Spatial Isolation Implies Zero Knowledge Even in a Quantum World[†]

Alessandro Chiesa
alexch@berkeley.edu
UC Berkeley

Michael A. Forbes
miforbes@illinois.edu
University of Illinois at Urbana–Champaign

Tom Gur
tom.gur@warwick.ac.uk
University of Warwick

Nicholas Spooner
nick.spooner@berkeley.edu
UC Berkeley

Abstract

Zero knowledge plays a central role in cryptography and complexity. The seminal work of Ben-Or et al. (STOC 1988) shows that zero knowledge can be achieved unconditionally for any language in $\mathbf{NEXP}$, as long as one is willing to make a suitable *physical assumption*: if the provers are spatially isolated, then they can be assumed to be playing independent strategies.

Quantum mechanics, however, tells us that this assumption is unrealistic, because spatially-isolated provers could share a quantum entangled state and realize a non-local correlated strategy. The MIP* model captures this setting.

In this work we study the following question: *does spatial isolation still suffice to unconditionally achieve zero knowledge even in the presence of quantum entanglement?*

We answer this question in the affirmative: we prove that every language in $\mathbf{NEXP}$ has a 2-prover *zero knowledge* interactive proof that is sound against entangled provers; that is, $\mathbf{NEXP} \subseteq \mathbf{ZK\text{-}MIP}^*$.

Our proof consists of constructing a zero knowledge interactive PCP with a strong algebraic structure, and then lifting it to the MIP* model. This lifting relies on a new framework that builds on recent advances in low-degree testing against entangled strategies, and clearly separates classical and quantum tools.

Our main technical contribution is the development of new algebraic techniques for obtaining unconditional zero knowledge; this includes a zero knowledge variant of the celebrated sumcheck protocol, a key building block in many probabilistic proof systems. A core component of our sumcheck protocol is a new algebraic commitment scheme, whose analysis relies on algebraic complexity theory.

Keywords: zero knowledge; multi-prover interactive proofs; quantum entangled strategies; interactive PCPs; sumcheck protocol; algebraic complexity

Extended abstract of this work appeared in the 59th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2018).

Contents

1 Introduction

Zero knowledge, the ability to demonstrate the validity of a claim without revealing any information about it, is a central notion in cryptography and complexity that has received much attention in the last few decades. Introduced in the seminal work of Goldwasser, Micali, and Rackoff [GMR89], zero knowledge was first demonstrated in the model of interactive proofs, in which a resource-unbounded prover interacts with a probabilistic polynomial-time verifier to the end of convincing it of the validity of a statement.

Goldreich, Micali, and Wigderson [GMW91] showed that every language in $\mathbf{NP}$ has a *computational* zero knowledge interactive proof, under the cryptographic assumption that (non-uniform) one-way functions exist. Ostrovsky and Wigderson [OW93] proved that this assumption is necessary.

Unfortunately, the stronger notion of *statistical* zero knowledge interactive proofs, where both soundness and zero knowledge hold unconditionally, is limited. For example, if $\mathbf{NP}$ had such proofs then the polynomial hierarchy would collapse to its second level [BHZ87; For87; AH91].

The celebrated work of Ben-Or et al. [BGKW88] demonstrated that the situation is markedly different when the verifier interacts with *multiple* provers, in a *classical world* where by spatially isolating the provers we ensure that they are playing independent strategies — this is the model of multi-prover interactive proofs (MIPs). They proved that every language having an MIP (i.e., every language in $\mathbf{NEXP}$ [BFL91]) also has a *perfect* zero knowledge MIP. This result tells us that *spatial isolation implies zero knowledge*.

In light of quantum mechanics, however, we know that spatial isolation *does not* imply independence, because the provers could share an entangled state and realize a strategy that is beyond that of independently acting provers. For example, it is possible for entangled provers to win a game (e.g., the magic square game) with probability 1, whereas independent provers can only win with probability at most $8/9$ [CHTW04].

Non-local correlations arising from local measurements on entangled particles play a fundamental role in physics, and their study goes back at least to Bell's work on the Einstein–Podolsky–Rosen paradox [Bel64]. Recent years have seen a surge of interest in MIPs with *entangled provers*, which correspond to the setting in which multiple non-communicating provers share an entangled state and wish to convince a classical verifier of some statement. This notion is captured by MIP* protocols, introduced by Cleve et al. [CHTW04]. A priori it is unclear whether these systems should be less powerful than standard MIPs, because of the richer class of *malicious* prover strategies, or more powerful, because of the richer class of *honest* prover strategies.

Investigating proof systems with entangled adversaries not only sharpens our understanding of entanglement as a computational resource, but also contributes insights to hardness of approximation and cryptography in a post-quantum world. However, while the last three decades saw the development of powerful ideas and tools for designing and analyzing proof systems with classical adversaries, despite much effort, there are only a handful of tools available for dealing with quantum entangled adversaries, and many fundamental questions remain open.

MIP* protocols were studied in a long line of work, culminating in a breakthrough result of Ito and Vidick [IV12], who in a technical tour-de-force showed that $\mathbf{NEXP} \subseteq \mathbf{MIP}^*$;[1] this result was further improved in [Vid16; NV18]. However, it is unknown whether these MIP protocols can achieve zero knowledge, which is the original motivation behind the classical MIP model. In sum, in this paper we pose the following question:

To what extent does spatial isolation imply unconditional zero knowledge in a quantum world?

[1] While this is the popular statement of the result, [IV12] show a stronger result, namely, that $\mathbf{NEXP}$ is exactly the class of languages decided by MIPs *sound against entangled provers*. Their honest provers are classical, and soundness holds also against entangled provers. This is also the case in our protocols. It remains unknown whether entanglement grants provers additional power: there is *no* known reasonable upper bound on $\mathbf{MIP}^*$.

1.1 Our results

Our main result is a strong positive answer to the foregoing question, namely, we show that the $\mathbf{NEXP} \subseteq \mathbf{MIP}^*$ result of Ito and Vidick [IV12] continues to hold even when we require zero knowledge.

Theorem 1. Every language in $\mathbf{NEXP}$ has a perfect zero knowledge 2-prover MIP*. In more detail,

$$\mathbf{NEXP} \subseteq \mathbf{PZK\text{-}MIP}^* \left[\begin{array}{rl} \text{number of provers:} & 2 \\ \text{round complexity:} & \mathrm{poly}(n) \\ \text{communication complexity:} & \mathrm{poly}(n) \\ \text{soundness error:} & 1/2 \end{array} \right].$$

We stress that the MIP* protocols of Theorem 1 enjoy both unconditional soundness against entangled provers as well as unconditional (perfect) zero knowledge against *any* (possibly malicious) verifier.

1.2 Other notions of quantum zero knowledge

To the best of our knowledge, this work is the first to study the notion of zero knowledge with entangled provers, as captured by the MIP* model. Nevertheless, zero knowledge has been studied in other settings in the quantum information and computation literature; we now briefly recall these.

Watrous [Wat02] introduced *honest-verifier* zero knowledge for quantum interactive proofs (interactive proofs in which the prover and verifier are quantum machines), and studied the resulting complexity class $\mathbf{QSZK_{HV}}$. Kobayashi [Kob03] studied a non-interactive variant of this notion. Damgård, Fehr, and Salvail [DFS04] achieve zero knowledge for $\mathbf{NP}$ against malicious quantum verifiers, but only via *arguments* (i.e., computationally sound proofs) in the common reference string model. Subsequently, Watrous [Wat09] constructed quantum interactive proofs that remain zero knowledge against malicious quantum verifiers.

Zero knowledge for quantum interactive proofs has since then remained an active area of research, and several aspects and variants of it were studied in recent works, including the power of public-coin interaction [Kob08], quantum proofs of knowledge [Unr12], zero knowledge in the quantum random oracle model [Unr15], zero knowledge proof systems for $\mathbf{QMA}$ [BJSW16], and oracle separations for quantum statistical zero knowledge [MW18].

All the above works consider protocols between a *single* quantum prover and a quantum verifier. In particular, they do not study entanglement as a shared resource between two (or more) provers.

In contrast, the MIP* protocols that we study differ from the protocols above in two main aspects: (1) our proof systems have multiple spatially-isolated provers that share an entangled state, and (2) it suffices that the honest verifier is a *classical* machine. Indeed, we show that, analogously to the classical setting, MIP* protocols can achieve *unconditional* zero knowledge for a much larger complexity class (namely, $\mathbf{NEXP}$) than possible for QSZK protocols (since $\mathbf{QSZK} \subseteq \mathbf{QIP} = \mathbf{PSPACE}$).

2 Techniques

We begin by discussing the challenge that arises when trying to prove that $\mathbf{NEXP} \subseteq \mathbf{PZK\text{-}MIP}^*$, by outlining a natural approach to obtaining zero knowledge MIP* protocols, and considering why it fails.

2.1 The challenge

We know that every language in $\mathbf{NEXP}$ has a (perfect) zero knowledge MIP protocol, namely, that $\mathbf{NEXP} \subseteq \mathbf{PZK\text{-}MIP}$ [BGKW88]. We also know that every language in $\mathbf{NEXP}$ has an MIP* protocol, namely, that $\mathbf{NEXP} \subseteq \mathbf{MIP}^*$ [IV12]. Is it then not possible to simply combine these two facts and deduce that every language in $\mathbf{NEXP}$ has a (perfect) zero knowledge MIP*?

The challenge is that the standard techniques used to construct zero knowledge MIP protocols do not seem compatible with those used to construct MIP* protocols for large classes. In fact, the former are precisely the type of techniques that prove to be very limited for obtaining soundness against entangled provers.

In more detail, while constructions of MIP (and PCP) protocols typically capitalize on an *algebraic* structure, known constructions of *zero knowledge* MIPs are of a *combinatorial* nature. For example, the zero knowledge MIP in [BGKW88] is based on a multi-prover information-theoretic commitment scheme, which can be thought of as a CHSH-like game. The zero knowledge MIP in [DFKNS92] is obtained via the standard transformation from zero knowledge PCPs, which is a form of consistency game. Unfortunately, these types of constructions do not appear resistant to entangled provers, nor is it clear how one can modify them to obtain this resistance without leveraging some algebraic structure.

Indeed, initial attempts to show that $\mathbf{NEXP} \subseteq \mathbf{MIP}^*$ (e.g., [IKPSY08; IKM09; KKMTV11]) tried to apply some black box transformation to an arbitrarily structured (classical) MIP protocol to force the provers to behave as if they are not entangled, and then appeal to standard MIP soundness. These works were only able to obtain limited protocols (e.g., with very large soundness error).

In their breakthrough paper, Ito and Vidick [IV12] overcame this hurdle and showed that $\mathbf{NEXP} \subseteq \mathbf{MIP}^*$ by taking a different route: rather than a black box transformation, they modified and reanalyzed a particular proof system, namely the MIP protocol for $\mathbf{NEXP}$ in [BFL91], while leveraging and crucially using its algebraic structure. (Subsequent works [Vid16; NV18] improved this result by reducing the number of provers and rounds to a minimum, showing MIP* protocols for $\mathbf{NEXP}$ with two provers and one round.)

In sum, the challenge lies in the apparent incompatibility between techniques used for zero knowledge and those used for soundness against entangled provers.

2.2 High-level overview

Our strategy for proving our main result is to bridge the aforementioned gap by isolating the role of algebra in granting soundness against entangled provers, and developing new algebraic techniques for zero knowledge. Our proof of Theorem 1 thus consists of two parts.

(I) **Lifting lemma:** a black box transformation from algebraically-structured classical protocols into corresponding MIP* protocols, which preserves zero knowledge.

(II) **Algebraic zero knowledge:** a new construction of zero knowledge algebraically-structured protocols for any language in $\mathbf{NEXP}$.

The first part is primarily a conceptual contribution, and it deals with quantum aspects of proof systems. The second part is our main technical contribution, and it deals with classical protocols (it does not require any

background in quantum information). We briefly discuss each of the parts, and then provide an overview of the first part in Section 2.3 and of the second part in Section 2.4.

In the first part of the proof, we build on recent advances in low-degree testing against entangled provers, and provide an abstraction of techniques in [IV12; Vid16; NV18]. We prove a *lifting lemma* (Lemma 8.1) that transforms a class of algebraically-structured classical protocols into MIP* protocols, while preserving zero knowledge. This provides a generic framework for constructing MIP* protocols, while decoupling the mechanisms responsible for soundness against entangled provers from other classical components.

In the second part of the proof, we construct an algebraically-structured zero knowledge classical protocol, which we refer to as a *low-degree interactive PCP*, to which we apply the lifting lemma, completing the proof. At the heart of our techniques is a strong zero knowledge variant of the sumcheck protocol [LFKN92] (a fundamental subroutine in many probabilistic proof systems), which we deem of independent interest. In turn, a key component in our zero knowledge sumcheck is a new algebraic commitment scheme, whose hiding property is guaranteed by algebraic query complexity lower bounds [AW09; JKRS09]. These shed more light on the connection of zero knowledge to algebraic complexity theory.

We summarize the roadmap towards proving Theorem 1 in Section 4.

2.3 Part I: lifting classical proof systems to MIP*

The first step towards obtaining a generic framework for transforming classical protocols into corresponding MIP* protocols is making a simple, yet crucial, observation. Namely, while the result in [IV12] is stated as a white box modification of the MIP protocol in [BFL91], we observe that the techniques used there can in fact be applied more generally. That is, we observe that *any* "low-degree interactive PCP", a type of algebraically structured proof system that underlies (implicitly and explicitly) many constructions in the probabilistic proof systems literature, can be transformed into a corresponding MIP* protocol.

The first part of the proof of Theorem 1 formalizes this idea, identifying sufficient conditions to apply the techniques of [IV12; Vid16], and showing a lifting lemma that transforms protocols satisfying these conditions into MIP* protocols. We relate features of the original protocol to those of the resulting MIP* protocols, such as round complexity and, crucially, zero knowledge.

To make this discussion more accurate, we next define and discuss low-degree interactive PCPs.

2.3.1 Low-degree interactive PCPs

An *Interactive PCP* (IPCP), a proof system whose systematic study was initiated by Kalai and Raz [KR08], naturally extends the notions of a probabilistically checkable proof (PCP) and an interactive proof (IP). An r-round IPCP is a two-phase protocol in which a computationally unbounded *prover* P tries to convince a polynomial-time *verifier* V that an input x, given to both parties, is in a language $\mathscr{L}$. First, the prover sends to the verifier a PCP oracle (a purported proof that $x \in \mathscr{L}$), which the verifier can query at any time. Second, the prover and verifier engage in an r-round IP, at the end of which the verifier either accepts or rejects.[2] Completeness and soundness are defined in the usual way.

In this work we consider a type of algebraically-structured IPCP, which we call a *low-degree IPCPs*. This notion implicitly (and semi-explicitly) underlies many probabilistic proof systems in the literature. Informally, a low-degree IPCP is an IPCP satisfying the following: (1) *low-degree completeness*, which states that the PCP oracle sent by the (honest) prover is a polynomial of low (individual) degree; (2) *low-degree soundness*, which relaxes soundness to hold only against provers that send PCP oracles that are low-degree polynomials.

[2] Alternatively, an IPCP can be viewed as a PCP that is verified *interactively* (by an IP, instead of a randomized algorithm).

Low-degree completeness and soundness can be viewed as a promise that the PCP oracle is a low-degree polynomial. Indeed, these conditions are designed to capture "compatibility" with low-degree testing: only protocols with low-degree completeness will pass a low-degree test with probability 1; moreover, adding a low-degree test to an IPCP with low-degree soundness results (roughly) in an IPCP with standard soundness.

2.3.2 From low-degree IPCP to MIP*

We show that any low-degree IPCP can be transformed into a corresponding MIP* protocol, in a way that preserves zero knowledge (for a sufficiently strong notion of zero knowledge IPCP). To this end, we use an entanglement-resistant low degree test, which allows us to essentially restrict the provers usage of the entangled state to strategies that can be approximately implemented via randomness shared among the provers. Informally, the idea is that by carefully invoking such a test, we can let one prover take on the role of the PCP oracle, and the other to take the role of the IPCP prover, and then emulate the entire IPCP protocol.

In more detail, we show a zero-knowledge-preserving transformation of low-degree IPCPs to MIP* protocols, which is captured by the following lifting lemma.

Lemma 2.1 (informally stated, see Lemma 8.1)**.** *There exists a transformation T that takes an r-round low-degree IPCP (P', V') for a language $\mathscr{L}$, and outputs a 2-prover $(\mathsf{r}^* + 2)$-round MIP* $(P_1, P_2, V) := T(P', V')$ for $\mathscr{L}$, where $\mathsf{r}^* = \max\{\mathsf{r}, 1\}$. Moreover, this transformation preserves zero knowledge.*[3]

We stress that the simplicity of the lifting lemma is a key feature since, as we describe below, it requires us to only make small structural changes to the IPCP protocol. This facilitates the preservation of various complexity measures and properties, such as zero knowledge.

To prove this lemma, a key tool that we use is a new low-degree test by Natarajan and Vidick [NV18],[4] which adapts the celebrated plane-vs-point test of Raz and Safra [RS97] to the MIP* model. A *low-degree test* is a procedure used to determine if a given function $f\colon \mathbb{F}^m \to \mathbb{F}$ is close to a low-degree polynomial or if, instead, it is far from all low-degree polynomials, by examining f at very few locations. In the plane-vs-point test, the verifier specifies a random 2-dimensional plane in $\mathbb{F}^m$ to one prover and a random point on this plane to the other prover; each prover replies with the purported value of f on the received plane or point; then the verifier checks that these values are consistent.

Informally, the analysis in [NV18] asserts that every entangled strategy that passes this test with high probably must satisfy an algebraic structure; more specifically, to pass this test the provers can only use their shared entangled state to (approximately) agree on a low-degree polynomial according to which they answer. We use the following soundness analysis of the this protocol. (See Section 6 for the standard quantum notation used in the theorem below.)

Theorem 2.2 ([NV18, Theorem 2], informally stated)**.** *There exists an absolute constant $c \in (0, 1)$ such that, for every soundness parameter $\varepsilon > 0$, number of variables $m \in \mathbb{N}$, degree $d \in \mathbb{N}$, and finite field $\mathbb{F}$, there exists a low-degree test T for which the following holds. For every* symmetric *entangled prover strategy and measurements $\{A_\alpha^z\}_{z\in\mathbb{F},\alpha\in\mathbb{F}^m}$ that are accepted by T with probability at least $1-\varepsilon$, there exists a measurement $\{L^Q\}_Q$, where Q is an m-variate polynomial of degree d, such that:*

1. Approximate consistency with $\{A_\alpha^z\}$: $\mathbb{E}_{\alpha\in\mathbb{F}^m} \sum_Q \sum_{z\neq Q(\alpha)} \langle\Psi| A_\alpha^z \otimes L^Q |\Psi\rangle \leq \varepsilon^c$.

2. Self-consistency of $\{L^Q\}$: $\sum_Q \langle\Psi| L^Q \otimes (\mathrm{Id} - L^Q) |\Psi\rangle \leq \varepsilon^c$.

[3] More accurately, we require the given IPCP to be zero knowledge with query bound that is roughly quadratic in the degree of the PCP oracle. See Section 8.2 for details.

[4] If we do not aim to obtain the optimal number of provers in our MIP* protocols, then it it suffices to use (an adaptation of) the low-degree test in [Vid16].

In fact, we actually use a more refined version, which tests a polynomial's *individual* degree rather than its *total* degree. In the classical setting, such a test is implicit in [GS06] via a reduction from individual-degree to total-degree testing. Informally, this reduction first invokes the test for low total degree, then performs univariate low-degree testing with respect to a random axis-parallel line in each axis. We extend this reduction and its analysis to the setting of MIP*. (See Section 7 for details.) The analysis of the low individual degree test was communicated to us by Thomas Vidick, to whom we are grateful for allowing us to include it here.

With the foregoing low-degree test at our disposal, we are ready to outline the simple transformation from low-degree IPCPs to MIP* protocols. We begin with a preprocessing step. Note that the low individual degree test provides us with means to assert that the provers can (approximately) only use their entangled state to choose a low-degree polynomial Q, and answer the verifier with the evaluation of Q on a *single*, uniformly distributed point (or plane). Thus, it is important that the IPCP verifier (which we start from) only makes a single uniform query to its oracle. By adapting techniques from [KR08], we can leverage the algebraic structure of the low-degree IPCP and capitalize on the interaction to ensure the IPCP verifier has this property, at essentially the cost of increasing the round complexity by 1.[5]

Thus we have a low-degree IPCP, with prover P and verifier V, in which the verification takes place as follows. Both P and V receive an explicit input x that is allegedly in the language $\mathscr{L}$. In addition, V is granted oracle access to a purported low-degree polynomial R, whose full description is known to P. The parties engage in an r-round interaction, at the end of which V is allowed to make a single uniform query to R and decide whether $x \in \mathscr{L}$ (with high probability).

We transform this IPCP into a 2-prover MIP* by considering the following protocol. First, the verifier chooses uniformly at random whether to (1) invoke a low-degree test, in which it asks one prover to evaluate R on a random plane or axis-parallel line and the other prover to evaluate R on a random point on this plane or line, or (2) emulate the IPCP protocol, in which one prover plays the role of the IPCP prover and the other acts as lookup for R.

We use the approximate consistency condition of Theorem 2.2 to assert that the lookup prover approximately answers according to a low-degree polynomial, and use the self-consistency condition to ensure that both provers are consistently answering according to the *same* low-degree polynomial.[6]

We remark that preserving zero knowledge introduces some subtle technicalities (which we resolve), the main of which is that because the analysis of the entanglement-resistant low individual degree test requires that the provers employ *symmetric* strategies, we need to perform a non-standard symmetrization (since standard symmetrization turns out to break zero knowledge in our case). See Section 8.2 for details.

2.3.3 Towards zero knowledge MIP* for nondeterministic exponential time

Equipped with the lifting lemma, we are left with the task of constructing classical zero knowledge low-degree IPCPs for all languages in $\mathbf{NEXP}$. We first explain why current constructions do *not* suffice for this purpose.

The first thing to observe is that the classical protocol for the $\mathbf{NEXP}$-complete language *Oracle 3SAT* by Babai, Fortnow, and Lund [BFL91] (neglecting the multilinearity test) can be viewed as low-degree IPCP. Indeed, in [BFL91] the protocol is stated as an "oracle protocol", which is equivalent to an IPCP. The oracle is encoded as a low-degree polynomial, and so low-degree completeness is satisfied. Alas, the foregoing protocol is *not* zero knowledge. We remark that since the MIP* protocol in [IV12] relies on the protocol in [BFL91], the former inherits the lack of zero knowledge from the latter.

[5] Indeed, if the original IPCP verifier makes a single uniform query to its oracle, then we can save a round in Lemma 2.1; that is, we obtain an MIP* with round complexity $\mathsf{r}^* + 1$, rather than $\mathsf{r}^* + 2$.

[6] Since the players are allowed the use of entanglement, we cannot hope for a single function that underlies their strategy. Indeed, the players could measure their entangled state to obtain shared randomness and select a random R according to which they answer.

Proceeding to consider classical *zero knowledge* proof systems, for example the protocols in [DFKNS92; KPT97; GIMS10], we observe that while some of these proof systems can be viewed as IPCPs, they are not *low-degree* IPCPs. This is because they achieve zero knowledge via combinatorial techniques that do *not* admit the algebraic structure that we require. We stress that the natural way of endowing an IPCP with algebraic structure by taking the low-degree extension of the PCP oracle does *not* necessarily preserve zero knowledge.[7] Correspondingly, the MIP* protocols in [Vid16; NV18], which rely on applying the low-degree extension code to a PCP, do not preserve zero knowledge for this reason.

Finally, we observe that recent advances in algebraic zero knowledge [BCFGRS17] (building on techniques from [BCGV16]) already provide us with a classical proof system that is compatible with our framework, and can thus be used to derive a zero knowledge MIP* protocol, albeit only for languages in $\#\mathbf{P}$.

To strengthen the aforementioned result and show that $\mathbf{NEXP} \subseteq \mathbf{PZK\text{-}MIP}^*$ (matching the $\mathbf{NEXP} \subseteq \mathbf{MIP}^*$ containment, and showing that zero knowledge can, in a sense, be obtained for "free" in the setting of MIP* protocols), we need to construct a *much stronger* zero knowledge low-degree IPCP. The second part of Theorem 1, which is our main technical contribution, provides exactly that. We proceed to provide an overview of the techniques that we use to construct such protocols.

2.4 Part II: new algebraic techniques for zero knowledge

The techniques discussed thus far tell us that, if we wish to obtain a zero knowledge MIP* for $\mathbf{NEXP}$, it suffices to obtain a zero knowledge *low-degree* IPCP for $\mathbf{NEXP}$ (an IPCP wherein the oracle is a low-degree polynomial). Doing so is the second part of our proof of Theorem 1, and for this we develop new algebraic techniques for obtaining zero knowledge protocols. Our techniques, which build on recent developments [BCGV16; BCFGRS17], stand in stark contrast to other known constructions of zero knowledge PCPs and interactive PCPs (such as [DFKNS92; KPT97; GIMS10]). We remind the reader that this part of our work only deals with classical protocols, and does not require any knowledge of quantum information.

2.4.1 A zero knowledge low-degree IPCP for $\mathbf{NEXP}$

Our starting point is the protocol of Babai, Fortnow, and Lund [BFL91] (the "BFL protocol"). We first recall how the BFL protocol works, in order to explain its sources of information leakage and how one could prevent them via algebraic techniques. These are the ideas that underlie our algebraic construction of an unconditional (perfect) zero knowledge low-degree IPCP for $\mathbf{NEXP}$.

The BFL protocol, and why it leaks. *Oracle 3SAT* (O3SAT) is the following $\mathbf{NEXP}$-complete problem: given a boolean formula B, does there exist a boolean function A (a witness) such that

$$B(z, b_1, b_2, b_3, A(b_1), A(b_2), A(b_3)) = 0 \quad \text{for all } z \in \{0,1\}^r, b_1, b_2, b_3 \in \{0,1\}^s \ \ ?$$

The BFL protocol is an IPCP for O3SAT that is then (generically) converted to an MIP. In the BFL protocol, the honest prover first sends a PCP oracle $\hat{A} \colon \mathbb{F}^s \to \mathbb{F}$ that is the unique multilinear extension (in some finite field $\mathbb{F}$) of a valid witness $A \colon \{0,1\}^s \to \{0,1\}$. The verifier must check that (a) $\hat{A}$ is a boolean function on $\{0,1\}^s$, and (b) $\hat{A}$'s restriction to $\{0,1\}^s$ is a valid witness for B. To do these checks, the verifier arithmetizes the formula B into an arithmetic circuit $\hat{B}$, and reduces the checks to conditions that involve $\hat{A}$, $\hat{B}$, and other low-degree polynomials. A technique in [BFLS91] allows the verifier to "bundle" all of these conditions into a single low-degree polynomial f such that (with high probability over the choice of f) the conditions hold if

[7] Intuitively, a single point in the encoded oracle can summarize a large amount of information from the original oracle (e.g., very large linear combinations).

and only if f sums to 0 on $\{0,1\}^{r+3s+3}$. The verifier checks that this is the case by engaging in a sumcheck protocol with the prover.[8]

We observe that the BFL protocol is *not* zero knowledge for two reasons: (i) the verifier has oracle access to $\hat{A}$ and, in particular, to the witness A; (ii) the prover's messages during the sumcheck protocol leak further information about A (namely, hard-to-compute partial sums of f, which itself depends on A).

