A Generalized Method for Proving Polynomial Calculus Degree Lower Bounds*

MLADEN MIKŠA, KTH Royal Institute of Technology, Sweden
JAKOB NORDSTRÖM, University of Copenhagen, Denmark

We study the problem of obtaining lower bounds for polynomial calculus (PC) and polynomial calculus resolution (PCR) on proof degree, and hence by [Impagliazzo et al. '99] also on proof size. [Alekhnovich and Razborov '03] established that if the clause-variable incidence graph of a CNF formula F is a good enough expander, then proving that F is unsatisfiable requires high PC/PCR degree. We further develop their techniques to show that if one can "cluster" clauses and variables in a way that "respects the structure" of the formula in a certain sense, then it is sufficient that the incidence graph of this clustered version is an expander. We also show how a weaker structural condition is sufficient to obtain lower bounds on width for the resolution proof system, and give a unified treatment that highlights similarities and differences between resolution and polynomial calculus lower bounds.

As a corollary of our main technical theorem, we prove that the functional pigeonhole principle (FPHP) formulas require high PC/PCR degree when restricted to constant-degree expander graphs. This answers an open question in [Razborov '02], and also implies that the standard CNF encoding of the FPHP formulas require exponential proof size in polynomial calculus resolution. Thus, while onto-FPHP formulas are easy for polynomial calculus, as shown in [Riis '93], both FPHP and onto-PHP formulas are hard even when restricted to bounded-degree expanders.

CCS Concepts: • **Theory of computation → Proof complexity**; **Complexity theory and logic**; *Abstract machines*; *Automated reasoning*.

Additional Key Words and Phrases: Proof complexity, polynomial calculus, polynomial calculus resolution, PCR, resolution, degree, width, size, functional pigeonhole principle, expander graph, lower bound

ACM Reference Format:
Mladen Mikša and Jakob Nordström. 2024. A Generalized Method for Proving Polynomial Calculus Degree Lower Bounds. *J. ACM* VV, N, Article AA (June 2024), 43 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

In one sentence, propositional proof complexity studies how hard it is to certify the unsatisfiability of formulas in conjunctive normal form (CNF). In its most general form, this is the question of whether coNP can be separated from NP or not, and as such it still appears almost completely out of reach. However, if one instead focuses on concrete proof systems, which can be thought of as restricted models of nondeterministic computation, then fruitful study is possible.

*This is an updated and strengthened full-length version of the paper with the same title that appeared in the *Proceedings of the 30th Annual Computational Complexity Conference (CCC '15)*.

Authors' Contact Information: Mladen Mikša, KTH Royal Institute of Technology, School of Electrical Engineering and Computer Science, Stockholm, Sweden, miksa@kth.se; Jakob Nordström, University of Copenhagen, Department of Computer Science (DIKU), Copenhagen, Denmark, jn@di.ku.dk.

ACM 1557-735X/2024/6-ARTAA
https://doi.org/XXXXXXX.XXXXXXX

1.1 Resolution and Polynomial Calculus

Perhaps the most well-studied proof system in proof complexity is *resolution* [11], in which one derives new disjunctive clauses from a CNF formula until an explicit contradiction is reached, and for which numerous exponential lower bounds on proof size have been shown (starting with [15, 25, 46]). Most of these lower bounds can be established by instead studying the *width* of proofs, i.e., the size of a largest clause appearing in the proofs, and arguing that any resolution proof for a certain formula must contain a large clause. It then follows from a celebrated paper by Ben-Sasson and Wigderson [9] that any resolution proof must also consist of very many clauses, and research since this paper has led to a well-developed machinery for showing width lower bounds, and hence also size lower bounds.

The focus of the current paper is the more general proof system *polynomial calculus resolution (PCR)*. This proof system was introduced by Clegg et al. [16] in a slightly weaker form that is usually referred to as *polynomial calculus (PC)* and was later extended by Alekhnovich et al. [1]. In PC and PCR clauses are translated to multilinear polynomials over some (fixed) field $\mathbb{F}$, and a CNF formula $\mathcal{F}$ is shown to be unsatisfiable by proving that the constant 1 lies in the ideal generated by the polynomials corresponding to the clauses of $\mathcal{F}$. Here the size of a proof is measured as the number of monomials when all polynomials in the proof are expanded out as linear combinations of monomials, and the width of a clause corresponds to the (total) *degree* of the polynomial representing the clause. Briefly, the difference between PC and PCR is that the latter proof system has separate formal variables for positive and negative literals over the same variable. Thanks to this, one can encode wide clauses into polynomials compactly regardless of the sign of the literals in the clauses, which allows PCR to simulate resolution efficiently. With respect to the degree measure PC and PCR are exactly the same, and furthermore the degree needed to prove in polynomial calculus that a formula is unsatisfiable is at most the width required in resolution.

In a work that served, interestingly enough, as a precursor to [9], Impagliazzo et al. [27] showed that strong lower bounds on the degree of PC proofs are sufficient to establish strong size lower bounds. The same argument works for PCR, and hence any lower bound on proof size obtained via a degree lower bound applies to both PC and PCR. In this paper, we will therefore be somewhat sloppy in distinguishing between the two proof systems, sometimes writing "polynomial calculus" to refer to both systems when the results apply to both PC and PCR.

In contrast to the situation for resolution after [9], the paper [27] has not been followed by a corresponding development of a generally applicable machinery for proving degree lower bounds. For fields of characteristic distinct from 2 it is sometimes possible to obtain lower bounds by doing an affine transformation from $\{0, 1\}$ to the "Fourier basis" $\{-1, +1\}$, an idea that seems to have appeared first in [12, 24]. For fields of arbitrary characteristic Alekhnovich and Razborov [2] developed a technique for general systems of polynomial equations, which when restricted to the standard encoding of CNF formulas F yields that polynomial calculus proofs require high degree if the corresponding bipartite clause-variable incidence graphs $G(F)$ are good enough expanders. Unfortunately, there are many formula families for which this is not true. One can have a constraint satisfaction problem where the constraint-variable incidence graph is an expander—say, for instance, for an unsatisfiable set of linear equations mod 2—but where each constraint is then translated into several clauses when encoded into CNF, meaning that the clause-variable incidence graph $G(F)$ will no longer be expanding. For some formulas this limitation is inherent—it is not hard to see that an inconsistent system of linear equations mod 2 is easy to refute in polynomial calculus over $\mathbb{F}_2$, and so good expansion for the constraint-variable incidence graph should *not* in itself be sufficient to imply hardness in general—but in other cases it would seem that some kind of expansion of this sort should still be enough, "morally speaking," to guarantee that the corresponding CNF formulas are hard.[1] Formalizing this intuition seems tricky, however, and although the method in [2] appears to be very powerful, the only papers we are aware of preceding the current work that successfully use and elaborate on the Alekhnovich–Razborov framework are [21, 22].

