Iterated Lower Bound Formulas: A Diagonalization-Based Approach to Proof Complexity

Rahul Santhanam
University of Oxford
United Kingdom
rahul.santhanam@cs.ox.ac.uk

Iddo Tzameret
Imperial College London
United Kingdom
iddo.tzameret@gmail.com

ABSTRACT

We propose a diagonalization-based approach to several important questions in proof complexity. We illustrate this approach in the context of the algebraic proof system IPS and in the context of propositional proof systems more generally.

We use the approach to give an explicit sequence of CNF formulas $\{\phi_n\}$ such that VNP $\neq$ VP iff there are no polynomial-size IPS proofs for the formulas ϕ_n. This provides a natural equivalence between proof complexity lower bounds and standard algebraic complexity lower bounds. Our proof of this fact uses the implication from IPS lower bounds to algebraic complexity lower bounds due to Grochow and Pitassi together with a diagonalization argument: the formulas ϕ_n themselves assert the non-existence of short IPS proofs for formulas encoding VNP $\neq$ VP at a different input length. Our result also has meta-mathematical implications: it gives evidence for the difficulty of proving strong lower bounds for IPS within IPS.

For any strong enough propositional proof system R, we define the *iterated R-lower bound formulas*, which inductively assert the non-existence of short R proofs for formulas encoding the same statement at a different input length, and propose them as explicit hard candidates for the proof system R. We observe that this hypothesis holds for Resolution following recent results of Atserias and Müller and of Garlik, and give evidence in favour of it for other proof systems.

CCS CONCEPTS

• **Theory of computation** → **Proof complexity**; **Algebraic complexity theory**; **Circuit complexity**; **Complexity theory and logic**.

KEYWORDS

diagonalization, proof complexity lower bounds, circuit complexity lower bounds, iterated lower bound formulas, Ideal Proof System, algebraic complexity

ACM Reference Format:
Rahul Santhanam and Iddo Tzameret. 2021. 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 '21), June 21–25, 2021, Virtual, Italy.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3406325.3451010

STOC '21, June 21–25, 2021, Virtual, Italy

ACM ISBN 978-1-4503-8053-9/21/06…$15.00
https://doi.org/10.1145/3406325.3451010

1 INTRODUCTION

Diagonalization has been used to show many of the foundational results in logic, including Cantor's theorem about the uncountability of the reals, Gödel's Incompleteness Theorems, and Turing's proof of the undecidability of the Halting Problem. It has also found extensive application in complexity theory, where many unconditional lower bounds use diagonalization directly or indirectly. However, surprisingly in the area of resource bounded proofs, namely proof complexity, diagonalization has had very little impact in so far.

Proof complexity studies the question of whether tautologies have short proofs in a given proof system. Cook and Reckhow [6] showed that NP $\neq$ coNP if and only if there are hard instances for every propositional proof system R, i.e., a sequence of instances that does not have short R-proofs. It is known that there are hard instances for relatively weak proof systems such as Resolution [12] and constant-depth Frege [1, 18, 21], but thus far it remains completely open whether the same is the case for the proof systems Frege and Extended Frege. Lower bounds on these systems seem extremely hard to attain, and there is even a shortage of good candidates for hard instances.

In this paper, we propose a diagonalization-based approach to make progress on several important directions in proof complexity, including *connections between proof complexity and circuit complexity*, *meta-mathematics of proof complexity lower bounds* and *explicit hard instances for propositional proof systems*. We first give some background on each of these directions.

Connections Between Proof Complexity and Circuit Complexity: It is an intriguing fact that many of the known proof complexity lower bounds are shown using techniques that were originally developed in the context of circuit complexity, e.g., the technique of random restrictions. Intuitively, it seems as though progress on lower bounds in both areas is stalled for similar reasons, but there are few formal connections known between the two areas. For weak proof systems, the notion of *feasible interpolation* [13] provides such a connection, enabling us to derive small circuits for certain computational problems from short proofs of related formulas. In the converse direction, recent lifting results [10] give a way to derive circuit complexity lower bounds for weak circuit classes from proof complexity lower bounds for related weak proof systems.

For strong proof systems, there are some intriguing connections known to algebraic complexity (cf. [22] for a survey). First, the strong algebraic proof system IPS (Ideal Proof System) introduced by Grochow and Pitassi in [11] was introduced in part to relate

proof complexity lower bounds to complexity class separations such as VP vs. VNP. Furthermore, we know that some natural conjectures from algebraic circuit complexity, such as the Shub and Smale conjecture [31] about hardness of expressing factorial numbers with $0, 1, -1$ constants and $\times, +$ gates, imply lower bounds on the IPS as shown recently in [3]. Moreover, unconditional lower bounds on some restricted versions of IPS have been shown to follow from certain algebraic circuits lower bounds in [7], while [19] established connections between Frege lower bounds and lower bounds on the weak model of noncommutative formulas. Finding further connections is an important problem, so that progress in either area can be transferred to the other.

Meta-mathematics of Proof Complexity Lower Bounds: Proof complexity lower bounds seem to be difficult to show, but there seems to be little formal justification for this. In contrast, the difficulty of showing computational complexity lower bounds is evidenced by barriers such as the relativization barrier and the natural proofs barrier. Understanding the barriers to lower bounds better in the context of proof complexity might help us to make progress. One approach is via connections to circuit complexity. Grochow and Pitassi [11] show that super-polynomial lower bounds for CNFs in the algebraic proof system IPS imply VNP ≠ VP; thus if we believe VNP ≠ VP is hard to show, then we must believe the same for IPS lower bounds. In a recent work, [20] take a different approach, formulating an analogue of the natural proofs barrier for proof complexity. They show unconditionally that for some (non-uniform) propositional proof system R, super-polynomial lower bounds on R-proofs for random truth table formulas cannot be shown efficiently in any non-uniform propositional proof system. However, their proof is non-constructive, and does not seem to yield any new information for commonly studied proof systems such as Frege and Extended Frege.

Explicit Hard Instances for Propositional Proof Systems: Known proof complexity lower bounds for proof systems such as Resolution and constant-depth Frege hold for explicit formulas, indeed the Pigeonhole Principle is hard in both cases. However, for Frege and above, there are few explicit candidates for hard instances, as discussed in [27]. Random CNFs of constant clause to variable density and random truth table formulas are plausible candidates, but are not explicit. The only plausible explicit candidates of which we are aware, apart from canonical examples such as reflection principles for strong proof systems[1], are explicit circuit lower bound tautologies and the more general proof complexity generators [2, 14]. However, even explicit circuit lower bound tautologies are no longer hard candidates for strong propositional proof systems that *can* prove circuit lower bounds - such a strong proof system can be defined by adding a circuit lower bound tautology as an axiom to Frege, if circuit lower bounds do indeed hold for functions in exponential time. In general, there is a need for more plausible candidates for hardness, prior to any approach to showing lower bounds for strong proof systems.

We make progress along the three directions above in the context of the algebraic proof system IPS and for propositional proof systems in general. We next explain our results.

[1]That is, principles expressing the soundness of proof systems that are conjectured to be stronger than the proof system we wish to lower bound.

1.1 Our Results

Our main technical result is an *equivalence* between super-polynomial algebraic circuit lower bounds for the Permanent and IPS lower bounds for a certain sequence of explicit CNFs. In the following informal statement of the result, we use "VNP ≠ VP" to denote appropriate CNF encodings of VNP ≠ VP (itself a sequence of statements asserting that the Permanent, as given by its list of monomials at a given input length, lacks small algebraic circuits) over a *finite* field $\mathbb{F}$, and $\mathsf{lb}_{IPS}(\text{“}\phi_n\text{”}, s)$ to denote appropriate CNF encodings of the statements that there are no IPS-proofs of ϕ_n of size $s(|\phi|)$, where $s : \mathbb{N} \to \mathbb{N}$ is a function (below, m refers to the size of "VNP ≠ VP").

Theorem 1.1 (Equivalence between Algebraic Circuit Lower Bounds and IPS Lower Bounds; informal). VNP ≠ VP *iff there is a constant c such that there are no polynomial-size IPS proofs of* $\mathsf{lb}_{IPS}(\text{“VNP} \neq \text{VP”}, m^c)$.

Note that the explicit formulas $\{\phi_n\}$ which lack efficient IPS proofs iff VNP ≠ VP are *themselves* encodings of IPS lower bounds for the statement VNP ≠ VP - this is how we use diagonalization. Our argument combines diagonalization ideas with the result of Grochow and Pitassi [11] that IPS lower bounds for CNFs imply circuit lower bounds. It is worth mentioning that the theorem is interesting only for IPS proofs of polynomial degrees (when the degree of the proofs is too big, the encoding $\{\phi_n\}$ itself becomes too big and hence provable in small size, relative to $|\{\phi_n\}|$). We give more details on the proof ideas in the next section. Our proof ideas generalize to give an analogue of Theorem 1.1 for *every* algebraic proof system that efficiently simulates IPS.

Theorem 1.1 is relevant to two of the three directions we cited as motivation earlier. It gives a new connection between algebraic complexity and proof complexity, which could be useful in either direction. It is also relevant to the meta-mathematics of IPS lower bounds. For a certain natural explicit family of formulas expressing algebraic circuit lower bounds for the Permanent, super-polynomial IPS lower bounds are *themselves* hard to show in IPS, if we believe that VNP ≠ VP. Thus, under a standard complexity conjecture, IPS finds it hard to reason about itself.

One possibility, of course, is that VNP ≠ VP actually *has* polynomial-size IPS proofs, in which case $\mathsf{lb}_{IPS}(\text{“VNP} \neq \text{VP”}, m^{\omega(1)})$ is not a tautology, and hence trivially lacks small IPS proofs. This, however, cannot happen if we assume a natural algebraic analogue of Razborov's Conjecture [16, 27]. Razborov conjectured that under standard circuit complexity assumptions, Frege cannot efficiently prove super-polynomial circuit lower bounds for any Boolean function (as expressed in the so-called truth table formulas). A reasonable algebraic analogue of Razborov's Conjecture is that IPS cannot efficiently prove super-polynomial algebraic circuit lower bounds for any polynomial, which rules out the possibility above.

Theorem 1.1 gives evidence that IPS finds it hard to reason efficiently about itself - could this hold for proof systems more generally? A heuristic way to think of Theorem 1.1 is as a *fixed point* theorem in the following sense: consider $\mathsf{lb}_R(\cdot, m^{\omega(1)})$ for a proof system R as an operator mapping formulas to formulas. Let lb_R^2 be the composition of this operator with itself. Then the sequence of formulas expressing that VNP ≠ VP is a fixed point for lb_R^2 in the

sense that it preserves truth when R is IPS. Indeed, our diagonalization approach is inspired partly by [4, 9] who showed implicitly that *every* sequence of formulas is a fixed point for lb_R^2 when R is Resolution, and by [20] who showed implicitly that for every strong enough (nonuniform) propositional proof system R (simulating Extended Frege), the *distribution* of random truth table formulas is a fixed point for lb_R^2.

Here we explore the idea that iterating lb_R provides an explicit hard sequence of formulas for R. Assume that R is not polynomially bounded, and let ϕ be a fixed formula that does not have $|\phi|^c$ size R-proofs, for some constant c. We define the *iterated lower bound formulas* $\mathrm{lb}_R^k(\phi, n^c)$ inductively as follows:

(1) $\mathrm{lb}_R^0(\phi, n^c) = \phi$;
(2) $\mathrm{lb}_R^{k+1}(\phi, n^c) = \mathrm{lb}_R(\mathrm{lb}_R^k(\phi, n^c), n^c)$.

We propose the **Iterated Lower Bound Hypothesis**: for every reasonably strong[2] propositional proof system R that is not polynomially bounded, there is a ϕ such that the sequence $\{\mathrm{lb}_R^k(\phi)\}$ is a sequence of hard instances for R.

This generically gives a candidate family of hard instances for *every* strong enough proof system. We believe that there is a win-win aspect to studying this hypothesis - even if it fails, this gives us information about whether propositional proof systems can reason about lower bounds for themselves. We are not currently aware of any *natural* propositional proof system R that is capable of this. Though there have been discussions on whether constant-depth Frege for example can efficiently prove its known lower bounds, even if this is the case our hypothesis would make sense for strong proof systems, as well as for weak proof systems.

We provide some evidence in favour of the hypothesis. First, as a corollary to [4, 9], it follows that the hypothesis holds for Resolution.

Theorem 1.2. *The Iterated Lower Bounds Hypothesis holds for Resolution.*

Second, we show that under a conjecture of Rudich [29] about non-existence of short propositional proofs for random truth table formulas, any finite number of iterations preserves hardness for random truth table formulas, for some strong propositional proof system R that efficiently simulates Extended Frege.

Theorem 1.3. *There is a propositional proof system R efficiently simulating Extended Frege such that under Rudich's Conjecture, for every large enough constant c and every positive integer k, $\mathrm{lb}_R^k(\phi, n^c)$ does not have R-proofs of size $|\mathrm{lb}_R^k(\phi, n^c)|^c$ with high probability over ϕ a random truth table formula.*

1.2 Overview of Techniques

We give a high-level overview of the ideas required to show Theorem 1.1. Informally, we would like to show that VNP ≠ VP iff IPS cannot efficiently prove IPS lower bounds for "VNP ≠ VP".

The proof builds on several technical constructions. The gist of the argument is a form of diagonalisation: the existence of a short proof of a proof complexity lower bound for the statement VNP ≠ VP implies in fact that there *is* a short proof of VNP ≠ VP (at a smaller input length) due to the reduction of Grochow and Pitassi [11], that we show is efficiently formalizable already inside IPS.

