Algebraic Proof Systems

A short survey on Nullstellensatz and IPS

Magnus Rahbek Hansen

August 11, 2026

Abstract

This document is a short exposition of the theory of *algebraic proof systems*. The main focus of this survey will be on *Nullstellensatz* (degree and sparsity) as well as *ideal proof systems*.

This document was made as part of the exam for the PhD course, "Advanced Themes in Algebraic Complexity", taken at the University of Copenhagen.

Contents

1 Introduction

Proof complexity theory is the study of the resources needed to prove that a statement is true (or equivalently, to refute a false statement). As a subfield of complexity theory, we care mostly about size of the proof. This is mainly due to the Cook–Reckhow program: prove super-polynomial lower bounds for increasingly strong propositional proof systems, with the final goal of proving NP $\neq$ coNP.

In this survey we study the sizes of proofs from an algebraic point of view. The basic idea being to encode an unsatisfiable Boolean formula as a system of polynomial equations and then give an algebraic certificate of the refutation via the Nullstellensatz identity which says that if

$$f_1(\mathbf{x}) = \cdots = f_m(\mathbf{x}) = 0$$

has no common solution, then there exist polynomials $g_1, \ldots, g_m$ such that

$$f_1 g_1 + \cdots + f_m g_m = 1.$$

The polynomials g_i can therefore be viewed as a proof of unsatisfiability, and different ways of measuring their complexity give rise to different algebraic proof systems.

We will first consider the classical Nullstellensatz proof system, where certificates are measured by degree or sparsity. Such Nullstellensatz lower bounds have been used to obtain lower bounds for stronger propositional systems, such as bounded-depth Frege with counting axioms [Bea+96; Bus+96].

Furthermore, we will take a look at a generalization of the classical Nullstellensatz proof system, called Ideal Proof System (IPS), introduced by Grochow and Pitassi [GP18]. In IPS we represent the certificate by an algebraic circuit and measure the complexity using measure from algebraic complexity theory. This gives a direct connection between proof complexity and algebraic complexity. In fact, super-polynomial IPS lower bounds would imply VP $\neq$ VNP.

The goal of this survey is to introduce these main ideas and to show the connection between algebraic circuit complexity and the complexity of proofs.

2 Preliminaries

For completeness (and for my own training), we give some background on the areas which combine into algebraic proof systems. Many readers will be familiar with (part of) this section and may skip these sections.

2.1 Proof Systems

A proof system is a formal way of certifying that a statement is correct. The two most important properties one wants from a proof system are:

1. **Sound:** Any statement that is *provable* is also *true*.
2. **Complete:** Any statement that is *true* is also *provable*.

By 'provable' we mean that there exists a proof. What this proof looks like depends on your proof system. We will later see some examples.

In the eyes of complexity theory we care mainly about the *length* of the proof and whether it is efficiently checkable: How long must the shortest proof be?

In this survey we will only consider propositional proof complexity, where statements are usually tautologies, or, equivalently, unsatisfiable propositional formulas. The latter viewpoint is quite useful for algebraic proof systems, as these systems often translate Boolean constraints into systems of polynomial equations, where unsatisfiability (often modulo the Boolean cube, $\langle x_i^2 - x_i \rangle$) correspond to unsatisfiability of the Boolean constraints.

2.1.1 Cook–Reckhow Program

The reason we care about polynomial-time verifiable proofs is largely due to Cook and Reckhows program. Cook and Reckhow formalized systems such as Frege by means of finitely many axiom schemes and inference rules, and proposed the program of proving super-polynomial lower bounds for increasingly strong proof systems. Many of these systems can be viewed as restricted versions of Frege, where the formulas occurring in the proof are represented by circuits from a restricted circuit class, such as AC^0, $\mathrm{AC}^0[p]$, or TC^0. The (far-reaching) goal of the program is to show that *no* propositional proof system is polynomially bounded, which would separate NP from coNP, which is a major open problem in complexity theory.

2.1.2 Frege systems - a dictionary

We list the relevant propositional proof systems below. Since we won't be working directly with their definition we give only an informal description. For a more formal description see [IS06; Kra19].

Resolution. Resolution is a refutation system for CNF formulas. A proof is a sequence of clauses, each of which is either an initial clause or is derived from two previous clauses by the resolution rule

$$\frac{C \vee x \qquad D \vee \neg x}{C \vee D}.$$

A resolution refutation of a CNF formula F is a derivation of the empty clause from the clauses of F.

Frege. A Frege system is a propositional proof system given by a finite set of axioms and inference rules, over the connectives $\neg, \wedge, \vee, \rightarrow$.

A Frege proof of a formula φ is a sequence of formulas, each of which is either an instance of an axiom or follows from previous formulas by an inference rule and whose final line is φ.

Extended Frege. An Extended Frege system is a Frege system with the addition of an extension rule, which allows the introduction of a fresh variable y together with an axiom

$$y \leftrightarrow \varphi,$$

where φ is a previously constructed formula not containing y.

Extended Frege may also be viewed as a circuit-based version of Frege.

AC^0-Frege. An AC^0-Frege system is a Frege system in which every formula appearing in the proof is represented by a constant-depth, polynomial-size Boolean formula, or equivalently circuit, over unbounded fan-in $\wedge, \vee$ gates and negations at the inputs. Thus, AC^0-Frege is the bounded-depth analog of Frege.