A blueprint for zero knowledge. We now describe the "blueprint" for an approach to achieve zero knowledge in the BFL protocol. The prover does not send $\hat{A}$ directly, but instead a *commitment* to it. After this, the prover and verifier engage in a sumcheck protocol with suitable zero knowledge guarantees; at the end of this protocol, the verifier needs to evaluate f at a point of its choice, which involves evaluating $\hat{A}$ at three points. Now the prover reveals the requested values of $\hat{A}$, without leaking any information beyond these, so that the verifier can perform its check. We explain how these ideas motivate the need for certain algebraic tools, which we later develop and use to instantiate our approach.

(1) Randomized low-degree extension. Even if the prover reveals only three values of $\hat{A}$, these may still leak information about A. We address this problem via a *randomized low-degree extension*. Indeed, while the prover in the BFL protocol sends the *unique* multilinear extension of A, one can verify that *any* extension of A of sufficiently low degree also works. We exploit this flexibility as follows: the prover randomly samples $\hat{A}$ in such a way that any three evaluations of $\hat{A}$ do not reveal any information about A. Of course, if any of these evaluations is within the systematic part $\{0,1\}^s$, then no extension of A has this property. Nevertheless, during the sumcheck protocol, the prover can ensure that the verifier chooses only evaluations outside of $\{0,1\}^s$ (by aborting if the verifier deviates), which incurs only a small increase in the soundness error.[9] With this modification in place, it suffices for the prover to let $\hat{A}$ be a random degree-4 extension of A: by a dimensionality argument, any 3 evaluations outside of $\{0,1\}^s$ are now independent and uniformly random in $\mathbb{F}$. We are thus able to reduce a claim about A to a claim which contains *no information* about A.

(2) Algebraic commitments. As is typical in zero knowledge protocols, the prover will send a *commitment* to $\hat{A}$, and then selectively reveal a limited set of evaluations of $\hat{A}$. The challenge in our setting is that this commitment must *also* be a low-degree polynomial, since we require a low-degree oracle. For this, we devise a new algebraic commitment scheme based on the sumcheck protocol; we discuss this in Section 2.4.2.

(3) Sumcheck in zero knowledge. We need a sumcheck protocol where the prover's messages leak little information about f. The prior work in [BCFGRS17] achieves an IPCP for sumcheck that is "weakly" zero knowledge: any verifier learns at most one evaluation of f for each query it makes to the PCP oracle. If the verifier could evaluate f by itself, as was the case in that paper, this guarantee would suffice for zero knowledge. In our setting, however, the verifier *cannot* evaluate f by itself because f is (necessarily) hidden behind the algebraic commitment.

One approach to compensate would be to further randomize $\hat{A}$ by letting $\hat{A}$ be a random extension of A of some well-chosen degree d. Unfortunately, this technique is incompatible with our low-degree IPCP to MIP* transformation: such a low-degree extension is at most d-wise independent, whereas our lifting lemma (Lemma 8.1), and more generally low-degree testing, requires zero knowledge against any $\Omega(d^2)$ queries.

We resolve this by relying on more algebraic techniques, achieving an IPCP for sumcheck with a much stronger zero knowledge guarantee: any malicious verifier that makes polynomially-many queries to the PCP oracle learns only a *single* evaluation of f. This suffices for zero knowledge in our setting: learning one evaluation of f implies learning only three evaluations of $\hat{A}$, which can be made "safe" if $\hat{A}$ is chosen to be a random extension of A of sufficiently high degree. Our sumcheck protocol uses as building blocks both our

[8] The soundness of the sumcheck protocol depends on the PCP oracle being the evaluation of a low-degree polynomial, and so the verifier in [BFL91] checks this using a low-degree test. In our setting of *low-degree* IPCPs a low-degree test is not necessary.

[9] The honest verifier will be defined so that it always chooses evaluations *outside* of $\{0,1\}^s$, so completeness is unaffected.

algebraic commitment scheme and the "weak" zero knowledge sumcheck in [BCFGRS17]; we summarize its construction in Section 2.4.3.

2.4.2 Algebraic commitments from algebraic query complexity lower bounds

We provide a high-level description of an information-theoretic commitment scheme in the low-degree IPCP model (i.e., a low-degree *interactive locking scheme* [GIMS10]). See Section 10 for the full details.[10]

In this scheme, the prover commits to a message by sending to the verifier a PCP oracle that perfectly hides the message; subsequently, the prover can reveal positions of the message by engaging with the verifier in an interactive proof, whose soundness guarantees statistical binding.

Committing to an element. We first consider the simple case of committing to a single element a in $\mathbb{F}$. Let k be a security parameter, and set $N := 2^k$. Suppose that the prover samples a random B in $\mathbb{F}^N$ such that $\sum_{i=1}^{N} B_i = a$, and sends B to the verifier as a commitment. Observe that any $N-1$ entries of B do not reveal any information about a, and so any verifier with oracle access to B that makes fewer than N queries cannot learn any information about a. However, as B is unstructured it is not clear how the prover can later convince the verifier that $\sum_{i=1}^{N} B_i = a$.

Instead, we can consider imbuing B with additional algebraic structure. Namely, the prover views B as a function from $\{0,1\}^k$ to $\mathbb{F}$, and sends its unique multilinear extension $\hat{B}\colon \mathbb{F}^k \to \mathbb{F}$ to the verifier. Subsequently, the prover can reveal a to the verifier, and then engage in a sumcheck protocol for the claim "$\sum_{\vec{\beta}\in\{0,1\}^k} \hat{B}(\vec{\beta}) = a$" to establish the correctness of a. The soundness of the sumcheck protocol protects the verifier against cheating provers and hence guarantees that this scheme is binding.

However, giving B additional structure calls into question the hiding property of the scheme. Indeed, surprisingly, a result of Juma et al. [JKRS09] shows that this new scheme is in fact *not* hiding (in fields of odd characteristic): it holds that $\hat{B}(2^{-1},\ldots,2^{-1}) = a \cdot 2^{-k}$ for any choice of B, so the verifier can learn a with only a single query to $\hat{B}$!

Sending an extension of B has created a new problem: querying the extension outside of $\{0,1\}^k$, the verifier can learn information that may require many queries to B to compute. Indeed, this additional power is precisely what underlies the soundness of the sumcheck protocol. To resolve this, we need to understand what the verifier can learn about B given some low-degree extension $\hat{B}$. This is precisely the setting of *algebraic query complexity* [AW09].[11]

Indeed the foregoing theory suggests a natural approach for overcoming the problem created by the extension of B: instead of considering the multilinear extension, we can let $\hat{B}$ be chosen uniformly at random from the set of degree-d extensions of B, for some $d > 1$. It is not hard to see that if d is very large (say, $|\mathbb{F}|$) then 2^k queries are required to determine the summation of $\hat{B}$ on $\{0,1\}^k$. However, we need d to be small to achieve soundness. Fortunately, a result of [JKRS09] shows that $d = 2$ suffices: given a random multiquadratic extension $\hat{B}$ of B, one needs 2^k queries to $\hat{B}$ to determine $\sum_{\vec{\beta}\in\{0,1\}^k} \hat{B}(\vec{\beta})$.[12]

Committing to a polynomial. The prover in our zero knowledge protocols needs to commit not just to a single element but rather to the evaluation of an m-variate polynomial Q over $\mathbb{F}$ of degree $d > 1$. We extend our ideas to this setting. We follow a similar general approach, however, arguing the hiding property now

[10] We use the commitment scheme perspective to illustrate the key ideas in our construction. In the technical sections, we prove the zero knowledge property directly using algebraic query complexity lower bounds, without explicitly using any commitment scheme.

[11] Interestingly, in [AW09] a connection between algebra and zero knowledge is also exhibited. Namely, to show that the result $\mathbf{NP} \subseteq \mathbf{CZK}$ [GMW91] algebrizes, it is necessary to exploit the algebraic structure of the oracle to design a zero knowledge protocol for verifying the existence of certain sets of query answers.

[12] This is the main reason why our application to constructing MIP* protocols requires low-degree test against entangled provers, rather than just a multilinearity test, as was used in [IV12].

requires a *stronger* algebraic query complexity lower bound than the one proved in [JKRS09]. Not only do we need to know that the verifier cannot determine $Q(\vec{\alpha})$ for a particular $\vec{\alpha} \in \mathbb{F}^m$, but we need to know that the verifier cannot determine $Q(\vec{\alpha})$ for *any* $\vec{\alpha} \in \mathbb{F}^m$, or even *any linear combination of any such values*. We prove that this stronger guarantee holds in the same parameter regime: if $d > 1$ then 2^k queries are both necessary and sufficient. See the discussion at the beginning of Section 10 for a more detailed overview.

Decommitting in zero knowledge. To use our commitment scheme in zero knowledge protocols, we must ensure that, in the decommitment phase, the verifier cannot learn any information beyond the value $a := Q(\vec{\alpha})$, for a chosen $\vec{\alpha}$. To decommit, the prover sends the value a and has to convince the verifier that the claim "$\sum_{\vec{\beta} \in \{0,1\}^k} \hat{B}(\vec{\alpha}, \vec{\beta}) = a$" is true. However, if the prover and verifier simply run the sumcheck protocol on this claim, the prover leaks partial sums $\sum_{\vec{\beta} \in \{0,1\}^{k-i}} \hat{B}(\vec{\alpha}, c_1, \ldots, c_i, \vec{\beta})$, for $c_1, \ldots, c_i \in \mathbb{F}$ chosen by the verifier, which could reveal additional information about Q. Instead, the prover and verifier run on this claim the IPCP for sumcheck of [BCFGRS17], whose "weak" zero knowledge guarantee ensures that this cannot happen. (Thus, in addition to the commitment, the honest prover also sends the evaluation of a random low-degree polynomial as required by the IPCP for sumcheck of [BCFGRS17].)

2.4.3 A zero knowledge sumcheck protocol

We describe the "strong" zero knowledge variant of the sumcheck protocol that we use in our construction. The protocol relies on the algebraic commitment scheme described in the previous section. We first cover some necessary background, and then describe our protocol.

Previous sumcheck protocols. The sumcheck protocol [LFKN92] is an IP for claims of the form "$\sum_{\vec{\alpha} \in H^m} F(\vec{\alpha}) = 0$", where H is a subset of a finite field $\mathbb{F}$ and F is an m-variate polynomial over $\mathbb{F}$ of small individual degree. The protocol has m rounds: in round i, the prover sends the univariate polynomial $g_i(X_i) := \sum_{\vec{\alpha} \in H^{m-i}} F(c_1, \ldots, c_{i-1}, X_i, \vec{\alpha})$, where $c_1, \ldots, c_{i-1} \in \mathbb{F}$ were sent by the verifier in previous rounds; the verifier checks that $\sum_{\alpha_i \in H} g_i(\alpha_i) = g_{i-1}(c_{i-1})$ and replies with a uniformly random challenge $c_i \in \mathbb{F}$. After round m, the verifier outputs the claim "$F(c_1, \ldots, c_m) = g_m(c_1, \ldots, c_m)$". If F is of sufficiently low degree and does not sum to a over the space, then the output claim is false with high probability. Note that the verifier does not need access to F.

The "weak" zero knowledge IPCP for sumcheck in [BCFGRS17] modifies the above protocol as follows. The prover first sends a PCP oracle that (allegedly) equals the evaluation of a random "masking" polynomial R; the verifier checks that R is (close to) low degree. Subsequently, the prover and verifier conduct the following interactive proof. The prover sends $z \in \mathbb{F}$ that allegedly equals $\sum_{\vec{\alpha} \in H^m} R(\vec{\alpha})$, and the verifier responds with a uniformly random challenge $\rho \in \mathbb{F}^*$. The prover and verifier now run the (standard) sumcheck protocol to reduce the claim "$\sum_{\vec{\alpha} \in H^m} \rho F(\vec{\alpha}) + R(\vec{\alpha}) = \rho a + z$" to a claim "$\rho F(\vec{c}) + R(\vec{c}) = b$", for a random $\vec{c} \in \mathbb{F}^m$. The verifier queries R at $\vec{c}$ and then outputs the claim "$F(\vec{c}) = \frac{b - R(\vec{c})}{\rho}$". If $\sum_{\vec{\alpha} \in H^m} F(\vec{\alpha}) \neq a$, then with high probability over ρ and the verifier's messages in the sumcheck protocol, this claim is false.

A key observation is that if the verifier makes no queries to R, then the prover's messages are identically distributed to the sumcheck protocol applied to a random polynomial Q. When the verifier does make queries to R, simulating the resulting conditional distribution involves techniques from Algebraic Complexity Theory, as shown in [BCFGRS17]. Given Q, the verifier's queries to $R(\vec{\alpha})$, for $\vec{\alpha} \in \mathbb{F}^m$, are identically distributed to $Q(\vec{\alpha}) - \rho F(\vec{\alpha})$. Thus, the simulator need only make at most one query to F for every query to R; that is, any verifier making q queries to R learns no more than it would learn by making q queries to F alone.

As discussed, this zero knowledge guarantee does not suffice for the application that we consider: in the $\mathbf{NEXP}$ protocol, the polynomial F is defined in terms of the $\mathbf{NEXP}$ witness. In this case the verifier can learn enough about F to break zero knowledge by making only $O(\deg(F))$ queries to R.

Our sumcheck protocol. The "strong" zero knowledge guarantee that we aim for is the following: any polynomial-time verifier learns no more than it would by making *one* query to F, regardless of its number of queries to the PCP oracle.

The main idea to achieve this guarantee is the following. The prover sends a PCP oracle that is an *algebraic commitment* Z to the aforementioned masking polynomial R. Then, as before, the prover and verifier run the sumcheck protocol to reduce the claim "$\sum_{\vec{\alpha} \in H^m} \rho F(\vec{\alpha}) + R(\vec{\alpha}) = \rho a + z$" to a claim "$\rho F(\vec{c}) + R(\vec{c}) = b$" for random $\vec{c} \in \mathbb{F}^m$.

We now face two problems. First, the verifier cannot simply query R at $\vec{c}$ and then output the claim "$F(\vec{c}) = \frac{b - R(\vec{c})}{\rho}$", since the verifier only has oracle access to the commitment Z of R. Second, the prover could cheat the verifier by having Z be a commitment to an R that is far from low degree, which allows cheating in the sumcheck protocol.

The first problem is addressed by the fact that our algebraic commitment scheme has a decommitment sub-protocol that is zero knowledge: the prover can reveal $R(\vec{c})$ in such a way that no other values about R are also revealed as a side-effect. As discussed, this relies on the protocol of [BCFGRS17], used as a subroutine.

The second problem is addressed by the fact that our algebraic commitment scheme is "transparent" to low-degree structure; that is, the algebraic structure of the scheme implies that if the commitment Z is a low-degree polynomial (as in a *low-degree* IPCPs), then R must also be low degree (and vice versa).

Overall, the only value that a malicious verifier can learn is $F(\vec{c})$, for $\vec{c} \in I^m$ of its choice (where I is some sufficiently large subset of $\mathbb{F}$, fixed in advance). More precisely, we prove the following theorem, which shows a strong zero knowledge sumcheck protocol.

Theorem 2.3 (Informally stated, see Theorem 11.1)**.** *There exists a low-degree IPCP for sumcheck, with respect to a low-degree polynomial F, that satisfies the following zero knowledge guarantee: the view of any probabilistic polynomial-time verifier in the protocol can be perfectly and efficiently simulated by a simulator that makes only a single query to F.*

Our sumcheck protocol leaks a single evaluation of F. We stress that this limitation is *inherent*: the honest verifier always outputs a true claim about one evaluation of F, which it cannot do without learning that evaluation. Nevertheless, this guarantee is strong enough for our application, as we can ensure that learning a single evaluation of F does not harm zero knowledge.

We remark that our strong zero knowledge sumcheck protocol can be transformed into a standard IPCP, by the standard technique of adding a (classical) low-degree test to the protocol.

3 Discussion and open problems

The framework that we use to prove that $\mathbf{NEXP} \subseteq \mathbf{PZK\text{-}MIP}^*$ elucidates the role that algebra plays in the design of proofs systems with entangled provers. Namely, we show that a large class of algebraic protocols (low-degree IPCPs) can be transformed in a black box manner to MIPs with entangled provers. This abstraction decouples the mechanisms responsible for soundness against entangled adversaries from other classical components in the proof system. In turn, this allows us to focus our attention on designing proof systems with desirable properties (zero knowledge, in this work), without having to deal with the complications that arise from entanglement, and then derive MIP* protocols from these classical protocols.

These ideas also enable us to re-interpret prior constructions of MIP* protocols at a higher level of abstraction. For example, the protocol in [IV12] can be viewed as applying our lifting lemma to the (low-degree) IPCP in [BFL91]. As another example, one can start with *any* PCP for some language $\mathscr{L}$, low-degree extend the PCP, and then apply our lifting lemma to obtain a corresponding MIP* protocol for $\mathscr{L}$; in fact, the protocol in [Vid16] can be viewed in this perspective.

In more detail, we say that a transformation from IPCP to MIP* is *black box* if it maps an IPCP protocol into an MIP* protocol whose verifier can be expressed as an algorithm that only accesses the queries and messages of the IPCP verifier, but does not access its input (apart from its length). The following corollary shows that any IPCP protocol can be transformed into an MIP* protocol via a black box transformation. While a proof of this fact is implicit in [Vid16; NV18], the framework developed in this paper allows us to crystallize its structure and give a compellingly short proof of it. (See, also, Fig. 1.)

Corollary 3.1. *There is a black box transformation that maps any* r*-round IPCP protocol for a language* $\mathscr{L}$ *to a* 2*-prover* $(\mathsf{r}+1)$*-round MIP** *for* $\mathscr{L}$

The round complexity of $\mathsf{r}+1$ in Corollary 3.1 is less than in our lifting lemma ($\mathsf{r}+2$), because now we do not require that zero knowledge is preserved. We make the foregoing discussion precise in Appendix B.

We conclude this section by discussing several open problems.

In this work we show that there exist perfect zero knowledge MIP* protocols for all languages in $\mathbf{NEXP}$, with *polynomially-many* rounds. Since round complexity is a crucial resource in any interactive proof system, it is essential to understand whether zero knowledge MIP* protocols with low round complexity exist. (After all, without the requirement of zero knowledge, every language in $\mathbf{NEXP}$ has a MIP* protocol with just *one round* [Vid16; NV18].) We remark that the "oracularization" technique of Ito et al. [IKM09] reduces the round complexity of any MIP* to one round, but this technique does *not* preserve zero knowledge.

Open Problem 1. Do there exist constant-round zero knowledge MIP* protocols for $\mathbf{NEXP}$?

At the beginning of this section, we reflected on the fact that known results that establish the power of MIP* protocols rely on *algebraic* structure, which enables classical-to-quantum black box transformations of protocols. But is algebraic structure inherently required, or does some combinatorial structure suffice?

Open Problem 2. Is there a richer class of classical protocols (beyond low-degree IPCPs) that can be black-box transformed into MIP* protocols?

For instance, could we replace low-degree polynomials with, say, error correcting codes with suitable local testability and decodability properties? One place to start would be to understand whether local testers for tensor product codes [BS06] are sound against entangled provers.

Open Problem 3. When suitably adapted to the multi-prover setting, is the random hyperplane test in [BS06] for tensor product codes sound against entangled provers?

4 Roadmap

In Section 5 we provide definitions needed for the technical sections, including that for a *low-degree IPCP*, which is central to our work. In Part I we prove that any low-degree IPCP can be transformed into an MIP* protocol, while preserving zero knowledge; see Lemma 8.1. In Part II we prove that every language in $\mathbf{NEXP}$ has a perfect zero knowledge low-degree IPCP; see Theorem 12.2. Combining the results proved in Parts I and II enables us to derive our main result, Theorem 1, which shows that every language in $\mathbf{NEXP}$ has a perfect zero knowledge 2-prover MIP*. Fig. 1 summarizes the roadmap towards proving Theorem 1.

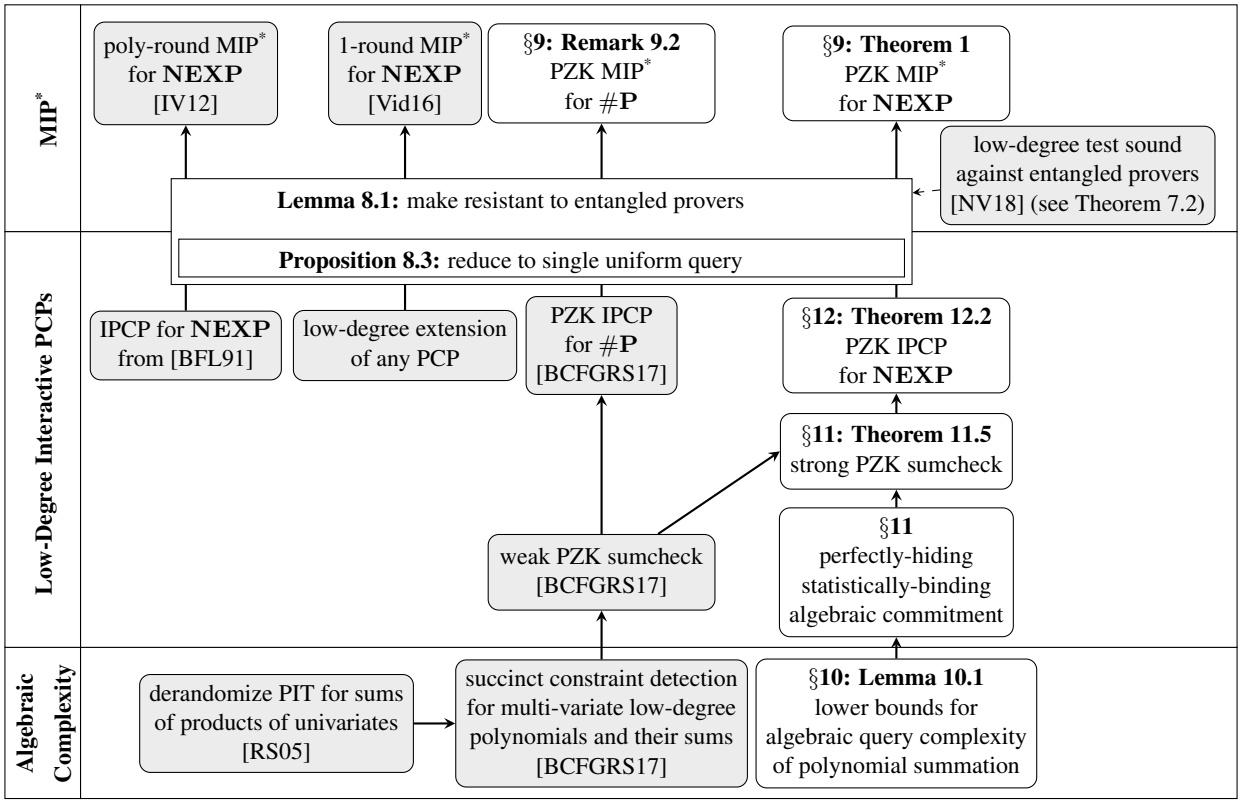

Figure 1: Diagram of the roadmap for proving Theorem 1. White blocks correspond to our new contributions, while grey blocks correspond to previous works. Building on techniques in algebraic complexity from [RS05; BCFGRS17], we prove lower bounds on algebraic query complexity of polynomial summation (Lemma 10.1). This allows us to construct the perfectly-hiding statistically-binding algebraic commitment scheme that underlies our strong perfect zero knowledge sumcheck protocol (Theorem 11.5, which also relies on the weak zero knowledge sumcheck protocol in [BCFGRS17]), and in turn, prove that there exists a perfect zero knowledge low-degree IPCP for any language in $\mathbf{NEXP}$ (Theorem 12.2). Finally, we show a lemma that lifts low-degree IPCPs to MIP* protocols (Lemma 8.1), while *preserving zero knowledge*, and use it to derive our main result (Theorem 1); namely, a perfect zero knowledge low-degree MIP* for any language in $\mathbf{NEXP}$. Taking an alternative route, we can apply our lifting lemma to a zero knowledge low-degree IPCP in [BCFGRS17] to obtain a weaker variant of our main result: a zero knowledge low-degree MIP* for any language in $\#\mathbf{P}$. We also reframe previous works [IV12; Vid16] via our framework.

5 Preliminaries

We cover the notation and basic definitions that are shared by both parts of this paper.

5.1 Notation

For $n \in \mathbb{N}$ we denote by $[n]$ the set $\{1, \ldots, n\}$. For $m, n \in \mathbb{N}$ we denote by $m+[n]$ the set $\{m+1, \ldots, m+n\}$. For a set X, $n \in \mathbb{N}$, $I \subseteq [n]$, and $\vec{x} \in X^n$, we denote by $\vec{x}_I$ the vector $\left(x_i\right)_{i \in I}$ that is $\vec{x}$ restricted to the coordinates in I.

Integrality. All (relevant) integers stated as real numbers are implicitly rounded to the closest integer.

Distance. The *relative Hamming distance* (or just *distance*), over alphabet Σ, between two strings $x, y \in \Sigma^n$ is $\Delta(x, y) := |\{i \in [n] \text{ s.t. } x_i \neq y_i\}|/n$. If $\Delta(x, y) \leq \epsilon$, we say that x is *ϵ-close* to y; otherwise we say that x is *ϵ-far* from y. Similarly, the *relative distance* of x from a non-empty set $S \subseteq \Sigma^n$ is $\Delta(x, S) := \min_{y \in S} \Delta(x, y)$. If $\Delta(x, S) \leq \epsilon$, we say that x is *ϵ-close* to S; otherwise we say that x is *ϵ-far* from S.

Functions, distributions, fields. We use $f \colon D \to R$ to denote a function with domain D and range R; given a subset $\tilde{D}$ of D, we use $f|_{\tilde{D}}$ to denote the restriction of f to $\tilde{D}$. Given a distribution $\mathcal{D}$, we write $x \leftarrow \mathcal{D}$ to denote that x is sampled according to $\mathcal{D}$. We denote by $\mathbb{F}$ a finite field and by $\mathbb{F}_q$ the field of size q. Arithmetic operations over $\mathbb{F}_q$ take time $\operatorname{polylog} q$ and space $O(\log q)$.

Polynomials. We denote by $\mathbb{F}[X_{1,\ldots,m}]$ the ring of m-variable polynomials over the field $\mathbb{F}$. Given a polynomial P in $\mathbb{F}[X_{1,\ldots,m}]$, $\deg_{X_i}(P)$ is the degree of P in the variable X_i. The *individual degree* of a polynomial is its maximum degree in any variable, $\deg(P) := \max_{1 \leq i \leq m} \deg_{X_i}(P)$. Throughout, unless explicitly specified otherwise, we will exclusively work with *individual* degree and often refer to it simply as degree. We denote by $\mathbb{F}[X^{\leq d}_{1,\ldots,m}]$ the subspace of all polynomials $P \in \mathbb{F}[X_{1,\ldots,m}]$ such that $\deg(P) \leq d$.