For this purpose we make the following assumptions, which we will justify later: (1) There is a reasonable CNF encoding of "VNP ≠ VP"; (2) there is a reasonable CNF encoding of the statement that there are no IPS lower bounds of size s for a CNF ϕ; (3) if ϕ is a tautology and there are short IPS proofs of "IPS does not efficiently prove ϕ", then there are short IPS proofs of "VNP ≠ VP". Assumption 3 can be thought of as the formalization of the Grochow-Pitassi implication from IPS lower bounds for CNFs to VNP ≠ VP within IPS. A priori, it is hard to see how to establish Assumption 3 since we only know that ϕ is a tautology and do not have proofs of this fact. It will turn out that the parameters can be set so that truth-table proofs of ϕ suffice.

Given these assumptions, we proceed as follows. First, we show the forward direction. Assume VNP ≠ VP. Then "VNP ≠ VP" is a CNF tautology, using Assumption 1. Assume for the sake of contradiction that lb_{IPS}("VNP ≠ VP", $n^{\omega(1)}$) has polynomial-size proofs in IPS. Then by Assumption 3, "VNP ≠ VP" has polynomial-size proofs in IPS. But this contradicts the soundness of IPS, since IPS has polynomial-size proofs of the statement that "VNP ≠ VP" requires superpolynomial-size IPS-proofs.

For the backward direction, either ϕ = "lb_{IPS}("VNP ≠ VP", $n^{\omega(1)}$)" is true or it is not. If it is true, then $\mathrm{lb}_{IPS}(\phi, n^{\omega(1)})$ implies VNP ≠ VP by [11], using Assumption 2. If it is false, then "VNP ≠ VP" has poly-size proofs in IPS. By the soundness of IPS, this implies VNP ≠ VP.

The above is only a template and hides many technical issues and details. For example, in practice, we work with the statement VNP ≠ VP at a given input length, and we need to clarify how the input lengths of different occurrences of the statement relate to each other.

Technically, for the sake of the formalisation in IPS we need to be able to speak at the IPS proof level about circuit lower bounds, proof complexity lower bounds, and the reduction between them as in [11]. To formalise statements about IPS proofs and circuit class separations we express polynomials as vectors of coefficients. To express the existence of small circuits we use universal circuits as defined in Raz [26]. To express that a given polynomial is computable by a small circuit we use a set of equations. Each equation states that the coefficient of a monomial computed by the universal circuit has the appropriate value.

Using this formalisation of polynomials computed by small circuits we can encode VP ≠ VNP as the statement expressing that the coefficient vector of the permanent polynomial is not equal to the coefficient vector of any small universal circuit. Similarly, the IPS proof predicate is expressed by stating the existence of a small universal circuit that computes (similarly, based on its monomial coefficients) the IPS certificate of a given CNF. For the purpose of expressing statements about algebraic circuits such as VP ≠ VNP as CNF formulas we first need to work over finite fields, and second need to devise ways to move from CNF formulas encoding circuits to the circuit they express.

1.3 Related Work

Following on unpublished work of Friedman [8], Pudlak [23, 24] showed finitistic versions of Gödel's Theorem: for any strong enough first-order theory T of arithmetic, there is a constant $\epsilon > 0$

[2] We formally define "reasonably strong" later.

such that the finitistic consistency principle $Con_T(n)$ stating that there are no T-proofs of $1 = 0$ of size at most n requires T-proofs of size at least n^{ϵ}. For first-order theories satisfying additional properties, stronger lower bounds approaching n can be obtained [24]. By the standard translation between first-order theories and propositional proof systems, this yields non-trivial lower bounds on the Q-proof size of the reflection principle for Q when Q is a strong enough propositional proof system. However, the lower bounds obtained in this way are *sub-linear* in the formula size, while we are interested in the question of *super-polynomial* size lower bounds. Note that the reflection principle in fact has proofs of polynomial size [23].

Diagonalization techniques have also been explored in work of Krajicek [15, 17]. In [17], the notion of *implicit proofs* is defined and studied. This concept is used in [15] to show a conditional result: if E requires exponential-size circuits and NP = coNP, then there is no p-optimal propositional proof system. Our techniques here are very different, and our results seem unrelated to those in [15].

On the other hand, the results of Grochow and Pitassi [11] are crucial to our work. Grochow and Pitassi defined the Ideal Proof System, an algebraic proof system for which proofs are verifiable by polynomial size circuits (and in polynomial time under a standard derandomization assumption). They showed that super-polynomial IPS lower bounds for CNFs would imply VNP ≠ VP. This connection between proof complexity lower bounds and circuit complexity lower bounds has been further developed in [3, 7, 19]. However, none of these previous works establish an *equivalence* between proof complexity lower bounds and standard circuit lower bounds, as in Theorem Theorem 1.1.

We are also inspired by recent work of [4, 20]. Atserias and Muller [4] settle the long-standing open problem of whether Resolution is automatable (assuming P ≠ NP) by giving a reduction from SAT to proof complexity lower bounds for Resolution via variants of the proof complexity lower bound formulas. Their reduction relies on the hardness of the Pigeonhole Principle, which only holds for weak proof systems such as Resolution, while we are interested here in strong proof systems such as IPS. Pich and Santhanam [20] unconditionally establish a version of the Natural Proofs barrier [28] in proof complexity, but they do so for a proof system that is defined *non-constructively* and moreover for instances which are *randomly generated*. In contrast, we are interested here in the metamathematics of proof complexity for well-studied and concrete proof systems such as IPS and for *explicit* instances. We do use ideas from [4, 20] to give evidence for the Iterated Lower Bounds Hypothesis.

Finally, the idea of iterating proof complexity lower bounds is novel to the best of our knowledge. Nevertheless, as discussed above, considering the proof complexity of proof complexity lower bounds has been investigated to different degrees of explicitness in the literature. And we refer the reader to the recent excellent survey by Pudlak [25] that explores this and other related questions.

2 PRELIMINARIES

2.1 Basic Algebraic Complexity

For an excellent treatise on algebraic circuits and their complexity see Shpilka and Yehudayoff [30]. Let $\mathbb{G}$ be a ring. Denote by $\mathbb{G}[X]$ the ring of (commutative) polynomials with coefficients from $\mathbb{G}$ and variables $X := \{x_1, x_2, \dots\}$.

Algebraic circuits and formulas over the ring $\mathbb{G}$ compute polynomials in $\mathbb{G}[X]$ via addition and multiplication gates, starting from the input variables and constants from the ring. More precisely, an *algebraic circuit* C is a finite directed acyclic graph (DAG) with *input nodes* (i.e., nodes of in-degree zero) and a single *output node* (i.e., a node of out-degree zero). Input nodes are labeled with either a variable or a ring element in $\mathbb{G}$. All the other nodes have *fan-in* (that is, in-degree) *two* and are labeled by either an addition gate + or a product gate ×. Every node in an algebraic circuit C *computes* a polynomial as follows: an input node computes the variable or scalar that labels it. A + (or ×) gate is said to compute the addition (product, resp.) of the (commutative) polynomials computed by its incoming nodes. The polynomial computed by a node u in an algebraic circuit C is denoted $\widehat{u}$. Given a circuit C, we denote by $\widehat{C}$ the polynomial computed by C, that is, the polynomial computed by the output node of C. The ***size*** of a circuit C is the number of nodes in it, denoted $|C|$, and the *depth* of a circuit is the length of the longest directed path in it. For an algebraic circuit C we write $C(a/x)$ to denote the *substitution instance* of C in which every occurrence of the node x is replaced by the sub-circuit a; in case $C(x)$ is written with its displayed variable(s) x we can write $C(x)(a/x)$ for this substitution instance. We say that a polynomial is *homogeneous* whenever every monomial in it has the same (total) degree.

Definition 2.1 (Syntactic-degree sdeg(·))**.** *Let C be a circuit (without division) and v a node in C. The* syntactic-degree sdeg(v) *of v is defined as follows:*

(1) If v is a field element or a variable, then sdeg(v) := 0 *and* sdeg(v) := 1*, respectively;*
(2) If $v = u + w$ then sdeg(v) := max{sdeg(u), sdeg(w)}*;*
(3) If $v = u \cdot w$ then sdeg(v) := sdeg(u) + sdeg(w)*.*

An algebraic circuit is said to be *syntactic-homogeneous* if for every plus gate $u + v$, $\deg(u) = \deg(v)$.

Algebraic Complexity Classes. We now recall some basic notions from algebraic complexity (for more details see [30, Sec. 1.2]). Over a ring R, VP_R (for "Valiant's P") is the class of families $f = (f_n)_{n=1}^{\infty}$ of formal polynomials f_n such that f_n has $\mathrm{poly}(n)$ input variables, is of $\mathrm{poly}(n)$ degree, and can be computed by algebraic circuits over R of $\mathrm{poly}(n)$ size. VNP_R (for "Valiant's NP") is the class of families g of polynomials $(g_n)_{n=1}^{\infty}$ such that g_n has $\mathrm{poly}(n)$ input variables and is of $\mathrm{poly}(n)$ degree, and can be written as

$$g_n\left(x_1, \dots, x_{\mathrm{poly}(n)}\right) = \sum_{\overline{e} \in \{0,1\}^{\mathrm{poly}(n)}} f_n(\overline{e}, \overline{x})$$

for some family $(f_n)_{n=1}^{\infty} \in \mathsf{VP}_R$.

A polynomial $f(\overline{x})$ is a *projection* of a polynomial $g(\overline{y})$ if $f(\overline{x}) = g(L(\overline{x}))$ identically as polynomials in $\overline{x}$, for some map L that assigns to each y_i either a variable or a constant. In other words, a projection of $g(\overline{y})$ is a substitution instance of $g(\overline{y})$ in which $\overline{y}$ variables are substituted by $\overline{x}$ variables or field elements. A family of polynomials (f_n) is a polynomial projection or *p-projection* oft another family (g_n) if there is a function $t(n) = n^{\Theta(1)}$ such that f_n is a projection of $g_{t(n)}$ for all (sufficiently large) n. The *permanent* polynomial $\sum_{\sigma \in S_n} \prod_{i=1}^{n} x_{i,\sigma(i)}$ (for S_n the permutation group on n elements) is complete under p-projections for VNP when the ring R is a field of characteristic different from 2. The *determinant* polynomial on the other hand is known to be in VP but is not known to be complete for VP under p-projections.

Two central questions in algebraic complexity theory are whether the permanent is a p-projection of the determinant (a stronger variant speaks about quasi-polynomial projections); and whether VP equals VNP [33–35]. Since the permanent is complete for VNP (under p-projections), showing VP≠VNP amounts to proving that the permanent cannot be computed by polynomial-size algebraic circuits.

2.2 Algebraic Proof Systems

Grochow and Pitassi [11] suggested the following algebraic proof system which is essentially a Nullstellensatz proof system ([5]) written as an algebraic circuit. A proof in the Ideal Proof System is given as a *single* polynomial. We provide below the *boolean* version of IPS (which includes the boolean axioms), namely the version that establishes the unsatisfiability over 0-1 of a set of polynomial equations. In what follows we follow the notation in [7]:

Definition 2.2 ((boolean) Ideal Proof System (IPS), Grochow-Pitassi [11]). *Let $f_1(\overline{x}),\ldots,f_m(\overline{x}),p(\overline{x})$ be a collection of polynomials in $\mathbb{F}[x_1,\ldots,x_n]$ over the field $\mathbb{F}$. An **IPS proof of** $p(\overline{x})=0$ **from** $\{f_j(\overline{x})=0\}_{j=1}^m$, showing that $p(\overline{x})=0$ is semantically implied from the assumptions $\{f_j(\overline{x})=0\}_{j=1}^m$ over 0-1 assignments, is an algebraic circuit $C(\overline{x},\overline{y},\overline{z})\in\mathbb{F}[\overline{x},y_1,\ldots,y_m,z_1,\ldots,z_n]$ such that (the equalities in what follows stand for formal polynomial identities[3]):*

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

*The **size of the IPS proof** is the size of the circuit C. The variables $\overline{y},\overline{z}$ are called the* placeholder variables *since they are used as placeholders for the axioms. An IPS proof $C(\overline{x},\overline{y},\overline{z})$ of $1=0$ from $\{f_j(\overline{x})=0\}_{j\in[m]}$ is called an **IPS refutation** of $\{f_j(\overline{x})=0\}_{j\in[m]}$ (note that in this case it must hold that $\{f_j(\overline{x})=0\}_{j=1}^m$ have no common solutions in $\{0,1\}^n$).*

Notice that the definition above adds the equations $\{x_i^2-x_i=0\}_{i=1}^n$, called the set of ***boolean axioms*** denoted $\overline{x}^2-\overline{x}$, to the system $\{f_j(\overline{x})=0\}_{j=1}^m$. This allows to refute over $\{0,1\}^n$ unsatisfiable systems of equations. Also, note that the first equality in the definition of IPS means that the polynomial computed by C is in the ideal generated by $\overline{y},\overline{z}$, which in turn, following the second equality, means that C witnesses the fact that 1 is in the ideal generated by $f_1(\overline{x}),\ldots,f_m(\overline{x}),x_1^2-x_1,\ldots,x_n^2-x_n$ (the existence of this witness, for unsatisfiable set of polynomials, stems from the Nullstellensatz theorem [5]).

In order to use IPS as a propositional proof system (namely, a proof system for propositional tautologies), we need to fix the encoding of clauses as algebraic circuits.

Definition 2.3 (algebraic translation of CNF formulas). *Given a CNF formula in the variables $\overline{x}$, every clause $\bigvee_{i\in P}x_i\vee\bigvee_{j\in N}\neg x_j$ is translated into $\prod_{i\in P}(1-x_i)\cdot\prod_{j\in N}x_j=0$. (Note that these terms are written as algebraic circuits as displayed, where products are not multiplied out.)*

Notice that in this way a 0-1 assignment to a CNF is satisfying iff the assignment is satisfying all the equations in the algebraic translation of the CNF.