AC^0-Frege with counting axioms. AC^0-Frege with counting axioms is AC^0-Frege along with an additional axiom expressing counting principles. A proof may use both the usual AC^0-Frege axioms and inference rules, and instances of these counting axioms.

$AC^0[p]$-Frege. An $AC^0[p]$-Frege system is a bounded-depth Frege system in which proof lines are constant-depth, polynomial-size circuits over unbounded fan-in $\wedge, \vee, \neg$ gates together with mod_p gates. Here a mod_p gate outputs 1 if and only if the number of its true inputs is nonzero modulo p.

2.1.3 Conjunctive Normal Form Formulas (CNFs)

A literal is either a variable x_i or its negation $\neg x_i$. A clause is a disjunction of literals, and a CNF formula is a conjunction of clauses:

$$F = C_1 \wedge \cdots \wedge C_m.$$

An assignment satisfies F if it satisfies every clause, and F is unsatisfiable if no assignment in $\{0,1\}^n$ satisfies it.

We may translate CNF formulas into a system of polynomials equations by writing $(1 - x_i)$ for the literal x_i and x_j for the literal $\neg x_j$ and taking the product to get a clause. For example:

$$C = x_1 \vee \neg x_2 \vee x_4, \quad \text{translates to} \quad p_C = (1 - x_1)x_2(1 - x_4).$$

More generally,

$$p_C = \prod_{x_i \in C} (1 - x_i) \prod_{\neg x_j \in C} x_j.$$

Hence, $F = C_1 \wedge \cdots \wedge C_m$ is unsatisfiable if and only if the system

$$p_{C_1} = 0, \ldots, p_{C_m} = 0, \qquad x_1^2 - x_1 = 0, \ldots, x_n^2 - x_n = 0$$

has no common Boolean solution. (Note the polynomials $x_i^2 - x_i = 0$ for the solution to be Boolean, anyway).

In Section 3 we will see what a proof of unsatisfiability of such a system can look like.

2.2 Algebraic Circuits

Algebraic complexity theory studies the resources needed to compute various classes of polynomials. It is analogous to Boolean complexity theory, but where the basic operations are arithmetic operations over a field rather than logical operations.

Algebraic computation models are of course useful to algebraic proof systems because their certificates are often polynomials. For example, in ideal proof systems (IPS), proving lower bounds on proof size becomes a question about proving algebraic circuit lower bounds for the certificate polynomials.

2.2.1 Definitions

The main model in algebraic complexity theory is the algebraic circuit: A directed acyclic graph whose leaves are variables or field constants, and whose internal gates are labelled by $+$ or $\times$. Such a circuit computes a polynomial in a very natural way.

For the sake of formality we will give the formal definitions of algebraic circuits:

Definition 2.1 ([SY10])**.** *Let $\mathbb{F}$ be a field and $X = \{x_1, \ldots, x_n\}$ be variables.*

An arithmetic circuit, *Φ, over $\mathbb{F}$ is a directed acyclic graph, $\Phi = (V, E)$, where every vertex has in-degree 0 or 2. The vertices of Φ are called* gates *and are labeled as follows:*

- *Vertices of in-degree 0 are labeled by elements of $\mathbb{F} \cup X$*
- *Vertices of in-degree 2 are labeled by $+$ or $\times$.*

An arithmetic formula *is an arithmetic circuit such that the graph, Φ, is a tree.*

Each gate of an arithmetic circuit naturally computes a polynomial: Define the function eval : $V \to \mathbb{F}(X)$ such that

- If $u \in V$ is labeled by $\alpha \in \mathbb{F} \cup X$ (i.e. of in-degree 0) then $\text{eval}(u) = \alpha$.
- If $u \in V$ is labeled by $* \in \{+, \times\}$ (i.e. of in-degree 2) and has children u_1 and u_2 such that $\text{eval}(u_1) = f_1$ and $\text{eval}(u_1) = f_2$ then $\text{eval}(u) = f_1 * f_2$

Example 2.2. *In the following two arithmetic circuits the unique sink computes* $x^2 + y^2 + 2xy$*:*

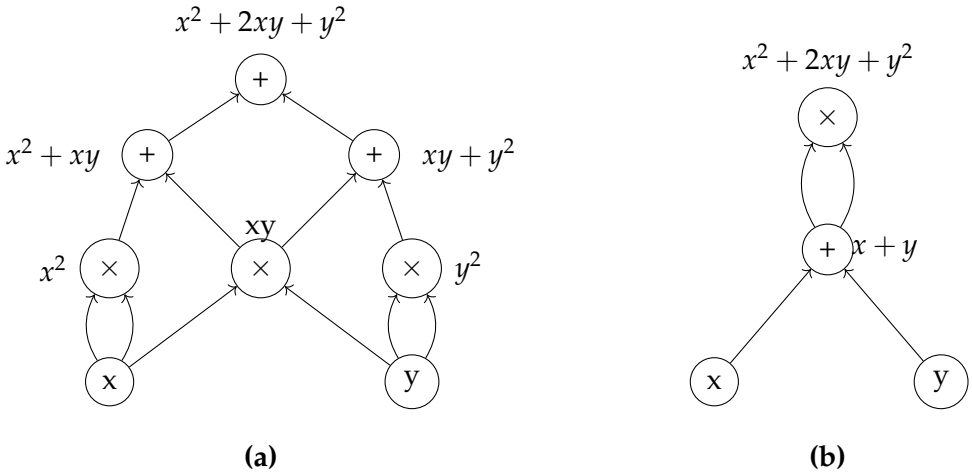

Figure 1: Two ways of computing $x^2 + y^2 + 2xy$

We can see that ***(a)*** *uses 6 operations while* ***(b)*** *only uses 2. Intuitively* ***(b)*** *is a more efficient circuit than* ***(a)****. This intuitive notion is formalized below as the* complexity measure *of arithmetic circuits.*

Definition 2.3. *Let* $\Phi = (V, E)$ *be an arithmetic circuit*

- *The* size *of* Φ *is the number of edges,* $\text{card}(E)$*.*
- *The* depth *of* Φ *is the length of the longest path in* Φ*.*

In Figure 1, **(a)** has size 12 and depth 3 and **(b)** has size 6 and depth 2.

