^2 && (\boldsymbol{U} \text{ is unitary})
\end{aligned}$$

By assumption, this quantity is $O(\delta)$. This concludes the proof. □

Proof of Theorem 10.4. We consider the completeness and soundness cases separately.

Completeness Let $\mathcal{S}_{\text{partial}} = (\psi, M^{\perp,x,v})$ be a partial (k,n,q)-register strategy for $\mathscr{G}_{\text{hide}}(S,x)$ which is also a real commuting EPR strategy, and for which the measurement $M_{a_2}^{\perp,x,v}$ has the form

$$M_{a_2}^{\perp,x,v} = \sum_{s \in \mathsf{Surfaces}_v} \Pi_s^v \otimes A_{a_2}^{s,x,v}.$$

To this strategy we will add matrices for the remaining questions.

- Question (Z,x): the measurement is

$$M_{a_1,a_2}^{(Z,x,v)} = \sum_{s \in \mathsf{Surfaces}_v} \Pi_s^v \cdot \tau_{a_1}^Z \otimes A_{a_2}^{s,x,v}.$$

  This is a well-defined measurement as Π_s^v is diagonal in the Z basis and thus commutes with $\tau_{a_1}^Z$.

- Question $(\perp, x, \{X, v_1, \ldots, v_k\})$: the measurement is

$$M_{\{a_{1,1},\ldots,a_{1,k}\},a_2}^{(\perp,x,\{X,v\})} = \sum_{s \in \mathsf{Surfaces}_v} \Pi_s^v \cdot \tau^X_{[\forall i, a_1 \cdot v_i = a_{1,i}]} \otimes A_{a_2}^{s,x,v}.$$

  This is a well-defined measurement as Π_s^v commutes with $\tau^X_{[\forall i, a_1 \cdot v_i = a_{1,i}]}$ by Lemma 10.2.

- Question $(\perp, \perp, \{X, v_1, \ldots, v_k\})$: the measurement is

$$M_{\{a_{1,1},\ldots,a_{1,k},\varnothing}^{(\perp,\perp,\{X,v\})} = \tau^X_{[\forall i, a_1 \cdot v_i = a_{1,i}]} \otimes I.$$

- Question $(X, \perp)$: the measurement is

$$M_{a_1}^{(X,\cdot)} = \tau_{a_1}^X \otimes I_{\mathsf{aux}}.$$

This is a $(1,n,q)$-register strategy for $\mathscr{G}_{\text{hide}}$ by design, and it is not hard to see that it achieves value 1 on the game. Assuming that the partial strategy $\mathcal{S}_{\text{partial}}$ is a real commuting EPR strategy, it is not hard to see that the full strategy above is also real (this is because if $M^{\perp,x,v}$ is real and of the given form, then the matrices $A_{a_2}^{s,x,v}$ must also be real). That the strategy is also commuting follows from the description of $\mathscr{G}_{\text{hide}}$. In particular, note that while $M_a^{Z,x,v}$ does *not* commute with $M^{\perp,x\{X,v\}}$ or with $M^{X,\perp}$, the test never requires these measurements to be measured at the same time.

Soundness Recall that a strategy $\mathcal{S}$ for this game consists of a state $|\psi\rangle = |\text{EPR}_q^n\rangle \otimes |\text{aux}\rangle$ and measurement operators of three types, corresponding to the four types of queries: $M_{\varnothing,a_2}^{(\perp,x,v)}$, $M_{a_1,a_2}^{(Z,x,v)}$, $M_{\{a'_{1,1},\ldots,a'_{1,k}\},a'_2}^{\perp,x,\{X,v\}}$, and $M_{a_1}^{(X,\cdot)}$.

We start by analyzing the third and fourth parts of the test. The goal of these parts of the test is to certify that when given the query $(\{X,v\},x)$, the prover returns k answers $\boldsymbol{a}'_{1,1},\ldots,\boldsymbol{a}'_{1,k}$ that are consistent with measuring the $X(v_1),\ldots,X(v_k)$ observables on the state. We certify this in two stages. In part three of the test, we ask the first prover to do a complete measurement in the X basis, and send the second prover the query $\{X,v\}$ indicating that it is to perform a partial X measurement, and check consistency of outcomes. Importantly, in this part of the test, we *cannot* send the second prover the query x, since the corresponding Π_s^v measurement does not

commute with the complete X measurement performed by the first prover. Thus, in part four of the test, we send one prover the query $\{X, v\}$ and the other $(x, \{X, v\})$, and check consistency of their outcomes.

Since $\mathcal{S}$ is a (k, n, q)-register strategy, Equation (37) implies that $M_{a_1}^{(X,\cdot)} = \tau_{a_1}^X \otimes I_{\mathsf{Bob}}$. We thus have from the third part of the test that

$$M^{\perp,\perp,\{X,v\}}_{\{a'_{1,1},\ldots,a'_{1,k}\}} \otimes I_{\mathsf{Bob}} \approx_\epsilon \tau^X_{[\forall i, u\cdot v_i = a'_{1,i}]} \otimes I_{\mathsf{Bob}}.$$

From the above equation and the fourth part of the test, we have

$$M^{\perp,x,\{X,v\}}_{\{a'_{1,1},\ldots,a'_{1,k}\}} \otimes I_{\mathsf{Bob}} \approx_\epsilon M^{\perp,\perp,\{X,v\}}_{\{a'_{1,1},\ldots,a'_{1,k}\}} \otimes I_{\mathsf{Bob}} \approx_\epsilon \tau^X_{[\forall i, u\cdot v_i = a'_{1,i}]} \otimes I_{\mathsf{Bob}}.$$

Next, we look at the first and second parts of $\mathscr{G}_{\text{hide}}(S, x)$. These are essentially two instances of the commutation test. The first part of $\mathscr{G}_{\text{hide}}$ certifies that the second outcome of $M^{Z,x,v}_{a_1,a_2}$ is consistent with $M^{\perp,x,v}_{\varnothing,a_2}$, and the hypothesis that the strategy $\mathcal{S}$ is a $(1, n, q)$-register strategy tells us that the first outcome of $M^{Z,x,v}_{a_1,a_2}$ is consistent with $\tau^Z_{a_1} \otimes I_{\mathsf{Bob}}$. Thus, applying the analysis of the commutation, it follows that

$$(M^{\perp,x,v}_{\varnothing,a_2} \cdot (\tau^Z_{a_1} \otimes I_{\mathsf{aux}})) \otimes I_{\mathsf{Bob}} \approx_\epsilon ((\tau^Z_{a_1} \otimes I_{\mathsf{Bob}}) \cdot M^{\perp,x,v}_{\varnothing,a_2}) \otimes I_{\mathsf{Bob}}.$$

A similar analysis for the second part of $\mathscr{G}_{\text{hide}}(S)$ certifies that

$$((\tau^X_{[\forall i, a_1\cdot v_i = a'_{1,i}]} \otimes I_{\mathsf{aux}}) \cdot M^{\perp,x,v}_{\varnothing,a_2}) \otimes I_{\mathsf{Bob}} \approx_\epsilon (M^{\perp,x,v}_{\varnothing,a_2} \cdot (\tau^X_{[\forall i, a_1\cdot v_i = a'_{1,i}]} \otimes I_{\mathsf{aux}})) \otimes I_{\mathsf{Bob}}.$$

As a result, it holds that $W \in \{X, Z\}$, by Lemma 10.8,

$$(M^{\perp,x,v}_{\varnothing,a_2} \cdot (W(u) \otimes I_{\mathsf{aux}})) \otimes I_{\mathsf{Bob}} \approx_\epsilon ((W(u) \otimes I_{\mathsf{aux}}) \cdot M^{\perp,x,v}_{\varnothing,a_2}) \otimes I_{\mathsf{Bob}}.$$

where if $W = X$, then u is chosen uniformly over V, and if $W = Z$, then u is drawn uniformly from $\mathbb{F}_q^n$. As a result, by Fact 4.38 and Fact 4.20,

$$\begin{aligned}
(M^{\perp,x,v}_{\varnothing,a_2} \cdot (Z(u')X(u) \otimes I_{\mathsf{aux}})) \otimes I_{\mathsf{Bob}} &\approx_0 (M^{\perp,x,v}_{\varnothing,a_2} \cdot (Z(u') \otimes I_{\mathsf{aux}})) \otimes (X(-u) \otimes I_{\mathsf{aux}}) \\
&\approx_\epsilon ((Z(u') \otimes I_{\mathsf{aux}}) \cdot M^{\perp,x,v}_{\varnothing,a_2}) \otimes (X(-u) \otimes I_{\mathsf{aux}}) \\
&\approx_0 ((Z(u') \otimes I_{\mathsf{aux}}) \cdot M^{\perp,x,v}_{\varnothing,a_2} \cdot (X(u) \otimes I_{\mathsf{aux}})) \otimes I_{\mathsf{Bob}} \\
&\approx_\epsilon ((Z(u')X(u) \otimes I_{\mathsf{aux}}) \cdot M^{\perp,x,v}_{\varnothing,a_2}) \otimes I_{\mathsf{Bob}},
\end{aligned}$$

on the distribution where v is chosen from S, and then $\boldsymbol{u} \sim V$, $\boldsymbol{u}' \sim \mathbb{F}_q^n$. Applying Lemma 10.9 and Proposition 10.7, we can therefore conclude that

$$M^{\perp,x,v}_{\varnothing,a_2} \otimes I_{\mathsf{Bob}} \approx_\epsilon (\mathscr{T}_{\mathcal{S}} \otimes I_{\mathsf{aux}})[M^x_{\varnothing,a_2}] \otimes I_{\mathsf{Bob}} = \left(\sum_{s \in \mathsf{Surfaces}_v} \Pi^v_s \otimes (A^{x,v,s}_{a_2})_{\mathsf{aux}} \right) \otimes I_{\mathsf{Bob}},$$

for some choice of measurements $A^{x,v,s}_{a_2}$ on the aux register. □

Part IV
NEEXP protocol

11 A review of a classical PCP theorem

We begin Part IV by reviewing the following classical PCP theorem:

$$\mathsf{Succinct\text{-}3Sat} \in \mathrm{PCP}[n, \mathrm{poly}(n)]. \tag{58}$$

This implies, by standard reduction, that Succinct-3Sat $\in$ MIP, which is the main result of [BFL91]. Reviewing this serves two purposes: (i) our MIP* protocols are inspired by this PCP construction, and (ii) their correctness is actually shown by reduction to the correctness of this PCP construction (Lemma 15.6 below). This section closely follows the excellent course notes of Harsha [Har10].

11.1 The instance

The input to the verifier is an instance of the Succinct-3Sat problem, i.e. a circuit $\mathcal{C}$ of size s with $3n+3$ inputs. We apply the Tseitin transformation to it to produce a formula $\mathcal{F}$ with $n' = 3n+3+s$ inputs. It encodes the 3Sat formula $\psi := \psi_{\mathcal{F}}$ on $N = 2^n$ variables which contains $(x_i^{b_1} \vee x_j^{b_2} \vee x_k^{b_3})$ as a clause if and only if $\mathcal{F}(i, j, k, b_1, b_2, b_3, w) = 1$ for some $w \in \{0,1\}^s$.

11.2 Encoding assignments

Writing $\mathcal{S} = \{0,1\}^n$, an assignment to the variables of ψ is a string $a \in \{0,1\}^{\mathcal{S}}$, or equivalently a string in $\{0,1\}^N$. The first step of the PCP theorem is to take the low-degree encoding of a. We begin by choosing parameters.

Definition 11.1. Recall that $N = 2^n$, $h = 2^{t_1}$, $q = 2^{t_2}$, and m are admissible parameters if $t_1 \leq t_2$ and $h^m \geq N$. We call them *exactly* admissible if the stronger condition $h^m = N = 2^n$ holds.

We select n, $h = 2^{t_1}$, $q = 2^{t_2}$, and m to satisfy our "rule of thumb" parameter settings (Equation (1)):

$$h = \Theta(n), \quad m = \Theta\left(\frac{n}{\log(n)}\right), \quad q = \Theta((n')^{10}).$$

Note that q depends on n' rather than just n.

It remains to choose H and π. Our construction requires that the permutation π be efficiently computable, and so we pick these according to the canonical low-degree encoding (Definition 3.8). This entails setting $H = H_{t_1,t_2}$. As for π, we modify the construction slightly. This is because the canonical low-degree encoding is designed for strings whose coordinates are indexed by an integer $i \in [n]$, which must first be converted to binary when computing π. However, the coordinates of our strings $a \in \{0,1\}^{\mathcal{S}}$ are indexed by elements of $\mathcal{S} = \{0,1\}^n$, which are already written in binary, allowing us to skip the conversion. Hence, within this section, we define $\pi := \pi_{n,t_1,t_2} : \mathcal{S} \to H^m$ by setting

$$\pi(b_1, \ldots, b_n) := \sigma_{n,t_1,t_2}(b_1, \ldots, b_n) = (\sigma(b_1, \ldots, b_{t_1}), \sigma(b_{t_1+1}, \ldots, b_{2t}), \ldots, \sigma(b_{n-t_1+1}, \ldots, b_n)),$$

where $\sigma := \sigma_{t_1,t_2}$. Given these parameters, an assignment a is encoded as a degree-$O(mh)$ polynomial $g_a : \mathbb{F}_q^m \to \mathbb{F}_q$.

The crucial property of π that we will need later is that it has an efficiently-computable, low-degree inverse. We will show this here. To do so, we begin by recalling the notation $\mathrm{ind}_{H,x}(y)$ for the indicator function of $x \in H$ over H:

$$\mathrm{ind}_{H,x}(y) = \frac{\prod_{b \neq x}(y-b)}{\prod_{b \neq x}(x-b)},$$

where b ranges over H. This is a degree-h polynomial which can be computed in time $\mathrm{poly}(h,q)$.

Definition 11.2. Let $N = 2^n$, $h = 2^{t_1}$, $q = 2^{t_2}$, and m be exactly admissible parameters. Set $H = H_{t_1,t_2}$, $\sigma = \sigma_{t_1,t_2}$, and $\pi = \pi_{n,t_1,t_2}$. Consider the function $\mu := \mu_{t_1,t_2} : \mathbb{F}_q \to \mathbb{F}_q^{t_1}$ whose i-th component is defined as

$$\mu_i(y) = \sum_{x \in H : \mathrm{tr}[e_i \cdot x] = 1} \mathrm{ind}_{H,x}(y).$$

Let $y = b_1 \cdot e_1 + \cdots + b_{t_1} \cdot e_{t_1}$ be an element of H. Then $\mu_i(y) = b_i$, and so $\mu(y) = (b_1, \ldots, b_{t_1})$. This means that $\mu(\sigma(b_1, \ldots, b_{t_1})) = (b_1, \ldots, b_{t_1})$. As a result, if we define the function $\nu := \nu_{n,t_1,t_2} : \mathbb{F}_q^m \to \mathcal{S}_n$ to be

$$\nu(x_1, \ldots, x_m) := (\mu(x_1), \ldots, \mu(x_m))$$

then $\nu(\pi(x)) = x$ for any $x \in \mathcal{S}_n$. Each component of ν is the sum of $\frac{h}{2}$ indicator functions, and is therefore degree-h and computable in time $\mathrm{poly}(h,q)$. As a result, ν is computable in time $\mathrm{poly}(n,h,q)$.

11.3 Encoding the formula

Our next step is to produce a similar "low-degree encoding" for the formula ψ. This will be a function $g_\psi : \mathbb{F}_q^{m'} \to \mathbb{F}_q$, for $m' = 3m + 3 + s$, with the property that for all $i, j, k \in \mathcal{S}$, $b_1, b_2, b_3 \in \{0,1\}$, and $w \in \{0,1\}^s$,

$$g_\psi(\pi(i), \pi(j), \pi(k), b_1, b_2, b_3, w) = \mathcal{F}(i, j, k, b_1, b_2, b_3, w).$$

This can be accomplished by setting $\mathcal{S}' := \{0,1\}^{n'}$, viewing $\mathcal{F}$ as computing a string $a_\psi \in \{0,1\}^{\mathcal{S}'}$, and setting g_ψ to be its low-degree encoding. However, the verifier in our protocol will be required to evaluate g_ψ on a particular input, and this seems challenging given that this g_ψ is computed by interpolating over an exponential number of points. Instead, we will produce a g_ψ which we can efficiently evaluate at any point using the fact that we have a succinct formula $\mathcal{F}$ representing ψ.

To begin, we convert $\mathcal{F}$ into an algebraic formula which operates over $\mathbb{F}_q$-valued inputs using arithmetization (cf. Definition 3.28). Set $\mathcal{F}_{\mathrm{arith}} := \mathrm{arith}_q(\mathcal{F})$. This is a function $\mathcal{F}_{\mathrm{arith}} : \mathbb{F}_q^{n'} \to \mathbb{F}_q$ with the property that for any $x \in \{0,1\}^{n'}$, $\mathcal{F}_{\mathrm{arith}}(x) = \mathcal{F}(x)$. Furthermore, $\mathcal{F}_{\mathrm{arith}}$ has degree $O(n')$ and is computable in time $\mathrm{poly}(n', q)$. We can now define the function g_ψ as follows.

Definition 11.3. Let $N = 2^n$, $h = 2^{t_1}$, $q = 2^{t_2}$, and m be exactly admissible parameters. Set $\nu := \nu_{n,t_1,t_2}$. Let $\mathcal{C}$ be a Succinct-3Sat instance whose Tseitin transformation $\mathcal{F}$ has $n' = 3n + 3 + s$ inputs and encodes the formula $\psi := \psi_{\mathcal{F}}$, and let $\mathcal{F}_{\mathrm{arith}} = \mathrm{arith}_q(\mathcal{F})$. Write $m' = 3m + 3 + s$. Then we define $g_\psi := g_{\psi,n,t_1,t_2} : \mathbb{F}_q^{m'} \to \mathbb{F}_q$ to be the function

$$g_\psi(x_1, x_2, x_3, b_1, b_2, b_3, w) := \mathcal{F}_{\mathrm{arith}}(\nu(x_1), \nu(x_2), \nu(x_3), b_1, b_2, b_3, w).$$

This is degree $h \cdot O(n')$ and can be computed in time $\mathrm{poly}(n', h, q)$.

For shorthand, we will often write inputs to g_ψ as tuples $(x, b, w) \in \mathbb{F}_q^{3m+3+s}$, where $x = (x_1, x_2, x_3)$ contains three strings in $\mathbb{F}_q^m$ and $b = (b_1, b_2, b_3)$ contains three numbers in $\mathbb{F}_q$.

11.4 Zero on subcube

Given a function $g : \mathbb{F}_q^m \to \mathbb{F}_q$, we would like to check that it is the low-degree encoding of an assignment which satisfies ψ. To do so, we define the following function.

Definition 11.4. Let $N = 2^n$, $h = 2^{t_1}$, $q = 2^{t_2}$, and m be exactly admissible parameters. Let $\mathcal{C}$ be a Succinct-3Sat instance whose Tseitin transformation $\mathcal{F}$ has $n' = 3n+3+s$ inputs and encodes the formula $\psi := \psi_{\mathcal{F}}$, and let $g_\psi := g_{\psi,n,t_1,t_2}$. Set $m' = 3m+3+s$. Then given a function $g : \mathbb{F}_q^m \to \mathbb{F}_q$, we define $\mathrm{sat}_{\psi,g} := \mathrm{sat}_{\psi,g,n,t_1,t_2} : \mathbb{F}_q^{m'} \to \mathbb{F}_q$ to be the function

$$\mathrm{sat}_{\psi,g}(x,b,w) := g_\psi(x,b,w) \cdot (g(x_1) - b_1)(g(x_2) - b_2)(g(x_3) - b_3).$$

The crucial property we would like to check is that $\mathrm{sat}_{\psi,g}$ is *zero on the subcube* $H_{\mathrm{zero}} := H^{3m} \otimes \{0,1\}^{3+s}$.

Proposition 11.5. *The function* $\mathrm{sat}_{\psi,g}$ *is zero on the subcube* H_{zero} *for some* $g : \mathbb{F}_q^m \to \mathbb{F}_q$ *if and only if* ψ *is satisfiable. If it* is *satisfiable,* g *may be taken to be degree-*$O(mh)$*, in which case* $\mathrm{sat}_{\psi,g}$ *is degree-*$O(mh + hn')$*.*

Note what Proposition 11.5 does *not* say. It does not say that if $\mathrm{sat}_{\psi,g}$ is zero on the subcube, then g is the low degree encoding of a satisfying assignment of ψ. It does not even say that g must be low-degree. (Indeed, g might have high degree, as $\mathrm{sat}_{\psi,g}$ only checks g on those numbers in the range of π.) What it *does* say is that *if* ψ is satisfiable, *then* there exists such a g which is low-degree: the low-degree encoding of a satisfying assignment. Our strategy, then, will be to verify that that g is low-degree and then use this fact to verify that $\mathrm{sat}_{\psi,g}$ is zero on the subcube. (We can then "forget" that g is low-degree, as it is no longer required for the analysis.)

To verify this that $\mathrm{sat}_{\psi,g}$ is zero on H_{zero}, we would like it to be encoded so that this is self-evidently true. This entails expanding $\mathrm{sat}_{\psi,g}$ in a "basis" of simple polynomials which are zero on the subcube. To begin, given a subset $S \subseteq \mathbb{F}_q$, define

$$\mathrm{zero}_S(x) := \prod_{b \in S} (x - b).$$

The following proposition shows how to expand into this "zero" basis.

Proposition 11.6. *Let* $f : \mathbb{F}_q^n \to \mathbb{F}_q$ *be a degree-*d *polynomial which is zero on the subcube* $H = H_1 \otimes \cdots \otimes H_n$*. Then there exist degree-*$(d-h)$ *"coefficient polynomials"* $c_1, \ldots, c_n$ *such that*

$$f(x) = \mathrm{zero}_{H,c}(x) := \sum_{i=1}^{n} \mathrm{zero}_H(x_i) \cdot c_i(x).$$

For simplicity, we will write $\mathrm{zero}_{H,c}$ instead of $\mathrm{zero}_{H_{\mathrm{zero}},c}$. We would like our proof to consist of the function g and the coefficient polynomials $c_1, \ldots, c_{m'}$ so that we may check the equality $\mathrm{sat}_{\psi,g} \equiv \mathrm{zero}_{H,c}$. The following lemma shows so long as these functions are low-degree, we can verify that they are equal, and therefore show that ψ is satisfiable.

Lemma 11.7. *Let* $N = 2^n$*,* $h = 2^{t_1}$*,* $q = 2^{t_2}$*, and* m *be exactly admissible parameters. Let* $\mathcal{C}$ *be a* Succinct-3Sat *instance whose Tseitin transformation* $\mathcal{F}$ *has* $n' = 3n + 3 + s$ *inputs and encodes the formula* $\psi := \psi_{\mathcal{F}}$*,. Set* $m' = 3m + 3 + s$*. Let* $g : \mathbb{F}_q^m \to \mathbb{F}_q$*, and set* $\mathrm{sat}_{\psi,g} := \mathrm{sat}_{\psi,g,n,t_1,t_2}$*. Let* $c_1, \ldots, c_{m'} : \mathbb{F}_q^{m'} \to \mathbb{F}_q$*, set* $H_{\mathrm{zero}} = H^{3m} \otimes \{0,1\}^{3+s}$*, and write* $\mathrm{zero}_{H_c} := \mathrm{zero}_{H_{\mathrm{zero}},c}$*. Suppose that* g *is degree-*d_1*, and suppose that* $c_1, \ldots, c_{m'}$ *are degree-*d_2*. Suppose*

$$\Pr_{\boldsymbol{x} \sim \mathbb{F}_q^{m'}}[\mathrm{sat}_{\psi,g}(\boldsymbol{x}) = \mathrm{zero}_{H,c}(\boldsymbol{x})] > \frac{\max\{O(hn') + 3d_1, h + d_2\}}{q}.$$

Then ψ *is satisfiable.*

Proof. By Definition 11.3, $\text{sat}_{\psi,g}$ has degree $h \cdot O(n') + 3d_1$. In addition, $\text{zero}_{H,c}$ has degree $h + d_2$. Define $f = \text{sat}_{\psi,g} - \text{zero}_{H,c}$. Then f has degree $\max\{O(hn') + 3d_1, h + d_2\}$. By assumption, $f(\boldsymbol{x}) = 0$ with probability larger than $\deg(f)/q$ over a uniformly random $\boldsymbol{x} \sim \mathbb{F}_q^{m'}$. By Schwartz-Zippel (Lemma 3.6), this means that $f \equiv 0$, which implies that $\text{sat}_{\psi,g} \equiv \text{zero}_{H,c}$. But $\text{zero}_{H,c}$ is self-evidently zero on the subcube H_{zero}, meaning that $\text{sat}_{\psi,g}$ is as well. By Proposition 11.5, ψ is satisfiable. □

Ensuring that $\text{sat}_{\psi,g}$ is low-degree requires ensuring that g is low-degree, and this can be accomplished with the low-degree test. Arguing similarly for $\text{zero}_{H,c}$ requires ensuring that each c_i is low-degree, and this can be done with the simultaneous plane-versus-point low-degree test (Theorem 3.19).

11.5 The PCP

We can now state the contents of our probabilistically checkable proof for the satisfiability of ψ. It consists of the following four tables.

1. A claimed low-degree polynomial $g : \mathbb{F}_q^m \to \mathbb{F}_q$.
2. A set of claimed low-degree polynomials $c_1, \ldots, c_{m'} : \mathbb{F}_q^{m'} \to \mathbb{F}_q$.
3. A "planes table", containing for each plane s in $\mathbb{F}_q^m$ a degree-d bivariate polynomial.
4. Another planes table, containing for each plane s in $\mathbb{F}_q^{m'}$ an m'-tuple of degree-d bivariate polynomials.

The verifier works as follows: first, it performs the low-degree test between g and its planes table. Second, it performs the simultaneous low-degree test between the c_i's and their plane table. Both of these use the degree parameter $d = \Theta((n')^2)$, which is chosen to upper-bound both $\Theta(mh)$ and $\Theta(mh + hn')$. Finally, it picks a uniformly random $(\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{w}) \in \mathbb{F}_q^{m'}$ and checks the equality $\text{sat}_{\psi,g}(\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{w}) = \text{zero}_{H,c}(\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{w})$. It accepts if all the tests accept individually.

When ψ is satisfiable, there is always a proof that makes the verifier accept with probability 1. This entails setting g to be the low-degree encoding of a satisfying assignment, and setting $c_1, \ldots, c_{m'}$ to be the coefficient polynomials of $\text{sat}_{\psi,g}$. The following proposition shows that when ψ is not satisfiable, the verifier always rejects with probability at least $\frac{1}{10}$.

Proposition 11.8. *If the verifier accepts with probability at least* $9/10$*, then* ψ *is satisfiable.*

Proof. If the verifier accepts with probability at least $9/10$, then each individual test accepts with probability at least $9/10$. Applying Theorems 3.12 and 3.19, we get degree-d functions $\overline{g} : \mathbb{F}_q^n \to \mathbb{F}_q$ and $\overline{c}_1, \ldots, \overline{c}_{m'} : \mathbb{F}_q^{m'} \to \mathbb{F}_q$ such that

$$\text{dist}(g, \overline{g}) \leq \tfrac{2}{10}, \quad \text{dist}(c, \overline{c}) \leq \tfrac{2}{10}, \quad \text{dist}(\text{sat}_{\psi,g}, \text{zero}_{H,c}) \leq \tfrac{1}{10}.$$

(Here, we are assuming that q is a sufficiently large function of m and h.) By the union bound, $\text{dist}(\text{sat}_{\psi,g}, \text{sat}_{\psi,\overline{g}}) \leq 3\text{dist}(g, \overline{g})$. As a result, by the triangle inequality

$$\begin{aligned}\text{dist}(\text{sat}_{\psi,\overline{g}}, \text{zero}_{H,\overline{c}}) &\leq \text{dist}(\text{sat}_{\psi,\overline{g}}, \text{sat}_{\psi,g}) + \text{dist}(\text{sat}_{\psi,g}, \text{zero}_{H,c}) + \text{dist}(\text{zero}_{H,c} + \text{zero}_{H,\overline{c}}) \\ &\leq 3\text{dist}(g, \overline{g}) + \text{dist}(\text{sat}_{\psi,g}, \text{zero}_{H,c}) + \text{dist}(c, \overline{c}) \leq 3 \cdot \tfrac{2}{10} + \tfrac{2}{10} + \tfrac{1}{10} = \tfrac{9}{10}.\end{aligned}$$

By Lemma 11.7, ψ is therefore satisfiable. □

Time and communication complexity.

