

Separation Results for Constant-Depth and Multilinear Ideal Proof Systems

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

January 15, 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.

- 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*.
- 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 (IPS_{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

31
32
33
34
35
36
37

38
39

40

41
42
43
44
45
46

47
48
49
50
51

52
53
54

55

56

functional method.

Given a polynomial $p(\mathbf{x})$, let $\text{Image}(p(\mathbf{x}))$ 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	1
1.1	Ideal Proof System	1
1.2	Algebraic Circuit Complexity	2
1.2.1	Separation results	3
1.3	Results and Techniques: the constant-depth IPS hierarchy theorem	4
1.4	Results and Techniques: multilinear-NC ¹ vs. multilinear-NC ²	6
2	Constant-depth Hierarchy	7
2.1	Adding addressing gadgets at sum gates	8
2.2	Proof	10
2.2.1	Upper bound on the refutation of f_Δ	10
2.2.2	Lower bound on the refutation of f_Δ	14
3	Multilinear separation theorem	15
3.1	Multilinear-NC ¹ vs. multilinear-NC ² -IPS	16
3.2	Proof of Theorem 3.5	18
	References	19
A	The complexity of refuting lifted subset-sum	23

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 $\{x_1, x_2, \dots, x_N\}$. Given a collection of polynomial axioms $f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x}) \in \mathbb{F}[\mathbf{x}]$, the goal is to certify that there is *no* Boolean assignment to the variables that simultaneously satisfies all the equalities $f_1(\mathbf{x}) = f_2(\mathbf{x}) = \dots = f_m(\mathbf{x}) = 0$. 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-

¹In 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.

95 algebraically. Specifically, if the system has no common zero over \mathbb{F} , then there exist polynomials
 96 $A_1(\mathbf{x}), \dots, A_m(\mathbf{x})$ and $B_1(\mathbf{x}), \dots, B_N(\mathbf{x})$ such that

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

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

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

103
 104 **Definition 1.1** (Ideal Proof System [GP18]). *Let $f_1, \dots, f_m \in \mathbb{F}[x_1, \dots, x_n]$ be a system of unsat-*
 105 *isfiable polynomials over the Boolean cube $\{0, 1\}^n$. In other words, there is no Boolean assignment*
 106 *$\mathbf{a} \in \{0, 1\}^n$ to the variables x_1, \dots, x_n so that $f_i(\mathbf{a}) = 0$ for all $i \in [m]$.*

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

- 109 • $C(\mathbf{x}, \mathbf{0}, \mathbf{0}) = 0$, and
- 110 • $C(\mathbf{x}, f_1, \dots, f_m, x_1^2 - x_1, \dots, x_n^2 - x_n) = 1$.

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

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

115

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

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

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

127 1.2 Algebraic Circuit Complexity

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

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

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

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

142 1.2.1 Separation results

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

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

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

152
 153 **Definition 1.2** (Rank measure [Raz04]). *Let $\mathbf{x} = \{x_1, \dots, x_{2n}\}$. Let $\mathbf{y} \cup \mathbf{z}$ be an equipartition
 154 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
 155 multilinear monomials in \mathbf{y} variables and columns labeled by multilinear monomials in \mathbf{z} variables.
 156 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
 157 is the coefficient of the monomial $m_{\mathbf{y}} \cdot m_{\mathbf{z}}$ in f . The measure is the rank of this matrix.*

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

160
 161 It was shown by Raz [Raz04] that a multilinear formula computing any full-rank polynomial $f(\mathbf{x})$
 162 requires size $n^{\Omega(\log n)}$. In our work, we build on the full-rank polynomial defined in a subsequent
 163 work of Raz and Yehudayoff [RY08].

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

170 Formally, it states the following.

171

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

175 1. There exists a constant-free² algebraic formula with input gates carrying labels from $\mathbf{x} \cup \{0, 1\}$
176 which computes $Q_\Delta(\mathbf{x})$ in depth Δ and size s .

177 2. Any algebraic formula computing $Q_\Delta(\mathbf{x})$ in depth $(\Delta/2 - 1)$ requires size $s^{\omega(1)}$.

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

182 We are now ready to state our contributions.

183 1.3 Results and Techniques: the constant-depth IPS hierarchy theorem

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

186

187 **Theorem 1.5** (Constant-depth IPS hierarchy). Let \mathbb{F} be a field of characteristic zero. The following
188 holds for every growing parameter $N \in \mathbb{N}$ and a depth parameter $\Gamma \in \mathbb{N}$ where $\Gamma = \mathcal{O}(1)$. For every
189 depth parameter Γ , there exists a multilinear polynomial $f_\Gamma \in \mathbb{F}[x_1, \dots, x_N]$ which is unsatisfiable
190 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
191 conditions hold:

192 1. There exists an IPS refutation for $f_\Gamma(\mathbf{x})$ in depth Γ and size $\mathcal{O}(s^5)$.

193 2. Any IPS refutation for $f_\Gamma(\mathbf{x})$ with depth $\leq \Gamma/10$ requires size $s^{\omega(1)}$.

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

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

203 Consider $g_\Delta(\mathbf{x}, y) - 1$. Using Theorem 1.3, we know that it does not have polynomial sized for-
204 mulas of depth $(\Delta/2 - 1)$. Moreover, due to the recent work on factors of constant-depth formu-
205 las [BKRRSS25], we also know that every multiple of the polynomial of depth $(\Delta/2 - \mathcal{O}(1))$ must
206 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 $\mathbf{x} \cup \{-1, 0, 1\}$.