Remark 2.4. The size of a circuit is double the number of vertices labeled by operations since these vertices have in-degree 2. So, if a circuit has no isolated vertices the size of a circuit is asymptotically equal to the number of vertices in the circuit.

2.2.2 Algebraic Complexity Classes

Informally, we consider a polynomial to be easy if it has a small circuit and hard if every circuit computing it is large. The class VP plays the role of efficiently computable polynomial families, while VNP plays the role of efficiently definable polynomial families (every monomial can be efficiently computed). Valiant's conjecture VP $\neq$ VNP is the algebraic analog of P $\neq$ NP.

The main classes we are concerned about is:

Definition 2.5 (VP)**.** *A family of polynomials* $\{f_N\}$ *is in* VP *if there is are polynomially bounded functions* $s, d : \mathbb{N} \to \mathbb{N}$ *such that* $\deg(f_N(\mathbf{x})) \leq d(N)$ *and* $f_N(\mathbf{x})$ *can be computed by a circuit of size* $s(N)$*.*

Definition 2.6 (VF)**.** *A family of polynomials* $\{f_N\}$ *is in* VF *if there is a polynomially bounded function* $s : \mathbb{N} \to \mathbb{N}$ *such that* $f_N(\mathbf{x})$ *can be computed by an algebraic formula of size* $s(N)$*.*

The definitions above are quite general: given an algebraic model $\mathcal{A}$ equipped with a notion of size, one may replace "algebraic formula" or "algebraic circuit" by $\mathcal{A}$ to obtain the corresponding complexity class.[1]

An algebraic version of NP is also defined and plays a central role:

Definition 2.7 (VNP). *A family of polynomials $\{f_N\}$ is in* VNP *if there is a polynomially bounded function $m : \mathbb{N} \to \mathbb{N}$ and a family of polynomials*

$$g_{m(N)+N}(x_1, ..., x_N, y_1, ..., y_{m(N)}) \in VP$$

such that

$$f_N(x_1, ..., x_N) = \sum_{\mathbf{y} \in \{0,1\}^{m(N)}} g_{m(N)+N}(x_1, ..., x_N, \mathbf{y})$$

Alternatively (and more intuitively), VNP are the polynomials for which the coefficients of a given monomial are computable in VP.

Just like in the Boolean world where we conjecture that P $\neq$ NP, we have an analogous open problem in algebraic complexity theory:

Conjecture 2.8. *$VP \neq VNP$.*

Most experts believe the above conjecture to be true.

3 Nullstellensatz-based proof systems

The main fundamental result that allows us to work algebraically with proof systems is the Hilbert Nullstellensatz which is a foundational theorem from algebraic geometry. It bridges geometry and algebra and was proved by David Hilbert in 1893.

More than 100 years after Hilbert's result, Nullstellensatz-based proof systems were introduced as an algebraic way to study propositional proof complexity [Bea+96]. These proof systems were used to prove the first exponential lower bounds for bounded-depth Frege systems with counting axioms [Bus+96].

There are two versions of Hilbert's Nullstellensatz; a weak and a strong version. For this survey we need only the weak:

A set of polynomials $f_1, ..., f_k$ is said to be *unsatisfiable* over S if there *does not* exist $\alpha \in S$ such that

$$f_1(\alpha) = \cdots = f_k(\alpha) = 0.$$

Theorem 3.1 (Weak Nullstellensatz). *Let $\mathbb{F}$ be an algebraically closed field. Let $f_1, ..., f_k \in \mathbb{F}[x_1, ..., x_n]$.*

[1] For example, we may define similar classes for models such as algebraic branching programs and read-once oblivious algebraic branching programs.

Then, $f_1, ..., f_k$ is unsatisfiable over $\mathbb{F}^n$ if and only if there exists $g_1, ..., g_k \in \mathbb{F}[x_1, ..., x_n]$ such that

$$f_1 g_1 + \cdots + f_k g_k = 1. \tag{(3.2)}$$

To give some intuition: If a solution exited then the left-hand side of eq. ((3.2)) would equal zero, while the right-hand side would remain 1.

It is not a far-reaching thought that if you were thinking of proof systems while reading the Hilbert Nullstellensatz, you would think to encode your problem as polynomials, i.e. the f_i's and consider the g_i's as the proof. And this is exactly what we will consider. The question (and the deciding factor) among the various Nullstellensatz-based proof systems is "how should we measure the complexity of the proof; the g_i's?" We could

- measure the maximal degree of g_i,
- measure the sparsity of g_i,
- measure the algebraic circuit size and/or depth of g_i,

and many other ways. Basically any complexity measure from algebraic complexity can (and basically has been) considered.