- **Question length:** The verifier performs two low-degree tests and draws a random point in $\mathbb{F}_q^{m'}$. These are of size $\Theta(m \log(q))$, $\Theta(m' \log(q))$, and $\Theta(m' \log(q))$, respectively, all of which are $O(n')$ bits.
- **Answer length:** The verifier performs one normal low-degree test, and then a second low-degree test with answer complexity m' times the normal answer complexity. These are of total length $(m'+1) \cdot d^2 \log(q) = O((n')^9)$. Finally, in the last test, it queries each of g and $c_1, \ldots, c_{m'}$ for a point in $\mathbb{F}_q$, a total communication cost of $(m'+1)\log(q) = O(n')$. In total, the answer length is $\mathrm{poly}(n')$.
- **Runtime:** The verifier runs in time $\mathrm{poly}(n')$. This includes computing $\mathrm{sat}_{\psi,g}(\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{w})$, which requires computing $g_\psi(\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{w})$, taking time $\mathrm{poly}(n', h, q) = \mathrm{poly}(n')$.

12 NEEXP preliminaries

12.1 Introspection games

In this section, we introduce introspection games. These are games in which, rather than the verifier sampling the questions, the provers sample them instead.

Definition 12.1 (Introspection games)**.** An *introspection game* is played between two provers Alice and Bob in which Alice returns two strings x_A and a and Bob returns two strings x_B and a' (the verifier does not specify a question). Here, x_A and x_B are interpreted as Alice and Bob's "share" of the jointly sampled "question" $x = (x_A, x_B)$, and a and a' are interpreted as their "answers". The verifier then applies its *evaluation function* V to the answers and accepts if $V(x_A, x_B, a, b) = 1$.

The following three facts show that we can convert between strategies for a "normal" game and strategies for an introspective version of the game. This allows us to prove soundness results for the "normal" game and "bootstrap" them up to the introspective game as well.

Fact 12.2. *This fact concerns two games and two strategies.*

1. *Let $\mathscr{G}_{\mathrm{intro}}$ be the introspective game with evaluation function V. Consider a strategy $\mathcal{S}_{\mathrm{intro}}$ for Alice and Bob with shared state $|\mathrm{intro}\rangle = |\mathrm{question}\rangle \otimes |\mathrm{answer}\rangle$ in which Alice and Bob's measurements are given by*

$$\{\mathsf{P}_{x_A} \otimes A_a^{x_A}\}_{x_A, a}, \quad \{\mathsf{Q}_{x_B} \otimes B_{a'}^{x_B}\}_{x_B, a'},$$

   respectively. Write $\mathcal{D}$ for the distribution on outcomes (x_A, x_B) when the measurement $\{\mathsf{P}_{x_A} \otimes \mathsf{Q}_{x_B}\}_{x_A, x_B}$ is performed on $|\mathrm{question}\rangle$.

2. *Let $\mathscr{G}$ be the "normal" game played as follows: sample $\boldsymbol{x} = (\boldsymbol{x}_A, \boldsymbol{x}_B) \sim \mathcal{D}$. Distribute the questions as follows:*

   - *Alice: give $\boldsymbol{x}_A$; receive $\boldsymbol{a}$.*
   - *Bob: give $\boldsymbol{x}_B$; receive $\boldsymbol{b}$.*

   Accept if $V(\boldsymbol{x}_A, \boldsymbol{x}_B, \boldsymbol{a}, \boldsymbol{b}) = 1$. Write $\mathcal{S}$ for the strategy with shared state $|\mathrm{answer}\rangle$ in which Alice's strategy is $\{A_a^{x_A}\}_a$ and Bob's strategy is $\{B_{a'}^{x_B}\}_{a'}$.

Then $\mathrm{val}_{\mathscr{G}}(\mathcal{S}) = \mathrm{val}_{\mathscr{G}_{\mathrm{intro}}}(\mathcal{S}_{\mathrm{intro}})$.

Fact 12.3. *Let* $\{A_a^x\}_x$ *and* $\{B_a^x\}_x$ *be measurements such that*

$$A_a^x \otimes I_{\mathsf{Bob}} \approx_\delta B_a^x \otimes I_{\mathsf{Bob}} \tag{59}$$

on state $|\text{answer}\rangle$ *and distribution* $\mathcal{D}$*. Next, let* $\{\mathsf{Q}_x\}_x$ *be a measurement and* $|\text{question}\rangle$ *be a bipartite state such that the distribution of measurement outcomes produced by measuring* $\{\mathsf{Q}_x \otimes I_{\mathsf{Bob}}\}_x$ *on* $|\text{question}\rangle$ *is* $\mathcal{D}$*. Then*

$$(\mathsf{Q}_x \otimes A_a^x)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_\delta (\mathsf{Q}_x \otimes B_a^x)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \tag{60}$$

on state $|\text{question}\rangle \otimes |\text{answer}\rangle$*. Moreover, if* Q_x *is a projective measurement, then the reverse implication holds: if Equation (62) holds on* $|\text{question}\rangle \otimes |\text{answer}\rangle$*, then Equation (61) holds on the state* $|\text{answer}\rangle$*.*

Proof. First, we show the forward implication. By definition, we want to bound

$$\begin{aligned}
&\sum_{x,a} \|(\mathsf{Q}_x \otimes A_a^x \otimes I_{\mathsf{Bob}} - \mathsf{Q}_x \otimes B_a^x \otimes I_{\mathsf{Bob}})\,|\text{question}\rangle \otimes |\text{answer}\rangle\,\|^2 \\
=&\sum_{x,a} \|(\mathsf{Q}_x \otimes I)_{\mathsf{question}} \otimes (A_a^x \otimes I - B_a^x \otimes I)_{\mathsf{answer}}\,|\text{question}\rangle \otimes |\text{answer}\rangle\,\|^2 \\
=&\sum_{x,a} \langle\text{question}| \otimes \langle\text{answer}|\,(\mathsf{Q}_x \otimes I)^2_{\mathsf{question}} \otimes (A_a^x \otimes I - B_a^x \otimes I)^2_{\mathsf{answer}}\,|\text{question}\rangle \otimes |\text{answer}\rangle \\
\leq&\sum_{x,a} \langle\text{question}| \otimes \langle\text{answer}|\,(\mathsf{Q}_x \otimes I)_{\mathsf{question}} \otimes (A_a^x \otimes I - B_a^x \otimes I)^2_{\mathsf{answer}}\,|\text{question}\rangle \otimes |\text{answer}\rangle \\
=&\mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \langle\text{answer}|\,(A_a^{\boldsymbol{x}} \otimes I - B_a^{\boldsymbol{x}} \otimes I)^2\,|\text{answer}\rangle \\
=&\mathop{\mathbf{E}}_{\boldsymbol{x}} \sum_a \|(A_a^{\boldsymbol{x}} \otimes I - B_a^{\boldsymbol{x}} \otimes I)^2\,|\text{answer}\rangle\,\|^2.
\end{aligned}$$

But this is at most δ by assumption. Now, for the reverse implication, note that if Q_x is projective, then the inequality above becomes an equality. □

Fact 12.4. *Let* $\{A_a^x\}_x$ *and* $\{B_a^x\}_x$ *be measurements such that*

$$(A_a^x)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_\delta I_{\mathsf{Alice}} \otimes (B_a^x)_{\mathsf{Bob}} \tag{61}$$

on state $|\text{answer}\rangle$ *and distribution* $\mathcal{D}$*. Next, let* $\{\mathsf{Q}_x\}_x$ *be a measurement and* $|\text{question}\rangle$ *be a bipartite state such that*

$$(\mathsf{Q}_x)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_\delta I_{\mathsf{Alice}} \otimes (\mathsf{Q}_x)_{\mathsf{Bob}}$$

on $|\text{question}\rangle$*. Furthermore, suppose that the distribution of measurement outcomes produced by measuring* $\{(\mathsf{Q}_x)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}\}_x$ *on* $|\text{question}\rangle$ *is* $\mathcal{D}$*. Then*

$$(\mathsf{Q}_x \otimes A_a^x)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_\delta I_{\mathsf{Alice}} \otimes (\mathsf{Q}_x \otimes B_a^x)_{\mathsf{Bob}} \tag{62}$$

on state $|\text{question}\rangle \otimes |\text{answer}\rangle$*.*

12.2 Subroutines and superregisters

In the next few sections, we will design a set of protocols to be used as a subroutine for our main NEEXP protocol. In doing so, we will encounter the following notational difficulty: a subroutine $\mathscr{G}$ might be a $\lambda = (k, n, q)$-register game, whereas the overall protocol which calls it might be a $\mu = (\ell, m, q)$-register game. When λ is not equal to μ, how can we use $\mathscr{G}$? We will consider two answers to this question. In both of them, we will consider the case when all the register field sizes are the same value "q", as this is the case relevant to our application.

Notation 12.5. First, the registers in λ might appear as a subset of the registers in μ. In this case, we will specify an injection $\kappa : [k] \to [\ell]$ such that $n_i = m_{\kappa(i)}$. Given a string $W = (W_1, \ldots, W_k)$, we write $\kappa(W)$ for the length-ℓ string with $W_{\kappa(i)}$ in coordinate i, for each i, and the "hide" symbol H in the remaining coordinates. Similarly, given string $a = (a_1, \ldots, a_\ell)$, we write $\kappa^{-1}(a)$ for the length-k string with $a_{\kappa(i)}$ in its i-th coordinate. Then *playing $\mathscr{G}$ on registers $\kappa(1), \ldots, \kappa(k)$* means to do the following.

1. Draw $(\boldsymbol{x}, \boldsymbol{x}')$ from $\mathscr{G}$.
2. Send $(\kappa(\boldsymbol{x}_1), \boldsymbol{x}_2)$ to Alice and $(\kappa(\boldsymbol{x}'_1), \boldsymbol{x}'_2)$ to Bob.
3. Receive $\boldsymbol{a}, \boldsymbol{a}'$. Accept if $\mathscr{G}$ accepts on the answers $(\kappa^{-1}(\boldsymbol{a}_1), \boldsymbol{a}_2)$ and $(\kappa^{-1}(\boldsymbol{a}'_1), \boldsymbol{a}'_2)$.

Notation 12.6. Second, the registers in λ might appear as the concatenation of register in μ. In this case, we will specify concatenation lengths $c(1) + \cdots + c(k) = \ell$ such that $n_1 = m_1 + \cdots + m_{c(1)}$, $n_2 = m_{c(1)+1} + \cdots + m_{c(1)+c(2)}$. Pictorially, the first register in λ will be created as the following concatenation:

$$\underbrace{|\mathrm{EPR}_q^{n_1}\rangle \otimes |\mathrm{EPR}_q^{n_2}\rangle \otimes \cdots \otimes |\mathrm{EPR}_q^{n_{c(1)-1}}\rangle \otimes |\mathrm{EPR}_q^{n_{c(1)}}\rangle}_{|\mathrm{EPR}_q^{n_1+\cdots+n_{c(1)}}\rangle}.$$

We refer to these concatenations of registers as *superregisters*. A Pauli basis query $W \in \{X, Z, H, \perp\}$ to a given superregister R can be simulated as follows:

1. Implement each Pauli basis query W by sending W to each register $r_{i_1}, \ldots, r_{i_c}$ in the superregister.
2. If $W \in \{X, Z\}$, the prover measures $\tau_{u_i}^W$ on each register r_i in R, and the verifier receives the outcomes $\boldsymbol{u}_{i_1} \in \mathbb{F}_q^{m_{i_1}}, \ldots, \boldsymbol{u}_{i_c} \in \mathbb{F}_q^{m_{i_c}}$, concatenated as $\boldsymbol{u} = (\boldsymbol{u}_{i_1}, \ldots, \boldsymbol{u}_{i_c})$.
3. If $W = H$, the prover performs $\underbrace{I \otimes \ldots \otimes I}_{c}$ on the registers in the superregister, and verifier receives c consecutive $\varnothing$'s, interpreted as a single $\varnothing$.
4. If $W = \perp$, the prover may perform any measurement it likes on the registers in the superregister. The verifier receives c consecutive $\varnothing$'s, interpreted as a single $\varnothing$.

The game $\mathscr{G}$ will usually be proven sound against λ-register strategies, but in our cases it will be straightforward to extend this soundness to μ-register strategies in the case when $\mathscr{G}$ is applied as a subroutine as detailed above. For example, suppose we know that a strategy A which passes $\mathscr{G}$ with probability $1 - \epsilon$ must satisfy $(A_a)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_\delta (B_a)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}$. Then it is straightforward to derive that if $\mathscr{G}$ is played as a subroutine on the second register of μ (this is the case when $k = 1$ and $n_1 = m_2$), and if A passes the subroutine with probability $1 - \epsilon$, then

$$(A_a)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_\delta (I_1 \otimes (B_a)_2 \otimes I_{3,\ldots,\ell})_{\mathsf{Alice}.} \otimes I_{\mathsf{Bob}}$$

Likewise, suppose $\mathscr{G}$ is played as a subroutine on one superregister consisting of all the registers of μ (this is the case when $k = 1$ and $n_1 = m_1 + \cdots + m_\ell$). If A passes the subroutine with probability $1 - \epsilon$, then

$$(A_a)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_\delta ((B_a)_{1,\ldots,\ell})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}.$$

For our applications, it will be straightforward to extend the soundness of our games to the case when they are played as subroutines, and we will leave this step implicit in our proofs.

13 The introspective low-degree test

In this section, we give the introspective low-degree test. This is an introspection game which simulates the classic surface-versus-point test, but is able to reduce the question complexity by making the provers sample the questions themselves. We allow for a fully general k-dimensional surface, though in our application we will only use $k = 1$ (lines) and $k = 2$ (planes).

Given an integer $n > 0$ and a power of two q, the introspective low-degree test is a $(k+1, n, q)$-register game. In other words, the provers share a state of the following form:

$$|\psi\rangle = |\mathrm{EPR}_q^n\rangle_{\mathsf{0}} \otimes |\mathrm{EPR}_q^n\rangle_{\mathsf{1}} \otimes \cdots \otimes |\mathrm{EPR}_q^n\rangle_{\mathsf{k}} \otimes |\mathrm{aux}\rangle_{\mathsf{aux}}.$$

The intended behavior is this: the "points" prover should measure the point $\boldsymbol{u} \in \mathbb{F}_q^n$ from register 0. The "surface" prover should measure directions $\boldsymbol{v} = \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k\}$ from registers 1 through k and then should receive their surface $\boldsymbol{s}$ from the surface measurement $\{\Pi_s^{\boldsymbol{v}}\}_{s \in \mathsf{Surfaces}_{\boldsymbol{v}}}$ on register 0. If the provers act honestly, then $\boldsymbol{u}$ will be a uniformly random point in $\boldsymbol{s}$, generating the same distribution as the questions in the surface-versus-point low-degree test.

The key difficulty is preventing the surface prover from fully measuring the register 0 and thus learning the value of the point $\boldsymbol{u}$. In this section, we design a test to enforce this behavior on the surface prover, using an introspected version of the partial data-hiding game developed in Section 10. This game lets us command the surface prover to erase all information about $\boldsymbol{u}$ except its value modulo linear combinations of the directions $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k$; we give it in Section 13.1 below. We use this test in Section 13.4 to design the introspective low-degree test and prove its soundness.

13.1 Introspected partial data-hiding

In this section, we give an introspected version of the partial data-hiding game, which will be used to implement the surface and intercept-scrambling measurements described above.

Definition 13.1. Let $k, n > 0$ be integers, let q be a power of 2, and let $\lambda = (k+1, n, q)$ be register parameters. Let x be an arbitrary query. Then the *introspected partial data-hiding game* $\mathscr{G}_{\mathrm{IntroHide}}(\lambda, x)$ is given in Figure 6.

The performance of the introspected partial data-hiding game is given in the following theorem.

Theorem 13.2. *Let $k, n > 0$ be integers, let q be a power of 2, and let $\lambda = (k+1, n, q)$ be register parameters. Let x be an arbitrary query. Then $\mathscr{G}_{\mathrm{IntroHide}} := \mathscr{G}_{\mathrm{IntroHide}}(\lambda, x)$ satisfies the following two properties.*

- ***Completeness:*** *Let $\mathcal{S}_{\mathrm{partial}} = (\psi, M^{\perp,Z,\ldots,Z,x})$ be a partial λ-register strategy for $\mathscr{G}_{\mathrm{IntroHide}}$, which is also a real commuting EPR strategy, and for which*

$$M^{\perp,Z,\ldots,Z,x}_{\varnothing,v_1,\ldots,v_k,a_2} = \sum_{s \in \mathsf{Surfaces}_v} \Pi_s^v \otimes \tau_{v_1}^Z \otimes \ldots \otimes \tau_{v_k}^Z \otimes A_{a_2}^{x,s,v},$$

Flip an unbiased coin $\boldsymbol{b} \sim \{0,1\}$, and perform one of the following three tests with probability 1/3 each.

1. Distribute the questions as follows:
   - Player $\boldsymbol{b}$: Give $(\bot, \underbrace{Z, \ldots, Z}_{k}, x)$; receive $(\varnothing, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_k, \boldsymbol{a}_2)$.
   - Player $\overline{\boldsymbol{b}}$: Give $(Z, \underbrace{Z, \ldots, Z}_{k}, x)$; receive $(\boldsymbol{a}'_1, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_k, \boldsymbol{a}'_2)$.

   Accept if $\boldsymbol{a}_2 = \boldsymbol{a}'_2$
2. Distribute the questions as follows:
   - Player $\boldsymbol{b}$: Give $(\bot, \underbrace{Z, \ldots, Z}_{k}, x)$; receive $(\varnothing, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_k, \boldsymbol{a}_2)$.
   - Player $\overline{\boldsymbol{b}}$: Give $(\bot, \underbrace{Z, \ldots, Z}_{k}, x, \{X\})$; receive $(\varnothing, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_k, \boldsymbol{a}'_2, \{\boldsymbol{a}'_{1,1}, \ldots, \boldsymbol{a}'_{1,k}\})$.

   Accept if $\boldsymbol{a}_2 = \boldsymbol{a}'_2$.
3. Distribute the questions as follows:
   - Player $\boldsymbol{b}$: Give $(X, \underbrace{Z, \ldots, Z}_{k}, \varnothing)$; receive $(\boldsymbol{a}_1, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_k, \varnothing)$.
   - Player $\overline{\boldsymbol{b}}$: Give $(\bot, \underbrace{Z, \ldots, Z}_{k}, \bot, \{X\})$; receive $(\varnothing, \boldsymbol{v}'_1, \ldots, \boldsymbol{v}'_k, \varnothing, \{\boldsymbol{a}'_{1,1}, \ldots \boldsymbol{a}'_{1,k}\})$.

   Accept if $\boldsymbol{v}_i = \boldsymbol{v}'_i$ and $\boldsymbol{a}'_{1,i} = \boldsymbol{v}_i \cdot \boldsymbol{a}_1$ for all $i \in \{1, \ldots, k\}$.
4. Distribute the questions as follows:
   - Player $\boldsymbol{b}$: Give $(\bot, x, \underbrace{Z, \ldots, Z}_{k}, \{X\})$; receive $(\varnothing, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_k, \boldsymbol{a}_2, \{\boldsymbol{a}_{1,1}, \ldots, \boldsymbol{a}_{1,k}\})$.
   - Player $\overline{\boldsymbol{b}}$: give $(\bot, \bot, \underbrace{Z, \ldots, Z}_{k}, \{X\})$; receive $(\varnothing, \boldsymbol{v}'_1, \ldots, \boldsymbol{v}'_k \varnothing, \{\boldsymbol{a}'_{1,1}, \ldots, \boldsymbol{a}'_{1,k}\})$.

   Accept if $\boldsymbol{a}_{1,i} = \boldsymbol{a}'_{1,i}$ for all $i \in \{1, \ldots, k\}$.

Figure 6: The introspected partial data-hiding game $\mathscr{G}_{\text{IntroHide}}(\lambda, x)$.

for some measurement $A_{a_2}^{x,s,v}$ acting only on the aux *register. Then there is a value-1 λ-register strategy for $\mathscr{G}_{\text{IntroHide}}$ extending $\mathcal{S}_{\text{partial}}$ which is also a real commuting EPR strategy.*

- ***Soundness:*** *Let $\mathcal{S} = (\psi, M)$ be a projective λ-register strategy passing $\mathscr{G}_{\text{IntroHide}}$ with probability at least $1-\epsilon$. Then there exists an ideal measurement $M'^{x}_{v_1,\ldots,v_k,a_2}$ with the property that*

$$M'^{x}_{v_1,\ldots,v_k,a_2} = \tau^Z_{v_1} \otimes \ldots \otimes \tau^Z_{v_k} \otimes \left(\sum_{s \in \mathsf{Surfaces}_v} \Pi^v_s \otimes (M^{x,s,v}_{a_2})_{\mathsf{aux}} \right),$$

such that the measurement $M^{\perp,Z\ldots,Z,x}_{\varnothing,v_1,\ldots,v_k,a_2}$ used by strategy $\mathcal{S}$ in response to the query $(\perp, \underbrace{Z,\ldots,Z}_{k}, x)$ is close to $M'^{x}_{v_1,\ldots,v_k,a_2}$:

$$(M^{\perp,Z,\ldots,Z,x}_{a_2})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_\epsilon (M'^{x}_{a_2})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}.$$

Proof. We first show completeness. Very similarly to the non-introspected partial data-hiding game, we introduce measurements for the remaining questions as follows:

$$\begin{aligned}
M^{Z,Z,\ldots,Z,x}_{a_1,v_1,\ldots,v_k,a_2} &= \sum_{s \in \mathsf{Surfaces}_v} (\Pi^v_s \cdot \tau^Z_{a_1}) \otimes \tau^Z_{v_1} \otimes \ldots \otimes \tau^Z_{v_k} \otimes A^{x,s,v}_{a_2}, \\
M^{X,Z,\ldots,Z}_{a_1,v_1,\ldots,v_k} &= \tau^X_{a_1} \otimes \tau^Z_{v_1} \otimes \ldots \otimes \tau^Z_{v_k} \otimes I_{\mathsf{aux}}, \\
M^{\perp,Z,\ldots,Z,x,\{X\}}_{\varnothing,v_1,\ldots,v_k,a_2,\{a_{1,1},\ldots,a_{1,k}\}} &= \sum_{s \in \mathsf{Surfaces}_v} (\Pi^v_s \cdot \tau^X_{[\forall i, a_1 \cdot v_i = a_{1,i}]}) \otimes \tau^Z_{v_1} \otimes \ldots \otimes \tau^Z_{v_k} \otimes A^{x,s,v}_{a_2,} \\
M^{\perp,Z,\ldots,Z,\{X\}}_{\varnothing,v_1,\ldots,v_k,\varnothing,\{a_{1,1},\ldots,a_{1,k}\}} &= \tau^X_{[\forall i, a_1 \cdot v_i = a_{1,i}]} \otimes \tau^Z_{v_1} \otimes \ldots \tau^Z_{v_k}.
\end{aligned}$$

By essentially the same arguments as in the proof of Theorem 10.4, it follows that these measurements define a value-1 real commuting EPR strategy for $\mathscr{G}_{\text{IntroHide}}$.

We now show soundness. Suppose that the provers succeed in the game with probability $1-\epsilon$ using a λ-register strategy. From the definition of a register strategy, we know that the measurement operators used by the provers have the following form.

$$\begin{aligned}
M^{\perp,Z,\ldots,Z,x}_{\varnothing,v_1,\ldots,v_k,a_2} &= (A^{x,v_1,\ldots,v_k}_{a_2})_{\mathsf{1,aux}} \otimes \tau^Z_{v_1} \otimes \cdots \otimes \tau^Z_{v_k}, \\
M^{\perp,Z,\ldots,Z,x,\{X\}}_{\varnothing,v_1,\ldots,v_k,a_2,\{a'_{1,1},\ldots,a'_{1,k}\}} &= (B^{x,v_1,\ldots,v_k}_{a_2,\{a'_{1,1},\ldots,a'_{1,k}\}})_{\mathsf{1,aux}} \otimes \tau^Z_{v_1} \otimes \cdots \otimes \tau^Z_{v_k}, \\
M^{Z,Z\ldots,Z,x}_{a_1,v_1,\ldots,v_k,a_2} &= \tau^Z_{a_1} \otimes \tau^Z_{v_1} \otimes \cdots \otimes \tau^Z_{v_k} \otimes (C^{x,a_1,v_1,\ldots,v_k}_{a_2})_{\mathsf{aux}}, \\
M^{\perp,Z,\ldots,Z,\perp,\{X\}}_{\varnothing,v_1,\ldots,v_k,\varnothing,\{a_{1,1},\ldots,a_{1,k}\}} &= (D^{v_1,\ldots,v_k}_{a_{1,1},\cdots a_{1,k}})_{\mathsf{1,aux}} \otimes \tau^Z_{v_1} \otimes \ldots \otimes \tau^Z_{v_k}.
\end{aligned}$$

where the operators $\{A^{x,v_1,\ldots,v_k}_{a_2}\}$, $\{B^{x,v_1,\ldots,v_k}_{a_2,\{a'_{1,1},\ldots,a'_{1,k}\}}\}$, $\{C^{x,a_1,v_1,\ldots,v_k}_{a_2}\}$, and $\{D^{v_1,\ldots,v_k}_{a_{1,1},\ldots,a_{1,k}}\}$ form valid POVMs for every choice of $x, a_1, v_1, \ldots, v_k$. We further know that the shared state of the provers is of the form

$$|\psi\rangle = |\mathrm{EPR}^n_q\rangle_{\mathsf{0}} \otimes |\mathrm{EPR}^n_q\rangle_{\mathsf{1}} \otimes \cdots \otimes |\mathrm{EPR}^n_q\rangle_{\mathsf{k}} \otimes |\mathrm{aux}\rangle_{\mathsf{aux}}.$$

From success in the four parts of the test, we may also deduce the following conditions:

$$\begin{aligned}
(M^{\perp,Z,\ldots,Z,x}_{a_2,v_1,\ldots,v_k})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} &\simeq_\epsilon I_{\mathsf{Alice}} \otimes (M^{Z,Z,\ldots,Z,x}_{v_1,\ldots,v_k,a_2})_{\mathsf{Bob}}, \\
(M^{\perp,Z,\ldots,Z,x}_{a_2,v_1,\ldots,v_k})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} &\simeq_\epsilon I_{\mathsf{Alice}} \otimes (M^{\perp,Z,\ldots,Z,x,\{X\}}_{a_2,v_1,\ldots,v_k}), \\
(M^{\perp,Z,\ldots,Z,\perp,\{X\}}_{v_1,\ldots v_k,\{a_{1,1},\ldots,a_{1,k}\}})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} &\simeq_\epsilon I_{\mathsf{Alice}} \otimes (\tau^X_{[\forall i, a_1 \cdot v_i = a_{1,i}]} \otimes \tau^Z_{v_1} \otimes \ldots \otimes \tau^Z_{v_k} \otimes I_{\mathsf{aux}})_{\mathsf{Bob}}, \\
(M^{\perp,Z,\ldots,Z,\perp,\{X\}}_{v_1,\ldots,v_k,\{a_{1,1},\ldots,a_{1,k}\}})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} &\simeq_\epsilon I_{\mathsf{Alice}} \otimes (M^{\perp,Z,\ldots,Z,x,\{X\}}_{v_1,\ldots,v_k,a_2,\{a_{1,1},\ldots,a_{1,k}\}})_{\mathsf{Bob}}.
\end{aligned}$$

We would now like to argue that the operators A, B, C, D form a good strategy for the partial data-hiding game. By Fact 12.3, it holds that under the uniform distribution over $v_1, \ldots, v_k$,

$$(A_{a_2}^{x,v_1,\ldots,v_k})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_\epsilon I_{\mathsf{Alice}} \otimes \Big(\sum_{a_1} \tau_{a_1}^Z \otimes C_{a_2}^{x,a_1,v_1,\ldots,v_k}\Big)_{\mathsf{Bob}},$$

$$(A_{a_2}^{x,v_1,\ldots,v_k})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_\epsilon I_{\mathsf{Alice}} \otimes \big(B_{a_2}^{x,v_1,\ldots,v_k}\big)_{\mathsf{Bob}},$$

$$(B_{\{a_{1,1},\ldots,a_{1,k}\}}^{x,v_1,\ldots,v_k})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq I_{\mathsf{Alice}} \otimes (D_{\{a_{1,1},\ldots,a_{1,k}\}}^{v_1,\ldots,v_k})_{\mathsf{Bob}}$$

$$(D_{\{a'_{1,1},\ldots,a'_{1,k}\}}^{v_1,\ldots,v_k})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_\epsilon I_{\mathsf{Alice}} \otimes (\tau^X_{[a_1\cdot v_1 = a'_{1,1},\ldots,a_1\cdot v_k = a'_{1,k}]} \otimes I_{\mathsf{aux}})_{\mathsf{Bob}},$$

as well as the same conditions with the Alice and Bob registers exchanged.