[3] That is, $C(\overline{x},\overline{0},\overline{0})$ computes the zero polynomial and $C(\overline{x},f_1(\overline{x}),\ldots,f_m(\overline{x}),x_1^2-x_1,\ldots,x_n^2-x_n)$ computes the polynomial $p(\overline{x})$.

Therefore, using Definition 2.3 to encode CNF formulas, boolean IPS is considered as a propositional proof system for the language of unsatisfiable CNF formulas, sometimes called *propositional IPS*. We say that an IPS proof is an ***algebraic IPS*** proof, if we do not use the boolean axioms $\overline{x}^2-\overline{x}$ in the proof. In our applications we are going to use algebraic IPS refutations, while sometimes explicitly adding the boolean axioms for some variables (while leaving them out for some other variables). *As a default when referring to IPS we mean the boolean* IPS *version. When we use algebraic IPS we will say that explicitly.*

The following is the main structural-complexity result for IPS. Notice that it already works for *algebraic* IPS and this will be important for us.

Theorem 2.1 (Grochow-Pitassi [11]). *For any ring R, a super-polynomial lower bound on algebraic* IPS *refutations (and hence also on* IPS *refutations) over R for any family of CNF formulas implies $\mathsf{VNP}_R\neq\mathsf{VP}_R$. The same result hold if we assume that the IPS refutation size lower bound holds* only infinitely often.

The following lemma is the key to the proof of the Theorem, and is used in our application:

Lemma 2.2. *Every family of unsatisfiable CNF formulas (φ_n) has a family of algebraic* IPS *(and hence also of* IPS*) certificates (C_n) in VNP_R.*

Proof of Theorem 2.1, assuming Lemma 2.2. For a given set $\mathcal{F}$ of unsatisfiable polynomial equations $F_1=\cdots=F_m=0$, a lower bound on algebraic IPS refutations of $\mathcal{F}$ is equivalent to giving the same circuit lower bound on *all* IPS certificates for $\mathcal{F}$. A super-polynomial lower bound on IPS implies that some function in VNP—namely, the VNP-IPS certificate guaranteed by Lemma 2.2—cannot be computed by polynomial-size algebraic circuits, and hence that VNP ≠ VP. □

2.2.1 Conventions and Notations for IPS Proofs. An IPS (algebraic or not) proof over a specific field or ring is sometimes denoted $\mathrm{IPS}_{\mathbb{F}}$ specifying explicitly it is over $\mathbb{F}$. For two algebraic circuits F,G, we define the *size of the circuit equation* $F=G$ to be the total circuit size of F and G, namely, $|F|+|G|$. For a set $\overline{\mathcal{F}}$ of equations between circuits we denote by $|\overline{\mathcal{F}}|$ to be the total size of the equations in the set.

Let $\overline{\mathcal{F}}$ denote a set of polynomial equations $\{f_i(\overline{x})=0\}_{i=1}^m$, and let $C(\overline{x},\overline{y},\overline{z})\in\mathbb{F}[\overline{x},\overline{y},\overline{z}]$ be an IPS proof of $f(\overline{x})$ from $\overline{\mathcal{F}}$ as in Definition 2.2. Then we write $C(\overline{x},\overline{\mathcal{F}},\overline{x}^2-\overline{x})$ to denote the circuit C in which y_i is substituted by $f_i(\overline{x})$ and z_i is substituted by the boolean axiom $x_i^2-x_i$. By a slight abuse of notation we also call $C(\overline{x},\overline{\mathcal{F}},\overline{x}^2-\overline{x})=f(\overline{x})$ an IPS proof of $f(\overline{x})$ from $\overline{\mathcal{F}}$ and $\overline{x}^2-\overline{x}$ (that is, displaying $C(\overline{x},\overline{y},\overline{z})$ *after* the substitution of the placeholder variables $\overline{y},\overline{z}$ by the axioms in $\overline{\mathcal{F}}$ and $\overline{x}^2-\overline{x}$, respectively).

The following fact shows that polynomial identities are proved for free in IPS:

Fact 2.3 ([3]). *If $F(\overline{x})$ is a circuit in the variables $\overline{x}$ over the field $\mathbb{F}$ that computes the zero polynomial, then there is an IPS proof of $F(\overline{x})=0$ of size $|F|$.*

For two polynomials $f(\overline{x}),g(\overline{x})$, *an IPS proof of $f(\overline{x})=g(\overline{x})$* from the assumptions $\overline{\mathcal{F}}$ is an IPS proof of $f(\overline{x})-g(\overline{x})=0$ (note that

in case $f(\overline{x})$ and $g(\overline{x})$ are identical as polynomials this is trivial to prove by Fact 2.3).

We denote by $C : \overline{\mathcal{F}} \vdash_{\text{IPS}}^{s} p = 0$ (resp. $C : \overline{\mathcal{F}} \vdash_{\text{IPS}}^{s} p = g$) the fact that $p = 0$ (resp. $p = g$) has an IPS proof $C(\overline{x}, \overline{y}, \overline{z})$ of size s from assumptions $\overline{\mathcal{F}}$, and we use $\vdash_{\text{IPS}}^{s,\, d}$ when we want to specify that the degree of the IPS proof is upper bounded by d. Assumptions $\overline{\mathcal{F}}$ can be written either as a set of equations or as a sequence of equations or sets thereof separated by commas. We may also suppress "= 0" and write simply $C : \overline{\mathcal{F}} \vdash_{\text{IPS}}^{s} p$ for $C : \overline{\mathcal{F}} \vdash_{\text{IPS}}^{s} p = 0$. Whenever we are only interested in claiming the existence of an IPS proof of size s of $p = 0$ from $\overline{\mathcal{F}}$ we suppress the C from the notation. Similarly, we can suppress the size parameter s from the notation. If F is a circuit computing a polynomial $\widehat{F} \in \mathbb{F}[\overline{x}]$, then we can talk about *an IPS proof C of F from assumptions $\overline{\mathcal{F}}$*, in symbols $C : \overline{\mathcal{F}} \vdash_{\text{IPS}} F$, meaning an IPS proof of $\widehat{F}$. Accordingly, for two circuits F, F' such that $\widehat{F} = \widehat{F'}$, we may speak about an *an IPS proof C of F from assumptions $\overline{\mathcal{F}}$* to refer to an IPS proof of F' from assumptions $\overline{\mathcal{F}}$. If $\{p_i = 0\}_{i=1}^{\infty}$ and $\{\overline{\mathcal{F}}_i\}_{i=0}^{\infty}$ are sequences of circuit equations and sets of circuit equations, respectively, then we write $\overline{\mathcal{F}}_i \vdash_{\text{IPS}}^{*} p_i = 0$ to denote that there is an IPS proof of $p_i = 0$ from the assumptions $\overline{\mathcal{F}}_i$ of size polynomial in $|\overline{\mathcal{F}}_i| + |p_i|$.

When we deal with *algebraic* IPS proofs we will use the same notation as above, only using IPS^{alg} instead of IPS.

3 PROOF COMPLEXITY CHARACTERIZATION OF VP ≠ VNP

We show that IPS cannot efficiently prove that the algebraic circuit class separation VP ≠ VNP is hard to prove in IPS. Note that this result is *unconditional*: IPS unconditionally does not have polynomial-size refutations of the statement asserting the existence of short IPS refutations for VP = VNP. On the other hand, if VP ≠ VNP is in fact easy to prove in IPS or if VP = VNP, then the result is less interesting, since then the result stems simply from soundness of IPS. Similarly, if the IPS refutations of VP = VNP are of exponential degree, then the result becomes trivially true again, since there *is* a polynomial-size refutation of our formulation of VP = VNP (since the formulation is exponential in itself). The more interesting scenario is under the reasonable assumption that indeed algebraic circuit class separations are hard to establish in polynomial-degree IPS, and in this case we show that this fact would not have efficient proofs in IPS.

3.1 Formalisations

3.1.1 CNF Encoding of Algebraic Circuit Equations. We are going to work at times with IPS that does *not* have the boolean axioms for all variables (but only for some variables that will be explicitly specified), namely algebraic IPS, denoted IPS^{alg} (for "IPS with no boolean axioms"). If we show that it is hard to refute that IPS without the boolean axioms has small refutations of some statements, then we also show that it is hard to refute that an IPS *with* the boolean axioms has small refutations of this statement (otherwise, a short refutation of the existence of IPS refutation with boolean axioms would imply a short refutation of the existence of a refutation in a weaker system). Hence in what follows even when we discuss CNF formulas translated to the algebraic setting we shall write precisely what the boolean axioms that we add to the formulas are.

For an algebraic circuit C and b a field element, we call $C = b$ a *circuit equation* (we sometimes use the same notion for equations between two circuits). We work over a *finite* field $\mathbb{F}_q$. This is necessary in our argument to be able to switch between formulas in CNF and algebraic circuit equations. Recall that a formula in CNF ("a CNF" for short) is a conjunction of clauses, where a clause is a disjunction of literals, and literals are variables or their negation. The algebraic translation of a formula in CNF is defined according to Definition 2.3. The size of objects like circuits, circuit equations, sets of circuit equations and formulas in CNF are denoted by $|\cdot|$ (where "$\cdot$" is replaced by the respective object).

Definition 3.1 (Algebraic extension axioms and unary bits)**.** *Given a circuit C and a gate g in C, we call the equation $x_g = \sum_{i=0}^{q-1} i \cdot x_{gi}$ the **algebraic extension axiom of** g, with the variable x_{g_i} being the ith **unary-bit of** g.*

Note that if the unary-bits of g are taken over $\{0, 1\}$ and assuming that $x_{gi} = 1$ for precisely one $0 \le i \le q-1$, then $x_g = i$ iff $x_{gi} = 1$. Note also that the unary-bits are disjoint from the (algebraic input) variables of the circuit C.

Definition 3.2 (Plain CNF encoding of algebraic circuits; $\text{cnf}(\cdot)$)**.** *Let $C(\overline{x}) = 0$ be a circuit equation in the variables $\overline{x}$. The **plain CNF encoding** of the circuit equation $C(\overline{x}) = 0$ denoted $\text{cnf}(C(\overline{x}) = 0)$ consists of the following CNFs in the unary-bits variables of all the gates in C:*

(1) If x_i is an input gate in C, the plain CNF encoding of C uses the variables $x_{x_i 0}, \ldots, x_{x_i (q-1)}$ that are the unary-bits of x_i, and consists of the clauses that express that precisely one unary-bit is 1 and all other unary-bits are 0: [4]

$$\bigvee_{j=0}^{q-1} x_{x_i j} \wedge \bigwedge_{j \neq \ell \in \{0,\ldots,1\}} (\neg x_{x_i j} \vee \neg x_{x_i \ell}). \tag{1}$$

(2) For every gate g in $C(\overline{x})$ and every satisfying assignment $\overline{\alpha}$ to the plain CNF encoding, the corresponding unary-bit x_{gi} evaluates to 1 iff the value of g is $i \in \{0, \ldots, q-1\}$ (when the algebraic inputs $\overline{x} \in \mathbb{F}_q^*$ to $C(\overline{x})$ take on the values corresponding to the boolean assignment $\overline{\alpha}$; "$*$" here means the Kleene star). This is ensured with the following equations: if $g = u \circ v$ is an internal gate in C (including the output gate, but excluding the input gates), for $\circ \in \{+, \times\}$, we have a CNF φ_g in the unary-bits variables of g, u, v that is satisfied by an assignment precisely when the output unary-bits of g get their correct values based on the (constant-size) truth table of $\circ$ over $\mathbb{F}_q$ and the input unary-bits of u, v (we ensure that if more than one unary-bit is assigned 1 in any of the unary-bits of g, u, v then the CNF is unsatisfiable).

(3) For the output gate g_{out} of C we add the equations: $x_{g_{out}0} = 1$ and $x_{g_{out}\, i} = 0$ for all $i = 1, \ldots, q-1$, which express that $g_{out} = 0$.

(4) For every unary-bit variable x_{gi} we have the boolean axiom (recall we write these boolean axioms explicitly since we are going to work with IPS^{alg}): $x_{gi}^2 - x_{gi} = 0$.

Definition 3.3 (Extended CNF encoding of a circuit equation; $\text{ecnf}(\cdot)$)**.** *Let $C(\overline{x})$ be a circuit in the $\overline{x}$ variables over the finite field $\mathbb{F}_q$. Then the **extended CNF encoding** of the circuit equation $C(\overline{x}) = 0$,*

[4] This conditions is needed only for inputs. For internal gates the CNFs expressing the truth table for the gate will make sure that only one output unary-bit is one.

in symbols $\mathrm{ecnf}(C(\overline{x}) = 0)$*, is defined to be a set of algebraic equations over* $\mathbb{F}_q$ *in the variables* x_g *and* $x_{g0}, \ldots, x_{gq-1}$ *which are the unary-bits variables corresponding to node* g *in* C*, that consists of:*

(1) the plain CNF encoding of $C(\overline{x}) = 0$*; and*
(2) the algebraic extension axiom of g*, for every gate* g *in* C*.*

Notice that the extended CNF encoding is not formally a CNF since it uses the algebraic extension axioms which are not clauses. Also, note that the extended CNF encoding does not contain the boolean axioms for the algebraic extension variables x_g, for g a gate in C, as this variable is meant to range over $\mathbb{F}_q$.

Since we work with extension variables for each gate in a given circuit equation $C(\overline{x}) = 0$, it is more convenient to express circuit equations as a *set* of equations that correspond to the *straight line program* (**SLP**) of $C(\overline{x})$ (which is an equivalent in strength formulation to algebraic circuits). In a SLP we have a sequence of equations between variable such that the extension variable for the output gate computes the value of the circuit assuming all equations hold as follows: we choose any topological order $g_1, g_2, \ldots, g_i, \ldots, g_{|C|}$ on the gates of the circuit C (that is, if g_j has a directed path to g_k in C then $j < k$) and define the following equations: $g_i = g_j \circ g_k$ for $\circ \in \{+, \times\}$ iff g_i is a $\circ$ gate in C with two incoming edges from g_j and g_k. And SLP representation of a circuit equation $C(\overline{x}) = 0$ means that we add to the SLP above the equation $g_{|C|} = 0$, where $g_{|C|}$ is the output gate of the circuit.