1.2 Pigeonhole Principle Formulas

One important direction in proof complexity, which is the reason research in this area was initiated by Cook and Reckhow [18], is to prove superpolynomial lower bounds on proof size for increasingly stronger propositional proof systems. For proof systems where such lower bounds have already been obtained, however, such as resolution and polynomial calculus, a somewhat orthogonal research direction has been to try to gain a better understanding of the strengths and weaknesses of a given method of reasoning by studying different combinatorial principles and determining how hard they are to prove for the corresponding proof system.

It seems fair to say that by far the most extensively studied such combinatorial principle is the *pigeonhole principle*. This principle is encoded into CNF as unsatisfiable formulas claiming that m pigeons can be mapped in a one-to-one fashion into n holes for $m > n$, but there are several choices exactly how to do this encoding. The most basic *pigeonhole principle (PHP) formulas* have clauses saying that every pigeon gets at least one pigeonhole and that no hole contains two pigeons. While these formulas are already unsatisfiable for $m \geq n + 1$, they do not a priori rule out that there might be “fat” pigeons residing in several holes. The *functional pigeonhole principle (FPHP) formulas* correspond more closely to our intuitive understanding of the pigeonhole principle in that they also contain *functionality* clauses specifying that every pigeon gets exactly one pigeonhole and not more. Another way of making the basic PHP formulas more constrained is to add *onto* clauses requiring that every pigeonhole should get a pigeon, yielding so-called *onto-PHP formulas*. Finally, the most restrictive encoding, and hence the hardest one when it comes to proving lower bounds, are the *onto-FPHP formulas* containing both functionality and onto clauses, i.e., saying that the mapping from pigeons to pigeonholes is a perfect matching. Razborov’s survey [39] gives a detailed account of these different flavours of the pigeonhole principle formulas and results for them with respect to various proof systems—we just quickly highlight some facts relevant to this paper below.

For the resolution proof system there is not much need to distinguish between the different PHP versions discussed above. The lower bound by Haken [25] for formulas with $m = n + 1$ pigeons can be made to work also for onto-FPHP formulas, and more recent works by Raz [36] and Razborov [38, 40, 41] show that the formulas remain exponentially hard (measured in the number of pigeonholes n) even for arbitrarily many pigeons m.

Interestingly enough, for polynomial calculus the story is very different. The first degree lower bounds were proven by Razborov [37], but for a different encoding than the standard translation from CNF, since translating wide clauses yields initial polynomials of high degree. Alekhnovich and Razborov [2] proved lower bounds for a 3-CNF version of the pigeonhole principle, from which it follows that the standard CNF encoding requires proofs of exponential size. However, as shown by Riis [43] the onto-FPHP formulas with $m = n + 1$ pigeons are easy for polynomial calculus. And while the encoding in [37] also captures the functionality restriction in some sense, it has remained

[1] In a bit more detail, what is shown in [2] is that if the constraint-variable incidence graph for a set of polynomial equations is a good expander, and if these polynomials have high immunity—i.e., do not imply other polynomials of significantly lower degree—then proving in polynomial calculus that this set of polynomial equations is inconsistent requires high degree. CNF formulas automatically have maximal immunity since a clause translated into a polynomial does not have any consequences of degree lower than the width of the clause in question, and hence expansion of the clause-variable incidence graph is sufficient to imply hardness for polynomial calculus. Any polynomial encoding of a linear equation mod 2 has a low-degree consequence over $\mathbb{F}_2$, though—namely, the linear equation itself—and this is why [2] (correctly) fails to prove lower bounds in this case.

open whether the standard CNF encoding of functional pigeonhole principle formulas translated to polynomials is hard (this question has been highlighted, for instance, in Razborov's open problems list [42]).

Another way of modifying the pigeonhole principle is to restrict the choices of pigeonholes for each pigeon by defining the formulas over a bipartite graph $H = (U \mathbin{\dot\cup} V, E)$ with $|U| = m$ and $|V| = n$ and requiring that each pigeon $u \in U$ goes to one of its neighbouring holes in $N(u) \subseteq V$. If the graph H has constant left degree, the corresponding *graph pigeonhole principle formula* has constant width and a linear number of variables, which makes it possible to apply [9, 27] to obtain exponential proof size lower bounds from linear width/degree lower bounds. A careful reading of the proofs in [2] reveals that this paper establishes linear polynomial calculus degree lower bounds (and hence exponential size lower bounds) for graph PHP formulas, and in fact also graph onto-PHP formulas, over constant-degree expanders. Razborov lists as one of the open problems in [39] whether this holds also for graph FPHP formulas, i.e., with functionality clauses added, from which exponential lower bounds on polynomial calculus proof size for the general FPHP formulas would immediately follow.

1.3 Our Results

We revisit the technique developed in [2] for proving polynomial calculus degree lower bounds for CNF formulas. Instead of studying the standard bipartite clause-variable incidence graph $G(F)$ of a CNF formula F (with clauses on the left, variables on the right, and edges encoding that a variable occurs in a clause) we consider graphs G' that can be constructed by clustering several clauses and/or variables into single vertices, reflecting the structure of the encoded combinatorial principle. The edges in this new graph G' are the ones induced by the original graph $G(F)$ in the natural way, i.e., there is an edge from a left cluster to a right cluster in G' if any clause in the left cluster has an edge to any variable in the right cluster in $G(F)$. We remark that the idea of clustering in itself is not new—it is already implicit in, for instance, the resolution lower bounds in [9] for Tseitin formulas (essentially a special form of unsatisfiable linear equations mod 2) and graph PHP formulas, as well as in the graph PHP lower bound for polynomial calculus in [2], and it is also used in the papers [21, 22] building on [2]. But whereas in the above works the clustering was done in a instance-specific way, we present a more general, abstract setting. We also consider any input sets of polynomials, not just polynomials obtained from translations of CNF formulas.

In this more general setting we then show that if the clustering is done in the right way, the proof strategy in [2] can still be made to work. If the clustered graph G' is a good enough expander (for a certain technical twist of the definition of expander that is described in Definitions 4.4 and 4.8 in Section 4.1), then this yields strong polynomial calculus degree lower bounds. It is clear that this cannot always work—as already discussed above, any inconsistent system of linear equations mod 2 is easy to refute in polynomial calculus over $\mathbb{F}_2$, even though for a random instance of this problem the clauses encoding each linear equation can be clustered to yield an excellent expander G'. Very informally (and somewhat incorrectly) speaking, the clustering should be such that if a cluster of polynomials P on the left is a neighbour of a variable cluster V on the right, then there should exist an assignment ρ to V such that all polynomials in P vanish under ρ and such that for every polynomial outside of P it either vanishes under ρ or is left completely untouched by ρ (see Definitions 4.1 and 4.3). Also, it turns out to be helpful not to insist that the clustering of variables on the right should be a partition, but that we should allow the same variable to appear in several clusters if needed (as long as the number of clusters for each variable is bounded).