207 depth $(\Delta/2 - \mathcal{O}(1))$. This property suffices for the multiples method to be applicable, as we explain
 208 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
 209 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
 210 $\{x_i^2 - x_i\}_i$.

211 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

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

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

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

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

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

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

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

239 **Proving the upper bound.** We work with the formula C that *computes* our hard polynomial
 240 instance. By construction of the polynomial, we have the guarantee that every gate in the formula
 241 evaluates to 0 or 1 over the hypercube. Using this fact, we prove by induction on the depth of the
 242 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
 243 Boolean axioms, where $\{g_1, \dots, g_t\}$ are the inputs to the gate g . This suffices to obtain the overall
 244 upper bound.

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

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

251 1.4 Results and Techniques: multilinear-NC¹ vs. multilinear-NC²

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

256 **Theorem 1.7** (multilinear NC¹ vs multilinear NC²-IPS). *Fix a field, \mathbb{F} of characteristic 0. For
 257 every growing parameter $N \in \mathbb{N}$, there is a multilinear polynomial $Q \in \mathbb{F}[x_1, \dots, x_N]$ which is
 258 unsatisfiable over $\{0, 1\}^N$ such that*

- 259 1. *There is a multilinear functional refutation for $Q(\mathbf{x})$, say $G(\mathbf{x})$, computable by a syntactic
 260 multilinear circuit of polynomial size and $O(\log^2 N)$ depth.*
- 261 2. *Any multilinear-NC¹- $\text{IPS}_{\text{LIN}'}$ for it requires size $N^{\Omega(\log N)}$.*

262 **Remark 1.8.** *Note that in the above theorem the lower bound holds for multilinear-NC¹- $\text{IPS}_{\text{LIN}'}$.
 263 However, the upper bound is not multilinear-NC²- $\text{IPS}_{\text{LIN}'}$. Instead, we only get that the refutation
 264 has multilinear-NC² circuits modulo the Boolean axioms. We do not get a bound on the complexity
 265 of the refutations of the Boolean axioms. In spite of this, we believe that the above result gives
 266 something we did not know before.*

- 267 • *Lower bounds for multilinear-NC¹- $\text{IPS}_{\text{LIN}'}$ are known since the work of [\[FSTW21\]](#). Their
 268 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 func-
 269 tional refutation is quite hard. Specifically, it encodes the Clique polynomial over the Boolean
 270 cube. This means that it cannot have small functional refutations unless VP equals VNP.
 271 (See [Appendix A](#)).*
- 272 • *It is known that there are subsystems of multilinear- IPS_{LIN} and multilinear- $\text{IPS}_{\text{LIN}'}$ that can
 273 refute interesting unsatisfiable instances (Section 4, [\[FSTW21\]](#)). For example, they can refute
 274 the subset-sum instances of the type $\sum_i \alpha_i x_i - \beta$, where α_i s are $O(1)$ and β is chosen such that
 275 the instance becomes unsatisfiable³. While such instances have multilinear upper bounds, the
 276 upper bound proofs seem to heavily rely on the fact that the subset-sum polynomial has degree
 277 1. Consider a simple instance $xy = 2$. This is an unsatisfiable instance over the Boolean
 278 cube. Here is one of its refutations:*

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

279 *Notice that the refutation for the Boolean axiom $y^2 - y$ is not multilinear. (There is an-
 280 other refutation for the same and in that, the refutation of the other Boolean axiom is not
 281 multilinear.) In fact, any refutation of this example is not multilinear. (See [Example 4.7](#)*

³They can allow slightly general α_i s. See Proposition 4.15 from [\[FSTW21\]](#) for more details.

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

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

288 To describe the components of the proofs, we start with a very simple example. Let $z \in \{0, 1\}$. In
289 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,
290 $\frac{1+z}{2}$ is a refutation of $2 - z$ modulo the Boolean axioms. We make use of this idea in our proof.
291 In order to prove the theorem, we again design a polynomial $p(\mathbf{x})$ such that it evaluates to 0 or 1
292 over the Boolean cube. Then, our unsatisfiable instance is $2 - p(\mathbf{x})$ and its functional refutation is
293 $(1 + p(\mathbf{x}))/2$.

294 If we can design a multilinear polynomial such that

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

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

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

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

313 2 Constant-depth Hierarchy

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

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

- 319 • $f_g(x_1, \dots, x_N)$ will denote the polynomial computed at the gate g .
- 320 • Let $\text{depth}(g)$ denote the depth of gate g , i.e. the length of the longest path from inputs to
- 321 the gate. Let $\text{size}(g)$ denote the number of wires in the sub-formula rooted at g . Finally, let
- 322 $\text{fanin}(g)$ denote the fan-in of gate g .

323 2.1 Adding addressing gadgets at sum gates

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

327 **Definition 2.1.** Let $n \in \mathbb{N}$. For each $0 \leq j \leq n$, let $t_n \in \mathbb{N}$ denote the smallest t_n such that
 328 $2^{t_n} > n$. Let $B_{n,0}(j) \subseteq \{0, 1, \dots, t_n\}$ denote the indices which are 0 in the binary representation of
 329 $j + 2^{t_n}$. Similarly, let $B_{n,1}(j) \subseteq \{0, 1, \dots, t_n\}$ denote the set of indices which are 1 in the binary
 330 representation of $j + 2^{t_n}$.
 331

The addressing gadget of j in n is defined as

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

332

333

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

336

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

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

341 *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

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

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

343

■

344 The following lemma shows how these addressing gadgets are applied:

345

346 **Lemma 2.3.** *Let $C(\mathbf{x})$ be a constant-free formula of size s and depth Δ computing some polynomial*
 347 *$f(\mathbf{x})$. 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 $t_g := t_{\text{fanin}(g)-1}$ and where $\mathbf{y}_g = \{y_{g,0}, \dots, y_{g,t_g}\}$ is a fresh set of variables for each addition gate, g . Then