Using the concept of extended CNF encoding we can now show how to efficiently go in IPS from a circuit equation written as a set of equations for the corresponding SLP to a CNF, and vice versa. The idea is to augment the SLP of $C(\overline{x}) = 0$ with $x_g = \sum_{i=0}^{q-1} i \cdot x_{gi}$ which is the algebraic extension axiom of g, for every gate g in C. We show that, efficiently in IPS, we can go from this representation of $C(\overline{x}) = 0$ to its extended CNF encoding, and vice versa.

Proposition 3.1 (Translating between extended CNFs and circuit equations)**.** *Let* $\mathbb{F}$ *be a finite field, and let* $C(\overline{x})$ *be a circuit in the* $\overline{x}$ *variables over* $\mathbb{F}$ *that is* written as a set of equations *corresponding to the SLP of* $C(\overline{x})$*. Then, the following both hold (recall that* $\vdash^*_{\mathrm{IPS^{alg}}}$ *means polynomial-size proofs):*

$$\mathrm{ecnf}(C(\overline{x}) = 0) \vdash^*_{\mathrm{IPS^{alg}}} C(\overline{x}) = 0 \tag{2}$$

$$\left\{x_g = \sum_{i=0}^{q-1} i \cdot x_{gi} : g \text{ a node in } C\right\}, C(\overline{x}) = 0 \vdash^*_{\mathrm{IPS^{alg}}} \mathrm{ecnf}(C(\overline{x}) = 0). \tag{3}$$

The proof uses the fact that over finite fields the truth table of each algebraic gate is easy to describe. It is omitted due to lack of space.

We wish to speak about CNF formulas and not extended CNF formulas. This is necessary since the work of [11], showing that an IPS lower bound implies VNP ≠ VP, is known to hold only for lower bounds against CNF formulas.

We have the following simple proposition:

Proposition 3.2. *Let* $C(\overline{x}) = 0$ *be a circuit equation over* $\mathbb{F}_q$*. Then,* $\mathrm{cnf}(C(\overline{x}) = 0)$ *is an unsatisfiable CNF iff* $\mathrm{ecnf}(C(\overline{x}) = 0)$ *is an unsatisfiable set of equations over* $\mathbb{F}_q$ *iff* $C(\overline{x}) = 0$ *is unsatisfiable over* $\mathbb{F}_q$*.*

Since by [11] every unsatisfiable CNF has an $\mathrm{IPS^{alg}}$ refutation computable in VNP, by the proposition above we get:

Corollary 3.3. *If* $\mathrm{ecnf}(C(\overline{x}) = 0)$ *is unsatisfiable then it has an* $\mathrm{IPS^{alg}}$ *refutation in* VNP*.*

3.1.2 Encoding Universal Circuits. To express in the theory that a circuit computes a certain polynomial we will use the concept of a universal circuit as introduced by Raz [26]. A universal circuit is an algebraic circuit that loosely speaking embeds all possible circuits of a certain size. More precisely, a universal circuit for the class of polynomials in $\mathbb{F}[\overline{x}]$ that have algebraic circuits of size at most t is a circuit $U(\overline{x}, \overline{w})$ with two sets of variables $\overline{x}$ and $\overline{w}$, such that $f(\overline{x}) \in \mathbb{F}[\overline{x}]$ has circuit of size at most t iff there is a fixed choice of values $\overline{\alpha}$ to $\overline{w}$ for which $U(\overline{x}, \overline{\alpha}) = f(\overline{x})$ (as a polynomial identity). Intuitively one can think of the $\overline{w}$ variables as the *circuit variables*, while the $\overline{x}$ variables are the algebraic variables of the circuit that is encoded by the $\overline{w}$ variables. Formally, $\overline{w}$ describe the edge labels put on edges of a universal circuit.

Raz [26] showed the existence of small algebraic universal circuits for homogeneous polynomials (see also an intuitive description in [30]):

Theorem 3.4 (Existence of universal circuits for homogeneous polynomials; Raz [26]). *Let* $\mathbb{F}$ *be a field and* $\overline{x}$ *be* n *variables, and let* $C^{\mathrm{hom}}_{t,d}$ *denote the class of all homogeneous polynomials of total degree exactly* d *in* $\mathbb{F}[\overline{x}]$ *that have algebraic circuits of size at most* t*. Then there is a circuit* $U(\overline{x}, \overline{w}) \in \mathbb{F}[\overline{x}, \overline{w}]$ *of size* $O(dt^4)$ *and syntactic-degree* d *such that* $\overline{w}$ *are* t *variables which are disjoint from* $\overline{x}$*, that is* universal *for* $C^{\mathrm{hom}}_{t,d}$ *in the following sense:* $f(\overline{x}) \in C^{\mathrm{hom}}_{t,d}$ *iff there exists* $\overline{\alpha} \in \mathbb{F}^t$ *such that* $U(\overline{x}, \overline{\alpha}) = f(\overline{x})$*.*

The idea behind the proof of Theorem 3.4 is to provide a normal form for circuits: every syntactic homogeneous circuit of degree d is reduced with a small increase in size to a normal form in which different choices of edge labels determine the polynomial the circuit computes.

Since we do not work necessarily with homogeneous circuits and polynomials we will assume that the universal circuit $U(\overline{x}, \overline{w})$ is in fact universal for *general* (non-homogeneous) circuits of a given degree d and size t. We can assume this by defining a universal circuit as a sum of the universal circuits for each homogeneous degree as follows (this does not constitute a restriction when considering polynomial degrees, since the classical result of Strassen [32] shows that every algebraic circuit computing a degree at most d polynomial can be written as a sum of homogeneous circuits for each of the homogeneous components of the polynomial; where in addition each homogeneous circuit has size polynomially bounded in the original circuit size).

Definition 3.4. *The* ***universal circuit for degree*** d ***and size*** t ***circuits*** *is defined as:*

$$U(\overline{x}, \overline{w}) = \sum_{i=0}^{d} U_i(\overline{x}, \overline{w}), \tag{4}$$

where $U_i(\overline{x}, \overline{w})$ *is the universal circuit for homogeneous* $\overline{x}$*-polynomials of* i *degree* $C^{\mathrm{hom}}_{t,i}$ *and where the* $\overline{w}$*-variables in each distinct* $U_i(\overline{x}, \overline{w})$ *are pairwise disjoint (namely, no variable* w_l *appears in both* $U_i(\overline{x}, \overline{w})$ *and* $U_j(\overline{x}, \overline{w})$ *for* $i \neq j$*).*

By Theorem 3.4 the size of $U(\overline{x}, \overline{w})$ is $\sum_{i=0}^{d} O(it^4) = O(d^2t^4)$.

In our formalisation we are going to express that a circuit computes a polynomial in $\mathbb{F}[\overline{x}]$ by stating that the coefficient vector of a universal circuit described by edge variables $\overline{w}$ equals some

fixed vector in $\mathbb{F}^N$, with N being the total number of possible $\overline{x}$-monomials in a degree at most d polynomial with a given number of variables, usually n. Hence, we represent a circuit of size s using s variables $\overline{w}$ for the s edges in the circuit $U(\overline{x},\overline{w})$. Note that $U(\overline{x},\overline{w})$ is a circuit in both $\overline{x},\overline{w}$, *while our formalisation will* not *use in the end the* $\overline{x}$*-variables*: each $\overline{x}$-monomial will be represented as a polynomial in the $\overline{w}$-variables only. In other words, the coefficient vector of the polynomial in the $\overline{x}$ variables computed by $U(\overline{x},\overline{w})$ is a vector of N polynomials in the $\overline{w}$ variables, where N is the total number of $\overline{x}$-monomials.

We need to show how to compute given $U(\overline{x},\overline{w})$ the coefficient of an $\overline{x}$-monomial M as a polynomial in the edge variables $\overline{w}$. Such a polynomial is denoted $\mathrm{Coeff}_M(U(\overline{x},\overline{w}))$.

Let $f(\overline{x},\overline{w}) \in \mathbb{F}[\overline{x},\overline{w}]$ be a polynomial, and let $M = \prod_{i\in I} x_i^{\alpha_i} \cdot \prod_{j\in J} w_j^{\beta_j}$ be a monomial in $f(\overline{x},\overline{w})$, for some $\alpha_i,\beta_i \in \mathbb{N}$ (where $0 \in \mathbb{N}$). Then, we call $\sum_{i\in I}\alpha_i$ the $\overline{x}$*-degree* of M.

Definition 3.5 ($\mathrm{Coeff}_M(\cdot)$)**.** *Let $f(\overline{x},\overline{w})$ be a polynomial in $\mathbb{F}[\overline{x},\overline{w}]$ in the pairwise disjoint sets of variables $\overline{x},\overline{w}$. Let M be an $\overline{x}$-monomial of degree j. Then,* $\mathrm{Coeff}_M(f(\overline{x},\overline{w}))$ *is the (polynomial) coefficient in $\mathbb{F}[\overline{w}]$ (that is, in the $\overline{w}$-variables only) of M in $f(\overline{x},\overline{w})$.*[5]

Note that $f(\overline{x},\overline{w}) = \sum_{M_i} M_i \cdot \mathrm{Coeff}_{M_i}(f(\overline{x},\overline{w}))$, where the M_i's are all possible $\overline{x}$-monomials of degree at most d, for d the maximal $\overline{x}$-degree of a monomial in $f(\overline{x},\overline{w})$.

Proposition 3.5 (Computation of coefficients)**.** *Let $f(\overline{x},\overline{w}) \in \mathbb{F}[\overline{x},\overline{w}]$ be a polynomial in $\mathbb{F}[\overline{x},\overline{w}]$ in the pairwise disjoint sets of variables $\overline{x},\overline{w}$. Suppose that M is an $\overline{x}$-monomial of degree d, and assume that there is an algebraic circuit computing $f(\overline{x},\overline{w})$ of size s and syntactic-degree ℓ. Then, there is a circuit of size $O(7^d \cdot s)$ computing* $\mathrm{Coeff}_M(f(\overline{x},\overline{w}))$ *of syntactic-degree $\ell^{O(1)}$.*

Proof. For each variable x_i in M we construct circuits that computes the polynomials g,h, respectively, such that $x_i \cdot g + h = f$, with h having no occurrences of x_i. Each such circuit-construction increases the size by a constant factor of 7 according to the claim below. Hence, after d iterations for each of the d variables in M we get a $O(7^d \cdot s)$-size circuit D computing the polynomial coefficient of M in $f(\overline{x},\overline{w})$. However, this polynomial coefficient may contain also monomials in *both* the $\overline{x}$ and $\overline{w}$ variables, and *to eliminate these monomials we simply assign zeroes to all* $\overline{x}$*-variables in* D.

It remains to prove the following claim (recall that $\widehat{C}$ is the polynomial computed by a circuit C and that $|C|$ is the size of C; we denote by $\widehat{C}(\overline{x}) \restriction_{x_i=0}$ the polynomial $\widehat{C}(\overline{x})$ where x_i is assigned 0).

Claim 3.6. *Let $C(\overline{x})$ be a circuit of syntactic-degree ℓ in the $\overline{x}$ variables over the field $\mathbb{F}$. Then, for every variable x_i there is a circuit of size $7|C|$ and syntactic-degree $\ell^{O(1)}$ that computes the polynomial $g(\overline{x})$, such that $\widehat{C}(\overline{x}) = x_i \cdot g(\overline{x}) + \widehat{C}(\overline{x}) \restriction_{x_i=0}$.*

The proof is by induction on the circuit size. Due to lack of space we omit the details. □

Remark 3.7. *It will be important for us that there are* small *universal circuits, namely that the size of $U(\overline{x},\overline{w})$ is small, since our conjectural hard candidates will use a circuit computing $U(\overline{x},\overline{w})$. The diagonalisation argument works regardless of the size of the instance, only that if the size of the formulas proved is too big, and specifically exponential in the number of variables, we land in the uninteresting case in which there will be no short refutations of the statement Φ expressing short IPS refutations of* VNP = VP*, simply because Φ is satisfiable, namely there are polynomial-size (in the unsatisfiable input CNF) refutations of* VP = VNP*. We explain this further in subsection 3.2.*

[5] Note that there can be polynomial coefficients in $\mathbb{F}[\overline{x},\overline{w}]$ of M that involve also the $\overline{x}$-variables. But we wish to consider the polynomial coefficients in the $\overline{w}$-variables alone. For this reason, in Proposition 3.5 we shall assign zero values to the $\overline{x}$-variables after taking the polynomial coefficient of M in the $\overline{w}$-variables.

3.1.3 Formalising VNP≠VP. The class VP is expressed using a universal circuit while the class VNP is expressed explicitly as the coefficient vector of the permanent polynomial, where permanent is known to be complete for VNP [33].

Let $\mathrm{perm}(\overline{x})$ be the permanent polynomial on the variables $\overline{x}$. We are to encode the negation of VP ≠ VNP (since we work with the refutation system IPS, we prove statements by refuting their negations):

Definition 3.6 (Formalisation of VP = VNP)**.** *The formalisation of* VNP = VP(t,n,d) *denoted* "VNP = VP(t,n,d)"*, expressing that there is a universal circuit for degree $d \geq n$ and size t circuits that computes the permanent polynomial of dimension n (with $\overline{x}$ being the n^2 variables of the permanent), is the following set of polynomial equations:*

$$\left\{\mathrm{Coeff}_{M_i}(U(\overline{x},\overline{w})) = b_i \ : \ 1 \leq i \leq N\right\}, \tag{5}$$

where $\overline{b} = \mathrm{coeffs}(\mathrm{perm}(\overline{x})) \in \mathbb{F}^N$ is the coefficient vector of the permanent polynomial of dimension n, $\overline{w}$ are the t edge variables, $\{M_i\}_{i=1}^N$ is the set of all possible $\overline{x}$-monomials of degree at most d, and $N = \sum_{j=0}^{d}\binom{n^2+j-1}{j} = 2^{O(n^2+d)}$ is the number of monomials of total degree at most d over n^2 variables.