This extension of the lower bound method in [2] makes it possible to present previously obtained polynomial calculus degree lower bounds in [2, 22, 33] in a clean, unified framework.[2] Moreover, it allows us to prove the following new results:

(1) If a bipartite graph $H = (U \mathbin{\dot{\cup}} V, E)$ with $|V| = n$ and $|U| = m = \mathrm{O}(n)$ is a boundary expander (a.k.a. unique-neighbour expander), then the graph FPHP formula over H requires proofs of linear polynomial calculus degree, and hence exponential polynomial calculus size.
(2) Since FPHP formulas can be turned into graph FPHP formulas by hitting them with a restriction, and since restrictions can only decrease proof size, it follows that FPHP formulas in the standard CNF encoding require proofs of exponential size in polynomial calculus.

This fills in the last missing pieces in our understanding of the different flavours of pigeonhole principle formulas with $n + 1$ pigeons and n holes for polynomial calculus. Namely, while onto-FPHP formulas are easy for polynomial calculus, both FPHP formulas and onto-PHP formulas are hard even when restricted to expander graphs.

We remark that after the preliminary version of this paper [34] was published, we have learned that a similar lower bound for FPHP formulas was obtained in [49]. We also note that a different, more abstract, treatment of the Alekhnovich–Razborov method was independently developed by Filmus [20]. The focus of [20] appears to be mainly on constructing different and more explicit proofs for the key technical lemmas in [2], however, and that work does not obtain any new lower bound results.

1.4 Organization of This Paper

The rest of this paper is organized as follows. We start by reviewing some preliminaries in Section 2. Next, we give a reader-friendly exposition of our main technical contributions in Section 3 by explaining how lower bounds can be obtained for resolution and polynomial calculus by constructing bipartite graphs representing the input (generalizing the concept of clause-variable incidence graph) and proving that these graphs satisfy simple combinatorial properties. We give a more formal treatment of our extension of the Alekhnovich–Razborov method in Section 4, including all technical details and proofs. In Section 5, we show how this method can be used to rederive some previous polynomial calculus degree lower bounds as well as to obtain new degree and size lower bounds for functional (graph) PHP formulas. We conclude in Section 6 by discussing some possible directions for future research.

2 Preliminaries

In this section we give a brief overview of the required proof complexity background (referring the reader to, for instance, the survey article [13] or the book [30] for a more detailed treatment), and then discuss some concepts from algebra that we will also need.

2.1 Proof Complexity Basics

A *literal* over a Boolean variable x is either the variable x itself (a *positive literal*) or its negation $\neg x$ or $\overline{x}$ (a *negative literal*). We define $\overline{\overline{x}} = x$. We identify *true* ($\top$) with 0 and *false* ($\bot$) with 1, and will use these pieces of notation interchangeably. We remark that this is the opposite of the standard convention in proof complexity, where we tend to think of 1 as *true*, but it is a more natural choice in the context of polynomial calculus, where "evaluating to true" means "vanishing." A *clause* $C = a_1 \lor \cdots \lor a_k$ is a disjunction of literals. A *CNF formula* $\mathcal{F} = C_1 \land \cdots \land C_m$ is a conjunction of

[2] However, the work [21] mentioned earlier has a problem-specific and non-standard notion of degree that will not fit easily into our general setting without sacrificing the simplicity that is the main goal of our constructions, and for this reason we will not discuss it further in this paper.

clauses. We think of clauses and CNF formulas as sets, so that order is irrelevant and there are no repetitions. The *width* $W(C)$ of a clause C is the number of literals $|C|$ in it, and the width $W(\mathcal{F})$ of a formula $\mathcal{F}$ is the maximum width of any clause in $\mathcal{F}$. A k-CNF formula has all clauses of width at most k, where we usually assume k to be some fixed constant.

A truth value assignment α to variables satisfies the literal x if $\alpha(x) = \top$ and the literal $\overline{x}$ if $\alpha(x) = \bot$. A clause C is satisfied by α if α satisfies some literal in it, and a CNF formula $\mathcal{F}$ is satisfied if α satisfies all $C \in \mathcal{F}$. We say that a CNF formula (or set of clauses) $\mathcal{F}$ *implies* a clause C, denoted $\mathcal{F} \vDash C$, if any truth value assignment that satisies $\mathcal{F}$ must also satisfy C. A CNF formula $\mathcal{F}$ without satisfying assignments is *unsatisfiable*.

Definition 2.1 (Resolution). A *resolution refutation* $\pi : \mathcal{F} \vdash \bot$ of a CNF formula $\mathcal{F}$ (also referred to as a *resolution proof* for $\mathcal{F}$) is an ordered sequence of clauses $\pi = (C_1, \ldots, C_\tau)$, such that $C_\tau = \bot$ is the contradictory empty clause not containing any literals, and such that each clause C_i, $1 \leq i \leq \tau$, is

- an *axiom clause* $C \in \mathcal{F}$ (an *axiom*); or
- a clause derived from two previous clauses in the sequence by the *resolution rule* $\frac{B \vee x \quad C \vee \overline{x}}{B \vee C}$.

The *length* (or *size*) $L(\pi)$ of a refutation $\pi = (C_1, \ldots, C_\tau)$ is the number of clauses τ and the *width* $W(\pi)$ is the maximal width of any clause in π. Taking the minimum over all resolution refutations of $\mathcal{F}$, we define the length $L_{\mathcal{R}}(\mathcal{F} \vdash \bot)$ and width $W_{\mathcal{R}}(\mathcal{F} \vdash \bot)$ of refuting $\mathcal{F}$ in resolution.

It is a standard fact that resolution is sound and complete; i.e., there is a resolution refutation of a CNF formula $\mathcal{F}$ if and only if $\mathcal{F}$ is unsatisfiable.

When using algebraic proof systems to refute unsatisfiable CNF formulas, a clause

$$C = \bigvee_{x \in L^+} x \vee \bigvee_{y \in L^-} \overline{y} \tag{2.1}$$

can be translated to the polynomial

$$p_{\mathcal{PC}}(C) = \prod_{x \in L^+} x \cdot \prod_{y \in L^-} (1 - y) \tag{2.2}$$

and a CNF formula is translated to the set of polynomials representing its clauses. Clearly, a CNF formula $\mathcal{F}$ is satisfiable if and only if the set of polynomials $\left\{p_{\mathcal{PC}}(C) \,\middle|\, C \in \mathcal{F}\right\}$ have a common root. We will also be interested in a setting where we have special variables $\overline{x}, \overline{y}, \ldots$ representing negated literals, where we emphasize that x and $\overline{x}$ are viewed as distinct formal variables. Using such variables we can translate the clause C in (2.1) to the monomial

$$p_{\mathcal{PCR}}(C) = \prod_{x \in L^+} x \cdot \prod_{y \in L^-} \overline{y}\,. \tag{2.3}$$

In what follows, a *monomial* m is a product of variables and a *term* t is a monomial multiplied by an arbitrary non-zero field element. We write $\mathit{Vars}(C)$ and $\mathit{Vars}(m)$ to denote the set of all variables appearing in a clause C or monomial (or term) m, respectively and extend this notation to CNF formulas and polynomials by taking unions.