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

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

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

- 350 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.*
- 351 2. *$f'(\mathbf{a}, \mathbf{b}) \in \{-1, 0, 1\}$ for any choice of $(\mathbf{a}, \mathbf{b}) \in \{0, 1\}^{|\mathbf{x}|+|\mathbf{y}|}$.*
- 352 3. *There exists $\mathbf{b} \in \mathbb{F}^{|\mathbf{y}|}$ such that $f'(\mathbf{x}, \mathbf{b}) = f(\mathbf{x})$.*
- 353 4. *$C'(\mathbf{x}, \mathbf{y})$ is of size $\mathcal{O}(s \log s)$ and depth at most $2\Delta + 2$.*

354 *Proof.* 1. This follows directly from [Lemma 2.2](#).

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

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

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

Then by [Lemma 2.2](#) every addition gate, g' , under this evaluation becomes

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

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

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

362 ■

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

365 **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
 366 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](#).*

367 2.2 Proof

368 *Proof of [Theorem 1.5](#).* Let \mathbb{F} be a field of characteristic zero and fix any depth $\Delta \in \mathbb{N}$. Let $Q_\Delta \in$
 369 $\mathbb{F}[x_1, \dots, x_n]$ be the polynomial from [[LST21](#), Theorem 2.1] satisfying the following conditions:

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

373 **Unsatisfiable instance.** Let $C(\mathbf{x})$ be the formula of depth Δ computing $Q_\Delta(\mathbf{x})$. Let $C'(\mathbf{x}, \mathbf{y})$
 374 denote the formula we get after applying the process mentioned in [Lemma 2.3](#) to $C(\mathbf{x})$, and let
 375 $f'_\Delta(\mathbf{x}, \mathbf{y})$ denote the polynomial computed by $C'(\mathbf{x}, \mathbf{y})$. By [Lemma 2.3](#), C' is of depth $2\Delta + 2$ and
 376 size $\mathcal{O}(s \log s)$.

377 Define the polynomial f_Δ as

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

378 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
 379 [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
 380 over the Boolean cube. Let C_Δ denote the formula for f_Δ of depth $2\Delta + 2$ and size $s_\Delta = \mathcal{O}(s \log s)$.

381 Now we show that f_Δ satisfies the two properties stated in [Theorem 1.5](#). That is, we prove upper
 382 and lower bounds on the complexity of the refutation of f_Δ .

383 2.2.1 Upper bound on the refutation of f_Δ

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

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

387 In order to prove the lemma, we use induction on the structure of C_Δ . We prove the following
 388 inductive lemma, which implies [Lemma 2.5](#).

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

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

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

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

392 • and $\text{depth}(E_{g,i}), \text{depth}(F_{g,i})$ is at most $2 \cdot \text{depth}(g)$.

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

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

396 where

397 • For every j , the polynomial E_j can be computed by a formula of depth $2(2\Delta + 2)$ and size
 398 $100 \cdot s_\Delta^4$.

399 • For every j , the polynomial F_j can be computed by a formula of depth $2(2\Delta + 2)$ and size
 400 $100 \cdot s_\Delta^4$.

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

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

402 Thus we have an IPS refutation for f_Δ of depth $2(2\Delta + 2) + 2$ and size at most $s_\Delta + O(1) + (n \cdot$
 403 $(100 \cdot s_\Delta^4 + O(1)) + (m \cdot (100 \cdot s_\Delta^4 + O(1)))$. As both m and n are bounded by s_Δ , we get that this
 404 quantity is bounded by $100s_\Delta^5$. This completes the proof⁴ of [Lemma 2.5](#). In what follows, we prove
 405 [Lemma 2.6](#).

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

408 case 1: $g = \prod_{\ell=0}^{r-1} g_\ell$.

409 In this case,

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

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

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

411 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
 412 $H_\ell = \prod_{t<\ell} g_t \cdot \prod_{t>\ell} g_t^2$. Then note that H_ℓ has size at most $2 \cdot \text{size}(g)$. By induction hypothesis for

⁴Note that we get a bound on the size of the Nullstellensatz refutation, thus a bound on IP_{LIN} refutation.

413 $g_\ell^2 - g_\ell$, we get that

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

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

416 **Size bound.** Before we start the analysis, we recall that we measure the size of a formula by the
417 number of wires in the formula. We also note some bounds on our parameters. We will assume
418 that $s_g > 1$. Let s_g be the short-hand for $\text{size}(g)$ and for a fixed g let $s_{g, \ell}$ denote $\text{size}(g_\ell)$ for
419 $0 \leq \ell \leq r - 1$.

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

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

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

$$423 \text{ size of } E_{g, i} \leq \left(\sum_{\ell} 2s_g + 100s_{g, \ell}^4 \right) + 3r \leq 3 \cdot s_g \cdot r + 3r + 100 \cdot (s_g - 1)^4.$$

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

425 The first bound comes from applying the bounds for H_ℓ , $E_{g_\ell, i}$ and counting the wires feeding
426 into the outer summation. The second bound comes from using [Equation \(3\)](#) and the fact that
427 $s_g - r \leq s_g - 1$.

$$428 \text{ size of } E_{g, i} \leq 6 \cdot s_g \cdot r + 100 \cdot s_g^4 - 100s_g^3/2 \leq 100 \cdot s_g^4.$$

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

431 case 2: $g = \sum_{\ell=0}^{r-1} g_\ell \cdot A_{g, \ell}$.

