Electronic Colloquium on Computational Complexity, Report No. 2 (2026)

Separation Results for Constant-Depth and Multilinear Ideal Proof Systems

Amik Raj Behera* Magnus Rahbek Dalgaard Hansen † Nutan Limaye ‡
Srikanth Srinivasan §

January 9, 2026

Abstract

In this work, we establish separation theorems for several subsystems of the Ideal Proof System (IPS), an algebraic proof system introduced by Grochow and Pitassi (J. ACM, 2018). Separation theorems are well-studied in the context of classical complexity theory, Boolean circuit complexity, and algebraic complexity.

In an important work of Forbes, Shpilka, Tzameret, and Wigderson (Theory of Computing, 2021), two proof techniques were introduced to prove lower bounds for subsystems of the IPS, namely the *functional method* and the *multiples method*. We use these techniques and obtain the following results.

1. **Hierarchy theorem for constant-depth IPS.** Recently, Limaye, Srinivasan, and Tavenas (J. ACM 2025) proved a hierarchy theorem for constant-depth algebraic circuits. We adapt the result and prove a hierarchy theorem for constant-depth IPS. We show that there is an unsatisfiable multilinear instance refutable by a depth-Δ IPS such that any depth-$(\Delta/10)$ IPS refutation for it must have superpolynomial size. This result is proved by building on the *multiples method*.

2. **Separation theorems for multilinear IPS.** In an influential work, Raz (Theory of Computing, 2006) unconditionally separated two algebraic complexity classes, namely multilinear NC^1 from multilinear NC^2. In this work, we prove a similar result for a well-studied fragment of multilinear-IPS.
   Specifically, we present an unsatisfiable instance such that its *functional refutation*, i.e., the unique multilinear polynomial agreeing with the inverse of the polynomial over the Boolean cube, has a small multilinear-NC^2 circuit. However, any multilinear-NC^1 IPS refutation ($\mathsf{IPS}_{\mathsf{LIN}}$) for it must have superpolynomial size. This result is proved by building on the

*Department of Computer Science, University of Copenhagen, Denmark. Supported by Srikanth Srinivasan's start-up grant from the University of Copenhagen. Email: ambe@di.ku.dk

†IT University of Copenhagen, Denmark, Email: ramh@itu.dk Supported by the Basic Algorithms Research Copenhagen (BARC), funded by VILLUM Foundation Grant 54451.

‡IT University of Copenhagen, Denmark, Email: nuli@itu.dk. Supported by Independent Research Fund Denmark (grant agreement No. 10.46540/3103-00116B) and is also supported by the Basic Algorithms Research Copenhagen (BARC), funded by VILLUM Foundation Grant 54451.

§Department of Computer Science, University of Copenhagen, Denmark. Supported by the European Research Council (ERC) under grant agreement no. 101125652 (ALBA). Email: srsr@di.ku.dk

ISSN 1433-8092

functional method.

Given a polynomial $p(\mathbf{x})$, let $\mathsf{Image}(p(\mathbf{x}))$ denote the set of values obtained when $p(\mathbf{x})$ is evaluated over the Boolean cube. Our crucial observation is that if the cardinality of this set is $\mathcal{O}(1)$, then the functional method and multiples method can be used to prove separation theorems for subsystems of the IPS. We obtain such polynomial instances by lifting the hard instances arising from algebraic circuit complexity with *addressing gadgets*.

Acknowledgments. The authors would like to thank Varun Ramanathan for helpful discussions during the early stages of the project.

Contents

1 Introduction

A proof system is defined by a collection of axioms together with a set of inference rules that determine how new statements can be derived from existing ones. The objective is to begin with the given axioms and apply these inference rules to derive theorems (or tautologies) within the system. A proof system is said to be *sound* if it proves only valid statements, and *complete* if every valid statement can be derived within it.

The field of *Propositional Proof Complexity* studies the comparative strength and efficiency of such systems in the propositional setting. A foundational result by Cook and Reckhow [CR79] established that if one could exhibit propositional tautologies that require exponentially large proofs (that is, proofs whose length—roughly corresponding to the number of inference steps—grows exponentially) in *every* propositional proof system, then this would separate the complexity classes NP and coNP. Thus, lower bounds in proof complexity are deeply connected to some of the central open problems in computational complexity theory.

In this work, we focus on *algebraic proof systems*, in which we consider unsatisfiable systems of polynomial equations, and reasoning proceeds through algebraic manipulations such as addition and multiplication of polynomials. Here, more specifically, we consider an algebraic proof system called the Ideal Proof System (IPS), which was introduced by Grochow and Pitassi [GP18]. In the last decade, different facets of this proof system have been investigated by a series of works [FSTW21; AF22; GHT22; ST21; HLT24a; LST25; BLRS25; EGLT25; CGMS25]. Our paper contributes to this line of research by studying separation theorems in this context.

Hierarchy theorems are a class of separation theorems that establish that more resources yield strictly more power. For instance, the classical Time Hierarchy Theorem [HS65] states that increasing the available running time strictly increases the computational power of a machine. Analogous results are known for several resources such as space [SHI65], circuit depth [Sip83; Has86], and circuit size [Juk12; Sha49]. Here, we raise the question about hierarchy theorems, and more generally we study separation theorems for different subsystems of the IPS.

In order to describe our results, we first start by giving a brief introduction to the Ideal Proof Systems Section 1.1. We then review some results from Algebraic Circuit Complexity in Section 1.2, which we will use crucially in our work. Our results and techniques are described in Section 1.3 and in Section 1.4.

1.1 Ideal Proof System

We begin by recalling the general framework of algebraic proof systems, focusing on the so-called *static* systems.[1] Let $\mathbf{x}$ denote the set of variables $\{x_1, x_2, \ldots, x_N\}$. Given a collection of polynomial axioms $f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x}) \in \mathbb{F}[\mathbf{x}]$, the goal is to certify that there is *no* Boolean assignment to the variables that simultaneously satisfies all the equalities $f_1(\mathbf{x}) = f_2(\mathbf{x}) = \cdots = f_m(\mathbf{x}) = 0$. To ensure that solutions are Boolean, the system is augmented with the *Boolean axioms* $\{x_i^2 - x_i = 0\}_{i\in[n]}$.

By Hilbert's Nullstellensatz, the unsatisfiability of this augmented system can be expressed al-

[1] In the literature, systems of this type are often referred to as static proof systems. Other variants, where proofs are given line-by-line, are known as dynamic proof systems. In this paper, we only consider static systems.

gebraically. Specifically, if the system has no common zero over $\mathbb{F}$, then there exist polynomials $A_1(\mathbf{x}), \ldots, A_m(\mathbf{x})$ and $B_1(\mathbf{x}), \ldots, B_N(\mathbf{x})$ such that

$$\sum_{i \in [m]} A_i(\mathbf{x}) \cdot f_i(\mathbf{x}) + \sum_{j \in [N]} B_j(\mathbf{x}) \cdot (x_j^2 - x_j) = 1. \tag{1}$$

This identity serves as a *refutation* (or *proof*) of the original system. The complexity of such a refutation is measured in terms of the complexity of the polynomials $\{A_i\}$ and $\{B_j\}$.

In the *Ideal Proof System (*IPS*)* introduced by Grochow and Pitassi [GP18], the polynomials $A_i(\mathbf{x})$ and $B_j(\mathbf{x})$ are represented by algebraic circuits. This gives rise to natural complexity parameters such as the *circuit size* and *circuit depth* of IPS proofs. We now formally define the ideal proof system.

Definition 1.1 (Ideal Proof System [GP18])**.** *Let $f_1, \ldots, f_m \in \mathbb{F}[x_1, \ldots, x_n]$ be a system of unsatisfiable polynomials over the Boolean cube $\{0,1\}^n$. In other words, there is no Boolean assignment $\mathbf{a} \in \{0,1\}^n$ to the variables $x_1, \ldots, x_n$ so that $f_i(\mathbf{a}) = 0$ for all $i \in [m]$.*

Given a class of algebraic circuits $\mathcal{C}$, a $\mathcal{C}$-IPS refutation *of the system of equations defined by $f_1, \ldots, f_m$ is an algebraic circuit $C \in \mathcal{C}$ in variables $x_1, \ldots, x_n, y_1, \ldots, y_m, z_1, \ldots, z_n$ such that*

- $C(\mathbf{x}, \mathbf{0}, \mathbf{0}) = 0$, *and*
- $C(\mathbf{x}, f_1, \ldots, f_m, x_1^2 - x_1, \ldots, x_n^2 - x_n) = 1$.

The size of the refutation is the size of the circuit C.

Further, if the circuit C has individual degree at most 1 in the variables $\mathbf{y}$ and $\mathbf{z}$, then we say that C is a $\mathcal{C}$-$\mathsf{IPS}_{\mathsf{LIN}}$ refutation. *If the circuit C has individual degree at most 1 in the variables $\mathbf{y}$ (but not necessarily in $\mathbf{z}$), then C is said to be a* $\mathcal{C}$-$\mathsf{IPS}_{\mathsf{LIN}'}$ refutation.

The general IPS where the class $\mathcal{C}$ is allowed be to be an algebraic circuit can polynomially simulate Extended Frege [GP18], one of the strongest known propositional proof systems. Moreover, establishing lower bounds for these kind of general IPS would imply strong algebraic circuit lower bounds, a central open problem in algebraic complexity.

While this continues to be an ambitious open problem we have many compelling new lower bound results for several restricted classes $\mathcal{C}$ such as roABPs, constant-depth circuits, and multilinear formulas [FSTW21; GHT22; HLT24b; BLRS25; EGLT25].

These lower bounds were established by using the already known lower bounds for the corresponding models of computation in algebraic complexity. We are also inspired by this framework. Namely, we use the separation results and hierarchy theorems from algebraic complexity theory to obtain similar results for the IPS. We now review the known separation results.

1.2 Algebraic Circuit Complexity

We start by recalling some of the standard models of computation relevant to our results.

Algebraic circuits, formulas, constant-depth circuits, multilinear polynomials and circuits. An *algebraic circuit* is a directed acyclic graph in which each node either computes a sum (or a linear

combination) of its inputs, or a product of its inputs. The leaf nodes are either variables or constants. The size of an algebraic circuit is the number of edges or wires in the circuit, and the depth of an algebraic circuit is the longest path from a leaf node (a source) to the output node (a sink). An *algebraic formula* is an algebraic circuit where the output of each node feeds into at most one other node; in other words, the underlying graph of an algebraic formula is a tree. An algebraic circuit/formula is said to be constant-depth circuit/formula, if its depth is a fixed constant independent of other parameters.

A polynomial $f(\mathbf{x}) \in \mathbb{F}[x_1, \ldots, x_n]$ is *multilinear* if in every monomial of the polynomial, the degree of any variable is at most 1. An algebraic circuit/formula is multilinear if every gate computes a multilinear polynomial. An algebraic circuit is syntactically multilinear if polynomials computed by the children of any multiplication gate compute polynomials on disjoint sets of variables.

1.2.1 Separation results

Our work relies heavily on the separation results known in algebraic complexity theory. Our result related to multilinear IPS is based on the following multilinear separation result.

Multilinear formulas vs circuits. One of the celebrated results in algebraic complexity is the separation between multilinear formulas and multilinear circuits. The result was established in an influential work of Raz [Raz04], which presented a polynomial that is computed by a polynomial sized multilinear circuit, but any multilinear formula for it requires superpolynomial size.