In polynomial calculus resolution, we are given a set of multivariate polynomials $\mathcal{P}$ from a polynomial ring $\mathbb{F}[x, \overline{x}, y, \overline{y}, z, \overline{z}, \ldots]$ over some fixed field $\mathbb{F}$, and the goal is to prove that these polynomials do not have a common $\{0, 1\}$-valued root. We will not have to worry about what field $\mathbb{F}$ is, since the results in this paper hold for all fields $\mathbb{F}$ regardless of characteristic.

Definition 2.2 (Polynomial calculus resolution (PCR) [1, 16]). A *polynomial calculus resolution (PCR) refutation* $\pi : \mathcal{P} \vdash \bot$ of a set of polynomials $\mathcal{P}$ (also referred to as a *PCR proof* for $\mathcal{P}$) over a field $\mathbb{F}$

is an ordered sequence of polynomials $\pi = (p_1, \ldots, p_\tau)$, such that $p_\tau = 1$ and each line p_i, $1 \le i \le \tau$, is

- a polynomial $p \in \mathcal{P}$ (an *axiom*);
- a *Boolean axiom* $x^2 - x$ or *complementarity axiom* $x + \overline{x} - 1$ for any variable x; or
- a polynomial obtained from one or two previous polynomials in the sequence by *linear combination* $\frac{q \quad r}{\alpha q + \beta r}$ or *multiplication* $\frac{q}{xq}$ for any $\alpha, \beta \in \mathbb{F}$ and any variable x.

If we drop complementarity axioms and only allow variables x without bars, then the proof system is called *polynomial calculus (PC)*.

The *size* $S(\pi)$ of a PC/PCR refutation $\pi = (p_1, \ldots, p_\tau)$ is the number of monomials in π (counted with repetitions)[3] when all polynomials are expanded out as linear combinations of monomials, the *degree* $Deg(\pi)$ is the maximal degree of any monomial appearing in π, and the *length* $L(\pi)$ is the number τ of polynomials in π. Taking the minimum over all PCR refutations of a set of polynomials $\mathcal{P}$, we define the size $S_{\mathcal{PCR}}(\mathcal{P} \vdash \perp)$, degree $Deg_{\mathcal{PCR}}(\mathcal{P} \vdash \perp)$, and length $L_{\mathcal{PCR}}(\mathcal{P} \vdash \perp)$ of refuting $\mathcal{P}$ in PCR (and analogously for PC).

We use the notation $\langle p_1, \ldots, p_m \rangle$ for the ideal generated by the polynomials p_i, $i \in [m]$. That is, $\langle p_1, \ldots, p_m \rangle$ is the minimal subset of polynomials containing all p_i that is closed under addition and multiplication by any polynomial. One way of viewing a polynomial calculus (PC or PCR) refutation is as a calculation in the ideal generated by the polynomials in $\mathcal{P}$ together with the Boolean and (for PCR) complementarity axioms. It follows from Hilbert's Nullstellensatz that such an ideal contains 1 if and only if the polynomials in $\mathcal{P}$ do not have a common $\{0, 1\}$-valued root.

A *restriction* ρ on a CNF formula $\mathcal{F}$ is a partial assignment to the variables of $\mathcal{F}$. We use $\mathrm{dom}(\rho)$ to denote the set of variables assigned by ρ. In a restricted formula $\mathcal{F}\!\upharpoonright_\rho$ all clauses satisfied by ρ are removed and all other clauses have falsified literals removed. For a polynomial p, restricting by ρ yields a polynomial $p\!\upharpoonright_\rho$ where all terms with a literal set to 0 are removed and in all other terms the literals set to 1 are removed. It is not hard to see that if π is a PC (or PCR) refutation of $\mathcal{P}$, then $\pi\!\upharpoonright_\rho$ is a PC (or PCR) refutation of $\mathcal{P}\!\upharpoonright_\rho$, and this restricted refutation has at most the same size, degree, and length as the original refutation.

2.2 Some Facts About Polynomial Calculus

As mentioned in the introduction, we have $Deg_{\mathcal{PCR}}(F \vdash \perp) = Deg_{\mathcal{PC}}(F \vdash \perp)$ for any CNF formula F. More generally, if we take any set of polynomials $\mathcal{P}$ in $\mathbb{F}[x, \overline{x}, y, \overline{y}, \ldots]$ and replace all occurrences of $\overline{x}$ by $(1 - x)$ to obtain a set of polynomials $\mathcal{P}^+$ in $\mathbb{F}[x, y, \ldots]$, then it holds that $Deg_{\mathcal{PCR}}(\mathcal{P} \vdash \perp) = Deg_{\mathcal{PCR}}(\mathcal{P}^+ \vdash \perp) = Deg_{\mathcal{PC}}(\mathcal{P}^+ \vdash \perp)$. It is not hard to see that PCR can simulate PC in the same degree, since PCR is strictly more expressive, and in the other direction one can essentially take any PCR refutation of $\mathcal{P}$ and make substitutions of $\overline{x}$ by $(1 - x)$ everywhere to obtain a valid PC refutation of $\mathcal{P}^+$. This might cause the number of monomials to blow up exponentially, but the degree remains the same. Hence, we can drop the subscript from the degree measure notation.

For any set of polynomials $\mathcal{P}$ of degree $Deg(\mathcal{P})$ we have the following relation between refutation size and refutation degree (which was originally proven for PC but the proof of which also works for PCR).

[3] We remark that the natural definition of size is to count monomials with repetition, but all lower bound techniques known actually establish slightly stronger lower bounds on the number of *distinct* monomials.

Theorem 2.3 ([27]). *Let $\mathcal{P}$ be a set of polynomials of degree $Deg(\mathcal{P})$ over n variables such that there is no $\{0,1\}$-assignment for which all polynomials $p \in \mathcal{P}$ evaluate to 0. Then it holds that*

$$S_{\mathcal{PCR}}(\mathcal{P} \vdash \bot) = \exp\left(\Omega\left(\frac{\bigl(Deg(\mathcal{P} \vdash \bot) - Deg(\mathcal{P})\bigr)^2}{n}\right)\right).$$

It follows from this theorem that in order to establish strong lower bounds on PCR proof size for sets of polynomials $\mathcal{P}$ of bounded degree $Deg(\mathcal{P}) = \mathrm{O}(1)$, it is sufficient to prove strong lower bounds on the PC degree of any proof of unsatisfiability. All the lower bounds presented in this paper are possible to obtain by studying (polynomials translations of) k-CNF formulas for $k = \mathrm{O}(1)$, using restrictions to reduce the width in the case when the original formulas have clauses of large size (such as PHP formulas). In the rest of this paper we will therefore only study PC and not PCR, and will focus on showing degree lower bounds. In particular, we will not need distinct formal variables $\overline{x}$ for negated literals over x, but will assume that clauses of a CNF formula are translated into polynomials as in (2.2).