The size of each circuit equation in Equation 5 is $O(7^j \cdot |U(\overline{x},\overline{w})|) = O(7^n \cdot d^2t^4)$ (with $j \leq d$ the degree of the $\overline{x}$-monomial M_i), meaning that the *size of* "VNP = VP(t,n,d)" is

$$O(7^n \cdot d^2t^4 \cdot N) = t^4 \cdot 2^{O(n^2+d)}, \tag{6}$$

and the syntactic-degree of each circuit equation in Equation 5 is $d^{O(1)}$.

3.1.4 Formalising IPS Refutations. To express an IPS lower bound as a set of polynomial equations we will formalise the negation of this statement, namely the existence of a small IPS refutation for a specific CNF formula.

Definition 3.7 (IPS refutation predicate $\mathrm{IPS}_{ref}(t,d,\overline{\mathcal{F}})$)**.** *Let $\overline{\mathcal{F}}$ be a CNF formula with m clauses and (for simplicity, since we deal with matrix inputs) n^2 variables $\overline{x}$ written as a set of polynomial equations according to Definition 2.3. Let $U(\overline{x},\overline{y},\overline{w})$ be a universal circuit for degree d and size t circuits in the $\overline{x}$ variables and the m placeholder variables $\overline{y}$ (*both *of these sets of variables are the algebraic variables in a polynomial computed by the universal circuit when assigned field values to the edge labels $\overline{w}$), and the t edge label variables $\overline{w}$. We formalise the existence of a size t and degree d circuit that computes the* IPS *refutation of $\overline{\mathcal{F}}$, denoted* $\mathrm{IPS}_{ref}(t,d,\overline{\mathcal{F}})$*, as follows:*

$$\mathrm{Coeff}_{M_i \restriction \overline{y}=\mathbf{0}}(U(\overline{x},\mathbf{0},\overline{w})) = 0, \tag{7}$$

$$\mathrm{Coeff}_{M_i \restriction \overline{y}=\overline{\mathcal{F}}}\left(U(\overline{x},\overline{\mathcal{F}},\overline{w})\right) = \begin{cases} 1, & M_i \restriction \overline{y} = \overline{\mathcal{F}} \\ 0, & otherwise, \end{cases} \quad \begin{array}{l} \text{is the} \\ \text{constant 1 monomial} \end{array} \tag{8}$$

where i ranges over $i \in [N]$ so that $\{M_i\}_{i=1}^{N}$ are the set of all possible $\overline{x} \cup \overline{y}$-monomials (monomials in both $\overline{x}, \overline{y}$ variables) of degree at most d, and $N = \sum_{j=0}^{d} \binom{n^2+m+j-1}{j} = 2^{O(n^2+m+d)}$ is the number of monomials of total degree at most d over $n^2 + m$ variables, $\mathbf{0}$ is the all-zero vector of length m and $M_i \restriction \overline{y} = \overline{a}$ denotes the monomial M_i in which the $\overline{y}$-variables are substituted according to the assignment $\overline{a} : \overline{y} \to \mathbb{F}[\overline{x}]$ of circuits in the $\overline{x}$-variables to the $\overline{y}$-variables (so that $M_i \restriction \overline{y} = \overline{\mathcal{F}}$ stands for $M_i \restriction \overline{y} = \overline{a}$, where $\overline{a}$ is the assignment of the ith clause in $\overline{\mathcal{F}}$ to y_i).

The size of each circuit equation in Equation 7 and Equation 8 is $O(7^j \cdot |U(\overline{x}, \overline{y}, \overline{w})| \cdot |\overline{\mathcal{F}}|) = O(7^d \cdot d^2 t^4 \cdot |\overline{\mathcal{F}}|)$ (with $j \leq d$ the degree of the $\overline{x}$-monomial M_i), meaning that the *size of* $\mathrm{IPS}_{ref}(t, d, \overline{\mathcal{F}})$ is

$$O(7^d \cdot d^2 t^4 \cdot |\overline{\mathcal{F}}| \cdot N) = t^4 \cdot |\overline{\mathcal{F}}| \cdot 2^{O(n^2+m+d)}, \quad (9)$$

and the syntactic-degree of each circuit equation in Equation 7 and Equation 8 is $d^{O(1)}$.

Encoding an IPS *lower bound* is simply providing an ostensible *unsatisfiable* set of polynomials that express the existence of size at most t and degree at most d IPS refutation of a CNF $\overline{\mathcal{F}}$. Thus refuting this set of polynomials amounts to proving an IPS lower bound.

Remark 3.8. *Note that our results are interesting only for IPS refutations of* sub-exponential degrees *d. If the degree d is exponential in the number of variables $n^2 + m$ in the IPS polynomial, then the size r of the IPS proof predicate is itself exponential in $n^2 + m$ and hence will have a polynomial in r sized IPS refutation in itself (hence, although the diagonalisation argument still holds in this range of parameters, it holds merely because even when $\mathrm{IPS}_{ref}(t, d, \overline{\mathcal{F}})$ is indeed unsatisfiable it has a polynomial-size IPS refutation since the number of variables is polynomial in its size).*

3.2 Characterizing VNP ≠ VP as a Proof Complexity Lower Bound

We are now ready to prove our main theorem that IPS cannot prove certain IPS lower bounds assuming VNP ≠ VP. The gist of the argument is showing how from a short IPS proof of an IPS lower bound on an unsatisfiable CNF formulas one gets a short IPS proof that VP ≠ VNP—this can be considered a formalisation in IPS of the Grochow and Pitassi argument [11]. Since we work with a *refutation* system, we will show equivalently that if IPS efficiently refutes the existence of small IPS refutations of some CNF formula, then there is an efficient IPS refutation of VP = VNP. We shall start from the CNF "VP = VNP" and reach a contradiction (in IPS). The idea now is this: if we have the CNF "VNP = VP(t, n, d)", meaning that perm$(\overline{x})$ of dimension n is computable by size t circuits (of syntactic-degree d), then we know in particular, by the completeness of the permanent for VNP and the fact that every unsatisfiable CNF has an IPS refutation computable in VNP (Theorem 2.1), that there is a "projection", namely, an assignment $\overline{a}$ of variables and field elements to $\overline{x}$, such that perm$(\overline{x} \restriction \overline{a})$ is the IPS refutation of the CNF "VP = VNP" of some lower dimension. Since the class of size t circuits is closed under assignments we conclude that perm$(\overline{x} \restriction \overline{a})$ has a size t circuit, that is, there is a small IPS refutation of "VP = VNP" of some lower dimension. But we assumed that IPS can (efficiently) *refute* the existence of such small IPS refutations for "VP = VNP", and we finish by soundness of IPS.

We are going to use the following lemma (proof omitted due to lack of space) that will allow us to express a proof complexity lower bound on a seemingly stronger ("less satisfiable") statement (and hence, stating such a refutation lower bound *weakens* the lower bound statement). In particular we will extend the statement of VP = VNP with additional extension axioms for certain new circuits $C_i(\overline{x})$. These extension axioms include the equations for gates in $C_i(\overline{x})$ written as SLPs and the algebraic extension axioms for each gate g in $C_i(\overline{x})$. On the other hand, these extension axioms *do not* express that any of the $C_i(\overline{x})$ equals any specific value. In particular this means that these additional extension axioms do not add any information to the statement (that is, they do not actually make the statement "less satisfiable"). In return we will be able to state that such additional extension axioms do not make a refutation lower bound against the statement stronger, in the sense that if we can efficiently refute the stronger (extended) statement then we can efficiently refute also the weaker (non extended) statement:

Lemma 3.9 (Disjoint extension axioms do not make refutations easier, provably in IPS)**.** *Let $\overline{F}$ be a set of circuit equations written as SLPs of total size s, $\overline{E}$ be a set of circuits (*not *circuit equations) each written as a SLP with extension variables (not appearing in $\overline{F}$) for every gate g in the circuits in $\overline{E}$, together with the algebraic extension axioms for each such gate g. Then,*

$$\mathsf{cnf}(\mathrm{IPS}^{\mathrm{alg}}_{ref}(t, \mathsf{ecnf}(\overline{F} \cup \overline{E}))) \vdash^{s^{O(1)}}_{\mathrm{IPS}} \mathsf{cnf}(\mathrm{IPS}^{\mathrm{alg}}_{ref}(t, \mathsf{ecnf}(\overline{F}))) . \quad (10)$$

Lemma 3.10 (Algebraic extension axioms do not make refutations easier, provably in IPS)**.** *For every set of circuit equations $\overline{F}$ written as SLPs of total size s the following holds:*

$$\mathsf{cnf}(\mathrm{IPS}^{\mathrm{alg}}_{ref}(t, \mathsf{ecnf}(\overline{F}))) \vdash^{s^{O(1)}}_{\mathrm{IPS}} \mathsf{cnf}(\mathrm{IPS}^{\mathrm{alg}}_{ref}(t, \mathsf{cnf}(\overline{F}))) . \quad (11)$$

For simplicity, we shall work with *polynomial degrees* IPS refutations, though our results hold true for higher degrees as well. On the other hand, when the degrees of the IPS refutations involved (namely, when d in $\lambda = \mathrm{IPS}_{ref}(t, d, \overline{\mathcal{F}})$ is exponential in the number of variables in λ) we arrive at an uninteresting scenario: assuming λ is unsatisfiable there is a polynomial-size in $|\lambda|$ refutation of λ simply because $|\lambda|$ is exponential in its number of variables (see also Remark 3.8). Similarly, when the degree of the IPS refutations is bigger than their size t we land again in the uninteresting case for similar reasons (see explanation of this in section 3.2). Therefore, in what follows we let $d : \mathbb{N} \to \mathbb{N}$ be a (monotone) *size function* $d(t) = t^{\epsilon}$ for some constant $0 < \epsilon < 1$. The function d stands for the syntactic-degree of the universal circuit used in either the IPS refutations or the universal circuits for computing the permanent.

Let $\varphi^{\mathrm{cnf}}_{r,m,d}$ denote the CNF encoding of the circuit equation "VNP = VP$(r, m, d(r))$" over the field $\mathbb{F}_q$ expressing that there are size r and syntactic-degree $d(r)$ circuits for the permanent of dimension m over $\mathbb{F}_q$. Let $\Phi_{t,r,m,d}$ denote the CNF formula $\mathsf{cnf}(\mathrm{IPS}_{ref}(t, d, \varphi^{\mathrm{cnf}}_{r,m,d}))$ expressing that IPS refutes $\varphi^{\mathrm{cnf}}_{r,m,d}$ in size t and degree $d(t)$ over $\mathbb{F}_q$.

Theorem 3.11 (main). *VP ≠ VNP over $\mathbb{F}_q$ iff the CNF family $\{\Phi_{t,r,m,d}\}$ does not have polynomial-size* IPS *refutations infinitely often in the following sense: for every sufficiently large constant c_0*

there exists a constant c_1 such that for infinitely many[6] $t, r, m \in \mathbb{N}$ *and every sufficiently large constant c_2, if $t > |\varphi^{\text{cnf}}_{r,m,d}|^{c_1}$ and $m^{c_1} < r < m^{c_2}$, $\Phi_{t,r,m,d}$ has no* IPS *refutation of size at most $|\Phi_{t,r,m,d}|^{c_0}$.*

Note that for *every* proof-complexity lower bound n^{c_0} that we want to establish, there *exists* a hard instance stating a proof complexity lower bound of n^{c_1} (where n is the size of the formulas refuted or proved).

Proof. ($\Leftarrow$) This is the easier direction. Assume that there exist sufficiently big constants c_0, c_1, c_2 such that for infinitely many $m \in \mathbb{N}$ the family of CNF formulas $\{\text{cnf}(\text{IPS}_{ref}(s, d, \varphi_{t,m,d}))\}_{n=1}^{\infty}$ does not have polynomial-size IPS refutations where $t < m^{c_0}$ (that is, the purported circuit for the permanent of dimension m is polynomially bounded in m) and $s < |\varphi_{t,m,d}|^{c_1}$ (that is, the purported size of an IPS refutation of the formula $\varphi_{t,m,d}$ is polynomially bounded in the size of the formula).

If $\{\text{cnf}(\text{IPS}_{ref}(s, \varphi_{t,m,d}))\}_{n=1}^{\infty}$ is an *unsatisfiable* CNF family for infinitely many $m \in \mathbb{N}$, then by Theorem 2.1 [11] VNP $\neq$ VP over $\mathbb{F}_q$ (note that the result of [11] holds also for lower bounds on IPS *without* the boolean axioms).

Otherwise, $\{\text{cnf}(\text{IPS}_{ref}(s, d, \varphi_{t,m,d}))\}_{n=1}^{\infty}$ is a satisfiable CNF family for infinitely many $m \in \mathbb{N}$. Thus, by Proposition 3.2 the set of polynomial equations $\text{IPS}_{ref}(s, d, \varphi_{t,m,d})$ is satisfiable over $\mathbb{F}_q$ for infinitely many $m \in \mathbb{N}$ and hence there is an IPS refutation of $\varphi_{t,m,d}$ for infinitely many $m \in \mathbb{N}$ and by soundness of IPS we get that infinitely often there is no size at most m^{c_0} for the permanent of dimension m, and we are done.