Low-degree extensions. Given a finite field $\mathbb{F}$, subset $H \subseteq \mathbb{F}$, and number of variables $m \in \mathbb{N}$, the *low-degree extension* (LDE) of a function $f \colon H^m \to \mathbb{F}$ is the *unique* polynomial of individual degree $|H| - 1$ that agrees with f on H^m, i.e., $\hat{f} \in \mathbb{F}[X^{\leq |H|-1}_{1,\ldots,m}]$ such that $\hat{f}(\vec{h}) = f(\vec{h})$ for all $\vec{h} \in H^m$. In particular, $\hat{f} \colon \mathbb{F}^m \to \mathbb{F}$ is defined as follows:

$$\hat{f}(\vec{X}) := \sum_{\vec{\beta} \in H^m} I_{H^m}(\vec{X}, \vec{\beta}) \cdot f(\vec{\beta}) \ ,$$

where $I_{H^m}(\vec{X}, \vec{Y}) := \prod_{i=1}^{m} \sum_{\omega \in H} \prod_{\gamma \in H \setminus \{\omega\}} \frac{(X_i - \gamma)(Y_i - \gamma)}{(\omega - \gamma)^2}$ is the unique polynomial in $\mathbb{F}[X^{\leq |H|-1}_{1,\ldots,m}]$ such that, for all $(\vec{\alpha}, \vec{\beta}) \in H^m \times H^m$, $I_{H^m}(\vec{\alpha}, \vec{\beta})$ equals 1 when $\vec{\alpha} = \vec{\beta}$ and equals 0 otherwise. Note that $I_{H^m}(\vec{X}, \vec{Y})$ can be generated and evaluated in time $\operatorname{poly}(|H|, m, \log |\mathbb{F}|)$ and space $O(\log |\mathbb{F}| + \log m)$, so $\hat{f}(\vec{\alpha})$ can be evaluated in time $|H|^m \cdot \operatorname{poly}(|H|, m, \log |\mathbb{F}|)$ and space $O(m \cdot \log |\mathbb{F}|)$.

Languages and relations. We denote by $\mathscr{L}$ a language consisting of *instances* x, and by $\mathscr{R}$ a (binary ordered) relation consisting of pairs (x, w), where x is the *instance* and w is the *witness*. We denote by $\operatorname{Lan}(\mathscr{R})$ the language corresponding to $\mathscr{R}$, and by $\mathscr{R}|_x$ the set of witnesses in $\mathscr{R}$ for x (if $x \notin \operatorname{Lan}(\mathscr{R})$ then $\mathscr{R}|_x := \emptyset$). We assume that $|w|$ is bounded by some computable function of $n := |x|$; in fact, we are mainly interested in relations arising from nondeterministic languages: $\mathscr{R} \in \mathbf{NTIME}(T)$ if there exists a $T(n)$-time machine M such that $M(x, w)$ outputs 1 if and only if $(x, w) \in \mathscr{R}$. We assume that $T(n) \geq n$.

Randomized algorithms and oracle access. We denote by $A^R(x)$ the output of an algorithm A when given an input x (explicitly) and query access to an oracle R. If A is probabilistic then $A^R(x)$ is a random variable and, when writing expressions such as "$\Pr[A^R(x) = z]$", we mean that the probability is taken over

A's internal randomness (in addition to other randomness beyond it). The algorithm is said to be b-*query* if it makes *strictly less than* b queries to its oracle. Given two interactive algorithms (protocols) A and B, we denote by $(A^R(x), B^R(y))(z)$ the output of $A^R(x)$ when interacting with $B^R(y)$ on common input z.

5.2 Low-degree interactive PCPs

An *interactive PCP* (IPCP) [KR08] is a probabilistically checkable proof (PCP) verifiable via an interactive proof (IP). In more detail, an IPCP protocol for a language $\mathscr{L}$ is a pair of probabilistic interactive algorithms (P, V), where the *prover* P is computationally unbounded and the *verifier* V runs in polynomial time. Both parties receive an (explicit) input $x \in \{0,1\}^n$, allegedly in the language $\mathscr{L}$, and engage in a two-phase protocol as follows. First, P sends to V an oracle proof string $\pi \in \{0,1\}^*$. Second, P and V^π (i.e., V with oracle access to π) engage in an interactive protocol, at the end of which V either accepts or rejects. The completeness property requires that, if $x \in \mathscr{L}$, then there exists a prover P such that $\Pr[(P, V)(x) = 1] = 1$. The soundness property requires that, if $x \notin \mathscr{L}$, then for any prover $\tilde{P}$ it holds that $\Pr[(\tilde{P}, V)(x) = 1] \leq \varepsilon(n)$, where ε is called the *soundness error*. Unless specified otherwise, we define our IPCP protocols with respect to a small constant soundness error, say, $\varepsilon = 1/2$.

A round of interaction consists of one message from each of the parties. We say that an IPCP has *round complexity* r if the second step of the interaction (the standard IP) consists of r rounds. The *PCP length* of an IPCP is the length of π. The *communication complexity* is the total number of bits exchanged between the parties *except* for the message that contains π. The *query complexity* is the number of queries that V makes to the PCP π. We write

$$\mathscr{L} \in \mathbf{IPCP}\left[\begin{array}{rl} \text{round complexity:} & \mathsf{r} \\ \text{PCP length:} & \mathsf{l} \\ \text{communication complexity:} & \mathsf{c} \\ \text{query complexity:} & \mathsf{q} \\ \text{soundness error:} & \varepsilon \end{array}\right]$$

to indicate that a language $\mathscr{L}$ has an IPCP with the specified parameters.

Low-degree IPCPs. A key tool that we use is *low-degree IPCPs*, a class of algebraically-structured IPCPs. Informally, these are IPCPs in which the PCP oracle is *promised* to be a low-degree polynomial. In more detail, given a field $\mathbb{F}$, number of variables m, and degree d, we say that an IPCP protocol (P, V) is an $(\mathbb{F}, d, m)$-low-degree IPCP if the following conditions hold.

- *Low-degree completeness*: The PCP oracle that the (honest) prover P sends is a polynomial Q in $\mathbb{F}[X_{1,\ldots,m}^{\leq d}]$.
- *Low-degree soundness*: soundness is merely required to hold against provers $\tilde{P}$ that send PCP oracles that are polynomials $\tilde{Q}$ in $\mathbb{F}[X_{1,\ldots,m}^{\leq d}]$.

We remark that the notion of low-degree IPCPs is closely related to *holographic* IPCPs and IPs [RRR16; GR17]. However, whereas in holographic proof systems the *input* is guaranteed to be encoded as a low-degree polynomial, in low-degree IPCPs the oracle may not be related to the input in any way.

Public-coin interaction. Our protocols and transformations refer to *public-coin* proof systems. We remark that the only part wherein we rely on public-coin interaction is in the transformation of IPCPs to low-degree IPCPs in Appendix A. In fact, for this transformation it suffices to rely on a weaker condition that is implied by public-coin interaction; namely, all we require is that the verifier queries the PCP oracle *after* the communication with the prover terminates.

Adaptivity. For simplicity, we assume that all (public-coin) IPCP verifiers make non-adaptive queries to their oracle. However, all of our results can be extended, in a straightforward way, to hold with respect to verifiers that make adaptive queries, at the cost of an increase in round complexity. (See Remark 8.5.)

Zero knowledge. We consider the standard notion of (perfect) zero knowledge for IPCPs [GIMS10; BCFGRS17]. Let A, B be algorithms and x, y strings. We denote by $\mathrm{View}\ \langle B(y), A(x)\rangle$ the *view* of $A(x)$

in an IPCP protocol with $B(y)$, i.e., the random variable $(x, r, s_1, \ldots, s_n, t_1, \ldots, t_m)$ where x is A's input, r is A's randomness, $s_1, \ldots, s_n$ are B's messages, and $t_1, \ldots, t_m$ are the answers to A's queries to the proof oracle sent by B.

An IPCP protocol (P, V) for a language $\mathscr{L}$ is *(perfect) zero knowledge against query bound* b if there exists a polynomial-time simulator algorithm S such that for every b-query algorithm $\tilde{V}$ and input $x \in \mathscr{L}$ it holds that $S^{\tilde{V}}(x)$ and $\text{View}\,\langle P(x), \tilde{V}(x)\rangle$ are identically distributed. We write

$$\mathscr{L} \in \mathbf{PZK\text{-}IPCP}\left[\begin{array}{rl} \text{round complexity:} & \mathsf{r} \\ \text{PCP length:} & \mathsf{l} \\ \text{communication complexity:} & \mathsf{c} \\ \text{query complexity:} & \mathsf{q} \\ \text{query bound:} & \mathsf{b} \\ \text{soundness error:} & \varepsilon \end{array}\right]$$

to indicate that a language $\mathscr{L}$ has a perfect zero knowledge IPCP with the specified parameters.

Remark 5.1 (straightline simulators)**.** The aforementioned works ([GIMS10; BCFGRS17]) consider a stronger notion of zero knowledge IPCPs in which the simulator is *straightline*, i.e., the simulator cannot rewind the verifier. All of the simulators that we construct in this work are straightline too; even so, all of the transformations presented in this work preserve zero knowledge even for simulators that rewind the verifier.

Part I
Low-degree IPCP to MIP*

In this part we build on recent advances in low-degree testing against entangled provers [IV12; Vid16; NV18] to prove a *lifting lemma* that transforms a class of algebraically-structured classical protocols, namely low-degree interactive PCPs, into MIP* protocols, while *crucially*, preserving zero knowledge.

Organization. We begin in Section 6 by covering the necessary preliminaries regarding quantum information and proof systems with entangled provers. In Section 7 we discuss the main technical tool that we need: a low-degree test against entangled provers, which we refine from a total degree to an individual degree test. Then, in Section 8 we state and prove our transformation of low-degree IPCP to MIP*, while preserving zero knowledge. Finally, in Section 9 we prove our main result (Theorem 1) by applying the foregoing transformation to a zero knowledge low-degree IPCP for $\mathbf{NEXP}$, which we construct in Part II.

6 Preliminaries: proof systems with entangled provers

We begin with standard preliminaries in quantum information. Let $\mathcal{H}$ be a finite-dimensional Hilbert space, and let $r \in \mathbb{N}$.

States and operators. We define *entangled quantum states*, which for brevity, we will refer to simply as entangled states. An r-register entangled state $|\Psi\rangle$ is a unit vector in $\mathcal{H}^{\otimes r}$. We say that $|\Psi\rangle$ is *permutation-invariant* if $\sigma |\Psi\rangle = |\Psi\rangle$ for every linear operator that permutes the r registers of $\mathcal{H}^{\otimes r}$. We denote by $\rho = \rho(|\Psi\rangle)$ the reduced density of $|\Psi\rangle$ on a number of registers that will be always clear from the context, so that for an operator A we have

$$\mathrm{Tr}_\rho(A) := \mathrm{Tr}(A\rho) = \langle\Psi| A \otimes \mathrm{Id} \otimes \cdots \otimes \mathrm{Id} |\Psi\rangle \ .$$

Let $\mathcal{L}(\mathcal{H})$ be the set of linear operators over $\mathcal{H}$. Denote by Id the identity operator in $\mathcal{L}(\mathcal{H})$. For $r \geq 2$, the entangled state $|\Psi\rangle$ induces a bilinear form on $\mathcal{L}(\mathcal{H}) \times \mathcal{L}(\mathcal{H})$, given by

$$\langle A, B\rangle_\Psi = \langle\Psi| A \otimes B \otimes \mathrm{Id}^{\otimes(r-2)} |\Psi\rangle \in \mathbb{C} \ ,$$

as well as a semi-norm, given by

$$\|A\|_\Psi = \sqrt{\langle\Psi| AA^\dagger \otimes \mathrm{Id}^{\otimes(r-1)} |\Psi\rangle} \ .$$

Measurements. All measurement in this work are POVMs (Positive Operator Valued Measures). A *measurement* on $\mathcal{H}$ is a finite set $A = \{A^i\}_{i\in S}$ where S is the set of measurement outcomes and the A^i's are non-negative definite operators on $\mathcal{H}$ such that $\sum_{i\in S} A^i = \mathrm{Id}$. A *sub-measurement* on $\mathcal{H}$ relaxes the aforementioned condition by only requiring that $\sum_{i\in S} A_i \leq \mathrm{Id}$. The following standard claim provides a quantitive bound on the distance between two measurements as a function of their correlation.

Claim 6.1 (Approximate consistency to trace distance [Vid11; Vid16])**.** *Let $|\Psi\rangle$ be a permutation-invariant entangled state on $r \geq 2$ registers, and let $\{A^z\}$, $\{B^z\}$ be single-register measurements with outcomes in the same set. Then,*

$$\sum_z \|A^z - B^z\|_\Psi^2 \leq O\left(\max\left\{1 - \sum_z \langle A^z, B^z\rangle_\Psi, 1 - \sum_z \langle A^z, A^z\rangle_\Psi\right\}\right) \ .$$

We shall also need a specific variant of Winter's gentle measurement lemma [Win99], due to Ogawa and Nagaoka [ON07], which formalizes the intuition that measurements that with high probability output a particular outcome on a certain quantum state imply that the post-measurement state is close to the original state.

Lemma 6.2 ([ON07])**.** *Let ρ be a density operator on a Hilbert space $\mathcal{H}$, and let $A, B \in \mathcal{L}(\mathcal{H})$ be such that $A^\dagger A, B^\dagger B \leq \mathrm{Id}$. Then,*

$$\left\| A\rho A^\dagger - B\rho B^\dagger \right\|_1 \leq 2\sqrt{\mathrm{Tr}\left((A-B)\rho(A-B)^\dagger\right)} \ .$$

MIPs with entangled provers. A *multi-prover interactive proof with entangled provers* (MIP*) [CHTW04] is a multi-prover interactive proof (MIP) in which the spatially-isolated (honest and malicious) provers are allowed to use entangled strategies, i.e., any strategy obtained by measuring a shared entangled state.

In more detail, a k-prover r-round MIP* for a language $\mathscr{L}$ is a tuple of probabilistic interactive algorithms $(P_1, \ldots, P_\mathsf{k}, V)$, where the *provers* $P_1, \ldots, P_\mathsf{k}$ are computationally unbounded and the *verifier* V runs in polynomial time. All parties receive an input $x \in \{0,1\}^n$, and the k provers share an entangled state $|\Psi\rangle \in \mathcal{H}^{\otimes \mathsf{k}}$ (which may depend on $|\Psi\rangle$), for a Hilbert space $\mathcal{H}$. The parties engage in a protocol of the following type. In each of the r rounds, each prover P_i receives a (classical) message from the verifier in a message register, performs a quantum operation on this register together with its share of the entangled state $|\Psi\rangle$, measures the message register, and sends back the (classical) outcome to the verifier.

We require perfect completeness and soundness with a given error ε. If $x \in \mathscr{L}$ then $(P_1, \ldots, P_\mathsf{k})$ make V accept with probability 1; if $x \notin \mathscr{L}$ then V rejects *every* prover strategy with probability at least $1 - \varepsilon(n)$, where ε is called the *soundness error*. We stress that the latter condition, soundness, makes no assumptions on the computational power of the provers, nor regarding the number of entangled qubits they share. Unless specified otherwise, we define an MIP* with respect to a small constant soundness error, say, $\varepsilon = 1/2$.[13]

We indicate that a language $\mathscr{L}$ has a k-prover r-round MIP* with soundness error ε and with communication complexity c (the total number of bits exchanged between the verifier and the provers) as follows:

$$\mathscr{L} \in \mathbf{MIP}^* \begin{bmatrix} \text{number of provers:} & \mathsf{k} \\ \text{round complexity:} & \mathsf{r} \\ \text{communication complexity:} & \mathsf{c} \\ \text{soundness error:} & \varepsilon \end{bmatrix} .$$

Zero knowledge. We extend the standard definition of perfect zero knowledge MIPs [BGKW88] to the setting of MIP* in the natural way. Denote by $\mathrm{View}\ \langle P_1, \ldots, P_\mathsf{k}, \tilde{V}\rangle(x)$ the *view* of a (possibly malicious) verifier $\tilde{V}$ in a k-prover MIP* with provers $P_1, \ldots, P_\mathsf{k}$ and input x; that is, the verifier's view is the random variable consisting of the input x, the verifier's random string, and the provers' messages to the verifier.

An MIP* $(P_1, \ldots, P_\mathsf{k}, V)$ for a language $\mathscr{L}$ is *perfect zero knowledge* if there exists a probabilistic polynomial-time simulator algorithm S such that for every probabilistic polynomial-time algorithm $\tilde{V}$ and input $x \in \mathscr{L}$ it holds that $S^{\tilde{V}}(x)$ and $\mathrm{View}\ \langle P_1, \ldots, P_\mathsf{k}, \tilde{V}\rangle(x)$ are identically distributed. All simulators for MIP* protocols in this work achieve the stronger notion of universal *straightline simulators* [FS89; DS98], in which the simulator do *not* rewind the verifier. We write

$$\mathscr{L} \in \mathbf{PZK\text{-}MIP}^* \begin{bmatrix} \text{number of provers:} & \mathsf{k} \\ \text{round complexity:} & \mathsf{r} \\ \text{communication complexity:} & \mathsf{c} \\ \text{soundness error:} & \varepsilon \end{bmatrix}$$

to indicate that a language $\mathscr{L}$ has an perfect zero knowledge MIP* with the specified parameters.

[13] This constant is arbitrary and can be amplified via parallel repetition for entangled strategies [KV11; Yue16].

Quantum malicious verifiers. The MIP* model requires quantum entangled provers and *classical* verifiers (in contrast to the QMIP* model, in which both the provers and the verifier are quantum), and so our notion of zero knowledge is with respect to classical verifiers. Nevertheless, we remark that our results extend to hold against *quantum* malicious verifiers. This is because: (1) the honest verifier is classical, and so the provers can enforce classical communication by systematically measuring the verifier's answers in the computational basis, and (2) all of our simulators are *straightline* (i.e., they do *not* rewind the verifier), and so they avoid the key hurdle for simulators of quantum verifiers, which is that quantum algorithms cannot be rewinded (as quantum information cannot be copied, and measurements are irreversible processes).

Symmetric strategies. Symmetry plays an important simplifying role in the analysis of MIP* protocols. Using the technique of Kempe et al. [KKMTV11], we can symmetrize an MIP* as follows.

Definition 6.3. *Let $(P_1, \ldots, P_{\mathsf{k}}, V)$ be a k-prover MIP* protocol with an entangled state $|\Psi\rangle$. The symmetrized prover strategy $(P'_1, \ldots, P'_{\mathsf{k}}) = \mathsf{Symm}(P_1, \ldots, P_{\mathsf{k}})$ with a* permutation-invariant *entangled state $|\Psi'\rangle$ is defined as follows. Let S_{k} be the set of all permutations of $[\mathsf{k}]$, and assume (via padding) that the registers of $|\Psi\rangle$ are of the same dimension. Set $|\Psi'\rangle := \sum_{\sigma \in S_{\mathsf{k}}} |\sigma(1)\rangle \cdots |\sigma(\mathsf{k})\rangle \otimes |\Psi^\sigma\rangle$, where the the register containing $|\sigma(i)\rangle$ is given to the i-th prover, and $|\Psi^\sigma\rangle$ is obtained by permuting $|\Psi\rangle$ according to σ. For every $i \in [\mathsf{k}]$, the prover P_i measures the register containing $|\sigma(i)\rangle$ and applies $P_{\sigma(i)}$.*

The following claim states that if the verifier treats provers symmetrically, then we can assume, without loss of generality, that each prover strategy is accepted with the same probability as its symmetrized strategy.

Claim 6.4 ([KKMTV11], restated)**.** *Let $(P_1, \ldots, P_{\mathsf{k}}, V)$ be a k-prover MIP* for a language $\mathscr{L}$. If the verifier V treats the provers symmetrically, then the probability that $\mathsf{Symm}(P_1, \ldots, P_{\mathsf{k}})$ makes V accept is equal to the probability that $(P_1, \ldots, P_{\mathsf{k}})$ makes V accept.*

7 Low individual-degree testing against entangled quantum strategies

A *low-degree test* is a procedure used to determine if a given function $f\colon \mathbb{F}^m \to \mathbb{F}$ is close to a low-degree polynomial in $\mathbb{F}[X_{1,\ldots,m}]$ or if, instead, it is far from all low-degree polynomials, by examining f at very few locations. A test typically consists of examining f at random restrictions, such as a point, line, or plane.

Low-degree tests can also be phrased in the setting of multiple non-communicating provers, where each prover is (allegedly) answering queries about the same function f that is being tested. For example, the celebrated line-vs-point test of Rubinfeld and Sudan [RS96] can be viewed as a 2-prover 1-round MIP protocol. The verifier specifies a random line in $\mathbb{F}^m$ to one prover and a random point on this line to the other prover; each prover replies with the purported value of f on the received line or point; then the verifier checks that these values are consistent.

Loosely speaking, the classical analysis of this test asserts the following conditions: (1) *approximate consistency with a low-degree polynomial*, i.e., each player acts as a lookup for a function that is (close to) a low-degree polynomial; and (2) *self-consistency between the provers*, i.e., both players answer according to the *same* function.

In this paper we rely on a similar low-degree test, the plane-vs-point test [RS97], adapted to the setting of MIP*, whose analysis asserts the quantum analogue of the conditions above. In fact, we use a more refined version, which tests a polynomial's *individual* degree rather than its *total* degree. In the classical setting, such a test is implicit in [GS06, Section 5.4.2] via a reduction from individual-degree to total-degree testing. Informally, this reduction first invokes the test for low total degree, then performs univariate low-degree testing with respect to a random axis-parallel line in each axis. The extension of this reduction to the quantum setting yields an MIP* for individual-degree testing.

Low individual degree test. Let $\mathrm{Planes}(U)$ be the set of all *planes* in a vector space U (every $s \in \mathrm{Planes}(U)$ is a 2-dimensional affine subspace of U). The plane-vs-point test, with respect to individual degree, for MIP* is the following 2-prover 1-round protocol.

Construction 7.1. Let $\mathbb{F}$ be a finite field, $m \in \mathbb{N}$ the number of variables, and $d \in \mathbb{N}$ the *individual* degree. The quantum plane-vs-point $(\mathbb{F}, d, m)$-low-individual-degree test is an MIP* (P_1, P_2, V) in which P_1, P_2 claim that a certain function $Q\colon \mathbb{F}^m \to \mathbb{F}$ is a polynomial of individual degree d, and the (honest) interaction is as follows.

First the verifier V symmetrizes the protocol: with probability $1/2$ it assigns the roles $\mathcal{P}_{\text{lookup}}$ to the first prover and $\mathcal{P}_{\text{plane}}$ to the second prover, and with probability $1/2$ it assigns $\mathcal{P}_{\text{plane}}$ to the first prover and $\mathcal{P}_{\text{lookup}}$ to the second prover. Then V chooses uniformly at random one of the following tests.

- $(\mathbb{F}, md, m)$*-total-degree test:*
  1. The verifier V samples a random plane $s \in \mathrm{Planes}(\mathbb{F}^m)$ and a random point $\alpha \in s$ on that plane.
  2. V sends the plane s to $\mathcal{P}_{\text{plane}}$, and the line α to $\mathcal{P}_{\text{lookup}}$.
  3. $\mathcal{P}_{\text{plane}}$ replies with $g := Q \circ s$ (the bivariate polynomial obtained by restricting Q to s).
  4. $\mathcal{P}_{\text{lookup}}$ replies with $z := Q(\alpha)$ (the value of Q at α).
  5. V checks that g is a polynomial of total degree md and accepts if and only if $g(\alpha) = z$.

- *Axis-parallel univariate* $(\mathbb{F}, d)$*-degree test:*
  1. The verifier V samples a random plane $s \in \mathrm{Planes}(\mathbb{F}^m)$, a random point $\alpha \in s$ on that plane, and a random axis-parallel line ℓ passing through the point α.
  2. V sends the plane s to $\mathcal{P}_{\text{plane}}$, and the line ℓ to $\mathcal{P}_{\text{lookup}}$.
  3. $\mathcal{P}_{\text{plane}}$ replies with $g := Q \circ s$ (the bivariate polynomial obtained by restricting Q to s).

4. $\mathcal{P}_{\text{lookup}}$ replies with $h := Q \circ \ell$ (the univariate polynomial obtained by restricting Q to ℓ).
5. V checks that h is a polynomial of individual degree d and accepts if and only if $g(\alpha) = h(\alpha)$.

Note that the verifier V treats the provers symmetrically. Perfect completeness follows since if Q is indeed a polynomial of individual degree d, then $g := Q \circ s$ is a bivariate polynomial of total degree md and $h := Q \circ s$ is a univariate polynomial of degree d, and so both provers can simply answer according to Q. We are grateful to Thomas Vidick for allowing to include the following theorem (and its proof), which shows that the plane-vs-point individual-degree test in Construction 7.1 is sound against entangled strategies.

Theorem 7.2 (quantum low individual degree test)**.** *There exist absolute constants $c \in [0,1]$ and $C \geq 1$ such that the following holds. Let $\varepsilon > 0$, $m, d \in \mathbb{N}$, and let $\mathbb{F}$ be a finite field of size $|\mathbb{F}| = (md/\varepsilon)^C$. Let P be a symmetric prover strategy using entangled state $|\Psi\rangle \in \mathcal{H} \otimes \mathcal{H}$ and projective measurements $\{A^z_\alpha\}_{\alpha \in \mathbb{F}^m, z \in \mathbb{F}}$. If the strategy (P, P) is accepted by the $(\mathbb{F}, d, m)$-low-degree test in Construction 7.1 with probability at least $1 - \varepsilon$, then there exists a measurement $\{L^Q\}_{Q \in \mathbb{F}[X^{\leq d}_{1,\ldots,m}]}$ that satisfies the following properties.*

1. Approximate consistency with $\{A^z_\alpha\}_{z \in \mathbb{F}, \alpha \in \mathbb{F}^m}$:

$$\mathbb{E}_{\alpha \in \mathbb{F}^m} \sum_{Q \in \mathbb{F}[X^{\leq d}_{1,\ldots,m}]} \sum_{\substack{z \in \mathbb{F} \\ z \neq Q(\alpha)}} \langle \Psi | \, A^z_\alpha \otimes L^Q \, | \Psi \rangle \leq \varepsilon^c \ .$$

2. Self-consistency of $\{L^Q\}_{Q \in \mathbb{F}[X^{\leq d}_{1,\ldots,m}]}$:

$$\sum_{Q \in \mathbb{F}[X^{\leq d}_{1,\ldots,m}]} \langle \Psi | \, L^Q \otimes (\mathrm{Id} - L^Q) \, | \Psi \rangle \leq \varepsilon^c \ .$$

Proof of Theorem 7.2. The proof relies on the analysis of the plane-vs-point test [RS97] for MIP*, due to Natarajan and Vidick [NV18], which asserts that the provers in an MIP* are answering according to a polynomial of low *total* degree. This new analysis improves on the analysis of the multilinearity test in [IV12] and the 3-prover low-degree test in [Vid16].[14]

Throughout, we fix $\varepsilon > 0$, $m, d \in \mathbb{N}$, and a finite field $\mathbb{F}$ such that $|\mathbb{F}| = (md/\varepsilon)^C$ for the absolute constant $C \geq 1$ in Theorem 7.2. Furthermore, we assume that all MIP* prover strategies are symmetric with respect to a permutation-invariant bipartite entangled state and that all measurements are projective.