Furthermore, it will be convenient for us to simplify the definition of PC so that axioms $x^2 - x$ are always applied implicitly whenever possible. We do this by defining the result of the multiplication operation to be the multilinearized version of the product. This can only decrease the degree (and size) of the refutation, and is in fact how polynomial calculus is defined in [2]. Hence, from now on whenever we refer to polynomials and monomials we mean multilinear polynomials and multilinear monomials, respectively, and polynomial calculus is defined over the (multilinear) polynomial ring $\mathbb{F}[x, y, z, \ldots]/\langle x^2 - x, y^2 - y, z^2 - z, \ldots\rangle$.

It might be worth noticing that for this modified definition of polynomial calculus it holds that any unsatisfiable k-CNF formula can be refuted in linear length, although the polynomials in the refutation might have an exponential number of monomials. This serves to illustrate the point that in contrast to resolution, for polynomial calculus the *size* of refutations, rather than the *length*, is the right measure on which to focus. This linear length upper bound is not hard to show, and in some sense might even be folklore, but since it does not seem to be too widely known we state it for the record and provide a proof.

Proposition 2.4. *Let $\mathcal{F} = \bigwedge_{i=1}^{m} C_i$ be an unsatisfiable k-CNF formula. Then $\mathcal{F}$ has a multilinear polynomial calculus refutation of length* $\mathrm{O}(km)$.

Proof. Given an unsatisfiable k-CNF formula $\mathcal{F} = \bigwedge_{i=1}^{m} C_i$, we claim that the polynomial $p_j = 1 - \prod_{i=1}^{j}(1 - C_i)$ can be derived in length $\mathrm{O}(kj)$ for $j = 1, \ldots, m$, where we identify the clause $C_i \in \mathcal{F}$ with the polynomial encoding $p_{\mathcal{PC}}(C_i)$ in (2.2) of this clause. The end result is the polynomial $p_m = 1 - \prod_{i=1}^{m}(1 - C_i)$. As $\mathcal{F}$ is unsatisfiable, for every $\{0,1\}$-assignment there is at least one C_i that evaluates to 1 and hence p_m evaluates to 1. Thus, p_m is equal to 1 on all $\{0,1\}$-assignments. However, it is a basic fact that every function $f : \{0,1\}^n \to \mathbb{F}$ is uniquely representable as a multilinear polynomial in $\mathbb{F}[x_1, \ldots, x_n]$. (Since the multilinear monomials span this vector space and are linearly independent, they form a basis.) Therefore, we can conclude that p_m is syntactically equal to the polynomial 1, and the proposition follows.

We proceed to establish the claim by induction. The base case is the polynomial p_1 that is equal to C_1. To prove the induction step, we need to show how to derive

$$p_{j+1} = 1 - \prod_{i=1}^{j+1}(1 - C_i) = 1 - (1 - C_{j+1})(1 - p_j) = p_j + C_{j+1} - C_{j+1}p_j \tag{2.4}$$

from p_j and C_{j+1} in $\mathrm{O}(k)$ steps. To start, we derive $C_{j+1}p_j$ from p_j, which can be done with $\mathrm{O}(k)$ multiplications and additions since the width/degree of C_{j+1} is upper-bounded by k. We derive p_{j+1}

in two more steps by first taking a linear combination of p_j and $C_{j+1}p_j$ to get $p_j - C_{j+1}p_j$ and then adding C_{j+1} to this to obtain $p_j - C_{j+1}p_j + C_{j+1} = p_{j+1}$. The proposition follows. □

We remark that for non-multilinear PC or PCR as in Definition 2.2 the linear-length refutation above will not work, since in general one might need an exponential number of applications of the Boolean axioms $x^2 - x$ to multilinearize the polynomials.

2.3 Some Algebra Basics

When proving lower bounds on polynomial calculus degree, a key step will be to define different polynomial ideals and to reason about other polynomials modulo these ideals. In order to do so we need to have an ordering of monomials (which, as just discussed in Section 2.2, we will always assume to be multilinear).

Definition 2.5 (Admissible ordering). We say that a total ordering $\prec$ on the set of all monomials over some fixed set of variables is *admissible* if the following conditions hold:

- If $Deg(m_1) < Deg(m_2)$, then $m_1 \prec m_2$.
- For any m_1, m_2, and m such that $m_1 \prec m_2$ and $Vars(m) \cap \big(Vars(m_1) \cup Vars(m_2)\big) = \emptyset$, it holds that $mm_1 \prec mm_2$.

We write $m_1 \preccurlyeq m_2$ to denote that $m_1 \prec m_2$ or $m_1 = m_2$.

Two terms $t_1 = \alpha_1 m_1$ and $t_2 = \alpha_2 m_2$ (for $\alpha_1, \alpha_2 \in \mathbb{F}$) are ordered in the same way as their underlying monomials m_1 and m_2.

One example of an admissible ordering is to first order monomials with respect to their degree and then lexicographically. In this paper we will let $\prec$ denote any admissible ordering, but the reader can think of the degree-lexicographical ordering without any particular loss of generality.

In what follows, when we write a polynomial p as a sum of terms $p = \sum_i t_i$ we implicitly assume that all terms are over distinct monomials.

Definition 2.6 (Leading, reducible, and irreducible terms). For a polynomial $p = \sum_i t_i$, the *leading term* $LT(p)$ of p with respect to an admissible ordering $\prec$ is the largest term t_i according to $\prec$. Let I be an ideal over the (multilinear) polynomial ring $\mathbb{F}[x, y, z, \ldots]/\langle x^2 - x, y^2 - y, z^2 - z, \ldots\rangle$. We say that a term t is *reducible modulo* I if there exists a polynomial $q \in I$ such that $t = LT(q)$, and that t is *irreducible modulo* I otherwise.

We have the following basic fact (the proof of which is provided for completeness).

Fact 2.7. *Let I be an ideal over $\mathbb{F}[x, y, z, \ldots]/\langle x^2 - x, y^2 - y, z^2 - z, \ldots\rangle$. Then any multilinear polynomial $p \in \mathbb{F}[x, y, z, \ldots]/\langle x^2 - x, y^2 - y, z^2 - z, \ldots\rangle$ can be written uniquely as a sum $p = q + r$, where $q \in I$ and r is a linear combination of irreducible terms modulo I.*

Proof. The fact that p can be written as $p = q + r$, with $q \in I$ and r containing only irreducible terms modulo I can be argued by induction over $LT(p)$. If $LT(p)$ is irreducible, then by induction we can write $p' = p - LT(p)$ on the required form $p' = q' + r'$, from which we get $p = q' + (LT(p) + r')$. If $LT(p)$ is reducible, then by definition there exists a polynomial $q \in I$ such that $LT(p) = LT(q)$, and $p' = p - q$ only contains terms that are smaller than $LT(p)$. Again we can write p' on the required form $p' = q' + r'$, from which we get $p = (q + q') + r'$.