432 In order to prove this case, we will make use of a couple of simple lemmas.

433 **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
434 $\text{size}(C_{\ell, j}) \leq 6 \cdot r$.*

435 *Proof.* Notice that for any addressing gadget attached to a gate g , it only uses variables from \mathbf{y}_g .
436 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
437 some partition of the indices of \mathbf{y}_g into Y and Y' .

438 Then, we get

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

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

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

440 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
 441 gates could be one of the following: either a variable appears as itself, or its square appears, or as
 442 $(1 - y)$ or as $(1 - y)^2$. The size of each linear factor can be bounded by $3 \cdot t_g$ and hence, the overall
 443 size can be bounded by $6 \cdot t_g$. This is upper bounded by $6 \cdot r$. ■

444 **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 $j \in \{0, \dots, t_g\}$ such that*
 445 *$A_{g,\ell} \times A_{g,\ell'} = C_{\ell,\ell',j} \cdot (y_j^2 - y_j)$, where $\text{size}(C_{\ell,\ell',j}) \leq 6 \cdot r$.*

446 *Proof.* Here, it is easy to observe that for $\ell \neq \ell'$, there must exist a variable y_j such that either y_j
 447 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,
 448 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
 449 in \mathbf{y}_g variables. As in [Lemma 2.7](#), here again we get $\text{size}(C_{\ell,\ell',j}) \leq 6 \cdot r$. ■

450 We will now resume the proof of case 2, i.e., the case when g is a sum gate. We will again analyze
 451 $g^2 - g$.

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

452 where we added and subtracted the same quantity (second and third summation). After rearrang-
 453 ing, we get

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

454 We now apply induction on the first term and apply [Lemma 2.7](#) and [Lemma 2.8](#) on the second and
 455 third terms, respectively.

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

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

457 **Size bound.**

458 We will now bound each term in the expression above.

459 **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
 460 be bounded inductively. Finally, the numbers of wires feeding into the outer summation can be
 461 bounded by $3r$. Thus we have, the following bound

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

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

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

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

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

466 Finally, we bound the size of (III).

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

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

$$468 \text{size}(F_{g,i}) \leq 25r^3 + 3rs + 100 \sum_{\ell} s_{\ell}^4 \leq 25r^3 + 3rs + 100(s-1)^4 \leq 25r^3 + 3rs + 100s^4 - 100s^3/2 \leq 100s^4$$

469 **Depth bound.** The depth is bound is similar to the one we had in case 1 above. $\text{depth}(E_{g,i}) \leq$
 470 $2 \cdot (\text{depth}(E_{g_\ell,i})) + 2 \leq 2 \cdot (\text{depth}(g) - 1) + 2 \leq 2 \cdot \text{depth}(g)$. The depth of $F_{g,i}$ can be bounded
 471 similarly. ■

472 **2.2.2 Lower bound on the refutation of f_{Δ}**

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

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

478 *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
 479 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

$$1 - C'''(\mathbf{x}, \mathbf{y}, 0, \mathbf{x}^2 - \mathbf{x}, \mathbf{y}^2 - \mathbf{y}) = f_\Delta \cdot h,$$

480 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
 481 formula for $f_\Delta(\mathbf{x}, \mathbf{y})$ whose size is $\text{poly}(s'')$ 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

$$f_\Delta(\mathbf{x}, \mathbf{b}) = Q_\Delta(\mathbf{x}),$$

483 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 com-
 484 puting Q_Δ at depth $\Delta/2 - 1$ requires size $s^{\omega(1)}$.

486 Putting all this together, we get that if $\Delta'' + 10 \leq \Delta/2 - 1$ then $C'''(\mathbf{x}, \mathbf{y}, u, \mathbf{v}, \mathbf{z})$ requires size
 487 $s^{\omega(1)}$. ■

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

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

497 We now use Lemma 2.5 and Lemma 2.9 to finish the proof of Theorem 1.5. Note that the depth
 498 of the IPS refutation for f_Δ is $4\Delta + 6$ and size is $\text{poly}(s)$, whereas any circuit of depth less than
 499 $\Delta/2 - 11$ requires superpolynomial size. Thus, for $\Gamma = 4\Delta + 6$, we get that f_Γ has an IPS refutation
 500 of depth Γ and any IPS refutation of depth less than $\Gamma/10$ requires superpolynomial size. This
 501 completes the proof of Theorem 1.5. ■

502 3 Multilinear separation theorem

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

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

506 1. The following identity holds.

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

507 2. $2 - f(\mathbf{x}) = 0$ is unsatisfiable over the Boolean cube.

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

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

514 3.1 Multilinear-NC¹ vs. multilinear-NC²-IPS

515 In this section we prove [Theorem 1.7](#).

516 We will construct a polynomial such that it is computable by a multilinear NC² circuit and such
 517 that it only takes Boolean values over the Boolean hypercube. This along with [Lemma 3.1](#) will
 518 give us the desired separation.

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

522 We first recall the hard polynomial defined by [\[RY08\]](#), which is a simplification of the polynomial de-
 523 fined by [\[Raz04\]](#), which showed the first separation between multilinear formulas and circuits.

524 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
 525 number more than 0, then

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

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

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

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

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

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

535 We construct such a polynomial by modifying $F(\mathbf{u}, \mathbf{v})$. We design the polynomial using new set of
536 gadget variables, which we call \mathbf{w} . These will serve a dual purpose, first, they will help us create
537 addressing gadgets for the $+$ gates and they will assume the role of \mathbf{v} variables in the definition of
538 $F(\mathbf{u}, \mathbf{v})$.