Recall that the $(\mathbb{F}, md, m)$-total-degree test (a sub-procedure in Construction 7.1) is an adaptation of the classical plane-vs-point test to the setting of 2-prover 1-round MIP*, in which the verifier specifies a random 2-dimensional plane in $\mathbb{F}^m$ to one prover and a random point on this plane to the other prover; each prover replies with the purported value of f on the received plane or point; and the verifier checks that these values are consistent. The following theorem shows that this test is sound against entangled quantum provers.

Theorem 7.3 (Natarajan and Vidick [NV18])**.** *There exists an absolute constant $c \in [0,1]$ such that the following holds. Let (P, P) be a symmetric prover strategy using an entangled state $|\Psi\rangle \in \mathcal{H} \otimes \mathcal{H}$ and measurements $\{A^z_\alpha\}_{z \in \mathbb{F}, \alpha \in \mathbb{F}^m}$. If the strategy (P, P) is accepted by the $(\mathbb{F}, md, m)$-total-degree test with probability at least $1 - \varepsilon$, then there exists a measurement $\{L^Q\}_{Q \in \mathbb{F}[X^{\leq d}_{1,\ldots,m}]}$ that satisfies the following.*

[14] The extension of the analysis to a low-degree test (rather a multilinearity test as in [IV12]) is crucial for our results (see discussion in Section 10). Instead, the improvement of the 3-prover test in [Vid16] to the 2-prover test in [NV18] simply reduces the number of provers required to obtain zero knowledge.

1. Approximate consistency with $\{A^z_\alpha\}_{z\in\mathbb{F},\alpha\in\mathbb{F}^m}$:

$$\mathbb{E}_{\alpha\in\mathbb{F}^m} \sum_{Q\in\mathbb{F}[X^{\leq d}_{1,\ldots,m}]} \sum_{\substack{z\in\mathbb{F}\\ z\neq Q(\alpha)}} \langle\Psi| A^z_\alpha \otimes L^Q |\Psi\rangle \leq \varepsilon^c \ .$$

2. Self-consistency of $\{L^Q\}_{Q\in\mathbb{F}[X^{\leq d}_{1,\ldots,m}]}$:

$$\sum_{Q\in\mathbb{F}[X^{\leq d}_{1,\ldots,m}]} \langle\Psi| L^Q \otimes (\mathrm{Id} - L^Q) |\Psi\rangle \leq \varepsilon^c \ .$$

We remark that the above result is stated in [NV18] for finite fields of prime order. Nevertheless, inspection of the proof there reveals that the result in fact holds for *any* finite field.

Recall that, in the $(\mathbb{F}, d, m)$-low-individual-degree test (Construction 7.1), the verifier flips a coin to choose whether to invoke the aforementioned $(\mathbb{F}, md, m)$-*total-degree test* or the *axis-parallel univariate* $(\mathbb{F}, d)$-*degree test*. In the latter, the verifier samples a random plane $s \in \mathrm{Planes}(\mathbb{F}^m)$, a random point $\alpha \in s$ on that plane, and a random axis-parallel line ℓ passing through the point α; sends the line to one prover and the plane to the other; and checks that the provers reply with low-degree polynomials that agree on α.

The *total*-degree test reduces the prover to performing a measurement with outcomes in the set of polynomials of total degree md; we can then argue that, given this, the total contribution of outcomes where the polynomial has *individual* degree greater than d in any one variable is small. We do so by relating the probability that the prover obtains these "bad" outcomes to the rejection probability in the axis-parallel univariate test.

Let T_{md} be the set of all m-variate polynomials over $\mathbb{F}$ of *total* degree md. Since the $(\mathbb{F}, d, m)$-low-individual-degree test invokes the $(\mathbb{F}, md, m)$-*total-degree test* with probability $1/2$, Theorem 7.3 implies that there exist measurements $\{L^Q\}_{Q\in T_{md}}$ such that the conclusions of Theorem 7.3 hold with respect to soundness error $\varepsilon' := 2\varepsilon$.

Let $\mathrm{Lines}(U)$ be the set of all *lines* in a vector space U (every $\ell \in \mathrm{Lines}(U)$ is a 1-dimensional affine subspace of U), and let $\{M^v_\ell\}_{\ell\in\mathrm{Lines}(\mathbb{F}^m),\, v\colon \ell\to\mathbb{F}}$ be the measurement applied by a prover when asked for a line ℓ. Without loss of generality, assume that the outcomes range over univariate polynomials of degree at most d, since any other outcome is rejected by the verifier.

Observe that the probability of the verifier rejecting in the axis-parallel univariate $(\mathbb{F}, d)$-degree test is at least the probability that the line obtained via the M-measurement disagrees with the point obtained via the A-measurement. More precisely,

$$\varepsilon' \geq \mathbb{E}_{\ell\in\mathrm{Lines}(\mathbb{F}^m),\, \alpha\in\ell} \sum_{z\in\mathbb{F},\, v\colon \ell\to\mathbb{F} \text{ s.t. } v(\alpha)\neq z} \langle A^z_\alpha, M^v_\ell\rangle_\Psi \ .$$

Recall that $\deg(Q)$ denotes the *individual degree* of a polynomial Q, and denote by $Q(\ell)$ the evaluations of Q over the line ℓ. Using the approximate consistency condition from Theorem 7.3 and the fact that the

marginal on α is uniform,

$$\begin{aligned}
2\varepsilon' + O\left(\sqrt{\varepsilon^c}\right) &\geq \mathbb{E}_{\ell \in \mathrm{Lines}(\mathbb{F}^m), \alpha \in \ell} \sum_{\substack{Q \in T_{md},\, v\colon \ell \to \mathbb{F} \\ \text{s.t. } Q(z) \neq v(z)}} \langle L^Q,\, M_\ell^v \rangle_\Psi \\
&\geq \mathbb{E}_{\ell \in \mathrm{Lines}(\mathbb{F}^m)} \sum_{\substack{Q \in T_{md} \\ \text{s.t. } Q(\ell) \neq v}} \langle L^Q,\, M_\ell^v \rangle_\Psi - O\left(\frac{md}{|\mathbb{F}|}\right) \\
&\geq \mathbb{E}_{\ell \in \mathrm{Lines}(\mathbb{F}^m)} \sum_{Q\colon \deg(Q) > d} \langle L^Q,\, \mathrm{Id} \rangle_\Psi - O\left(\frac{md}{|\mathbb{F}|}\right) \ ,
\end{aligned}$$

where the second inequality holds since distinct polynomials (of total degree md) on ℓ intersect in at most md points, and the last inequality holds since any univariate (line) polynomial v considered has degree at most d (polynomials with a higher degree would be immediately rejected by the verifier).

To conclude, if a m-variate polynomial Q has at least one variable in which the individual degree is larger than d, then its restriction to a random axis-parallel line will have degree larger than d with probability at least $O(\frac{1}{m} - \frac{md}{|\mathbb{F}|})$ over the choice of the line. This concludes the proof, by our assumption regarding the size of the field $\mathbb{F}$. □

8 Lifting from low-degree IPCP to MIP* while preserving zero knowledge

Recall that low-degree IPCPs are IPCP protocols in which the PCP oracle is *promised* to be a low-degree polynomial (see Section 5.2). We prove that any low-degree IPCP can be transformed into a corresponding MIP*, *while preserving zero knowledge*.

Lemma 8.1 (lifting lemma)**.** *Let $C \geq 1$ be the absolute constant in Theorem 7.2. There exists a transformation T that maps any r-round $(\mathbb{F}, d, m)$-low-degree IPCP (P', V') for a language $\mathscr{L}$, where $m, d \in \mathbb{N}$ and $\mathbb{F}$ is a finite field of size $|\mathbb{F}| > \max\{(2md)^C, 5d\mathsf{q}\}$, into a 2-prover $(\mathsf{r}^* + 2)$-round MIP* $(P_1, P_2, V) := T(P', V')$ for $\mathscr{L}$, where $\mathsf{r}^* := \max\{\mathsf{r}, 1\}$.*

Furthermore, if the IPCP (P', V') is zero knowledge with query bound $\mathsf{b} \geq 2(d+1)^2 + d\mathsf{q} + 1$ (q denotes the query complexity of the honest verifier), then the MIP (P_1, P_2, V) is zero knowledge.*

In the rest of this section we prove Lemma 8.1. Specifically, in Section 8.1 we begin with a classical preprocessing step (a query reduction); in Section 8.2 we present our transformation; in Section 8.3 we prove soundness against entangled provers; and in Section 8.4 we prove preservation of zero knowledge. The conceptual contribution of Lemma 8.1 is that it provides an abstraction of techniques in [IV12; Vid16].

Remark 8.2 (on preserving round complexity)**.** If we do not wish to preserve zero knowledge, then the round complexity of the MIP* that is obtained in Lemma 8.1 can be reduced by 1 (see discussion at the end of Section 8.2). In addition, if the original low-degree IPCP makes a single, uniformly distributed query to its PCP oracle, then the preprocessing step is not required, and we can save an additional round. In particular, if both conditions occur, we obtain an r^*-round MIP*, fully preserving round complexity.

8.1 Classical preprocessing

The preprocessing step, which is purely classical, allows us to transform any low-degree IPCP into one that makes a single uniform query, at only a small cost in parameters. Crucially, this transformation *preserves zero knowledge* (with minor deterioration in the zero knowledge query bound).

Proposition 8.3. *There exists a transformation T such that, for every $m, d \in \mathbb{N}$ and finite field $\mathbb{F}$, if (P', V') is a public-coin*[15] *$(\mathbb{F}, d, m)$-low-degree IPCP with parameters*

$$\left|\begin{array}{rl} \textit{round complexity:} & \mathsf{r} \\ \textit{PCP length:} & \mathsf{l} \\ \textit{communication complexity:} & \mathsf{c} \\ \textit{query complexity:} & \mathsf{q} \\ \textit{oracle} \in & \mathbb{F}[X_{1,\ldots,m}^{\leq d}] \\ \textit{soundness error:} & \varepsilon \end{array}\right|,$$

then $(P'', V'') := T(P', V')$ is a low-degree IPCP for $\mathscr{L}$ with parameters

$$\left|\begin{array}{rl} \textit{round complexity:} & \mathsf{r}'=\mathsf{r}+1 \\ \textit{PCP length:} & \mathsf{l}'=\mathsf{l} \\ \textit{communication complexity:} & \mathsf{c}'=\mathsf{c}+\mathrm{poly}(m,d,\mathsf{q}) \\ \textit{query complexity:} & \mathsf{q}'=1 \\ \textit{oracle} \in & \mathbb{F}[X_{1,\ldots,m}^{\leq d}] \\ \textit{soundness error:} & \varepsilon'=\varepsilon+\frac{d\mathsf{q}}{|\mathbb{F}|-\mathsf{q}} \end{array}\right|,$$

where the verifier's single query is uniformly distributed. Furthermore, if (P', V') is (perfect) zero knowledge with query bound b, then (P'', V'') is (perfect) zero knowledge with query bound $\mathsf{b} - (d\mathsf{q} + 1)$.

[15] In fact, it suffices to satisfy a weaker condition that is implied by public-coin interaction. Specifically, our transformation also works for *private-coin* IPCPs as long as the verifier queries the PCP oracle *after* the interaction with the prover terminates.

The proof of Proposition 8.3 is via a straightforward adaptation of a technique from [KR08], while keeping track of its effect on zero knowledge; we defer this proof to Appendix A.

Let $m, d \in \mathbb{N}$, and let $\mathbb{F}$ be a finite field of size $|\mathbb{F}| > (md/\varepsilon)^C$. Let (P', V') be an r-round $(\mathbb{F}, d, m)$-low-degree IPCP for a language $\mathscr{L}$. Denote its oracle by R, query complexity by q, PCP length by l, communication complexity by c, and soundness error by $\varepsilon = 1/8$.[16] If (P', V') is zero knowledge with respect to a query bound, denote this bound by b.

We apply Proposition 8.3 to (P', V') in order to obtain the $(\mathbb{F}, d, m)$-low-degree IPCP (P'', V''), with parameters as stated in the lemma above, whose verifier makes a single uniformly distributed query to its PCP oracle. In particular, note that the new zero knowledge query bound is $\mathsf{b}' \geq 2(d+1)^2$ and the soundness is $\frac{1}{8} + \frac{d\mathsf{q}}{|\mathbb{F}|-\mathsf{q}} \leq 1/4$. We then proceed to transform (P'', V'') to an MIP* in Section 8.2.

Remark 8.4 (prover-oblivious queries)**.** After the preprocessing, the verifier makes a single uniform query, which means that its queries are a random variable that is *independent* of the prover messages (but may be correlated with the verifier messages). We refer to this property as *prover-oblivious queries*.

Remark 8.5 (on adaptivity)**.** We assumed that all IPCP verifiers make non-adaptive queries to their oracle. However, we can extend all of our results, in a straightforward way, to hold with respect to verifiers that make adaptive queries, at the cost of an increase in round complexity. Specifically, by the public-coin property of our IPCP verifiers, we can assume without loss of generality that the verifier performs its queries *after* the interaction with the prover ceases. After which, the verifier can ask the prover for the evaluation of the oracle, instead of actually querying it (at the cost of an additional round of interaction per adaptive query), and then perform all queries, non-adaptively, at the end.

8.2 The transformation

Recall that (P'', V'') is an r-round IPCP protocol for a language $\mathscr{L}$ with soundness error ε, whose completeness and soundness conditions are with respect to a low-degree PCP oracle $R \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}]$ to which the verifier makes a single uniform query.

To construct an MIP* for $\mathscr{L}$, we follow the proof overview presented in Section 2. However, there is an additional complication that we need to deal with, which we discuss next.

Symmetrization and zero knowledge. Our high-level strategy for constructing a zero knowledge MIP* for $\mathscr{L}$ is to let one entangled prover simulate the PCP oracle R and the other one simulate the IPCP prover P, while using the entanglement-resistant low-degree test to assert that the prover simulating R actually answers according to a low-degree polynomial.

Recall that the analysis of the low-degree test (Theorem 7.2) requires that the provers employ *symmetric* strategies. Typically, this is handled by letting the verifier randomly choose the roles that the provers play. However, in the setting of zero knowledge MIP* such a symmetrization causes problems.

Specifically, to prove zero knowledge we need to consider *malicious* verifiers that may abuse the interaction to learn from the provers. In particular, it turns out that if the verifier asks both provers to take the role of the IPCP prover P, then the protocol may *no longer* be zero knowledge condition. Indeed, the particular zero knowledge IPCP that we construct in Part II to the end of obtaining our zero knowledge MIP* loses its zero knowledge property if the verifier is allowed to perform two parallel interactions with the prover.

We overcome this difficulty via the following (non-standard) symmetrization. First, the provers flip a coin (by performing a measurement on $|\Psi\rangle$) to decide which prover is *primary* and which is *secondary*, and send

[16] The soundness error is reduced, via standard parallel repetition, to $1/8$, since we next apply a transformation that slightly increases the soundness error, and we wish to end up with soundness error at most $1/4$ towards the MIP* transformation.

its outcome to the verifier. The secondary prover may only be assigned with the role of plane or point lookup, whereas the primary prover may also be assigned with the role of the IPCP prover. This allows the verifier to enforce that only one prover takes the role of the IPCP prover, while keeping the provers' strategy symmetric.

Below we describe how to construct an MIP* for $\mathscr{L}$.

Construction 8.6. We construct a 2-prover MIP* (P_1, P_2, V) for the language $\mathscr{L}$. The provers P_1 and P_2 share an entangled state $|\Psi\rangle$, and all three parties receive an explicit input x. The (honest) interaction takes place as follows.

1. *Symmetrization.* The provers flip a coin (by performing a measurement on $|\Psi\rangle$) to decide which prover is *primary* and which one is *secondary*; each prover then declares its decision to the verifier.[17] The primary prover may be assigned a role in $\{\mathcal{P}_{\text{main}}, \mathcal{P}_{\text{lookup}}, \mathcal{P}_{\text{plane}}\}$, and the secondary prover only in $\{\mathcal{P}_{\text{lookup}}, \mathcal{P}_{\text{plane}}\}$.
2. The verifier chooses uniformly at random between the following procedures.
   - *Low individual degree test.* The verifier V performs the low individual degree test of Construction 7.1. Recall that with probability $1/4$ the verifier sends a random point $\alpha \in \mathbb{F}^m$ to the secondary prover.
   - *IPCP emulation.*
     (a) The verifier V assigns the primary prover the role $\mathcal{P}_{\text{main}}$ and the secondary prover the role $\mathcal{P}_{\text{lookup}}$.
     (b) V asks $\mathcal{P}_{\text{lookup}}$ for an evaluation of R at a uniformly chosen point $\vec{\beta} \in \mathbb{F}^m$.
     (c) $\mathcal{P}_{\text{main}}$ and V emulate the interaction of the IPCP (P'', V''). This generates a value $c \in \mathbb{F}$ such that, with probability at least $1 - \varepsilon$, $x \in \mathscr{L}$ if and only if $R(\vec{\beta}) = c$.
     (d) $\mathcal{P}_{\text{lookup}}$ replies with an element $\tilde{z} \in \mathbb{F}$.
     (e) V accepts if and only if $c = \tilde{z}$.

The honest prover strategy in Construction 8.6 is symmetric and so we write $\mathsf{P} := P_1 = P_2$. The round complexity of the MIP*in Construction 8.6 is $\mathsf{r}^* + 2$, because the parties assign roles in the first round, then either engage in a 1-round low-degree test protocol or an $\mathsf{r}^* + 1$-round IPCP protocol.[18]

For completeness, if $x \in \mathscr{L}$, then there exists a low-degree polynomial $R \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}]$ that the IPCP verifier V'' accepts. Hence, after the primary prover is chosen, if the verifier selects the low-degree test, then both $\mathcal{P}_{\text{lookup}}$ and $\mathcal{P}_{\text{plane}}$ can simply answer according to R, and if the verifier choses the IPCP emulation, then the prover given role $\mathcal{P}_{\text{main}}$ acts according to the strategy of P'', and $\mathcal{P}_{\text{lookup}}$ acts as a lookup for R. In the case of the low-degree test, the foregoing strategy is accepted with probability 1, whereas in the IPCP emulation, the strategy inherits its completeness directly from the IPCP (P'', V'').

We next argue soundness (Section 8.3) and preservation of zero knowledge (Section 8.4).

Preserving round complexity sans zero knowledge. As mentioned in Remark 8.2, the round complexity of the MIP* in Construction 8.6 can be improved by 1. This is achieved by letting the (honest) *verifier* choose at random which prover is primary and which is secondary (replacing the first step in Construction 8.6).

While this modification may break the zero knowledge property (as it allows the verifier to engage in protocols that abuse the interaction with the prover, e.g., by allowing the verifier to set both provers as the main prover[19]), this modification has essentially no effect on the soundness analysis, which we show next.

[17] The honest provers described play an *asymmetric* strategy. This is for the sake of exposition: since the verifier considers the provers symmetrically, Claim 6.4 implies that there exists a symmetric strategy which causes the verifier to accept with probability 1.

[18] This relies on the IPCP verifier satisfying the prover-oblivious queries property.

[19] Indeed, the particular zero knowledge IPCP (shown in Part II) that we use to obtain our main result (Theorem 1 see also

8.3 Soundness analysis

We argue soundness against entangled quantum provers for the MIP* from Construction 8.6. Namely, we prove soundness with respect to a large constant soundness error $1-\varepsilon$, and then we amplify the soundness to the desired constant via parallel repetition for entangled strategies [KV11; Yue16].[20] Note that this preserves the complexities required by the conclusion of Lemma 8.1.

Let $x \notin \mathscr{L}$. Let $(\tilde{\mathsf{P}}_1, \tilde{\mathsf{P}}_2)$ be the strategy employed by the provers P_1, P_2, which uses the entangled state $|\Psi\rangle \in \mathcal{H} \otimes \mathcal{H}$. Suppose towards contradiction that the verifier accepts with probability at least $1-\varepsilon/2$, for constant ε to be determined later. We show that this implies a strategy that fools the (classical) IPCP with probability greater than $1/4$.

The verifier V with probability $1/2$ performs the low individual degree test of Construction 7.1. Let $\{A^z_\alpha\}_{z\in\mathbb{F},\alpha\in\mathbb{F}^m}$ and $\{B^z_\alpha\}_{z\in\mathbb{F},\alpha\in\mathbb{F}^m}$ be the projective measurements describing the strategies $\tilde{\mathsf{P}}_1$ and $\tilde{\mathsf{P}}_2$ in this test. Since by assumption the verifier accepts with probability at least $1-\varepsilon/2$, the low-degree test passes with probability at least $1-\varepsilon$. Moreover, since in this test the verifier V treats the provers P_1 and P_2 symmetrically, then by Claim 6.4 the low-degree test also passes with probability at least $1-\varepsilon$ when the provers employ the symmetrized strategy $(\tilde{\mathsf{P}}, \tilde{\mathsf{P}}) = \mathsf{Symm}(\tilde{\mathsf{P}}_1, \tilde{\mathsf{P}}_2)$.

Let $\{C^z_\alpha\}_{z\in\mathbb{F},\alpha\in\mathbb{F}^m}$ be the projective measurement describing the strategy $\tilde{\mathsf{P}}$ in this test. By Theorem 7.2, there exists an absolute constant $c \in [0,1]$ and a measurement $\{L^Q\}_{Q\in\mathbb{F}[X^{\leq d}_{1,\ldots,m}]}$ such that for $\delta := \varepsilon^c$ it holds that

$$\mathbb{E}_{\alpha\in\mathbb{F}^m} \sum_{Q\in\mathbb{F}[X^{\leq d}_{1,\ldots,m}]} \sum_{\substack{z\in\mathbb{F}\\ z\neq Q(\alpha)}} \langle\Psi|\, C^z_\alpha \otimes L^Q \,|\Psi\rangle \leq \delta \ , \tag{1}$$

and

$$\sum_{Q\in\mathbb{F}[X^{\leq d}_{1,\ldots,m}]} \langle\Psi|\, L^Q \otimes (\mathrm{Id} - L^Q) \,|\Psi\rangle \leq \delta \ . \tag{2}$$

Furthermore, since $\{C^z_\alpha\}_{z\in\mathbb{F},\alpha\in\mathbb{F}^m}$ corresponds to the strategy $(\tilde{\mathsf{P}}, \tilde{\mathsf{P}}) = \mathsf{Symm}(\tilde{\mathsf{P}}_1, \tilde{\mathsf{P}}_2)$, Eq. (1) implies that

$$\mathbb{E}_{\alpha\in\mathbb{F}^m} \sum_{Q} \sum_{\substack{z\in\mathbb{F}\\ z\neq Q(\alpha)}} \langle\Psi|\, A^z_\alpha \otimes L^Q \,|\Psi\rangle \leq \frac{1}{2} + \delta \ \text{ and } \ \mathbb{E}_{\alpha\in\mathbb{F}^m} \sum_{Q} \sum_{\substack{z\in\mathbb{F}\\ z\neq Q(\alpha)}} \langle\Psi|\, B^z_\alpha \otimes L^Q \,|\Psi\rangle \leq \frac{1}{2} + \delta \ . \tag{3}$$

Let $\tilde{\mathsf{P}}_{\mathrm{LD1}}$ and $\tilde{\mathsf{P}}_{\mathrm{LD2}}$ be the strategies derived from $\tilde{\mathsf{P}}_1$ and $\tilde{\mathsf{P}}_2$ by replacing the (arbitrary) measurements $\{A^z_\alpha\}_{z\in\mathbb{F},\alpha\in\mathbb{F}^m}$ and $\{B^z_\alpha\}_{z\in\mathbb{F},\alpha\in\mathbb{F}^m}$, respectively, with the (low-degree) measurement $\{L^z_\alpha\}_{z\in\mathbb{F},\alpha\in\mathbb{F}^m}$ given by

$$L^z_\alpha := \sum_{\substack{Q\in\mathbb{F}[X^{\leq d}_{1,\ldots,m}]\\ Q(\alpha)=z}} L^Q \ .$$

We now consider the IPCP emulation. Without loss of generality we designate the primary prover as P_1 and the secondary prover as P_2. With probability at least $1/2$, the verifier performs IPCP emulation; since the success probability of the malicious prover strategy $(\tilde{\mathsf{P}}_1, \tilde{\mathsf{P}}_2)$ is $1-\varepsilon/2$, this succeeds with probability at least $1-\varepsilon$. Recall that in this case P_1 is given the role of $\mathcal{P}_{\mathrm{main}}$ and P_2 the role of $\mathcal{P}_{\mathrm{lookup}}$. In this setting, P_1

Remark 8.7) was observed to lose its zero knowledge property if the verifier is allowed to perform two parallel interactions with the prover (see [BCFGRS17, Remark 5.6]).

[20] While the known parallel repetition theorems for entangled strategies are much weaker than their classical counterparts (having polynomial rather than exponential decay), this difference is immaterial in our setting.

acts under strategy $\tilde{\mathsf{P}}_1$, whereas P_2 measures according to $\{B_\alpha^z\}_{z\in\mathbb{F},\alpha\in\mathbb{F}^m}$ under strategy $\tilde{\mathsf{P}}_2$ and according to $\{L_\alpha^z\}_{z\in\mathbb{F},\alpha\in\mathbb{F}^m}$ under strategy $\tilde{\mathsf{P}}_{\text{LD2}}$

We show that the probability that V (falsely) accepts the strategy $(\tilde{\mathsf{P}}_1, \tilde{\mathsf{P}}_2)$ is close, up to an additive 2δ factor, to the probability it accepts the strategy $(\tilde{\mathsf{P}}_1, \tilde{\mathsf{P}}_{\text{LD2}})$. We describe the system via the following four registers.

1. $\mathcal{A}$ is the (classical) register wherein the message from V to $\mathcal{P}_{\text{lookup}}$ is stored.
2. $\mathcal{B}$ is the register that corresponds to the private space of $\mathcal{P}_{\text{lookup}}$.
3. $\mathcal{C}$ is the register that consists of the rest of the system (everything but $\mathcal{A}$, $\mathcal{B}$, and the ancilla).
4. $\mathcal{D}$ is the ancilla $\mathcal{P}_{\text{lookup}}$ uses to store its answers to V.