($\Rightarrow$) We work over $\mathbb{F}_q$. First, consider the plain CNF encoding $\varphi^{\text{cnf}}_{r,m,d} = \text{cnf}(\text{“VNP} = \text{VP}(r, m, d(r))\text{”})$ and the extended CNF encoding $\varphi^{\text{ecnf}}_{r,m,d} = \text{ecnf}(\text{“VNP} = \text{VP}(r, m, d(r))\text{”})$, and denote by $\varphi^{\star}_{r,m,d}$ the extended CNF $\varphi^{\text{ecnf}}_{r,m,d}$ together with additional extension axioms $\overline{E}$ for some new set of circuits written as SLPs and a new set of algebraic extension axioms for all the gates in these new circuits that we will specify later (all the variables in these new equations will be disjoint from the original variables in $\varphi^{\text{ecnf}}_{r,m,d}$).

We assume that:

Assumption 1: VP $\neq$ VNP over $\mathbb{F}_q$. Namely, there is no constant c such that for all but constant many $n \in \mathbb{N}$, the permanent polynomial $\text{perm}(\overline{x})$ of dimension n has an algebraic circuit of size at most n^c, over $\mathbb{F}_q$.

And we will prove that:

Conclusion 1: Let c_0 be some sufficiently large constant. There exists a constant c_1 such that for every sufficiently big constant $c_2 > c_1$, for infinitely many $t, r, m \in \mathbb{N}$ with $t > |\varphi^{\star}_{r,m,d}|^{c_1}$ and $m^{c_1} < r < m^{c_2}$ the following CNF

$$\text{cnf}(\text{IPS}^{\text{alg}}_{ref}(t, d, \varphi^{\star}_{r,m,d})) \tag{12}$$

has no IPS refutations of size at most $|\text{cnf}(\text{IPS}^{\text{alg}}_{ref}(t, d, \varphi^{\star}_{r,m,d}))|^{c_0}$.

We argue that as long as the set of new axioms $\overline{E}$ in $\varphi^{\star}_{r,m,d}$ have total algebraic circuit-size polynomial in $|\varphi^{\text{cnf}}_{r,m,d}|$, to conclude the theorem Conclusion 1 above suffices:

[6] Here we mean infinitely many distinct t's, infinitely many distinct r's and infinitely many distinct m's.

Claim 3.12. *If Conclusion 1 above holds then also the same conclusion holds when the CNF*

$$\text{cnf}(\text{IPS}_{ref}(t, d, \varphi^{\text{cnf}}_{r,m,d})) \tag{13}$$

replaces Equation 12 in Conclusion 1.

Scheme of the main argument. Under Assumption 1 above, let c_3 be some fixed constant (that is derived from Lemma 3.14), let c_0 be any constant. Suppose that $t, r, m \in \mathbb{N}$ are such that (recall that "$\vdash^{s,\,d}_{\text{IPS}}$" means size s and degree d IPS proofs):

$$\underbrace{\text{cnf}(\text{IPS}^{\text{alg}}_{ref}(t, d, \overbrace{\varphi^{\text{ecnf}}_{r,m,d}}^{\sigma}))}_{\lambda} \vdash^{|\lambda|^{c_0},\ d(|\lambda|)}_{\text{IPS}} 1 = 0, \tag{14}$$

where $\varphi^{\text{ecnf}}_{r,m,d}$ is unsatisfiable over $\mathbb{F}_q$ by Assumption 1 (for sufficiently big r, m and when $r < m^{c_2}$ for a sufficiently big constant c_2 that we pick in what follows), and moreover $\lambda \vdash^{|\lambda|^{c_0},\ d(|\lambda|)}_{\text{IPS}} 1 = 0$ is by an assumption that there are i.o. short degree $d(|\lambda|)$ refutations for the statement λ that expresses there are short refutations of $\varphi^{\text{ecnf}}_{r,m,d}$.

Note that if this latter assumption is *incorrect* then we finish our proof, since then there exists a constant c_1 such that for a sufficiently big constant $c_2 > c_1$, and where $t > |\varphi^{\text{ecnf}}_{r,m,d}|^{c_1}$ and $m^{c_1} < r < m^{c_2}$, $\text{cnf}(\text{IPS}^{\text{alg}}_{ref}(t, d, \varphi^{\text{ecnf}}_{r,m,d}))$ has no degree $d(|\lambda|)$ IPS refutations of size at most $|\text{cnf}(\text{IPS}_{ref}(t, d, \varphi^{\text{ecnf}}_{r,m,d}))|^{c_0}$ (and assuming c_0 is big enough, there is no IPS refutation of size at most $|\text{cnf}(\text{IPS}_{ref}(t, d, \varphi^{\star}_{r,m,d}))|^{c_0}$ of $\text{cnf}(\text{IPS}_{ref}(t, d, \varphi^{\star}_{r,m,d}))$ as well).

Let

$$\gamma = \varphi^{\star}_{t,k,d} \text{ and } k = 6(\ell + N'),$$

where ℓ, N' are the number of variables and clauses in $\varphi_{r,m,d}$, respectively, and assume that Equation 14 holds and

$$t > k^{c_1}$$

for a constant c_1 that we pick big enough so that

$$|\gamma|^{c_3} + |\lambda|^{c_0} = |\gamma|^{c_3} + |\gamma|^{c' \cdot c_0} < |\gamma|^{c_1} \tag{15}$$

(we use here that $|\lambda| = |\gamma|^{c'}$ for some constant c', by Equation 6 and Equation 9); and moreover, as before, $\varphi^{\star}_{t,k,d}$ is $\varphi^{\text{ecnf}}_{t,k,d}$ augmented with additional extension axioms $\overline{E}$ for some new set of circuits written as SLPs and a new set of algebraic extension axioms for all the gates in these new circuits that we will specify later (all the variables in these new equations will be disjoint from the original variables in $\varphi^{\text{ecnf}}_{t,k,d}$).

In Lemma 3.14 in the sequel we construct an IPS^{alg} derivation of $\text{IPS}^{\text{alg}}_{ref}(t, d, \varphi^{\text{ecnf}}_{r,m,d})$ from $\varphi^{\star}_{t,k,d}$ of size at most $|\gamma|^{c_3}$ and degree at most $d(|\lambda|)$, assuming that $\varphi^{\text{ecnf}}_{r,m,d}$ is indeed unsatisfiable (which, as mentioned above, follows from our Assumption 1 and the fact that r, m are big enough and $r < m^{c_2}$ for a sufficiently big constant $c_2 > c_1$), from which we can conclude, using Equation 14, that:

$$\overbrace{\varphi^{\star}_{t,k,d}}^{\gamma} \vdash^{|\gamma|^{c_3},\ d(|\gamma|)}_{\text{IPS}^{\text{alg}}} \underbrace{\text{cnf}(\text{IPS}^{\text{alg}}_{ref}(t, d, \overbrace{\varphi^{\text{ecnf}}_{r,m,d}}^{\sigma}))}_{\lambda} \vdash^{|\lambda|^{c_0},\ d(|\lambda|)}_{\text{IPS}} 1 = 0 \tag{16}$$

(note that for a CNF T, if we have $T\left|\frac{s}{\text{IPS}}1 = 0\right.$ we have also $T\left|\frac{s}{\text{IPS}^{\text{alg}}}1 = 0\right.$, because all variables in the CNF T have boolean axioms explicitly added to T by Definition 3.2, hence we have $\varphi^{\star}_{t,k,d}\left|\frac{|\gamma|^{c_3},\, d(|\gamma|)}{\text{IPS}^{\text{alg}}}\right. \text{cnf}(\text{IPS}^{\text{alg}}_{ref}(t,d,\varphi^{\text{ecnf}}_{r,m,d}))\left|\frac{|\lambda|^{c_0},\, d(|\lambda|)}{\text{IPS}^{\text{alg}}}\right. 1 = 0$).

By Equation 15, Equation 16 means that there exists a w sized and degree $d(|\gamma|)$ IPS^{alg} refutation of $\varphi^{\star}_{t,k,d}$ whenever $w \geq |\gamma|^{c_1}$, namely

$\text{IPS}^{\text{alg}}_{ref}(w,\varphi^{\star}_{t,k})$ is satisfiable over $\mathbb{F}_q$. Thus, by Proposition 3.2 $\text{cnf}(\text{IPS}^{\text{alg}}_{ref}(w,d,\varphi^{\star}_{t,k,d}))$ is a satisfiable CNF formula, and so by soundness of IPS: $\text{cnf}(\text{IPS}^{\text{alg}}_{ref}(w,d,\overbrace{\varphi^{\star}_{t,k,d}}^{\gamma}))\not\left|\frac{}{\text{IPS}}\right. 1 = 0$.

In other words, we have concluded that there are no IPS refutations of $\text{cnf}(\text{IPS}^{\text{alg}}_{ref}(w,d,\varphi^{\star}_{t,k,d}))$, and in particular no such IPS refutations of size at most $|\text{cnf}(\text{IPS}^{\text{alg}}_{ref}(w,d,\varphi^{\star}_{t,k,d}))|^{c_0}$ and degree at most $d(w)$ exist, whenever $w \geq |\gamma|^{c_1}$ and $t > k^{c_1}$, concluding the theorem.

Remark 3.13. *(1) Our lower bound in Conclusion Equation 12 and Equation 13 holds for all $t > k^{c_1}$. Note that as $t > k^{c_1}$ becomes bigger refuting $\text{IPS}_{ref}(t,d,\varphi^{\text{cnf}}_{r,m,d})$ becomes even* harder*, hence if there are no* IPS *refutations of $\text{IPS}_{ref}(t,d,\varphi^{\text{cnf}}_{r,m,d})$ of size $|\lambda|^{c_0}$ then there are no such refutations also for bigger t's.*

(2) Our lower bound in Conclusion Equation 12 and Equation 13 holds for all $m^{c_1} < r < m^{c_2}$. Note that as $r > m^{c_1}$ becomes bigger it becomes harder *to refute $\varphi_{r,m,d}$ (as this would establish a* stronger *lower bound). Hence, the lower bound against $\text{IPS}_{ref}(t,d,\varphi^{\text{cnf}}_{r,m,d})$ becomes* easier *to prove. On the other hand, once $r > m^{c_1}$ becomes too big, namely exceeds m^{c_2}, $\varphi_{r,m,d}$ may become* satisfiable*, hence, refuting $\text{IPS}_{ref}(t,d,\varphi^{\text{cnf}}_{r,m,d})$ becomes easier: since $\varphi^{\text{cnf}}_{r,m,d}$ has a satisfying assignment, we can use this assignment to refute the existence of a refutation of $\varphi^{\text{cnf}}_{r,m,d}$.*

It remains to construct an IPS^{alg} derivation of $\text{IPS}^{\text{alg}}_{ref}(t,d,\varphi^{\text{ecnf}}_{r,m,d})$ from $\varphi^{\star}_{t,k,d}$ of size at most $|\varphi^{\star}_{t,k,d}|^{c_3}$:

Lemma 3.14. *There is a constant c_3 such that if $\varphi_{r,m,d}$ is unsatisfiable, then under the above notation and parameters:*

$$\overbrace{\varphi^{\star}_{t,k,d}}^{\gamma}\left|\frac{|\gamma|^{c_3},\, d(|\gamma|)}{\text{IPS}^{\text{alg}}}\right. \text{cnf}(\text{IPS}^{\text{alg}}_{ref}(t,\overbrace{\varphi^{\text{ecnf}}_{r,m,d}}^{\sigma})).$$

Due to lack of space we omit the full proof, but we note that the proof is based on the following (contrapositive) formalization of the Grochow-Pitassi implication from proof complexity lower bounds to circuit lower bounds:

Proposition 3.15 (Grochow-Pitassi formalization in IPS^{alg})**.** *There is an IPS^{alg} derivation from $\varphi^{\star}_{t,k,d}$, denoted γ, of $\text{IPS}^{\text{alg}}_{ref}(t,d,\varphi^{\text{ecnf}}_{r,m,d})$, with size at most $|\gamma|^{b}$, for some constant b.*

This concludes the proof of Theorem 3.11. □

The interesting range of parameters. Let us now discuss different ranges of parameters and what they mean for Theorem 3.11. The degree of IPS refutations we work with should not be exponential in the number of variables in formulas, as otherwise, although the argument is correct, it is trivial since there is always a short refutation. For that reason we deal with only polynomial degree IPS refutations (though one can also deal with sub-exponential degrees).

Similarly, it is important for us that the size of our formulas is small enough. The reason is once more that otherwise we get to the following uninteresting case: the size of the formula $\text{cnf}(\text{IPS}_{ref}\left(t,d,\varphi^{\text{cnf}}_{r,m,d}\right))$ is exponential in the number of its variables, hence it must be easy to refute. Thus, if it is satisfiable it has no IPS refutation by soundness of IPS, and if it is unsatisfiable it has an IPS refutation of size polynomial in its size; this bring us to Part 2 in the dichotomy Theorem 4.2: the hard candidates alternate between being *satisfiable* and hence having no short refutations by soundness, and being *unsatisfiable* but having a short refutation trivially.

More precisely, we distinguish between two types of exponential growth functions: good and bad, as follows. By subsubsection 3.1.3, the size of "VNP = VP(t,n,d)" is $t^4 \cdot 2^{O(n^2+d)}$ with t the number of variables. Whenever $t \geq (n^2+d)^c$, for some constant $c > 1$, the term $t^4 \cdot 2^{O(n^2+d)}$ is considered a *good exponential* because it is exponentially smaller than 2^t, and the latter term is considered a *bad exponential* (recall that the degree $d \leq t^{\epsilon}$ for some $0 < \epsilon < 1$, hence $t = (n^2+d)^c$ is a valid equality).

Corollary 3.16. *Theorem 3.11 is interesting whenever $t \geq (n^2 + d(t))^c$ (and when we assume that $d(t) \leq 2^{\epsilon}$ for some $0 < \epsilon < 1$), in the sense that in this range of parameters the hard candidate $\text{cnf}(\text{IPS}_{ref}(t,d,\varphi^{\text{cnf}}_{r,m,d}))$ is conjectured to be both unsatisfiable and for which no polynomial-size IPS refutations exist.*

We also have the following immediate corollary:

Corollary 3.17. *Theorem 3.11 holds for any proof system that simulates IPS.*

4 CANDIDATE HARD FORMULAS FOR EVERY PROPOSITIONAL PROOF SYSTEM

4.1 Iterated Lower Bound Formulas

We use *pps* to refer to a propositional proof system.