The key idea involves coming up with a complexity measure for polynomials, which attains a large value for the hard polynomial, but it is considerably small for all multilinear formulas of small size. The measure from [Raz04] is defined as follows.

Definition 1.2 (Rank measure [Raz04])**.** *Let $\mathbf{x} = \{x_1, \ldots, x_{2n}\}$. Let $\mathbf{y} \cup \mathbf{z}$ be an equipartition of $\mathbf{x}$, i.e. $|\mathbf{y}| = |\mathbf{z}|$. For a given polynomial $f(\mathbf{x})$, let $M_{\mathbf{y},\mathbf{z}}(f)$ be a matrix with rows labeled by multilinear monomials in $\mathbf{y}$ variables and columns labeled by multilinear monomials in $\mathbf{z}$ variables. For a monomial $m_{\mathbf{y}}$ in $\mathbf{y}$ variables and $m_{\mathbf{z}}$ in $\mathbf{z}$ variables, the $M_{\mathbf{y},\mathbf{z}}(f)[m_{\mathbf{y}}, m_{\mathbf{z}}]^{th}$ entry of the matrix is the coefficient of the monomial $m_{\mathbf{y}} \cdot m_{\mathbf{z}}$ in f. The measure is the rank of this matrix.*

We will say that a polynomial f is full-rank with respect to a partition $\mathbf{y}, \mathbf{z}$ if the rank of $M_{\mathbf{y},\mathbf{z}}(f)$ is full, i.e. 2^n.

It was shown by Raz [Raz04] that a multilinear formula computing any full-rank polynomial $f(\mathbf{x})$ requires size $n^{\Omega(\log n)}$. In our work, we build on the full-rank polynomial defined in a subsequent work of Raz and Yehudayoff [RY08].

Constant-depth hierarchy theorem In our work we establish a constant-depth hierarchy theorem for constant-depth IPS. For this, the starting point is the constant-depth hierarchy theorem by Limaye, Srinivasan, and Tavenas [LST21]. For every depth Δ, they design a polynomial that is computable by polynomial size depth Δ circuits but any circuit of depth even one smaller than Δ requires superpolynomial size for it. As circuits can be converted to formulas with polynomial blow-up when the depth is constant, we state the formula version of the hierarchy theorem below.

Formally, it states the following.

Theorem 1.3 (Constant-depth algebraic formulas hierarchy)**.** *[LST21, Theorem 5]. For every depth parameter $\Delta = \mathcal{O}(1)$, there exists an explicit set-multilinear polynomial $Q_\Delta \in \mathbb{F}[x_1, \ldots, x_n]$ such that:*

1. *There exists a constant-free[2] algebraic formula with input gates carrying labels from $\mathbf{x} \cup \{0, 1\}$ which computes $Q_\Delta(\mathbf{x})$ in depth Δ and size s.*
2. *Any algebraic formula computing $Q_\Delta(\mathbf{x})$ in depth $(\Delta/2 - 1)$ requires size $s^{\omega(1)}$.*

Remark 1.4. *In [LST21], a tighter separation is obtained. Namely, the depth hierarchy separates two consecutive depths, Δ vs. $\Delta - 1$. However, the formulas arising from this are not constant-free. Depth hierarchy for constant-free formulas can be obtained by a slight loss in parameters, as mentioned in the statement above.*

We are now ready to state our contributions.

1.3 Results and Techniques: the constant-depth IPS hierarchy theorem

As our first result, we prove a depth-hierarchy theorem for constant-depth IPS. More specifically, we prove the following theorem.

Theorem 1.5 (Constant-depth IPS hierarchy)**.** *Let $\mathbb{F}$ be a field of characteristic zero. The following holds for every growing parameter $N \in \mathbb{N}$ and a depth parameter $\Gamma \in \mathbb{N}$ where $\Gamma = \mathcal{O}(1)$. For every depth parameter Γ, there exists a multilinear polynomial $f_\Gamma \in \mathbb{F}[x_1, \ldots, x_N]$ which is unsatisfiable over $\{0, 1\}^N$ (i.e. there exists no $\mathbf{a} \in \{0, 1\}^N$ for which $f_\Gamma(\mathbf{a}) = 0$) such that the following two conditions hold:*

1. *There exists an IPS refutation for $f_\Gamma(\mathbf{x})$ in depth Γ and size $\mathcal{O}(s^5)$.*
2. *Any IPS refutation for $f_\Gamma(\mathbf{x})$ with depth $\leqslant \Gamma/10$ requires size $s^{\omega(1)}$.*

To describe the proof strategy, we prove the above theorem for a simpler case. Let $Q_\Delta(\mathbf{x})$ be the polynomial used by [LST21] in Theorem 1.3. Now consider $g_\Delta(\mathbf{x}, y)$ defined as $Q_\Delta(\mathbf{x}) \cdot y \cdot (1 - y)$, where y is a new variable. First, observe that the polynomial evaluates to 0 over the Boolean cube. In fact, this is true if we only consider the Boolean evaluations of y. Therefore, $g_\Delta(\mathbf{x}, y) - 1$ is unsatisfiable. Moreover, $g_\Delta(\mathbf{x}, y) - 1 \equiv 1 \mod y^2 - y$. And it is easy to see that $g_\Delta(\mathbf{x}, y)$ has the same upper bound as $Q_\Delta(\mathbf{x})$. Thus, we get the upper bound.

For the lower bound, we will use the multiples method. The method was introduced in [FSTW21] and it has been used successfully for IPS lower bounds in [FSTW21; AF22; And25]. We now describe how one can use this method to obtain a lower bound.

Consider $g_\Delta(\mathbf{x}, y) - 1$. Using Theorem 1.3, we know that it does not have polynomial sized formulas of depth $(\Delta/2 - 1)$. Moreover, due to the recent work on factors of constant-depth formulas [BKRRSS25], we also know that every multiple of the polynomial of depth $(\Delta/2 - \mathcal{O}(1))$ must have superpolynomial size. That is, the polynomial $g_\Delta(\mathbf{x}, y) - 1$ and all its multiples are hard for

[2] A circuit or formula $C(\mathbf{x})$ is constant free if it has no constants except at the inputs where all input gates have labels from $\mathbf{x} \cup \{-1, 0, 1\}$.

depth $(\Delta/2 - \mathcal{O}(1))$. This property suffices for the multiples method to be applicable, as we explain next. Specifically, our polynomial system is $f_\Delta = g_\Delta(\mathbf{x}, y) - 1$, $\{x_i^2 - x_i\}_{i\in[N]}$, and $\{y^2 - y\}$. IPS refutation is such that $C(\mathbf{x}, y, u, \mathbf{0}, 0) = 0$ and $C(\mathbf{x}, y, f_\Delta, \mathbf{x}^2 - \mathbf{x}, y^2 - y) = 1$, where $\mathbf{x}^2 - \mathbf{x}$ denotes $\{x_i^2 - x_i\}_i$.

We now express $C(\mathbf{x}, y, f_\Delta, \mathbf{x}^2 - \mathbf{x}, y^2 - y)$ as a univariate in f_Δ and we obtain

$$\sum_{i\geqslant 1} C_i(\mathbf{x}, y, \mathbf{x}^2 - \mathbf{x}, y^2 - y) f_\Delta^i = 1 - C(\mathbf{x}, y, 0, \mathbf{x}^2 - \mathbf{x}, y^2 - y)$$

for some $C_i s$. This shows that a multiple of f_Δ has the same complexity as $C(\mathbf{x}, y, 0, \mathbf{x}^2 - \mathbf{x}, y^2 - y)$. But we know that all the multiples of f_Δ are hard. This gives an IPS lower bound.

Remark 1.6. *The above proof outline basically proves Theorem 1.5 when the hard instance is non-multilinear. Note that the polynomial Q_Δ is multilinear, but the hard instance is non-multilinear in y. We extend the ideas presented in the outline above and obtain a hard instance that is multilinear.*

A hard multilinear instance. The technical challenge in our work is designing an instance that is *multilinear*. We seek such an instance for the following reason: In algebraic proof complexity, the goal is to find an instance that is itself quite *easy* to compute, but its refutation is hard. There are many different ways to quantify easiness. However, one of the standard ways is to ask for a hard instance to be multilinear. Almost all the known hard instances in this literature are multilinear. (See for instance [FSTW21; GHT22; HLT24b]). In fact, the hard instance naturally arising from the algebraic encoding of CNF SAT is also multilinear.

There are some challenges that arise when we require a multilinear instance.

Making the instance multilinear. As mentioned above, the hard polynomial Q_Δ from [LST21] is already multilinear. However, unfortunately, we do not know how to upper bound the complexity of IPS refutations of Q_Δ itself with respect to depth Δ. To fix this, we modify Q_Δ so that the (new) hard instance takes only $\mathcal{O}(1)$ distinct values over Boolean evaluations. We rely on this fact for our upper bound (see the discussion below). Our upper bound proof is further simplified if the polynomial takes only 0-1 values over the Boolean cube. So, we bake these two properties into the design of the polynomial: (a) it is multilinear, and (b) it takes only Boolean values over the Boolean hypercube.

We start with Q_Δ as in Theorem 1.3 and take the constant-depth formula implementation for it. In this formula, we introduce *an addressing gadget* for each + gate. An addressing gadget is a multilinear polynomial that works like a *multiplexer*. For a 0-1 values as inputs to the gadget polynomial, it *activates* one of the inputs to the plus gate and *suppresses* all the other inputs. As a result, if we inductively maintain 0-1 evaluations for all the gates over the Boolean cube, the gadget allows us to propagate this property to the next gate.

Proving the upper bound. We work with the formula C that *computes* our hard polynomial instance. By construction of the polynomial, we have the guarantee that every gate in the formula evaluates to 0 or 1 over the hypercube. Using this fact, we prove by induction on the depth of the formula that for a gate g, the polynomial $g^2 - g$ is in the ideal generated by $\{g_i^2 - g_i\}_i$ and the Boolean axioms, where $\{g_1, \ldots, g_t\}$ are the inputs to the gate g. This suffices to obtain the overall upper bound.

Proving the lower bound. The lower bound proof proceeds by observing that there exists an assignment to the gadget variables such that under that assignment, the hard instance becomes equal to the hard polynomial Q_Δ from Theorem 1.3. As this polynomial and all its multiples are hard (due to [BKRRSS25]), we obtain our lower bound using the multiples method.

The advantage of using the multiples method (instead of the functional) is that we obtain the lower bound theorem for IPS and not just for the more restrictive $\mathsf{IPS}_{\mathsf{LIN}}$.

1.4 Results and Techniques: multilinear-NC^1 vs. multilinear-NC^2

In this section, we state our separation theorem for a multilinear-IPS system. We say that for a polynomial instance $Q(\mathbf{x})$ unsatisfiable over the Boolean cube, a functional refutation is a polynomial $G(\mathbf{x})$ such that $G(\mathbf{x}) \cdot Q(\mathbf{x}) \equiv 1 \mod \mathbf{x}^2 - \mathbf{x}$. Further, we will say that it is a multilinear functional refutation is $G(\mathbf{x})$ is multilinear. We prove the following theorem.