Let $\sigma \in \mathcal{H}_\mathcal{A} \otimes \mathcal{H}_\mathcal{B} \otimes \mathcal{H}_\mathcal{C} \otimes \mathcal{H}_\mathcal{D}$ be the global entangled state of the system prior to the measurement performed by $\mathcal{P}_{\text{lookup}}$, σ_B be the global state of the system after $\mathcal{P}_{\text{lookup}}$ measures according to $\{ B_\alpha^z \}_{z\in\mathbb{F},\alpha\in\mathbb{F}^m}$, and σ_T be the global state of the system after $\mathcal{P}_{\text{lookup}}$ measures according to $\{L^Q\}_{Q\in\mathbb{F}[X_{1,\dots,m}^{\le d}]}$, where after the measurements $\mathcal{P}_{\text{lookup}}$ discards the post-measurement state. Note that

$$\sigma = \mathbb{E}_{\alpha\in\mathbb{F}^m} |\alpha\rangle\langle\alpha| \otimes \sigma_\alpha^{\mathcal{B},\mathcal{C}} \ ,$$

$$\sigma_B = \mathrm{Tr}_\mathcal{B}\left[\mathbb{E}_{\alpha\in\mathbb{F}^m} |\alpha\rangle\langle\alpha| \otimes (B_\alpha^z \otimes \mathrm{Id}_\mathcal{C})\sigma_\alpha^{\mathcal{B},\mathcal{C}}(B_\alpha^z \otimes \mathrm{Id}_\mathcal{C}) \otimes \sum_{z\in\mathbb{F}} |z\rangle\langle z|\right] \ ,$$

$$\sigma_T = \mathrm{Tr}_\mathcal{B}\left[\mathbb{E}_{\alpha\in\mathbb{F}^m} |\alpha\rangle\langle\alpha| \otimes (L_\alpha^z \otimes \mathrm{Id}_\mathcal{C})\sigma_\alpha^{\mathcal{B},\mathcal{C}}(L_\alpha^z \otimes \mathrm{Id}_\mathcal{C}) \otimes \sum_{z\in\mathbb{F}} |z\rangle\langle z|\right] \ ,$$

where $\sigma_\alpha^{\mathcal{B},\mathcal{C}}$ denotes the entangled state σ after V asked the question α, restricted to the registers $\mathcal{B},\mathcal{C},\mathcal{D}$, and $\mathrm{Id}_\mathcal{C}$ denotes the identity operator over $\mathcal{H}_\mathcal{D}$.

Recall that $(\tilde{\mathsf{P}}_1, \tilde{\mathsf{P}}_2, V)(x)$ and $(\tilde{\mathsf{P}}_1, \tilde{\mathsf{P}}_{\text{LD2}}, V)(x)$ denote the random variables representing the output of the verifier V when interacting with provers employing strategies $(\tilde{\mathsf{P}}_1, \tilde{\mathsf{P}}_2)$ and $(\tilde{\mathsf{P}}_1, \tilde{\mathsf{P}}_{\text{LD2}})$, respectively, on input $x \notin \mathscr{L}$. Observe that

$$\begin{aligned}
&\left|\Pr[(\tilde{\mathsf{P}}_1, \tilde{\mathsf{P}}_2, V)(x) = 1] - \Pr[(\tilde{\mathsf{P}}_1, \tilde{\mathsf{P}}_{\text{LD2}}, V)(x) = 1]\right| \\
&\le \frac{1}{2}\left\|\sigma_B - \sigma_T\right\|_1 \\
&= \frac{1}{2}\left\|\mathrm{Tr}_\mathcal{B}\mathbb{E}_{\alpha\in\mathbb{F}^m} |\alpha\rangle\langle\alpha| \otimes \left((B_\alpha^z \otimes \mathrm{Id}_\mathcal{C})\sigma_\alpha^{\mathcal{B},\mathcal{C}}(B_\alpha^z \otimes \mathrm{Id}_\mathcal{C}) - (L_\alpha^z \otimes \mathrm{Id}_\mathcal{C})\sigma_\alpha^{\mathcal{B},\mathcal{C}}(L_\alpha^z \otimes \mathrm{Id}_\mathcal{C})\right) \otimes \sum_{z\in\mathbb{F}} |z\rangle\langle z|\right\|_1 \ , \\
&\le \frac{1}{2}\mathbb{E}_{\alpha\in\mathbb{F}^m}\left\|\sum_{z\in\mathbb{F}} \left((B_\alpha^z \otimes \mathrm{Id}_\mathcal{C})\sigma_\alpha^{\mathcal{B},\mathcal{C}}(B_\alpha^z \otimes \mathrm{Id}_\mathcal{C}) - (L_\alpha^z \otimes \mathrm{Id}_\mathcal{C})\sigma_\alpha^{\mathcal{B},\mathcal{C}}(L_\alpha^z \otimes \mathrm{Id}_\mathcal{C})\right) \otimes |z\rangle\langle z|\right\|_1 \\
&\le \mathbb{E}_{\alpha\in\mathbb{F}^m}\sqrt{\sum_{z\in\mathbb{F}} \mathrm{Tr}\left((B_\alpha^z - L_\alpha^z)\rho(B_\alpha^z - L_\alpha^z)^\dagger\right)} \qquad \text{(by Lemma 6.2)} \\
&\le \sqrt{\mathbb{E}_{\alpha\in\mathbb{F}^m}\sum_{z\in\mathbb{F}} \mathrm{Tr}_\rho\left((B_\alpha^z - L_\alpha^z)^2\right)} \qquad \text{(by Jensen's inequality)} \\
&\le \frac{1}{2} + \delta \qquad \text{(by Eq. (2) and Claim 6.1).}
\end{aligned}$$

By the triangle inequality, the total success probability of $(\tilde{\mathsf{P}}_1, \tilde{\mathsf{P}}_{\mathrm{LD2}})$ is at least $1 - \varepsilon - \frac{1}{2} - \delta > \frac{1}{2} - 2\delta$. To conclude the proof, choose ε to be a sufficiently small constant such that $\delta < 1/8$, and note that when using strategy $(\tilde{\mathsf{P}}_1, \tilde{\mathsf{P}}_{\mathrm{LD2}})$ the prover $\mathcal{P}_{\text{lookup}}$ can measure the prior entanglement obliviously to its question, and so the strategy can be realize merely via shared randomness.

Therefore we can construct a malicious prover $\tilde{P}''$ that will fool the $(\mathbb{F}, d, m)$-low-degree IPCP verifier V'' for $\mathscr{L}$ (which we started from) with probability at least $\frac{1}{2} - 2\delta > 1/4$, by implementing the strategy $\tilde{\mathsf{P}}''$ in the natural way. That is, $\tilde{P}''$ samples some $Q \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}]$ according to the distribution induced by $\{L^Q\}$ and $|\Psi\rangle$ and sends Q as the oracle. It then interacts with V according to the strategy $\tilde{\mathsf{P}}_1$.

8.4 Preserving zero knowledge

We argue that Construction 8.6 preserves zero knowledge. Suppose that the IPCP (P'', V'') is zero knowledge with query bound $\mathsf{b}' \geq 2(d+1)^2$, and let S'' be the corresponding simulator. We explain how to construct a simulator S for the MIP* (P_1, P_2, V).

Given a malicious verifier $\tilde{V}$ for the MIP*, we design a "malicious" verifier $\tilde{V}''$ for the IPCP protocol that, when interacting with the IPCP prover P'', outputs the view of the malicious MIP* verifier $\tilde{V}$ when interacting with the (honest) MIP* provers P_1, P_2. The simulator S is then given by running $(S'')^{\tilde{V}''}$, and returning the output of $\tilde{V}''$. We first describe the operation of $\tilde{V}''$.

1. Begin simulating $\tilde{V}$.
2. Flip a coin, and send the outcome to $\tilde{V}$. Receive from $\tilde{V}$ the role assignments for the provers; if the secondary prover is assigned to be $\mathcal{P}_{\text{main}}$, we simulate as if it has aborted.
3. The remainder of the simulation is divided up with respect to prover role.
   (a) Every message $\tilde{V}$ sends to the prover assigned to be $\mathcal{P}_{\text{main}}$, if any, is forwarded to P'', and the responses of P'' are forwarded to $\tilde{V}$.
   (b) If any prover is assigned the role of $\mathcal{P}_{\text{lookup}}$, then if $\tilde{V}$ sends a query point $\alpha \in \mathbb{F}^m$ to this prover, query the oracle at α, and send the answer to $\tilde{V}$; if $\tilde{V}$ sends an axis-parallel line ℓ to this prover, query the oracle at $(d+1)$ points on ℓ in order to interpolate $R \circ \ell$, and send this polynomial to $\tilde{V}$.
   (c) If any prover is assigned the role of $\mathcal{P}_{\text{plane}}$, then if $\tilde{V}$ sends a query plane $s \in \mathrm{Planes}(\mathbb{F}^m)$ to this prover, query the oracle at a set of points sufficient to interpolate $R \circ \ell$ (of size at most $(d+1)^2$), and send this polynomial to $\tilde{V}$.
4. Output the view of the simulated $\tilde{V}$.

It is clear that the output of $\tilde{V}''$ when interacting with P'' is exactly the view of $\tilde{V}$ in the MIP*. The number of queries $\tilde{V}''$ makes is at most $(d+1)^2 + d + 1 \leq 2(d+1)^2$. By the zero knowledge guarantee for S'', provided $\mathsf{b}' \geq 2(d+1)^2$, the view of $\tilde{V}''$ is perfectly simulated, and so in particular its output in simulation is identically distributed to the output of $\tilde{V}''$ when interacting with P''.

Remark 8.7. Observe that if it were possible for the verifier to assign both provers to be $\mathcal{P}_{\text{main}}$, as is the case with the standard symmetrization, then the above argument would not go through. The reason is that the two interactions may be correlated in some way that we cannot simulate.

9 Zero knowledge MIP* for nondeterministic exponential time

Recall that our plan is to prove Theorem 1 in two steps: (1) construct a zero knowledge low-degree IPCP for any language in $\mathbf{NEXP}$; (2) invoke the lifting lemma (Lemma 8.1) on this low-degree IPCP in order to obtain a zero knowledge MIP* for $\mathbf{NEXP}$. So far, in Part I, we have obtained the tools for deriving a zero knowledge MIP* from a zero knowledge low-degree IPCP. The goal of Part II is to construct such a zero knowledge low-degree IPCP for any language in $\mathbf{NEXP}$; that is, in Part II we prove the following theorem.

Theorem 9.1 (concisely stated; see Theorem 12.2 for the full statement)**.** *There exists a constant $c \in \mathbb{N}$ such that for every query bound function b and language $\mathscr{L} \in \mathbf{NEXP}$ the following holds. There exists an $(\mathbb{F}, d, m)$-low-degree IPCP for $\mathscr{L}$, where $d, m = O(n^c \log \mathsf{b})$ and $\mathbb{F}$ is a field with $|\mathbb{F}| = \Omega((n^c \log \mathsf{b})^4)$, that is perfect zero knowledge against all b-query malicious verifiers and has the following parameters:*

round complexity:	$\mathrm{poly}(n)+O(\log \mathsf{b})$
PCP length:	$\mathrm{poly}(2^n, \mathsf{b})$
communication complexity:	$\mathrm{poly}(n, \log(\mathsf{b}))$
query complexity:	$\mathrm{poly}(n, \log(\mathsf{b}))$
oracle $\in$	$\mathbb{F}[X_{1,\ldots,m}^{\leq d}]$
soundness error:	$1/2$

In this section we prove Theorem 1 by taking the zero knowledge low-degree IPCP in Theorem 9.1 and lifting it via Lemma 8.1 to obtain a 2-prover zero knowledge MIP*, concluding the proof of Theorem 1.

Let $\mathscr{L}$ be a language in $\mathbf{NEXP}$, and let (P', V') be the $(\mathbb{F}, d, m)$-low-degree IPCP for $\mathscr{L}$ implied by Theorem 9.1, with respect to query bound b, $d, m = O(n^c \log \mathsf{b})$, and a finite field $\mathbb{F}$ of size $|\mathbb{F}| = \mathrm{poly}(n)$ such that $|\mathbb{F}| > \max\{(2md)^C, 5d\mathsf{q}, \Omega((n^c \log \mathsf{b})^4)\}$ (where C is the constant from Theorem 7.2).

The $(\mathbb{F}, d, m)$-low-degree IPCP (P', V') satisfies the conditions of the lifting lemma (Lemma 8.1), and thus we can apply the transformation T in Lemma 8.1 to the low-degree IPCP (P', V') to obtain a perfect zero knowledge 2-prover MIP* $(P_1, P_2, V) := T(P', V')$ for $\mathscr{L}$, with round complexity $\mathrm{poly}(n) + O(\log \mathsf{b}) = \mathrm{poly}(n)$, communication complexity $\mathrm{poly}(n, \log(\mathsf{b}), d, \mathsf{q}, m) = \mathrm{poly}(n)$, and soundness error $1/2$. This concludes the proof of our main result, Theorem 1.

Remark 9.2 (zero knowledge MIP* for $\#\mathbf{P}$ via known IPCPs)**.** As mentioned in Section 2.3.3, a recent work in algebraic zero knowledge [BCFGRS17] (building on techniques from [BCGV16]) obtains a zero knowledge low-degree IPCP for any language in $\#\mathbf{P}$. By replacing our IPCP for $\mathbf{NEXP}$ in Theorem 9.1 with their IPCP for $\#\mathbf{P}$, we can derive a zero knowledge MIP*, albeit only for languages in $\#\mathbf{P}$.

Part II
Low-degree IPCP with zero knowledge

The purpose of this part is to show that there exists a perfect zero knowledge low-degree IPCP for any language in $\mathbf{NEXP}$, which is the remaining step in our construction of perfect zero knowledge MIP* protocols for $\mathbf{NEXP}$ (as discussed in Section 9). To this end we build on advances in algebraic zero knowledge [BCFGRS17] and ideas from algebraic complexity theory, and we develop new techniques for obtaining algebraic zero knowledge.

Organization. We begin in Section 10, where we show new algebraic query complexity lower bounds on polynomial summation. Then, in Section 11 we construct our strong zero knowledge sumcheck protocol, whose analysis relies on the foregoing algebraic query complexity lower bounds. Finally, in Section 12 we use our strong zero knowledge sumcheck protocol to show a perfect zero knowledge low-degree IPCP for any language in $\mathbf{NEXP}$.

10 Algebraic query complexity of polynomial summation

We have outlined in Section 2.4.2 an algebraic commitment scheme based on the sumcheck protocol and lower bounds on the algebraic query complexity of polynomial summation. The purpose of this section is to describe this construction in more detail, and then provide formal statements for the necessary lower bounds.

The setting: algebraic commitment schemes. We begin with the case of committing to a single element $a \in \mathbb{F}$. The prover chooses a uniformly random string $B \in \mathbb{F}^N$ such that $\sum_{i=1}^{N} B_i = a$, for some $N \in \mathbb{N}$. Fixing some $d \in \mathbb{N}$, $G \subseteq \mathbb{F}$ and $k \in \mathbb{N}$ such that $|G| \leq d+1$ and $|G|^k = N$, the prover views B as a function from G^k to $\mathbb{F}$ (via an arbitrary ordering on G^k) and sends the evaluation of a degree-d extension $\hat{B} \colon \mathbb{F}^k \to \mathbb{F}$ of B, chosen uniformly at random from all such extensions. The verifier can test that $\hat{B}$ is indeed (close to) a low-degree polynomial but (ideally) cannot learn any information about a without reading *all* of B (i.e., without making N queries). Subsequently, the prover can decommit to a by convincing the verifier that $\sum_{\vec{\beta} \in G^k} \hat{B}(\vec{\beta}) = a$ via the sumcheck protocol.

To show that the above is a commitment scheme, we must show both *binding* and *hiding*. Both properties depend on the choice of d. The binding property follows from the soundness of the sumcheck protocol, and we thus would like the degree d of $\hat{B}$ to be as small as possible. A natural choice would be $d = 1$ (so $|G| = 2$), which makes $\hat{B}$ the unique multilinear extension of B. However (as discussed in Section 2.4.2) this choice of parameters does not provide any hiding: it holds that $\sum_{\beta \in \{0,1\}^k} B(\beta) = \hat{B}(2^{-1}, \ldots, 2^{-1}) \cdot 2^k$ (as long as $\mathrm{char}(\mathbb{F}) \neq 2$). We therefore need to understand how the choice of d affects the number of queries to $\hat{B}$ required to compute a. This is precisely the setting of *algebraic query complexity*, which we discuss next.

The algebraic query complexity (defined in [AW09] to study "algebrization") of a function f is the (worst-case) number of queries to some low-degree extension $\hat{B}$ of a string B required to compute $f(B)$. This quantity is bounded from above by the standard query complexity of f, but it may be the case (as above) that the low-degree extension confers additional information that helps in computing f with fewer queries. The usefulness of this information depends on the parameters d and G of the low-degree extension. Our question amounts to understanding this dependence for the function $\textsc{Sum} \colon \mathbb{F}^N \to \mathbb{F}$ given by $\textsc{Sum}(B) := \sum_{i=1}^{N} B_i$. It is known that if $G = \{0, 1\}$ and $d = 2$ then the algebraic query complexity of $\textsc{Sum}$ is exactly N [JKRS09].

For our purposes, however, it is not enough to commit to a single field element. Rather, we need to commit to the evaluation of a polynomial $Q \colon \mathbb{F}^m \to \mathbb{F}$ of degree d_Q, which we do as follows. Let K be a subset of $\mathbb{F}$ of size $d_Q + 1$. The prover samples, for each $\vec{\alpha} \in K^m$, a random string $B^{\vec{\alpha}} \in \mathbb{F}^N$ such that

$\textsc{Sum}(B^{\vec{\alpha}}) = Q(\vec{\alpha})$. The prover views these strings as a function $B\colon K^m \times G^k \to \mathbb{F}$, and takes a low-degree extension $\hat{B}\colon \mathbb{F}^m \times \mathbb{F}^k \to \mathbb{F}$. The polynomial $\hat{B}(\vec{X}, \vec{Y})$ has degree d_Q in $\vec{X}$ and d in $\vec{Y}$; this is a commitment to Q because $\sum_{\vec{\beta} \in G^k} \hat{B}(\vec{X}, \vec{\beta})$ is a degree-d_Q polynomial that agrees with Q on K^m, and hence equals Q.

Once again we will decommit to $Q(\vec{\alpha})$ using the sumcheck protocol, and so for binding we need d to be small. For hiding, as in the single-element case, if d is too small, then a few queries to $\hat{B}$ can yield information about Q. Moreover, it could be the case that the verifier can leverage the fact that $\hat{B}$ is a *joint* low-degree extension to learn some linear combination of evaluations of Q. We must exclude these possibilities in order to obtain our zero knowledge guarantees.

New algebraic query complexity lower bounds. The foregoing question amounts to a generalization of algebraic query complexity where, given a list of strings $B_1, \ldots, B_M$, we determine how many queries we need to make to their *joint* low-degree extension $\hat{B}$ to determine any nontrivial linear combination $\sum_{i=1}^{M} c_i \cdot \textsc{Sum}(B_i)$. We will show that the "generalized" algebraic query complexity of $\textsc{Sum}$ is exactly N, provided $d \geq 2(|G| - 1)$ (which is also the case for the standard algebraic query complexity).

In the remainder of the section we state our results in a form equivalent to the above, which is more useful to us. Denote by $\mathbb{F}[X_{1,\ldots,m}^{\leq d}, Y_{1,\ldots,k}^{\leq d'}]$ the set of all $(m+k)$-variate polynomials of individual degree d in the variables $X_1, \ldots, X_m$ and individual degree d' in the variables $Y_1, \ldots, Y_k$. Given an arbitrary polynomial $Z \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}, Y_{1,\ldots,k}^{\leq d'}]$, we ask how many queries are required to determine any nontrivial linear combination of $\sum_{\vec{y} \in G^k} Z(\vec{\alpha}, \vec{y})$ for $\vec{\alpha} \in \mathbb{F}^m$. The following lemma is more general: it states that not only do we require many queries to determine *any* linear combination, but that the number of queries grows linearly with the number of independent combinations that we wish to learn.

Lemma 10.1 (algebraic query complexity of polynomial summation)**.** *Let $\mathbb{F}$ be a field, $m, k, d, d' \in \mathbb{N}$, and G, K, L be finite subsets of $\mathbb{F}$ such that $K \subseteq L$, $d' \geq |G| - 2$, and $|K| = d + 1$. If $S \subseteq \mathbb{F}^{m+k}$ is such that there exist matrices $C \in \mathbb{F}^{L^m \times \ell}$ and $D \in \mathbb{F}^{S \times \ell}$ such that for all $Z \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}, Y_{1,\ldots,k}^{\leq d'}]$ and all $i \in \{1, \ldots, \ell\}$*

$$\sum_{\vec{\alpha} \in L^m} C_{\vec{\alpha},i} \sum_{\vec{y} \in G^k} Z(\vec{\alpha}, \vec{y}) = \sum_{\vec{q} \in S} D_{\vec{q},i} Z(\vec{q}) \ ,$$

then $|S| \geq \operatorname{rank}(BC) \cdot (\min\{d' - |G| + 2, |G|\})^k$, where $B \in \mathbb{F}^{K^m \times L^m}$ is such that column $\vec{\alpha}$ of B represents $Z(\vec{\alpha})$ in the basis $(Z(\vec{\beta}))_{\vec{\beta} \in K^m}$.

We remark that in Appendix C we prove upper bounds showing that, in some cases, Lemma 10.1 is tight.

Proof of Lemma 10.1. We use a rank argument. First, since Z has individual degree at most d in $\vec{X}$, we can rewrite any such linear combination in the following way:

$$\sum_{\vec{\alpha} \in L^m} C_{\vec{\alpha},i} \sum_{\vec{y} \in G^k} Z(\vec{\alpha}, \vec{y}) = \sum_{\vec{\alpha} \in L^m} C_{\vec{\alpha},i} \sum_{\vec{\beta} \in K^m} b_{\vec{\beta},\vec{\alpha}} \sum_{\vec{y} \in G^k} Z(\vec{\alpha}, \vec{y}) = \sum_{\vec{\alpha} \in K^m} C'_{\vec{\alpha},i} \sum_{\vec{y} \in G^k} Z(\vec{\alpha}, \vec{y}) = \sum_{\vec{q} \in S} D_{\vec{q},i} Z(\vec{q}) \ ,$$

where $C' := BC$. If $d' = |G| - 2$, then the bound is trivial. Otherwise, let H be some arbitrary subset of G of size $\min\{d' - |G| + 2, |G|\}$. Let $P_0 \subseteq \mathbb{F}[X_{1,\ldots,m}^{\leq d}, Y_{1,\ldots,k}^{\leq |H|-1}]$ be such that for all $p \in P_0$ and for all $\vec{q} \in S$, $p(\vec{q}) = 0$. Since these are at most S linear constraints, P_0 has dimension at least $(d+1)^m |H|^k - |S|$.

Let $B_0 \in \mathbb{F}^{n \times (d+1)^m |H|^k}$ be a matrix whose rows form a basis for the vector space $\{\big(p(\vec{\alpha}, \vec{\beta})\big)_{\vec{\alpha} \in K^m, \vec{\beta} \in H^k} : p \in P_0\}$ of evaluations of polynomials in P_0 on $K^m \times H^k$; we have $n \geq (d+1)^m |H|^k - |S|$. By an averaging argument there exists $\vec{\beta}_0 \in H^k$ such that the submatrix $B_{\vec{\beta}_0}$ consisting of columns $(\vec{\alpha}, \vec{\beta}_0)$ of B_0 for each $\vec{\alpha} \in K^m$ has rank at least $(d+1)^m - |S|/|H|^k$.

Let $q \in \mathbb{F}[Y_{1,\ldots,k}^{\leq |G|-1}]$ be the polynomial such that $q(\vec{\beta}_0) = 1$, and $q(\vec{y}) = 0$ for all $\vec{y} \in G^k - \{\vec{\beta}_0\}$. For arbitrary $p \in P_0$, let $Z(\vec{X}, \vec{Y}) := q(\vec{Y})p(\vec{X}, \vec{Y}) \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}, Y_{1,\ldots,k}^{\leq |H|+|G|-2}]$. Observe that our choice of H ensures that the degree of Z in $\vec{Y}$ is at most d'. Then for all $i \in \{1, \ldots, \ell\}$, it holds that

$$\sum_{\vec{\alpha} \in K^m} C'_{\vec{\alpha},i} \sum_{\vec{y} \in G^k} Z(\vec{\alpha}, \vec{y}) = \sum_{\vec{\alpha} \in K^m} C'_{\vec{\alpha},i} \cdot p(\vec{\alpha}, \vec{\beta}_0) = \sum_{\vec{q} \in S} D_{\vec{q},i} \cdot Z(\vec{\alpha}, \vec{y}) = 0 \ .$$

Thus the column space of C' is contained in the null space of $B_{\vec{\beta}_0}$, and so the null space of $B_{\vec{\beta}_0}$ has rank at least $\mathrm{rank}(C')$. Hence $(d+1)^m - \mathrm{rank}(C') \geq \mathrm{rank}(B_{\vec{\beta}_0}) \geq (d+1)^m - |S|/|H|^k$, so $|S| \geq \mathrm{rank}(C') \cdot |H|^k$, which yields the theorem. □

Implications. We state below special cases of Lemma 10.1 that suffice for our zero knowledge applications.

Corollary 10.2. *Let $\mathbb{F}$ be a finite field, G be a subset of $\mathbb{F}$, and $d, d' \in \mathbb{N}$ with $d' \geq 2(|G| - 1)$. If $S \subseteq \mathbb{F}^{m+k}$ is such that there exist $(c_{\vec{\alpha}})_{\vec{\alpha} \in \mathbb{F}^m}$ and $(d_{\vec{\beta}})_{\vec{\beta} \in \mathbb{F}^{m+k}}$ such that*

- *for all $Z \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}, Y_{1,\ldots,k}^{\leq d'}]$ it holds that $\sum_{\vec{\alpha} \in \mathbb{F}^m} c_{\vec{\alpha}} \sum_{\vec{y} \in G^k} Z(\vec{\alpha}, \vec{y}) = \sum_{\vec{q} \in S} d_{\vec{q}} Z(\vec{q})$,and*
- *there exists $Z' \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}, Y_{1,\ldots,k}^{\leq d'}]$ such that $\sum_{\vec{\alpha} \in \mathbb{F}^m} c_{\vec{\alpha}} \sum_{\vec{y} \in G^k} Z'(\vec{\alpha}, \vec{y}) \neq 0$,*

then $|S| \geq |G|^k$.

Next, we give an equivalent formulation of Corollary 10.2 in terms of random variables that we use in later sections. (Essentially, the linear structure of the problem implies that "worst-case" statements are equivalent to "average-case" statements.)

Corollary 10.3 (equivalent statement of Corollary 10.2)**.** *Let $\mathbb{F}$ be a finite field, G be a subset of $\mathbb{F}$, and $d, d' \in \mathbb{N}$ with $d' \geq 2(|G| - 1)$. Let Q be a subset of $\mathbb{F}^{m+k}$ with $|Q| < |G|^k$, and let Z be uniformly random in $\mathbb{F}[X_{1,\ldots,m}^{\leq d}, Y_{1,\ldots,k}^{\leq d'}]$. Then, the ensembles $\left(\sum_{\vec{y} \in G^k} Z(\vec{\alpha}, \vec{y})\right)_{\vec{\alpha} \in \mathbb{F}^m}$ and $\left(Z(\vec{q})\right)_{\vec{q} \in Q}$ are independent.*

Proof of Corollary 10.3. We will need a simple fact from linear algebra: that "linear independence equals statistical independence". That is, if we sample an element from a vector space and examine some subsets of its entries, these distributions are independent if and only if there does not exist a linear dependence between the induced subspaces. The formal statement of the claim is as follows.