These are precisely the conditions of winning the partial data-hiding game $\mathscr{G}_{\text{hide}}(S, x)$, where S is the set of all tuples $v_1, \ldots, v_k$ in $\mathbb{F}_q^n$, with probability $1 - O(\epsilon)$. Hence, by Theorem 10.4, it follows that there exists a measurement $A'^{x,v_1,\ldots,v_k}_{a_2}$ such that

$$A'^{x,v_1,\ldots,v_k}_{a_2} = \sum_{s \in \mathsf{Surfaces}_v} \Pi_s^v \otimes A_a^{s,x,v},$$

$$(A_{a_2}^{x,v_1,\ldots,v_k})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_\epsilon (A'^{x,v_1,\ldots,v_k}_{a_2})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}.$$

The operator M' in the conclusion of the theorem can then be taken to be

$$M'^{\perp,Z,\ldots,Z,x}_{a_2,v_1,\ldots,v_k} = (A'^{x,v_1,\ldots,v_k}_{a_2}) \otimes \tau^Z_{v_1} \otimes \ldots \otimes \tau^Z_{v_k}. \qquad \square$$

13.2 An introspective surface sampler

In this section, we will use the introspective data hiding game to implement the "surface prover". This is a prover who samples a surface $\boldsymbol{s}$ from register 0 using the $\Pi^{\boldsymbol{v}}$ measurement and then reports back $\boldsymbol{s}$ to the verifier, along with a degree-d polynomial $\boldsymbol{f} : \boldsymbol{s} \to \mathbb{F}_q$. As above, the prover is expected *not* to measure register 0 any further, so that $\boldsymbol{f}$ depends only on $\boldsymbol{s}$ and $\boldsymbol{v}$. We can enforce this by running the introspective data hiding game and interpreting the provers' answers as $\boldsymbol{a}_2 = \{\boldsymbol{s}, \boldsymbol{f}\}$. However, we must also verify that $\boldsymbol{s}$ corresponds to the actual surface measured by the prover in the 0-th register and not some other surface. We do this with a slight modification to the introspective data hiding game we call the "introspective surface sampling game".

Definition 13.3. Let $k, n, d > 0$ be integers, let q be a power of 2, and let $\lambda = (k+1, n, q)$ be register parameters. Then the *introspective surface sampling game* $\mathscr{G}_{\text{IntroSurfSamp}}(\lambda, d)$ is given in Figure 7.

- Play the game $\mathscr{G}_{\text{IntroHide}}(\lambda, x)$ with $x =$ "surface", and with the answer a_2 taking the form $\{\boldsymbol{s}, \boldsymbol{f}\}$, where $\boldsymbol{s}$ is a surface and $\boldsymbol{f}$ is a degree-d function $\boldsymbol{f} : \boldsymbol{s} \to \mathbb{F}_q$.
- Consider the test in Item 1 of $\mathscr{G}_{\text{IntroHide}}(\lambda, x)$. Here, Player $\boldsymbol{b}$ replies with the answer $(\varnothing, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_k, \{\boldsymbol{s}, \boldsymbol{f}\})$, and Player $\bar{\boldsymbol{b}}$ replies with $(\boldsymbol{a}'_1, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_k, \{\boldsymbol{s}', \boldsymbol{f}'\})$. In the case where this test is chosen, accept if $\mathscr{G}_{\text{IntroHide}}(\lambda, x)$ accepts and also if $\boldsymbol{s}$ is the surface $\{\boldsymbol{a}'_1 + \sum_{i=1}^k \lambda_i \boldsymbol{v}_i : \lambda_1, \ldots, \lambda_k \in \mathbb{F}_q\}$. (We call this additional check the "Correct Surface Check".) If this query is not given to the provers, then accept if $\mathscr{G}_{\text{IntroHide}}(\lambda, x)$ accepts.

Figure 7: The game $\mathscr{G}_{\text{IntroSurfSamp}}(\lambda, d)$.

Notation 13.4. In the case when a prover is given the question $(\bot, Z, \ldots, Z, \text{“surface”})$, we refer to it as the *surface prover*. It has the following intended behavior.

1. **Surface prover:**

   Input: Pauli basis queries $(\bot, \underbrace{Z, \ldots, Z}_{k})$ and an auxiliary query “surface”.

   Output: Pauli basis answers $\varnothing$ and $v_1, \ldots, v_k \in \mathbb{F}_q^n$, a k-dimensional surface s, and the coefficients of a degree-d polynomial function $f : \mathbb{F}_q^k \to \mathbb{F}_q$, where the domain of f is to interpreted as s.

   Goal: The prover measures Π^v on register 0 and sets s to be its outcome. They then set $f = g|_s$, where $g : \mathbb{F}_q^n \to \mathbb{F}_q$ is a global degree-d polynomial selected independently of s or v.

In the case when $k = 1$, we may also refer to it as the *lines* prover, and in the case when $k = 2$, we may also refer to it as the *planes* prover. We will also refer to the *surface prover's measurement*, which refers to the measurement $\{A_{v_1,\ldots,v_k,s,f}\}$ given by

$$A_{v_1,\ldots,v_k,s,f} = M^{\bot,Z,\ldots,Z,\text{“surface”}}_{\varnothing,v_1,\ldots,v_k,s,f}.$$

The following theorem shows that the introspective surface sampling game forces the surface prover to output the correct surface s.

Theorem 13.5. *Let $k, n, d > 0$ be integers, let q be a power of 2, and let $\lambda = (k+1, n, q)$ be register parameters. Write $\{A_{v_1,\ldots,v_k,s,f}\}$ for the surface prover's measurement. Then $\mathscr{G}_{\text{IntroSurfSamp}} := \mathscr{G}_{\text{IntroSurfSamp}}(\lambda, d)$ has the following two properties.*

- ***Completeness:*** *Suppose there is a degree-d polynomial $g : \mathbb{F}_q^n \to \mathbb{F}_q$ such that*

  $$A_{v_1\ldots,v_k,s,f} = \Pi^v_s \otimes \tau^Z_{v_1} \otimes \cdots \otimes \tau^Z_{v_k} \otimes I_{\mathsf{aux}} \cdot \mathbf{1}[f = g|_s].$$

  Then there is a value-1 λ-register strategy for $\mathscr{G}_{\text{IntroSurfSamp}}$ with A as the surface prover's measurement.

- ***Soundness:*** *Let $\mathcal{S}$ be a projective λ-register strategy which passes $\mathscr{G}_{\text{IntroSurfSamp}}$ with probability at least $1 - \epsilon$. Then there exists an ideal measurement A' of the form*

  $$A'_{v,s,f} = \Pi^v_s \otimes \tau^Z_{v_1} \otimes \ldots \otimes \tau^Z_{v_k} \otimes (M^{s,v}_f)_{\mathsf{aux}},$$

  with $M^{s,v}_f$ an arbitrary measurement on the aux register, such that A' is close to the surface provers' measurement A in $\mathcal{S}$:

  $$(A_{v,s,f})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_{\text{poly}(\epsilon)} (A'_{v,s,f})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}.$$

  In particular, the surface output by A' is the same surface measured by A' in register 0.

Proof. First, the completeness follows immediately from the completeness guarantee of Theorem 13.2.

Next, we show soundness. Passing with probability $1 - \epsilon$ implies passing $\mathscr{G}_{\text{IntroHide}}(\lambda, \text{“surface”})$ with probability $1 - \epsilon$. By Theorem 13.2, this implies an ideal measurement $M'^x_{v_1,\ldots,v_k,s,f}$ with the property that

$$M'_{v_1,\ldots,v_k,s,f} = \tau^Z_{v_1} \otimes \cdots \otimes \tau^Z_{v_k} \otimes \left(\sum_{s' \in \mathsf{Surfaces}_v} \Pi^v_{s'} \otimes (M^{s',v}_{s,f})_{\mathsf{aux}} \right),$$

Flip an unbiased coin $\boldsymbol{b} \sim \{0,1\}$. Distribute the questions as follows.

- Player $\boldsymbol{b}$: Give $(\perp, \underbrace{Z, \ldots, Z}_{k}$, "surface"); receive $(\varnothing, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_k, \boldsymbol{s}, \boldsymbol{f})$.
- Player $\overline{\boldsymbol{b}}$: Give $(Z, \underbrace{H, \ldots, H}_{k}$, "point"); receive $(\boldsymbol{u}, \underbrace{\varnothing, \ldots, \varnothing}_{k}, \boldsymbol{\nu})$, where $\boldsymbol{\nu} \in \mathbb{F}_q$.

Accept if $\boldsymbol{f}(\boldsymbol{u}) = \boldsymbol{\nu}$.

Figure 8: The game $\mathscr{G}_{\text{IntroCross}}(\lambda, d)$.

$$(A_{v_1,\ldots,v_k,s,f})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_\epsilon (M'_{v_1,\ldots,v_k,s,f})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}.$$

(Note that the measured surface s' in M' is allowed to be different than the output surface s.) Next, set $B_{v_1,\ldots,v_k,s} := \Pi_s^v \otimes \tau_{v_1}^Z \otimes \cdots \otimes \tau_{v_k}^Z \otimes I_{\mathsf{aux}}$. Then passing the Correct Surface Check with probability $1 - O(\epsilon)$ implies that

$$(A_{v,s})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_\epsilon I_{\mathsf{Alice}} \otimes (B_{v,s})_{\mathsf{Bob}}.$$

Note that $M'_{v,s,f}$ and $B_{v,s}$ commute. Thus, if we define $C_{v,s,f} := M'_{v,s,f} \cdot B_{v,s} = B_{v,s} \cdot M'_{v,s,f}$, then by Fact 4.20,

$$\begin{aligned}
(A_{v,s,f})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} &= (A_{v,s,f} \cdot A_{v,s})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \\
&\approx_\epsilon (A_{v,s,f})_{\mathsf{Alice}} \otimes (B_{v,s})_{\mathsf{Bob}} \\
&\approx_\epsilon (M'_{v,s,f})_{\mathsf{Alice}} \otimes (B_{v,s})_{\mathsf{Bob}} \\
&\approx_\epsilon (M'_{v,s,f} \cdot B_{v,s})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \\
&= (C_{v,s,f})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}},
\end{aligned}$$

where the second-to-last step is by Fact 4.38. Now, set $C_f^{s,v} := M_{s,f}^{s,v}$. Then we can write

$$C_{v_1,\ldots,v_k,s,f} = \Pi_s^v \otimes \tau_{v_1}^Z \otimes \cdots \otimes \tau_{v_k}^Z \otimes (C_f^{s,v})_{\mathsf{aux}}.$$

These matrices are almost of the form guaranteed by the theorem, except they do not necessarily form a measurement because the matrices $C_f^{s,v}$ do not necessarily sum to the identity. However, this is still sufficient to imply the theorem by Fact 4.25. □

13.3 The introspective cross-check

In this section, we introduce the other subroutine in the introspective low-degree test. In this subroutine, known as the "introspective cross-check", we introduce a new prover known as the "points prover". This is a prover who samples a point $\boldsymbol{u}$ from register 0 and then reports back a value $\boldsymbol{\nu} \in \mathbb{F}_q$ interpreted as their assignment to the point $\boldsymbol{u}$. By data hiding, we can assume the points prover does not read registers 1 through k. Then the introspective cross-check queries the points prover and the surface prover and checks that their outputs agree on the point $\boldsymbol{u}$.

Definition 13.6. Let $k, n, d > 0$ be integers, let q be a power of 2, and let $\lambda = (k+1, n, q)$ be register parameters. The *introspective cross-check*, denoted $\mathscr{G}_{\text{IntroCross}}(\lambda, d)$, is defined in Figure 8.

Notation 13.7. In the case when a prover is given the question $(Z, H, \ldots, H$, "point"), we refer to it as the *points prover*. It has the following intended behavior.

2. **Points prover:**

   Input: Pauli basis queries $(Z, H, \ldots, H)$ and auxiliary query "point".

   Output: String $u \in \mathbb{F}_q^n$ and $\varnothing, \ldots, \varnothing$. A number $\nu \in \mathbb{F}_q$.

   Goal: The prover sets $\nu = g(u)$, where $g : \mathbb{F}_q^n \to \mathbb{F}_q$ is a global degree-d polynomial selected independently of u.

We will also refer to the *point prover's measurement*, which refers to the measurement $\{B_{u,\nu}\}$ given by

$$B_{u,\nu} = M_{u,\varnothing,\ldots,\varnothing,\nu}^{Z,H,\ldots,H,\text{“point”}}.$$

Our next lemma shows that if the surface prover's measurement for $\boldsymbol{f}$ depends only on the surface $\boldsymbol{s}$ and directions $\boldsymbol{v}$ and not on the point $\boldsymbol{u}$, then we can relate the value of the introspective cross-check to its non-introspected variant, $\mathscr{G}_{\text{surface}}$.

Lemma 13.8. *Let $k, n, d > 0$ be integers, let q be a power of 2, and let $\lambda = (k+1, n, q)$ be register parameters. Let $\mathcal{S}$ be a λ-register strategy for $\mathscr{G}_{\text{IntroCross}} := \mathscr{G}_{\text{IntroCross}}(\lambda, d)$. Let $\{A_{v,s,f}\}$ be the surface prover's measurement and $\{B_{u,\nu}\}$ be the point prover's measurement, and write $|\text{aux}\rangle$ for the auxiliary state. Suppose*

$$A_{v,s,f} = \Pi_s^v \otimes \tau_{v_1}^Z \otimes \cdots \otimes \tau_{v_k}^Z \otimes A_f^{s,v},$$
$$B_{u,\nu} = \tau_u^Z \otimes \underbrace{I \otimes \ldots \otimes I}_{k} \otimes B_\nu^u,$$

where $\{A_f^{s,v}\}$ and $\{B_\nu^u\}$ are POVM measurements on the auxiliary register. Consider the strategy $\mathcal{S}_{\text{surface}} = (\text{aux}, \{A^{s,v}, B^u\})$ for the game $\mathscr{G}_{\text{surface}} := \mathscr{G}_{\text{surface}}(n, q, k, d)$. Then

$$\text{val}_{\mathscr{G}_{\text{surface}}}(\mathcal{S}_{\text{surface}}) = \text{val}_{\mathscr{G}_{\text{IntroCross}}}(\mathcal{S}).$$

Proof. By the definition of Π_s^v and τ_u^Z, it follows that for any choice of k vectors v, if Alice and Bob each measure their half of register 0 using the measurements Π_s^v and τ_u^Z, respectively, then the measurement outcomes obtained will be pairs $(\boldsymbol{s}, \boldsymbol{u})$ where $\boldsymbol{s}$ is a uniformly random surface in $\mathsf{Surfaces}_{\boldsymbol{v}}$ and $\boldsymbol{u}$ is a uniformly random point in $\boldsymbol{s}$. Moreover, if Alice measures her half of registers 1 through k, she will obtain a uniformly random k-tuple $\boldsymbol{v} = \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k\} \subseteq \mathbb{F}_q^n$. Combining these facts, we see that if Alice measures her half of registers 0 through k with $\Pi_s^v \otimes \tau_{v_1}^Z \otimes \ldots \otimes \tau_{v_k}^Z$, and Bob measure his half with $\tau_u^Z \otimes \underbrace{I \otimes \ldots \otimes I}_{k}$, they obtain a pair of outcomes $(\boldsymbol{x}_A = (\boldsymbol{s}, \boldsymbol{v}), \boldsymbol{x}_B = \boldsymbol{u})$ distributed exactly according to the question distribution in $\mathscr{G}_{\text{surface}}$. Thus, applying Fact 12.2, we conclude that $\text{val}_{\mathscr{G}_{\text{surface}}}(\mathcal{S}_{\text{surface}}) = \text{val}_{\mathscr{G}_{\text{IntroCross}}}(\mathcal{S})$. □

13.4 The introspective low-degree test

In this section, we state the completed introspective low-degree test.

Definition 13.9. Let $k, n, d > 0$ be integers, let q be a power of 2, and let $\lambda = (k+1, n, q)$ be register parameters. The *introspective surface-versus-point low-degree test*, denoted $\mathscr{G} := \mathscr{G}_{\text{IntroLowDeg}}(\lambda, d)$, is defined in Figure 9. It has the following properties:

$$\mathsf{Q\text{-}length}(\mathscr{G}) = O(1), \quad \mathsf{A\text{-}length}(\mathscr{G}) = O(kn\log(q) + (d+k)^k \log(q)),$$
$$\mathsf{Q\text{-}time}(\mathscr{G}) = O(1), \quad \mathsf{A\text{-}time}(\mathscr{G}) = \text{poly}(kn\log(q), (d+k)^k \log(q)).$$

With probability $\frac{1}{2}$ each, perform one of the following three tests.

1. **Surface sampler test:** Play $\mathscr{G}_{\text{IntroSurfSamp}}(\lambda, d)$.
2. **Cross-check test:** Play $\mathscr{G}_{\text{IntroCross}}(\lambda, d)$.

Figure 9: The game $\mathscr{G}_{\text{IntroLowDeg}}(\lambda, d)$.

The question complexities are immediate. As for the answer length, the provers return $(k+1)$ elements of $\mathbb{F}_q^n$ and degree-d polynomials on k-surfaces, encoded as $\mathbb{F}_q$-valued strings of length $d[k] \leq (d+k)^k$. Finally, all operations made by the verifier, such as polynomial evaluation, are efficient, so the answer time complexity is polynomial in the answer length.

Naturally, we analyze the introspective low-degree test via introspection. This involves a reduction to the non-introspected version of the game, i.e. the "normal" surface-versus-point low-degree test. By Theorem 4.40 we know quantum soundness for this test in the $k = 2$ (i.e. planes) case. As a result, we get soundness for the introspective low-degree test in this case as well.

Theorem 13.10. *Fix $k = 2$. Let $n, d > 0$ be integers, let q be a power of 2, and let $\lambda = (k+1, n, q)$ be register parameters. Write $\mathscr{G} := \mathscr{G}_{\text{IntroLowDeg}}(\lambda, d)$.*

- ***Completeness:*** *Suppose there is a degree-d polynomial $g : \mathbb{F}_q^n \to \mathbb{F}_q$ such that*

$$B_{u,\nu} = \tau_u^Z \otimes I_1 \otimes I_2 \otimes I_{\mathsf{aux}} \cdot \mathbf{1}[\nu = g(u)].$$

  Then there is a value-1 λ-register real commuting EPR strategy for $\mathscr{G}$ with B as the point prover's measurement.

- ***Soundness:*** *There exists a constant $c > 0$ and a function $\delta(\epsilon) = \text{poly}(\epsilon, dm/q^c)$ such that the following holds. Suppose $\mathcal{S}$ is a projective λ-register strategy with value $1 - \epsilon$. Write $\{B_{u,\nu}\}$ for the point prover's measurement. Then there exists a POVM $\{G_g\}$ in $\text{PolyMeas}(n, d, q)$ such that*

$$(B_{u,\nu})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} (\tau_u^Z \otimes I_1 \otimes I_2 \otimes (G_{[g(u)=\nu]})_{\mathsf{aux}})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}.$$

Proof. Throughout this proof, we will write $\{A_{v,s,f}\}$ for the surface prover's measurement. We first show completeness. Assign the surface prover's measurement as follows:

$$A_{v,s,f} := \Pi_s^v \otimes \tau_{v_1}^Z \otimes \tau_{v_2}^Z \otimes I_{\mathsf{aux}} \cdot \mathbf{1}[f = g|_s].$$

This is clearly a λ-register strategy. By the completeness case of Theorem 13.5, this can be extended into a real commuting EPR strategy that passes $\mathscr{G}_{\text{IntroSurfSamp}}(\lambda, d)$ with probability 1. By Lemma 13.8, the strategy using A and B passes the cross check with the same probability as the honest classical strategy to $\mathscr{G}_{\text{surface}}(n, q, k, d)$ answering according to the low-degree polynomial g, which is 1. Hence, this strategy passes both parts of $\mathscr{G}$ with probability 1.

Now, we show soundness. The strategy $\mathcal{S}$ is a λ-register strategy, so we can write the points prover's measurement as

$$B_{u,\nu} = \tau_u^Z \otimes I_1 \otimes I_2 \otimes (B_\nu^u)_{\mathsf{aux}}.$$

Passing $\mathscr{G}_{\text{IntroSurfSamp}}(\lambda, d)$ with probability $1 - 2\epsilon$ implies via Theorem 13.5 a measurement $A'_{u,v,f}$ such that

$$A'_{v,s,f} = \Pi_s^v \otimes \tau_{v_1}^Z \otimes \tau_{v_2}^Z \otimes ((A')_f^{s,v})_{\mathsf{aux}},$$

$$(A_{v,s,f})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_{\mathrm{poly}(\epsilon)} (A'_{v,s,f})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}},$$

where $\{(A')_f^{s,v}\}_f$ is a measurement on the auxiliary register. By assumption, the measurement $\{A_{v,s,f}\}$ is projective, so we can apply Fact 4.31 to deduce that replacing $A_{v,s,f}$ by $A'_{v,s,f}$ changes the game value by at most $\mathrm{poly}(\epsilon)$. Moreover, by applying Theorem 4.2, we can, by performing a dilation of the auxiliary space, simulate the $A'_{v,s,f}$ measurements by a projective measurement of the form

$$A''_{v,s,f} = \Pi_s^v \otimes \tau_{v_1}^Z \otimes \tau_{v_2}^Z \otimes ((A'')_f^{s,v})_{\mathsf{aux}},$$

where $(A'')_f^{s,v}$ is a projective measurement on the (expanded) aux register. (Note that a direct invocation of Naimark's theorem Theorem 4.1 would not have sufficed as the dilated measurement would not necessarily act as desired on the non-aux registers.) Using the dilated A'' measurements instead of A' does not change the value of the game. Thus, we deduce that the projective strategy using measurements $B_{u,\nu}$ and $A''_{v,s,f}$ passes $\mathscr{G}_{\mathrm{IntroCross}}(\lambda, d)$ with probability $1 - \mathrm{poly}(\epsilon)$.

Now we are in a position to reduce to the soundness of the non-introspective game. Define the strategy $\mathcal{S}_{\mathrm{Plane}} := (\mathrm{aux}, \{B^u, (A'')^{s,v}\})$. Then by Lemma 13.8, $\mathcal{S}_{\mathrm{Plane}}$ also passes $\mathscr{G}_{\mathrm{Plane}}$ with probability $1 - \mathrm{poly}(\epsilon)$. Applying Theorem 4.40, we have that there exists a measurement $\{G_g\}$ in $\mathrm{PolyMeas}(n, d, q)$ such that

$$B_\nu^u \otimes I \approx_{\delta(\epsilon)} G_{[\nu = g(u)]} \otimes I$$

on state $|\mathrm{aux}\rangle$.

The theorem then follows from Fact 12.3. □

13.5 The introspective simultaneous low-degree test

In this section, we extend the introspective low-degree test to handle multiple functions at once. This is the introspective version of the simultaneous low-degree test from Definition 3.13.

Definition 13.11. Let $m \geq 1$. Let $k, n, d > 0$ be integers, let q be a power of 2, and let $\lambda = (k+1, n, q)$ be register parameters. The *introspective simultaneous low-degree test*, denoted $\mathscr{G} := \mathscr{G}_{\mathrm{IntroLowDeg}}(\lambda, d, m)$, is defined by the following modifications to the introspective low-degree test. First, the prover roles are modified as follows.

- **Surface prover:** Rather than returning a function $f : s \to \mathbb{F}_q$, it should return m functions $f_1, \ldots, f_m : s \to \mathbb{F}_q$. The intent is that $f_i = g_i|_s$ for each i, where each $g_i : \mathbb{F}_q^n \to \mathbb{F}_q$ is a global degree-d polynomial selected independently of s or v.
- **Points prover:** Rather than returning a single number $\nu \in \mathbb{F}_q$, it should return m numbers $\nu_1, \ldots, \nu_m \in \mathbb{F}_q$. The intent is that $\nu_i = g_i(u)$ for each i, where each g_i is selected independently of u.

Next, the subroutines are modified as follows.

- **Introspective surface sampling game:** The answer $\boldsymbol{a}_2$ has the form $\boldsymbol{s}, \boldsymbol{f}_1, \ldots, \boldsymbol{f}_m$ (rather than $\boldsymbol{s}, \boldsymbol{f}$ for a single function $\boldsymbol{f}$).
- **Introspective cross-check:** Receive $\boldsymbol{f}_1, \ldots, \boldsymbol{f}_m : \boldsymbol{s} \to \mathbb{F}_q$ from the surface prover and $\boldsymbol{\nu}_1, \ldots, \boldsymbol{\nu}_m \in \mathbb{F}_q$ from the points prover (rather than a single $\boldsymbol{f}$ and $\boldsymbol{\nu}$). Check that $\boldsymbol{f}_i(\boldsymbol{u}) = \boldsymbol{\nu}_i$ for all i.

It has the following properties:

$$\mathsf{Q\text{-}length}(\mathscr{G}) = O(1), \quad \mathsf{A\text{-}length}(\mathscr{G}) = O(kn\log(q) + m(d+k)^k \log(q)),$$
$$\mathsf{Q\text{-}time}(\mathscr{G}) = O(1), \quad \mathsf{A\text{-}time}(\mathscr{G}) = \mathrm{poly}(kn\log(q), m(d+k)^k \log(q)).$$

The following theorem gives the performance of the introspective simultaneous low-degree test in the case of $k = 2$ (i.e. planes).

Theorem 13.12. *Fix $k = 2$. Let $n, d, m > 0$ be integers, let q be a power of 2, and let $\lambda = (k+1, n, q)$ be register parameters. Write $\mathscr{G} := \mathscr{G}_{\mathrm{IntroLowDeg}}(\lambda, d, m)$.*

- ***Completeness:*** *Suppose there are degree-d polynomials $g_1, \ldots, g_m : \mathbb{F}_q^n \to \mathbb{F}_q$ such that*

$$B_{u,\nu_1,\ldots,\nu_m} = \tau_u^Z \otimes I_1 \otimes I_2 \otimes I_{\mathsf{aux}} \cdot \mathbf{1}[\forall i,\ \nu_i = g_i(u)].$$

  Then there is a value-1 λ-register real commuting EPR strategy for $\mathscr{G}$ with B as the point prover's measurement.

- ***Soundness:*** *There exists a constant $c > 0$ and a function $\delta(\epsilon) = \mathrm{poly}(\epsilon, d(n+m)/q^c)$ such that the following holds. Suppose $\mathcal{S}$ is a projective λ-register strategy with value $1 - \epsilon$. Write $\{B_{u,\nu_1,\ldots,\nu_m}\}$ for the point prover's measurement. Then there exists a POVM $\{G_{g_1,\ldots,g_m}\}$ in $\mathrm{PolyMeas}(n, d, q, m)$ such that*

$$(B_{u,\nu_1,\ldots,\nu_m})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} (\tau_u^Z \otimes I_1 \otimes I_2 \otimes (G_{[g_1(u),\ldots,g_m(u)=\nu_1,\ldots,\nu_m]})_{\mathsf{aux}})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}.$$

The proof, which we omit, is analogous to the proof of Theorem 13.10, except rather than reducing to Theorem 4.40, we reduce to the soundness of the non-introspective simultaneous low-degree test given by Theorem 4.43.

14 The intersecting lines test

The introspective low-degree test forces a prover to sample a point from a register and return the evaluation of a global function at that point. In our eventual protocol, we will want the prover to use the same global function to answer point queries sampled from *multiple* different registers. In this section, we design a game which allows us to "transfer" global functions used from one register to another. We keep in mind the following picture:

$$|\psi\rangle = |\mathrm{EPR}_q^n\rangle_1 \otimes |\mathrm{EPR}_q^n\rangle_2 \otimes |\mathrm{aux}\rangle_{\mathsf{aux}}.$$

We view register 1 as the register with the global function and register 2 as the register we would like to transfer this global function to.