Definition 4.1. *Given pps R, propositional formula ϕ and size function $s : \mathbb{N} \to \mathbb{N}$, $\text{lb}_R(\phi,s)$ is a propositional DNF formula of size $\text{poly}(|\phi| + s(|\phi|))$ over $\text{poly}(|\phi| + s(|\phi|))$ variables expressing that there is no R-proof of ϕ having size $s(|\phi|)$.*

More explicitly, the formula $\text{lb}_R(\phi,s)$ contains s variables $y_1,\ldots,y_s$ encoding R-proofs of length s and $\text{poly}(|\phi| + s)$ auxiliary variables encoding the computation of the relation R, to verify that $y_1,\ldots,y_s$ does not constitute an R-proof of ϕ.

Definition 4.2 (Reasonably Strong Proof System)**.** *We say that a pps R is reasonably strong if it satisfies the following conditions:*

(1) R p-simulates Res *(Resolution).*
(2) R is closed under partial assignments, i.e., if there are polynomial-size R-proofs of ϕ and $\overline{a}$ is a partial truth assignment to the variables of ϕ, then there are polynomial-size R-proofs of $\phi(\overline{a})$.
(3) R has poly-size proofs of its own reflection principle $\text{lb}_R(\phi,s) \vee \phi$.
(4) R is closed under modus ponents, i.e., if there are polynomial-size R-proofs of $\tau \vee \neg\phi$ and of $\phi \vee \psi$ for DNFs τ,ϕ,ψ, then there are polynomial-size R-proofs of $\tau \vee \psi$.

The conditions above are satisfied for any standard strong enough pps, eg., Frege and Extended Frege.

We now formally define iterated lower bound formulas relative to a pps R.

Definition 4.3 (Iterated Lower Bound Formulas)**.** *Given pps R, propositional formula ϕ and size function $s : \mathbb{N} \to \mathbb{N}$, the sequence $\{\mathrm{lb}_R^k(\phi, s)\}, k = 0 \ldots \infty$ is defined inductively as follows:*

(1) $\mathrm{lb}_R^0(\phi, s) = \phi$

(2) $\mathrm{lb}_R^{k+1}(\phi, s) = \mathrm{lb}_R(\mathrm{lb}_R^k(\phi, s), s)$.

Lemma 4.1. *Let R be any reasonably strong pps, and let τ be a non-tautology. Then there are R-proofs of $\mathrm{lb}_R(\tau, s)$ of size poly$(|\mathrm{lb}_R(\tau, s)|)$.*

Proof. Since τ is a non-tautology, there is an assignment $\overline{a}$ to the variables of τ such that $\tau(\overline{a})$ evaluates to false. Since R is reasonably strong, it simulates Res and hence can prove $\neg\tau(\overline{a})$ by substituting the assignment $\overline{a}$ into τ. Since R is reasonably strong, R can prove its own reflection principle efficiently, and since R is closed under partial assignments, it can also prove $\mathrm{lb}_R(\tau, s) \vee \tau(\overline{a})$. It follows from the closure under modus ponens that R can prove $\mathrm{lb}_R(\phi, s)$ efficiently. □

We show a dichotomy for iterated lower bound formulas: either they are all hard (and hence all tautologies), or they divide up according to the parity of k, with one parity corresponding to non-tautologies and the other parity corresponding to tautologies with short proofs.

Theorem 4.2 (Dichotomy for Iterated Lower Bound Formulas). *Let R be a reasonably strong pps and $s : \mathbb{N} \to \mathbb{N}$. There is a constant c such that for every s with $s(n) > n^c$ for all $n \in \mathbb{N}$ and for every ϕ that does not have R-proofs of size $s(|\phi|)$, exactly one of the following holds:*

(1) For every integer $k \geq 0$, $\phi_k = \mathrm{lb}_R^k(\phi, s)$ is a tautology that does not have R-proofs of size $s(|\phi_k|)$

(2) There is an integer $k \geq 0$ such that for every integer $i \geq 0$, ϕ_{k+i} is not a tautology (and hence does not have R-proofs of any size) if i is odd, and ϕ_{k+i} is a tautology with R-proofs of size at most poly$(s(|\phi_{k+i}|))$ if i is even.

Proof. Suppose that for every integer $k \geq 0$, ϕ_k is a tautology. We show that this implies that ϕ_k is also hard in the sense that it does not have R-proofs of size $s(|\phi_k|)$. Indeed, since ϕ_j is a tautology for every j, it follows that ϕ_{k+1} is a tautology. Since ϕ_{k+1} asserts that ϕ_k does not have R-proofs of size $s(|\phi_k|)$, it follows that ϕ_k is indeed hard as claimed. Thus, in this case, the first item in the statement of the theorem holds.

Otherwise, since $\phi_0 = \phi$ is a tautology with no R-proofs of size $s(|\phi|)$, there is a least positive integer k such that ϕ_k is a tautology but ϕ_{k+1} is a non-tautology. Since ϕ_{k+1} asserts that ϕ_k does not have R-proofs of size $s(|\phi_k|)$, it follows that ϕ_k does indeed have R-proofs of size $s(|\phi_k|)$. We will show by induction in this case that for each integer $i \geq 0$ the following statement $S(i)$ holds: ϕ_{k+2i+1} is not a tautology (and hence does not have R-proofs of any size), and ϕ_{k+2i} is a tautology with R-proofs of size at most $s(|\phi_{k+2i}|)$.

We first establish the base case $S(0)$. When $i = 0$, by assumption on ϕ_k, we have that $\phi_{k+2i} = \phi_k$ is a tautology with R-proofs of size at most $s(|\phi_k|)$. Also, by assumption on k, ϕ_{k+1} is a non-tautology, and since R is sound, it follows that ϕ_{k+1} does not have R-proofs of any size.

For the inductive step, we assume that $S(i)$ has been shown and deduce $S(i+1)$. Since $S(i)$ is true, we have that ϕ_{k+2i} is a tautology with R-proofs of size at most $s(|\phi_{k+2i}|)$ and ϕ_{k+2i+1} is not a tautology (and hence does not have R-proofs of any size). We have that ϕ_{k+2i+2} asserts that ϕ_{k+2i+1} does not have R-proofs of size $s(|\phi_{k+2i+1}|)$, which is tautologous since ϕ_{k+2i+1} does not have R-proofs of any size. In order to show that ϕ_{k+2i+2} has R-proofs of size at most $s(|\phi_{k+2i+2}|)$, we simply apply Lemma 4.1 with $\tau = \phi_{k+2i+1}$, where c is a constant such that $\mathrm{lb}_R(\tau, s)$ has R-proofs of size at most $|\mathrm{lb}_R(\tau, s)|^c$. Since $s(n) > n^c$ for all $n \in \mathbb{N}$, we have that $\mathrm{lb}_R(\tau, s)$ has R-proofs of size at most $s(|\mathrm{lb}_R(\tau, s)|)$. It follows that ϕ_{k+2i+3} is not a tautology, since ϕ_{k+2i+3} asserts that ϕ_{k+2i+2} does not have R-proofs of size at most $s(|\phi_{k+2i+2}|)$, which is false by the previous line, and we are done. □

We call the first item of Theorem 4.2 the *useful* case of the dichotomy, as this is the case that gives us hard tautologies for R.

Iterated Lower Bound Hypothesis: Let R be a reasonably strong pps that is not polynomially bounded. Then there is a super-polynomial function $s : \mathbb{N} \to \mathbb{N}$ and a formula ϕ with no R-proofs of size $s(|\phi|)$ such that for all non-negative integers k, $\phi_k = \mathrm{lb}_R^k(\phi, s)$ is a tautology that does not have R-proofs of size $s(|\phi_k|)$.

We observe that the Iterated Lower Bounds Hypothesis fails if *optimal proof systems* exist. The existence of optimal proof systems is not widely believed; indeed, under the assumption, there is a complexity collapse.

Proposition 4.3. *Suppose there exists an optimal proof system R that is reasonably strong. Then for any time-constructible super-polynomial function $s : \mathbb{N} \to \mathbb{N}$ and any formula ϕ, there is a positive integer k such that $\phi_k = \mathrm{lb}_R^k(\phi, s)$ is not a tautology.*

Proof. Suppose there exists an optimal proof system R that is reasonably strong. Assume, for the sake of contradiction, that there is a formula ϕ and a time-constructible super-polynomial function s such that $\phi_k = \mathrm{lb}_R^k(\phi, s)$ is a tautology for every positive integer k. Define the pps R' which is R plus the formulas ϕ_k added as axioms. Thus R' has polynomial-size (indeed size zero) proofs of ϕ_k for each k. R' is indeed a pps: it is complete since R is a pps, it is sound since the formulas ϕ_k are tautologies by assumption, and it is polynomially verifiable since R is polynomially verifiable and the sequence ϕ_k is polynomial-time computable by time-constructibility of s.

Since R is optimal, we have that R polynomially simulates R'. Since R' has constant-size proofs of each ϕ_k, this means that there are R-proofs of ϕ_k of size at most $|\phi_k|^c$ for each k, where c is a constant. Choose k large enough so that $s(|\phi_k|) > |\phi_k|^c$. SInce ϕ_{k+1} is a tautology, we have that ϕ_k does not have proofs of size $s(|\phi_k|)$ and hence does not have proofs of size $|\phi_k|^c$, which is a contradiction. □

Recently, it was shown that Resolution is NP-hard to automate, by using a reduction based on proof complexity lower bound formulas [4, 9].

Theorem 4.4 (Atserias-Muller [4], Garlik [9]). *Let $s : \mathbb{N} \to \mathbb{N}$ be any super-polynomial function such that $s(n) = 2^{n^{o(1)}}$. For any*

formula ϕ, if ϕ is a tautology, then $\tau = \mathsf{lb}_{\mathsf{Res}}(\phi)$ does not have Res*-proofs of size $s(|\tau|)$.*

It follows by induction that the Iterated Lower Bounds Hypothesis holds at least for the relatively weak proof system Resolution.

Corollary 4.5. *Let $s : \mathbb{N} \to \mathbb{N}$ be any super-polynomial function such that $s(n) = 2^{n^{o(1)}}$ and let ϕ be a tautology with no Resolution proofs of size $s(|\phi|)$ (eg., the Pigeonhole Principle). Then for all non-negative integers k, $\phi_k = \mathsf{lb}^k_{\mathsf{Res}}(\phi, s)$ is a tautology that does not have R-proofs of size $s(|\phi_k|)$.*

4.2 Iteration Preserves Hardness for Random Truth Table Formulas

Definition 4.4 (Truth Table Formulas)**.** *Given a boolean function f_n on n variables and a size parameter t, $\mathsf{ttable}(f, t)$ is a propositional DNF formula of size $N = \tilde{O}(2^n s^3)$ over $\tilde{O}(s)$ variables expressing that f does not have boolean circuits of size s. The distribution $\mathsf{Randtt}(n, t)$ over $\mathsf{ttable}(f, t)$, where f is chosen uniformly at random from all boolean functions on n variables, is called the distribution of random truth table formulas.*

Conjecture 4.6 (Rudich's Conjecture [29])**.** *There is a constant ℓ for which there is no sequence of polynomial-size non-deterministic circuits $\{C_m\}$ such that for infinitely many m for which $m = 2^n$ for some non-negative integer n:*

(1) If C_m accepts a string y, then y is the truth table of a boolean function f_n that does not have boolean circuits of size n^ℓ.
(2) C_m accepts at least an inverse polynomial fraction of all inputs.

Definition 4.5 (Distributional Iterated Lower Bound Formulas)**.** *Let D_N be a distribution on propositional formulas of size N. Given pps R and size function $s : \mathbb{N} \to \mathbb{N}$, the sequence of distributions $\{\mathsf{lb}^k_R(D_N, s)\}, k = 0 \ldots \infty$ is defined inductively as follows:*

(1) $\mathsf{lb}^0_R(D_N, s) = D_N$;
(2) $\mathsf{lb}^{k+1}_R(\phi, s)$ is the distribution on formulas $\mathsf{lb}_R(\phi, s)$ where ϕ is sampled from $\mathsf{lb}_R(\mathsf{lb}^k_R(D_n, s), s)$.

Theorem 4.7. *If Rudich's Conjecture holds, then there exist a pps R efficiently simulating Extended Frege and a constant $\ell > 0$ such that for every large enough $c > 0$ and every non-negative integer k, $\mathsf{lb}^k_R(D_N, N^c)$ is a tautology with no R-proofs of size $|lb^k_R(D_N, N^c)|^c$ with probability $1 - o(1)$ for all large enough N, where $D_N = \mathsf{Randtt}(n, n^\ell$ (for N an appropriate function of n and ℓ as given by Definition 4.4).*

Proof. The proof is by induction on k. Let $\ell > 0$ be the constant in Conjecture 4.6 and let R be a pps and c be a large enough constant to be specified later.

We will establish the case $k = 0$ for *every* pps R, under Rudich's Conjecture. We have that $\mathsf{lb}^k_R(D_N, N^c) = D_N$. Since a random boolean function on n variables has circuits of size n^ℓ with exponentially small probability, a formula τ sampled according to D_N is a tautology with high probability. Moreover, Rudich's Conjecture implies that τ does not have R-proofs (or indeed proofs in *any* propositional proof system) of size N^c with high probability. To see this, note that R with size bound s defines a non-deterministic algorithm A_R running in time $\mathrm{poly}(s)$ for the problem $\mathsf{MCSP}[n^\ell]$ asking whether a given truth table y of a boolean function f on n bits has circuits of size n^ℓ: A_R checks if $\mathsf{ttable}(f, n^\ell)$ has R-proofs of size s. A_R only accepts on hard boolean functions, by the soundness of R, satisfying the first item of Conjecture 4.6. If A_R accepted τ for even a $1/N$ fraction of truth-table tautologies τ, this would contradict the second item of Conjecture 4.6.