Claim 10.4. *Let $\mathbb{F}$ be a finite field and D a finite set. Let $V \subseteq \mathbb{F}^D$ be an $\mathbb{F}$-vector space, and let $\vec{v}$ be a random variable that is uniform over V. For any subdomains $S, S' \subseteq D$, the restrictions $\vec{v}|_S$ and $\vec{v}|_{S'}$ are statistically dependent if and only if there exist constants $(c_i)_{i \in S}$ and $(d_i)_{i \in S'}$ such that:*

- *there exists $\vec{w} \in V$ such that $\sum_{i \in S} c_i w_i \neq 0$, and*
- *for all $\vec{w} \in V$, $\sum_{i \in S} c_i w_i = \sum_{i \in S'} d_i w_i$.*

Proof of Claim 10.4. For arbitrary $\vec{x} \in \mathbb{F}^S, \vec{x}' \in \mathbb{F}^{S'}$, we define the quantity

$$p_{\vec{x},\vec{x}'} := \Pr_{\vec{v} \in V}\left[\vec{v}|_S = \vec{x} \wedge \vec{v}|_{S'} = \vec{x}'\right] \ .$$

Let $d := \dim(V)$, and let $B \in \mathbb{F}^{D \times d}$ be a basis for V. Let $B_S \in \mathbb{F}^{S \times d}$ be B restricted to rows corresponding to elements of S, and let $B_{S'}$ be defined likewise. Finally, let $B_{S,S'} \in \mathbb{F}^{(|S|+|S'|) \times d}$ be the matrix whose rows are the rows of B_S, followed by the rows of $B_{S'}$. Then

$$p_{\vec{x},\vec{x}'} = \Pr_{\vec{z} \in \mathbb{F}^d}\left[B_{S,S'} \cdot \vec{z} = (\vec{x}, \vec{x}')\right] \ .$$

One can verify that, for any matrix $A \in \mathbb{F}^{m \times n}$,

$$\Pr_{\vec{z} \in \mathbb{F}^n}[A\vec{z} = \vec{b}] = \begin{cases} \mathbb{F}^{-\operatorname{rank}(A)} & \text{if } \vec{b} \in \operatorname{colsp}(A), \text{ and} \\ 0 & \text{otherwise.} \end{cases}$$

Observe that $\operatorname{colsp}(B_{S,S'}) \subseteq \operatorname{colsp}(B_S) \times \operatorname{colsp}(B_{S'})$, and equality holds if and only if $\operatorname{rank}(B_{S,S'}) = \operatorname{rank}(B_S) + \operatorname{rank}(B_{S'})$. It follows that $p_{\vec{x},\vec{x}'} = \Pr_{\vec{v} \in V}[\vec{v}|_S = \vec{x}] \cdot \Pr_{\vec{v} \in V}[\vec{v}|_{S'} = \vec{x}']$ if and only if $\operatorname{rank}(B_{S,S'}) = \operatorname{rank}(B_S) + \operatorname{rank}(B_{S'})$. By the rank-nullity theorem and the construction of $B_{S,S'}$, this latter condition holds if and only if $\operatorname{nul}(B_{S,S'}^T) \subseteq \operatorname{nul}(B_S^T) \times \operatorname{nul}(B_{S'}^T)$. To conclude the proof, it remains only to observe that the condition in the claim is equivalent to the existence of vectors $\vec{c} \in \mathbb{F}^S$, $\vec{d} \in \mathbb{F}^{S'}$ such that $\vec{c} \notin \operatorname{nul}(B_S^T)$ but $(\vec{c}, -\vec{d}) \in \operatorname{nul}(B_{S,S'}^T)$. □

Now, observe that

$$\left\{ \left(\left(Z(\vec{\gamma})\right)_{\vec{\gamma} \in \mathbb{F}^{m+k}}, \left(\sum_{\vec{y} \in G^k} Z(\vec{\alpha}, \vec{y}) \right)_{\vec{\alpha} \in \mathbb{F}^m} \right) : Z \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}, Y_{1,\ldots,k}^{\leq d'}] \right\}$$

is an $\mathbb{F}$-vector space with domain $\mathbb{F}^{m+k} \cup \mathbb{F}^m$. Consider subdomains $\mathbb{F}^m$ and S. Since $|S| < |G|^k$, by Lemma 10.1 there exist no constants $(c_{\vec{\alpha}})_{\alpha \in \mathbb{F}^m}$, $(d_{\vec{\gamma}})_{\vec{\gamma} \in S}$ such that the conditions of the claim hold. This concludes the proof of Corollary 10.3 □

11 Zero knowledge sumcheck from algebraic query lower bounds

We leverage our lower bounds on the algebraic query complexity of polynomial summation (Section 10) to obtain an analogue of the sumcheck protocol with a strong zero knowledge guarantee, which we then use to obtain a zero knowledge low-degree IPCP for $\mathbf{NEXP}$ (Section 12).

The sumcheck protocol [LFKN92] is an Interactive Proof for claims of the form $\sum_{\vec{x} \in H^m} F(\vec{x}) = a$, where H is a subset of a finite field $\mathbb{F}$, F is an m-variate polynomial over $\mathbb{F}$ of individual degree at most d, and a is an element of $\mathbb{F}$. The sumcheck protocol is *not* zero knowledge (unless $\#\mathbf{P} \subseteq \mathbf{BPP}$).

Prior work [BCFGRS17] obtains a sumcheck protocol, in the IPCP model, with a certain (weak) zero knowledge guarantee. In that protocol, the prover first sends a proof oracle that consists of the evaluation of a random m-variate polynomial R of individual degree at most d; after that, the prover and the verifier run the (standard) sumcheck protocol on a new polynomial obtained from F and R. The purpose of R is to "mask" the partial sums, which are the intermediate values sent by the prover during the sumcheck protocol.

The zero knowledge guarantee in [BCFGRS17] is the following: *any verifier that makes q queries to R learns at most q evaluations of F*. This guarantee suffices to obtain a zero knowledge protocol for $\#\mathbf{P}$ (the application in [BCFGRS17]), because the verifier can evaluate F efficiently at any point (as F is merely an arithmetization of a 3SAT formula).

We achieve a much stronger guarantee: *any verifier that makes polynomially-many queries to R learns at most a single evaluation of F* (that, moreover, lies within a chosen subset I^m of $\mathbb{F}^m$). Our application requires this guarantee because we use the sumcheck simulator as a sub-simulator in a larger protocol, where F is a randomized low-degree extension of some function that is hard to compute for the verifier. The randomization introduces bounded independence, which makes a small number of queries easy to simulate (where "small" means somewhat less than the degree).

The main idea to achieve the above zero knowledge guarantee is the following. Rather than sending the masking polynomial R directly, the prover sends a (perfectly-hiding and statistically-binding) commitment to it in the form of a random $(m+k)$-variate polynomial Z. The "real" mask is recovered by summing out k variables: $R(\vec{X}) := \sum_{\vec{\beta} \in G^k} Z(\vec{X}, \vec{\beta})$. Our lower bounds on the algebraic query complexity of polynomial summation (Section 10) imply that any q queries to Z, with $q < |G|^k$, yield *no information* about R. The prover, however, can elect to decommit to $R(\vec{c})$, for a single point $\vec{c} \in I^m$ chosen by the verifier. This is achieved using the weak zero knowledge sumcheck protocol in [BCFGRS17] as a subroutine: the prover sends $w := R(\vec{c})$ and then proves that $w = \sum_{\vec{\beta} \in G^k} Z(\vec{c}, \vec{\beta})$.

The protocol thus proceeds as follows. Given a security parameter $\lambda \in \mathbb{N}$, the prover sends the evaluations of two polynomials $Z \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}, Y_{1,\ldots,k}^{\leq 2\lambda}]$ and $A \in \mathbb{F}[Y_{1,\ldots,k}^{\leq 2\lambda}]$ as proof oracles (A is the mask for the subroutine in [BCFGRS17]). The prover sends a field element z, which is (allegedly) the summation of Z over $H^m \times G^k$. The verifier replies with a random challenge $\rho \in \mathbb{F} \setminus \{0\}$. The prover and the verifier then engage in the standard (not zero knowledge) sumcheck protocol on the claim "$\sum_{\vec{\alpha} \in H^m} \rho F(\vec{\alpha}) + R(\vec{\alpha}) = \rho a + z$". This reduces checking the correctness of this claim to checking a claim of the form "$\rho F(\vec{c}) + R(\vec{c}) = b$", for some $\vec{c} \in I^m$ and $b \in \mathbb{F}$; the prover then decommits to $w := R(\vec{c})$ as above. In sum, the verifier deduces that, with high probability, the claim "$\rho F(\vec{c}) = b - w$" is true if and only if the original claim was.

If the verifier could evaluate F, then the verifier could simply check the aforementioned claim and either accept or reject. However, we do not give the verifier access to F and, instead, we follow [Mei13; GKR15] and phrase sumcheck as a *reduction* from a claim about a sum of a polynomial over a large product space to a claim about the evaluation of that polynomial at a single point. This view of the sumcheck protocol is useful later on when designing more complex protocols, which employ sumcheck as a sub-protocol. The completeness and soundness definitions, which we will formally define in Section 11.3, are thus modified

according to this viewpoint, where the verifier does *not* have access to F and simply outputs the claim at the end of the protocol.

We state a simplified version of the main theorem of this section; the full version is given as Theorem 11.5.

Theorem 11.1. *For every finite field $\mathbb{F}$ and $d, k, \lambda \in \mathbb{N}$, $2\lambda \leq d$, there exists an $(\mathbb{F}, d, m+k+1)$-low-degree IPCP system (P, V), for the sumcheck problem with respect to polynomials in $\mathbb{F}[X^{\leq d}_{1,\ldots,m}]$, which is zero knowledge against $\lambda^k - 1$ queries, where the simulator makes a single query to the summand polynomial.*

We prove Theorem 11.1 in the next subsections, by showing and analyzing a construction that implements the ideas we outlined above. We begin by stating the required preliminaries regarding sampling partial sums of random low-degree polynomials.

11.1 Sampling partial sums of random low-degree polynomials

We recall an algorithm due to Ben-Sasson et al. [BCFGRS17] for adaptively sampling random low-degree multivariate polynomials from spaces with exponentially large dimension.

Let $\mathbb{F}$ be a finite field, m, d positive integers, and H a subset of $\mathbb{F}$. Recall that $\mathbb{F}[X^{\leq d}_{1,\ldots,m}]$ is the subspace of $\mathbb{F}[X_{1,\ldots,m}]$ consisting of those polynomials with individual degrees at most d. We denote by $\mathbb{F}^{\leq m}$ the set of all vectors over $\mathbb{F}$ of length at most m. Given $Q \in \mathbb{F}[X^{\leq d}_{1,\ldots,m}]$ and $\vec{\alpha} \in \mathbb{F}^{\leq m}$, we define $Q(\vec{\alpha}) := \sum_{\vec{\gamma} \in H^{m-|\vec{\alpha}|}} Q(\vec{\alpha}, \vec{\gamma})$; that is, the answer to a query that specifies only a prefix of the variables is the sum of the values obtained by letting the remaining variables range over H.

In Section 11 we rely on the fact, formally stated below and proved in [BCFGRS17], that one can efficiently sample the distribution $R(\vec{\alpha})$, where R is uniformly random in $\mathbb{F}[X^{\leq d}_{1,\ldots,m}]$ and $\vec{\alpha} \in \mathbb{F}^{\leq m}$ is fixed, *even conditioned on any polynomial number of (consistent) values for $R(\vec{\alpha}_1), \ldots, R(\vec{\alpha}_\ell)$*, for any choice of $\vec{\alpha}_1, \ldots, \vec{\alpha}_\ell \in \mathbb{F}^{\leq m}$. More precisely, the sampling algorithm runs in time that is only $\mathrm{poly}(\log|\mathbb{F}|, m, d, |H|, \ell)$, which is much faster than the trivial running time of $\Omega(d^m)$ achieved by sampling R explicitly. This "succinct" sampling follows from the notion of *succinct constraint detection* studied in [BCFGRS17] for the case of partial sums of low-degree polynomials.

Lemma 11.2 ([BCFGRS17])**.** *There exists a probabilistic algorithm $\mathcal{A}$ such that, for every finite field $\mathbb{F}$, positive integers m, d, subset H of $\mathbb{F}$, subset $S = \{(\alpha_1, \beta_1), \ldots, (\alpha_\ell, \beta_\ell)\} \subseteq \mathbb{F}^{\leq m} \times \mathbb{F}$, and $(\alpha, \beta) \in \mathbb{F}^{\leq m} \times \mathbb{F}$,*

$$\Pr\left[\mathcal{A}(\mathbb{F}, m, d, H, S, \alpha) = \beta\right] = \Pr_{R \leftarrow \mathbb{F}[X^{\leq d}_{1,\ldots,m}]}\left[R(\alpha) = \beta \;\middle|\; \begin{array}{c} R(\alpha_1) = \beta_1 \\ \vdots \\ R(\alpha_\ell) = \beta_\ell \end{array}\right].$$

Moreover $\mathcal{A}$ runs in time $m(d\ell|H| + d^3\ell^3) \cdot \mathrm{poly}(\log|\mathbb{F}|) = \ell^3 \cdot \mathrm{poly}(m, d, |H|, \log|\mathbb{F}|)$.

11.2 Strong zero knowledge sumcheck

We present our strong zero knowledge sumcheck protocol within the IPCP model. For brevity, throughout, we will refer to the weak zero knowledge IPCP for sumcheck in [BCFGRS17] simply as the "weak-ZK sumcheck protocol".

Construction 11.3. Fix a finite field $\mathbb{F}$, and $d, m, \lambda \in \mathbb{N}$ with $2\lambda \leq d$. Let G be any subset of $\mathbb{F}$ of size λ. In the protocol (P, V):

- P and V receive an instance (H, a) as common input;
- P additionally receives a summand polynomial $F \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}]$ as an oracle.

The interaction between P and V proceeds as follows:

1. P draws uniformly random polynomials $Z \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}, Y_{1,\ldots,k}^{\leq 2\lambda}]$ and $A \in \mathbb{F}[Y_{1,\ldots,k}^{\leq 2\lambda}]$, and sends as an oracle the polynomial
$$O(W, \vec{X}, \vec{Y}) := W \cdot Z(\vec{X}, \vec{Y}) + (1 - W) \cdot A(\vec{Y}) \in \mathbb{F}[X_{1,\ldots,m+k+1}^{\leq d}] \ ;$$
note that Z can be recovered as $O(1, \cdot)$ and A as $O(0, \vec{0}, \cdot)$.
2. P sends $z := \sum_{\vec{\alpha} \in H^m} \sum_{\vec{\beta} \in G^k} Z(\vec{\alpha}, \vec{\beta})$ to V.
3. V draws a random element ρ_1 in $\mathbb{F} \setminus \{0\}$ and sends it to P.
4. P and V run the *standard* sumcheck IP [LFKN92] on the statement "$\sum_{\vec{\alpha} \in H^m} Q(\vec{\alpha}) = \rho_1 a + z$" where
$$Q(X_1, \ldots, X_m) := \rho_1 F(X_1, \ldots, X_m) + \sum_{\vec{\beta} \in G^k} Z(X_1, \ldots, X_m, \vec{\beta}) \ ,$$
with P playing the role of the prover and V that of the verifier, and the following modification.
For $i = 1, \ldots, m$, in the i-th round, V samples its random element c_i from the set I rather than from all of $\mathbb{F}$; if P ever receives $c_i \in \mathbb{F} \setminus I$, it immediately aborts. In particular, in the m-th round, P sends a polynomial $g_m(X_m) := \rho_1 F(c_1, \ldots, c_{m-1}, X_m) + \sum_{\vec{\beta} \in G^k} Z(c_1, \ldots, c_{m-1}, X_m, \vec{\beta})$ for some $c_1, \ldots, c_{m-1} \in I$.
5. V sends $c_m \in I$ to P.
6. P sends the element $w := \sum_{\vec{\beta} \in G^k} Z(\vec{c}, \vec{\beta})$ to V, where $\vec{c} := (c_1, \ldots, c_m)$.
7. P and V engage in the weak-ZK sumcheck protocol with respect to the claim $\sum_{\vec{\beta} \in G^k} Z(\vec{c}, \vec{\beta}) = w$, using A as the oracle. If the verifier in that protocol rejects, so does V.
8. V outputs the claim "$F(\vec{c}) = \frac{g_m(c_m) - w}{\rho_1}$".

Remark 11.4. Formally, the protocol in Theorem 11.1 is not presented as a proper low-degree IPCP, but rather as a reduction with respect to some fixed, yet *inaccessible* low-degree polynomial. Nevertheless, this reduction perspective is consistent with our application, and indeed when in Section 12 we use the protocol in Theorem 11.1 as a sub-procedure, we obtain a low-degree IPCP per our definition in Section 5.

11.3 Analysis of the protocol

The following theorem, which is a more elaborate version of Theorem 11.1, provides an analysis of Construction 11.3. We stress that the protocol will satisfy a relaxed notion of soundness, similar to low-degree soundness, where the "no" instances are required to be low-degree polynomials. This suffices for our applications.

Theorem 11.5. *For every finite field $\mathbb{F}$, $d, k, \lambda \in \mathbb{N}$, $2\lambda \leq d$, there exists an $(\mathbb{F}, d, m + k + 1)$-low-degree IPCP system (P, V) such that, for every $F \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}]$, the following holds.*

- COMPLETENESS. *If $\sum_{\vec{\alpha}\in H^m} F(\vec{\alpha}) = a$, then $V(H,a)$, when interacting with $P^F(H,a)$, outputs a true claim of the form "$F(\vec{\gamma}) = a$" (with $\vec{\gamma} \in \mathbb{F}^m$ and $a \in \mathbb{F}$) with probability* 1.

- SOUNDNESS. *If $\sum_{\vec{\alpha}\in H^m} F(\vec{\alpha}) \neq a$, then for any malicious prover $\tilde{P}$ it holds that $V(H,a)$, when interacting with $\tilde{P}$, outputs a true claim "$F(\vec{\gamma}) = a$" (with $\vec{\gamma} \in \mathbb{F}^m$ and $a \in \mathbb{F}$) with probability at most $\frac{md}{|I|} + \frac{kd+2}{|\mathbb{F}|-1}$.*

- ZERO KNOWLEDGE. *There exists a simulator S such that if $\sum_{\vec{\alpha}\in H^m} F(\vec{\alpha}) = a$, then for every λ^k-query malicious verifier $\tilde{V}$, the following two distributions are equal*

$$S^{\tilde{V},F}(H,a) \quad \textit{and} \quad \text{View}\,\langle P^F(H,a), \tilde{V}\rangle \ .$$

  Moreover:

  - *S makes a single query to F at a point in I^m;*
  - *S runs in time*

$$(m+k)((d+\lambda)\mathsf{q}_{\tilde{V}}|H| + (d+\lambda)^3\mathsf{q}_{\tilde{V}}^3)\cdot \text{poly}(\log|\mathbb{F}|) = \text{poly}(\log|\mathbb{F}|, d, m, \lambda, k, |H|)\cdot \mathsf{q}_{\tilde{V}}^3 \ ,$$

    where $\mathsf{q}_{\tilde{V}}$ is $\tilde{V}$'s query complexity;
  - *S's behavior does not depend on a until after the simulated $\tilde{V}$ sends its first message.*

Remark 11.6 (space complexity)**.** With two-way access to the random tape, the prover can be made to run in space complexity $\text{poly}(\log|\mathbb{F}|, d, m, \lambda, k, |H|)$.

Remark 11.7 (straightline simulators)**.** Inspection shows that our simulators, in fact, achieve the stronger notion of universal *straightline simulators* [FS89; DS98], in which the simulator do *not* rewind the verifier.

Proof. Completeness is immediate from the protocol description and the completeness property of the sumcheck sub-protocols it invokes. Soundness follows from the fact that, fixing F such that $\sum_{\vec{\alpha}\in H^m} F(\vec{\alpha}) \neq a$, we can argue as follows:

- For every polynomial $Z \in \mathbb{F}[X_{1,\ldots,m+k}^{\leq d}]$, with probability $1 - \frac{1}{|\mathbb{F}|-1}$ over the choice of ρ_1 it holds that $\sum_{\vec{\alpha}\in H^m} Q(\vec{\alpha}) \neq \rho_1 a + z$, i.e., the sumcheck claim is false.

- Therefore, by the soundness guarantee of the sumcheck protocol, with probability at least $1 - md/|I|$, either the verifier rejects or $\rho_1 F(\vec{c}) + \sum_{\vec{\beta}\in G^k} Z(\vec{c},\vec{\beta}) \neq g_m(c_m)$.

- Finally, we distinguish between two cases depending on $\tilde{P}$:
  - If $\tilde{P}$ sends $w \neq \sum_{\vec{\beta}\in G^k} Z(\vec{c},\vec{\beta})$, then by the soundness guarantee of the weak-ZK sumcheck protocol, the verifier rejects with probability at least $1 - \frac{k\cdot d+1}{|\mathbb{F}|}$.
  - If $\tilde{P}$ sends $w = \sum_{\vec{\beta}\in G^k} Z(\vec{c},\vec{\beta})$, then $F(\vec{c}) \neq \frac{g_m(c_m)-w}{\rho_1}$ with probability 1.

Taking a union bound on the above cases yields the claimed soundness error.

To show the perfect zero knowledge guarantee, we need to construct a suitably-efficient simulator that perfectly simulates the view of any malicious verifier $\tilde{V}$. We first construct an *inefficient* simulator S_{slow} and prove that its output follows the desired distribution; afterwards, we explain how the simulator can be made efficient.

The simulator S_{slow}, given (straightline) access to $\tilde{V}$ and oracle access to F, works as follows:

1. Draw $Z_{\text{sim}} \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}, Y_{1,\ldots,k}^{\leq 2\lambda}]$. Run the weak-ZK sumcheck simulator S'.
2. Begin simulating $\tilde{V}$. Its queries to O are answered by making the appropriate queries to Z_{sim} and the simulated A provided by S'.
3. Send $z_{\text{sim}} := \sum_{\vec{\alpha} \in H^m} \sum_{\vec{\beta} \in G^k} Z_{\text{sim}}(\vec{\alpha}, \vec{\beta})$.
4. Receive $\tilde{\rho}$. Draw $Q_{\text{sim}} \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}]$ uniformly at random conditioned on $\sum_{\vec{\alpha} \in H^m} Q_{\text{sim}}(\vec{\alpha}) = \tilde{\rho} a + z_{\text{sim}}$, then engage in the sumcheck protocol on the claim "$\sum_{\vec{\alpha} \in H^m} Q_{\text{sim}}(\vec{\alpha}) = \tilde{\rho} a + z_{\text{sim}}$". If in any round $\tilde{V}$ sends $c_i \notin I$ as a challenge, abort.
5. Let $\vec{c} \in I^m$ be the point chosen by $\tilde{V}$ in the sumcheck protocol above. Query $F(\vec{c})$, and set $w_{\text{sim}} := Q_{\text{sim}}(\vec{c}) - \tilde{\rho} F(\vec{c})$; send this value to the verifier.
6. Draw $Z'_{\text{sim}} \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}, Y_{1,\ldots,k}^{\leq 2\lambda}]$ uniformly at random conditioned on
   - $\sum_{\vec{\beta} \in G^k} Z'_{\text{sim}}(\vec{c}, \vec{\beta}) = w_{\text{sim}}$, and
   - $Z'_{\text{sim}}(\vec{\gamma}) = Z_{\text{sim}}(\vec{\gamma})$ for all previous queries $\vec{\gamma}$ to Z.

   From this point on, answer all queries to Z with Z'_{sim}.
7. Use S' to simulate the sumcheck protocol for the claim "$\sum_{\vec{\beta} \in G^k} Z'_{\text{sim}}(\vec{c}, \vec{\beta}) = w_{\text{sim}}$".
8. Output the view of the simulated $\tilde{V}$.

To prove that this simulator outputs the correct distribution, we consider the information that the verifier receives in each step and show that the corresponding random variable is distributed identically to the view of the verifier in the real protocol. It will be convenient to define $R(\vec{X}) = \sum_{\vec{\beta} \in G^k} Z(\vec{X}, \vec{\beta})$, and R_{sim} likewise. We proceed with a step-by-step analysis (the relevant steps are in Steps 2 to 7).

In Step 2 and Step 3, the verifier has query access to a uniformly random polynomial Z and receives its summation over $H^m \times G^k$, exactly as in the real protocol.

In Step 4, we simulate the (standard) sumcheck protocol on the polynomial Q_{sim}, which is chosen uniformly at random conditioned on $\sum_{\vec{\alpha} \in H^m} Q_{\text{sim}}(\vec{\alpha}) = \rho_1 a + z_{\text{sim}}$. A key observation is that this is the distribution of Q in the real protocol. To see this, note that by Corollary 10.3, we have that $\tilde{\rho}$, being a function of fewer than λ^k queries to Z, is independent of R given $\sum_{\vec{\alpha} \in H^m} R(\vec{\alpha}) = z$. Then $\tilde{\rho} F$ is a random variable conditionally independent of R, and so $Q = R + \tilde{\rho} F$ is a uniformly random polynomial such that $\sum_{\vec{\alpha} \in H^m} Q(\vec{\alpha}) = \tilde{\rho} a + z$.[21]

In Step 5, we send $w_{\text{sim}} Q_{\text{sim}}(\vec{c}) - \tilde{\rho} F(\vec{c})$ to the verifier. In the real protocol, we send $w \sum_{\vec{\beta} \in G^k} Z(\vec{c}, \vec{\beta}) = Q(\vec{c}) - \tilde{\rho} F(\vec{c})$, where the latter equality is by the definition of Q. Since Q and Q_{sim} are identically distributed, then w andw_{sim} are also identically distributed.

In Step 6, we replace Z_{sim} with a new oracle Z'_{sim} (that is consistent with Z_{sim} on all points in which it was queried), which is a commitment to R'_{sim} such that $R'_{\text{sim}}(\vec{c}) = w_{\text{sim}}$. Consider any future query to Z, which happens after this replacement. We show that this query is distributed exactly as in the original protocol. By Corollary 10.3, the following holds for any $\vec{q} \in \mathbb{F}^{m+k}, a \in \mathbb{F}$, where $U \subseteq \mathbb{F}^{m+k} \times \mathbb{F}$ is the set of previous query-answer pairs.

$$\Pr_{Z'_{\text{sim}}}\left[Z'_{\text{sim}}(\vec{q}) = a \;\middle|\; \begin{array}{c} Z'_{\text{sim}}(\vec{\gamma}) = b \quad \forall (\vec{\gamma}, b) \in U \\ \sum_{\vec{\beta} \in G^k} Z'_{\text{sim}}(\vec{c}, \vec{\beta}) = w_{\text{sim}} \end{array} \right] = \Pr_{Z}\left[Z(\vec{q}) = a \;\middle|\; \begin{array}{c} Z(\vec{\gamma}) = b \quad \forall (\vec{\gamma}, b) \in U \\ \sum_{\vec{\beta} \in G^k} Z(\vec{X}, \vec{\beta}) \equiv Q(\vec{X}) - \tilde{\rho} F(\vec{X}) \end{array} \right]$$

[21] Note that if $\tilde{\rho}$ were not independent of R then this may not be true.