To accomplish this, we introduce a new test called the "intersecting lines test". This involves performing two introspective line-versus-point low-degree tests. The first uses register 1 as its point register and register 2 as its slope register. This gives us a points prover who samples $\boldsymbol{u}$ from register 1 and returns a label on it and a line prover who samples $\boldsymbol{v}$ from register 2 and returns a function on the line $\{\boldsymbol{u} + \lambda \boldsymbol{v}\}$, and we know that if the points prover labels their point using a low-degree polynomial g, then the line prover must label their line with the same polynomial g. The second low-degree test uses register 2 as its point register and register 1 as its slope register. This gives a second line prover who returns a function on the line $\{\boldsymbol{v} + \lambda \boldsymbol{u}\}$. Noting that the point $\boldsymbol{u} + \boldsymbol{v}$ is contained in both line provers' lines, we can check consistency between their functions by comparing them on this point, forcing the second line prover to label their line using g as well. This then entails that the second line prover from the second low-degree test must also label their point $\boldsymbol{v}$ using g. Thus, we have successfully "transferred" the function g from the first register to the second.

In Section 14.1, we first introduce the intersecting lines test and prove soundness. Following that, in Section 14.2 we introduce an introspective version of this test which will later be used in our NEEXP protocol.

14.1 The intersecting lines test

Definition 14.1 (Intersecting lines test)**.** Let $n, d > 0$ be integers, and let q be a power of 2. The *intersecting lines test*, denoted $\mathscr{G}_{\text{intersect}}(n, q, d)$, is defined as follows. Sample $\boldsymbol{u}, \boldsymbol{v}$ uniformly at random from $\mathbb{F}_q^n$, and let $\boldsymbol{\ell}$ and $\boldsymbol{\ell}'$ be the two lines $\boldsymbol{\ell} := \{\boldsymbol{u}+\lambda\boldsymbol{v} : \lambda \in \mathbb{F}_q\}$ and $\boldsymbol{\ell}' := \{\boldsymbol{v}+\lambda\boldsymbol{u} : \lambda \in \mathbb{F}_q\}$. The test is performed as follows.

- The line $\boldsymbol{\ell}$ and $\boldsymbol{v}$ are given to Alice, who responds with a degree-d polynomial $\boldsymbol{f} : \boldsymbol{\ell} \to \mathbb{F}_q$.
- The line $\boldsymbol{\ell}'$ and $\boldsymbol{u}$ are given to Bob, who responds with a degree-d polynomial $\boldsymbol{f}' : \boldsymbol{\ell}' \to \mathbb{F}_q$.

Alice and Bob pass the test if $\boldsymbol{f}(\boldsymbol{u} + \boldsymbol{v}) = \boldsymbol{f}'(\boldsymbol{u} + \boldsymbol{v})$.

We begin by showing that although Bob knows $\boldsymbol{u}$, since he doesn't know $\boldsymbol{v}$, the point $\boldsymbol{u} + \boldsymbol{v}$ looks like a uniform point in $\boldsymbol{\ell}'$ to him.

Fact 14.2. *Conditioned on $\boldsymbol{\ell}'$ and $\boldsymbol{u}$, the point $\boldsymbol{u}+\boldsymbol{v}$ is distributed as a uniformly random element in $\boldsymbol{\ell}'$.*

Proof. Let $\boldsymbol{w}$ be a point in $\mathbb{F}_q^n$ such that $\boldsymbol{\ell}' = \{\boldsymbol{w} + \lambda\boldsymbol{u} : \lambda \in \mathbb{F}_q\}$. Then for any $c \in \mathbb{F}_q$, $\boldsymbol{\ell}'$ is also equal to the set $\{(\boldsymbol{w} + c\boldsymbol{u}) + (\lambda - c)\boldsymbol{u} : \lambda \in \mathbb{F}_q\}$. Hence, $\boldsymbol{v}$ is equally likely to be any element in the set $\{\boldsymbol{w} + c\boldsymbol{u} : c \in \mathbb{F}_q\}$, and therefore so is $\boldsymbol{u} + \boldsymbol{v}$. Since this set is also equal to $\boldsymbol{\ell}'$, this proves the fact. □

We will be interested in the case when Alice responds using a global function $g : \mathbb{F}_q^n \to \mathbb{F}_q$, always setting $\boldsymbol{f} = g|_{\boldsymbol{\ell}}$. In this case, the following lemma shows that to succeed with high probability, Bob must usually play the same global function as Alice.

Lemma 14.3. *Let (ψ, M) be a POVM strategy for the intersecting lines game with value $1 - \epsilon$. Suppose further that there is a measurement $\{G_g\}_g$ in* PolyMeas(n, d, q) *such that $M_f^{\ell,v} = G_{[g|_\ell = f]}$. Then*

$$(M_f^{\ell',u})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_{\epsilon+d/q} I_{\mathsf{Alice}} \otimes (G_{[g|_{\ell'}=f]})_{\mathsf{Bob}}.$$

Proof. Success on the test implies that

$$(G_{[g(u+v)=\nu]})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_\epsilon I_{\mathsf{Alice}} \otimes (M^{\ell',u}_{[f(u+v)=\nu]})_{\mathsf{Bob}}.$$

But by Fact 14.2, conditioned on $\boldsymbol{\ell}'$ and $\boldsymbol{u}$, $\boldsymbol{u}+\boldsymbol{v}$ is distributed as a uniformly random point in $\boldsymbol{\ell}'$. As a result, if we let $\boldsymbol{w}$ be a uniformly random point in $\boldsymbol{\ell}'$, then

$$(G_{[g(w)=\nu]})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_\epsilon I_{\mathsf{Alice}} \otimes (M^{\ell',u}_{[f(w)=\nu]})_{\mathsf{Bob}}.$$

The lemma then follows from Proposition 4.42. □

14.2 The introspective intersecting lines test

Now we introduce the introspective intersecting lines test. This will be an introspective version of the intersecting lines test.

Definition 14.4. Let $n, d > 0$ be integers, let q be a power of 2, and let $\lambda = (2, n, q)$ be register parameters. The *introspective intersecting lines test*, denoted $\mathscr{G}_{\text{IntroIntersect}}(\lambda, d)$, is a λ-register game involving two registers, named "1" and "2", and a possible third auxiliary register. It involves two line-versus point low-degree tests, instantiated as follows.

With probability $\frac{1}{4}$ each, perform one of the following four tests.

1. **Low degree test 1:** Play $\mathscr{G}_1$.
2. **Low degree test 2:** Play $\mathscr{G}_2$.
3. **Intersecting lines test:** Flip an unbiased coin $\boldsymbol{b} \sim \{0,1\}$. Assign the first role to Player $\boldsymbol{b}$ and the second role to Player $\overline{\boldsymbol{b}}$.
   - Lines_1: Receive $\boldsymbol{\ell}$, $\boldsymbol{v}$, $\boldsymbol{f} : \boldsymbol{\ell} \to \mathbb{F}_q$.
   - Lines_2: Receive $\boldsymbol{\ell}'$, $\boldsymbol{u}$, $\boldsymbol{f}' : \boldsymbol{\ell}' \to \mathbb{F}_q$.

   Accept if $\boldsymbol{\ell}$ and $\boldsymbol{\ell}'$ both contain $\boldsymbol{u} + \boldsymbol{v}$ and $\boldsymbol{f}(\boldsymbol{u} + \boldsymbol{v}) = \boldsymbol{f}'(\boldsymbol{u} + \boldsymbol{v})$.
4. **Consistency test:** Assign the first role to Player 1 and the second role to Player 2.
   - Points_1: Receive $\boldsymbol{\nu}$.
   - Points_1: Receive $\boldsymbol{\nu}'$.

   Accept if $\boldsymbol{\nu} = \boldsymbol{\nu}'$.

Figure 10: The game $\mathscr{G}_{\mathrm{IntroIntersect}}(\lambda, d)$.

- Let $\mathcal{G}_1$ be a copy of $\mathscr{G}_{\mathrm{IntroLowDeg}}(\lambda, d)$ using register 1 as the point register and register 2 as the slope register. Write Lines_1 for the surface prover in $\mathcal{G}_1$ and write Points_1 for the points prover.
- Let $\mathcal{G}_2$ be a copy of $\mathscr{G}_{\mathrm{IntroLowDeg}}(\lambda, d)$ using register 2 as the point register and register 1 as the slope register. Write Lines_2 for the surface prover in $\mathcal{G}_2$ and write Points_2 for the points prover.

Then $\mathscr{G}_{\mathrm{IntroIntersect}}(\lambda, d)$ is defined in Figure 10.

Remark 14.5. We remark that although the test runs two separate introspective low-degree tests, we cannot from these alone conclude that either of the points provers answers according to a global function. This is because we use the lines ($k = 1$) introspective low-degree test, whereas from Theorem 13.10 we only know soundness for the planes ($k = 2$) introspective low-degree test. Hence, proving soundness for the introspective intersecting lines test will require an additional assumption, i.e. that one of the two points provers already answers queries according to a global function.

Our main result about the introspective intersecting lines test is the following theorem.

Theorem 14.6. *Let $n, d > 0$ be integers, let q be a power of 2, and let $\lambda = (2, n, q)$ be register parameters. Write $\mathscr{G} := \mathscr{G}_{\mathrm{IntroIntersect}}(\lambda, d)$. Write A for the point prover's measurement in $\mathscr{G}_1$, and write B for the point prover's measurement in $\mathscr{G}_2$.*

- ***Completeness:*** *Suppose there is a degree-d polynomial $g : \mathbb{F}_q^n \to \mathbb{F}_q$ such that*

$$A_{u,\nu} = \tau_u^Z \otimes I_2 \otimes I_{\mathsf{aux}} \cdot \mathbf{1}[\nu = g(u)], \quad B_{v,\nu} = I_1 \otimes \tau_v^Z \otimes I_{\mathsf{aux}} \cdot \mathbf{1}[\nu = g(v)].$$

  Then there is a value-1 λ-register real commuting EPR strategy strategy for $\mathscr{G}$ extending A and B.

- ***Soundness:*** *There exists a function $\delta(\epsilon) = \mathrm{poly}(\epsilon, d/q)$ such that the following holds. Let $\mathcal{S}$ be a projective λ-register strategy which passes $\mathscr{G}$ with probability $1-\epsilon$. Further, suppose that there exists a projective measurement $\{G_g\}_g$ in $\mathrm{PolyMeas}(n,d,q)$ acting on the auxiliary register such that*

$$A_{u,\nu} = \tau_u^Z \otimes I_2 \otimes G_{[g(u)=\nu]}.$$

Then

$$(B_{v,\nu})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} (I_1 \otimes \tau_v^Z \otimes G_{[g(v)=\nu]})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}.$$

Furthermore,

$$\mathsf{Q\text{-}length}(\mathscr{G}) = O(1), \quad \mathsf{A\text{-}length}(\mathscr{G}) = O(n\log(q) + d\log(q)),$$
$$\mathsf{Q\text{-}time}(\mathscr{G}) = O(1), \quad \mathsf{A\text{-}time}(\mathscr{G}) = \mathrm{poly}(n\log(q), d\log(q)).$$

Proof of Theorem 14.6. The runtime and communication complexities follows from the $k=1$ case of the low-degree test. The completeness follows immediately from the completeness of the introspective low-degree test.

Now, we show soundness. Write C for the line prover's measurement in $\mathscr{G}_1$, and write E for the line prover's measurement in $\mathscr{G}_2$. We can write the point provers' measurements as

$$A_{u,\nu} = \tau_u^Z \otimes I_2 \otimes G_{[g(u)=\nu]}, \qquad B_{v,\nu} = I_1 \otimes \tau_v^Z \otimes B_\nu^v.$$

The strategy passes the consistency test with probability $1-\delta(\epsilon)$. As a result,

$$A_{u,\nu} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes A_{u,\nu}.$$

By Fact 12.2, this implies that

$$G_{[g(u)=\nu]} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g(u)=\nu]} \tag{63}$$

on state $|\mathrm{aux}\rangle$ and the uniform distribution on $\mathbb{F}_q^n$. By Fact 4.26, this implies that

$$G_{[g|_\ell=f]} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g|_\ell=f]}, \tag{64}$$

where ℓ is distributed as $\boldsymbol{\ell} = \{\boldsymbol{u} + \lambda\boldsymbol{v} : \lambda \in \mathbb{F}_q\}$ for uniformly random $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{F}_q^n$.

Next, the strategy passes both introspective low-degree tests $\mathscr{G}_1$ and $\mathscr{G}_2$ with probability $1-\delta(\epsilon)$. By Theorem 13.5, this implies measurements $\{C_f^{\ell,v}\}$ and $\{E_{f'}^{\ell',u}\}$ on the auxiliary register such that

$$(C_{\ell,v,f})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} \left(\Pi_\ell^v \otimes \tau_v^Z \otimes C_f^{\ell,v}\right)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}, \tag{65}$$

$$(E_{\ell',u,f'})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} \left(\tau_u^Z \otimes \Pi_{\ell'}^u \otimes E_{f'}^{\ell',u}\right)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}. \tag{66}$$

By Fact 4.32, we can assume Equation (65) holds with equality, incurring a loss of only $\delta(\epsilon)$ in the game value. (We will do the same for Equation (66) later.)

The strategy is now in a form that allows us to apply Lemma 13.8 to the introspective cross-check in $\mathscr{G}_1$. This implies that the measurements $G_{[g(u)=\nu]}$ and $C_f^{\ell,v}$ give a good strategy for the line-versus point test. In other words,

$$C_{[f(u)=\nu]}^{\ell,v} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g(u)=\nu]}.$$

on state $|\mathrm{aux}\rangle$. By Proposition 4.42, this implies that

$$C_f^{\ell,v} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g|_\ell=f]}.$$

Via Equation (64), this implies

$$C_f^{\ell,v} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g|_\ell = f]} \approx_{\delta(\epsilon)} G_{[g|_\ell = f]} \otimes I_{\mathsf{Bob}}.$$

As a result, by Fact 12.3,

$$(C_{\ell,v,f})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} (\Pi_\ell^v \otimes \tau_v^Z \otimes G_{[g|_\ell = f]})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}.$$

By assumption, the right-hand side is projective. As a result, by Fact 4.32, we can assume this expression holds with equality, incurring a loss of only $\delta(\epsilon)$ in the game value. Following this, we apply Fact 4.32 again to assume Equation (66) holds with equality.

The distribution given by on (ℓ, v) and (ℓ', u) when we measure with C and E is exactly the question distribution of the (non-introspective) intersecting lines test. As a result, Fact 12.2 implies that the measurements $G_{[g_\ell = f]}$ and $E_{f'}^{\ell',u}$ pass the intersecting lines test with probability $1 - \delta(\epsilon)$. In other words,

$$E_{[f'(u+v)=\nu]}^{\ell',u} \otimes I_{\mathsf{Alice}} \simeq_{\delta(\epsilon)} I_{\mathsf{Bob}} \otimes G_{[g(u+v)=\nu]}.$$

on state $|\mathrm{aux}\rangle$. Then by Lemma 14.3,

$$E_{f'}^{\ell',u} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g|_{\ell'} = f']}.$$

Via Equation (64), this implies

$$E_{f'}^{\ell',u} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g|_\ell = f']} \approx_{\delta(\epsilon)} G_{[g|_\ell = f']} \otimes I_{\mathsf{Bob}}.$$

Thus, Fact 12.3 implies that

$$(E_{\ell',u,f'})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} \left(\tau_u^Z \otimes \Pi_\ell^u \otimes G_{[g|_{\ell'} = f']}\right)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}.$$

By assumption, the right-hand side is projective. As a result, by Fact 4.32, we can assume this expression holds with equality, incurring a loss of only $\delta(\epsilon)$ in the game value.

The strategy is now in a form that allows us to apply Lemma 13.8 to the introspective cross-check in $\mathscr{G}_2$. This implies that the measurements B_ν^v and $G_{[g|_{\ell'}(v)=\nu]} = G_{[g(v)=\nu]}$ give a good strategy for the line-versus-point low-degree test. In other words,

$$B_\nu^v \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g(v)=\nu]}.$$

on state $|\mathrm{aux}\rangle$. Via Equation (63), this implies

$$B_\nu^v \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g(v)=\nu]} \approx_{\delta(\epsilon)} G_{[g(v)=\nu]} \otimes I_{\mathsf{Bob}}.$$

As a result, by Fact 12.3,

$$(B_{v,\nu})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} (I_1 \otimes \tau_v^Z \otimes G_{[g(v)=\nu]})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}.$$

This completes the proof of the theorem. □

15 The introspective NEEXP protocol

In this question, we give the complete short-question, introspective NEEXP protocol. The goal is a protocol for Succinct-Succinct-3Sat instances of size s_{inst} with $\text{poly}(s_{\text{inst}})$ question length and running time and $\text{poly}(2^{s_{\text{inst}}})$ answer length and running time. Our construction will be an introspective version of the classical PCP construction from Section 11, in which we replace the low-degree tests and simultaneous low-degree tests with our introspective low-degree test and introspective simultaneous low-degree test.

We summarize the protocol here. Given the Succinct-Succinct-3Sat instance $\mathcal{C}_{\text{inst}}$ of size s_{inst}, let $\mathcal{C}$ be the size-s, $(3n+3)$-input Succinct-3Sat instance it succinctly represents, where s and n are roughly exponential in s_{inst}. Following Section 11, we would like the introspective prover to sample strings $\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3 \in \mathbb{F}_q^m$ and $(\boldsymbol{b}, \boldsymbol{w}) \in \mathbb{F}_q^{3+s}$, which they should return to the verifier. In addition, they should return the evaluations $g(\boldsymbol{x}_1), g(\boldsymbol{x}_2), g(\boldsymbol{x}_3)$ and $c_1(\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{w}), \ldots, c_{m'}(\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{w})$, where g and the c_i's are purported degree-d polynomials. This suggests using the following registers:

$$|\text{EPR}_q^m\rangle_1 \otimes |\text{EPR}_q^m\rangle_2 \otimes |\text{EPR}_q^m\rangle_3 \otimes |\text{EPR}_q^{3+s}\rangle_4 .$$

The difficulty in this protocol is ensuring that the polynomials involved are low-degree. To begin, we can run the introspective low-degree test on the first register, which guarantees that $g(\boldsymbol{x}_1)$ corresponds to a low-degree polynomial. Doing the same on registers 2 and 3 would guarantee the functions evaluated on $\boldsymbol{x}_2$ and $\boldsymbol{x}_3$ are also low-degree polynomials, but it would not guarantee that the prover is using *same* low-degree polynomial g on all three. Instead, we run the introspective intersecting lines test twice, ensuring that prover evaluates $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, and $\boldsymbol{x}_3$ using the same function g.

Next, we consider the coefficient polynomials $c_1, \ldots, c_{m'}$. They are evaluated on the concatenated outputs of the four registers, i.e. the string $(\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{w})$. As a result, we view the four registers as a single superregister of length $m' = 3m + 3 + s$, and we would like to perform the introspective simultaneous low-degree test on this superregister. However, this test requires two additional superregisters of length m' to serve as the direction registers. As a result, the shared state between the two provers will be of the following form:

$$|\psi\rangle = (|\text{EPR}_q^m\rangle_1) \otimes |\text{EPR}_q^m\rangle_2 \otimes |\text{EPR}_q^m\rangle_3 \otimes |\text{EPR}_q^{3+s}\rangle_4)_{\mathsf{SuperReg1}} \otimes (|\text{EPR}_q^{m'}\rangle_5)_{\mathsf{SuperReg2}} \otimes (|\text{EPR}_q^{m'}\rangle_6)_{\mathsf{SuperReg3}} \otimes |\text{aux}\rangle_{\mathsf{aux}} .$$

Having checked that the provers' functions are low-degree, we conclude with a consistency check between g and the c_i's to ensure that they encode a satisfying assignment to our Succinct-3Sat. In Section 11, this was done by the "formula test", i.e. the check that $\text{sat}_{\psi,g}(\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{w}) = \text{zero}_{H,c}(\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{w})$. Here, this will be accomplished by an introspective version of this test, in which the provers sample $\boldsymbol{x}$, $\boldsymbol{b}$, and $\boldsymbol{w}$ themselves. Passing this test with high probability proves that $\mathcal{C}_{\text{inst}}$ is a YES instance of the Succinct-Succinct-3Sat problem.

This section is organized as follows. In Section 15.1, we will discuss the register parameters algorithm, needed for the register compiler from Section 5. Next, Section 15.2 introduces the introspective formula game. Finally, Section 15.3 completes the construction and gives the introspective NEEXP game.

15.1 Computing the register parameters

Given the Succinct-Succinct-3Sat instance $\mathcal{C}_{\text{inst}}$ of size s_{inst}, let $\mathcal{C}$ be the size-s, $(3n+3)$-input Succinct-3Sat instance it succinctly represents. To compile our protocol to one sound against general provers, we need a register parameters algorithm which runs in time $\text{poly}(s_{\text{inst}})$ (Definition 5.9).

As described above, the register parameters will be simple functions of the numbers s and n (for example m, a simple function of n to be determined later). However, s and n themselves may not be easy to compute, as the natural way of computing them involves first computing $\mathcal{C}$, a time $2^{s_{\text{inst}}}$ task. We solve this by "guessing" values for these numbers which are guaranteed to be larger than the actual values, and then later "fixing" the circuit $\mathcal{C}$ so that it actually has the guessed input length and size. This is detailed in the following definition.

Definition 15.1. Let $\mathcal{C}_{\text{inst}}$ be a size-s_{inst} instance of the Succinct-Succinct-3Sat problem.

1. Let $\mathcal{C}$ be the size-s Succinct-3Sat instance it succinctly represents. This circuit takes inputs i, j, k, each of some length n, and bits b_1, b_2, b_3. Then s and n can both be trivially upper-bounded by $N := 2^{s_{\text{inst}}}$.

2. Consider a new circuit $\mathcal{C}_{\text{pad}}$ with inputs $i, j, k \in \{0,1\}^N$ and $b \in \{0,1\}^3$. We write $i = (i_1, i_2)$, where i_1 is of length $N - n$ and i_2 is of length n, and likewise for j and k. Let this circuit act as follows:

   - Compute the $\vee$ of the bits in i_1, j_1, and k_1. Output 0 if this is 1.
   - Otherwise, output $\mathcal{C}_{\text{dec}}(i_2, j_2, k_2, b_1, b_2, b_3)$.

   As defined, this circuit has size $s + 3(N-n) + 2 \leq 4N =: S$, and we will pad it with additional gates in a trivial manner so that it has exactly S gates. It can be checked that it succinctly represents the same 3Sat formula as $\mathcal{C}_{\text{dec}}$.

We set $\text{PadC}(\mathcal{C}_{\text{inst}}) := \mathcal{C}_{\text{pad}}$, $\text{PadN}(\mathcal{C}_{\text{inst}}) := N$, and $\text{PadS}(\mathcal{C}_{\text{inst}}) := 4 \cdot N$. We note that given $\mathcal{C}_{\text{inst}}$, the value of N is efficiently computable.

15.2 An introspective formula game

In this section, we introduce the "introspective formula game". This game is the introspective version of the formula check in Section 11, in which we check $\text{sat}_{\psi,g}(\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{w}) = \text{zero}_{H,c}(\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{w})$ on a randomly chosen point $(\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{w})$ in $\mathbb{F}_q^{m'}$. Prior to stating the introspective formula game, we will begin by recalling what this notation means.

Let $\mathcal{C}_{\text{inst}}$ be a size-(s_{inst}) Succinct-Succinct-3Sat instance. Let $\mathcal{C} = \text{PadC}(\mathcal{C}_{\text{inst}})$ be a Succinct-3Sat instance, and let $n = \text{PadN}(\mathcal{C}_{\text{inst}})$ and $s = \text{PadS}(\mathcal{C}_{\text{inst}})$. Then $\mathcal{C}$ is a size-s, $(3n+3)$-variable circuit which is a YES instance of the Succinct-3Sat problem if and only if $\mathcal{C}_{\text{inst}}$ is a YES instance of the Succinct-Succinct-3Sat problem. Introduce $h = 2^{t_1}$, $q = 2^{t_2}$, and m such that $N = 2^n$, h, q, and m are exactly admissible parameters (Definition 11.1). Set $n' = n + 3 + s$ and $m' = m + 3 + s$. We also recall the following pieces of notation.

- (Definition 3.8): Write $H := H_{t_1,t_2}$.
- (Definition 11.4): Given a function $g : \mathbb{F}_q^m \to \mathbb{F}_q$, recall the notation $\text{sat}_{\psi,g} := \text{sat}_{\psi,g,n,t_1,t_2}$.
- (Proposition 11.6): Writing $H_{\text{zero}} = H^{3m} \otimes \{0,1\}^{3+s}$. Given $c_1, \ldots, c_{m'} : \mathbb{F}_q^{m'} \to \mathbb{F}_q$, recall the notation $\text{zero}_{H,c} = \text{zero}_{H_{\text{zero}},c}$.

Before stating the introspective formula game, we must first dispense with the following annoying technicality.

Flip an unbiased coin $\boldsymbol{b} \sim \{0,1\}$.

- Player $\boldsymbol{b}$: Give $(Z,\ Z,\ Z,\ Z,$ "formula"$)$; receive $\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{u}_3, (\boldsymbol{b}, \boldsymbol{w})$ and $\boldsymbol{\nu}_1, \boldsymbol{\nu}_2, \boldsymbol{\nu}_3$ and $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_{M'}$.

Compute $\mathrm{sat}_{\psi,\boldsymbol{\nu}}(\boldsymbol{u}, \boldsymbol{b}, \boldsymbol{w})$ and $\mathrm{zero}_{H,\boldsymbol{\mu}}(\boldsymbol{u}, \boldsymbol{b}, \boldsymbol{w})$. Accept if they are equal.

Figure 11: The game $\mathscr{G}_{\mathrm{IntroForm}}(\mathcal{C}_{\mathrm{inst}}, h, q, m)$.

Notation 15.2. In the classical case (Section 11), we have a fixed proof which contains fixed functions which may or may not be low-degree. In the quantum case, however, we are dealing not with a fixed proof but an interactive prover, and the formula prover may not respond based on fixed functions (their responses might be randomized, for example). To account for this, we modify the definitions of sat and zero as follows. First, we recall the notation $g_\psi := g_{\psi,n,t_1,t_2}$ (Definition 11.3).

- Given $\nu_1, \nu_2, \nu_3 \in \mathbb{F}_q$, define

$$\mathrm{sat}_{\psi,\nu}(x, b, w) := g_\psi(x, b, w) \cdot (\nu_1 - b_1)(\nu_2 - b_2)(\nu_3 - b_3).$$

- Given $\mu_1, \ldots, \mu_{m'} \in \mathbb{F}_q$, define

$$\mathrm{zero}_{H,\mu}(x) = \sum_{i=1}^{m'} \mathrm{zero}_{(H_{\mathrm{zero}})_i}(x_i) \cdot \mu_i,$$

  where by definition $(H_{\mathrm{zero}})_i = H$ for $i \in [3m]$ and $(H_{\mathrm{zero}})_i = \{0,1\}$ otherwise.

We note that if there is a function g such that $\nu_i = g(x_i)$, then $\mathrm{sat}_{\psi,\nu} = \mathrm{sat}_{\psi,g}$. Similarly, if there are functions $c_1, \ldots, c_{m'}$ such that $\mu_i = c_i(x)$, then $\mathrm{zero}_{H,\mu} = \mathrm{zero}_{H,c}$.

Now we state the introspective formula game.

Definition 15.3. Let $\mathcal{C}_{\mathrm{inst}}$ be a size-(s_{inst}) Succinct-Succinct-3Sat instance. Let $n = \mathrm{PadN}(\mathcal{C}_{\mathrm{inst}})$ and $s = \mathrm{PadS}(\mathcal{C}_{\mathrm{inst}})$. Suppose n, $h = 2^{t_1}$, $q = 2^{t_2}$, and m are exactly admissible parameters. The *introspective formula game*, denoted $\mathscr{G} := \mathscr{G}_{\mathrm{IntroForm}}(\mathcal{C}_{\mathrm{inst}}, h, q, m)$, is defined in Figure 11. This is a $\lambda_{\mathcal{C}_{\mathrm{inst}},q} := (4, \ell, q)$-register game, for $\ell = (m, m, m, 3+s)$. Furthermore,

$$\mathsf{Q\text{-}length}(\mathscr{G}) = O(1), \quad \mathsf{A\text{-}length}(\mathscr{G}) = O(m' \log(q)),$$

$$\mathsf{Q\text{-}time}(\mathscr{G}) = O(1), \quad \mathsf{A\text{-}time}(\mathscr{G}) = \mathrm{poly}(s, n, n', h, q, m').$$

Notation 15.4. In the case when a prover is given the question $(Z, Z, Z, Z,$ "formula"$)$, we refer to it as the *formula prover*. It has the following intended behavior.