Theorem 1.7 (multilinear NC^1 vs multilinear NC^2-IPS)**.** *Fix a field, $\mathbb{F}$ of characteristic 0. For every growing parameter $N \in \mathbb{N}$, there is a multilinear polynomial $Q \in \mathbb{F}[x_1, \ldots, x_N]$ which is unsatisfiable over $\{0,1\}^N$ such that*

1. *There is a multilinear functional refutation for $Q(\mathbf{x})$, say $G(\mathbf{x})$, computable by a syntactic multilinear circuit of polynomial size and $O(\log^2 N)$ depth.*
2. *Any multilinear-NC^1-$\mathsf{IPS}_{\mathsf{LIN}'}$ for it requires size $N^{\Omega(\log N)}$.*

Remark 1.8. *Note that in the above theorem the lower bound holds for multilinear-NC^1-$\mathsf{IPS}_{\mathsf{LIN}'}$. However, the upper bound is not multilinear-NC^2-$\mathsf{IPS}_{\mathsf{LIN}'}$. Instead, we only get that the refutation has multilinear-NC^2 circuits modulo the Boolean axioms. We do not get a bound on the complexity of the refutations of the Boolean axioms. In spite of this, we believe that the above result gives something we did not know before.*

- *Lower bounds for multilinear-NC^1-$\mathsf{IPS}_{\mathsf{LIN}'}$ are known since the work of [FSTW21]. Their hard instance is a* lifted subset-sum*, i.e. $f(\mathbf{x}, \mathbf{z}) = \sum_{i,j} z_{i,j} x_i x_j$. We observe that its functional refutation is quite hard. Specifically, it encodes the Clique polynomial over the Boolean cube. This means that it cannot have small functional refutations unless VP equals VNP. (See Appendix A).*
- *It is known that there are subsystems of multilinear-$\mathsf{IPS}_{\mathsf{LIN}}$ and multilinear-$\mathsf{IPS}_{\mathsf{LIN}'}$ that can refute interesting unsatisfiable instances (Section 4, [FSTW21]). For example, they can refute the subset-sum instances of the type $\sum_i \alpha_i x_i - \beta$, where α_is are $O(1)$ and β is chosen such that the instance becomes unsatisfiable*[3]*. While such instances have multilinear upper bounds, the upper bound proofs seem to heavily rely on the fact that the subset-sum polynomial has degree 1. Consider a simple instance $xy = 2$. This is an unsatisfiable instance over the Boolean cube. Here is one of its refutations:*

$$x^2y^2 - xy = x^2y^2 - x^2y + x^2y - xy = x^2(y^2 - y) + y(x^2 - x).$$

Notice that the refutation for the Boolean axiom $y^2 - y$ is not multilinear. (There is another refutation for the same and in that, the refutation of the other Boolean axiom is not multilinear.) In fact, any refutation of this example is not multilinear. (See Example 4.7

[3] They can allow slightly general α_is. See Proposition 4.15 from [FSTW21] for more details.

in [FSTW21].) This gives an indication that a degree-2 (or degree-greater-than-2) hard instance may not necessarily have multilinear proofs.

- *Our lower bounds are obtained using the functional method. This ensures that the hardness of our instance can be ascribed to the hardness of refuting the instance irrespective of the complexity of the refutations of the Boolean axioms. Thus, the result achieves a separation for the functional refutation of our instance.*

To describe the components of the proofs, we start with a very simple example. Let $z \in \{0,1\}$. In this case, it is easy to see that $2 - z$ is unsatisfiable and $2 - z \times \frac{1+z}{2} \equiv 1$ modulo $z^2 - z$. That is, $\frac{1+z}{2}$ is a refutation of $2 - z$ modulo the Boolean axioms. We make use of this idea in our proof. In order to prove the theorem, we again design a polynomial $p(\mathbf{x})$ such that it evaluates to 0 or 1 over the Boolean cube. Then, our unsatisfiable instance is $2 - p(\mathbf{x})$ and its functional refutation is $(1 + p(\mathbf{x}))/2$.

If we can design a multilinear polynomial such that

- it is computed by multilinear NC^2 circuits,
- any multilinear NC^1 circuit for it requires $N^{\Omega(\log N)}$ size,
- and it takes only Boolean values over the Boolean cube

then we get the separation. From a famous work of Raz [Raz04] we get a polynomial that satisfies the first two properties listed above. A subsequent work of Raz and Yehudayoff [RY08] also gives another candidate polynomial. Unfortunately, neither of them have the third property. We tweak the polynomial from [RY08] using the addressing gadgets to ensure that we get a multilinear polynomial with all these properties.

Applicability of the technique. The proof method used for proving Theorem 1.7 points to its applicability to other scenarios. For example, the same proof method can be applied in the context of the constant-depth hierarchy theorem (as in Theorem 1.5). The upper bound stays as is, but the lower bound is obtained using the method described above. This will work and will give a lower bound for $\mathsf{IPS}_{\mathsf{LIN}}$ instead of a lower bound for IPS.

There are other separation results known in algebraic complexity, especially in the multilinear setting. For example, results of [RY09; CELS18]. Our proof method is likely to be applicable in all these settings to obtain separation results originating for different sub-systems of the IPS from these separation results, just like we proved Theorem 1.7 from the separation results of Raz [Raz04; RY08].

2 Constant-depth Hierarchy

In this section, we will prove Theorem 1.5. To do so, we start with the hard polynomials from Theorem 1.3 and modify them by using addressing gadgets.

Throughout this section, we will assume without loss of generality that every algebraic formula C is layered and has alternating addition and multiplication gates, with the top gate being an addition gate. For every gate g in a formula $C(x_1, \ldots, x_N)$,

- $f_g(x_1, \ldots, x_N)$ will denote the polynomial computed at the gate g.
- Let depth(g) denote the depth of gate g, i.e. the length of the longest path from inputs to the gate. Let size(g) denote the number of wires in the sub-formula rooted at g. Finally, let fanin(g) denote the fan-in of gate g.

2.1 Adding addressing gadgets at sum gates

In this subsection, we define a modification for any given algebraic formula, ensuring that the new formula is a $\{0,1\}$-valued on Boolean inputs. Furthermore, the polynomial computed by the original formula can be easily retrieved from the new formula via partial evaluation of its variables.

Definition 2.1. *Let $n \in \mathbb{N}$. For each $0 \leqslant j \leqslant n$, let $t_n \in \mathbb{N}$ denote the smallest t_n such that $2^{t_n} > n$. Let $B_{n,0}(j) \subseteq \{0, 1, ..., t_n\}$ denote the indices which are 0 in the binary representation of $j + 2^{t_n}$. Similarly, let $B_{n,1}(j) \subseteq \{0, 1, ..., t_n\}$ denote the set of indices which are 1 in the binary representation of $j + 2^{t_n}$.*

The addressing gadget *of j in n is defined as*

$$A_{n,j}(y_0, ..., y_{t_n}) = \prod_{i \in B_{n,0}(j)} (1 - y_i) \prod_{i \in B_{n,1}(j)} y_i.$$

Note that $A_{n,j}$ uses the same set of variables $\{y_0, ..., y_{t_n}\}$ exactly once for all $0 \leqslant j \leqslant n$ since $j + 2^{t_n}$ always uses exactly $t_n + 1$ bits for $j < 2^{t_n}$. In particular, $A_{n,j}$ is multilinear.

Lemma 2.2. *Let $n, j \in \mathbb{N}$ with $0 \leqslant j \leqslant n$ and let $(b_0, ..., b_{t_n}) \in \{0,1\}^{t_n+1}$, with t_n as defined in Definition 2.1. Then the following is true over any field, $\mathbb{F}$, of characteristic $p \neq 2$:*

1. $A_{n,j}(b_0, ..., b_{t_n}) = \begin{cases} 1 & \text{if } b_i = 0 \text{ for all } i \in B_{n,0}(j) \text{ and } b_i = 1 \text{ for all } i \in B_{n,1}(j), \\ 0 & \text{for all other choices of } (b_0, ..., b_{t_g}) \in \{0,1\}^{t_g+1} \end{cases}$
2. $A_{n,j}(\frac{1}{2}, \frac{1}{2}, ..., \frac{1}{2}, 2^{t_n}) = 1$.

Proof. 1. This is clear by construction.

2. Since the bit corresponding to y_{t_n} is always 1 in the binary representation of $j + 2^{t_n}$, we have that y_{t_n} is a factor of $A_{n,j}$. Thus the evaluation becomes

$$\left(\prod_{i \in B_{n,0}(j)} \left(1 - \frac{1}{2}\right) \prod_{i \in B_{n,1}(j) \setminus \{t_n\}} \frac{1}{2} \right) \cdot 2^{t_n} = \frac{1}{2^{t_n}} \cdot 2^{t_n} = 1,$$

since $|B_{n,0}(j)| + |B_{n,1}(j) \setminus \{t_n\}| = t_n$.

■

The following lemma shows how these addressing gadgets are applied:

Lemma 2.3. *Let $C(\mathbf{x})$ be a constant-free formula of size s and depth Δ computing some polynomial $f(\mathbf{x})$. We construct a new formula $C'(\mathbf{x}, \mathbf{y})$ computing a new polynomial, $f'(\mathbf{x}, \mathbf{y})$, as follows:*

For any addition gate,

$$g(\mathbf{x}) = \sum_{j=0}^{\text{fanin}(g)-1} g_j(\mathbf{x})$$

of $C(\mathbf{x})$, we replace g by the subcircuit,

$$g'(\mathbf{x}, \mathbf{y}_g) = \sum_{j=0}^{\text{fanin}(g)-1} g_j(\mathbf{x}) \cdot A_{g,j}(\mathbf{y}_g),$$

where we abuse notation and write $A_{g,j} := A_{\text{fanin}(g)-1,j}$ and $t_g := t_{\text{fainin}(g)-1}$ and where $\mathbf{y}_g = \{y_{g,0}, ..., y_{g,t_g}\}$ is a fresh set of variables for each addition gate, g. Then

$$\mathbf{y} = \bigcup_{g \text{ addition gate in } C(\mathbf{x})} \mathbf{y}_g,$$

with $|\mathbf{y}| = \mathcal{O}(s \log s)$. We leave multiplication gates unchanged.

Then the following are true over any field, $\mathbb{F}$, of characteristic $p \neq 2$:

1. *$g'(\mathbf{x}, \mathbf{b}) = g_j(\mathbf{x})$ if $\mathbf{b}$ is the binary representation of $j + 2^{t_g}$ as a vector.*
2. *$f'(\mathbf{a}, \mathbf{b}) \in \{-1, 0, 1\}$ for any choice of $(\mathbf{a}, \mathbf{b}) \in \{0, 1\}^{|\mathbf{x}|+|\mathbf{y}|}$.*
3. *There exists $\mathbf{b} \in \mathbb{F}^{|\mathbf{y}|}$ such that $f'(\mathbf{x}, \mathbf{b}) = f(\mathbf{x})$.*
4. *$C'(\mathbf{x}, \mathbf{y})$ is of size $\mathcal{O}(s \log s)$ and depth at most $2\Delta + 2$.*

Proof.

1. This follows directly from Lemma 2.2.

2. This follows directly from part 1 and induction on the circuit layers. The base case follows from the assumption that C is constant-free.