In this survey we will touch upon all the these proof systems, see how they relate and what deeper results they could imply.

3.1 Nullstellensatz - Degree

Degree lower bounds for the Nullstellensatz certificate might initially seem odd to study, but it turns out that in some cases such lower bounds can be transferred to stronger propositional proof systems, most importantly to AC^0-Frege with counting axioms. Indeed, if we denote by $\deg_{\mathrm{NS}_{\mathbb{Z}_m}}$ the minimum degree of the Nullstellensatz certificate over the ring $\mathbb{Z}/m\mathbb{Z}$ and denote by Count_q^N the unsatisfiable CNF based on the assertion that a set of N elements can be partitioned into blocks of size q, with $q \nmid N$. Then for fixed depth and a prime q not dividing m,

$$\deg_{\mathrm{NS}_{\mathbb{Z}_m}}(\mathrm{Count}_q^N) \geq N^{\omega(1)}$$

gives a AC^0-Frege + counting lower bound of

$$\mathrm{size}_{\mathrm{AC}^0-\mathrm{Frege}+\mathrm{counting}}(\mathrm{Count}_q^N) \geq 2^{N^{\omega(1)}}.$$

All of this was deduced by [Bus+96], and it is recommended to read their paper for more details.

3.1.1 d-designs

One of the main tools to prove lower bounds for degree-Nullstellensatz are d-designs. In fact, d-design-type tools show up quite a few times in algebraic proof systems, such as polynomial calculus.

Definition 3.3 (d-design)**.** *Let $F = \{f_1, ..., f_k\}$ be a set of $\mathbb{F}$-polynomials and let $d \geq 0$.*

A d-design *for F is a linear operator $D : \mathbb{F}[x_1, ..., x_n] \to \mathbb{F}$ such that the following is true:*

1. $D(1) = 1$
2. *D is linear. I.e. $D(\alpha \cdot f + \beta \cdot g) = \alpha D(f) + \beta D(g)$ for all $\alpha, \beta \in \mathbb{F}$ and $f, g \in \mathbb{F}[x_1, ..., x_n]$.*
3. *$D(f_i \cdot g) = 0$ for all $f_i \in F$ and all $g \in \mathbb{F}[x_1, ..., x_n]$ such that $\deg(f_i \cdot g) \leq d$.*

Remark 3.4. Some remarks to understand this definition:

- By linearity, D is completely determined by its values on monomials, and condition 2 and 3 need only hold true for monomials.
- The property of being a d-design is independent of the values on monomials of degree $> d$.
- If the boolean axioms are included, $\{x_1^2 - x_1, ..., x_n^2 - x_n\} \subseteq F$, then, by virtue of condition 2 and 3, D is determined by its values on multilinear monomials. (Because then we'd have $D(x_i^2 \cdot g) = D(x_i \cdot g)$, by linearity).

The strength of d-designs comes from the fact that they completely characterize the degree of the Nullstellensatz refutation:

Theorem 3.5. *$F = \{f_1, ..., f_k\}$ has a d-design if and only if F does* not *have a Nullstellensatz refutation of degree $\leq d$.*

We will only show on implication ($\Rightarrow$). For the converse see [Bus98]

Proof. Assume that $F = \{f_1, ..., f_k\}$ has a d-design, D, and assume for contradiction that there exists a Nullstellensatz refutation:

$$\sum_i f_i g_i = 1$$

of degree $\leq d$. Applying D on both sides gives:

$$0 = \sum_i D(f_i g_i) = D\left(\sum_i f_i g_i\right) = D(1) = 1$$

which is a contradiction. □

The following is an example of how d-design based lower bounds can look. This example is due to [Gri01].

Example 3.6 (Knapsack Problem)**.** *Consider the following unsatisfiable system over* $\mathbb{Q}$*:*

$$f(\mathbf{x}) = \sum_{i=1}^{n} x_i - \frac{1}{2} \quad \text{and boolean axioms} \quad x_i^2 - x_i, \quad i \in [n].$$

(The boolean axioms force $x_i \in \{0, 1\}$ *and so* $\sum x_i$ *can never be* $1/2$*).*

Goal: *Prove Nullstellensatz refutation has degree at least* $n + 1$*.*

We wish to apply theorem 3.5 with $F = \{f, \mathbf{x}^2 - \mathbf{x}\}$*. To this end, define*

$$a_r = \frac{\frac{1}{2}(\frac{1}{2} - 1) \cdots (\frac{1}{2} - r + 1)}{n(n-1) \cdots (n - r + 1)}$$

and use it to define an operator, D*, on monomials* $m = x_1^{t_1} \cdots x_n^{t_n}$

$$D(m) = a_{|\operatorname{supp}(m)|} \quad \text{where} \quad \operatorname{supp}(m) = \{x_i \mid t_i > 0\}.$$

We extend D *linearly to all polynomials. Note that* $D(1) = a_0 = 1$ *and so* D *satisfies 1 and 2 of definition 3.3, per design.*

Since $x_i^2 m$ *and* $x_i m$ *have the same support we have that*

$$D((x_i^2 - x_i)m) = 0$$

for all monomials, m*. So definition 3.3 part 3 is satisfied for* $x_i^2 - x_i$*.*