3. **Formula prover:**

   Input: Pauli basis queries (Z, Z, Z, Z) and auxiliary query "formula".

   Output: Strings $u_1, u_2, u_3 \in \mathbb{F}_q^m$ and $(b, w) \in \mathbb{F}_q^{3+s}$. Three numbers $\nu_1, \nu_2, \nu_3 \in \mathbb{F}_q$ and m' numbers $\mu_1, \ldots, \mu_{m'} \in \mathbb{F}_q$.

   Goal: The prover should act as follows.

   - The prover sets $\nu_1 = g(u_1)$, $\nu_2 = g(u_2)$, $\nu_3 = g(u_3)$, where $g : \mathbb{F}_q^m \to \mathbb{F}_q$ is a global degree-d_1 polynomial selected independently of u.

- They then set $\mu_i = c_i(u_1, u_2, u_3, b, w)$, where for each i, $c_i : \mathbb{F}_q^{m'} \to \mathbb{F}_q$ is a global degree-d_2 polynomial selected independently of (u, b, w).

Here, d_1 and d_2 are polynomial degrees which will be selected later. We will also refer to the *formula prover's measurement*, which refers to the measurement $\{F_{u,b,w,\nu_i,\mu_j}\}$ such that

$$F_{u,b,w,\nu_1,\nu_2,\nu_3,\mu_1,\ldots,\mu_{m'}} = M^{Z,Z,Z,Z,\text{“formula”}}_{u,b,w,\nu_1,\nu_2,\nu_3,\mu_1,\ldots,\mu_{m'}}.$$

We begin by showing the completeness case of the introspective formula game.

Proposition 15.5 (Introspective formula game completeness)**.** *Suppose $\mathcal{C}_{\text{inst}}$ is a YES instance of the* Succinct-Succinct-3Sat *problem. Let $a : \{0,1\}^n \to \{0,1\}$ be a satisfying assignment to the* 3Sat *instance it encodes, and let $g := g_a : \mathbb{F}_q^m \to \mathbb{F}_q$ be its low-degree encoding. Let $c_1, \ldots, c_{m'} : \mathbb{F}_q^{m'} \to \mathbb{F}_q$ be the coefficient polynomials guaranteed to make $\text{sat}_{\psi,g} = \text{zero}_{H_{\text{zero}},c}$ by Proposition 11.6. Both g and the c_i's are degree-$O(hn')$ polynomials. Consider the $\lambda_{\mathcal{C}_{\text{inst}},q}$-register strategy (ψ, A) with no auxiliary register in which*

$$A_{u,b,w,\nu,\mu} = \tau^Z_{u_1} \otimes \tau^Z_{u_2} \otimes \tau^Z_{u_3} \otimes \tau^Z_{b,w} \cdot \mathbf{1}[\nu_i = g(u_i), \mu_j = c_j(u, b, w)],$$

where the indices range over $i \in [3]$ and $j \in [m']$. Then this strategy passes $\mathscr{G}_{\text{IntroForm}}(\mathcal{C}_{\text{inst}}, h, q, m)$ with probability 1.

Proof. This game is simply the oracularized version of the formula check in the classical PCP. The proposition follows from the discussion in Section 11.5. □

Our next lemma covers the soundness case of the introspective formula game. It concerns provers of a particular form, namely those whose measurements correspond to low-degree polynomials. We show that if there exists such a prover with nonnegligible value, then the formula must be satisfiable.

Lemma 15.6 (Formula game partial soundness)**.** *Let $\mathcal{C}_{\text{inst}}$ be a* Succinct-Succinct-3Sat *instance, and set $\mathscr{G} := \mathscr{G}_{\text{IntroForm}}(\mathcal{C}_{\text{inst}}, h, q, m)$. Let $\mathcal{S} = (\psi, A)$ be a $\lambda_{\mathcal{C}_{\text{inst}},q}$-register strategy. Consider a measurement on the auxiliary register*

$$G = \{G_{g,c_1,\ldots,c_{m'}}\}$$

with outcomes degree-d_1 polynomials $g : \mathbb{F}_q^m \to \mathbb{F}_q$ and degree-d_2 polynomials $c_1, \ldots, c_{m'} : \mathbb{F}_Q^{m'} \to \mathbb{F}_q$. Suppose A has the following form: for each u, b, w, ν, and μ,

$$A_{u,b,w,\nu,\mu} = \tau^Z_{u_1} \otimes \tau^Z_{u_2} \otimes \tau^Z_{u_3} \otimes \tau^Z_{b,w} \otimes \left(G_{[g(u_i)=\nu_i, c_j(u,b,w)=\mu_j]}\right)_{\mathsf{aux}}, \tag{67}$$

where the subscript of the G measurement ranges over all $i \in [3]$ and $j \in [m']$. If the probability $\mathcal{S}$ passes $\mathcal{G}$ is at least

$$\frac{\max\{O(hn') + 3d_1, h + d_2\}}{q},$$

then ψ is satisfiable.

Proof. Consider the following three step strategy:

1. Measure the auxiliary register with $\{G_{g,c_1,\ldots,c_{m'}}\}$, receiving functions $\boldsymbol{g}$, $\boldsymbol{c}_1$, …, $\boldsymbol{c}_{m'}$.
2. Measure the EPR registers in the Z basis, receiving $\boldsymbol{u}$, $\boldsymbol{b}$, and $\boldsymbol{w}$.

3. Output $\boldsymbol{u}$, $\boldsymbol{b}$, $\boldsymbol{w}$, $\boldsymbol{g}(\boldsymbol{u}_1)$, $\boldsymbol{g}(\boldsymbol{u}_2)$, $\boldsymbol{g}(\boldsymbol{u}_3)$ and $\boldsymbol{c}_1(\boldsymbol{u}, \boldsymbol{b}, \boldsymbol{w})$ through $\boldsymbol{c}_{M'}(\boldsymbol{u}, \boldsymbol{b}, \boldsymbol{w})$.

This passes the formula game with probability $\mathrm{val}_{\mathcal{G}}(\mathcal{S})$. Then there exists functions g, c_1, ..., $c_{m'}$ such that conditioned on measuring them in step one, this strategy passes with probability at least $\mathrm{val}_{\mathcal{G}}(\mathcal{S})$. By the remark at the end of Notation 15.2, this is the probability that

$$\mathrm{sat}_{\psi,g}(\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{w}) = \mathrm{zero}_{H,c}(\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{w}),$$

where $(\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{w})$ is drawn from $\mathbb{F}_q^{m'}$ uniformly at random. The lemma follows from Lemma 11.7. □

15.3 The complete introspective protocol

In this section, we introduce the introspective protocol for NEEXP and prove its correctness. The introspective NEEXP protocol builds on top of the introspective formula game by using a series of introspective low-degree tests to ensure that the formula prover satisfies the condition in Equation (67). Having done this, we can then apply Lemma 15.6, ensuring that if a strategy passes with high probability, then the instance is satisfiable.

Definition 15.7. Let $\mathcal{C}_{\mathrm{inst}}$ be a size-(s_{inst}) Succinct-Succinct-3Sat instance. Let $n = \mathrm{PadN}(\mathcal{C}_{\mathrm{inst}})$ and $s = \mathrm{PadS}(\mathcal{C}_{\mathrm{inst}})$. The verifier chooses $h = 2^{t_1}$, $q = 2^{t_2}$, m, and d such that m, h, q, and m are exactly admissible parameters satisfying

$$h = \Theta(n), \quad m = \Theta\left(\frac{n}{\log(n)}\right), \quad q = \mathrm{poly}(n), \quad d = O(hn') = O(n^2).$$

(We will choose the polynomial for q in Theorem 15.8 below.) The verifier sets $\lambda = (6, \ell, q)$, where $\ell = (m, m, m, 3 + s, m', m')$.

We begin by instantiating the following list of subroutines.

- Let $\lambda_{\mathrm{LD}} = (3, m, q)$ be register parameters. Let $\mathscr{G}_{\mathrm{LD}}$ be a copy of $\mathscr{G}_{\mathrm{IntroLowDeg}}(\lambda_{\mathrm{LD}}, d)$, using register 1 as the point register and registers 2 and 3 as the direction registers. Write Points_1 for the points prover.
- Let $\lambda_{\mathrm{IL}} = (2, m, q)$ be register parameters. Let $\mathscr{G}_{\mathrm{IL1}}$ be a copy of $\mathscr{G}_{\mathrm{IntroIntersect}}(\lambda_{\mathrm{IL}}, d)$ on registers 1 and 2 whose points prover for register 1 is Points_1 from $\mathscr{G}_{\mathrm{LD}}$. Write Points_2 for the points prover on register 2.
- Let $\mathscr{G}_{\mathrm{IL2}}$ be a copy of $\mathscr{G}_{\mathrm{IntroIntersect}}(\lambda_{\mathrm{IL}}, d)$ on registers 1 and 3 whose points prover for register 1 is Points_1 from $\mathscr{G}_{\mathrm{LD}}$. Write Points_3 for the points prover on register 3.
- Let $\mathscr{G}_{\mathrm{F}}$ be a copy of $\mathscr{G}_{\mathrm{IntroForm}}(\mathcal{C}_{\mathrm{inst}}, h, q, m)$ on registers 1, 2, 3, and 4. Write $\mathsf{Formula}$ for the formula prover.
- Let $\lambda_{\mathrm{LDSUP}} = (3, m', q)$ be register parameters. Let $\mathscr{G}_{\mathrm{LDSUP}}$ be a copy of $\mathscr{G}_{\mathrm{IntroLowDeg}}(\lambda_{\mathrm{LDSUP}}, d, 3+m')$, applied to the following three superregisters: registers 1 through 4 are combined into the point superregister, register 5 is used as the first direction superregister, and register 6 is used as the second direction superregister. In addition, use $\mathsf{Formula}$ from $\mathscr{G}_{\mathrm{form}}$ as its points prover.

Then the *introspective* NEEXP *game*, denoted $\mathscr{G}_{\mathrm{IntroNEEXP}}(\mathcal{C}_{\mathrm{inst}})$, is defined in Figure 12.

The main result Part IV is the following theorem.

With probability $\frac{1}{9}$ each, perform one of the following nine tests.

1. **Low degree test:** Play $\mathscr{G}_{\mathrm{LD}}$.
2. **Intersecting lines test 1:** Play $\mathscr{G}_{\mathrm{IL1}}$.
3. **Intersecting lines test 2:** Play $\mathscr{G}_{\mathrm{IL2}}$.
4. **Simultaneous low degree test:** Play $\mathscr{G}_{\mathrm{LDSUP}}$.
5. **Formula test:** Player $\mathscr{G}_{\mathrm{F}}$.

For the remaining tests, flip an unbiased coin $\boldsymbol{b} \sim \{0,1\}$. Assign the first role to Player $\boldsymbol{b}$ and the second role to Player $\overline{\boldsymbol{b}}$.

6. **Consistency test 1:**
   - Points_1: Receive $\boldsymbol{\nu}$.
   - $\mathsf{Formula}$: Receive $\boldsymbol{\nu}_1$.

   Accept if $\boldsymbol{\nu} = \boldsymbol{\nu}_1$.
7. **Consistency test 2:**
   - Points_2: Receive $\boldsymbol{\nu}$.
   - $\mathsf{Formula}$: Receive $\boldsymbol{\nu}_2$.

   Accept if $\boldsymbol{\nu} = \boldsymbol{\nu}_2$.
8. **Consistency test 3:**
   - Points_3: Receive $\boldsymbol{\nu}$.
   - $\mathsf{Formula}$: Receive $\boldsymbol{\nu}_3$.

   Accept if $\boldsymbol{\nu} = \boldsymbol{\nu}_3$.
9. **Consistency test 4:**
   - $\mathsf{Formula}$: Receive $\boldsymbol{\nu}_1, \boldsymbol{\nu}_2, \boldsymbol{\nu}_3$ and $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_{m'}$.
   - $\mathsf{Formula}$: Receive $\boldsymbol{\nu}'_1, \boldsymbol{\nu}'_2, \boldsymbol{\nu}'_3$ and $\boldsymbol{\mu}'_1, \ldots, \boldsymbol{\mu}'_{m'}$.

   Accept if $\boldsymbol{\nu}_i = \boldsymbol{\nu}'_i$ and $\boldsymbol{\mu}_j = \boldsymbol{\mu}'_j$ for all $i \in [3]$, $j \in [m']$.

Figure 12: The game $\mathscr{G}_{\mathrm{IntroNEEXP}}(\mathcal{C}_{\mathrm{inst}})$.

Theorem 15.8. *Let $\mathcal{C}_{\text{inst}}$ be a size-(s_{inst})* Succinct-Succinct-3Sat *instance. Let q be a sufficiently large $\text{poly}(n)$ and $\epsilon > 0$ a sufficiently small constant such that Equation (72) is at least $\frac{1}{2}$ and Equation (73) is less than $\frac{1}{2}$. Write $\mathscr{G} := \mathscr{G}_{\text{IntroNEEXP}}(\mathcal{C}_{\text{inst}})$.*

- ***Completeness:*** *Suppose $\mathcal{C}_{\text{inst}}$ encodes a satisfiable formula. Then there is a value-1 λ-register strategy for $\mathscr{G}$ with no auxiliary register.*
- ***Soundness:*** *If there is a λ-register strategy for $\mathscr{G}$ with value at least $1-\epsilon$, then $\mathcal{C}_{\text{inst}}$ encodes a satisfiable formula.*

Furthermore,

$$\mathsf{Q\text{-}length}(\mathscr{G}) = O(1), \quad \mathsf{A\text{-}length}(\mathscr{G}) = \text{poly}(2^{s_{\text{inst}}}),$$
$$\mathsf{Q\text{-}time}(\mathscr{G}) = O(1), \quad \mathsf{A\text{-}time}(\mathscr{G}) = \text{poly}(2^{s_{\text{inst}}}).$$

Proof. The question lengths and question runtimes are both $O(1)$ because all involved subtests have $O(1)$ question complexity. The answer lengths and question runtimes are both $\text{poly}(2^{s_{\text{inst}}})$ because all our parameters are at most polynomial in $n = 2^{s_{\text{inst}}}$, and the question lengths and question runtimes of all involved subtests are polynomial in these parameters.

We name the measurements used by the provers as follows.

$$\mathsf{Points}_1 : A, \quad \mathsf{Points}_2 : B, \quad \mathsf{Points}_3 : C, \quad \mathsf{Formula} : F.$$

We will write the identity matrix on registers 5 and 6 as $I_{5,6} := I_5 \otimes I_6$.

Completeness. Suppose $\mathcal{C}_{\text{inst}}$ encodes a satisfiable formula. By Proposition 15.5, there are degree-d polynomials $g : \mathbb{F}_q^m \to \mathbb{F}_q$ and $c_1, \dots, c_{m'} : \mathbb{F}_q^{m'} \to \mathbb{F}_q$ such that if we define

$$F_{u,b,w,\nu,\mu} = \tau^Z_{u_1} \otimes \tau^Z_{u_2} \otimes \tau^Z_{u_3} \otimes \tau^Z_{b,w} \otimes I_{5,6} \cdot \mathbf{1}[\nu_i = g(u_i), \mu_j = c_j(u,b,w)],$$

then this strategy passes the formula test with probability 1. We extend this strategy to the remaining measurements as follows.

$$A_{u_1,\nu_1} = \tau^Z_{u_1} \otimes I_2 \otimes I_3 \otimes I_4 \otimes I_{5,6} \cdot \mathbf{1}[\nu_1 = g(u_1)],$$
$$B_{u_2,\nu_2} = I_1 \otimes \tau^Z_{u_2} \otimes I_3 \otimes I_4 \otimes I_{5,6} \cdot \mathbf{1}[\nu_2 = g(u_2)],$$
$$C_{u_3,\nu_3} = I_1 \otimes I_2 \otimes \tau^Z_{u_3} \otimes I_4 \otimes I_{5,6} \cdot \mathbf{1}[\nu_3 = g(u_3)].$$

By the completeness of the introspective low-degree and intersecting lines tests, these can be extended to a strategy which passes the whole test with probability 1.

Soundness. Throughout this proof, we use $\delta(\epsilon)$ to represent functions of the form

$$\delta(\epsilon) = \text{poly}(\epsilon, m \cdot d/q^e),$$

where $e > 0$ is an absolute constant.

Low-degree tests. The strategy passes the introspective low-degree test with probability $1-\delta(\epsilon)$. Applying Theorem 13.10, there is a measurement $G = \{G_g\}$ in $\mathrm{PolyMeas}(m, d, q)$ such that

$$(A_{u_1,\nu_1})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} \left(\tau^Z_{u_1} \otimes I_2 \otimes I_3 \otimes I_4 \otimes I_{5,6} \otimes (G_{[g(u_1)=\nu_1]})_{\mathsf{aux}}\right)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}.$$

By Fact 4.32, we can assume this holds with equality with a loss of only $\delta(\epsilon)$ in the game value. In addition, by Theorem 4.1, we can assume that the G measurements are all projective.

Next, the strategy passes the two introspective intersecting lines tests with probability $1-\delta(\epsilon)$ each. By Theorem 14.6, this implies that

$$(B_{u_2,\nu_2})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} \left(I_1 \otimes \tau^Z_{u_2} \otimes I_3 \otimes I_4 \otimes I_{5,6} \otimes (G_{[g(u_2)=\nu_2]})_{\mathsf{aux}}\right)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}, \tag{68}$$

$$(C_{u_3,\nu_3})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} \left(I_1 \otimes I_2 \otimes \tau^Z_{u_3} \otimes I_4 \otimes I_{5,6} \otimes (G_{[g(u_3)=\nu_3]})_{\mathsf{aux}}\right)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}. \tag{69}$$

Similarly, the strategy passes the introspective simultaneous low-degree test with probability $1-\delta(\epsilon)$. Applying Theorem 13.12, there is a measurement $J = \{J_{f_1,f_2,f_3,c_1,\ldots,c_{m'}}\}$ in $\mathrm{PolyMeas}(m', d, q, 3+m')$ such that

$$\begin{aligned}(F_{u,b,w,\nu,\mu})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \\ \approx_{\delta(\epsilon)} \left(\tau^Z_{u_1} \otimes \tau^Z_{u_2} \otimes \tau^Z_{u_3} \otimes \tau^Z_{b,w} \otimes I_{5,6} \otimes (J_{[f_i(u,b,w)=\nu_i, c_j(u,b,w)=\mu_j]})_{\mathsf{aux}}\right)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}},\end{aligned} \tag{70}$$

where the subscript of the J measurement ranges over all $i \in [3]$ and $j \in [m']$. By Fact 4.32, we can assume Equations (68) to (70) holds with equality with a loss of only $\delta(\epsilon)$ in the game value. In addition, by Theorem 4.1, we can assume that the J measurements are all projective.

Consistency tests. The strategy passes the four consistency tests with probability $1-\delta(\epsilon)$ each, implying

$$\begin{aligned}(F_{u_1,\nu_1})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} &\simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes (A_{u_1,\nu_1})_{\mathsf{Bob}}, \\ (F_{u_2,\nu_2})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} &\simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes (B_{u_2,\nu_2})_{\mathsf{Bob}}, \\ (F_{u_3,\nu_3})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} &\simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes (C_{u_3,\nu_3})_{\mathsf{Bob}}, \\ (F_{u,b,w,\nu,\mu})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} &\simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes (F_{u,b,w,\nu,\mu})_{\mathsf{Bob}}.\end{aligned}$$

By introspection (Fact 12.4), these imply the following statements:

$$\begin{aligned}(J_{[f_1(u,b,w)=\nu_1]})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} &\simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes (G_{[g(u_1)=\nu_1]}), \\ (J_{[f_2(u,b,w)=\nu_2]})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} &\simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes (G_{[g(u_2)=\nu_2]}), \\ (J_{[f_3(u,b,w)=\nu_3]})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} &\simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes (G_{[g(u_3)=\nu_3]}), \\ (J_{[c_j(u,b,w)=\mu_j]})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} &\simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes (J_{[c_j(u,b,w)=\mu_j]}),\end{aligned}$$

where the subscript of the J measurement ranges over all $i \in [3]$ and $j \in [m']$. Here, these statements are with respect to the strategy's auxiliary state and to the uniform distribution on $(\boldsymbol{u}, \boldsymbol{b}, \boldsymbol{w}) \in \mathbb{F}_q^{m'}$.

Now we apply Fact 4.34. To do so, let us specify the sets $\mathcal{G}_i$ and the distance parameter. The three sets $\mathcal{G}_2$, $\mathcal{G}_3$, and $\mathcal{G}_4$ will just contain all degree-d polynomials $g : \mathbb{F}_q^{m'} \to \mathbb{F}_q$. (Note that we can view the outputs of G_g as degree-d polynomials which disregard all of their input (u, b, w) aside from one of the three strings u_1, u_2, or u_3.) By Schwarz-Zippel, these have distance at least $1 - d/q$. The remaining set, $\mathcal{G}_1$, is defined as follows: for each tuple of degree-d polynomials

$c_1, \ldots, c_{m'}$, it contains a function c defined as $c(u,b,w) = (c_1(u,b,w), \ldots, c_{m'}(u,b,w))$. Any two nonequal $c, c' \in \mathcal{G}_1$ have some coordinate i in which $c_i \neq c'_i$, and on this coordinate alone they will have distance at least $1 - d/q$ by Schwarz-Zippel. Thus, c and c' have distance at least $1 - d/q$.

Define the measurement $\{K_{g,c_1,\ldots,c_{m'}}\}$ as

$$K_{g,c_1,\ldots,c_{m'}} := G_g \cdot J_{c_1,\ldots,c_{m'}} \cdot G_g,$$

Then Fact 4.34 implies that

$$(J_{[f_i(u,b,w)=\nu_i, c_j(u,b,w)=\mu_j]})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes (K_{[g(u_i)=\nu_i, c'_j(u,b,w)=\mu_j]}),$$

where the subscripts range over all $i \in [3]$ and $j \in [m']$. By introspection (Fact 12.4), this implies that

$$(F_{u,b,w,\nu,\mu})_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} \left(\tau^Z_{u_1} \otimes \tau^Z_{u_2} \otimes \tau^Z_{u_3} \otimes \tau^Z_{b,w} \otimes I_{5,6} \otimes (K_{[g(u_i)=\nu_i, c_j(u,b,w)=\mu_j]})_{\mathsf{aux}}\right)_{\mathsf{Alice}} \otimes I_{\mathsf{Bob}}, \quad (71)$$

where the subscripts range over all $i \in [3]$ and $j \in [m']$. By Fact 4.32, we can assume Equation (71) holds with equality with a loss of only $\delta(\epsilon)$ in the game value.

Formula test: At this point, the formula prover's strategy F satisfies the condition in Equation (67) with $d_1 = d_2 = d$. In addition, it passes the introspective formula test with probability

$$1 - \mathrm{poly}(\epsilon, m \cdot d/q), \quad (72)$$

which by our setting of parameters is at least $\frac{1}{2}$. Finally, our setting of parameters also implies that

$$\frac{\max\{O(hn') + 3d, h + d\}}{q} \quad (73)$$

is less than $\frac{1}{2}$. As a result, we can apply Lemma 15.6 to conclude that ψ is satisfiable. □

Theorem 15.8 only proves soundness of the introspective NEEXP protocol against λ-register strategies. Our last step is to compile this protocol into one which is sound against *general* strategies, while only slightly increasing the question length.

Corollary 15.9. *There is an absolute constant $\epsilon > 0$ such that the following is true. Let $\mathcal{C}_{\mathrm{inst}}$ be a size-(s_{inst}) Succinct-Succinct-3Sat instance. Then there exists a game $\mathscr{G} := \mathscr{G}_{\mathrm{IntroNEEXP}}(\mathcal{C}_{\mathrm{inst}})$ with the following properties.*

- ***Completeness:*** *Suppose $\mathcal{C}_{\mathrm{inst}}$ encodes a satisfiable formula. Then there is a value-1 real commuting EPR strategy for $\mathscr{G}$.*
- ***Soundness:*** *If there is a strategy for $\mathscr{G}$ with value at least $1-\epsilon$, then $\mathcal{C}_{\mathrm{inst}}$ encodes a satisfiable formula.*

Furthermore,

$$\mathsf{Q\text{-}length}(\mathscr{G}) = O(s_{\mathrm{inst}}), \quad \mathsf{A\text{-}length}(\mathscr{G}) = \mathrm{poly}(2^{s_{\mathrm{Inst}}}),$$
$$\mathsf{Q\text{-}time}(\mathscr{G}) = O(s_{\mathrm{inst}}), \quad \mathsf{A\text{-}time}(\mathscr{G}) = \mathrm{poly}(2^{s_{\mathrm{inst}}}).$$

Proof. Let $n = \mathrm{PadN}(\mathcal{C}_{\mathrm{inst}})$ and $s = \mathrm{PadS}(\mathcal{C}_{\mathrm{inst}})$. Set $m = \Theta(n/\log(n))$ and $q = \mathrm{poly}(n)$, as in Definition 15.7. Set $\lambda = (6, \ell, q)$, where $\ell = (m, m, m, 3+s, 3m+3+s, 3m+3+s)$. Then $\mathscr{G}_{\mathrm{IntroNEEXP}}(\mathcal{C}_{\mathrm{inst}})$ is a λ-register game. Furthermore, by Definition 15.1, the register parameters are computable in time $\mathrm{poly}(s_{\mathrm{inst}})$.

Let $\epsilon > 0$ be as in Theorem 15.8, and select a constant $\epsilon' > 0$ and $\eta_1, \ldots, \eta_6 = 1/\mathrm{poly}(n)$ such that $\delta(\epsilon') \leq \epsilon$, where $\delta(\epsilon') = \mathrm{poly}(\epsilon', \eta_1, \ldots, \eta_6)$ is as in Corollary 5.10. Now, if we apply Corollary 5.10, it gives us a game $\mathscr{G}$ with the following properties.

- If $\mathcal{C}$ is a "Yes" instance, then $\mathscr{G}_{\mathrm{IntroNEEXP}}(\mathcal{C}_{\mathrm{inst}})$ has a value-1 strategy with no auxiliary state, which implies that $\mathscr{G}$ has a value-1 commuting EPR strategy.
- If $\mathcal{C}$ is a "No" instance, then every λ-register strategy for $\mathscr{G}_{\mathrm{IntroNEEXP}}(\mathcal{C}_{\mathrm{inst}})$ has value less than $1-\epsilon$. By our choice of parameters, this is less than $1-\delta(\epsilon')$, which implies that $\mathscr{G}$ has no strategy with value $1-\epsilon'$.

Furthermore, $\log(m) = O(\log(n)) = O(s_{\mathrm{inst}})$ and $\log(s) = O(\log(n)) = O(s_{\mathrm{inst}})$, giving us our desired question complexities, and $\mathrm{poly}(m) = \mathrm{poly}(n) = \mathrm{poly}(2^{s_{\mathrm{inst}}})$ and $\mathrm{poly}(s) = \mathrm{poly}(n) = \mathrm{poly}(2^{s_{\mathrm{inst}}})$, giving us our desired answer complexities. □

Part V
Answer reduction

16 Testing error-correcting codes

In Section 17 below, rather than the prover sending the verifier their entire "large" answer a, they will instead encode it into $\mathrm{Enc}(a)$ using an error correcting code and allow the verifier to query individual bits of the encoding. (The fact that the verifier is allowed to query bits of the encoding rather than the original string stems from the PCPP technology we use. See Section 17.3 for more details.) In this section, we develop the tests which verify that provers are performing this task honestly, so that when we query a subset of the bits I, they respond based on the bits of a codeword which was sampled independently of I. We will develop such a test for the low-degree code (Section 16.1).

Our proofs are entirely standard: we start with the known property tester for this code (i.e.. Theorem 4.40), which allows us to query the prover's codeword at a uniformly random location. Then we use the local decodability properties of this code to allow us to query arbitrary subsets of coordinates. We begin by stating a slightly nonstandard definition of error-correcting codes relevant to our application.