3. For each addition gate, g', we let

$$\mathbf{b}_g = \left(\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2}, 2^{t_g}\right).$$

   Then by Lemma 2.2 every addition gate, g', under this evaluation becomes

$$g'(\mathbf{x}, \mathbf{b}_g) = \sum_{j=0}^{\text{fanin}(g)-1} g_j(\mathbf{x}) \cdot A_{g,j}(\mathbf{b}_g) = \sum_{j=0}^{\text{fanin}(g)-1} g_j(\mathbf{x}) = g(\mathbf{x}).$$

4. Let n_g denote the fanin of gate g (addition or multiplication). Since $A_{g,j}$ can be computed by formula of size $\mathcal{O}(\log n_g)$ and as $n_g \leqslant s$ for any g, we get that C' has a formula size at most $\mathcal{O}(s \log s)$.

   For each addition gate, g, we need to add a multiplication layer to multiply all the $g_j \cdot A_{g,j}$. Since $A_{g,j}$ has depth 2, we get a depth of at most $2\Delta + 2$.

■

Only part 3 of Lemma 2.3 requires $\mathbb{F}$ to be of characteristic $p \neq 2$. The rest of the statement holds true over *any* field.

Remark 2.4. *If the input gates of $C(\mathbf{x})$ carry labels from $\mathbf{x} \cup \{0,1\}$, then $f'(\mathbf{a},\mathbf{b}) \in \{0,1\}$ for any choice of $(\mathbf{a},\mathbf{b}) \in \{0,1\}^{|\mathbf{x}|+|\mathbf{y}|}$. In particular, this applies to the formula of Q_Δ from Theorem 1.3.*

2.2 Proof

Proof of Theorem 1.5. Let $\mathbb{F}$ be a field of characteristic zero and fix any depth $\Delta \in \mathbb{N}$. Let $Q_\Delta \in \mathbb{F}[x_1,\ldots,x_n]$ be the polynomial from [LST21, Theorem 2.1] satisfying the following conditions:

1. There is a constant-free algebraic formula computing Q_Δ in depth Δ and size s. Denote this by $C(x_1,\ldots,x_n)$.
2. For every algebraic formula computing Q_Δ with depth $\Delta/2-1$ requires size $s^{\omega(1)}$.

Unsatisfiable instance. Let $C(\mathbf{x})$ be the formula of depth Δ computing $Q_\Delta(\mathbf{x})$. Let $C'(\mathbf{x},\mathbf{y})$ denote the formula we get after applying the process mentioned in Lemma 2.3 to $C(\mathbf{x})$, and let $f'_\Delta(\mathbf{x},\mathbf{y})$ denote the polynomial computed by $C'(\mathbf{x},\mathbf{y})$. By Lemma 2.3, C' is of depth $2\Delta+2$ and size $\mathcal{O}(s\log s)$.

Define the polynomial f_Δ as

$$f_\Delta(\mathbf{x},\mathbf{y}) \;:=\; f'_\Delta(\mathbf{x},\mathbf{y}) - 2.$$

Let $(\mathbf{a},\mathbf{b}) \in \{0,1\}^{|\mathbf{x}|+|\mathbf{y}|}$ be any Boolean assignment to the variables in f'_Δ. By Lemma 2.3 (and Remark 2.4), we get that $f'_\Delta(\mathbf{a},\mathbf{b}) \in \{0,1\}$, so $f_\Delta(\mathbf{a},\mathbf{b}) \in \{-2,-1\}$. Hence, $f_\Delta = 0$ is not satisfiable over the Boolean cube. Let C_Δ denote the formula for f_Δ of depth $2\Delta+2$ and size $s_\Delta = O(s\log s)$.

Now we show that f_Δ satisfies the two properties stated in Theorem 1.5. That is, we prove upper and lower bounds on the complexity of the refutation of f_Δ.

2.2.1 Upper bound on the refutation of f_Δ

This section is dedicated to the proof of the following lemma.

Lemma 2.5 (Upper Bound)**.** *Let $f_\Delta \in \mathbb{F}[\mathbf{x},\mathbf{y}]$ be as defined above. There exists a constant-depth* IPS *refutation of depth $\Delta' = 4\Delta+6$ and size $s' \leqslant 100 s_\Delta^5$.*

In order to prove the lemma, we use induction on the structure of C_Δ. We prove the following inductive lemma, which implies Lemma 2.5.

Lemma 2.6. *Let g be any gate in C_Δ. Then,*

$$g^2 - g = \sum_{i=1}^{n} E_{g,i}\cdot(x_i^2 - x_i) + \sum_{j=1}^{m} F_{g,j}\cdot(y_j^2 - y_j),$$

where m is the number of $\mathbf{y}$ variables and

- $\mathrm{size}(E_{g,i})$, $\mathrm{size}(F_{g,i})$ *is at most* $100\cdot(\mathrm{size}(g))^4$,

- *and* $\mathrm{depth}(E_{g,i})$, $\mathrm{depth}(F_{g,i})$ *is at most* $2 \cdot \mathrm{depth}(g)$.

We assume Lemma 2.6 and prove Lemma 2.5. We know that f'_Δ can be computed by a formula of depth $2\Delta + 2$ and size s_Δ. Using Lemma 2.6, we know that there exists polynomials E_j and F_j such that

$$(f'_\Delta)^2 - f'_\Delta \;=\; \sum_{j=1}^{n} E_j(\mathbf{x},\mathbf{y}) \cdot (x_j^2 - x_j) \;+\; \sum_{j=1}^{m} F_j(\mathbf{x},\mathbf{y}) \cdot (y_j^2 - y_j),$$

where

- For every j, the polynomial E_j can be computed by a formula of depth $2(2\Delta + 2)$ and size $100 \cdot s_\Delta^4$.
- For every j, the polynomial F_j can be computed by a formula of depth $2(2\Delta + 2)$ and size $100 \cdot s_\Delta^4$.

As $f_\Delta = f'_\Delta - 2$, we get,

$$\frac{-1}{2}\left((f'_\Delta(\mathbf{x},\mathbf{y}) + 1) \cdot f_\Delta(\mathbf{x},\mathbf{y}) \;+\; \sum_{j=1}^{n} E_j(\mathbf{x},\mathbf{y}) \cdot (x_j^2 - x_j) \;+\; \sum_{j=1}^{m} F_j(\mathbf{x},\mathbf{y}) \cdot (y_j^2 - y_j)\right) \;=\; 1.$$

Thus we have an IPS refutation for f_Δ of depth $2(2\Delta + 2) + 2$ and size at most $s_\Delta + O(1) + (n \cdot (100 \cdot s_\Delta^4 + O(1))) + (m \cdot (100 \cdot s_\Delta^4 + O(1)))$. As both m and n are bounded by s_Δ, we get that this quantity is bounded by $100 s_\Delta^5$. This completes the proof[4] of Lemma 2.5. In what follows, we prove Lemma 2.6.

Proof of Lemma 2.6. We will prove the statement by induction on the structure of C_Δ. The base case is trivial. For the induction step, we have two cases: either g is a $\times$ gate or a $+$ gate.

<u>case 1:</u> $g = \prod_{\ell=0}^{r-1} g_\ell$.
In this case,

$$\begin{aligned} g^2 - g =& (g_0 \cdot g_1 \cdot \ldots \cdot g_{r-1})^2 - (g_0 \cdot g_1 \cdot \ldots \cdot g_{r-1}) \\ =& (g_0 \cdot g_1 \cdot \ldots \cdot g_{r-1})^2 - g_0(g_1 \cdot \ldots \cdot g_{r-1})^2 + g_0(g_1 \cdot \ldots \cdot g_{r-1})^2 - (g_0 \cdot g_1 \cdot \ldots \cdot g_{r-1}) \\ =& (g_0^2 - g_0)(g_1 \cdot \ldots \cdot g_{r-1})^2 + \big((g_1 \cdot \ldots \cdot g_{r-1})^2 - (g_1 \cdot \ldots \cdot g_{r-1})\big) \end{aligned}$$

Using the same idea as above, i.e., a telescoping summation, we get

$$g^2 - g = \sum_{\ell=0}^{r-1} \prod_{t<\ell} g_t \cdot \prod_{t>\ell} g_t^2 \cdot (g_\ell^2 - g_\ell), \tag{2}$$

where t takes values between 0 and $r-1$ and $\prod_{t<\ell} g_t = 1$ if $\ell = 0$ and $\prod_{t>\ell} g_t^2 = 1$ if $\ell = r-1$. Let $H_\ell = \prod_{t<\ell} g_t \cdot \prod_{t>\ell} g_t^2$. Then note that H_ℓ has size at most $2 \cdot \mathrm{size}(g)$. By induction hypothesis for

[4] Note that we get a bound on the size of the Nullstellensatz refutation, thus a bound on $\mathsf{IPS}_{\mathsf{LIN}}$ refutation.

$g_\ell^2 - g_\ell$, we get that

$$g^2 - g = \sum_{i=1}^{n} \underbrace{\sum_{\ell=0}^{r-1} H_\ell \cdot E_{g_\ell,i}}_{E_{g_i}} \cdot (x_i^2 - x_i) + \sum_{j=1}^{m} \underbrace{\sum_{\ell=0}^{r-1} H_\ell \cdot F_{g_\ell,i}}_{F_{g,j}} \cdot (y_j^2 - y_j)$$

The above expression now allows us to bound the size and depth of $E_{g,i}$ for every $i \in [n]$ and the size of $F_{g,j}$ for every $j \in [m]$ as follows.

Size bound. Before we start the analysis, we recall that we measure the size of a formula by the number of wires in the formula. We also note some bounds on our parameters. We will assume that $s_g > 1$. Let s_g be the short-hand for size(g) and for a fixed g let $s_{g,\ell}$ denote size(g_ℓ) for $0 \leqslant \ell \leqslant r-1$.

$$\sum_\ell s_{g,\ell} \leqslant (s_g - r) \text{ and } \sum_\ell s_{g,\ell}^4 \leqslant (s_g - r)^4 \tag{3}$$

Moreover, we have for any parameter $s > 1$, $(s-1)^4 \leqslant s^4 - s^3/2$.

Now we will bound the size of $E_{g,i}$. In order to so that, we have already seen that size of H_ℓ is at most s_g. We can also bound the size of $E_{g_\ell,i}$ inductively. Thus,

$$\text{size of } E_{g,i} \leqslant \left(\textstyle\sum_\ell 2s_g + 100 s_{g,\ell}^4\right) + 3r \leqslant 3 \cdot s_g \cdot r + 3r + 100 \cdot (s_g - 1)^4.$$

A similar bound can be proved on the size of $F_{g,j}$.

The first bound comes from applying the bounds for H_ℓ, $E_{g_\ell,i}$ and counting the wires feeding into the outer summation. The second bound comes from using Equation (3) and the fact that $s_g - r \leqslant s_g - 1$.

$$\text{size of } E_{g,i} \leqslant 6 \cdot s_g \cdot r + 100 \cdot s_g^4 - 100 s_g^3/2 \leqslant 100 \cdot s^4.$$

Depth bound. $\text{depth}(E_{g,i}) \leqslant 2 \cdot (\text{depth}(E_{g_\ell,i})) + 2 \leqslant 2 \cdot (\text{depth}(g) - 1) + 2 \leqslant 2 \cdot \text{depth}(g)$. The depth of $F_{g,i}$ can be bounded similarly.

<u>case 2:</u> $g = \sum_{\ell=0}^{r-1} g_\ell \cdot A_{g,\ell}$.
In order to prove this case, we will make use of a couple of simple lemmas.