It remains to check part 3 of definition 3.3 for f*:*

Let m *be a monomial such that* $\deg(f \cdot m) \leq n$ *and let* $r = |\operatorname{supp}(m)|$*. Then* $r \leq n - 1$ *and*

$$D\left(m \cdot \sum_{i=1}^{n} x_i\right) = ra_r + (n - r)a_{r+1}$$

where ra_r *comes from the* x_i*'s which are in* $\operatorname{supp}(m)$ *and* $(n - r)a_{r+1}$ *comes from the* x_i*'s which are not.*

We now note that

$$a_{r+1} = \frac{\frac{1}{2} - r}{n - r} a_r,$$

and so we get

$$D(f \cdot m) = (r - \frac{1}{2})a_r + (n - r)a_{r+1} = (r - \frac{1}{2})a_r + (\frac{1}{2} - r)a_r = 0.$$

Thus, D *is an* n*-design, and it follows from theorem 3.5 that there is no Nullstellensatz refutation of degree at most* n*.*

3.2 Nullstellensatz Sparsity

Another, (and to some more natural), way of measuring the size of a Nullstellensatz refutation is by counting the number of monomials; called its sparsity.

However, it turns out degree-Nullstellensatz can be lifted to sparsity-Nullstellensatz:

Theorem 3.7 ([Fil+24]). *Let $F = \{f_1, ..., f_k\} \cup \{\mathbf{x}_i^2 - \mathbf{x}\}$ be system of polynomials in $\mathbb{F}[x_1, ..., x_n]$, which requires Nullstellensatz degree d.*

Let $\hat{F} = \{\hat{f}_1, ..., \hat{f}_k\} \cup \{\mathbf{x}^2 - \mathbf{x}, \mathbf{y}^2 - \mathbf{y}\}$ be the system of polynomials in $\mathbb{F}[x_1, ..., x_n, y_1, ..., y_n]$ obtained by applying the change of variables,[2]

$$x_i \mapsto x_i + y_i - 2x_i y_i$$

to F.

Then $\hat{F}$ requires sparsity $2^{\omega(d)}$.

Proof Idea. Randomly fix one variable in each pair (x_i, y_i) to

$$(z_i, 0) \quad \text{or} \quad (z_i, 1) \quad \text{or} \quad (0, z_i) \quad \text{or} \quad (1, z_i)$$

each with probability $1/4$, so that $x_i + y_i - 2x_i y_i$ becomes either z_i or $1 - z_i$. This composition followed restriction preserves the Nullstellensatz degree lower bound of F.

A monomial of degree at least d survives with probability at most $(3/4)^d$. Hence, if the proof had fewer than $(4/3)^d$ monomials, a union bound would imply that there is a restriction killing all monomials of degree $\geq d$ which would imply the existence of a refutation of F of degree $< d$. □

For a degree d Nullstellensatz proof, we can construct the proof explicitly in time $n^{O(d)}$, since there are at most $n^{O(d)}$ monomials of degree $\leq d$ and one can then perform linear algebra to construct the refutations of the Nullstellensatz proof.

But for sparsity, it could be the case that if a refutation is sparse then there is a much faster algorithm, since the worst case of having all $n^{O(d)}$ monomials in the refutation is no longer possible. The problem lies in knowing *which* monomials could appear in the proof. If we knew this, then it's again just a matter of using linear algebra.

However, it was shown in [Rez+21] that the existence of an algorithm that could always output a refutation in time $\text{poly}(|F| + s_F)$, where s_F is the smallest refutation of the unsatisfiable CNF, F, would imply that P = NP.

[2] This change of variables can be thought of as composition with $XOR = x_i \oplus y_i = x_i + y_i - 2x_i y_i$.

4 Ideal Proof Systems (IPS)

In 2018 Ideal Proof System (IPS) was introduced by Grochow and Pitassi [GP18]. Since then this proof system and its variation have been studied a lot.

Intuitively IPS should be thought of as Nullstellensatz but where the proof is now given by algebraic circuits. There are different ways to do this, giving us IPS, IPS_{LIN} and $\text{IPS}_{\text{LIN}'}$. Furthermore, we can restrict to any algebraic circuit class, $\mathcal{C}$ yielding many more proof systems to analyze: $\mathcal{C}$-IPS, $\mathcal{C}$-IPS_{LIN}, $\mathcal{C}$-$\text{IPS}_{\text{LIN}'}$.

Definition 4.1 ([GP18])**.** *Let $f_1, ..., f_k \in \mathbb{F}[x_1, ..., x_n]$ be a system of polynomials such that there is no Boolean assignment $\alpha \in \{0,1\}^n$ to the variables $x_1, ..., x_n$ so that $f_i(\alpha) = 0$ for all i.*

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

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

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

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

It is clear that $\mathcal{C}$-$\text{IPS}_{\text{LIN}'}$ is a fragment of $\mathcal{C}$-IPS and that $\mathcal{C}$-IPS_{LIN} is a fragment of $\mathcal{C}$-$\text{IPS}_{\text{LIN}'}$.

The definition of IPS can be a bit unintuitive, but it is really just ideal membership and Nullstellensatz in disguise:

- IPS works because after substituting $y_i \mapsto f_i$ and $z_i \mapsto x_i^2 - x_i$ we get an element of the ideal, $\langle f_1, .., f_k, x_1^2 - x_1, ..., x_n^2 - x_n \rangle$, generated by the axioms, and since

  $$C(\mathbf{x}, f_1, ..., f_k, x_1^2 - x_1, ..., x_n^2 - x_n) = 1$$

  we may conclude this ideal contains 1, which means the system of polynomials given by the axioms has no solution. (Since $V(1) = \emptyset$).