Definition 16.1 (Error-correcting codes)**.** Let m and q be integers, and let $\eta \in [0,1]$. An (n, m, q, η)-*error-correcting code* $\mathrm{Code} = (\mathrm{Enc}, \mathrm{Dec}, \mathrm{Sub})$ is defined as follows.

- Sub is a subset of $\mathbb{F}_q^m$ such that for each $x \neq y \in \mathrm{Sub}$, x and y have normalized Hamming agreement at most η (i.e. the probability, over a uniformly random $\boldsymbol{i} \in [m]$, that $x_{\boldsymbol{i}} = y_{\boldsymbol{i}}$ is at most η).
- $\mathrm{Enc} : \{0,1\}^n \to \mathrm{Sub} \subseteq \mathbb{F}_q^m$ is the *encoding map*.
- $\mathrm{Dec} : \mathbb{F}_q^m \to \{0,1\}^n \cup \{\perp\}$ is the *decoding map*. For each $x \in \{0,1\}^n$, $\mathrm{Dec}(\mathrm{Enc}(x)) = x$. In addition, for every w not in the range of Enc, $\mathrm{Dec}(w) = \perp$.

Remark 16.2. The purpose of the subset Sub is this: in this section, we are designing games which test that a prover responds according to an error-correcting code. This means that the prover should respond based on the encoding $\mathrm{Enc}(x)$ of some string $x \in \{0,1\}^n$. However, the games we design may only be able to test if the prover responds based on a string in Sub, which contains the encodings $\mathrm{Enc}(x)$ but may include other strings as well. This definition ensures that these other strings are still far from each other in Hamming distance.

The next definition defines a subset tester.

Definition 16.3. Let $\mathrm{Code} = (\mathrm{Enc}, \mathrm{Dec}, \mathrm{Sub})$ be an (n, m, q, η)-error-correcting code. Let k be an integer. Given a game $\mathscr{G}(\cdot)$ whose inputs are from the set of subsets of $[m]$ of size k and a probability distribution $\mathcal{D}$ over this set, we write $\mathscr{G}(\mathcal{D})$ for the game in which we first sample $\boldsymbol{I} \sim \mathcal{D}$ and then play $\mathscr{G}(\boldsymbol{I})$. Then $\mathscr{G}$ is a *k-subset tester with robustness $\delta(\epsilon)$* for Code if it satisfies the following two properties.

- **Completeness:** Let (ψ, M) be an EPR strategy in which $\{M_w\}$ is a measurement with outcomes in $\{0,1\}^n$. Consider the partial strategy (ψ, G) in which

$$G^I_{a_1,\ldots,a_k} := M_{[\mathrm{Enc}(w)|_I = a_1,\ldots,a_k]}.$$

  Then this can be extended to a (full) real commuting EPR strategy which, for each I, passes $\mathscr{G}(I)$ with probability 1.

- **Soundness:** For any distribution $\mathcal{D}$, consider a strategy (ψ, M) which passes $\mathscr{G}(\mathcal{D})$ with probability $1-\epsilon$. Then there exists a measurement $\{G_w\}_w$ with outcomes w in Sub such that

$$M^I_{a_1,\ldots,a_k} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[w|_I = a_1,\ldots,a_k]}.$$

16.1 Testing the low-degree code

In this section, we show how to test the low-degree code. This is essentially an exercise in generalizing Theorem 4.40 to arbitrary subsets. We begin with some notation.

Notation 16.4. We write $\mathcal{F}^m_{q,k}$ for the family $\mathcal{F}^m_{q,k} = \{F \subseteq \mathbb{F}_q^m \mid |F| \leq k\}$.

Now we define the low-degree code tester.

Definition 16.5. Let m, q, and d be integers. Let k be an integer, and let F be an element of $\mathcal{F}^m_{q,k}$. Then $\mathscr{G}_{\mathrm{LDsubset}}(m, q, d, F)$ is the game defined in Figure 13.

The performance of the low-degree subset game is given by the following theorem.

Theorem 16.6. *Consider low-degree parameters* $\mathsf{params} = (n, q, h, H, m, \mathcal{S}, \pi)$. *Set* $d = m(h-1)$. *Set* $m' = q^m$. *We will identify strings in* $\mathbb{F}_q^{m'}$ *with functions* $g : \mathbb{F}_q^m \to \mathbb{F}_q$. *Given* $a \in \{0,1\}^n$, *define* $\mathrm{Enc}(a) = g_a$ *and* $\mathrm{Dec}(g_a) = a$. *For all other* $g : \mathbb{F}_q^m \to \mathbb{F}_q$ *(i.e. those which are not the low-degree encoding of a string a), define* $\mathrm{Dec}(g) = \perp$. *Finally, define* Sub *to be the set of degree d polynomials* $g : \mathbb{F}_q^m \to \mathbb{F}_q$. *Then* $\mathrm{Code} = (\mathrm{Enc}, \mathrm{Dec}, \mathrm{Sub})$ *is an* $(n, m', q, d/q)$*-error-correcting code.*

Furthermore, there exists a constant $c > 0$ *and a function* $\delta(\epsilon) = \mathrm{poly}(\epsilon, dm/q^c)$ *such that the following holds. Let k be an integer. Then* $\mathscr{G}_{\mathrm{LDsubset}} := \mathscr{G}_{\mathrm{LDsubset}}(m, q, d, \cdot)$ *is a k-subset tester for* LDCode *with robustness* $\delta(\epsilon)$.

Finally,

$$\mathsf{Q\text{-}time}(\mathscr{G}_{\mathrm{LDsubset}}) = \mathrm{poly}(m, k, \log q), \quad \mathsf{A\text{-}time}(\mathscr{G}_{\mathrm{LDsubset}}) = \mathrm{poly}(m, d^k, \log q),$$

$$\mathsf{Q\text{-}length}(\mathscr{G}_{\mathrm{LDsubset}}) = O(km \log q), \quad \mathsf{A\text{-}length}(\mathscr{G}_{\mathrm{LDsubset}}) = O(d^k \log(q)).$$

With probability $\frac{1}{2}$ each, perform one of the following two tests.

1. **Low-degree:** Perform $\mathscr{G}_{\text{Surface}}(m, d, q, 2)$.
2. **Cross-check:** Flip an unbiased coin $\boldsymbol{b} \sim \{0,1\}$. Let $\boldsymbol{s}$ be a uniformly random subspace of dimension $k+1$ containing the points in F. With probability $\frac{1}{2}$ each:
   (a) Let $\boldsymbol{w}$ be a uniformly random point in $\boldsymbol{s}$. Distribute the questions as follows:
   - Player $\boldsymbol{b}$: give $\boldsymbol{w}$; receive a value $\boldsymbol{y} \in \mathbb{F}_q$.
   - Player $\overline{\boldsymbol{b}}$: give $\boldsymbol{s}$; receive a degree-d polynomial $\boldsymbol{g} : \boldsymbol{s} \to \mathbb{F}_q$.

   Accept if $\boldsymbol{g}(\boldsymbol{w}) = \boldsymbol{y}$.
   (b) Distribute the questions as follows:
   - Player $\boldsymbol{b}$: give $\boldsymbol{s}$; receive a degree-d polynomial $\boldsymbol{g} : \boldsymbol{s} \to \mathbb{F}_q$.
   - Player $\overline{\boldsymbol{b}}$: give F; receive a function $\boldsymbol{f} : \boldsymbol{F} \to \mathbb{F}_q$.

   Accept if $\boldsymbol{g}|_F = \boldsymbol{f}$.

Figure 13: The game $\mathscr{G}_{\text{LDsubset}}(m, q, d, F)$.

Before proving this, we need the following proposition.

Proposition 16.7. *Let $F \subseteq \mathbb{F}_q^m$ be of size at most k. Consider the distribution $\mathcal{D}_{\text{twostep}}$ on points $x \in \mathbb{F}_q^m$ generated by the following two-step process: (i) let $\boldsymbol{s}$ be a uniformly random subspace of size $k+1$ containing F, and (ii) draw $\boldsymbol{x}$ uniformly at random from $\boldsymbol{s}$. Let $\mathcal{D}_{\text{unif}}$ be the uniform distribution on $\mathbb{F}_q^m$. Then $d_{\text{TV}}(\mathcal{D}_{\text{twostep}}, \mathcal{D}_{\text{unif}}) \le 1/q$.*

Proof. Let $x_1, \dots, x_\ell$ be a maximal set of linearly independent elements from F. A uniformly random subspace of size $k+1$ containing F can be generated as follows: first, pick a uniformly random nonzero vector $\boldsymbol{y}_{\ell+1}$ linearly independent from F, then pick a uniformly random nonzero vector $\boldsymbol{y}_{\ell+2}$ linearly independent from $F \cup \{\boldsymbol{y}_{\ell+1}\}$, and so forth. Set $\boldsymbol{s} = \operatorname{span}\{x_1, \dots, x_\ell, \boldsymbol{y}_{\ell+1}, \dots, \boldsymbol{y}_{k+1}\}$. A uniformly random point in $\boldsymbol{s}$ will be of the form

$$\boldsymbol{x} = \boldsymbol{t}_1 x_1 + \cdots + \boldsymbol{t}_\ell x_\ell + \boldsymbol{t}_{\ell+1}\boldsymbol{y}_{\ell+1} + \cdots + \boldsymbol{t}_{k+1}\boldsymbol{y}_{k+1},$$

where each $\boldsymbol{t}_i$ is a uniformly random element in $\mathbb{F}_q$. Because all the $\boldsymbol{y}_i$'s are linearly independent, the linear combination $\boldsymbol{t}_{\ell+1}\boldsymbol{y}_{\ell+1} + \cdots + \boldsymbol{t}_{k+1}\boldsymbol{y}_{k+1}$ is zero only when $\boldsymbol{t}_{\ell+1} = \cdots = \boldsymbol{t}_{k+1} = 0$. Otherwise, this linear combination is distributed as a uniformly random nonzero vector linearly independent from F. Thus, with probability $(q^{k+1-\ell})^{-1}$, $\boldsymbol{x}$ is distributed as a uniformly random vector in the span of F, and otherwise it is distributed as a uniformly random vector outside the span of F. Given that the span of F has q^ℓ points, the total variation distance is

$$\frac{1}{2}\left|\frac{q^\ell}{q^m} - \frac{1}{q^{k+1-\ell}}\right| + \frac{1}{2}\left|\frac{q^m - q^\ell}{q^m} - \left(1 - \frac{1}{q^{k+1-\ell}}\right)\right| = q^\ell\left(\frac{1}{q^{k+1}} - \frac{1}{q^m}\right) \le \frac{1}{q}. \qquad \square$$

Now we prove Theorem 16.6.

Proof of Theorem 16.6. The fact that Code is an $(n, m', q, d/q)$-error-correcting code follows from Schwartz-Zippel (Lemma 3.6).

Completeness. Let (ψ, M) be an EPR strategy in which $\{M_w\}$ is a measurement with outcomes in $\{0,1\}^n$. Consider the strategy (ψ, G) in which for any subset of points $F = \{y_1, \ldots, y_\ell\}$,

$$G^I_{a_1,\ldots,a_\ell} := M_{[g_x(y_1),\ldots,g_x(y_\ell)=a_1,\ldots,a_\ell]}.$$

(This covers the case of points ($\ell = 1$) and subsets F ($\ell = k$).) In addition, for any subspace s,

$$G^s_f := M_{[g_x|_s=f]}.$$

(This covers the case of the 2-dimensional subspaces used in $\mathscr{G}_{\text{Point}}$ and the $(k+1)$-dimensional subspaces used for the local decoding.) By construction, (ψ, G) is an EPR strategy, and it is easy to see that it is a *commuting* one as well.

We claim that (ψ, G) passes $\mathscr{G}_{\text{LDsubset}}(m, q, d, \mathcal{D})$ with probability 1. We begin with the low-degree test. By Fact 4.37, $M_w \otimes I_{\mathsf{Bob}} \simeq_0 I_{\mathsf{Alice}} \otimes M_w$. Then by Fact 4.26,

$$M_{[g_x(w)=b]} \otimes I_{\mathsf{Bob}} \simeq_0 I_{\mathsf{Alice}} \otimes M_{[g_x(w)=b]}.$$

This implies passing the low-degree test with probability 1, because

$$G^s_{[f(w)=b]} \otimes I_{\mathsf{Bob}} = M_{[g_x(w)=b]} \otimes I_{\mathsf{Bob}} \simeq_0 I_{\mathsf{Alice}} \otimes M_{[g_x(w)=b]} = I_{\mathsf{Alice}} \otimes G^w_b.$$

A similar argument shows the other tests pass with probability 1 as well.

Soundness. Let $\mathcal{D}$ be a distribution, and let (ψ, M) be a strategy which passes $\mathscr{G}_{\text{LDsubset}}(m, q, d, \mathcal{D})$ with probability $1 - \epsilon$. The outline of the proof is as follows: first we will use the low degree test in Item 1 to ensure the test correctly answers low-degree point queries. Item 2a will then bootstrap this to subspaces, and Item 2b will further bootstrap this to subsets, proving the theorem.

Using the low-degree test. Passing the test with probability $1 - \epsilon$ means passing the low-degree test with probability at least $1 - 2\epsilon$. By Theorem 4.40, this means that there exists a POVM measurement $G \in \text{PolyMeas}(m, d, q)$ such that

$$M^w_b \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g(w)=b]}, \qquad G_g \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_g, \tag{74}$$

where the first is on the uniform distribution over $\mathbb{F}_q^m$.

Bootstrapping to subspaces. Define $\mathcal{D}_{\text{twostep}}$ to be the two-step sampling process $(\boldsymbol{F}, \boldsymbol{s}, \boldsymbol{w})$ as in Item 2a. By Proposition 16.7, the marginal distribution on $\boldsymbol{w}$ has total variation distance at most $1/q$ with $\mathcal{D}_{\text{unif}}$. As a result, we can apply Fact 4.21 to Equation (74), yielding

$$M^w_b \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g(w)=b]} \tag{75}$$

on distribution $\mathcal{D}_{\text{twostep}}$. Similarly, by Fact 4.26,

$$G_{[g(w)=b]} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g(w)=b]} \tag{76}$$

on distribution $\mathcal{D}_{\text{twostep}}$.

Next, because the strategy passes the test in Item 2a with probability at least $1 - 4\epsilon$,

$$M^w_y \otimes I_{\mathsf{Bob}} \simeq_\epsilon I_{\mathsf{Alice}} \otimes M^s_{[g(w)=y]}. \tag{77}$$

on distribution $\mathcal{D}_{\text{twostep}}$. Combining Equations (75) to (77) with our second triangle inequality (Fact 4.29),

$$M^s_{[g(w)=y]} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g(w)=b]}.$$

By Proposition 4.42, we conclude that

$$M^s_f \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g|_s=f]}. \tag{78}$$

Concluding with subsets. The strategy passes the test in Item 2b with probability at least $1 - 4\epsilon$. As a result,

$$M_f^F \otimes I_{\mathsf{Bob}} \simeq_\epsilon I_{\mathsf{Alice}} \otimes M^s_{[g|_F=f]}.$$

Applying Fact 4.26 to Equation (78),

$$M^s_{[g|_F=f]} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[h|_F=h]}.$$

Similarly, applying Fact 4.26 to Equation (74),

$$G_{[h|_F=f]} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[h|_F=f]}$$

Applying the triangle inequality (Fact 4.29) to these three equations, we get

$$M_f^F \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_{[g|_F=f]}$$

with respect to distribution $\mathcal{D}_{\text{twostep}}$, and therefore, by Fact 4.23, with respect to $\mathcal{D}$. □

16.2 Efficiently decodable codes

Our application requires error-correcting codes with two further properties. The first property is that the decoding map $\mathrm{Dec}(\cdot)$ be efficiently computable. (The encoding map, on the other hand, is allowed arbitrary complexity. This is because we will leave the task of computing the encoding maps to the provers.) The second, more technical property is we require that the code *embed* the codeword, in the following sense: the encoding $\mathrm{Enc}(x)$ of a string x should actually *contain* the string x, and the function for where to find each bit of x in $\mathrm{Enc}(x)$ should be efficiently computable.

Definition 16.8 (Efficiently-decodable error-correcting codes)**.** Let $m, q : \mathbb{Z}^+ \to \mathbb{Z}^+$, and let $\eta : \mathbb{Z}^+ \to [0,1]$. Let $t_{\mathrm{Dec}}, t_{\mathrm{Emb}} : \mathbb{Z}^+ \to \mathbb{Z}^+$. We say that $\mathrm{Code}_n = (\mathrm{Enc}_n, \mathrm{Dec}_n, \mathrm{Sub}_n)$ is an $(n, m, q, \eta, t_{\mathrm{Dec}}, t_{\mathrm{Emb}})$-*efficient code family* if the following three conditions are true.

- For each n, $(\mathrm{Enc}_n, \mathrm{Dec}_n, \mathrm{Sub}_n)$ is an $(n, m(n), q(n), \eta(n))$-error-correcting code.
- There exists an algorithm $\mathrm{Alg}_{\mathrm{Dec}}$ which, on input (n, w), outputs $\mathrm{Dec}_n(w)$. Furthermore, $\mathrm{Alg}_{\mathrm{Dec}}$ runs in time $t_{\mathrm{Dec}}(n)$.
- There exists an embedding $\mu_n : [n] \to [m(n)]$ such that for each $i \in [n]$, $x_i = (\mathrm{Enc}_n(x))_{\mu_n(i)}$. Furthermore, there is an algorithm $\mathrm{Alg}_{\mathrm{Emb}}$ which, on input (n, i), computes $\mu_n(i)$ in time $t_{\mathrm{Emb}}(n)$.

Now, we show that the low-degree code is efficiently-decodable. The decoding algorithm follows a simple strategy: assuming that the input is a proper encoding of a message, they can directly read off the message from the input. Then they compute the encoding of the purported message and check that it equals the input.

Fact 16.9. *There is a* $(n, m', q, \eta, t_{\mathrm{Dec}}, t_{\mathrm{Emb}})$*-error-correcting code* Code *with parameters set as follows:*

$$m'(n) = \mathrm{poly}(n), \quad q(n) = \mathrm{polylog}(n), \quad \eta(n) = \frac{1}{\mathrm{polylog}(n)},$$
$$t_{\mathrm{Dec}}(n) = \mathrm{poly}(n), \qquad t_{\mathrm{Emb}}(n) = \mathrm{polylog}(n).$$

In addition, Code *has a* k*-subset test* $\mathscr{G}$ *with robustness* $\delta(\epsilon) = \mathrm{poly}(\epsilon, 1/\log(n))$ *such that*

$$\mathsf{Q\text{-}time}(\mathscr{G}) = \mathrm{poly}(\log n, k), \quad \mathsf{A\text{-}time}(\mathscr{G}) = \mathrm{poly}(\log(n)^k),$$
$$\mathsf{Q\text{-}length}(\mathscr{G}) = O(k \log n), \quad \mathsf{A\text{-}length}(\mathscr{G}) = O(\log(n)^{2k}).$$

Proof. We instantiate the canonical low-degree encoding from Definition 3.8 with the "rule of thumb" parameters from Equation (1):

$$h(n) = \Theta(\log(n)), \qquad m(n) = \Theta\left(\frac{\log(n)}{\log\log(n)}\right), \qquad q(n) = \mathrm{polylog}(n).$$

If we set $d(n) = m(n) \cdot (h(n) - 1)$, then this is a code with distance $\eta(n) = 1 - d(n)/q(n) = 1 - 1/\mathrm{polylog}(n)$. In addition, it has length $m'(n) = q(n)^{m(n)} = \mathrm{poly}(n)$. Finally, the canonical low-degree encoding gives us the embedding $\mu_{\mathrm{Emb}} := \sigma_{m,t_1,t_2}$. By Proposition 3.9, it takes time $t_{\mathrm{Emb}}(n) = \mathrm{polylog}(n)$ to compute.

Now we design the decoding algorithm $\mathrm{Alg}_{\mathrm{Dec}}$. On input (n, w), it rejects if w is not length m'. Otherwise, it interprets w as a function $f : \mathbb{F}_q^m \to \mathbb{F}_q$. It queries g on the points $\pi(1), \ldots, \pi(n)$. Let $a \in \{0,1\}^n$ be the received answers. If g is a codeword, it equals the low-degree function g_a. So the algorithm simply iterates over all $x \in \mathbb{F}_q^m$ and checks that $f(x) = g_a(x)$. By Proposition 3.9, computing $g_a(x)$ can be done in time $\mathrm{poly}(n)$, and so this takes time $t_{\mathrm{Dec}}(n) = \mathrm{poly}(n)$ in total.

Finally, the performance of the subset tester follows from Theorem 16.6 with our setting of parameters. □

17 Answer reduction

In this section, we carry out the answer reduction. Our main result will be to take the $\mathrm{poly}(n)$ question complexity, $O(2^n)$ answer complexity MIP^* protocol for Succinct-Succinct-3Sat given by Corollary 15.9 and convert it to one whose answer complexity is also $\mathrm{poly}(n)$; this is Theorem 17.12 below.

Our answer reduction will apply to any game with a value-1 real commuting EPR strategy. We will require two properties of these strategies: first, that they can be extended to strategies that pass subset tests with probability 1, as in Definition 16.3; and second, that they are "oracularizable". We explain this second property in the next section.

17.1 Oracularization

Our technique will not work for all entangled games but only for a subset, for which a single prover can simulate both prover's actions if required to. We call such games "oracularizable" games.

Definition 17.1. Given a two-player entangled game $\mathscr{G}$, its *oracularization* is the game $C_{\mathrm{oracle}}(\mathscr{G})$ given in Figure 14. If $\mathscr{G}$ is value-1, then we call it *oracularizable*, if $\mathrm{val}(C_{\mathrm{oracle}}(\mathscr{G})) = 1$ as well. We also note that for *any* game $\mathscr{G}$, if $\mathrm{val}(\mathscr{G}) \leq 1 - \delta$, then $\mathrm{val}(C_{\mathrm{oracle}}(\mathscr{G})) \leq 1 - O(\delta)$.

A real commuting EPR strategy allows "Player $\boldsymbol{b}$" to sample both questions $\boldsymbol{x}_0$ and $\boldsymbol{x}_1$ simultaneously. As a result, if a game $\mathscr{G}$ has a value-1 real commuting EPR strategy, then it is oracularizable.

The value of oracularization is that when the verifier checks $V(\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{a}_0, \boldsymbol{a}_1) = 1$, both $\boldsymbol{a}_0$ and $\boldsymbol{a}_1$ come from the same prover rather than two different provers. This seems like a minor change, but in fact it makes all the difference. Our goal is to reduce the verifier's runtime by having the provers encode their answers using PCP technology. When the answers come from *both* provers, the relevant piece of PCP technology is a *distributed* PCP, but it is known by a simple argument of Reingold that distributed PCPs do not exist (see the discussion in [ARW17]). The key difficulty comes from the fact that Alice needs to prepare her PCP proof without knowing Bob's question and answer, and vice versa, and this turns out to be impossible in general. On the other hand, when the answers come from a single prover, we can use traditional PCPs to implement the

Given a game $\mathscr{G}$, sample a tuple $(\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{C}) \sim \mathscr{G}$, and flip two unbiased coins $\boldsymbol{b}, \boldsymbol{c} \sim \{0,1\}$. With probability $\frac{1}{2}$ each, perform one of the following two tests.

1. **Verify:** Distribute the questions as follows:
   - Player $\boldsymbol{b}$: send the pair $(\boldsymbol{x}_0, \boldsymbol{x}_1)$ and receive answers $(\boldsymbol{a}_0, \boldsymbol{a}_1)$.
   - Player $\overline{\boldsymbol{b}}$: send $\boldsymbol{x}_{\boldsymbol{c}}$ and receive an answer $\boldsymbol{a}_2$.
2. **Consistency:** Play the consistency game with question $\boldsymbol{x}_0, \boldsymbol{x}_1$.

Accept if $\boldsymbol{a}_2 = \boldsymbol{a}_{\boldsymbol{c}}$ and $V(\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{a}_0, \boldsymbol{a}_1) = 1$.

Figure 14: The oracularized game $C_{\text{oracle}}(\mathscr{G})$.

answer reduction, of which we have a variety of constructions. We note that oracularized games *do* still have checks between players, but these are equality checks and will be easy to implement in the answer reduction regime.

17.2 Probabilistically checkable proofs of proximity

In this section, we introduce the main PCP technology we will use for our answer reduction. In the oracularized game, the provers want to convince us not just that $V(\cdot,\cdot,\cdot,\cdot)$ is satisfiable—which we already know to be true by construction—but that $(\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{a}_0, \boldsymbol{a}_1)$ is a particular assignment which satisfies it. For this, we need a stronger notion of a PCP called a *probabilistically checkable proof of proximity (PCPP)*. These allow one to check that an input x is close to a satisfying assignment of a circuit C (hence the "proximity") by making a small number of queries to x. These were originally introduced in the independent works of [BSGH+06] and [DR06] (where they were called *assignment testers*).

In our case, we will need even stronger PCPPs in which the verifier is not only query-efficient but *time*-efficient as well. The history of these time-efficient PCPPs goes back to the original proof of $\mathsf{MIP} = \mathsf{NEXP}$ and the various attempts to "scale it down" [O'D05]. The most famous line of research considered proof systems in which the verifier's query complexity is restricted, and this eventually led to the proof of the PCP theorem [AS98, ALM+98]. A parallel line of research considered proof systems in which the verifier's runtime is restricted (so-called "transparent" proofs) [BFLS91]. The latter of these was revisited in the work of Ben-Sasson et al. [BSGH+05], who showed that both lines of research could be remerged in the "scaled down" setting by constructing a PCPP in which the verifier is both query-efficient *and* time-efficient. Though their main result is actually sufficient for our purposes, we will cite the work of Mie [Mie09], which improves on their result in the regime we care about. Finally, we note the work of Meir [Mei14], who reproves the bounds of Ben-Sasson et al. [BSGH+05] using combinatorial methods.

To our knowledge, ours is the first use of a time-efficient PCPP specifically for its time-efficient properties in the quantum literature. Natarajan and Vidick [NV18a] used the time-efficient PCPP of [BSGH+05] to prove the quantum games PCP conjecture, but the property they needed was not that it was time-efficient, but that the bits of the proof are linear functions of the bits of the assignment. We note that we do not need this property here.

In this literature, it is common to consider "pair languages" consisting of strings (x, y) in which x is small and given to the verifier and y is large and accessible only through query access. This maps perfectly onto our scenario, in which the verifier supplies the "small" questions $\boldsymbol{x}_0, \boldsymbol{x}_1$ and the prover supplies the "large" answers $\boldsymbol{a}_0, \boldsymbol{a}_1$.

Definition 17.2. A *pair language* L is a subset of $\{0,1\}^* \times \{0,1\}^*$. Given $x \in \{0,1\}^*$, we write $L_x = \{y \in \{0,1\}^* \mid (x,y) \in L\}$.

The next two definitions state the notion of an efficient PCPP verifier.

Definition 17.3 ([BSGH+05, Definition 2.1])**.** Let $r, q : \mathbb{Z}^+ \to \mathbb{Z}^+$ and $t : \mathbb{Z}^+ \times \mathbb{Z}^+ \to \mathbb{Z}^+$. An (r,q,t)*-restricted PCPP verifier* is a probabilistic machine that, given a string x (called the *explicit input*) and a number K (in binary) as well as oracle access to an *implicit input* $y \in \{0,1\}^K$ and to a *proof oracle* $\pi \in \{0,1\}^*$, tosses $r(|x|+K)$ coins, queries the oracles (y,π) for a total of $q(|x|+K)$ symbols, runs in time $t(|x|,K)$, and outputs a Boolean verdict in $\{\text{accept}, \text{reject}\}$.

Definition 17.4 ([BSGH+05, Definition 2.2])**.** For functions $r, q : \mathbb{Z}^+ \to \mathbb{Z}^+$, $t : \mathbb{Z}^+ \times \mathbb{Z}^+ \to \mathbb{Z}^+$, and constants $s, \gamma \in [0,1]$, a pair language $L \subseteq \{0,1\}^* \times \{0,1\}^*$ is in $\text{PCPP}_{s,\gamma}[r,q,t]$ if there exists an (r,q,t)-restricted PCPP verifier V with the following properties:

- **Completeness:** If $(x,y) \in L$ then there exists a π such that $\mathbf{Pr}_R[V^{y,\pi}(x,|y|;R) \text{ accepts}] = 1$, where $V^{y,\pi}(x,|y|;R)$ denotes the decision V on input $(x,|y|)$, oracle access to (y,π), and coin tosses R.
- **Soundness:** If (x,y) is such that y is γ-far from $L_x \cap \Sigma^{|y|}$, then for every π it holds that $\mathbf{Pr}_R[V^{y,\pi}(x,|y|;R) \text{ accepts}] \leq s$.