Lemma 2.7. *Let $g = \sum_{\ell=0}^{r-1} g_\ell \cdot A_{g,\ell}$, then for any ℓ, $(A_{g,\ell})^2 - (A_{g,\ell}) = \sum_{j=0}^{t_g} C_{\ell,j} \cdot (y_{g,j}^2 - y_{g,j})$, where size$(C_{\ell,j}) \leqslant 6 \cdot r$.*

Proof. Notice that for any addressing gadget attached to a gate g, it only uses variables from $\mathbf{y}_g$. For concreteness let the addressing gadget be given by $A_{g,\ell}(\mathbf{y}_g) = \prod_{t \in Y} y_{g,t} \times \prod_{t \in Y'} (1 - y_{g,t})$, for some partition of the indices of $\mathbf{y}_g$ into Y and Y'.

Then, we get

$$(A_{g,\ell})^2 - (A_{g,\ell}) = \left(\prod_{t \in Y} y_{g,t} \times \prod_{t \in Y'} (1 - y_{g,t})\right)^2 - \left(\prod_{t \in Y} y_{g,t} \times \prod_{t \in Y'} (1 - y_{g,t})\right)$$

Again using the idea of telescoping summations with respect to the $\mathbf{y}_g$ variables, we can show that

$$(A_{g,\ell})^2 - (A_{g,\ell}) = \sum_{j=0}^{t_g} C_{\ell,j} \cdot (y_{g,j}^2 - y_{g,j})$$

where, $C_{\ell,j}$ consists of monomials in $\mathbf{y}_g$ and it is a $\Pi\Sigma$ circuit in $\mathbf{y}_g$ variables. The input the product gates could be one of the following: either a variable appears as itself, or its square appears, or as $(1-y)$ or as $(1-y)^2$. The size of each linear factor can be bounded by $3 \cdot t_g$ and hence, the overall size can be bounded by $6 \cdot t_g$. This is upper bounded by $6 \cdot r$. ■

Lemma 2.8. *Let $g = \sum_{\ell=0}^{r-1} g_\ell \cdot A_{g,\ell}$, then for any $\ell \neq \ell'$, there exists a $j \in \{0, \ldots, t_g\}$ such that $A_{g,\ell} \times A_{g,\ell'} = C_{\ell,\ell',j} \cdot (y_j^2 - y_j)$, where $\text{size}(C_{\ell,\ell',j}) \leqslant 6 \cdot r$.*

Proof. Here, it is easy to observe that for $\ell \neq \ell'$, there must exist a variable y_j such that either y_j divides $A_{g,\ell}$ and $(1-y_j)$ divides $A_{g,\ell'}$ or vice-versa. Thus, $y_j \cdot (1-y_j)$ divides $A_{g,\ell} \times A_{g,\ell'}$. Thus, we get $A_{g,\ell} \times A_{g,\ell'} = C_{\ell,\ell',j} \cdot (y_j^2 - y_j)$, where $C_{\ell,\ell',j}$ is simply the circuit consisting of a polynomial in $\mathbf{y}_g$ variables. As in Lemma 2.7, here again we get $\text{size}(C_{\ell,\ell',j}) \leqslant 6 \cdot r$. ■

We will now resume the proof of case 2, i.e., the case when g is a sum gate. We will again analyze $g^2 - g$.

$$\begin{aligned}
g^2 - g &= \left(\sum_{\ell=0}^{r-1} g_\ell \cdot A_{g,\ell}\right)^2 - \sum_{\ell=0}^{r-1} g_\ell \cdot A_{g,\ell} \\
&= \sum_{\ell=0}^{r-1} (g_\ell \cdot A_{g,\ell})^2 + \sum_{\ell \neq \ell'} g_\ell \cdot A_{g,\ell} \cdot g_{\ell'} \cdot A_{g,\ell'} - \sum_{\ell=0}^{r-1} g_\ell \cdot A_{g,\ell} \\
&= \sum_{\ell=0}^{r-1} (g_\ell \cdot A_{g,\ell})^2 - \sum_{\ell=0}^{r-1} g_\ell \cdot (A_{g,\ell})^2 + \sum_{\ell=0}^{r-1} g_\ell \cdot (A_{g,\ell})^2 - \sum_{\ell=0}^{r-1} g_\ell \cdot A_{g,\ell} + \sum_{\ell \neq \ell'} g_\ell \cdot g_{\ell'} \cdot A_{g,\ell} \cdot A_{g,\ell'},
\end{aligned}$$

where we added and subtracted the same quantity (second and third summation). After rearranging, we get

$$g^2 - g = \sum_{\ell=0}^{r-1} \left(g_\ell^2 - g_\ell\right) \cdot (A_{g,\ell})^2 + \sum_{\ell=0}^{r-1} g_\ell \cdot \left(A_{g,\ell}^2 - A_{g,\ell}\right) + \sum_{\ell \neq \ell'} g_\ell \cdot g_{\ell'} \cdot A_{g,\ell} \cdot A_{g,\ell'},$$

We now apply induction on the first term and apply Lemma 2.7 and Lemma 2.8 on the second and third terms, respectively.

$$\begin{aligned}
g^2 - g = &\sum_{i=1}^{n} \underbrace{\sum_{\ell=0}^{r-1} (A_{g,\ell})^2 \cdot E_{g_\ell,i}}_{(E_{g,i})} \cdot (x_i^2 - x_i) + \sum_{j=1}^{m} \underbrace{\sum_{\ell=0}^{r-1} (A_{g,\ell})^2 \cdot F_{g_\ell,j}}_{(I)} \cdot (y_j^2 - y_j) \\
&+ \sum_{j=0}^{t_g} \underbrace{\sum_{\ell=0}^{r-1} C_{\ell,j} \cdot g_\ell}_{(II)} \cdot (y_j^2 - y_j) + \underbrace{\sum_{\ell \neq \ell'} g_\ell \cdot g_{\ell'} \cdot C_{\ell,\ell',j}}_{(III)} \cdot (y_j^2 - y_j)
\end{aligned}$$

Using this expression, we can now derive size and depth bounds on the refutation size.

Size bound.

We will now bound each term in the expression above.

Bounding the size of $E_{g,i}$. The size of $A_{g,\ell}$ can be bounded by $2r$. The size of each $E_{g_\ell,i}$ can be bounded inductively. Finally, the numbers of wires feeding into the outer summation can be bounded by $3r$. Thus we have, the following bound

$$\begin{aligned}\text{size}(E_{g,i}) &\leqslant \sum_{\ell=0}^{r-1}(2r + 100s_{g,\ell}^4) + 3r \leqslant 2r^2 + 100\sum_{\ell} s_{g,\ell}^4 + 3r \leqslant 5r^2 + 100\sum_{\ell} s_{g,\ell}^4 \\ &\leqslant 5r^2 + 100(s-r)^4 \qquad \text{Using Equation (3)} \\ &\leqslant 5r^2 + 100s^4 - 100s^3/2 \leqslant 100s^4. \qquad \text{Using the bound on } s-r \text{ and } (s-1)^4\end{aligned}$$

Bounding the size of $F_{g,i}$. The bound on the size of $F_{g,i}$ can be obtained by analyzing terms (I), (II), and (III) above. Note that, the bound on term (I) is identical to the bound on $E_{g,i}$. So, we will have

$$\text{size of (I)} \leqslant 5r^2 + 100\sum_{\ell} s_{g,\ell}^4 \qquad (a).$$

To bound the size of (II), we will use Lemma Lemma 2.7. We get

$$\text{size of (II)} \leqslant \left(\sum_{\ell} 6\cdot r + s_{g,\ell}\right) + 3\cdot r \leqslant 9r^2 + \sum_{\ell} s_{g,\ell} \qquad (b)$$

Finally, we bound the size of (III).

$$\begin{aligned}\text{size of (III)} &\leqslant \left(\sum_{\ell\neq\ell'} 6\cdot r + s_{g,\ell} + s_{g,\ell'}\right) + 4\cdot r^2 \\ &= \sum_{\ell\neq\ell'} 6\cdot r + \sum_{\ell\neq\ell'} s_{g,\ell} + \sum_{\ell\neq\ell'} s_{g,\ell'} + 4\cdot r^2 \\ &\leqslant 6r^3 + 2r\sum_{\ell\neq\ell'} s_{g,\ell} + 4r^2 \qquad (c)\end{aligned}$$

Putting (a), (b), and (c) together and by combining terms we get that

$\text{size}(F_{g,i}) \leqslant 25r^3 + 3rs + 100\sum_{\ell} s_{\ell}^4 \leqslant 25r^3 + 3rs + 100(s-1)^4 \leqslant 25r^3 + 3rs + 100s^4 - 100s^3/2 \leqslant 100s^4$

Depth bound. The depth is bound is similar to the one we had in case 1 above. $\text{depth}(E_{g,i}) \leqslant 2\cdot(\text{depth}(E_{g_\ell,i})) + 2 \leqslant 2\cdot(\text{depth}(g)-1) + 2 \leqslant 2\cdot\text{depth}(g)$. The depth of $F_{g,i}$ can be bounded similarly. ■

2.2.2 Lower bound on the refutation of f_Δ

In this section we focus on the lower bound of the size of the constant-depth IPS refutation of f_Δ. Recall that $Q_\Delta(\mathbf{x})$ denotes the polynomial from [LST21, Theorem 2.1] and let s denote the size of the depth-Δ circuit computing Q_Δ.

Lemma 2.9 (Lower Bound)**.** *Let f_Δ be as defined above. Every constant-depth* IPS *refutation of depth $\Delta'' \leqslant \Delta/2 - 11$ requires size $s'' = s^{\omega(1)}$.*

Proof. Let $C''((\mathbf{x},\mathbf{y}), u, \mathbf{v}, \mathbf{z})$ be an IPS refutation of $f_\Delta(\mathbf{x},\mathbf{y})$. Let s'' and Δ'' denote the circuit size and the depth of C'', respectively. Then we have the following facts:

1. By definition $C''((\mathbf{x},\mathbf{y}), 0, \mathbf{0}, \mathbf{0}) = 0$ and $C''((\mathbf{x},\mathbf{y}), f_\Delta, \mathbf{x}^2 - \mathbf{x}, \mathbf{y}^2 - \mathbf{y}) = 1$ so using [FSTW21, Lemma 6.1] we have that
$$1 - C''((\mathbf{x},\mathbf{y}), 0, \mathbf{x}^2 - \mathbf{x}, \mathbf{y}^2 - \mathbf{y}) = f_\Delta \cdot h,$$
for some polynomial $h(\mathbf{x},\mathbf{y})$.
2. By [BKRRSS25, Theorem 1.1], from $1 - C''((\mathbf{x},\mathbf{y}), 0, \mathbf{x}^2 - \mathbf{x}, \mathbf{y}^2 - \mathbf{y})$ we can extract an algebraic formula for $f_\Delta(\mathbf{x},\mathbf{y})$ whose size is $\mathrm{poly}(s'')$ and whose depth is at most $\Delta'' + 10$.
3. By construction of $f_\Delta(\mathbf{x},\mathbf{y})$ and by Lemma 2.3 there exists some $\mathbf{b} \in \mathbb{F}^{|\mathbf{y}|}$ such that
$$f_\Delta(\mathbf{x},\mathbf{b}) = Q_\Delta(\mathbf{x}),$$
so any formula computing f_Δ at depth $\Delta'' + 10$ also computes Q_Δ at depth $\Delta'' + 10$.
4. [LST21, Theorem 2.1] (also stated in Theorem 1.3) states that every algebraic formula computing Q_Δ at depth $\Delta/2 - 1$ requires size $s^{\omega(1)}$.