  $\mathcal{C}$-IPS is the most general since it can be non-linear in the placeholder variables y_i, z_i.

- $\mathcal{C}$-IPS_{LIN} are also called Hilbert-like. The reason for this is that we require y_i and z_i to be linear, meaning the polynomial computed by C will be of the form

  $$\sum_{j=1}^{k} y_j g_j(\mathbf{x}) + \sum_{i=1}^{n} z_i h_i(\mathbf{x})$$

which, after substitution, becomes

$$\sum_{j=1}^{k} f_j(\mathbf{x})g_j(\mathbf{x}) + \sum_{i=1}^{n}(x_i^2 - x_i)h_i(\mathbf{x}) = 1$$

which is exactly Nullstellensatz! Thus, $\mathcal{C}$-IPS$_{\text{LIN}}$ is the true $\mathcal{C}$-circuit-size analog of Nullstellensatz.

- $\mathcal{C}$-IPS$_{\text{LIN}'}$ is an intermediate version between $\mathcal{C}$-IPS and $\mathcal{C}$-IPS$_{\text{LIN}}$ and is linear in the 'main' axioms $(f_1, ..., f_k)$ but non-linear in the boolean axioms $(x_1^2 - x_1, ..., x_n^2 - x_n)$, which makes this model a bit more flexible than $\mathcal{C}$-IPS$_{\text{LIN}}$.

IPS can easily be shown to be sound. Completeness follows from the Nullstellensatz. (However, the completeness of the restricted case, $\mathcal{C}$-IPS, depends on the class $\mathcal{C}$).

4.1 Motivation for Ideal Proof Systems

But is it useful? Let's look at some motivating theorems. In their seminal work, [GP14], Grochow and Pitassi show the following theorem:

Theorem 4.2 ([GP14]). *Super-polynomial lower bounds for the Ideal Proof System imply that the permanent does not have polynomial-size algebraic circuits, that is, VNP $\neq$ VP.*

While this is not too surprising, prior this work [GP14], essentially all results were in the opposite direction: A circuit complexity lower bound yielding a proof complexity lower bound

Furthermore, Under a reasonable assumption on polynomial identity testing (PIT) [GP14] show that Extended Frege is equivalent to the Ideal Proof System. Using this together with theorem 4.2 suggest that Extended Frege lower bounds are difficult because such results would imply VP $\neq$ VNP.

Finally, [GP14] makes the point that many algebraic proof systems can be defined in terms of complexity measures on IPS by looking at the degree of the IPS$_{\text{LIN}}$ certificate and considering so-called "rule-based" circuits for polynomial calculus.

Many other foundational results were shown in [GP14] and it is a recommended read.

There is also good reason to study the fragment, IPS$_{\text{LIN}}$. Indeed, it turns out general IPS proofs can be efficiently simulated by IPS$_{\text{LIN}}$ proofs, given that the axioms to be refuted are described by small algebraic circuits. This was a result of [For+21] which is another seminal work outlining strategies and lower bounds for various IPS-based systems.

Proposition 4.3 ([For+21]). *Let $f_1, ..., f_m$ be unsatisfiable polynomials (with Boolean axioms included in this set) with an IPS refutation C such that $f_1, ..., f_m, C$ are computed by size-s formulas/circuits. Then $f_1, ..., f_m$ have an IPS$_{LIN}$ refutation, C', which can be computed by a* poly(s, d, m)*-size formula/circuits.*

[For+21] is also famous for outlining two methods for proving lower bounds for IPS. These are quite fundamental, and we will give a short outline of these methods here:

4.2 The Multiples Method

Consider one single unsatisfiable non-Boolean axiom, $f = 0$, together with the boolean axioms, $x_i^2 - x_i = 0$.

A general IPS certificate may depend non-linearly on the placeholder y for f:

$$C(\mathbf{x}, y, \mathbf{z}) = C_0(\mathbf{x}, \mathbf{z}) + yC_1(\mathbf{x}, \mathbf{z}) + y^2C_2(\mathbf{x}, \mathbf{z}) + \cdots$$

substituting $y = f$ and $z_i = x_i^2 - x_i$ we get

$$f(\mathbf{x}) \cdot H(\mathbf{x}) = 1 - C_0(\mathbf{x}, \mathbf{x}^2 - \mathbf{x})$$

since $C(\mathbf{x}, f, \mathbf{x}^2 - \mathbf{x}) = 1$ by definition. The right-hand side is a nonzero polynomial with complexity lower bounded by the complexity of C. Thus, a small IPS refutation gives a small nonzero multiple of f. Taking the contrapositive of this we see that if every multiple of f is hard for $\mathcal{C}$ then we must get a hard $\mathcal{C}$-IPS certificate. If $\mathcal{C}$ is closed under factoring we know that every multiple of a hard polynomial is hard and so automatically get a corresponding IPS lower bound.

This applies directly to full IPS as it doesn't require linearity in any of the axiom placeholders.

4.3 The Functional Method

Consider one single unsatisfiable non-Boolean axiom, $f = 0$, together with the boolean axioms, $x_i^2 - x_i = 0$.

An IPS_{LIN} refutation has the form

$$fg + \sum_{i=1}^{n} (x_i^2 - x_i)B_i = 1$$

so over the boolean cube this gives

$$g = \frac{1}{f} \mod \langle x_i^2 - x_i \rangle$$

Therefore, to prove a lower bound for $\mathcal{C}$-IPS_{LIN}, it suffices to show that every polynomial g computed by $\mathcal{C}$ which agrees with $1/f$ on the Boolean cube must be large. For some models, such as roABP, this is always the case.