We define R to be the pps that is Extended Frege together with axioms stating that $\mathsf{ttable} f^{\mathsf{SAT}}_n, 2n^\ell$ holds, where f^{SAT}_n is the truth table of SAT on n variables. The axioms indeed hold under Rudich's Conjecture, as Rudich's Conjecture implies that NP does not have polynomial-size circuits. It is easy to verify that R is reasonably strong according to Definition 4.2.

Now suppose we have established the assertion for all non-negative integers smaller than k and would like to establish it for k. The inductive strategy builds partly on ideas in [20]. We have by the inductive assumption that with probability $1 - o(1)$ for ψ sampled from D_N, $\mathsf{lb}^{k-1}_R(\psi, N^c)$ does not have R-proofs of size $|\mathsf{lb}^{k-1}_R(\psi, N^c)|^c$. It follows immediately that with probability $1 - o(1)$, for ψ sampled from D_N, $\mathsf{lb}^k_R(\psi, N^c)$ is a tautology.

For the lower bound on $\mathsf{lb}^k_R(D_N, N^c)$, we will treat k differently depending on its parity. If k is even, we will show inductively that if $\psi_k = \mathsf{lb}^k_R(\psi, N^c)$ is a tautology, then so is ψ. Indeed, this is trivially true when $k = 0$, and we show that if ψ_k is a tautology, then so is ψ_{k-2}. Assume contrapositively that ψ_{k-2} is not a tautology. This means that ψ_{k-1} is a tautology with R-proofs of size $|\psi_{k-1}|^c$, since R is reasonably strong, using Lemma 4.1, where c is greater than the exponent of the polynomial in Lemma 4.1. But this implies ψ_k is not a tautology, contradicting our assumption on ψ.

Now we use the assumption of Rudich's Conjecture to complete the inductive step. It remains to prove that with probability $1 - o(1)$, for ψ sampled from D_N, $\mathsf{lb}^k_R(\psi, N^c)$ does not have R-proofs of size $|\mathsf{lb}^k_R(\psi, N^c)|^c$. Suppose, for the sake of contradiction that with probability $\Omega(1)$, for infinitely many N, for ψ sampled from D_N, $\mathsf{lb}^k_R(\psi, N^c)$ has R-proofs of the desired size. We use this to define a non-deterministic polynomial-time algorithm that accepts a constant fraction of truth tables of hard boolean functions on n bits and does not accept any easy boolean functions on n bits, for infinitely many n, in contradiction to Rudich's Conjecture. Given a truth table of a boolean function f_n, the algorithm checks if there is a R-proof of $\psi_k = \mathsf{lb}^k_R(\psi, N^c)$ of size at most $|\psi_k|^c$, where $\psi = \mathsf{ttable}(f_n, n^\ell)$, and accepts if yes. If the algorithm does accept on ψ_k, then by the soundness of R, ψ_k is a tautology, and since k is even, so is ψ by the inductive argument in the previous para. Hence f is indeed a hard boolean function, as desired. Moreover, for f_n chosen uniformly at random, the algorithm accepts with probability $\Omega(1)$ by assumption, for infinitely many n, contradicting Rudich's Conjecture.

If k is odd, we need a slightly more involved argument, which generalizes the argument used to show Lemma 2 in [20]. Since Rudich's Conjecture holds, it follows that there are *succinct hitting sets* against polynomial-size non-deterministic circuits, i.e., a sequence $\{H_m\}$ of sets of strings in $\{0,1\}^m$, where each H_m is a truth table of a boolean function on $\log(m)$ inputs with circuits of size $\log(m)^\ell$ (we assume without loss of generality that m is a power of 2), such that for every sequence $\{C_m\}$ of non-deterministic circuits that accept a $\Omega(1)$ fraction of their inputs, at least one element of H_m is accepted by C_m for all large enough m. Also, since Rudich's Conjecture holds, it follows that SAT does not have polynomial-size

circuits. Using a straightforward argument, the sequence $\{H'_m\}$ of sets of strings in $\{0,1\}^m$, where each H'_m is $f^{\mathsf{SAT}}_{\log(m)} \oplus y$ for $y \in H_m$, is also a succinct hitting set sequence against polynomial-size non-deterministic circuits, but in this case the sets consist of truth tables of *hard* boolean functions on $\log(m)$ inputs. We have that $\mathrm{ttable}(z, n^\ell)$ is a tautology for $z \in H'_m$ when $m = 2^n$ is large enough. Moreover, by the same argument as in the proof of Lemma 2 in [20], for each $z \in H'_m$, $\mathrm{ttable}(z, n^\ell)$ has R-proofs of size at most $|\mathrm{ttable}(z, n^\ell)|^c$, using the fact that R p-simulates Extended Frege and has $\mathrm{ttable}(f^{\mathsf{SAT}}_n, 2n^\ell)$ as an axiom.

These facts can be used to show by the inductive argument in the proof of Theorem 4.2 that the second item of Theorem 4.2 holds for the formulas $\psi_k = \mathrm{lb}^k_R(\psi, N^c)$ where $\psi = \mathrm{ttable}(z, n^\ell)$ for $z \in H'_m$: they are tautologies with short R-proofs when k is even, and non-tautologies (and hence without any R-proofs at all) when k is odd. In the current case of our inductive step, k is odd, and hence every such formula ψ_k is a non-tautology, and hence does not have short proofs. Now suppose for the sake of contradiction, that with probability $\Omega(1)$ over uniformly chosen boolean function f_n on n variables, for infinitely many n, $\mathrm{lb}^k_R(\psi, N^c)$ has short R-proofs, where $\psi = \mathrm{ttable}(f_n, n^\ell)$. This means that the polynomial-time non-deterministic algorithm that on input f_n, checks if there is a short R-proof of $\mathrm{lb}^k_R(\mathrm{ttable}(f_n, n^\ell), N^c)$ accepts a constant fraction of functions f_n. However, none of the functions z for $z \in H'_m$ is accepted. This contradicts the assumption that $\{H'_m\}$ is a hitting set sequence for all m, and hence also contradicts Rudich's Conjecture. □

ACKNOWLEDGEMENTS

We wish to thank Jan Pich for very helpful discussions during the work on this paper. We also thank anonymous reviewers of this work for very useful comments that improved the exposition.

Part of this work was done while the second author was on a sabbatical visit to Oxford University. Both authors were supported in part by ERC Consolidator Grant Agreement No. 615075.

REFERENCES

[1] Miklós Ajtai. 1988. The complexity of the pigeonhole principle. In *Proceedings of the IEEE 29th Annual Symposium on Foundations of Computer Science*. 346–355.
[2] Michael Alekhnovich, Eli Ben-Sasson, Alexander A. Razborov, and Avi Wigderson. 2004. Pseudorandom generators in propositional proof complexity. *SIAM J. Comput.* 34, 1 (2004), 67–88 (electronic).
[3] Yaroslav Alekseev, Dima Grigoriev, Edward A. Hirsch, and Iddo Tzameret. 2020. Semi-algebraic proofs, IPS lower bounds, and the τ-conjecture: can a natural number be negative?. In *52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020*. 54–67.
[4] Albert Atserias and Moritz Müller. 2020. Automating Resolution is NP-Hard. *J. ACM* 67, 5 (2020), 31:1–31:17. https://doi.org/10.1145/3409472
[5] Paul Beame, Russell Impagliazzo, Jan Krajíček, Toniann Pitassi, and Pavel Pudlák. 1996. Lower bounds on Hilbert's Nullstellensatz and propositional proofs. *Proc. London Math. Soc. (3)* 73, 1 (1996), 1–26. https://doi.org/10.1112/plms/s3-73.1.1
[6] Stephen A. Cook and Robert A. Reckhow. 1979. The Relative Efficiency of Propositional Proof Systems. *J. Symb. Log.* 44, 1 (1979), 36–50. https://doi.org/10.2307/2273702
[7] Michael A. Forbes, Amir Shpilka, Iddo Tzameret, and Avi Wigderson. 2016. Proof Complexity Lower Bounds from Algebraic Circuit Complexity. In *31st Conference on Computational Complexity, CCC 2016, May 29 to June 1, 2016, Tokyo, Japan.* 32:1–32:17. https://doi.org/10.4230/LIPIcs.CCC.2016.32
[8] Harvey Friedman. 1979. On the consistency, completeness and correctness problems. Unpublished.
[9] Michal Garlík. 2019. Resolution Lower Bounds for Refutation Statements. In *44th Int. Symp. Math. Found. CS MFCS 2019 (LIPIcs, Vol. 138)*. 37:1–37:13.
[10] Mika Göös, Pritish Kamath, Robert Robere, and Dmitry Sokolov. 2019. Adventures in Monotone Complexity and TFNP. In *10th Innovations in Theoretical Computer Science Conference, ITCS 2019, January 10-12, 2019, San Diego, California, USA.* 38:1–38:19. https://doi.org/10.4230/LIPIcs.ITCS.2019.38
[11] Joshua A. Grochow and Toniann Pitassi. 2018. Circuit Complexity, Proof Complexity, and Polynomial Identity Testing: The Ideal Proof System. *J. ACM* 65, 6 (2018), 37:1–37:59. https://doi.org/10.1145/3230742
[12] Armin Haken. 1985. The intractability of resolution. *Theoret. Comput. Sci.* 39, 2-3 (1985), 297–308.
[13] Jan Krajíček. 1997. Interpolation theorems, lower bounds for proof systems, and independence results for bounded arithmetic. *The Journal of Symbolic Logic* 62, 2 (1997), 457–486.
[14] Jan Krajíček. 2001. Tautologies from pseudo-random generators. *Bull. Symbolic Logic* 7, 2 (2001), 197–212.
[15] Jan Krajicek. 2004. Diagonalization in proof complexity. *Fundamenta Mathematicae* 182 (2004), 181–192.
[16] Jan Krajicek. 2004. Dual weak pigeonhole principles, pseudo-surjective functions, and provability of circuit lower bounds. *Journal of Symbolic Logic* 69, 1 (2004), 265–286.
[17] Jan Krajicek. 2004. Implicit Proofs. *Journal of Symbolic Logic* 69, 2 (2004), 387–397.
[18] Jan Krajíček, Pavel Pudlák, and Alan Woods. 1995. An exponential lower bound to the size of bounded depth Frege proofs of the pigeonhole principle. *Random Structures Algorithms* 7, 1 (1995), 15–39.
[19] Fu Li, Iddo Tzameret, and Zhengyu Wang. 2018. Characterizing Propositional Proofs as Noncommutative Formulas. In *SIAM Journal on Computing*, Vol. 47. 1424–1462.
[20] Jan Pich and Rahul Santhanam. 2019. Why are proof complexity lower bounds hard?. In *60th Annual IEEE Symposium on Foundations of Computer Science FOCS 2019, November 9-12, 2019, Baltimore, Maryland USA*.
[21] Toniann Pitassi, Paul Beame, and Russell Impagliazzo. 1993. Exponential lower bounds for the pigeonhole principle. *Comput. Complexity* 3, 2 (1993), 97–140.
[22] Tonnian Pitassi and Iddo Tzameret. 2016. Algebraic Proof Complexity: Progress, Frontiers and Challenges. *ACM SIGLOG News* 3, 3 (2016).
[23] Pavel Pudlak. 1986. On the length of proofs of finitistic consistency statements in first order theories. *Studies in Logic and the Foundations of Mathematics* 120 (1986), 165–196.
[24] Pavel Pudlak. 1987. Improved bounds to the length of proofs of finite consistency statements. *Contemp. Math.* 65 (1987), 309–331.
[25] Pavel Pudlak. 2020. Reflection principles, propositional proof systems, and theories. *ArXiV* (July 2020).
[26] Ran Raz. 2010. Elusive Functions and Lower Bounds for Arithmetic Circuits. *Theory of Computing* 6, 1 (2010), 135–177. https://doi.org/10.4086/toc.2010.v006a007
[27] Alexander A. Razborov. 2015. Pseudorandom Generators Hard for k-DNF Resolution and Polynomial Calculus Resolution. *Annals of Mathematics* 181 (2015), 415–472.
[28] Alexander A. Razborov and Steven Rudich. 1997. Natural proofs. *J. Comput. System Sci.* 55, 1, part 1 (1997), 24–35.
[29] Steven Rudich. 1997. Super-bits, Demi-bits, and NP/qpoly-natural Proofs. In *RANDOM'97, Proceedings*, Vol. 1269. 85–93.
[30] Amir Shpilka and Amir Yehudayoff. 2010. Arithmetic Circuits: A survey of recent results and open questions. *Foundations and Trends in Theoretical Computer Science* 5, 3-4 (2010), 207–388. https://doi.org/10.1561/0400000039
[31] Michael Shub and Steve Smale. 1995. On the Intractability of Hilbert's Nullstellensatz and an Algebraic Version of "NP≠P?". *Duke Math. J.* 81 (1995), 47–54.
[32] Volker Strassen. 1973. Vermeidung von Divisionen. *J. Reine Angew. Math.* 264 (1973), 182–202. (in German).
[33] Leslie G. Valiant. 1979. Completeness classes in algebra. In *Proceedings of the 11th Annual ACM Symposium on the Theory of Computing*. ACM, 249–261.
[34] Leslie G. Valiant. 1979. The Complexity of Computing the Permanent. *Theor. Comput. Sci.* 8 (1979), 189–201. https://doi.org/10.1016/0304-3975(79)90044-6
[35] Leslie G. Valiant. 1982. Reducibility by algebraic projections. *Logic and Algorithmic: International Symposium in honour of Ernst Specker* 30 (1982), 365–380.