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

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

540 where $\tilde{W}^{[i,j]}$ consists of variables we will use for the addressing gadgets. Recall that for $n \in \mathbb{N}$, t_n
541 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
542 a shorthand for $\ell([i+1, j-1])$. Let $\tilde{W}^{[i,j]} = \{w_r^{[i,j]} \mid 0 \leq r \leq t_{i,j}\}$.

543 Finally, we define $\mathbf{w} = \bigcup_{[i,j]} W^{[i,j]}$, where the union is over all intervals $[i, j]$, where $1 \leq i < j \leq n$
544 and $\ell([i, j])$ is even⁵. Here, the size of any set $W^{[i,j]}$ is $O(\log n)$ and hence we have $O(n^2 \log n)$ -
545 many w variables. We will use $m(n)$ to denote the cardinality of \mathbf{w} and m , when n is clear from
546 the context.

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

548 *1. If $\ell([i, j]) > 0$ and even, then*

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

549 *g_r is the addressing gadget, i.e.*

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

550 *Here, the sets $B_{n_{i,j},0}$ and $B_{n_{i,j},1}$ are defined as in [Definition 2.1](#).*

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

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

556

557 **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*
558 *from [Definition 3.4](#). Then, the following statements hold.*

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

562 Before we present the proof for the theorem, we will use it to prove our main theorem [Theorem 1.7](#),
563 which we recall below.

564

⁵ We only use intervals of even length inductively.

565 **Theorem 1.7** (multilinear NC¹ vs multilinear NC²-IPS). *Fix a field, \mathbb{F} of characteristic 0. For*
 566 *every growing parameter $N \in \mathbb{N}$, there is a multilinear polynomial $Q \in \mathbb{F}[x_1, \dots, x_N]$ which is*
 567 *unsatisfiable over $\{0, 1\}^N$ such that*

- 568 1. *There is a multilinear functional refutation for $Q(\mathbf{x})$, say $G(\mathbf{x})$, computable by a syntactic*
 569 *multilinear circuit of polynomial size and $O(\log^2 N)$ depth.*
- 570 2. *Any multilinear-NC¹-IPS_{LIN'} for it requires size $N^{\Omega(\log N)}$.*

571 *Proof of Theorem 1.7.* Let $N = n + m$, where n is the cardinality of \mathbf{u} and m is the cardinality of
 572 \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
 573 $Q(\mathbf{x})$ is unsatisfiable over the Boolean hypercube. Using Lemma 3.1, we know that the refutation
 574 for $Q(\mathbf{x})$ is $((P(\mathbf{x})) + 1)/2$ modulo the Boolean axioms. Thus, the functional refutation of $Q(\mathbf{x})$ is
 575 computable by multilinear circuit of size $\text{poly}(n)$ and depth $O(\log^2 n)$.

576 Moreover, from Theorem 3.5 we know that $P(\mathbf{x})$ requires multilinear NC¹ circuit of size $n^{\Omega(\log n)}$.
 577 As the refutation is $(P(\mathbf{x}) + 1)/2$, we also get that the multilinear NC¹-IPS_{LIN'} refutation⁶ for it
 578 must have size $n^{\Omega(\log n)}$. As N and n are polynomially related, this also gives a $N^{\Omega(\log N)}$ lower
 579 bound on the proof size of multilinear NC¹-IPS_{LIN} refutations. ■

580 3.2 Proof of Theorem 3.5

581 **Part 1 of Theorem 3.5.** We prove this statement by using the inductive structure of $P(\mathbf{u}, \mathbf{w})$.
 582 Specifically, we will show that for any interval $[i, j]$, the polynomial corresponding to it, i.e.,
 583 $p_{i,j}(\mathbf{u}, \mathbf{w}) \in \{0, 1\}$ when evaluated over the Boolean hypercube. We induct on the length of the
 584 interval. We only need to consider even length intervals.

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

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

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

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

588 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 $w_{\text{leaf}}^{[i,j]} = 1$
 589 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 $p_{i+1,j-1}$.

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

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

591 Now, suppose the variables in the set $\tilde{W}^{[i,j]}$ are set to 0s and 1s such that the boolean assignment
 592 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 $p_{r+1,j} \in \{0, 1\}$.
 593 That finishes the proof.

⁶ As we use functional method, we get a lower bound in IPS_{LIN'} and not just for IPS_{LIN}.

594 **Part 2 of Theorem 3.5.** Notice that polynomial $F(\mathbf{u}, \mathbf{v})$ defined by [RY08] is very similar to
 595 $P(\mathbf{u}, \mathbf{w})$. Instead of \mathbf{v} variables, we have small local changes using the addressing gadgets. The
 596 addressing gadgets themselves are constant-depth (unbounded fan-in) multilinear circuits. It is
 597 easy to see that by implementing the inductive definition of $p(\mathbf{u}, \mathbf{v})$, we will obtain a polynomial
 598 size and polynomial depth multilinear circuit. By using a depth reduction result of [RY08], we can
 599 obtain a polynomial size and $O(\log^2 n)$ -depth multilinear circuit for the polynomial.

600 **Part 3 of Theorem 3.5.** Firstly, we will prove that for every partition of variables in \mathbf{u} into two
 601 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
 602 then imply a lower bound for the IPS proof size. This step is quite standard, but we will present
 603 it for completeness.

604 **Lemma 3.6.** *Let $n \in \mathbb{N}$ and $p_{1,2n}$ be as defined above. Then, for any partition of \mathbf{u} into $\mathbf{y} \cup \mathbf{z}$
 605 each of cardinality n , there exists an assignment to variables in \mathbf{w} to field constants, such that
 606 $\text{rank}(M_{\mathbf{y}, \mathbf{z}}(p_{1,2n})) = 2^n$.*

607 *Proof.* We will prove this by induction on n .