Observe that the left hand side describes the distribution of the answer to oracle query $\vec{q}$ provided by the simulator after we replace the Z-oracle, and the right hand side describes the distribution of the answer to the same query in the real protocol.

In Step 7, we make use of the weak-ZK simulator for the decommitment. Since, after the replacement of Z_{sim} by Z'_{sim}, the statement we are proving is true, we can use its zero knowledge guarantee. This ensures that the only information the verifier gains is the value $R(\vec{c}) = w$, which we already simulate, and a number of evaluations of Z equal to the number of queries to A, which we can fold into the query bound. This concludes the argument for the correctness of the inefficient simulator S_{slow}.

To complete the proof of zero knowledge, we note that S_{slow} can be transformed into an efficient simulator S by using succinct constraint detection for the Reed–Muller code extended with partial sums [BCFGRS17]: more precisely, we can use the algorithm of Lemma 11.2 to answer both point and sum queries to Z, A, and Q, in a stateful way, maintaining corresponding tables $\mathsf{ans}_{Z_{\mathrm{sim}}}$, $\mathsf{ans}_{A_{\mathrm{sim}}}$, and $\mathsf{ans}_{Q_{\mathrm{sim}}}$. □

12 Zero knowledge low-degree IPCP for NEXP

In this section we use the zero knowledge sumcheck protocol developed in Section 11 (along with the [BCFGRS17] protocol) to build a zero knowledge low-degree IPCP for $\mathbf{NEXP}$, which is the key technical component in our proof of Theorem 1.

Our protocol is based on the IPCP for NEXP of [BFL91]. Recall that in this protocol, the prover first sends a low-degree extension of a $\mathbf{NEXP}$ witness, and then engages in the [LFKN92] sumcheck protocol on a polynomial related to the instance. To make this zero knowledge, the prover first takes a *randomized* low-degree extension R of the witness (which provides some bounded independence), and then sets the oracle to be an algebraic commitment to R. Namely, the prover draws a polynomial uniformly at random subject to the condition that "summing out" a few of its variables yields R, and places its evaluation in the oracle.

The prover and verifier then engage in the zero knowledge sumcheck detailed in Section 11 with respect to the [BFL91] polynomial. This ensures that the verifier learns nothing through the interaction except for a single evaluation of the summand polynomial, which corresponds to learning a constant number of evaluations of the randomized witness. Bounded independence ensures that these evaluations do not leak any information. The prover decommits these evaluations to the verifier, using the "weak" zero knowledge sumcheck protocol in [BCFGRS17].

Following [BFLS91], the arithmetization encodes bit strings as elements in H^m for some H of size $\mathrm{poly}(|B|)$, rather than with $H = \{0,1\}$ as in [BFL91], for improved efficiency.

We start by defining the *oracle 3-satisfiability problem*, which is the $\mathbf{NEXP}$-complete problem used in [BFL91] to construct two-prover interactive proofs for $\mathbf{NEXP}$.

Definition 12.1 ($\mathscr{R}_{\mathrm{O3SAT}}$)**.** *The* **oracle 3-satisfiability relation***, denoted $\mathscr{R}_{\mathrm{O3SAT}}$, consists of all instance-witness pairs $(x,w) = \big((r,s,B),A\big)$, where r,s are positive integers, $B\colon \{0,1\}^{r+3s+3} \to \{0,1\}$ is a boolean formula, and $A\colon \{0,1\}^s \to \{0,1\}$ is a function, that satisfy the following condition:*

$$\forall\, z \in \{0,1\}^r,\ \forall\, b_1,b_2,b_3 \in \{0,1\}^s,\ B\big(z,b_1,b_2,b_3,A(b_1),A(b_2),A(b_3)\big) = 1\ .$$

In the rest of this section, we prove the following theorem, which shows that every language in $\mathbf{NEXP}$ has a perfect zero knowledge low-degree IPCP with polynomial communication and query complexity.

Theorem 12.2 (PZK low-degree IPCP for $\mathbf{NEXP}$)**.** *There exists $c \in \mathbb{N}$ such that for any query bound function $\mathsf{b}(n)$, some integers $d(n) = \Omega(n^c), m(n) = O(n^c \log \mathsf{b})$, and any sequence of fields $\mathbb{F}(n)$ that are extension fields of $\mathbb{F}_2$ with $|\mathbb{F}(n)| = \Omega((n^c \log \mathsf{b})^4)$, the $\mathbf{NEXP}$-complete relation $\mathscr{R}_{\mathrm{O3SAT}}$ has a public-coin, non-adaptive $(\mathbb{F},d,m)$-low-degree IPCP, with parameters*

$$\left|\begin{array}{rl} \textit{round complexity:} & O(n,\mathsf{b}) \\ \textit{PCP length:} & \mathrm{poly}(2^n,\mathsf{b}) \\ \textit{communication complexity:} & \mathrm{poly}(n,\log \mathsf{b}) \\ \textit{query complexity:} & \mathrm{poly}(n,\log \mathsf{b}) \\ \textit{oracle} \in & \mathbb{F}[X_{1,\ldots,m}^{\le d}] \\ \textit{soundness error:} & 1/2 \end{array}\right|,$$

which is zero knowledge with query bound b.

Proof. We begin with the arithmetization of the problem.

Arithmetization. Let $\hat{B}\colon \mathbb{F}^m \to \mathbb{F}$ be the "direct" arithmetization of the negation of the constraint B: rewrite B by using ANDs and NOTs; negate its output; replace each $\mathrm{AND}(a,b)$ with $a \cdot b$ and $\mathrm{NOT}(a)$ with $1-a$. For every $\vec{x} \in \{0,1\}^{r+3s+3}$, $\hat{B}(\vec{x}) = 0$ if $B(\vec{x})$ is true, and $\hat{B}(\vec{x}) = 1$ if $B(\vec{x})$ is false. Note that $\hat{B}$ is computable in time $\mathrm{poly}(|B|)$ and has total degree $O(|B|)$.

Note that $(r, s, B) \in \mathscr{R}_{\mathrm{O3SAT}}$ if and only if there exists a multilinear function $\hat{A}\colon \mathbb{F}^s \to \mathbb{F}$ (the multilinear extension of the assignment A) that is boolean on $\{0,1\}^s$ such that $\hat{B}(\vec{z}, \vec{b}_1, \vec{b}_2, \vec{b}_3, \hat{A}(\vec{b}_1), \hat{A}(\vec{b}_2), \hat{A}(\vec{b}_3)) = 0$, for all $\vec{z} \in \{0,1\}^r$, $\vec{b}_1, \vec{b}_2, \vec{b}_3 \in \{0,1\}^s$.

The requirement that $\hat{A}$ is boolean on $\{0,1\}^s$ can be encoded by 2^s constraints: $\hat{A}(\vec{b})(1 - \hat{A}(\vec{b})) = 0$ for every $\vec{b} \in \{0,1\}^s$. These constraints together can be expressed as follows:

$$\Big\{ g_1(\vec{\alpha}) := \hat{B}(\vec{z}, \vec{b}_1, \vec{b}_2, \vec{b}_3, \hat{A}(\vec{b}_1), \hat{A}(\vec{b}_2), \hat{A}(\vec{b}_3)) = 0 \Big\}_{\vec{z} \in \{0,1\}^r,\, \vec{b}_i \in \{0,1\}^s}$$
$$\Big\{ g_2(\vec{\beta}) := \hat{A}(\vec{b})(1 - \hat{A}(\vec{b})) = 0 \Big\}_{\vec{b} \in \{0,1\}^s}$$

Let F be the polynomial over $\mathbb{F}$ given by

$$F(\vec{X}, \vec{Y}) := \sum_{\vec{\alpha} \in \{0,1\}^{r+3s}} \Big(g_1(\vec{\alpha}) \vec{X}^{\vec{\alpha}} + g_2(\vec{\alpha}_{[s]}) \vec{Y}^{\vec{\alpha}} \Big) \ ,$$

where $\vec{X}^{\vec{\alpha}} := X_1^{\alpha_1} \cdots X_\ell^{\alpha_\ell}$ for $\vec{\alpha} \in \{0,1\}^\ell$, and $\vec{\alpha}_{[s]}$ are the first s coordinates in $\vec{\alpha}$.

Note that $F \equiv 0$ if and only if all the above constraints hold. Since F is a polynomial of total degree $r + 3s$, if $F \not\equiv 0$, then F is zero on at most an $\frac{r+3s}{|\mathbb{F}|}$ fraction of points in $\mathbb{F}^{2(r+3s)}$.

For $\alpha_i \in \{0,1\}$ it holds that $X_i^{\alpha_i} = 1 + (X_i - 1)\alpha_i$, so we can also write

$$F(\vec{X}, \vec{Y}) = \sum_{\vec{\alpha} \in \{0,1\}^{r+3s}} \left(g_1(\vec{\alpha}) \cdot \prod_{i=1}^{r+3s} (1 + (X_i - 1)\alpha_i) + g_2(\vec{\alpha}_{[s]}) \cdot \prod_{i=1}^{r+3s} (1 + (Y_i - 1)\alpha_i) \right) \ .$$

Let H be a subfield of $\mathbb{F}$ of size $\mathrm{poly}(r + s + \log \mathsf{b})$; define $m_1 := r/\log|H|$ and $m_2 := s/\log|H|$ (assuming without loss of generality that both are integers). For $i \in \{1,2\}$, let $\gamma_i\colon H^{m_i} \to \{0,1\}^{m_i \log|H|}$ be the lexicographic order on H^{m_i}. The low-degree extension $\hat{\gamma}_i$ of γ_i is computable by an arithmetic circuit constructible in time $\mathrm{poly}(|H|, m_i, \log|\mathbb{F}|)$ [GKR15, Claim 4.2]. Let $\gamma\colon H^{m_1+3m_2} \to \{0,1\}^{r+3s}$ be such that $\gamma(\vec{\alpha}, \vec{\beta}_1, \vec{\beta}_2, \vec{\beta}_3) = (\gamma_1(\vec{\alpha}), \gamma_2(\vec{\beta}_1), \gamma_2(\vec{\beta}_2), \gamma_2(\vec{\beta}_3))$ for all $\vec{\alpha} \in H^{m_1}$, $\vec{\beta}_1, \vec{\beta}_2, \vec{\beta}_3 \in H^{m_2}$; let $\hat{\gamma}\colon \mathbb{F}^{m_1+3m_2} \to \mathbb{F}^{r+3s}$ be its low-degree extension.

We can use the above notation to write F equivalently as

$$\begin{aligned} F(\vec{X}, \vec{Y}) = & \sum_{\substack{\vec{\alpha} \in H^{m_1} \\ \vec{\beta}_1, \vec{\beta}_2, \vec{\beta}_3 \in H^{m_2}}} g_1(\hat{\gamma}(\vec{\alpha}, \vec{\beta}_1, \vec{\beta}_2, \vec{\beta}_3)) \prod_{i=1}^{r+3s} (1 + (X_i - 1)\hat{\gamma}(\vec{\alpha}, \vec{\beta}_1, \vec{\beta}_2, \vec{\beta}_3)_i) \\ & \qquad + g_2(\hat{\gamma}_2(\vec{\beta}_1)) \prod_{i=1}^{r+3s} (1 + (Y_i - 1)\hat{\gamma}(\vec{\alpha}, \vec{\beta}_1, \vec{\beta}_2, \vec{\beta}_3)_i) \\ =: & \sum_{\substack{\vec{\alpha} \in H^{m_1} \\ \vec{\beta}_1, \vec{\beta}_2, \vec{\beta}_3 \in H^{m_2}}} f(\vec{X}, \vec{Y}, \vec{\alpha}, \vec{\beta}_1, \vec{\beta}_2, \vec{\beta}_3) \ , \end{aligned}$$

where f has degree at most d in each variable. We are now ready to specify the protocol.

Low-degree IPCP for $\mathscr{R}_{\mathbf{O3SAT}}$. Let $k := \lceil \log 100\mathsf{b} / \log|H| \rceil$. The interaction is as follows.

1. The prover draws a polynomial Z uniformly at random from $\mathbb{F}[X_{1,\ldots,m_2}^{\leq |H|+2}, Y_{1,\ldots,k}^{\leq 2|H|}]$, subject to the condition that $\sum_{\vec{\beta} \in G^k} Z(\vec{\alpha}, \vec{\beta}) = A(\gamma_2(\vec{\alpha}))$ for all $\vec{\alpha} \in H^{m_2}$. It then generates an oracle π_0 for the $|H|^k$-strong zero

knowledge sumcheck protocol (Section 11) on input $(\mathbb{F}, m_1 + 3m_2, d, H, 0)$ and oracles π_1, π_2, π_3 for the invocation of the weak zero knowledge sumcheck protocol in [BCFGRS17] on input $(\mathbb{F}, k, 2|H|, H, \cdot)$. (In both zero knowledge sumchecks, the oracle message does not depend on the claim itself.) The prover sends an oracle which is the "bundling" of the evaluations of Z with $(\pi_0, \pi_1, \pi_2, \pi_3)$.[22]

2. The verifier chooses $\vec{x}, \vec{y} \in \mathbb{F}^{r+3s}$ uniformly at random and sends them to the prover. The prover and verifier engage in the zero knowledge sumcheck protocol of Section 11 with respect to the claim "$F(\vec{x}, \vec{y}) = 0$" with $I = \mathbb{F} \setminus H$, using π_1 as the oracle message. This reduces the claim to checking that $f(\vec{x}, \vec{y}, \vec{c}, \vec{c}'_1, \vec{c}'_2, \vec{c}'_3) = a$ for uniformly random $\vec{c} \in (\mathbb{F} \setminus H)^{m_1}$, $\vec{c}'_1, \vec{c}'_2, \vec{c}'_3 \in (\mathbb{F} \setminus H)^{m_2}$, and some $a \in \mathbb{F}$ provided by the prover.

3. The prover provides $h_i := A(\gamma_2(\vec{c}'_i))$ for each $i \in \{1, 2, 3\}$. The verifier substitutes these values into the expression for f to check the above claims, and rejects if they do not hold.

4. The prover and verifier engage in the zero knowledge sumcheck protocol in [BCFGRS17] with respect to the claims "$\sum_{\vec{\beta} \in H^k} Z(\vec{c}'_i, \vec{\beta}) = h_i$", for each $i \in \{1, 2, 3\}$, using π_i as the oracle message.

Completeness. If $((r, s, B), A) \in \mathscr{R}_{\mathrm{O3SAT}}$, then $F(\vec{X}, \vec{Y})$ is the zero polynomial; hence $F(\vec{x}, \vec{y}) = 0$ for all $\vec{x}, \vec{y} \in \mathbb{F}^{r+3s}$. Completeness follows from the completeness of the zero knowledge sumcheck protocols.

Low-degree soundness. Suppose that $(r, s, B) \notin \mathscr{L}(\mathscr{R}_{\mathrm{O3SAT}})$, and let $(\tilde{Z}, \tilde{\pi}_0, \tilde{\pi}_1, \tilde{\pi}_2, \tilde{\pi}_3)$ be the PCP oracle sent by a malicious prover. By the low-degree soundness condition, this is a collection of polynomials of individual degree at most d. Let $\tilde{A} := \sum_{\vec{\beta} \in H^k} \tilde{Z}(\vec{X}, \vec{\beta})$, which we think of as playing the role of $\hat{A}(\gamma_2(\cdot))$ (the assignment) in F. Observe that $\tilde{A}$ has individual degree at most d.

If $(r, s, B) \notin \mathscr{L}(\mathscr{R}_{\mathrm{O3SAT}})$, then there is no choice of $\hat{A}$ such that $F(\vec{X}, \vec{Y})$ is the zero polynomial. Thus, $F(\vec{x}, \vec{y}) = 0$ with probability at most $(r + 3s)/|\mathbb{F}|$ over the choice of $\vec{x}, \vec{y}$. By the soundness of the zero knowledge sumcheck protocol (Theorem 11.5), the verifier outputs a false claim "$f(\vec{x}, \vec{y}, \vec{\alpha}) = a$" with probability at least $1 - O((m_1 + 3m_2 + k)d)/(|\mathbb{F}| - |H|))$. If substituting h_i for $\hat{A}(\gamma_2(\vec{c}'_i))$ in f does not yield a, then the verifier rejects. Otherwise, it must be the case that for at least one $i \in \{1, 2, 3\}$, $\tilde{A}(\vec{c}'_i) \neq h_i$. By the soundness of the sumcheck protocol in [BCFGRS17], the verifier rejects with probability at least $1 - O(\frac{kd}{|\mathbb{F}|})$. Taking a union bound, the verifier rejects with probability at least $1 - O((m_1 + 3m_2 + k)d/|\mathbb{F}|) = 1 - O((r + s + \log \mathsf{b})d/|\mathbb{F}|)$.

Zero knowledge. Perfect zero knowledge is achieved via the following (straightline) simulator.

1. Draw a uniformly random polynomial $Z_{\mathrm{sim}} \in \mathbb{F}[X^{\leq |H|+2}_{1,\ldots,m_2}, Y^{\leq 2|H|}_{1,\ldots,k}]$.
2. Invoke the $|H|^k$-strong ZK sumcheck simulator on input $(\mathbb{F}, m_1 + 3m_2, \deg(f), H, 0)$, and use it to answer queries to π_0 throughout. In parallel, run three copies of the simulator for the weak ZK sumcheck in [BCFGRS17], with respect to input $(\mathbb{F}, k, 2|H|, H, \cdot)$, and use them to answer queries to π_1, π_2, π_3 respectively. (Recall that the behavior of each simulator does not depend on the claim being proven until after the first simulated message, so we can choose these later.)
3. Receive $\vec{x}, \vec{y} \in \mathbb{F}^{r+3s}$ from $\tilde{V}$.
4. Simulate the strong ZK sumcheck protocol on the claim "$F(\vec{x}, \vec{y}) = 0$". The subsimulator will query f at a single location $\vec{c} \in (\mathbb{F} - H)^{r+3s}$. Reply with the value $f(\vec{x}, \vec{y}, \vec{c})$, for $\vec{c} = (\vec{c}_0, \vec{c}_1, \vec{c}_2, \vec{c}_3) \in (\mathbb{F} \setminus H)^{r+3s}$.

[22] By "bundling" we refer to a standard technique of sending a single low-degree polynomial which encodes a list of low-degree polynomials. More precisely, the bundling of $P_1(\vec{X}), \ldots, P_\ell(\vec{X})$ is the polynomial $P(W, \vec{X}) := \sum_{\alpha \in S} I_S(W, \alpha) P_{\gamma(\alpha)}(\vec{X})$, for some $S \subseteq \mathbb{F}$ such that $|S| = \ell$ and $\gamma : S \to \{1, \ldots, \ell\}$ an ordering of S. Observe that (a) $P(i, \vec{X}) \equiv P_i(\vec{X})$ for all $i = 1, \ldots, k$; (b) $\deg(P) = \max\{\deg(P_1), \ldots, \deg(P_\ell), |S| - 1\}$; (c) any query to P can be answered by querying each P_i at that point, and so the zero knowledge guarantee is unaffected except for reducing the query bound by a factor ℓ.

Computing this requires the values $\hat{A}(\hat{\gamma}(\vec{c}_i))$ for $i \in \{1,2,3\}$; we substitute each of these with $h^i_{\text{sim}} \in \mathbb{F}$ drawn uniformly at random (except that if $\vec{c}_i = \vec{c}_j$ for $i \neq j$, then fix $h^i_{\text{sim}} = h^j_{\text{sim}}$).

5. For $i \in \{1,2,3\}$, simulate the weak ZK sumcheck protocol in [BCFGRS17] with respect to the claim "$\sum_{\vec{\beta} \in H^k} Z(\vec{\alpha}, \vec{\beta}) = h^i_{\text{sim}}$". Whenever the subsimulator queries Z, answer using Z_{sim}.

The verifier's view consists of its interaction with P during the four sumcheck protocols it invokes and its queries to the oracle. The strong zero knowledge sumcheck subsimulator in Section 11 guarantees that the queries to π_0 and the first sumcheck are perfectly simulated given a single query to f at the point $\vec{c} \in (\mathbb{F} \setminus H)^{r+3s}$ chosen by $\tilde{V}$. Since $\hat{A}'(\vec{X}) = \sum_{\vec{\beta} \in H^k} Z(\vec{X}, \vec{\beta}) \in \mathbb{F}[X^{\leq |H|+2}_{1,\ldots,m}]$, the evaluation of $\hat{A}$ at any 3 points outside of H^m does not determine its value at any point in H^m. In particular, this means that the values of the h_i's sent by the prover in the original protocol are independently uniformly random in $\mathbb{F}$ (except if $\vec{c}_i = \vec{c}_j$ for $i \neq j$ as above). Thus the h^i_{sim}'s are identically distributed to the h_i's, and therefore both the prover message and the simulator's query are perfectly simulated.

The sumcheck simulator in [BCFGRS17] ensures that the view of the verifier in the rest of the sumchecks is perfectly simulated given $q_{\tilde{V}}$ queries to Z, where $q_{\tilde{V}}$ is the number of queries the verifier makes across all π_i, $i \in \{1,2,3\}$. Hence, the number of "queries" the simulator makes to Z_{sim} is strictly less than $100\mathsf{b}$ (because $\tilde{V}$ is b-query). By Corollary 10.3, any set of strictly less than $100\mathsf{b}$ queries to Z is independent of $\hat{A}'$, and so the answers are identically distributed to the answers to those queries if they were made to a uniformly random polynomial, which is the distribution of Z_{sim}.

Clearly, drawing a uniformly random polynomial in $Z_{\text{sim}} \in \mathbb{F}[X^{\leq |H|+2}_{1,\ldots,m_2}, Y^{\leq 2|H|}_{1,\ldots,k}]$ cannot be done in polynomial time. However, we can instead use the algorithm of Lemma 11.2 to draw Z (a straightforward modification allows us to handle different degrees in $\vec{X}, \vec{Y}$; alternatively, we could simply set the degree bound for both to be $2|H|$). The running time of the simulator is then $\text{poly}(\log |\mathbb{F}|, m_1, m_2, k, |H|)$. $\square$

A Reducing query complexity while preserving zero knowledge

We prove Proposition 8.3 by showing that any low-degree IPCP can be transformed into a low-degree IPCP that makes a single uniform query, at only a small cost in parameters, while *preserving zero knowledge*.

Let $m, d \in \mathbb{N}$, and let $\mathbb{F}$ be a finite field of size $|\mathbb{F}| > (md/\varepsilon)^C$. Let (P, V) be an r-round $(\mathbb{F}, d, m)$-low-degree IPCP for a language $\mathscr{L}$. Denote its oracle by R, query complexity by q, PCP length by l, communication complexity by c, and soundness error by $\varepsilon = 1/2$.

We transform (P, V) into a low-degree IPCP (P', V') for $\mathscr{L}$ with parameters

$$\left|\begin{array}{rl} \text{round complexity:} & \mathsf{r}'=\mathsf{r}+1 \\ \text{PCP length:} & \mathsf{l}'=\mathsf{l} \\ \text{communication complexity:} & \mathsf{c}'=\mathsf{c}+\mathrm{poly}(d,\mathsf{q},m) \\ \text{query complexity:} & \mathsf{q}'=1 \\ \text{oracle} \in & \mathbb{F}[X_{1,\ldots,m}^{\leq d}] \\ \text{soundness error:} & \varepsilon'=\varepsilon+\frac{d\mathsf{q}}{|\mathbb{F}|-\mathsf{q}} \end{array}\right| ,$$

where the new honest verifier's single query is uniformly distributed. Furthermore, if (P, V) is (perfect) zero knowledge with query bound b, then (P', V') is (perfect) zero knowledge with query bound $\mathsf{b} - (d\mathsf{q} + 1)$.

We reduce the query complexity of the IPCP verifier V from q to 1 by using the standard approach of leveraging algebraic structure and additional interaction with the prover, while making sure that the query reduction preserves zero knowledge and that the (single) query that V makes is uniformly distributed. Specifically, if the verifier wants to query the oracle R at every point in a set A, the verifier asks the prover to provide the restriction of R to a curve that contains all points in A. Since we wish to make a single uniform query, rather than asking for the curve of minimal degree (which is unique), we choose the curve at random from all such curves of degree at most $|A|$. This technique is used by [KR08] to show the same result for general public-coin IPCPs (with standard soundness).[23] Since we require only low-degree soundness, we can dramatically simplify their proof. Consider the following protocol.

Construction A.1. Let (P, V) be a q-query $(\mathbb{F}, d, m)$-low-degree IPCP for $\mathscr{L}$ in which the (honest) oracle is some $R \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}]$. We construct an $(\mathbb{F}, d, m)$-low-degree IPCP (P', V') for $\mathscr{L}$ in which V' makes a single uniformly distributed query to R. We may assume that $|\mathbb{F}| > \mathsf{q}$, otherwise the stated soundness guarantee is trivial.

1. *Random curve.* V' chooses a random $\vec{r} \in \mathbb{F}^m$ and a random $t \in \mathbb{F} \setminus S$, for some $S \subseteq \mathbb{F}$ with $|S| = \mathsf{q}$. V' computes a curve $\gamma \colon \mathbb{F} \to \mathbb{F}^m$ of degree q such that $\{\gamma(s)\}_{s \in S} = A$ and $\gamma(t) = \vec{r}$ and sends it to P'. P' replies with the coefficients of the polynomial $\rho \colon \mathbb{F} \to \mathbb{F}$, of degree at most $d\mathsf{q}$, that (allegedly) is the restriction of R to γ.
2. *Consistency.* V queries R at $\vec{r}$ and receives an answer a; it rejects if $a \neq \rho(t)$. (Recall that $\vec{r} = \gamma(t)$.)
3. *Emulating the multi-query verifier.* V' rules according to the decision predicate of V with respect to the transcript of the "interaction phase" and the answers to the query set A, as indicated by the curve γ.

One can verify that the complexity of the protocol is as stated. Since the evaluation of the curve γ (which has degree q) at a random $t \in \mathbb{F} \setminus S$ is a random variable that is uniformity distributed over $\mathbb{F}^m$, the single query that the verifier makes is uniformly distributed. Completeness is immediate by construction. For soundness, since ρ is a univariate polynomial of degree at most $d\mathsf{q}$, and P' does not know t, if the check $\rho(t) = R(\vec{r})$ (in Step 2) passes with probability greater than $\frac{d\mathsf{q}}{|\mathbb{F}|-\mathsf{q}}$, then all the answers to the query set A, as indicated by γ, are consistent with R. Perfect zero knowledge is preserved because the polynomial ρ can be computed efficiently by making $d\mathsf{q} + 1$ queries to R, and so the additional information provided by the prover could have been computed by the malicious verifier itself.

[23] We remark that the transformation in [KR08] also implicitly assumes that the IPCP is public coin.

B From PCP to MIP* via a black box transformation

We show that any (non-adaptive) PCP, and more generally any IPCP, can be transformed into an MIP* in a black box way. While a proof of this fact is implicit in [Vid16; NV18], the machinery developed in Part I allows us to elucidate its structure and give a compellingly short proof of it.