To argue uniqueness, suppose that we can write $p = q_1 + r_1 = q_2 + r_2$ for $r_1 \neq r_2$. Then rearranging yields $r_1 - r_2 = q_2 - q_1 \in I$ which shows that the leading term in $r_1 - r_2$ is not irreducible after all. This is a contradiction. □

This fact is what allows us to reduce polynomials modulo an ideal in a well-defined manner.

Definition 2.8 (Reduction operator). Let $p \in \mathbb{F}[x, y, z, \ldots]/\langle x^2 - x, y^2 - y, z^2 - z, \ldots\rangle$ be any multilinear polynomial and let I be an ideal over $\mathbb{F}[x, y, z, \ldots]/\langle x^2 - x, y^2 - y, z^2 - z, \ldots\rangle$. The *reduction operator R_I* is the operator that when applied to p returns the sum of irreducible terms $R_I(p) = r$ such that $p - r \in I$.

We conclude our brief algebra review by stating two observations that are more or less immediate, but are helpful enough for us to want to highlight them explicitly.

Observation 2.9. *For any two ideals I_1, I_2 such that $I_1 \subseteq I_2$ and any two polynomials p, p' it holds that $R_{I_2}(p \cdot R_{I_1}(p')) = R_{I_2}(pp')$.*

Proof. Let us write

$$p' = q' + r' \tag{2.5}$$

for $q' \in I_1$ and r' a linear combination of irreducible terms over I_1. In the same way, let us write

$$p \cdot R_{I_1}(p') = pr' = q + r \tag{2.6}$$

for $q \in I_2$ and r a linear combination of irreducible terms over I_2. Then

$$pp' = pq' + pr' = pq' + q + r \tag{2.7}$$

where $pq' + q \in I_2$. By the uniqueness in Fact 2.7, we conclude that the equality $R_{I_2}(pp') = r = R_{I_2}(p \cdot R_{I_1}(p'))$ holds. □

Observation 2.10. *Suppose that the term t is irreducible modulo the ideal I and let ρ be any partial assignment of variables in $Vars(t)$ to values in $\mathbb{F}$ such that $t{\restriction_\rho} \neq 0$. Then $t{\restriction_\rho}$ is also irreducible modulo I.*

Proof. Let $t = m_\rho t'$ where m_ρ is the product of all variables in t assigned by ρ, and let $\alpha = m_\rho{\restriction_\rho}$. Then $t{\restriction_\rho} = \alpha t'$, where by assumption we have $\alpha \neq 0$. If there is a polynomial $q \in I$ such that $LT(q) = t{\restriction_\rho}$, then $\alpha^{-1} m_\rho q \in I$ and $LT(\alpha^{-1} m_\rho q) = \alpha^{-1} m_\rho t{\restriction_\rho} = m_\rho t' = t$, contradicting that t is irreducible. (Note that this final step crucially uses that $\prec$ is not only degree-respecting but admissible.) □

A more concise way of phrasing Observation 2.10 is that the set of irreducible monomials is downward-closed under restrictions.

3 Lower Bounds from Graph Expansion and Combinatorial Games

Many lower bounds in proof complexity are proved by arguing in terms of expansion. One common approach is to associate a bipartite graph $G(\mathcal{F})$ with the CNF formula $\mathcal{F}$ with clauses on one side and variables on the other and with edges encoding that a variable occurs in a clause (the so-called *clause-variable incidence graph* mentioned in the introduction), and establish lower bounds provided that this graph is well-connected. In particular, the following notion of expansion often plays an important role.

Definition 3.1 (Bipartite boundary expander). A bipartite graph $G = (U \,\dot\cup\, V, E)$ is a *bipartite (s, δ)-boundary expander* if for every set of vertices $U' \subseteq U$, $|U'| \leq s$, it holds that $|\partial(U')| \geq \delta|U'|$, where the *boundary* $\partial(U') = \{v \in V : |N(v) \cap U'| = 1\}$ consists of all vertices on the right-hand side V that have a unique neighbour in U' on the left-hand side.

The method we present in this work, which is an extension of the techniques developed by Alekhnovich and Razborov [2], is a variation of the theme of proof complexity lower bounds via graph expansion. As already discussed, however, we will need a slightly more general graph construction where constraints and variables can be grouped into clusters, and we also want to consider not just translations of CNF formulas but arbitrary sets of polynomials as inputs.

In this section, we give a high-level description of the lower bound method in terms of a simple combinatorial game played on bipartite graphs. This will allow us to give a unified treatment of lower bound techniques for resolution width and polynomial calculus degree, highlighting similarities and differences between the two proof systems. Our hope is that this language could also provide a convenient way to teach resolution and polynomial calculus lower bounds as, say, part of an advanced course in computational complexity theory. For simplicity, in this section we only consider CNF formulas, although for polynomial calculus all concepts are easy to extend to general (multilinear) polynomials.

We want to point out that we will present no new results in this section, and also will not prove why our combinatorial game is sufficient to establish degree lower bounds for polynomial calculus. Readers who wish to study the technical details of our new contributions can therefore skip ahead to Section 4.

3.1 Incidence Graphs for Sets of Clauses and Variables

Throughout this section, we let $\mathcal{F}$ denote a CNF formula over variables $\mathcal{V}$. In order to generalize the clause-variable incidence graphs to sets of clauses and variables, we consider partitions of the clauses into $\mathcal{F} = E \cup \bigcup_{i=1}^{m} F_i$, where E is a special clause set that should be satisifiable. We also consider divisions of the variables $\mathcal{V} = \bigcup_{j=1}^{n} V_j$. Importantly, this need *not* be a partition, but we will want it to be "close" to a partition. More formally, we say that $\mathcal{V} = \bigcup_{j=1}^{n} V_j$ has *overlap* bounded by ℓ if any variable x appears in at most ℓ different sets V_j, and we will want our division into variable subsets to have overlap bounded by a constant.

In what follows, we will often overload notation and consider $\mathcal{F}$ and $\mathcal{V}$ to be endowed with such a partition and division, respectively. We let any such representation of $\mathcal{F}$ and $\mathcal{V}$ define a bipartite graph $(\mathcal{F}, \mathcal{V})_E$ in the following way:

- The left vertex set is $\{F_1, \ldots, F_m\}$.
- The right vertex set is $\{V_1, \ldots, V_n\}$.
- There is an edge (F_i, V_j) for every F_i and V_j such that $Vars(F_i) \cap V_j \neq \emptyset$.

The special clause set E is not part of this graph, but it will "filter" which truth value assignments to consider when playing the combinatorial games to be described next.[4]

Note that we do not discuss above how to partition the clauses or divide the variables—the $(\mathcal{F}, \mathcal{V})_E$-graph construction is well-defined for any partitions and divisions. Intuitively, though, the construction of this graph will be guided by an understanding of the combinatorial structure of the formula $\mathcal{F}$, and the goal is to construct a graph for which winning strategies can be found in the combinatorial games which we will discuss below.