608 **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
 609 get $p_{1,2} = \frac{1}{2} \cdot (1 + u_1 \cdot u_2)$ and the statement trivially holds.

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

612 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,
 613 $p_{1,2n} = \frac{1}{2} (1 + u_1 \cdot u_{2n}) \cdot p_{2,2n-1}$. By induction hypothesis, $p_{2,2n-1}$ is full rank, i.e. 2^{n-1} , under every
 614 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
 615 not use the variables u_1 and u_{2n} . Hence, we are done in this case.

616 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
 617 $[r + 1, 2n]$ evenly split \mathbf{y} and \mathbf{z} variables. We set $w_{\text{top}}^{[1,2n]} = 1$ and the variables in the addressing
 618 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
 619 $p_{1,r}, p_{r+1,2n}$ and observing that the two polynomials do not share any variables we get the desired
 620 bound on the rank of $p_{1,2n}$. ■

621 References

- 622 [And25] Robert Andrews. “Algebraic Pseudorandomness in VNC⁰”. In: *40th Computational*
 623 *Complexity Conference (CCC 2025)*. Ed. by Srikanth Srinivasan. Vol. 339. Leib-
 624 niz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss
 625 Dagstuhl – Leibniz-Zentrum für Informatik, 2025, 15:1–15:15. ISBN: 978-3-95977-
 626 379-9. DOI: [10.4230/LIPIcs.CCC.2025.15](https://doi.org/10.4230/LIPIcs.CCC.2025.15). URL: [https://drops.dagstuhl.de/](https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.CCC.2025.15)
 627 [entities/document/10.4230/LIPIcs.CCC.2025.15](https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.CCC.2025.15) (cit. on p. 4).

- 628 [AF22] Robert Andrews and Michael A. Forbes. “Ideals, determinants, and straightening:
629 proving and using lower bounds for polynomial ideals”. In: *Proceedings of the 54th*
630 *Annual ACM SIGACT Symposium on Theory of Computing*. STOC 2022. Rome,
631 Italy: Association for Computing Machinery, 2022, pp. 389–402. ISBN: 9781450392648.
632 DOI: [10.1145/3519935.3520025](https://doi.org/10.1145/3519935.3520025). URL: [https://doi.org/10.1145/3519935.](https://doi.org/10.1145/3519935.3520025)
633 [3520025](https://doi.org/10.1145/3519935.3520025) (cit. on pp. 1, 4).
- 634 [BLRS25] Amik Raj Behera, Nutan Limaye, Varun Ramanathan, and Srikanth Srinivasan.
635 “New Bounds for the Ideal Proof System in Positive Characteristic”. In: *52nd In-*
636 *ternational Colloquium on Automata, Languages, and Programming (ICALP 2025)*.
637 Ed. by Keren Censor-Hillel, Fabrizio Grandoni, Joël Ouaknine, and Gabriele Pup-
638 pis. Vol. 334. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl,
639 Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025, 22:1–22:20.
640 ISBN: 978-3-95977-372-0. DOI: [10.4230/LIPIcs.ICALP.2025.22](https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ICALP.2025.22). URL: [https :](https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ICALP.2025.22)
641 [//drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ICALP.2025.22](https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ICALP.2025.22)
642 (cit. on pp. 1, 2).
- 643 [BKRRSS25] Somnath Bhattacharjee, Mrinal Kumar, Shanthanu S. Rai, Varun Ramanathan,
644 Ramprasad Satharishi, and Shubhangi Saraf. “Closure under factorization from a
645 result of Furstenberg”. In: *CoRR* abs/2506.23214 (2025). DOI: [10.48550/ARXIV.](https://arxiv.org/abs/2506.23214)
646 [2506.23214](https://arxiv.org/abs/2506.23214). arXiv: [2506.23214](https://arxiv.org/abs/2506.23214) (cit. on pp. 4, 6, 15).
- 647 [Bür98] Peter Bürgisser. “Completeness and Reduction in Algebraic Complexity Theory”.
648 In: 7 (Aug. 1998). DOI: [10.1007/978-3-662-04179-6_1](https://doi.org/10.1007/978-3-662-04179-6_1) (cit. on p. 23).
- 649 [CGMS25] Prerona Chatterjee, Utsab Ghosal, Partha Mukhopadhyay, and Amit Sinhababu.
650 *IPS Lower Bounds for Formulas and Sum of ROABPs*. July 2025. DOI: [10.48550/](https://arxiv.org/abs/2507.09515)
651 [arXiv.2507.09515](https://arxiv.org/abs/2507.09515). arXiv: [2507.09515](https://arxiv.org/abs/2507.09515) [cs.CC]. URL: [https://arxiv.org/abs/](https://arxiv.org/abs/2507.09515)
652 [2507.09515](https://arxiv.org/abs/2507.09515) (cit. on p. 1).
- 653 [CELS18] Suryajith Chillara, Christian Engels, Nutan Limaye, and Srikanth Srinivasan. “A
654 Near-Optimal Depth-Hierarchy Theorem for Small-Depth Multilinear Circuits”. In:
655 *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018,*
656 *Paris, France, October 7-9, 2018*. Ed. by Mikkel Thorup. IEEE Computer Society,
657 2018, pp. 934–945. DOI: [10.1109/FOCS.2018.00092](https://doi.org/10.1109/FOCS.2018.00092). URL: [https://doi.org/10.](https://doi.org/10.1109/FOCS.2018.00092)
658 [1109/FOCS.2018.00092](https://doi.org/10.1109/FOCS.2018.00092) (cit. on p. 7).
- 659 [CKS19] Chi-Ning Chou, Mrinal Kumar, and Noam Solomon. “Closure Results for Polyno-
660 mial Factorization”. In: *Theory of Computing* 15 (2019), pp. 1–34. DOI: [10.4086/](https://doi.org/10.4086/TOC.2019.V015A013)
661 [TOC.2019.V015A013](https://doi.org/10.4086/TOC.2019.V015A013). URL: <https://doi.org/10.4086/toc.2019.v015a013> (cit.
662 on p. 15).
- 663 [CR79] Stephen A. Cook and Robert A. Reckhow. “The relative efficiency of propositional
664 proof systems”. In: *Journal of Symbolic Logic* 44.1 (1979), pp. 36–50. DOI: [10.2307/](https://doi.org/10.2307/2273702)
665 [2273702](https://doi.org/10.2307/2273702) (cit. on p. 1).