Putting all this together, we get that if $\Delta'' + 10 \leqslant \Delta/2 - 1$ then $C''((\mathbf{x},\mathbf{y}), u, \mathbf{v}, \mathbf{z})$ requires size $s^{\omega(1)}$. ■

Remark 2.10. *We use the recent factorization result of [BKRRSS25] to obtain our lower bound. However, we would like to also note that this is not necessary in our case. Due to a results of [CKS19], it is known that small-degree factors of any polynomial computed by constant-depth circuits/formulas of polynomial size can also be computed by constant-depth circuits/formulas of polynomial size. The hard polynomial from [LST21] as well as our addressing gadgets have small degree, our hard instance has small degree (i.e. logarithmic in the number of variables). Thus, all its factors are also of small degree. That is, [CKS19] is applicable in our case.*

We present the proof using [BKRRSS25] as it will help adapt our proof strategy to other scenarios more directly if we obtain strong algebraic complexity lower bounds in the future.

We now use Lemma 2.5 and Lemma 2.9 to finish the proof of Theorem 1.5. Note that the depth of the IPS refutation for f_Δ is $4\Delta + 6$ and size is $\mathrm{poly}(s)$, whereas any circuit of depth less than $\Delta/2 - 11$ requires superpolynomial size. Thus, for $\Gamma = 4\Delta + 6$, we get that f_Γ has an IPS refutation of depth Γ and any IPS refutation of depth less than $\Gamma/10$ requires superpolynomial size. This completes the proof of Theorem 1.5. ■

3 Multilinear separation theorem

We start by proving a lemma that will be useful in the rest of the section.

Lemma 3.1. *Let $N \in \mathbb{N}$ and let $\mathbf{x} = \{x_1, \ldots, x_N\}$. Let $f(\mathbf{x})$ be a multilinear polynomial such that $f(\mathbf{a}) \in \{0,1\}$ for any $a \in \{0,1\}^N$, then*

1. *The following identity holds.*

$$(2 - f(\mathbf{x})) \times \frac{1 + f(\mathbf{x})}{2} \equiv 1 \mod \mathbf{x}^2 - \mathbf{x}$$

2. $2 - f(\mathbf{x}) = 0$ *is unsatisfiable over the Boolean cube.*

3. *The unique multilinear function* $g(\mathbf{x})$ *obeying* $g(\mathbf{x})(2 - f(\mathbf{x})) \equiv 1 \mod \mathbf{x}^2 - \mathbf{x}$ *is* $(1 + f(\mathrm{x}))/2$. *Thus, it has the same multilinear circuit size and depth as* $2 - f(x)$.

Proof. The first part of the lemma is a simple check. If $f(\mathbf{a}) = 1$ the left hand side evaluates to 1. Similarly, when $f(\mathbf{a}) = 0$ it again evaluates to 1. The second part follows because we have assumed that $f(\mathbf{x})$ only takes Boolean values. Finally, the third part follows immediately from the first part. ■

3.1 Multilinear-NC^1 vs. multilinear-NC^2-IPS

In this section we prove Theorem 1.7.

We will construct a polynomial such that it is computable by a multilinear NC^2 circuit and such that it only takes Boolean values over the Boolean hypercube. This along with Lemma 3.1 will give us the desired separation.

Notation. Let $[n] = \{1, \ldots, n\}$ and let $\mathbf{u} = \{u_1, \ldots, u_{2n}\}$ and $\mathbf{v} = \{v_{i,j,k}\}_{i,j,k \in [2n]}$. For $i, j \in [n]$ let $[i, j]$ denote the interval $\{k \mid i \leqslant k \text{ and } k \leqslant j\}$. Let $\ell([i, j])$ denote the length of the interval, i.e., $j - i + 1$. When $j < i$, then $[i, j] = \varnothing$.

We first recall the hard polynomial defined by [RY08], which is a simplification of the polynomial defined by [Raz04], which showed the first separation between multilinear formulas and circuits.

The polynomial is defined inductively as follows. For $i \in [n]$, let $f_{i,i}(\mathbf{u}, \mathbf{v}) = 1$. If $\ell([i, j])$ is an even number more than 0, then

$$f_{i,j}(\mathbf{u}, \mathbf{v}) = (1 + u_i u_j) \cdot f_{i+1,j-1}(\mathbf{u}, \mathbf{v}) \; + \sum_{r \in [i+1, j-1]} v_{i,r,j} \cdot f_{i,r}(\mathbf{u}, \mathbf{v}) \cdot f_{r+1,j}(\mathbf{u}, \mathbf{v})$$

Finally, the hard polynomial is $F(\mathbf{u}, \mathbf{v}) = f_{1,2n}(\mathbf{u}, \mathbf{v})$. They prove the following theorem about the polynomial.

Lemma 3.2 ([Raz04], [RY08])**.** *Let* $n \in \mathbb{N}$ *and let* $\mathbf{u} = \{u_1, \ldots, u_{2n}\}$ *and* $\mathbf{v} = \{v_{i,j,k}\}_{i,j,k \in [2n]}$ *be two sets of variables. Let* $F(\mathbf{u}, \mathbf{v})$ *be the polynomial defined above. Then the following holds.*

1. $F(\mathbf{u}, \mathbf{v})$ *can be computed by a multilinear circuit of size* $\mathrm{poly}(n)$ *and depth* $O(\log^2 n)$.

2. *Any multilinear formula computing* $F(\mathbf{u}, \mathbf{v})$ *must have size* $n^{\Omega(\log n)}$.

Remark 3.3. *Note that* $F(\mathbf{u}, \mathbf{v})$ *when evaluated over the Boolean hypercube can take large values. (In fact,* $F(\mathbf{1}, \mathbf{1})$ *grows exponentially with* n.*) For Lemma 3.1 to be applicable, we need a polynomial that takes only Boolean values over the hypercube.*

We construct such a polynomial by modifying $F(\mathbf{u}, \mathbf{v})$. We design the polynomial using new set of gadget variables, which we call $\mathbf{w}$. These will serve a dual purpose, first, they will help us create addressing gadgets for the + gates and they will assume the role of $\mathbf{v}$ variables in the definition of $F(\mathbf{u}, \mathbf{v})$.

Let $\mathbf{u} = \{u_1, \ldots, u_{2n}\}$. For an interval $[i, j]$, let $W^{[i,j]}$ denote the following set of variables.

$$W^{[i,j]} = \{w_{\text{top}}^{[i,j]}\} \cup \{w_{\text{leaf}}^{[i,j]}\} \cup \tilde{W}^{[i,j]},$$

where $\tilde{W}^{[i,j]}$ consists of variables we will use for the addressing gadgets. Recall that for $n \in \mathbb{N}$, t_n denotes the smallest integer such that $2^{t_n} > n$. Let $t_{i,j}$ be the shorthand for $t_{\ell([i+1,j-1])}$ and $n_{i,j}$ be a shorthand for $\ell([i+1, j-1])$. Let $\tilde{W}^{[i,j]} = \{w_r^{[i,j]} \mid 0 \leqslant r \leqslant t_{i,j}\}$.

Finally, we define $\mathbf{w} = \bigcup_{[i,j]} W^{[i,j]}$, where the union is over all intervals $[i, j]$, where $1 \leqslant i < j \leqslant n$ and $\ell([i, j])$ is even[5]. Here, the size of any set $W^{[i,j]}$ is $O(\log n)$ and hence we have $O(n^2 \log n)$-many w variables. We will use $m(n)$ to denote the cardinality of $\mathbf{w}$ and m, when n is clear from the context.

Definition 3.4. *The hard polynomial is defined inductively as follows: If $\ell([i, j]) = 0$ then $p_{[i,j]}(u, w) = 1$. If $\ell([i, j]) > 0$ and even, then*

$$\begin{aligned} p_{i,j}(\mathbf{u}, \mathbf{w}) = \quad & (1 - w_{top}^{[i,j]}) \left((1 - w_{leaf}^{[i,j]}) + w_{leaf}^{[i,j]} \cdot u_i \cdot u_j \right) \times p_{i+1,j-1} \\ + \quad & w_{top}^{[i,j]} \times \left(\textstyle\sum_{r \in [i+1, j-1]} g_r(\tilde{W}^{[i,j]}) \times p_{i,r} \cdot p_{r+1,j} \right), \end{aligned}$$

g_r is the addressing gadget, i.e.

$$g_r(\tilde{W}^{[i,j]}) = \prod_{t \in B_{n_{i,j},0}(r)} (1 - w_t^{[i,j]}) \cdot \prod_{t \in B_{n_{i,j},1}(r)} w_t^{[i,j]}$$

Here, the sets $B_{n_{i,j},0}$ and $B_{n_{i,j},1}$ are defined as in Definition 2.1.

Finally, $P(\mathbf{u}, \mathbf{w}) = p_{1,2n}(\mathbf{u}, \mathbf{w})$.

We will now prove that the polynomial $P(\mathbf{u}, \mathbf{w})$ defined above retains the properties of the polynomial $F(\mathbf{u}, \mathbf{v})$, that is, it is computed by multilinear circuits and it is hard for multilinear formulas. Additionally, we will show that the polynomial only takes Boolean values over the Boolean hypercube. Formally, we prove the following theorem.

Theorem 3.5. *Let $n \in \mathbb{N}$ and let $\mathbf{u}$ and $\mathbf{w}$ be as defined above. Also, let $P(\mathbf{u}, \mathbf{w})$ be the polynomial from Definition 3.4. Then, the following statements hold.*

1. *$P(\mathbf{u}, \mathbf{w}) \in \{0, 1\}$ when evaluated over the Boolean hypercube.*
2. *$P(\mathbf{u}, \mathbf{w})$ can be computed by a multilinear circuit of size $\text{poly}(n)$ and depth $O(\log^2 n)$.*
3. *Any multilinear formula computing $P(\mathbf{u}, \mathbf{w})$ must have size $n^{\Omega(\log n)}$.*

Before we present the proof for the theorem, we will use it to prove our main theorem Theorem 1.7, which we recall below.

[5] We only use intervals of even length inductively.

Theorem 1.7 (multilinear NC^1 vs multilinear NC^2-IPS)**.** *Fix a field, $\mathbb{F}$ of characteristic 0. For every growing parameter $N \in \mathbb{N}$, there is a multilinear polynomial $Q \in \mathbb{F}[x_1, \ldots, x_N]$ which is unsatisfiable over $\{0,1\}^N$ such that*

1. *There is a multilinear functional refutation for $Q(\mathbf{x})$, say $G(\mathbf{x})$, computable by a syntactic multilinear circuit of polynomial size and $O(\log^2 N)$ depth.*
2. *Any multilinear-NC^1-$\mathsf{IPS}_{\mathsf{LIN}'}$ for it requires size $N^{\Omega(\log N)}$.*

Proof of Theorem 1.7. Let $N = n + m$, where n is the cardinality of $\mathbf{u}$ and m is the cardinality of $\mathbf{w}$ and let $\mathbf{x} = \mathbf{u} \cup \mathbf{w}$. We will define $Q(\mathbf{x}) = 2 - P(\mathbf{x})$, where P is as in Theorem 3.5. Clearly $Q(\mathbf{x})$ is unsatisfiable over the Boolean hypercube. Using Lemma 3.1, we know that the refutation for $Q(\mathbf{x})$ is $((P(\mathbf{x})) + 1)/2$ modulo the Boolean axioms. Thus, the functional refutation of $Q(\mathbf{x})$ is computable by multilinear circuit of size $\text{poly}(n)$ and depth $O(\log^2 n)$.