This method also gives a lower bound for $\text{IPS}_{\text{LIN}'}$, since any non-linear dependence on the boolean axioms disappear modulo $\langle x_i^2 - x_i \rangle$

4.4 What kinds of instances are desirable?

How interesting a separation of $\mathcal{C}$-IPS from $\mathcal{D}$-IPS can vary a lot depending on the instance used.

A rough hierarchy of desirability of separation (and more generally, lower bound) instances is as follows, with 1 most desirable:

1. **Hard natural CNF**:
   CNFs are the propositional objects that a lot of proof complexity is intended to refute. A lower bound for an algebraic system on a CNF is immediately a lower bound for a family of ordinary Boolean contradictions [Ale+19]. In particular, a constant-depth IPS lower bound over $\mathbb{F}_p$ for a CNF would imply an $\mathrm{AC}^0[p]$-Frege lower bound, which is a major open problem. [HLT26]

2. **Simple axioms in both models giving a genuine separation**:
   In this case the separation is caused by the difference in proving power between $\mathcal{C}$ and $\mathcal{D}$, not by one model being unable to represent the input. However, the input it not 'natural'.

3. **An instance whose axiom is itself hard for the lower-bounded model**:
   A lower bound of such an instance will likely only show that one model cannot compute an input axiom.

Lower bounds of instance 1, above, are very difficult and only recently in [HLT26] has a CNF instance lower bound for roABP-IPS been proven.

On the other end, it turns out that under weak closure properties a separation of instance 3, above, is not much more difficult than algebraic complexity class separation for many complexity class pairs, $(\mathcal{C}, \mathcal{D})$. The following two subsubsections show the construction used to do this.

4.4.1 Construction of Separation Results via Multiples Method

Say we're given a polynomial $f(\mathbf{x})$ that is computable in $\mathcal{C}$ but not computable in $\mathcal{D}$. Define the unsatisfiable polynomial

$$\hat{f}(\mathbf{x}, y) = 1 - f(\mathbf{x}) \cdot y \cdot (y - 1)$$

Then it's easy to see that $\hat{f} = 1$ over the Boolean cube, and that $\hat{f}$ has the same upper bound as f.

For the lower bound, it follows directly from the multiples' method, so long as $\mathcal{D}$ is closed under factoring.

4.4.2 Construction of Separation Results via Functional Method

Say we're given a polynomial $f(\mathbf{x})$ that is computable in $\mathcal{C}$ but not computable in $\mathcal{D}$. Using the recipe of section 2.1 in [Beh+26] we can 'Booleanize' f to get a polynomial $f_{bool}(\mathbf{x}, \mathbf{y})$ such that we can assume $f_{bool}(\{0,1\}^N) = \{0,1\}$. From here we can define

$$\hat{f} = 1 - 2f_{bool}$$

Since $\hat{f}$ has image $\{-1, 1\}$ over the boolean cube, it follows that $\hat{f}^{-1} = \hat{f}$ modulo $\langle x_i^2 - x_i, y_j^2 - y_j \rangle$. Furthermore, it is possible to partially evaluate $\hat{f}$ at some point $\beta \in \mathbb{F}^k$ such that $\hat{f}(\mathbf{x}, \beta) = f(\mathbf{x})$. So if $\mathcal{C}$ is closed under the transformation $f \mapsto \hat{f}$ and $\mathcal{D}$ is closed under partial evaluation, then $\hat{f}$ separates $\mathcal{C}$-IPS$_{\text{LIN}}$ from $\mathcal{D}$-IPS$_{\text{LIN}}$, since the certificate,

$$\hat{f}\hat{f}^{-1} + \sum_{v \in \mathbf{x} \cup \mathbf{y}} (v^2 - v) B_v = 1$$

would require $\mathcal{D}$ to be able to compute f.

The upper bound for the B_v polynomials in $\mathcal{C}$ would of course require a bit more work, but can often be done. (In fact, it turns out one can choose $B_v = -\frac{\text{ml}_{<v}(\hat{f}^2-1) - \text{ml}_{\leq v}(\hat{f}^2-1)}{v^2-v}$, given some ordering of the variables).[3]

5 Summary and Further Research

We've covered two ways of measuring the complexity of algebraic proofs:

1. The classic application of the Nullstellensatz yielding a measure via degrees and sparsity.
2. The newer ideal proof systems which measures the algebraic complexity of the certificate itself.

Certainly both have promising applications, and IPS is particularly interesting for algebraic complexity theorists; circuit lower bounds can be used to prove proof complexity lower bounds and vice versa.

IPS restricted to a particular class of polynomials give a large (infinite in some sense) family of proof systems laying between classical Nullstellensatz proof systems and unrestricted IPS. This makes it possible to take many smaller steps towards the goal of unrestricted IPS lower bounds via CNF formulas!

5.1 So... Where are we today?

For unrestricted IPS essentially no non-trivial lower bounds are currently known. Indeed, a super-polynomial IPS lower bound would already imply VP $\neq$ VNP, which may help explain why such a lower bound would be difficult to achieve.

Much more progress have been made for IPS restricted to certain classes of polynomials. The two methods of [For+21] (functional/multiples) reduce proof lower bounds to more familiar algebraic complexity questions, and has been used to achieve many lower bounds/separation results in the restricted setting.