- 666 [EGLT25] Tal Elbaz, Nashlen Govindasamy, Jiaqi Lu, and Iddo Tzameret. *Lower Bounds*
667 *against the Ideal Proof System in Finite Fields*. June 2025. DOI: [10.48550/arXiv.](https://doi.org/10.48550/arXiv.2506.17210)
668 [2506.17210](https://doi.org/10.48550/arXiv.2506.17210). arXiv: [2506.17210](https://arxiv.org/abs/2506.17210) [cs.CC]. URL: [https://arxiv.org/abs/2506.](https://arxiv.org/abs/2506.17210)
669 [17210](https://arxiv.org/abs/2506.17210) (cit. on pp. 1, 2).
- 670 [FSTW21] Michael A. Forbes, Amir Shpilka, Iddo Tzameret, and Avi Wigderson. “Proof Com-
671 plexity Lower Bounds from Algebraic Circuit Complexity”. In: *Theory Comput.* 17
672 (2021), pp. 1–88. URL: <https://theoryofcomputing.org/articles/v017a010/>
673 (cit. on pp. 1, 2, 4–7, 15, 23).
- 674 [GHT22] Nashlen Govindasamy, Tuomas Hakoniemi, and Iddo Tzameret. “Simple Hard In-
675 stances for Low-Depth Algebraic Proofs”. In: *2022 IEEE 63rd Annual Symposium*
676 *on Foundations of Computer Science (FOCS)*. 2022, pp. 188–199. DOI: [10.1109/](https://doi.org/10.1109/FOCS54457.2022.00025)
677 [FOCS54457.2022.00025](https://doi.org/10.1109/FOCS54457.2022.00025) (cit. on pp. 1, 2, 5).
- 678 [GP18] Joshua A. Grochow and Toniann Pitassi. “Circuit Complexity, Proof Complexity,
679 and Polynomial Identity Testing: The Ideal Proof System”. In: *J. ACM* 65.6 (Nov.
680 2018). ISSN: 0004-5411. DOI: [10.1145/3230742](https://doi.org/10.1145/3230742). URL: [https://doi.org/10.1145/](https://doi.org/10.1145/3230742)
681 [3230742](https://doi.org/10.1145/3230742) (cit. on pp. 1, 2).
- 682 [HLT24a] Tuomas Hakoniemi, Nutan Limaye, and Iddo Tzameret. “Functional Lower Bounds
683 in Algebraic Proofs: Symmetry, Lifting, and Barriers”. In: *Proceedings of the 56th*
684 *Annual ACM Symposium on Theory of Computing*. STOC 2024. Vancouver, BC,
685 Canada: Association for Computing Machinery, 2024, pp. 1396–1404. ISBN: 9798400703836.
686 DOI: [10.1145/3618260.3649616](https://doi.org/10.1145/3618260.3649616). URL: [https://doi.org/10.1145/3618260.](https://doi.org/10.1145/3618260.3649616)
687 [3649616](https://doi.org/10.1145/3618260.3649616) (cit. on p. 1).
- 688 [HLT24b] Tuomas Hakoniemi, Nutan Limaye, and Iddo Tzameret. “Functional Lower Bounds
689 in Algebraic Proofs: Symmetry, Lifting, and Barriers”. In: *Proceedings of the 56th*
690 *Annual ACM Symposium on Theory of Computing*. STOC 2024. Vancouver, BC,
691 Canada: Association for Computing Machinery, 2024, pp. 1396–1404. ISBN: 9798400703836.
692 DOI: [10.1145/3618260.3649616](https://doi.org/10.1145/3618260.3649616). URL: [https://doi.org/10.1145/3618260.](https://doi.org/10.1145/3618260.3649616)
693 [3649616](https://doi.org/10.1145/3618260.3649616) (cit. on pp. 2, 5).
- 694 [HS65] Juris Hartmanis and Richard E. Stearns. “On the Computational Complexity of Al-
695 gorithms”. In: *Transactions of the American Mathematical Society* 117 (May 1965),
696 pp. 285–306. ISSN: 0002-9947. DOI: [10.2307/1994208](https://doi.org/10.2307/1994208) (cit. on p. 1).
- 697 [Has86] J Hastad. “Almost optimal lower bounds for small depth circuits”. In: *Proceedings*
698 *of the Eighteenth Annual ACM Symposium on Theory of Computing*. STOC ’86.
699 Berkeley, California, USA: Association for Computing Machinery, 1986, pp. 6–20.
700 ISBN: 0897911938. DOI: [10.1145/12130.12132](https://doi.org/10.1145/12130.12132). URL: [https://doi.org/10.1145/](https://doi.org/10.1145/12130.12132)
701 [12130.12132](https://doi.org/10.1145/12130.12132) (cit. on p. 1).