3.2 The Resolution Edge Game and Width Lower Bounds

Let us now present a combinatorial game on $(\mathcal{F}, \mathcal{V})_E$-graphs that can be used to prove resolution lower bounds. The game is played by two players, whom we refer to as Adversary and Satisfier, and the intention is to design the game so that winning strategies for Satisfier correspond to resolution lower bounds.

Definition 3.2 (Resolution edge game). Given a CNF formula $\mathcal{F}$ over variables $\mathcal{V}$, let $(\mathcal{F}, \mathcal{V})_E$ be a bipartite graph as constructed above (with the clause set E being satisfiable), and suppose

[4] Jumping ahead a bit for expert readers, if we want to prove lower bounds for, e.g., pigeonhole principle formulas, then it is natural to choose the set E so that it is only satisfied by assignments corresponding to partial matchings of pigeons to holes.

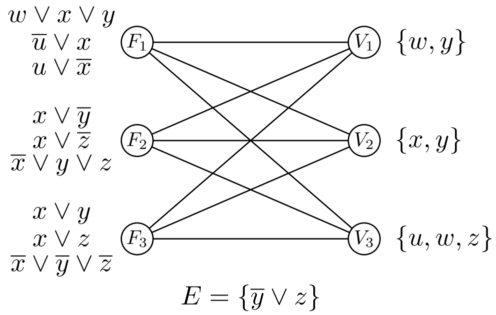


Fig. 1. Generalized incidence graph to illustrate the resolution edge game.

that (F_i, V_j) is an edge in this graph. Then the *resolution edge game on* (F_i, V_j) with respect to the *filtering set* E is the following game:

(1) Adversary provides a total assignment α such that $\alpha(E) = \top$.
(2) Satisfier chooses a partial assignment $\rho : V_j \longrightarrow \{\top, \bot\}$.
(3) Satisfier wins if substituting ρ for V_j in α yields an assignment $\alpha' = \alpha[\rho/V_j]$ such that $\alpha'(F_i \land E) = \top$.

We say that Satisfier *wins the game on* (F_i, V_j) if there is a strategy that always produces a winning assignment α' given any α satisfying E. Satisfier *wins the resolution edge game on* $(\mathcal{F}, \mathcal{V})_E$ if there is a winning strategy for all edges in the graph.

Let us make this more concrete by giving an example.

Example 3.3. Consider the toy formula

$$\begin{aligned} &(w \lor x \lor y) \land (\overline{u} \lor x) \land (u \lor \overline{x}) \land (x \lor \overline{y}) \land (x \lor \overline{z}) \land \\ &(\overline{x} \lor y \lor z) \land (x \lor y) \land (x \lor z) \land (\overline{x} \lor \overline{y} \lor \overline{z}) \land (\overline{y} \lor z) \end{aligned} \tag{3.1}$$

over variables $\{u, w, x, y, z\}$. Suppose we have the partition into clause sets

$$F_1 = \{w \lor x \lor y,\ \overline{u} \lor x,\ u \lor \overline{x}\} \tag{3.2a}$$
$$F_2 = \{x \lor \overline{y},\ x \lor \overline{z},\ \overline{x} \lor y \lor z\} \tag{3.2b}$$
$$F_3 = \{x \lor y,\ x \lor z,\ \overline{x} \lor \overline{y} \lor \overline{z}\} \tag{3.2c}$$
$$E = \{\overline{y} \lor z\} \tag{3.2d}$$

and the division into variable sets

$$V_1 = \{w, y\} \tag{3.3a}$$
$$V_2 = \{x, y\} \tag{3.3b}$$
$$V_3 = \{u, w, z\} \tag{3.3c}$$

with overlap $\ell = 2$. This yields the $(\mathcal{F}, \mathcal{V})_E$-graph in Figure 1. Let us analyse the resolution edge game played on three different edges:

(1) For the top-left-to-top-right edge (F_1, V_1), Adversary can play any total assignment α_1 extending $\rho_1 = \{u \mapsto \bot, x \mapsto \top, y \mapsto \bot\}$. Then Satisfier cannot win, since $\rho_1(u \lor \overline{x}) = \bot$ but the assignments to u and x cannot be changed (they are not in V_1). This shows that any graph construction yielding an edge (F_i, V_j) for which some clause $C \in F_i$ has no variables in V_j is a bad idea from Satisfier's point of view.

(2) A slightly more interesting example is the top-left-to-mid-right edge (F_1, V_2). Here Adversary can play any total assignment α_2 extending $\rho_2 = \{u \mapsto \bot, w \mapsto \bot, y \mapsto \bot, z \mapsto \bot\}$, again forcing a loss for Satisfier. To see this, note first that the variables u, w, and z cannot be flipped, since they are not in V_2. Furthermore, changing the assignment to $y \in V_2$ is not possible either, since this falsifies E. But this means that the only variable Satisfier can adjust is x, and this is not sufficient since $F_1{\restriction}_{\rho_2} = \{x, \overline{x}\}$.

(3) Finally, consider the mid-left-to-mid-right edge (F_2, V_2). For this edge Satisfier has a winning strategy. Given any assignment α_3 from Adversary such that $\alpha_3(E) = \top$, Satisfier can set $\alpha'(x) = \alpha_3(y \vee z)$ and leave the assignments to the other variables unchanged. The filtering set is still satisfied, since the assignments to y and z were not touched. Also, F_2 evaluates to true since these clauses encode $x \leftrightarrow (y \vee z)$, and x is assigned precisely so as to satisfy this. Hence, $\alpha'(F_2 \wedge E) = \top$ and Satisfier wins the game on (F_2, V_2).

A moment of reflection reveals that one way to make sure that Satisfier can always win the resolution edge game on $(\mathcal{F}, \mathcal{V})_E$ is to let this graph be the standard clause-variable incidence graph with each clause and variable forming its own vertex and with $E = \emptyset$. In order to prove resolution lower bounds, we will need not only that Satisfier has a winning strategy, however, but also that the $(\mathcal{F}, \mathcal{V})_E$-graph is an expander in the sense of Definition 3.1.

Definition 3.4 (Resolution expander). An $(\mathcal{F}, \mathcal{V})_E$-graph is an (s, δ, E)*-resolution expander* if:

- Satisfier wins the resolution edge game on $(\mathcal{F}, \mathcal{V})_E$.
- The $(\mathcal{F}, \mathcal{V})_E$-graph is a bipartite (s, δ)-boundary expander, i.e., for all left vertex subsets $\mathcal{F}' \subseteq \mathcal{F} \setminus \{E\}$ with $|\mathcal{F}'| \leq s$ it holds that $|\partial(\mathcal{F}')| \geq \delta|\mathcal{F}'|$.

(It is important to note here that $|\mathcal{F}'|$ measures the number of vertices, i.e., the *number of clause subsets*, and not the total number of clauses.)