[3] Proof of this can be found in my article that is in work in progress...

However, these methods often don't work for CNF formulas, and new techniques may be required for lower bounds of CNF formulas. Thus obtaining lower bounds for CNF instances is much more difficult, but also much more relevant to propositional proof complexity. Recent work [HLT26] has pushed in this direction, but more work is needed.

5.2 Open Problems

A few concrete open problems to ponder:

1. Prove a constant-depth IPS lower bound over $\mathbb{F}_p$ for a CNF. Such a result would imply an $\mathrm{AC}^0[p]$-Frege lower bound.
2. Prove lower bounds for stronger restricted IPS systems on natural CNF instances. Current best is for roABP-IPS.
3. Find genuine separations between $\mathcal{C}$-IPS and $\mathcal{D}$-IPS where the axioms are efficiently computable in both $\mathcal{C}$ and $\mathcal{D}$, or better yet, CNF formulas.
4. Develop techniques which can be efficiently used to prove lower bounds of CNF instances.

References

[Bea+96] Paul Beame, Russell Impagliazzo, Jan Krajíček, Toniann Pitassi, and Pavel Pudlák. "Lower Bounds on Hilbert's Nullstellensatz and Propositional Proofs". In: *Proceedings of the London Mathematical Society* s3-73.1 (1996), pp. 1–26 (cited on pages 2, 7).

[Bus+96] S. Buss, R. Impagliazzo, J. Krajíček, P. Pudlák, A. A. Razborov, and J. Sgall. "Proof complexity in algebraic systems and bounded depth Frege systems with modular counting". In: *computational complexity* 6.3 (Sept. 1996), pp. 256–298 (cited on pages 2, 7, 8).

[Bus98] Samuel R. Buss. "Lower Bounds on Nullstellensatz Proofs via Designs". In: *Proof Complexity and Feasible Arithmetics*. Ed. by Paul Beame and Samuel R. Buss. Vol. 39. DIMACS Series in Discrete Mathematics and Theoretical Computer Science. Providence, RI: American Mathematical Society, 1998, pp. 59–71 (cited on page 9).

[Gri01] D. Grigoriev. "Complexity of Positivstellensatz proofs for the knapsack". In: *computational complexity* 10.2 (Dec. 2001), pp. 139–154 (cited on page 9).

[IS06] Russell Impagliazzo and Nathan Segerlind. "Constant-Depth Frege Systems with Counting Axioms Polynomially Simulate Nullstellensatz Refutations". In: *ACM Transactions on Computational Logic* 7.2 (2006), pp. 199–218 (cited on page 3).

[SY10] Amir Shpilka and Amir Yehudayoff. "Arithmetic Circuits: A survey of recent results and open questions". In: *Foundations and Trends in Theoretical Computer Science* 5 (Jan. 2010), pp. 207–388 (cited on page 5).

[GP14] Joshua A. Grochow and Toniann Pitassi. *Circuit complexity, proof complexity, and polynomial identity testing*. 2014. arXiv: 1404.3820 [cs.CC] (cited on page 13).

[GP18] Joshua A. Grochow and Toniann Pitassi. "Circuit Complexity, Proof Complexity, and Polynomial Identity Testing: The Ideal Proof System". In: *J. ACM* 65.6 (Nov. 2018) (cited on pages 2, 12).

[Ale+19] Yaroslav Alekseev, Dima Grigoriev, Edward A. Hirsch, and Iddo Tzameret. *Semi-Algebraic Proofs, IPS Lower Bounds and the τ-Conjecture: Can a Natural Number be Negative?* 2019. arXiv: 1911.06738 [cs.CC] (cited on page 15).

[Kra19] Jan Kraj'iček. *Proof Complexity*. Vol. 170. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 2019 (cited on page 3).

[For+21] Michael A. Forbes, Amir Shpilka, Iddo Tzameret, and Avi Wigderson. "Proof Complexity Lower Bounds from Algebraic Circuit Complexity". In: *Theory Comput.* 17 (2021), pp. 1–88 (cited on pages 13, 16).

[Rez+21] Susanna F. de Rezende, Mika Göös, Jakob Nordström, Toniann Pitassi, Robert Robere, and Dmitry Sokolov. "Automating algebraic proof systems is NP-hard". In: *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*. STOC 2021. Virtual, Italy: Association for Computing Machinery, 2021, pp. 209–222 (cited on page 11).

[Fil+24] Yuval Filmus, Edward A. Hirsch, Artur Riazanov, Alexander Smal, and Marc Vinyals. "Proving Unsatisfiability with Hitting Formulas". In: *15th Innovations in Theoretical Computer Science Conference (ITCS 2024)*. Ed. by Venkatesan Guruswami. Vol. 287. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024, 48:1–48:20 (cited on page 11).

[Beh+26] Amik Raj Behera, Magnus Rahbek Dalgaard Hansen, Nutan Limaye, and Srikanth Srinivasan. "Separation Results for Constant-Depth and Multilinear Ideal Proof Systems". In: *Electronic Colloquium on Computational Complexity* TR26.TR26-002 (2026). ECCC: TR26-002 (cited on page 15).

[HLT26] Tuomas Hakoniemi, Nutan Limaye, and Iddo Tzameret. *Hard CNF Instances for Ideal Proof Systems*. 2026. arXiv: 2605.04544 [cs.CC] (cited on pages 15, 17).