Mie's time-efficient PCPP is states as follows.

Theorem 17.5 ([Mie09, Theorem 1])**.** *Suppose that L is a pair language in $\mathsf{NTIME}(T)$ for some non-decreasing function $T : \mathbb{Z}^+ \to \mathbb{Z}^+$. Then, for every two constants $s, \gamma > 0$, we have $L \in \text{PCPP}_{s,\gamma}[r,q,t]$, for*

- *Randomness complexity* $r(m) = \log_2 T(m) + O(\log\log T(m))$.
- *Query complexity* $q(m) = O(1)$,
- *Verification time* $t(n,K) = \text{poly}(n, \log K, \log T(n+K))$.

We note that this is in fact a much stronger than what we will actually need. In particular, we will only apply this to languages L in *deterministic* $\mathsf{TIME}(T)$, which are trivially in $\mathsf{NTIME}(T)$.

17.3 Composing with an error-correcting code

The verifier in a PCPP rejects any input which is γ-far from an accepting input, but of course we want our verifier to reject *all* non-accepting inputs, no matter their distance. To do this, we will (i) encode the verifier's inputs using an error-correcting code and (ii) check that the inputs are properly encoded (using, for example, the low-degree test). This approach of composing a PCPP with an error-correcting code is standard and stretches back in spirit to the transparent proofs of [BFLS91] (see the discussion of this in [BSGH+06]).

Now we show how to compose an MIP^* game with an error-correcting code.

Definition 17.6 (Error-correcting the provers' answers)**.** Let $V = (\text{Alg}_\text{Q}, \text{Alg}_\text{A})$ be an MIP^* verifier (the language it verifies is not important). Suppose on inputs of size n it has question length $\ell_\text{Q}(n)$ answer length $\ell_\text{A}(n)$. Write L_A for the language decided by Alg_A. Let $\text{Code}_k = (\text{Enc}_k, \text{Dec}_k, \text{Sub}_k)$ be a $(k, m, q, \eta, t_\text{Dec}, t_\text{Emb})$-efficient code family with decoding algorithm Alg_Dec. Then $L_\text{A} \circ \text{Code}$ is a new language defined as follows: suppose $(\mathsf{input}, x_0, x_1, y_0, y_1) \in L_\text{A}$. Let n be the length of input and $\ell = \ell_\text{A}(n)$. Then $(\mathsf{input}, x_0, x_1, \text{Enc}_\ell(y_0), \text{Enc}_\ell(y_1)) \in L_\text{A} \circ \mathsf{Code}$.

Now, we prove a couple of properties about the composed verifier. First, we show that its runtime is not much slower than the original verifier's.

Proposition 17.7 (Runtime of the composed verifier)**.** *Let V and Code_k be as in Definition 17.6. Suppose Alg_A runs in time $T(n)$. Then there is an algorithm, which we denote $\mathrm{Alg}_\mathrm{A} \circ \mathsf{Code}$, deciding the language $L_\mathrm{A} \circ \mathsf{Code}$. In addition, on inputs $(\mathsf{input}, x_0, x_1, z_0, z_1)$ in which $|\mathsf{input}| = n$, $|x_0| = |x_1| = \ell_\mathrm{Q}(n)$, and $|z_0| = |z_1| = m(\ell_\mathrm{A}(n))$, the algorithm runs in time $T(n) + t_\mathrm{Dec}(\ell_\mathrm{A}(n))$.*

Proof. On input $(\mathsf{input}, x_0, x_1, z_0, z_1)$, we define the action of $\mathrm{Alg}_\mathrm{A} \circ \mathsf{Code}$ as follows.

1. Compute n, the length of input. Set $\ell := \ell_\mathrm{A}(n)$.
2. Check that z_0 and z_1 have length $m(\ell)$. If they don't, reject.
3. Compute $y_0 = \mathrm{Alg}_\mathrm{Dec}(\ell, z_0)$ and $y_1 = \mathrm{Alg}_\mathrm{Dec}(\ell, z_1)$. If either y_0 or y_1 is $\perp$, reject.
4. Otherwise, we know that $y_0, y_1 \in \{0,1\}^\ell$. Run $\mathrm{Alg}_\mathrm{A}(\mathsf{input}, x_0, x_1, y_0, y_1)$. Accept if it accepts, and reject if it rejects.

It is immediate that $\mathrm{Alg}_\mathrm{A} \circ \mathsf{Code}$ computes $L_\mathrm{A} \circ \mathsf{Code}$. As for the time complexity, Item 3 runs in time $t_\mathrm{Dec}(\ell_\mathrm{A}(n))$ and Item 4 runs in time $T(n)$. Combined, these two give the bound in the proposition statement. □

Next, we show that this construction solves the "problem" discussed at the beginning of the section, namely that if we perform answer reduction by replacing $\mathrm{Alg}_\mathrm{A} \circ \mathsf{Code}$ with a PCPP verifier, rather than just Alg_A, then the verifier will reject *all* inputs which are not in the language, not just those which are δ-far, provided that those inputs are encoded as per Definition 17.6.

Proposition 17.8. *Let V and Code_k be as in Definition 17.6. Let $s, \gamma > 0$ be constants, and let V_PCPP be the PCPP verifier for the language $L_\mathrm{A} \circ \mathsf{Code}$ guaranteed by Theorem 17.5 with these parameters. Suppose that $1 - \eta(k) \geq 2\gamma$ for all k. Then we have the following soundness condition.*

- ***Soundness:*** *Consider $(\mathsf{input}, x_0, x_1, z_0, z_1)$ for input of length n, x_0 and x_1 of length $\ell_\mathrm{Q}(n)$, and $z_0, z_1 \in \mathrm{Sub}_\ell$, for $\ell := \ell_\mathrm{A}(n)$. Suppose this does not correspond to the encoding of an accepting assignment in L_A. In other words, suppose that there are no $y_0, y_1 \in \{0,1\}^\ell$ such that $(\mathsf{input}, x_0, x_1, y_0, y_1)$ is in L_A and $z_0 = \mathrm{Enc}_\ell(y_0), z_1 = \mathrm{Enc}_\ell(y_1)$. Then V_PCPP accepts $(\mathsf{input}, x_0, x_1, z_0, z_1)$ with probability at most s. In math, for every π it holds that*

$$\Pr_R[V_\mathrm{PCPP}^{z_0, z_1, \pi}(\mathsf{input}, x_0, x_1, |z_0| + |z_1|; R) \text{ accepts}] \leq s.$$

Proof. Given $(\mathsf{input}, x_0, x_1, z_0, z_1)$, write $A := (L_\mathrm{A} \circ \mathsf{Code})_{\mathsf{input}, x_0, x_1} \cap \mathbb{F}_{q(\ell)}^{|z_0|+|z_1|}$. By assumption, (z_0, z_1) is not in A. Using this, we would like to show that (z_0, z_1) is in fact γ-far from A, in which case the PCPP verifier accepts with probability at most s.

To do this, suppose $(z_0', z_1') \in A$. By design, there exists $y_0', y_1' \in \{0,1\}^\ell$ such that $z_0' = \mathrm{Enc}(y_0')$ and $z_1' = \mathrm{Enc}(y_1')$. This means that $z_0', z_1' \in \mathrm{Sub}_\ell$. On the other hand, since (z_0, z_1) is not in A, we must have either $z_0' \neq z_0$ or $z_1' \neq z_1$ (or both). We will assume the first without loss of generality. Then by the distance property of the code, since $z_0, z_0' \in \mathrm{Sub}_\ell$, their normalized Hamming distance is at least $1 - \eta(\ell) \geq 2\gamma$. This immediately means that (z_0, z_1) and (z_0', z_1'). are at least γ-far apart, and we are done. □

17.4 The answer reduction protocol

We are almost ready to state the answer reduction protocol. Before doing so, we discuss one final nuisance, which is that we will also need the prover to encode their *proof* with an error-correcting code. The reason is that we would like to query the proof on a view $\boldsymbol{J}$ sampled by the PCPP verifier. However, the prover might cheat and respond based only on the view $\boldsymbol{J}$ rather than a global proof π. To prevent this, we force them to commit to a global error-correcting encoding of their proof π using a tester as in Definition 16.3. Then, we use the fact that the error-correcting code *embeds* their string to allow us to extract the view $\boldsymbol{J}$ by asking for the coordinates in $\mu(\boldsymbol{J})$.

We now state the answer reduction protocol.

Definition 17.9. We instantiate the answer-reduced MIP^* protocol with the following algorithms and parameters.

- Let $V = (\mathrm{Alg}_{\mathrm{Q}}, \mathrm{Alg}_{\mathrm{A}})$ be an MIP^* verifier for a language L. Write L_{A} for the language decided by $\mathrm{Alg}_{\mathrm{A}}$. Suppose on inputs of size n, the verifier V has question length $\ell_{V,\mathrm{Q}}(n)$, answer length $\ell_{V,\mathrm{A}}(n)$, question time $t_{V,\mathrm{Q}}(n)$, and answer time $t_{V,\mathrm{A}}(n)$.
- Let $\mathrm{Code}_k = (\mathrm{Enc}_k, \mathrm{Dec}_k, \mathrm{Sub}_k)$ be a $(k, m, q, \eta, t_{\mathrm{Dec}}, t_{\mathrm{Emb}})$-efficient code family with decoding algorithm $\mathrm{Alg}_{\mathrm{Dec}}$ and embedding μ_k.
- Let $\mathscr{G}_k$ be a game which tests for Code_k with robustness $\chi_k(\epsilon)$. Suppose it has question length $\ell_{\mathscr{G},\mathrm{Q}}(k)$, answer length $\ell_{\mathscr{G},\mathrm{A}}(k)$, question time $t_{\mathscr{G},\mathrm{Q}}(k)$, and answer time $t_{\mathscr{G},\mathrm{A}}(k)$.
- Let $s, \delta > 0$ be constants, and let V_{PCPP} be the PCPP verifier for the language $L_{\mathrm{A}} \circ \mathsf{Code}$ guaranteed by Theorem 17.5 with these parameters. Suppose on inputs of size n it has proof length $\ell_\pi(n)$. By Proposition 17.7, $L_{\mathrm{A}} \circ \mathrm{Code}$ is in time $t_{\mathrm{compose}}(n) = t_{V,\mathrm{A}}(n) + t_{\mathrm{Dec}}(\ell_{V,\mathrm{A}}(n))$. We can therefore write V_{PCPP}'s verification time as

$$t_{\mathrm{PCPP}}(n) = \mathrm{poly}(n + \ell_{V,\mathrm{Q}}(n), \log(m(\ell_{V,\mathrm{A}}(n))), \log(t_{\mathrm{compose}}(n))).$$

  Finally, $\ell_\pi(n) = t_{\mathrm{compose}}(n) \cdot \mathrm{polylog}(t_{\mathrm{compose}}(n))$.

Write $\ell_1 := \ell_{V,\mathrm{A}}(n)$ and $\ell_2 := \ell_\pi(n)$. Then the *answer reduction game* $\mathscr{G}_{\mathrm{answer}}(\mathsf{input}; V, \mathrm{Code}, \mathscr{G}, s, \delta)$ is given in Figure 15. We write V_{answer} for the corresponding verifier.

Theorem 17.10. *Suppose V, Code, $\mathscr{G}$, and V_{PCPP} are as in Definition 17.9. Suppose s, γ are chosen to be constants such that $\eta(k) \geq 2\gamma$ for all k. Suppose further that V has the following property: for any input in L, the provers have a real commuting EPR strategy with value 1. Then V_{answer} is also an MIP^* verifier for L with the following two conditions:*

- *(Completeness) If $\mathsf{input} \in L$, then there is a value-1 strategy.*
- *(Soundness) Given input, suppose there is a strategy with value $1 - \epsilon$. Then there is a strategy for V on input input with value $1 - \delta(\epsilon)$, where $\delta(\epsilon)$ is given by*

$$\delta(\epsilon) := \mathrm{poly}(\chi_{\ell_1}(\mathrm{poly}(\epsilon)), \chi_{\ell_2}(\mathrm{poly}(\epsilon)), \eta(\ell_1), \eta(\ell_2)).$$

Hence, if we choose our parameters so that $1 - \delta(\epsilon)$ is greater than the soundness of V, this implies that V_{answer} is an MIP^* verifier for L with soundness $1 - \epsilon$.

Flip two unbiased coins $\boldsymbol{b}, \boldsymbol{c} \sim \{0,1\}$. Sample questions $(\boldsymbol{x}_0, \boldsymbol{x}_1) \sim \mathrm{Alg}_{\mathrm{Q}}(\mathsf{input})$. Sample a view $\boldsymbol{I}_0, \boldsymbol{I}_1, \boldsymbol{J} \sim V_{\mathrm{PCPP}}(\mathsf{input}, \boldsymbol{x}_0, \boldsymbol{x}_1)$. Set $\boldsymbol{J}' = \mu_{\ell_2}(\boldsymbol{J})$. Select $\boldsymbol{i}_0, \boldsymbol{i}_1 \in [m(\ell_1)]$ and $\boldsymbol{j} \in [m(\ell_\pi(n))]$ uniformly at random. Set $\boldsymbol{T}_0 = \boldsymbol{I}_0 \cup \{\boldsymbol{i}_0\}$, $\boldsymbol{T}_1 = \boldsymbol{I}_1 \cup \{\boldsymbol{i}_1\}$, and $\boldsymbol{U} = \boldsymbol{J}' \cup \{\boldsymbol{j}\}$. With probability $\frac{1}{8}$ each, perform one of the following eight tests.

1. **Verify:** Distribute the question as follows:
   - Player $\boldsymbol{b}$: give $(\boldsymbol{x}_0, \boldsymbol{x}_1)$, $\boldsymbol{T}_0, \boldsymbol{T}_1, \boldsymbol{U}$; receive $\boldsymbol{a}_0, \boldsymbol{a}_1, \boldsymbol{a}_2$.

   Accept if $V_{\mathrm{PCPP}}(\mathsf{instance}, \boldsymbol{x}_0, \boldsymbol{x}_1)$ accepts on $\boldsymbol{a}_0|_{\boldsymbol{I}_0}, \boldsymbol{a}_1|_{\boldsymbol{I}_1}, \boldsymbol{a}_2|_{\boldsymbol{J}'}$.
2. **Cross checks:**
   (a) **Consistency test:** Distribute the questions as follows:
   - Player $\boldsymbol{b}$: give $(\boldsymbol{x}_0, \boldsymbol{x}_1)$, $\boldsymbol{T}_0, \boldsymbol{T}_1, \boldsymbol{U}$; receive $\boldsymbol{a}_0, \boldsymbol{a}_1, \boldsymbol{a}_2$.
   - Player $\overline{\boldsymbol{b}}$: give $(\boldsymbol{x}_0, \boldsymbol{x}_1)$, $\boldsymbol{T}_0, \boldsymbol{T}_1, \boldsymbol{U}$; receive $\boldsymbol{a}'_0, \boldsymbol{a}'_1, \boldsymbol{a}'_2$.

   Accept if $\boldsymbol{a}_0 = \boldsymbol{a}'_0$, $\boldsymbol{a}_1 = \boldsymbol{a}'_1$, and $\boldsymbol{a}_2 = \boldsymbol{a}'_2$.

   (b) **Answer cross-check:** Distribute the questions as follows:
   - Player $\boldsymbol{b}$: give $(\boldsymbol{x}_0, \boldsymbol{x}_1)$, $\boldsymbol{T}_0, \boldsymbol{T}_1, \boldsymbol{U}$; receive $\boldsymbol{a}_0, \boldsymbol{a}_1, \boldsymbol{a}_2$.
   - Player $\overline{\boldsymbol{b}}$: give $\boldsymbol{x}_{\boldsymbol{c}}, \boldsymbol{T}'_{\boldsymbol{c}}$; receive $\boldsymbol{a}'_{\boldsymbol{c}}$.

   Accept if $\boldsymbol{a}_{\boldsymbol{c}} = \boldsymbol{a}'_{\boldsymbol{c}}$.

   (c) **Proof cross-check:** Distribute the questions as follows:
   - Player $\boldsymbol{b}$: give $(\boldsymbol{x}_0, \boldsymbol{x}_1)$, $\boldsymbol{T}_0, \boldsymbol{T}_1, \boldsymbol{U}$; receive $\boldsymbol{a}_0, \boldsymbol{a}_1, \boldsymbol{a}_2$.
   - Player $\overline{\boldsymbol{b}}$: give $\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{U}$; receive $\boldsymbol{a}'_2$.

   Accept if $\boldsymbol{a}_2 = \boldsymbol{a}'_2$.
3. **Code checks:**
   (a) **Answer code check:** Sample questions $(\boldsymbol{w}_0, \boldsymbol{w}_1) \sim \mathscr{G}_{\ell_1}(\boldsymbol{T}_{\boldsymbol{c}})$. Distribute the questions as follows:
   - Player $\boldsymbol{b}$: give $\boldsymbol{x}_{\boldsymbol{c}}, \boldsymbol{w}_0$; receive $\boldsymbol{a}_0$.
   - Player $\overline{\boldsymbol{b}}$: give $\boldsymbol{x}_{\boldsymbol{c}}, \boldsymbol{w}_1$; receive $\boldsymbol{a}_1$.

   Accept if $\mathscr{G}_{\ell_1}(\boldsymbol{T}_{\boldsymbol{c}})$ accepts on $\boldsymbol{a}_0, \boldsymbol{a}_1$.

   (b) **Proof code check:** Sample questions $(\boldsymbol{w}_0, \boldsymbol{w}_1) \sim \mathscr{G}_{\ell_2}(\boldsymbol{U})$. Distribute the questions as follows:
   - Player $\boldsymbol{b}$: give $\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{w}_0$; receive $\boldsymbol{a}_0$.
   - Player $\overline{\boldsymbol{b}}$: give $\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{w}_1$; receive $\boldsymbol{a}_1$.

   Accept if $\mathscr{G}_{\ell_2}(\boldsymbol{U})$ accepts on $\boldsymbol{a}_0, \boldsymbol{a}_1$.

Figure 15: The answer reduction game $\mathscr{G}_{\mathrm{answer}}(\mathsf{input}; V, \mathrm{Code}, \mathscr{G}, s, \delta)$.

Furthermore, the question and answer lengths and runtimes are dominated by two subroutines: the "Verify" subroutine S_1 and the "Code Check" subroutine S_2 (consisting of both the answer code check and the proof code check). The complexity of the Verify subroutine is

$$\begin{aligned}
\mathsf{Q\text{-}length}(S_1) &= O(\ell_{V,\mathrm{Q}}(n) + \log(m(\ell_{V,\mathrm{A}}(n))) + \log(m(\ell_\pi(n)))),\\
\mathsf{A\text{-}length}(S_1) &= O(\log(q(\ell_{V,\mathrm{A}}(n))) + \log(q(\ell_\pi(n)))),\\
\mathsf{Q\text{-}time}(S_1) &= O\left(t_{V,\mathrm{Q}}(n) + t_{\mathrm{PCPP}}(n) + t_{\mathrm{Emb}}(\ell_\pi(n))\right),\\
\mathsf{A\text{-}time}(S_1) &= O(t_{\mathrm{PCPP}}(n)).
\end{aligned}$$

In addition, the complexity of the Code Check subroutine is

$$\begin{aligned}
\mathsf{Q\text{-}length}(S_2) &= O(\ell_{\mathscr{G},\mathrm{Q}}(\ell_{V,\mathrm{A}}(n)) + \ell_{\mathscr{G},\mathrm{Q}}(\ell_\pi(n)) + \ell_{V,\mathrm{Q}}(n)),\\
\mathsf{A\text{-}length}(S_2) &= O(\ell_{\mathscr{G},\mathrm{A}}(\ell_{V,\mathrm{A}}(n)) + \ell_{\mathscr{G},\mathrm{A}}(\ell_\pi(n))),\\
\mathsf{Q\text{-}time}(S_2) &= O(t_{\mathscr{G},\mathrm{Q}}(\ell_{V,\mathrm{A}}(n)) + t_{\mathscr{G},\mathrm{Q}}(\ell_\pi(n)) + t_{V,\mathrm{Q}}(n) + t_{\mathrm{Emb}}(\ell_\pi(n))),\\
\mathsf{A\text{-}time}(S_2) &= O(t_{\mathscr{G},\mathrm{A}}(\ell_{V,\mathrm{A}}(n)) + t_{\mathscr{G},\mathrm{A}}(\ell_\pi(n))).
\end{aligned}$$

Thus, the complexity of the overall protocol is the sum of these two.

Proof. The fact that S_1 and S_2 dominate the lengths and runtimes of the protocol is because S_1 dominates the lengths and runtimes of the two cross-check subroutines, whose questions and answers are subsets of those in S_1. Now we compute the complexity of S_1.

- **Question length:** The pair $(\boldsymbol{x}_0, \boldsymbol{x}_1)$ has total length $\ell_{V,\mathrm{Q}}(n)$ by definition. The pair $\boldsymbol{I}_0, \boldsymbol{I}_1$ are subsets of indices of constant size into each of the implicit inputs of $L_\mathrm{A} \circ \mathrm{Code}$, which are supposed to be encodings of strings of size $\ell_{V,\mathrm{A}}(n)$. Hence, the encodings have size $m(\ell_{V,\mathrm{A}}(n))$, and so each input is specified with the log of this many bits. Finally, $\boldsymbol{J}$ is a constant-sized set of indices into a proof of size $\ell_\pi(n)$, and $\mu(\boldsymbol{J})$ converts these into indices into an encoding of of this proof. As the encoding has length $m(\ell_\pi(n))$, each index can be specified with the log of this many bits.
- **Answer length:** The strings $\boldsymbol{a}_0, \boldsymbol{a}_1$ contains values from an error-correcting code with alphabet $q(\ell_{V,\mathrm{A}}(n))$, and the string $\boldsymbol{a}_2$ contains values from an error-correcting code with alphabet $q(\ell_\pi(n))$.
- **Question time:** The running time of Alg_Q is $t_{V,\mathrm{Q}}(n)$. The running time of V_{PCPP} is $t_{\mathrm{PCPP}}(n)$. Finally, the running time to compute $\mu(\boldsymbol{J})$ given $\boldsymbol{J}$ is $t_{\mathrm{Emb}}(\ell_\pi(n))$.
- **Answer time:** The running time is simply the running time of V_{PCPP}, i.e. $t_{\mathrm{PCPP}}(n)$.

As for the complexity of S_2, it just performs the code tester $\mathscr{G}_k$ for message lengths $k = \ell_{V,\mathrm{A}}(n)$ and $\ell_\pi(n)$ and so inherits the lengths and runtimes of the tester for these two values of k, except on top of that it also has to sample $(\boldsymbol{x}_0, \boldsymbol{x}_1)$ and compute $\boldsymbol{J}'$. Sampling $(\boldsymbol{x}_0, \boldsymbol{x}_1)$ takes time $t_{V,\mathrm{Q}}(n)$ and contributes $O(\ell_{V,\mathrm{Q}}(n))$ to the question lengths, and computing $\boldsymbol{J}'$ takes time $t_{\mathrm{Emb}}(\ell_\pi(n))$.

Completeness. Suppose input is in L. Then there is a real commuting EPR strategy (ψ, M) with value 1 for V on input. We will use this to demonstrate a value-1 strategy for V_{answer}. This will be the strategy (ψ, G) which uses the same EPR state $|\psi\rangle$ and has measurement matrices G defined as follows.

Fix questions x_0, x_1, T_0, T_1, and U. We begin by defining the simplest measurement,

$$G_{a_c}^{x_c,T_c} := M^{x_c}_{[\mathrm{Enc}_{\ell_1}(z_c)|_{T_c}=a_c]}. \tag{79}$$

If Alice and Bob measure with M^{x_0} and M^{x_1} and receive strings z_0, z_1, then because this strategy is value 1, we will always have $V(\mathsf{input}, x_0, x_1, z_0, z_1) = 1$. As a result, there always exists *some* proof for V_{PCPP} that $(\mathsf{input}, x_0, x_1, \mathrm{Enc}_{\ell_1}(z_0), \mathrm{Enc}_{\ell_1}(z_1))$ is in $L_{\mathrm{A}} \circ \mathsf{Code}$. We denote this proof $\pi(x_0, x_1, z_0, z_1)$; if there are multiple such proofs, we pick one arbitrarily. Then we define the measurement

$$G_{a_2}^{x_0,x_1,U} := (M^{x_0} \cdot M^{x_1})_{[\mathrm{Enc}_{\ell_2}(\pi(x_0,x_1,z_0,z_1))|_U = a_2}. \tag{80}$$

Next, we define the measurement

$$G_{a_0,a_1,a_2}^{x_0,x_1,T_0,T_1,U} := (M^{x_0} \cdot M^{x_1})_{[\mathrm{Enc}_{\ell_1}(z_0)|_{T_0}, \mathrm{Enc}_{\ell_1}(z_1)|_{T_1}, \mathrm{Enc}_{\ell_2}(\pi(x_0,x_1,z_0,z_1))|_U = a_0,a_1,a_2]}.$$

Now, via Equations (79) and (80), the G measurement is exactly of the form required by Definition 16.3. As a result, it can be extended to a measurement which passes the answer and proof code checks with probability 1. Performing this extension concludes the design of the strategy.

By construction, this strategy passes the answer and proof code checks with probability 1. As for the remaining tests, let us begin with the answer cross-check in the case of $\boldsymbol{c} = 0$, the other case being symmetric. Because M is a real commuting EPR strategy, by Fact 4.37 we have that $M_a^x \otimes I_{\mathsf{Bob}} \simeq_0 I_{\mathsf{Alice}} \otimes M_a^x$ for any distribution on x. If we consider the measurement $(M^{x_0} \cdot M^{x_1})_{z_0,z_1}$, then $(M^{x_0} \cdot M^{x_1})_{z_0} = M_{z_0}^{x_0}$. As a result,

$$(M^{x_0} \cdot M^{x_1})_{z_0} \otimes I_{\mathsf{Bob}} \simeq_0 I_{\mathsf{Alice}} \otimes M_{z_0}^{x_0}.$$

Finally, by data processing (Fact 4.26), this implies that

$$(M^{x_0} \cdot M^{x_1})_{[\mathrm{Enc}_{\ell_1}(z_0)|_{T_0} = a_0]} \otimes I_{\mathsf{Bob}} \simeq_0 I_{\mathsf{Alice}} \otimes M^{x_0}_{[\mathrm{Enc}_{\ell_1}(z_0')|_{T_0} = a_0]}.$$

But this is equivalent to saying that $G_{a_0}^{x_0,x_1,T_0,T_1,U} \otimes I_{\mathsf{Bob}} \simeq_0 I_{\mathsf{Alice}} \otimes G_{a_0}^{x_0,T_0}$, which implies passing the cross-check test with probability 1. A similar argument holds for the other tests, with the exception of the verification step.

Consider the measurement $M_{z_0}^{x_0} \cdot M_{z_1}^{x_1}$. By construction and the properties of the PCPP verifier, if this measurement always outputs z_0, z_1 such that $V(\mathsf{input}, x_0, x_1, z_0, z_1) = 1$, then the G strategy always passes the verify step. But because M is a real commuting EPR strategy, $M_z^x \otimes I_{\mathsf{Bob}} \simeq_0 I_{\mathsf{Alice}} \otimes M_z^x$, which implies that

$$M_{z_0}^{x_0} \otimes M_{z_1}^{x_1} \approx_0 M_{z_0}^{x_0} \cdot M_{z_1}^{x_1} \otimes I_{\mathsf{Bob}}.$$

Thus, these two measurements have the same output distribution. But the left-hand side always outputs z_0, z_1 which satisfy the verifier, because this strategy passes the verifier with probability 1. This concludes the completeness step.

Soundness. Suppose input is not in L. Let (ψ, M) be a strategy that passes with probability $1 - \epsilon$.