Moreover, from Theorem 3.5 we know that $P(\mathbf{x})$ requires multilinear NC^1 circuit of size $n^{\Omega(\log n)}$. As the refutation is $(P(\mathbf{x}) + 1)/2$, we also get that the multilinear NC^1-$\mathsf{IPS}_{\mathsf{LIN}'}$ refutation[6] for it must have size $n^{\Omega(\log n)}$. As N and n are polynomially related, this also gives a $N^{\Omega(\log N)}$ lower bound on the proof size of multilinear NC^1-$\mathsf{IPS}_{\mathsf{LIN}}$ refutations. ■

3.2 Proof of Theorem 3.5

Part 1 of Theorem 3.5. We prove this statement by using the inductive structure of $P(\mathbf{u}, \mathbf{w})$. Specifically, we will show that for any interval $[i, j]$, the polynomial corresponding to it, i.e., $p_{i,j}(\mathbf{u}, \mathbf{w}) \in \{0, 1\}$ when evaluated over the Boolean hypercube. We induct on the length of the interval. We only need to consider even length intervals.

Base case. Suppose $\ell([i, j]) = 0$ then the statement trivially holds.

Inductive step. Suppose $\ell([i, j]) > 0$. The polynomial $p_{i,j}$ is as defined in Definition 3.4.

Suppose $w_{\text{top}}^{[i,j]} = 0$, then

$$p_{i,j} = \left((1 - w_{\text{leaf}}^{[i,j]}) + w_{\text{leaf}}^{[i,j]} \cdot u_i \cdot u_j\right) \cdot p_{i+1,j-1}.$$

Notice that if $w_{\text{leaf}}^{[i,j]} = 0$ then $p_{i,j} = p_{i+1,j-1}$, which by induction hypothesis is Boolean. If $w_{\text{leaf}}^{[i,j]} = 1$ then $p_{i,j} = u_i u_j p_{i+1,j-1}$, which is either 0 or 1 for Boolean values of u_i, u_j and $p_{i+1,j-1}$.

On the other hand, if $w_{\text{top}}^{[i,j]} = 1$, then

$$p_{i,j} = \left(\sum_{r \in [i+1,j-1]} g_r(\tilde{W}^{[i,j]}) \times p_{i,r} \cdot p_{r+1,j}\right).$$

Now, suppose the variables in the set $\tilde{W}^{[i,j]}$ are set to 0s and 1s such that the boolean assignment equals $r + 2^{t_{i,j}}$, then we get $p_{i,j} = p_{i,r} \cdot p_{r+1,j}$. By inductive assumption $p_{i,r} \in \{0, 1\}$ and $p_{r+1,j} \in \{0, 1\}$. That finishes the proof.

[6] As we use functional method, we get a lower bound in $\mathsf{IPS}_{\mathsf{LIN}'}$ and not just for $\mathsf{IPS}_{\mathsf{LIN}}$.

Part 2 of Theorem 3.5. Notice that polynomial $F(\mathbf{u}, \mathbf{v})$ defined by [RY08] is very similar to $P(\mathbf{u}, \mathbf{w})$. Instead of $\mathbf{v}$ variables, we have small local changes using the addressing gadgets. The addressing gadgets themselves are constant-depth (unbounded fan-in) multilinear circuits. It is easy to see that by implementing the inductive definition of $p(\mathbf{u}, \mathbf{v})$, we will obtain a polynomial size and polynomial depth multilinear circuit. By using a depth reduction result of [RY08], we can obtain a polynomial size and $O(\log^2 n)$-depth multilinear circuit for the polynomial.

Part 3 of Theorem 3.5. Firstly, we will prove that for every partition of variables in $\mathbf{u}$ into two sets of equal size say, $\mathbf{u} = \mathbf{y} \cup \mathbf{z}$, the rank of the matrix $M_{\mathbf{y},\mathbf{z}}(p_{1,2n})$ is equal to 2^n. The bound will then imply a lower bound for the IPS proof size. This step is quite standard, but we will present it for completeness.

Lemma 3.6. *Let $n \in \mathbb{N}$ and $p_{1,2n}$ be as defined above. Then, for any partition of $\mathbf{u}$ into $\mathbf{y} \cup \mathbf{z}$ each of cardinality n, there exists an assignment to variables in $\mathbf{w}$ to field constants, such that $rank(M_{\mathbf{y},\mathbf{z}}(p_{1,2n})) = 2^n$.*

Proof. We will prove this by induction on n.

Base case. Suppose $n = 1$, then $p_{1,2} = \left((1 - w_{\text{leaf}}^{[1,2]}) + w_{\text{leaf}}^{[1,2]} \cdot u_1 \cdot u_2\right)$. By setting $w_{\text{leaf}}^{[1,2]} = 1/2$, we get $p_{1,2} = \frac{1}{2} \cdot (1 + u_1 \cdot u_2)$ and the statement trivially holds.

Inductive step. Let $n > 1$. We consider two cases. Either u_1 and u_{2n} are in the same part under the partition of $\mathbf{u}$ into $\mathbf{y} \cup \mathbf{z}$ or they are in different parts.

u_1 and u_{2n} in different parts We will set $w_{\text{top}}^{[1,2n]} = 0$ and $w_{\text{leaf}}^{[1,2n]} = 1/2$. Under this substitution, $p_{1,2n} = \frac{1}{2}(1 + u_1 \cdot u_{2n}) \cdot p_{2,2n-1}$. By induction hypothesis, $p_{2,2n-1}$ is full rank, i.e. 2^{n-1}, under every equi-sized partition of its variables. And the rank for $(1 + u_1 u_{2n})$ is 2. Note also that $p_{2,2n-1}$ does not use the variables u_1 and u_{2n}. Hence, we are done in this case.

u_1 and u_{2n} in same part In this case, there is an $r \in [i + 1, j - 1]$ such that the intervals $[1, r]$ and $[r + 1, 2n]$ evenly split $\mathbf{y}$ and $\mathbf{z}$ variables. We set $w_{\text{top}}^{[1,2n]} = 1$ and the variables in the addressing gadget to the binary encoding of $r + 2^{t_{i,j}}$. This gives $p_{1,2n} = p_{1,r} \cdot p_{r+1,2n}$. Using induction on $p_{1,r}, p_{r+1,2n}$ and observing that the two polynomials do not share any variables we get the desired bound on the rank of $p_{1,2n}$. ■

References

[And25] Robert Andrews. "Algebraic Pseudorandomness in VNC^0". In: *40th Computational Complexity Conference (CCC 2025)*. Ed. by Srikanth Srinivasan. Vol. 339. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025, 15:1–15:15. ISBN: 978-3-95977-379-9. DOI: 10.4230/LIPIcs.CCC.2025.15. URL: https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.CCC.2025.15 (cit. on p. 4).

[AF22] Robert Andrews and Michael A. Forbes. "Ideals, determinants, and straightening: proving and using lower bounds for polynomial ideals". In: *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*. STOC 2022. Rome, Italy: Association for Computing Machinery, 2022, pp. 389–402. ISBN: 9781450392648. DOI: 10.1145/3519935.3520025. URL: https://doi.org/10.1145/3519935.3520025 (cit. on pp. 1, 4).

[BLRS25] Amik Raj Behera, Nutan Limaye, Varun Ramanathan, and Srikanth Srinivasan. "New Bounds for the Ideal Proof System in Positive Characteristic". In: *52nd International Colloquium on Automata, Languages, and Programming (ICALP 2025)*. Ed. by Keren Censor-Hillel, Fabrizio Grandoni, Joël Ouaknine, and Gabriele Puppis. Vol. 334. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025, 22:1–22:20. ISBN: 978-3-95977-372-0. DOI: 10.4230/LIPIcs.ICALP.2025.22. URL: https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ICALP.2025.22 (cit. on pp. 1, 2).

[BKRRSS25] Somnath Bhattacharjee, Mrinal Kumar, Shanthanu S. Rai, Varun Ramanathan, Ramprasad Saptharishi, and Shubhangi Saraf. "Closure under factorization from a result of Furstenberg". In: *CoRR* abs/2506.23214 (2025). DOI: 10.48550/ARXIV.2506.23214. arXiv: 2506.23214 (cit. on pp. 4, 6, 15).

[Bür98] Peter Bürgisser. "Completeness and Reduction in Algebraic Complexity Theory". In: 7 (Aug. 1998). DOI: 10.1007/978-3-662-04179-6_1 (cit. on p. 23).

[CGMS25] Prerona Chatterjee, Utsab Ghosal, Partha Mukhopadhyay, and Amit Sinhababu. *IPS Lower Bounds for Formulas and Sum of ROABPs*. July 2025. DOI: 10.48550/arXiv.2507.09515. arXiv: 2507.09515 [cs.CC]. URL: https://arxiv.org/abs/2507.09515 (cit. on p. 1).

[CELS18] Suryajith Chillara, Christian Engels, Nutan Limaye, and Srikanth Srinivasan. "A Near-Optimal Depth-Hierarchy Theorem for Small-Depth Multilinear Circuits". In: *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*. Ed. by Mikkel Thorup. IEEE Computer Society, 2018, pp. 934–945. DOI: 10.1109/FOCS.2018.00092. URL: https://doi.org/10.1109/FOCS.2018.00092 (cit. on p. 7).

[CKS19] Chi-Ning Chou, Mrinal Kumar, and Noam Solomon. "Closure Results for Polynomial Factorization". In: *Theory of Computing* 15 (2019), pp. 1–34. DOI: 10.4086/TOC.2019.V015A013. URL: https://doi.org/10.4086/toc.2019.v015a013 (cit. on p. 15).

[CR79] Stephen A. Cook and Robert A. Reckhow. "The relative efficiency of propositional proof systems". In: *Journal of Symbolic Logic* 44.1 (1979), pp. 36–50. DOI: 10.2307/2273702 (cit. on p. 1).

[EGLT25] Tal Elbaz, Nashlen Govindasamy, Jiaqi Lu, and Iddo Tzameret. *Lower Bounds against the Ideal Proof System in Finite Fields*. June 2025. DOI: 10.48550/arXiv.2506.17210. arXiv: 2506.17210 [cs.CC]. URL: https://arxiv.org/abs/2506.17210 (cit. on pp. 1, 2).

[FSTW21] Michael A. Forbes, Amir Shpilka, Iddo Tzameret, and Avi Wigderson. "Proof Complexity Lower Bounds from Algebraic Circuit Complexity". In: *Theory Comput.* 17 (2021), pp. 1–88. URL: https://theoryofcomputing.org/articles/v017a010/ (cit. on pp. 1, 2, 4–7, 15, 23).

[GHT22] Nashlen Govindasamy, Tuomas Hakoniemi, and Iddo Tzameret. "Simple Hard Instances for Low-Depth Algebraic Proofs". In: *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*. 2022, pp. 188–199. DOI: 10.1109/FOCS54457.2022.00025 (cit. on pp. 1, 2, 5).