- 702 [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
703 matching upper bound yielding a nonuniform circuit size hierarchy. Springer, 2012.
704 ISBN: 978-3-642-24507-7 (cit. on p. 1).
705
- 706 [LST21] Nutan Limaye, Srikanth Srinivasan, and Sébastien Tavenas. “Superpolynomial Lower
707 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](https://doi.org/10.1109/FOCS52979.2021.00083) (cit. on pp. 3–5, 10, 14, 15).
708
709
- 710 [LST25] Jiaqi Lu, Rahul Santhanam, and Iddo Tzameret. *$AC^0[p]$ -Frege Cannot Efficiently
711 Prove that Constant-Depth Algebraic Circuit Lower Bounds are Hard*. Tech. rep.
712 TR25-134. Electronic Colloquium on Computational Complexity (ECCC), Sept.
713 2025. URL: <https://eccc.weizmann.ac.il/report/2025/134/> (cit. on p. 1).
- 714 [Raz04] Ran Raz. “Multilinear- $NC_1 \neq$ Multilinear- NC_2 ”. In: *Proceedings of the 45th Annual
715 IEEE Symposium on Foundations of Computer Science*. FOCS ’04. USA: IEEE
716 Computer Society, 2004, pp. 344–351. ISBN: 0769522289. DOI: [10.1109/FOCS.2004.42](https://doi.org/10.1109/FOCS.2004.42). URL: <https://doi.org/10.1109/FOCS.2004.42> (cit. on pp. 3, 7, 16).
717
- 718 [RY08] Ran Raz and Amir Yehudayoff. “Balancing Syntactically Multilinear Arithmetic
719 Circuits”. In: *computational complexity* 17.4 (Dec. 2008), pp. 515–535. ISSN: 1420-
720 8954. DOI: [10.1007/s00037-008-0254-0](https://doi.org/10.1007/s00037-008-0254-0). URL: <https://doi.org/10.1007/s00037-008-0254-0> (cit. on pp. 3, 7, 16, 19).
721
- 722 [RY09] Ran Raz and Amir Yehudayoff. “Lower Bounds and separations for constant depth
723 multilinear circuits”. English. In: *Computational Complexity* 18.2 (June 2009), pp. 171–
724 207. ISSN: 1016-3328. DOI: [10.1007/s00037-009-0270-8](https://doi.org/10.1007/s00037-009-0270-8) (cit. on p. 7).
- 725 [ST21] Rahul Santhanam and Iddo Tzameret. “Iterated Lower Bound Formulas: A Diagonalization-
726 Based Approach to Proof Complexity”. In: *Proceedings of the 53rd Annual ACM
727 SIGACT Symposium on Theory of Computing (STOC 2021)*. ECCC TR21-138;
728 journal version in *SIAM J. Comput.*, 2025. ACM, 2021, pp. 234–247. DOI: [10.1145/3406325.3451010](https://doi.org/10.1145/3406325.3451010) (cit. on p. 1).
729
- 730 [Sha49] Claude. E. Shannon. “The synthesis of two-terminal switching circuits”. In: *The
731 Bell System Technical Journal* 28.1 (1949), pp. 59–98. DOI: [10.1002/j.1538-7305.1949.tb03624.x](https://doi.org/10.1002/j.1538-7305.1949.tb03624.x) (cit. on p. 1).
732
- 733 [Sip83] Michael Sipser. “Borel sets and circuit complexity”. In: *Proceedings of the Fifteenth
734 Annual ACM Symposium on Theory of Computing*. STOC ’83. New York, NY,
735 USA: Association for Computing Machinery, 1983, pp. 61–69. ISBN: 0897910990.
736 DOI: [10.1145/800061.808733](https://doi.org/10.1145/800061.808733). URL: <https://doi.org/10.1145/800061.808733>
737 (cit. on p. 1).

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

742 A The complexity of refuting lifted subset-sum

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

748

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

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

753 In order to prove the lemma, we start with some notation and preliminaries. Let $V \subseteq [n]$ and let
754 $K_V = \{(i, j) \mid i, j \in V, i < j\}$. Let $e = (i, j)$ denote a pair from the set K_V , then we use z_e to denote
755 $z_{i,j}$.

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

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

757 $C_{n/2}(\mathbf{x}, \mathbf{z})$ is VNP complete⁷.

758 We are now ready to prove [Lemma A.1](#).

759 *Proof of Lemma A.1.* We introduce some notation. We use $\binom{[n]}{2}$ to denote the set $\{(i, j) \mid i, j \in$
760 $[n], i < j\}$. Let the subset-sum instance (without the lift) be

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

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

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

⁷The 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 $x_i s = 1$ in the above polynomial, we can obtain this polynomial.

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

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

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

768 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
 769 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
 770 easy to see that the functional refutation of $F(\mathbf{x}, \mathbf{z})$ can be obtained from the refutation of $f(\mathbf{z})$ by
 771 monomial substitution. This is because we only need to preserve the refutation over the Boolean
 772 cube. Such a monomial substitution can result in a non-multilinear polynomial. Let $\text{ml}[\cdot]$ denote
 773 the following map defined for monomials over a set of variables, say \mathbf{y} : $\text{ml}[\prod_i y_i^{a_i}] = \prod_i y_i^{\min\{a_i, 1\}}$.
 774 The map extends linearly and can be defined as a map from $\mathbb{F}[\mathbf{y}] \rightarrow \mathbb{F}[\mathbf{y}]$ for any polynomial ring
 775 $\mathbb{F}[\mathbf{y}]$.

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

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

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