We first define what we mean by *black box*. Informally, we call a transformation black box if it does not depend on the language being decided.[24] The following definition formalizes this notion.

Definition B.1. *A transformation T maps IPCP to MIP* if, given as input an IPCP (P, V) for a language $\mathscr{L}$, outputs an MIP* for the language $\mathscr{L}$. Such a transformation is* black box *if the verifier in the resulting MIP* can be expressed as an algorithm with access only to the queries and messages of $V(x)$ but no access to the input x, apart from its length.*

We stress that Definition B.1 also applies to PCPs (by viewing them as 0-round IPCPs).

The transformation is in two stages. First, we convert the IPCP into a low-degree IPCP by encoding the oracle as a low-degree polynomial. Second, we apply the transformation in Lemma 8.1 (which includes invoking the query reduction in Proposition 8.3) to convert the low-degree IPCP into an MIP*. Besides being black box in the formal sense, the resulting MIP* verifier is simple to describe: it is the original verifier, composed with an interactive query-reduction protocol and a low-degree test for entangled provers.

In sum, the above yields the following corollary.

Corollary B.2. *There exists a black box transformation that maps any r-round IPCP for a language $\mathscr{L}$ to a 2-prover $(\mathsf{r}+1)$-round MIP* for $\mathscr{L}$.*

Proof. Let (P, V) be an r-round IPCP for $\mathscr{L}$; denote its PCP oracle by R, query complexity by q, and proof length by l. Let $m, d \in \mathbb{N}$ be such that $\mathsf{l} \leq d^m$, and let $\mathbb{F}$ be a finite field with $|\mathbb{F}| > \max\{(2md)^C, 5d\mathsf{q}\}$ (where C is the constant from Theorem 7.2). Encode the oracle R of the IPCP as a polynomial $\hat{R} \in \mathbb{F}[X_{1,\ldots,d}^{\leq m}]$ by computing the low-degree extension of R (see Section 5.1); the new oracle is the evaluation of $\hat{R}$ over $\mathbb{F}^m$. Since this encoding is systematic, the verifier V can directly query $\hat{R}$ at the positions that correspond to its query set. Completeness and soundness are clearly preserved, as are the query and communication complexities. Proof length is increased from l to $|\mathbb{F}|^m$. The resulting IPCP satisfies the conditions of a low-degree IPCP, and so we can apply the exact argument as in Section 9 to obtain the desired MIP*. Straightforward inspection shows that all of the applied transformations are indeed black box. □

Remark B.3. Zero knowledge is *not* preserved by this transformation because taking the low-degree extension of the oracle may allow the verifier to learn global information that cannot, in general, be simulated via a small number of queries. (For example, a single point of the low-degree extension may amount to a summation over exponentially many points; see Appendix C.)

If we apply Corollary B.2 to any PCP (i.e., any 0-round IPCP) for $\mathbf{NEXP}$ (for example, the one in [BFLS91]), we immediately recover the following result from [NV18], which shows that every language in $\mathbf{NEXP}$ has an MIP* with optimal round complexity and number of provers.

Corollary B.4. *Every language in $\mathbf{NEXP}$ has a 1-round 2-prover MIP*.*

[24] This rules out degenerate "transformations" that ignore the given PCP or IPCP (P, V) for the language $\mathscr{L}$, and simply output an MIP* for $\mathscr{L}$ unrelated to (P, V). (Such degenerate transformations are trivially implied by the inclusion $\mathbf{NEXP} \subseteq \mathbf{MIP}^*$.)

C Algebraic query complexity upper bounds

We show that in certain cases the degree constraints in Lemma 10.1 are tight.

C.1 Multilinear polynomials

The first result is for the case of multivariate polynomials over any finite field, where $H \subseteq \mathbb{F}$ is arbitrary. The proof is a simple extension of a proof due to [JKRS09] for the case $H = \{0, 1\}$.

Theorem C.1 (multilinear polynomials)**.** *Let $\mathbb{F}$ be a finite field, H a subset of $\mathbb{F}$, and $\gamma := \sum_{\alpha \in H} \alpha$. For every $P \in \mathbb{F}[X_{1,\ldots,m}^{\leq 1}]$ (i.e., for every m-variate multilinear polynomial P) it holds that*

$$\sum_{\vec{\alpha} \in H^m} P(\vec{\alpha}) = \begin{cases} P\big(\frac{\gamma}{|H|}, \ldots, \frac{\gamma}{|H|}\big) \cdot |H|^m & \text{if } \mathrm{char}(\mathbb{F}) \nmid |H| \\ \kappa \cdot \gamma^m & \text{if } \mathrm{char}(\mathbb{F}) \mid |H| \end{cases} ,$$

where κ is the coefficient of $X_1 \cdots X_m$ in P.

Proof. First suppose that $\mathrm{char}(\mathbb{F})$ does not divide $|H|$. Let $\vec{\alpha}$ be uniformly random in H^m; in particular, α_i and α_j are independent for $i \neq j$. For every monomial $m(\vec{X}) = X_1^{e_1} \cdots X_m^{e_m}$ with $e_1, \ldots, e_m \in \{0, 1\}$,

$$\mathbb{E}[M(\vec{\alpha})] = \mathbb{E}[\alpha_1^{e_1} \cdots \alpha_m^{e_m}] = \mathbb{E}[\alpha_1^{e_1}] \cdots \mathbb{E}[\alpha_m^{e_m}] = \mathbb{E}[\alpha_1]^{e_1} \cdots \mathbb{E}[\alpha_m]^{e_m} = M(\mathbb{E}[\alpha_1], \ldots, \mathbb{E}[\alpha_m]) \ .$$

Since P is a linear combination of monomials, $\mathbb{E}[P(\vec{\alpha})] = P(\mathbb{E}[\vec{\alpha}])$. Each α_i is uniformly random in H, so $\mathbb{E}[\alpha_i] = \frac{1}{|H|} \sum_{\alpha \in H} \alpha = \frac{\gamma}{|H|}$, and thus $P(\mathbb{E}[\vec{\alpha}]) = P(\frac{\gamma}{|H|}, \ldots, \frac{\gamma}{|H|})$, which implies that $\mathbb{E}[P(\vec{\alpha})] = P(\frac{\gamma}{|H|}, \ldots, \frac{\gamma}{|H|})$. To deduce the claimed relation, it suffices to note that $\mathbb{E}[P(\vec{\alpha})] = \frac{1}{|H|^m} \sum_{\vec{\alpha} \in H^m} P(\vec{\alpha})$.

Next suppose that $\mathrm{char}(\mathbb{F})$ divides $|H|$. For every monomial $m(\vec{X}) = X_1^{e_1} \cdots X_m^{e_m}$ with $e_1, \ldots, e_m \in \{0, 1\}$:

- if there exists $j \in [m]$ such that $e_j = 0$ then

$$\sum_{\vec{\alpha} \in H^m} M(\vec{\alpha}) = |H| \sum_{\alpha_1, \ldots, \alpha_{j-1}, \alpha_{j+1}, \ldots, \alpha_m \in H} \alpha_1^{e_1} \cdots \alpha_{j-1}^{e_{j-1}} \alpha_{j+1}^{e_{j+1}} \cdots \alpha_m^{e_m} = 0 \ .$$

- if instead $e_1 = \cdots = e_m = 1$ then

$$\sum_{\vec{\alpha} \in H^m} M(\vec{\alpha}) = \sum_{\vec{\alpha} \in H^m} \prod_{i=1}^{m} \alpha_i = \prod_{i=1}^{m} \sum_{\alpha_i \in H} \alpha_i = \left(\sum_{\alpha \in H} \alpha \right)^m \ . \qquad \square$$

The following corollary shows that for prime fields of odd size, the value of $\sum_{\vec{\alpha} \in H^m} P(\vec{\alpha})$ can be computed efficiently for any $H \subseteq \mathbb{F}$ using at most a single query to P.

Corollary C.2. *Let $\mathbb{F}$ be a prime field of odd size, H a subset of $\mathbb{F}$, and $\gamma := \sum_{\alpha \in H} \alpha$. For every $P \in \mathbb{F}[X_{1,\ldots,m}^{\leq 1}]$ (i.e., for every m-variate multilinear polynomial P) it holds that*

$$\sum_{\vec{\alpha} \in H^m} P(\vec{\alpha}) = \begin{cases} P\big(\frac{\gamma}{|H|}, \ldots, \frac{\gamma}{|H|}\big) \cdot |H|^m & \text{if } \mathrm{char}(\mathbb{F}) \nmid |H| \\ 0 & \text{if } \mathrm{char}(\mathbb{F}) \mid |H| \end{cases} .$$

Proof. Theorem C.1 implies both cases. If $\mathrm{char}(\mathbb{F})$ does not divide $|H|$, then the claimed value is as in the theorem. If instead $\mathrm{char}(\mathbb{F})$ divides $|H|$, then it must be the case that $H = \mathbb{F}$, since $p := \mathrm{char}(\mathbb{F})$ equals $|\mathbb{F}|$; in this case, $\gamma = \sum_{\alpha \in H} \alpha = (p-1)p/2$, which is divisible by p since 2 must divide $p - 1$ (as p is odd). $\square$

C.2 Subsets with group structure

We show that if H is assumed to have some group structure, then few queries may suffice even for polynomials of degree greater than one. In particular, Lemma C.3 shows that if H is a multiplicative subgroup of $\mathbb{F}$ and $d \leq |H|$, then one query suffices; Lemma C.5 shows that if H is an additive subgroup of $\mathbb{F}$, then the answer depends on a polynomial related to H.

Lemma C.3 (multiplicative groups)**.** *Let $\mathbb{F}$ be a field, H a finite multiplicative subgroup of $\mathbb{F}$, and m, d positive integers with $d < |H|$. For every $P \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}]$,*

$$\sum_{\vec{\alpha} \in H^m} P(\vec{\alpha}) = P(0, \ldots, 0) \cdot |H|^m \ .$$

Remark C.4. The hypothesis that $d < |H|$ is necessary for the lemma, as we now explain. Choose $H = \mathbb{K}^\times$, where $\mathbb{K}$ is a proper subfield of $\mathbb{F}$, $m = 1$, and $d = |H|$. Consider the polynomial $X^{|H|}$, which has degree at least d: $X^{|H|}$ vanishes on 0; however, $X^{|H|}$ evaluates to 1 everywhere on H so that its sum over H equals $|H| \neq 0$. (Note that if H is a multiplicative subgroup of $\mathbb{F}$ then $\mathrm{char}(\mathbb{F}) \nmid |H|$ because $|H|$ equals $\mathrm{char}(\mathbb{F})^k - 1$ for some positive integer k.)

Proof. The proof is by induction on the number of variables m. The base case is when $m = 1$, which we argue as follows. The group H is cyclic, because it is a (finite) multiplicative subgroup of a field; so let ω generate H. Writing $P(X_1) = \sum_{j=0}^{d} \beta_j X_1^j$ for some $\beta_0, \ldots, \beta_d \in \mathbb{F}$, we have

$$\sum_{\alpha_1 \in H} P(\alpha_1) = \sum_{i=0}^{|H|-1} P(\omega^i) = \sum_{i=0}^{|H|-1} \sum_{j=0}^{d} \beta_j \omega^{ij} = \sum_{j=0}^{d} \beta_j \sum_{i=0}^{|H|-1} (\omega^j)^i = \beta_0 |H| = f(0)|H| \ ,$$

which proves the base case. The second-to-last equality follows from the fact that for every $\gamma \in H$,

$$\sum_{i=0}^{|H|-1} \gamma^i = \begin{cases} |H| & \text{if } \gamma = 1 \\ \frac{\gamma^{|H|}-1}{\gamma - 1} = 0 & \text{if } \gamma \neq 1 \end{cases} \ .$$

For the inductive step, assume the statement for any number of variables less than m; we now prove that it holds for m variables as well. Let P_α denote P with the variable X_1 fixed to α. Next, apply the inductive assumption below in the second equality (with $m - 1$ variables) and last one (with 1 variable), to obtain

$$\begin{aligned} \sum_{\vec{\alpha} \in H^m} P(\alpha_1, \ldots, \alpha_m) &= \sum_{\alpha_1 \in H} \sum_{(\alpha_2, \ldots, \alpha_m) \in H^{m-1}} P_{\alpha_1}(\alpha_2, \ldots, \alpha_m) \\ &= |H|^{m-1} \sum_{\alpha_1 \in H} P_{\alpha_1}(0^{m-1}) \\ &= |H|^{m-1} \sum_{\alpha_1 \in H} P(\alpha_1, 0, \ldots, 0) \\ &= |H|^m P(0, \ldots, 0) \ , \end{aligned}$$

as claimed. □

Lemma C.5 (additive groups)**.** *Let $\mathbb{F}$ be a field, H a finite additive subgroup of $\mathbb{F}$, and m, d positive integers with $d < |H|$. For every $\vec{v} \in \mathbb{F}^m$, $P \in \mathbb{F}[X_{1,\ldots,m}^{\leq d}]$,*

$$\sum_{\vec{\alpha} \in H^m} P(\vec{\alpha} + \vec{v}) = \kappa \cdot a_0^m \ ,$$

where κ is the coefficient of $X_1^{|H|-1} \cdots X_m^{|H|-1}$ in P, and a_0 is the (formal) linear term of the subspace polynomial $\prod_{h \in H}(X - h)$. In particular, if P has total degree strictly less than $m(|H| - 1)$, then the above sum evaluates to 0.

Proof. Without loss of generality, let $d := |H| - 1$. The proof is by induction on the number of variables m. When $m = 1$, we have that $P(X) = \sum_{j=0}^{d} \beta_j X^j$ for some $\beta_0, \ldots, \beta_d \in \mathbb{F}$. Then

$$\sum_{\alpha \in H} P(\alpha + v) = \sum_{\alpha \in H} \sum_{j=0}^{d} \beta_j (\alpha + v)^j = \sum_{j=0}^{d} \beta_j \sum_{\alpha \in H} (\alpha + v)^j = \beta_d a_0$$

where the final equality follows by [BC99, (Proof of) Theorem 1], and the fact that $d = |H| - 1$.

For the inductive step, assume the statement for $m - 1$ variables; we now prove that it holds for m variables as well. Let P_α denote P with the variable X_1 fixed to α; we have $P_\alpha(X_2, \ldots, X_m) = \sum_{\vec{e} \in \{0,\ldots,d\}^m} \beta_{\vec{e}} \cdot \alpha^{e_1} X_2^{e_2} \ldots X_m^{e_m}$. Next, apply the inductive hypothesis below in the second equality (with $m - 1$ variables) to obtain

$$\sum_{\vec{\alpha} \in H^m} P(\vec{\alpha} + \vec{v}) = \sum_{\alpha_1 \in H} \sum_{(\alpha_2,\ldots,\alpha_m) \in H^{m-1}} P_{\alpha_1 + v_1}(\alpha_2 + v_2, \ldots, \alpha_m + v_m) = \sum_{\alpha_1 \in H} a_0^{m-1} \kappa(\alpha_1 + v_1) \ ,$$

where $\kappa(X_1) := \sum_{j=0}^{d} \beta_{(j,d,\ldots,d)} X_1^j$. Applying the hypothesis again for 1 variable yields

$$\sum_{\alpha_1 \in H} a_0^{m-1} \kappa(\alpha_1 + v_1) = a_0^m \cdot \beta_{(d,\ldots,d)} \ ,$$

and the claim follows. □

Acknowledgments

We are grateful to Thomas Vidick for multiple technical and conceptual suggestions that greatly improved our results and their presentation, as well as for allowing us to include his proof of Theorem 7.2. We thank Claude Crépeau for useful comments on an earlier version of this paper. We also thank Zeph Landau, Chinmay Nirkhe, and Igor Shinkar for helpful discussions.

References

[AH91] William Aiello and Johan Håstad. "Statistical Zero-Knowledge Languages can be Recognized in Two Rounds". In: *Journal of Computer and System Sciences* 42.3 (1991). Preliminary version appeared in FOCS '87., pp. 327–345.

[AW09] Scott Aaronson and Avi Wigderson. "Algebrization: A New Barrier in Complexity Theory". In: *ACM Transactions on Computation Theory* 1.1 (2009), 2:1–2:54.

[BC99] Nigel P. Byott and Robin J. Chapman. "Power Sums over Finite Subspaces of a Field". In: *Finite Fields and Their Applications* 5.3 (July 1999), pp. 254–265.

[BCFGRS17] Eli Ben-Sasson, Alessandro Chiesa, Michael A. Forbes, Ariel Gabizon, Michael Riabzev, and Nicholas Spooner. "Zero Knowledge Protocols from Succinct Constraint Detection". In: *Proceedings of the 15th Theory of Cryptography Conference*. TCC '17. 2017, pp. 172–206.

[BCGV16] Eli Ben-Sasson, Alessandro Chiesa, Ariel Gabizon, and Madars Virza. "Quasilinear-Size Zero Knowledge from Linear-Algebraic PCPs". In: *Proceedings of the 13th Theory of Cryptography Conference*. TCC '16-A. 2016, pp. 33–64.

[Bel64] John S Bell. "On the Einstein Podolsky Rosen paradox". In: (1964).

[BFL91] László Babai, Lance Fortnow, and Carsten Lund. "Non-Deterministic Exponential Time has Two-Prover Interactive Protocols". In: *Computational Complexity* 1 (1991). Preliminary version appeared in FOCS '90., pp. 3–40.

[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*. STOC '91. 1991, pp. 21–32.

[BGKW88] 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*. STOC '88. 1988, pp. 113–131.

[BHZ87] Ravi B. Boppana, Johan Håstad, and Stathis Zachos. "Does co-NP have short interactive proofs?" In: *Information Processing Letters* 25.2 (1987), pp. 127–132.

[BJSW16] Anne Broadbent, Zhengfeng Ji, Fang Song, and John Watrous. "Zero-Knowledge Proof Systems for QMA". In: *Proceedings of the 57th Annual Symposium on Foundations of Computer Science*. FOCS '16. 2016, pp. 31–40.

[BS06] Eli Ben-Sasson and Madhu Sudan. "Robust locally testable codes and products of codes". In: *Random Structures and Algorithms* 28.4 (2006), pp. 387–402.

[CHTW04] Richard Cleve, Peter Hoyer, Benjamin Toner, and John Watrous. "Consequences and limits of nonlocal strategies". In: *Proceedings of the 19th IEEE Annual Conference on Computational Complexity*. 2004, pp. 236–249.

[DFKNS92] Cynthia Dwork, Uriel Feige, Joe Kilian, Moni Naor, and Shmuel Safra. "Low Communication 2-Prover Zero-Knowledge Proofs for NP". In: *Proceedings of the 11th Annual International Cryptology Conference*. CRYPTO '92. 1992, pp. 215–227.

[DFS04] Ivan Damgård, Serge Fehr, and Louis Salvail. "Zero-Knowledge Proofs and String Commitments Withstanding Quantum Attacks". In: *Proceedings of the 24th Annual International Cryptology Conference*. CRYPTO '04. 2004, pp. 254–272.

[DS98] Cynthia Dwork and Amit Sahai. "Concurrent Zero-Knowledge: Reducing the Need for Timing Constraints". In: *Proceedings of the 18th Annual International Cryptology Conference*. CRYPTO '98. 1998, pp. 442–457.

[For87] Lance Fortnow. "The Complexity of Perfect Zero-Knowledge (Extended Abstract)". In: *Proceedings of the 19th Annual ACM Symposium on Theory of Computing*. STOC '87. 1987, pp. 204–209.

[FS89] Uriel Feige and Adi Shamir. "Zero Knowledge Proofs of Knowledge in Two Rounds". In: *Proceedings of the 9th Annual International Cryptology Conference*. CRYPTO '89. 1989, pp. 526–544.

[GIMS10] Vipul Goyal, Yuval Ishai, Mohammad Mahmoody, and Amit Sahai. "Interactive locking, zero-knowledge PCPs, and unconditional cryptography". In: *Proceedings of the 30th Annual Conference on Advances in Cryptology*. CRYPTO'10. 2010, pp. 173–190.

[GKR15] Shafi Goldwasser, Yael Tauman Kalai, and Guy N. Rothblum. "Delegating Computation: Interactive Proofs for Muggles". In: *Journal of the ACM* 62.4 (2015), 27:1–27:64.

[GMR89] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. "The knowledge complexity of interactive proof systems". In: *SIAM Journal on Computing* 18.1 (1989). Preliminary version appeared in STOC '85., pp. 186–208.

[GMW91] Oded Goldreich, Silvio Micali, and Avi Wigderson. "Proofs that Yield Nothing But Their Validity or All Languages in NP Have Zero-Knowledge Proof Systems". In: *Journal of the ACM* 38.3 (1991). Preliminary version appeared in FOCS '86., pp. 691–729.

[GR17] Tom Gur and Ron D Rothblum. "A hierarchy theorem for interactive proofs of proximity". In: *Proceedings of the 8th Innovations in Theoretical Computer Science Conference*. Vol. 67. ITCS '17. 2017.

[GS06] Oded Goldreich and Madhu Sudan. "Locally testable codes and PCPs of almost-linear length". In: *Journal of the ACM* 53 (4 2006). Preliminary version in STOC '02., pp. 558–655.

[IKM09] Tsuyoshi Ito, Hirotada Kobayashi, and Keiji Matsumoto. "Oracularization and Two-Prover One-Round Interactive Proofs against Nonlocal Strategies". In: *Proceedings of the 24th IEEE Annual Conference on Computational Complexity*. CCC '09. 2009, pp. 217–228.

[IKPSY08] Tsuyoshi Ito, Hirotada Kobayashi, Daniel Preda, Xiaoming Sun, and Andrew Chi-Chih Yao. "Generalized Tsirelson Inequalities, Commuting-Operator Provers, and Multi-prover Interactive Proof Systems". In: *Proceedings of the 23rd Annual IEEE Conference on Computational Complexity*. CCC '08. 2008, pp. 187–198.

[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*. FOCS '12. 2012, pp. 243–252.

[JKRS09] Ali Juma, Valentine Kabanets, Charles Rackoff, and Amir Shpilka. "The Black-Box Query Complexity of Polynomial Summation". In: *Computational Complexity* 18.1 (2009), pp. 59–79.

[KKMTV11] Julia Kempe, Hirotada Kobayashi, Keiji Matsumoto, Ben Toner, and Thomas Vidick. "Entangled games are hard to approximate". In: *SIAM Journal on Computing* 40.3 (2011), pp. 848–877.

[Kob03] Hirotada Kobayashi. "Non-interactive Quantum Perfect and Statistical Zero-Knowledge". In: *Proceedings of the 14th Algorithms and Computation International Symposium*. 2003, pp. 178–188.

[Kob08] Hirotada Kobayashi. "General Properties of Quantum Zero-Knowledge Proofs". In: *Proceedings of the 5th Theory of Cryptography Conference*. TCC '08. 2008, pp. 107–124.

[KPT97] Joe Kilian, Erez Petrank, and Gábor Tardos. "Probabilistically checkable proofs with zero knowledge". In: *Proceedings of the 29th Annual ACM Symposium on Theory of Computing*. STOC '97. 1997, pp. 496–505.

[KR08] Yael Kalai and Ran Raz. "Interactive PCP". In: *Proceedings of the 35th International Colloquium on Automata, Languages and Programming*. ICALP '08. 2008, pp. 536–547.

[KV11] Julia Kempe and Thomas Vidick. "Parallel repetition of entangled games". In: *Proceedings of the 43rd ACM Symposium on the Theory of Computing*. STOC '11. 2011, pp. 353–362.

[LFKN92] Carsten Lund, Lance Fortnow, Howard J. Karloff, and Noam Nisan. "Algebraic Methods for Interactive Proof Systems". In: *Journal of the ACM* 39.4 (1992), pp. 859–868.

[Mei13] Or Meir. "IP = PSPACE Using Error-Correcting Codes". In: *SIAM Journal on Computing* 42.1 (2013), pp. 380–403.

[MW18] Sanketh Menda and John Watrous. "Oracle Separations for Quantum Statistical Zero-Knowledge". In: *CoRR* abs/1801.08967 (2018).

[NV18] Anand Natarajan and Thomas Vidick. "Two-Player Entangled Games are NP-Hard". In: *Proceedings of the 32nd Annual IEEE Conference on Computational Complexity*. CCC '18. 2018, 20:1–20:18.

[ON07] Tomohiro Ogawa and Hiroshi Nagaoka. "Making good codes for classical-quantum channel coding via quantum hypothesis testing". In: *IEEE Transactions on Information Theory* 53.6 (2007), pp. 2261–2266.

[OW93] Rafail Ostrovsky and Avi Wigderson. "One-Way Functions are Essential for Non-Trivial Zero-Knowledge". In: *Proceedings of the 2nd Israel Symposium on Theory of Computing Systems*. ISTCS '93. 1993, pp. 3–17.

[RRR16] Omer Reingold, Ron Rothblum, and Guy Rothblum. "Constant-Round Interactive Proofs for Delegating Computation". In: *Proceedings of the 48th ACM Symposium on the Theory of Computing*. STOC '16. 2016, pp. 49–62.

[RS05] Ran Raz and Amir Shpilka. "Deterministic polynomial identity testing in non-commutative models". In: *Computational Complexity* 14.1 (2005). Preliminary version appeared in CCC '04., pp. 1–19.

[RS96] Ronitt Rubinfeld and Madhu Sudan. "Robust Characterizations of Polynomials with Applications to Program Testing". In: *SIAM Journal on Computing* 25.2 (1996), pp. 252–271.

[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*. STOC '97. 1997, pp. 475–484.

[Unr12] Dominique Unruh. "Quantum Proofs of Knowledge". In: *Proceedings of the 31st Annual International Conference on Theory and Application of Cryptographic Techniques*. EUROCRYPT '12. 2012, pp. 135–152.

[Unr15] Dominique Unruh. "Non-Interactive Zero-Knowledge Proofs in the Quantum Random Oracle Model". In: *Proceedings of the 34th Annual International Conference on Theory and Application of Cryptographic Techniques*. EUROCRYPT '15. 2015, pp. 755–784.

[Vid11] Thomas Vidick. "The Complexity of Entangled Games". PhD thesis. University of California, Berkeley, 2011.

[Vid16] Thomas Vidick. "Three-player entangled XOR games are NP-hard to approximate". In: *SIAM Journal on Computing* 45.3 (2016), pp. 1007–1063.

[Wat02] John Watrous. "Limits on the Power of Quantum Statistical Zero-Knowledge". In: *Proceedings of the 43rd Symposium on Foundations of Computer Science*. FOCS '02. 2002, p. 459.

[Wat09] John Watrous. "Zero-Knowledge against Quantum Attacks". In: *SIAM Journal on Computing* 39.1 (2009), pp. 25–58.

[Win99] Andreas Winter. “Coding theorem and strong converse for quantum channels”. In: *IEEE Transactions on Information Theory* 45.7 (1999), pp. 2481–2485.

[Yue16] Henry Yuen. “A Parallel Repetition Theorem for All Entangled Games”. In: *Proceedings of the 43rd International Colloquium on Automata, Languages, and Programming*. ICALP ’16. 2016, 77:1–77:13.