Code checks. Passing the overall test with probability $1-\epsilon$ means the strategy passes the answer code check with probability $1 - 8\epsilon$. Given values c, x_c, write $1 - \epsilon_{c,x_c}$ for the probability the code check passes conditioned on these values. Then with probability at least $1 - 8\epsilon^{1/2}$, $\epsilon_{c,x_c} \leq \epsilon^{1/2}$. When this occurs, Theorem 16.6 implies that there exists a measurement $\{G_w^{x_c}\}_w$ with outcomes in Sub_{ℓ_1} such that

$$M_a^{x_c,T_c} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G^{x_c}_{[w|_{T_c} = a]}$$

with respect to the distribution of $\boldsymbol{T}_c$ conditioned on c and x_c. When this does *not* occur, we still can assume such a measurement so that

$$M_a^{x_c,T_c} \otimes I_{\mathsf{Bob}} \simeq_1 I_{\mathsf{Alice}} \otimes G^{x_c}_{[w|_{T_c}=a]}$$

trivially, by Fact 4.19. Thus, if we average over $\boldsymbol{c}$ and $\boldsymbol{x_c}$,

$$M_a^{x_c,T_c} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G^{x_c}_{[w|_{T_c}=a]} \tag{81}$$

with respect to the distribution on $\boldsymbol{c}, \boldsymbol{x_c}, \boldsymbol{T_c}$. A similar argument with respect to the consistency guarantee of Theorem 16.6 implies that

$$G_w^{x_c} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G_w^{x_c}. \tag{82}$$

By Fact 4.26, this implies that

$$G^{x_c}_{[w|_{T_c}=a]} \otimes I_{\mathsf{Bob}} \simeq I_{\mathsf{Alice}} \otimes G^{x_c}_{[w|_{T_c}=a]}.$$

As a result, if we apply Fact 4.13 to this and Equation (81) and then use the triangle inequality (Fact 4.28), we conclude

$$M_a^{x_c,I_c} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} G^{x_c}_{[w|_{I_c}=a]} \otimes I_{\mathsf{Bob}} \tag{83}$$

with respect to the distribution on $\boldsymbol{c}, \boldsymbol{x_c}, \boldsymbol{I_c}$.

Applying a similar argument yet again, this time to the proof code check, implies that for every x_0, x_1, there exists a measurement $\{H_w^{x_0,x_1}\}_w$ with outcomes in Sub_{ℓ_2} such that

$$M_a^{x_0,x_1,U} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} H^{x_0,x_1}_{[w|_U=a]} \otimes I_{\mathsf{Bob}} \tag{84}$$

with respect to the distribution on $\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{U}$. Thus, by Fact 4.32, we can assume that Equations (83) and (84) hold with equality with a loss of only $\delta(\epsilon)$ in the game value. In addition, by Theorem 4.1, we can assume that the G and H measurements are all projective, possibly replacing ψ with a different state.

Cross checks. Our next step is to apply the cross-checks. Passing these with probability $1-\delta(\epsilon)$ implies the bounds

$$M_{a_0}^{x_0,x_1,T_0,T_1,U} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes M_{a_0}^{x_0,T_0} = I_{\mathsf{Alice}} \otimes G^{x_0}_{[w|_{T_0}=a_0]}, \tag{85}$$

$$M_{a_1}^{x_0,x_1,T_0,T_1,U} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes M_{a_1}^{x_1,T_1} = I_{\mathsf{Alice}} \otimes G^{x_1}_{[w|_{T_1}=a_1]}, \tag{86}$$

$$M_{a_2}^{x_0,x_1,T_0,T_1,U} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes M_{a_2}^{x_0,x_1,U} = I_{\mathsf{Alice}} \otimes H^{x_0,x_1}_{[w|_U=a_2]},$$

$$M_{a_0,a_1,a_2}^{x_0,x_1,T_0,T_1,U} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes M_{a_0,a_1,a_2}^{x_0,x_1,T_0,T_1,U}. \tag{87}$$

At this point, we would like to apply Fact 4.35. To do so, we have to verify the distance property of our functions, and this will follow from the fact that we augmented our index sets $\boldsymbol{I}_0$, $\boldsymbol{I}_1$, and $\boldsymbol{J}'$ with an additional uniformly random index. To see this, consider two nonequal w and w' in Sub_{ℓ_1}. Then for them to agree on $\boldsymbol{T}_0$, they must agree on $\boldsymbol{i}_0$, and this happens only $\eta(\ell_1)$ fraction of the time. The same holds for $\boldsymbol{U}$, with the bound of $\eta(\ell_2)$. As a result, Fact 4.35 implies the following: consider the POVM measurement $\{\Lambda^{x_0,x_1}_{w_0,w_1,\pi}\}$ with outcomes w_0, w_1 in Sub_{ℓ_1} and π in Sub_{ℓ_2} defined as

$$\Lambda^{x_0,x_1}_{w_0,w_1,\pi} := G^{x_0}_{w_0} \cdot G^{x_1}_{w_1} \cdot H^{x_0,x_1}_{\pi} \cdot G^{x_1}_{w_1} \cdot G^{x_0}_{w_0}. \tag{88}$$

Then

$$M^{x_0,x_1,T_0,T_1,U}_{a_0,a_1,a_2} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes \Lambda^{x_0,x_1}_{[w_0|_{T_0},w_1|_{T_1},\pi|_U=a_0,a_1,a_2]}. \tag{89}$$

From this, Equation (87) implies

$$M^{x_0,x_1,T_0,T_1,U}_{a_0,a_1,a_2} \otimes I_{\mathsf{Bob}} \approx_{\delta(\epsilon)} \Lambda^{x_0,x_1}_{[w_0|_{T_0},w_1|_{T_1},\pi|_U=a_0,a_1,a_2]} \otimes I_{\mathsf{Bob}}. \tag{90}$$

Thus, by Fact 4.32, we can assume that Equation (90) holds with equality by replacing M with G, incurring a loss of only $\delta(\epsilon)$ in the game value. (Unlike before, here we do *not* invoke Theorem 4.1 on J^{x_0,x_1} to make it a projective measurement, as that would likely change the structure in Equation (88), which we will need later.)

Verification. The strategy passes the verify check with probability $1-\delta(\epsilon)$. By Equation (90) (which we now assume is equality), this is the same probability as if we (i) sample $\boldsymbol{x}_0, \boldsymbol{x}_1$, (ii) use Λ to draw $\boldsymbol{w}_0, \boldsymbol{w}_1, \boldsymbol{\pi}$, (iii) draw $\boldsymbol{I}_0, \boldsymbol{I}_1, \boldsymbol{J}$ conditioned on $\boldsymbol{x}_0, \boldsymbol{x}_1$, (iv) then draw $\boldsymbol{T}_0, \boldsymbol{T}_1, \boldsymbol{U}$ conditioned on $\boldsymbol{I}_0, \boldsymbol{I}_1, \boldsymbol{J}$, (v) compute $\boldsymbol{a}_0 = \boldsymbol{w}_0|_{\boldsymbol{T}_0}$, $\boldsymbol{a}_1 = \boldsymbol{w}_1|_{\boldsymbol{T}_1}$, and $\boldsymbol{a}_2 = \boldsymbol{w}_2|_{\boldsymbol{U}}$, and (vi) give $\boldsymbol{a}_0|_{\boldsymbol{I}_0}, \boldsymbol{a}_1|_{\boldsymbol{I}_1}, \boldsymbol{a}_2|_{\boldsymbol{J}}$ to V_{PCPP} and accept if it accepts.

Condition on a fixed choice of x_0, x_1 and a draw for w_0, w_1, π. The PCPP verifier receives answers to its $\boldsymbol{I}_0$ and $\boldsymbol{I}_1$ queries based on w_0 and w_1, which are in Sub_{ℓ_1}. In addition, although π is in Sub_{ℓ_2} and may not correspond to the encoding of an actual proof string, the verifier only queries it at points in the image of the embedding μ_{ℓ_2}. As a result, the answers V_{PCPP} receives to its J queries are consistent with *some* fixed proof string. Thus, by Proposition 17.8, since $1-\eta(k) \geq 2\gamma$ for all k, if the probability the verifier accepts is greater than s, then there are strings $y_0, y_1 \in \{0,1\}^{\ell_1}$ such that $w_0 = \mathrm{Enc}_{\ell_1}(y_0)$, $w_1 = \mathrm{Enc}_{\ell_1}(y_1)$ and $V(\mathsf{input}, x_0, x_1, y_0, y_1) = 1$. Averaging over all $\boldsymbol{x}_0, \boldsymbol{x}_1$ and $\boldsymbol{w}_0, \boldsymbol{w}_1, \boldsymbol{\pi}$, we conclude that

$$\mathbf{Pr}[V(\mathsf{input}, \boldsymbol{x}_0, \boldsymbol{x}_1, \mathrm{Dec}_{\ell_1}(\boldsymbol{w}_0), \mathrm{Dec}_{\ell_1}(\boldsymbol{w}_1)) = 1] \geq \frac{1-\delta(\epsilon)-s}{1-s} = 1-\delta(\epsilon). \tag{91}$$

Recall that the decoding map is one-to-one except on those strings not in the range of the encoding map, which it maps to $\perp$ instead. As we can assume that the verifier V always rejects when it receives $\perp$ for an answer, this tells us that $\mathrm{Dec}_{\ell_1}(\boldsymbol{w}_0), \mathrm{Dec}_{\ell_1}(\boldsymbol{w}_1) \neq \perp$ with probability at least $1-\delta(\epsilon)$.

Wrapping it up. Now we give a strategy for causing the verifier V to accept with high probability on input. It uses state ψ, and given question x it applies the measurement $\{A^x_a\}_a$ defined as

$$A^x_a := G^x_{[\mathrm{Dec}_{\ell_1}(w)=a]}.$$

Consider the verifier V' which samples $(\boldsymbol{x}_0, \boldsymbol{x}_1)$, gives them to Alice and Bob, receives $\boldsymbol{w}_0, \boldsymbol{w}_1$, and accepts if $V(\mathsf{input}, \boldsymbol{x}_0, \boldsymbol{x}_1, \mathrm{Dec}_{\ell_1}(\boldsymbol{w}_0), \mathrm{Dec}_{\ell_1}(\boldsymbol{w}_1)) = 1$. Then V accepts on strategy A with the same probability that V' accepts on strategy G. In other words, if we define $S(x_0, x_1)$ to be the set of (w_0, w_1) such that

$$V(\mathsf{input}, x_0, x_1, \mathrm{Dec}_{\ell_1}(w_0), \mathrm{Dec}_{\ell_1}(w_1)) = 1,$$

then the probability that V' accepts on strategy G is

$$\mathop{\mathbf{E}}_{(\boldsymbol{x}_0,\boldsymbol{x}_1)} \sum_{w_0,w_1 \in S(\boldsymbol{x}_0,\boldsymbol{x}_1)} \langle\psi| G^{x_0}_{w_0} \otimes G^{x_1}_{w_1} |\psi\rangle. \tag{92}$$

To show this is large, we begin by showing that the G's commute with each other. To see this, note that Equations (85) and (86) implies that for a fixed $c \in \{0,1\}$,

$$\Lambda^{x_0,x_1}_{[w_c|_{T_c}=a_c]} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G^{x_c}_{[w'_c|_{T_c}=a_c]}.$$

However, by the distance properties of our code and the fact that $\boldsymbol{T}_c$ contains a uniformly random index, this implies that

$$\Lambda^{x_0,x_1}_{w_c} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G^{x_c}_{w_c}. \tag{93}$$

As a result,

$$\begin{aligned} G^{x_0}_{w_0} \cdot G^{x_1}_{w_1} \otimes I_{\mathsf{Bob}} &\approx_{\delta(\epsilon)} G^{x_0}_{w_0} \otimes \Lambda^{x_0,x_1}_{w_1} \\ &\approx_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes \Lambda^{x_0,x_1}_{w_1} \cdot \Lambda^{x_0,x_1}_{w_0} \\ &\approx_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes \Lambda^{x_0,x_1}_{w_0} \cdot \Lambda^{x_0,x_1}_{w_1} \\ &\approx_{\delta(\epsilon)} G^{x_1}_{w_1} \otimes \Lambda^{x_0,x_1}_{w_0} \\ &\approx_{\delta(\epsilon)} G^{x_1}_{w_1} \cdot G^{x_0}_{w_0} \otimes I_{\mathsf{Bob}}. \end{aligned}$$

A similar argument as the one establishing (93) implies that

$$G^{x_c}_{w_c} \otimes I_{\mathsf{Bob}} \simeq_{\delta(\epsilon)} I_{\mathsf{Alice}} \otimes G^{x_c}_{w_c}.$$

Thus,

$$\begin{aligned} G^{x_0}_{w_0} \otimes G^{x_1}_{w_1} &= G^{x_0}_{w_0} \cdot G^{x_0}_{w_0} \otimes G^{x_1}_{w_1} \\ &\approx_{\delta(\epsilon)} G^{x_0}_{w_0} \cdot G^{x_0}_{w_0} \cdot G^{x_1}_{w_1} \otimes I_{\mathsf{Bob}} \\ &\approx_{\delta(\epsilon)} G^{x_0}_{w_0} \cdot G^{x_1}_{w_1} \cdot G^{x_0}_{w_0} \otimes I_{\mathsf{Bob}} \\ &= \Lambda^{x_0,x_1}_{w_0,w_1} \otimes I_{\mathsf{Bob}}. \end{aligned}$$

As a result, by Fact 4.31, Equation (92) is at least $1 - \delta(\epsilon)$ by Equation (91). This concludes the proof of the theorem. □

17.5 Applying the answer reduction protocol

In this section, we instantiate Theorem 17.10 with the low-degree code and then apply it to our NEEXP protocol.

Theorem 17.11. *Let $V = (\mathrm{Alg}_{\mathrm{Q}}, \mathrm{Alg}_{\mathrm{A}})$ be an MIP* verifier for a language L. Write L_{A} for the language decided by $\mathrm{Alg}_{\mathrm{A}}$. Suppose on inputs of size n, the verifier V has question length $\ell_{V,\mathrm{Q}}(n)$, answer length $\ell_{V,\mathrm{A}}(n)$, question time $t_{V,\mathrm{Q}}(n)$, and answer time $t_{V,\mathrm{A}}(n)$. Then there exists another MIP* verifier V_{ans} for L with the following parameters.*

$$\begin{aligned} \mathsf{Q\text{-}length}(V_{\mathrm{ans}}) &= O(\ell_{V,\mathrm{Q}}(n) + \log(\ell_{V,\mathrm{A}}(n)) + \log(t_{V,\mathrm{A}}(n))), \\ \mathsf{A\text{-}length}(V_{\mathrm{ans}}) &= O(\mathrm{polylog}(\ell_{V,\mathrm{A}}(n)) + \mathrm{polylog}(t_{V,\mathrm{A}}(n))), \\ \mathsf{Q\text{-}time}(V_{\mathrm{ans}}) &= O\left(t_{V,\mathrm{Q}}(n)\right) + \mathrm{poly}(n + \ell_{V,\mathrm{Q}}(n), \log(\ell_{V,\mathrm{A}}(n)), \log(t_{V,\mathrm{A}}(n))), \\ \mathsf{A\text{-}time}(V_{\mathrm{ans}}) &= \mathrm{poly}(n + \ell_{V,\mathrm{Q}}(n), \log(\ell_{V,\mathrm{A}}(n)), \log(t_{V,\mathrm{A}}(n))). \end{aligned}$$

Proof. We instantiate the low-degree code in Fact 16.9. It gives an error correcting code with parameters $(n, \mathrm{poly}(n), \mathrm{polylog}(n), \mathrm{polylog}(n)^{-1}, \mathrm{poly}(n), \mathrm{polylog}(n))$ and a c-subset test $\mathscr{G}_k$ with robustness $\chi_k(\epsilon) = \mathrm{poly}(\epsilon, \log(k)^{-1})$ such that

$$\mathsf{Q\text{-}time}(\mathscr{G}_k) = \mathrm{poly}(\log k, c), \quad \mathsf{A\text{-}time}(\mathscr{G}_k) = \mathrm{poly}(\log(k)^c),$$

$$\mathsf{Q\text{-}length}(\mathscr{G}_k) = O(c \log k), \quad \mathsf{A\text{-}length}(\mathscr{G}_k) = O(\log(k)^{2c}).$$

We then apply Theorem 17.10 with $s, \gamma = \frac{1}{10}$. At this point, the theorem follows immediately, but as deriving it can be cumbersome, we fill in the details.

By construction, $t_{\mathrm{Dec}}(n) = \mathrm{poly}(n)$. As a result,

$$t_{\mathrm{compose}}(n) = t_{V,\mathrm{A}}(n) + t_{\mathrm{Dec}}(\ell_{V,\mathrm{A}}(n)) = t_{V,\mathrm{A}}(n) + \mathrm{poly}(\ell_{V,\mathrm{A}}(n)).$$

Thus,

$$\ell_\pi(n) = t_{\mathrm{compose}}(n) \cdot \mathrm{polylog}(t_{\mathrm{compose}}(n)) = \mathrm{poly}(t_{V,\mathrm{A}}(n), \ell_{V,\mathrm{A}}(n)).$$

Now, $m(n) = \mathrm{poly}(n)$. Thus,

$$\begin{aligned} t_{\mathrm{PCPP}}(n) &= \mathrm{poly}(n + \ell_{V,\mathrm{Q}}(n), \log(m(\ell_{V,\mathrm{A}}(n))), \log(t_{\mathrm{compose}}(n))) \\ &= \mathrm{poly}(n + \ell_{V,\mathrm{Q}}(n), \log(\ell_{V,\mathrm{A}}(n)), \log(t_{V,\mathrm{A}}(n))). \end{aligned}$$

Furthermore, $q(n) = \mathrm{polylog}(n)$ and $t_{\mathrm{Emb}}(n) = \mathrm{polylog}(n)$. As a result,

$$\begin{aligned} \log(m(\ell_\pi(n))) &= O(\log(\ell_{V,\mathrm{A}}(n)) + \log(t_{V,\mathrm{A}}(n))), \\ \log(q(\ell_\pi(n))) &= O(\log\log(\ell_{V,\mathrm{A}}(n)) + \log\log(t_{V,\mathrm{A}}(n))), \\ t_{\mathrm{Emb}}(\ell_\pi(n)) &= \mathrm{poly}(\log(\ell_{V,\mathrm{A}}(n)), \log(t_{V,\mathrm{A}}(n))). \end{aligned}$$

The theorem now follows from applying these bounds to Theorem 17.10. □

Crucially, although polynomial factors of $t_{V,Q}(n)$ and $\ell_{V,Q}(n)$ appear in Theorem 17.11, only the *logarithms* of $t_{V,A}(n)$ and $\ell_{V,A}(n)$ appear in this theorem. As a result, if we apply this to Corollary 15.9, we arrive at our main result.

Theorem 17.12. *There is an* MIP^* *verifier* $\mathscr{G}$ *for* Succinct-Succinct-3Sat *with parameters*

$$\mathsf{Q\text{-}length}(\mathscr{G}) = O(n), \quad \mathsf{A\text{-}length}(\mathscr{G}) = \mathrm{poly}(n),$$

$$\mathsf{Q\text{-}time}(\mathscr{G}) = \mathrm{poly}(n), \quad \mathsf{A\text{-}time}(\mathscr{G}) = \mathrm{poly}(n).$$

References

[ALM+98] Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. Proof verification and the hardness of approximation problems. *Journal of the ACM*, 45(3):501–555, 1998. 1, 17.2

[ARW17] Amir Abboud, Aviad Rubinstein, and Ryan Williams. Distributed PCP theorems for hardness of approximation in P. In *Proceedings of the 58th Annual IEEE Symposium on Foundations of Computer Science*, 2017. 2.5, 17.1

[AS98] Sanjeev Arora and Shmuel Safra. Probabilistic checking of proofs: a new characterization of NP. *Journal of the ACM*, 45(1):70–122, 1998. 1, 17.2

[Bel64] John Bell. On the Einstein Podolsky Rosen paradox. *Physics*, 1(3):195–200, 1964. 1

[BFL91] László Babai, Lance Fortnow, and Carsten Lund. Non-deterministic exponential time has two-prover interactive protocols. *Computational complexity*, 1(1):3–40, 1991. 1, 2.2, 11

[BFLS91] László Babai, Lance Fortnow, Leonid A Levin, and Mario Szegedy. Checking computations in polylogarithmic time. In *Proceedings of the 23rd Annual ACM Symposium on Theory of Computing*, pages 21–32, 1991. 17.2, 17.3

[BOGKW88] Michael Ben-Or, Shafi Goldwasser, Joe Kilian, and Avi Wigderson. Multi-prover interactive proofs: How to remove intractability assumptions. In *Proceedings of the 20th Annual ACM Symposium on Theory of Computing*, pages 113–131, 1988. 1

[BSGH+05] Eli Ben-Sasson, Oded Goldreich, Prahladh Harsha, Madhu Sudan, and Salil Vadhan. Short PCPs verifiable in polylogarithmic time. In *Proceedings of the 20th Annual IEEE Conference on Computational Complexity*, pages 120–134, 2005. 17.2, 17.3, 17.4

[BSGH+06] Eli Ben-Sasson, Oded Goldreich, Prahladh Harsha, Madhu Sudan, and Salil Vadhan. Robust PCPs of proximity, shorter PCPs, and applications to coding. *SIAM Journal on Computing*, 36(4):889–974, 2006. 17.2, 17.3

[CGJV18] Andrea Coladangelo, Alex Grilo, Stacey Jeffery, and Thomas Vidick. Verifier-on-a-leash: new schemes for verifiable delegated quantum computation, with quasilinear resources. In *21st Conference on Quantum Information Processing*, 2018. 8.2

[CHTW04] Richard Cleve, Peter Hoyer, Benjamin Toner, and John Watrous. Consequences and limits of nonlocal strategies. In *Proceedings of the 19th Annual IEEE Conference on Computational Complexity*, pages 236–249, 2004. 1

[Col06] Roger Colbeck. *Quantum and relativistic protocols for secure multi-party computation*. PhD thesis, University of Cambridge, 2006. 1

[CY14] Matthew Coudron and Henry Yuen. Infinite randomness expansion with a constant number of devices. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, pages 427–436, 2014. 1

[DR06] Irit Dinur and Omer Reingold. Assignment testers: Towards a combinatorial proof of the PCP theorem. *SIAM Journal on Computing*, 36(4):975–1024, 2006. 17.2

[Eke91] Artur Ekert. Quantum cryptography based on Bell's theorem. *Physical review letters*, 67(6):661, 1991. 1

[EPR35] Albert Einstein, Boris Podolsky, and Nathan Rosen. Can quantum-mechanical description of physical reality be considered complete? *Physical review*, 47(10):777, 1935. 1

[FJVY19] Joseph Fitzsimons, Zhengfeng Ji, Thomas Vidick, and Henry Yuen. Quantum proof systems for iterated exponential time, and beyond. In *Proceedings of the 51st Annual ACM Symposium on Theory of Computing*, 2019. 1, 1, 1

[FL18] Bill Fefferman and Cedric Yen-Yu Lin. A complete characterization of unitary quantum space. In *Proceedings of the 9th Innovations in Theoretical Computer Science*, pages 4:1–4:21, 2018. 1

[FV15] Joseph Fitzsimons and Thomas Vidick. A multiprover interactive proof system for the local Hamiltonian problem. In *Proceedings of the 6th Innovations in Theoretical Computer Science*, pages 103–112, 2015. 1

[Har10] Prahladh Harsha. Lecture 9 from Limits of Approximation Algorithms: PCPs and Unique Games. Found at http://www.tcs.tifr.res.in/~prahladh/teaching/2009-10/limits/lectures/lec09.pdf, 2010. 11

[Hås97] Johan Håstad. Some optimal inapproximability results. In *Proceedings of the 29th Annual ACM Symposium on Theory of Computing*, pages 1–10, 1997. 1

[IKW12] Tsuyoshi Ito, Hirotada Kobayashi, and John Watrous. Quantum interactive proofs with weak error bounds. In *Proceedings of the 3rd Innovations in Theoretical Computer Science*, pages 266–275, 2012. 1

[IV12] Tsuyoshi Ito and Thomas Vidick. A multi-prover interactive proof for NEXP sound against entangled provers. In *Proceedings of the 53rd Annual IEEE Symposium on Foundations of Computer Science*, pages 243–252, 2012. 1

[Ji17] Zhengfeng Ji. Compression of quantum multi-prover interactive proofs. In *Proceedings of the 49th Annual ACM Symposium on Theory of Computing*, pages 289–302, 2017. 1, 1

[KRR14] Yael Kalai, Ran Raz, and Ron Rothblum. How to delegate computations: the power of no-signaling proofs. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, pages 485–494, 2014. 1

[MBG+13] Alfred J Menezes, Ian F Blake, XuHong Gao, Ronald C Mullin, Scott A Vanstone, and Tomik Yaghoobian. *Applications of finite fields*. Springer Science & Business Media, 2013. 3.1, 3.1

[Mei14] Or Meir. Combinatorial PCPs with efficient verifiers. *Computational Complexity*, 23(3):355–478, 2014. 17.2

[Mie09] Thilo Mie. Short PCPPs verifiable in polylogarithmic time with $o(1)$ queries. *Annals of Mathematics and Artificial Intelligence*, 56(3-4):313–338, 2009. 17.2, 17.5

[MR08] Dana Moshkovitz and Ran Raz. Sub-constant error low degree test of almost-linear size. *SIAM Journal on Computing*, 38(1):140–180, 2008. 3.5

[MY98] Dominic Mayers and Andrew Yao. Quantum cryptography with imperfect apparatus. In *Proceedings of the 39th Annual IEEE Symposium on Foundations of Computer Science*, pages 503–509, 1998. 1

[NV18a] Anand Natarajan and Thomas Vidick. Low-degree testing for quantum states, and a quantum entangled games PCP. In *Proceedings of the 59th Annual IEEE Symposium on Foundations of Computer Science*, 2018. 1, 1, 2.3, 3.6, 4.41, 4.9, 4.9, 6, 6.1, 6.1, 6.4, 6.1, 6.5, 6.6, 17.2

[NV18b] Anand Natarajan and Thomas Vidick. Two-player entangled games are NP-hard. In *Proceedings of the 33rd Annual IEEE Conference on Computational Complexity*, 2018. 1, 1, 1, 4.7, 4.40, 4.41

[O'D05] Ryan O'Donnell. A history of the PCP theorem, 2005. 17.2

[Pap94] Christos Papadimitriou. *Computational complexity*. Addison Wesley, 1994. 3.7, 3.7, 3.7, 3.7

[Per12] Attila Pereszlényi. Multi-prover quantum Merlin-Arthur proof systems with small gap. Technical report, arXiv:1205.2761, 2012. 1

[RS97] Ran Raz and Shmuel Safra. A sub-constant error-probability low-degree test, and a sub-constant error-probability PCP characterization of NP. In *Proceedings of the 29th Annual ACM Symposium on Theory of Computing*, pages 475–484, 1997. 3.5, 3.12

[RUV13] Ben Reichardt, Falk Unger, and Umesh Vazirani. A classical leash for a quantum system: Command of quantum systems via rigidity of CHSH games. *Nature*, 496:456–460, 2013. 1

[Sch80] Jacob Schwartz. Fast probabilistic algorithms for verification of polynomial identities. *Journal of the ACM*, 27(4):701–717, 1980. 3.6

[Slo16] William Slofstra. Tsirelson's problem and an embedding theorem for groups arising from non-local games. Technical report, arXiv:1606.03140, 2016. 1, 1

[Slo19] William Slofstra. The set of quantum correlations is not closed. In *Forum of Mathematics, Pi*, volume 7, page e1, 2019. 1, 1

[SW88] Stephen Summers and Reinhard Werner. Maximal violation of Bell's inequalities for algebras of observables in tangent spacetime regions. *Annales de l'IHP Physique théorique*, 49(2):215–243, 1988. 1

[Tsi80] Boris Tsirelson. Quantum generalizations of Bell's inequality. *Letters in Mathematical Physics*, 4(2):93–100, 1980. 1

[Vid11] Thomas Vidick. *The complexity of entangled games.* PhD thesis, University of California, Berkeley, 2011. 4.4

[Vid16] Thomas Vidick. Three-player entangled XOR games are NP-hard to approximate. *SIAM Journal on Computing*, 45(3):1007–1063, 2016. 1, 4.7

[WBMS16] Xingyao Wu, Jean-Daniel Bancal, Matthew McKague, and Valerio Scarani. Device-independent parallel self-testing of two singlets. *Physical Review A*, 93(6):062121, 2016. 6.1

[Zip79] Richard Zippel. Probabilistic algorithms for sparse polynomials. In *Proceedings of the 2nd International Symposium on Symbolic and Algebraic Manipulation*, pages 216–226, 1979. 3.6