[GP18] Joshua A. Grochow and Toniann Pitassi. "Circuit Complexity, Proof Complexity, and Polynomial Identity Testing: The Ideal Proof System". In: *J. ACM* 65.6 (Nov. 2018). ISSN: 0004-5411. DOI: 10.1145/3230742. URL: https://doi.org/10.1145/3230742 (cit. on pp. 1, 2).

[HLT24a] Tuomas Hakoniemi, Nutan Limaye, and Iddo Tzameret. "Functional Lower Bounds in Algebraic Proofs: Symmetry, Lifting, and Barriers". In: *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*. STOC 2024. Vancouver, BC, Canada: Association for Computing Machinery, 2024, pp. 1396–1404. ISBN: 9798400703836. DOI: 10.1145/3618260.3649616. URL: https://doi.org/10.1145/3618260.3649616 (cit. on p. 1).

[HLT24b] Tuomas Hakoniemi, Nutan Limaye, and Iddo Tzameret. "Functional Lower Bounds in Algebraic Proofs: Symmetry, Lifting, and Barriers". In: *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*. STOC 2024. Vancouver, BC, Canada: Association for Computing Machinery, 2024, pp. 1396–1404. ISBN: 9798400703836. DOI: 10.1145/3618260.3649616. URL: https://doi.org/10.1145/3618260.3649616 (cit. on pp. 2, 5).

[HS65] Juris Hartmanis and Richard E. Stearns. "On the Computational Complexity of Algorithms". In: *Transactions of the American Mathematical Society* 117 (May 1965), pp. 285–306. ISSN: 0002-9947. DOI: 10.2307/1994208 (cit. on p. 1).

[Has86] J Hastad. "Almost optimal lower bounds for small depth circuits". In: *Proceedings of the Eighteenth Annual ACM Symposium on Theory of Computing*. STOC '86. Berkeley, California, USA: Association for Computing Machinery, 1986, pp. 6–20. ISBN: 0897911938. DOI: 10.1145/12130.12132. URL: https://doi.org/10.1145/12130.12132 (cit. on p. 1).

[Juk12] Stasys Jukna. *Boolean Function Complexity: Advances and Frontiers*. Texts in Theoretical Computer Science. Chs. 1–2 survey Shannon's lower bound and Lupanov's matching upper bound yielding a nonuniform circuit size hierarchy. Springer, 2012. ISBN: 978-3-642-24507-7 (cit. on p. 1).

[LST21] Nutan Limaye, Srikanth Srinivasan, and Sébastien Tavenas. "Superpolynomial Lower Bounds Against Low-Depth Algebraic Circuits". In: *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*. 2021, pp. 804–814. DOI: 10.1109/FOCS52979.2021.00083 (cit. on pp. 3–5, 10, 14, 15).

[LST25] Jiaqi Lu, Rahul Santhanam, and Iddo Tzameret. *AC^0[p]-Frege Cannot Efficiently Prove that Constant-Depth Algebraic Circuit Lower Bounds are Hard*. Tech. rep. TR25-134. Electronic Colloquium on Computational Complexity (ECCC), Sept. 2025. URL: https://eccc.weizmann.ac.il/report/2025/134/ (cit. on p. 1).

[Raz04] Ran Raz. "Multilinear- $\mathrm{NC}_1 \neq$ Multilinear- NC_2". In: *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*. FOCS '04. USA: IEEE Computer Society, 2004, pp. 344–351. ISBN: 0769522289. DOI: 10.1109/FOCS.2004.42. URL: https://doi.org/10.1109/FOCS.2004.42 (cit. on pp. 3, 7, 16).

[RY08] Ran Raz and Amir Yehudayoff. "Balancing Syntactically Multilinear Arithmetic Circuits". In: *computational complexity* 17.4 (Dec. 2008), pp. 515–535. ISSN: 1420-8954. DOI: 10.1007/s00037-008-0254-0. URL: https://doi.org/10.1007/s00037-008-0254-0 (cit. on pp. 3, 7, 16, 19).

[RY09] Ran Raz and Amir Yehudayoff. "Lower Bounds and separations for constant depth multilinear circuits". English. In: *Computational Complexity* 18.2 (June 2009), pp. 171–207. ISSN: 1016-3328. DOI: 10.1007/s00037-009-0270-8 (cit. on p. 7).

[ST21] Rahul Santhanam and Iddo Tzameret. "Iterated Lower Bound Formulas: A Diagonalization-Based Approach to Proof Complexity". In: *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing (STOC 2021)*. ECCC TR21-138; journal version in *SIAM J. Comput.*, 2025. ACM, 2021, pp. 234–247. DOI: 10.1145/3406325.3451010 (cit. on p. 1).

[Sha49] Claude. E. Shannon. "The synthesis of two-terminal switching circuits". In: *The Bell System Technical Journal* 28.1 (1949), pp. 59–98. DOI: 10.1002/j.1538-7305.1949.tb03624.x (cit. on p. 1).

[Sip83] Michael Sipser. "Borel sets and circuit complexity". In: *Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing*. STOC '83. New York, NY, USA: Association for Computing Machinery, 1983, pp. 61–69. ISBN: 0897910990. DOI: 10.1145/800061.808733. URL: https://doi.org/10.1145/800061.808733 (cit. on p. 1).

[SHI65] Richard E. Stearns, Juris Hartmanis, and Philip M. Lewis II. "Hierarchies of Memory Limited Computations". In: *Proceedings of the 6th Annual Symposium on Switching Circuit Theory and Logical Design (FOCS)*. IEEE, 1965, pp. 179–190. DOI: 10.1109/FOCS.1965.11 (cit. on p. 1).

A The complexity of refuting lifted subset-sum

Let $F(\mathbf{x},\mathbf{z}) = \sum_{i<j\in[n]} z_{i,j}x_ix_j - \beta$ be the lifted subset-sum instance, where $\beta \in \Theta(n^3)$. Clearly, it is an unsatisfiable instance. It was used in [FSTW21] to prove a lower bound on the the size of the multilinear formula $\mathsf{IPS}_{\mathsf{LIN}'}$. Here, we further analyze the hardness of refuting this instance. We show that its refutation must have high complexity under a standard complexity assumption. Specifically, we prove the following.

Lemma A.1. *If $F(\mathbf{x},\mathbf{z})$ has a polynomial size multilinear $\mathsf{IPS}_{\mathsf{LIN}'}$ refutation, then $\mathsf{VP} = \mathsf{VNP}$.*

This makes our hard instance in Theorem 1.7 interesting. On the one hand we obtain an equally strong lower bound as in [FSTW21], and on the other hand we also obtain a reasonably good upper bound on the functional refutation of our instance.

In order to prove the lemma, we start with some notation and preliminaries. Let $V \subseteq [n]$ and let $K_V = \{(i,j) \mid i,j \in V, i<j\}$. Let $e=(i,j)$ denote a pair from the set K_V, then we use z_e to denote $z_{i,j}$.

Lemma A.2 ([Bür98])**.** *Let $C_\ell(\mathbf{x},\mathbf{z})$ be the Clique polynomial defined as follows.*

$$C_\ell(\mathbf{x},\mathbf{z}) = \sum_{V\subseteq[n],|V|=\ell} \prod_{e\in K_V} z_e \prod_{i\in V} x_i$$

$C_{n/2}(\mathbf{x},\mathbf{z})$ is VNP complete[7]*.*

We are now ready to prove Lemma A.1.

Proof of Lemma A.1. We introduce some notation. We use $\binom{[n]}{2}$ to denote the set $\{(i,j) \mid i,j \in [n], i<j\}$. Let the subset-sum instance (without the lift) be

$$f(\mathbf{z}) = \sum_{(i,j)\in\binom{[n]}{2}} z_{i,j} - \beta = 0$$

for $\beta > n^2$. (This instance is the same as the subset-sum instance in [FSTW21, Section 5] up to relabeling.) In [FSTW21, Proposition B.1], they gave an explicit description of its multilinear functional refutation, i.e. they exactly computed the multilinear polynomial $g(\mathbf{z})$ such that

$$g(\mathbf{z}) = \frac{1}{\left(\sum_{(i,j)\in\binom{[n]}{2}} z_{i,j} - \beta\right)}, \qquad \text{for every } \mathbf{z}\in\{0,1\}^n.$$

[7] The Clique polynomial from [Bür98] slightly differs from the polynomial we have here. Namely, it is $C_\ell(\mathbf{z}) = \sum_{V\subseteq[n],|V|=\ell}\prod_{e\in K_V} z_e$. However, by substituting $x_is = 1$ in the above polynomial, we can obtain this polynomial.

They showed that every functional refutation of $f(\mathbf{z})$ can be expressed as a linear combination of the elementary symmetric polynomials of degree k, for every $k \in [n]$. More precisely, they showed that

$$g(\mathbf{z}) = \sum_{k=0}^{n} \alpha_k \cdot \sum_{\substack{S \subseteq \binom{[n]}{2} \\ |S|=k}} \prod_{(i,j)\in S} z_{i,j}, \tag{4}$$

where α_k is a non-zero constant that only depends on k and β.

Now, we will first change the input instance to $F(\mathbf{x}, \mathbf{z}) = \sum_{(i,j)} z_{i,j} x_i x_j$, where the sum is over the set $\binom{[n]}{2}$. As $F(\mathbf{x}, \mathbf{z})$ can be obtained from $f(\mathbf{z})$ by a monomial substitution $z_{i,j} \mapsto z_{i,j} x_i x_j$, it is easy to see that the functional refutation of $F(\mathbf{x}, \mathbf{z})$ can be obtained from the refutation of $f(\mathbf{z})$ by monomial substitution. This is because we only need to preserve the refutation over the Boolean cube. Such a monomial substitution can result in a non-multilinear polynomial. Let $\mathsf{ml}[\cdot]$ denote the following map defined for monomials over a set of variables, say $\mathbf{y}$: $\mathsf{ml}\left[\prod_i y_i^{a_i}\right] = \prod_i y_i^{\min\{a_i,1\}}$. The map extends linearly and can be defined as a map from $\mathbb{F}[\mathbf{y}] \to \mathbb{F}[\mathbf{y}]$ for any polynomial ring $\mathbb{F}[\mathbf{y}]$.

Let $G(\mathbf{x}, \mathbf{z})$ denote the unique multilinear refutation of $F(\mathbf{x}, \mathbf{z})$. Then, using Equation (4) we get

$$G(\mathbf{x}, \mathbf{z}) = \sum_{k=0}^{n} \alpha_k \cdot \sum_{S \subseteq \binom{[n]}{2}, |S|=k} \mathsf{ml}\left[\prod_{(i,j)\in S} z_{i,j} x_i x_j\right]$$

Suppose we assume that $F(\mathbf{x}, \mathbf{z})$ has a polynomial-size $\mathsf{IPS}_{\mathsf{LIN}'}$ refutation. This implies that there is a multilinear circuit of polynomial size computing $G(\mathbf{x}, \mathbf{z})$. We further isolate the degree $\binom{n/2}{2}$ component in $\mathbf{z}$ variables by interpolating it out and further degree n component in $\mathbf{x}$ variables by another interpolation. It is easy to see that the polynomial this computes equals $C_{n/2}(\mathbf{x}, \mathbf{z})$ (up to scaling by a coefficient). Assuming $\mathsf{VP} \neq \mathsf{VNP}$, this gives a contradiction. ■

ECCC ISSN 1433-8092
https://eccc.weizmann.ac.il