We say that a CNF formula $\mathcal{F}$ over variables $\mathcal{V}$ *admits an* (s, δ, E)*-resolution expander* if there is a partition of the clauses $\mathcal{F} = E \cup \bigcup_{i=1}^{m} F_i$ and a division of the variables $\mathcal{V} = \bigcup_{j=1}^{n} V_j$ such that the resulting $(\mathcal{F}, \mathcal{V})_E$-graph is an (s, δ, E)-resolution expander.

Constructing resolution expanders is sufficient to prove resolution width lower bounds.

Theorem 3.5 (essentially [9]). *If a CNF formula $\mathcal{F}$ admits an (s, δ, E)-resolution expander with overlap ℓ, then any resolution refutation of $\mathcal{F}$ requires width larger than $\delta s/(2\ell)$.*

Let us start by outlining how the proof for Theorem 3.5 goes. Following [9], we prove the theorem by defining a "progress measure" $\mu : \{\text{clauses}\} \longrightarrow \mathbb{N}$ satisfying the following properties.

Property 3.6. *For any axiom clause $A \in \mathcal{F}$ it holds that $\mu(A) = \mathrm{O}(1)$.*

Property 3.7. *For any two clauses $C \vee x$ and $D \vee \overline{x}$ it holds for the resolvent $C \vee D$ that $\mu(C \vee D) \leq \mu(C \vee x) + \mu(D \vee \overline{x})$.*

Property 3.8. *For the empty clause $\bot$ it holds that $\mu(\bot) > s$.*

Properties 3.6–3.8 immediately imply the following claim (since the measure μ cannot more than double at every resolution step).

Claim 3.9. *In any resolution refutation of $\mathcal{F}$ there is some clause C with $\mu(C) \in (s/2, s]$.*

Theorem 3.5 then follows from a second claim saying that clauses with medium-large progress measure must have high width.

Claim 3.10. *Any clause C with $\mu(C) = \sigma \leq s$ has width at least $\geq \delta\sigma/\ell$.*

Now let us fill in the details in this outline. We say that a set of clauses F *implies* a clause D, denoted $F \vDash D$, if any truth value assignment that satisies F must also satisfy D. Given an $(\mathcal{F}, \mathcal{V})_E$-graph for $\mathcal{F}$, we define μ by

$$\mu(C) = \min\bigl\{|\mathcal{F}'| \,:\, \textstyle\bigwedge_{F \in \mathcal{F}'} F \wedge E \vDash C\bigr\} \tag{3.4}$$

(i.e., $\mu(C)$ is the smallest number of subsets of clauses on the left-hand side of the graph that together with the filtering set E imply C). We proceed to show how Properties 3.6–3.8 and Claim 3.10 follow from the assumption that $(\mathcal{F}, \mathcal{V})_E$ is an (s, δ, E)-resolution expander with overlap ℓ.

Consider any axiom clause $A \in \mathcal{F} = E \cup \bigcup_{i=1}^{m} F_i$. If $A \in E$, then $\mu(A) = 0$ since $E \vDash A$, and if $A \in F_i$ for some $F_i \in \mathcal{F} \setminus \{E\}$, then $\mu(A) = 1$ since $F_i \wedge E \vDash A$. Hence, $\mu(A) = \mathrm{O}(1)$ for all axioms, and Property 3.6 holds.

For Property 3.7, consider the resolvent $C \vee D$ of $C \vee x$ and $D \vee \overline{x}$. Fix minimal-size left vertex sets $\mathcal{F}_1$ and $\mathcal{F}_2$ such that $\bigwedge_{F \in \mathcal{F}_1} F \wedge E \vDash C \vee x$ and $\bigwedge_{F \in \mathcal{F}_2} F \wedge E \vDash D \vee \overline{x}$. Then it holds that $\bigwedge_{F \in \mathcal{F}_1 \cup \mathcal{F}_2} F \wedge E \vDash C \vee D$, so $\mu(C \vee D) \leq |\mathcal{F}_1 \cup \mathcal{F}_2| \leq |\mathcal{F}_1| + |\mathcal{F}_2| = \mu(C \vee x) + \mu(D \vee \overline{x})$.

Note that so far we did not use that $(\mathcal{F}, \mathcal{V})_E$ is an (s, δ, E)-resolution expander, but this will be needed to establish Property 3.8. Consider any $\mathcal{F}' \subseteq \mathcal{F} \setminus \{E\}$ such that $|\mathcal{F}'| = s$, and let us write $\mathcal{F}' = \{F_1, \ldots, F_s\}$ where F_i are left vertices in $(\mathcal{F}, \mathcal{V})_E$. We want to argue that $\bigwedge_{F_i \in \mathcal{F}'} F_i \wedge E \nvDash \bot$, which shows that we must have $\mu(\bot) > s$ as desired.

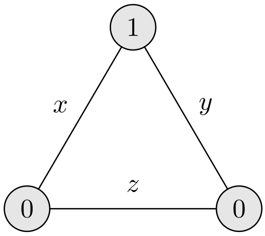
By the expansion properties of $(\mathcal{F}, \mathcal{V})_E$ it holds that $|\partial(\mathcal{F}')| \geq \delta|\mathcal{F}'| > 0$. This means that the left vertex set $\mathcal{F}'$ has a unique neighbour, or, in other words, that there is a left vertex set F_s and a right vertex set V_s (possibly after relabelling) such that $V_s \in N(F_s) \setminus N\bigl(\bigcup_{j=1}^{s-1} F_j\bigr)$. By the same argument, we can find $F_{s-1} \in \mathcal{F}' \setminus \{F_s\}$ and $V_{s-1} \in N(F_{s-1}) \setminus N\bigl(\bigcup_{j=1}^{s-2} F_j\bigr)$. We can repeat this reasoning inductively—this is sometimes referred to as a *peeling argument*—to find a matching $F_1 \leftrightarrow V_1, F_2 \leftrightarrow V_2, \ldots, F_s \leftrightarrow V_s$ such that $V_i \in N(F_i) \setminus N\bigl(\bigcup_{j=1}^{i-1} F_j\bigr)$ for all $i = 1, \ldots, s$ (i.e., V_i is not a neighbour of any F_j, $j < i$).

Take any total truth value assignment α such that $\alpha(E) = 1$. Note that such an assignment exists, since E should be satisfiable by definition. Since Satisfier wins the resolution edge game on (F_1, V_1), there is an assignment α_1 such that $\alpha_1(F_1 \cup E) = 1$. Satisfier also wins the game on (F_2, V_2), and so α_1 can be modified on V_2 to obtain an assignment α_2 such that $\alpha_2(F_1 \cup F_2 \cup E) = 1$—here we use that $\mathit{Vars}(F_1) \cap V_2 = \emptyset$ since there is no edge (F_1, V_2), and so the modification of α_1 to α_2 does not change the fact that F_1 is satisfied. Continuing in this way, Satisfier can play the edge game on (F_i, V_i) for $i = 3, 4 \ldots, s$, in each step modifying α_{i-1} on V_i to obtain α_i that satisfies both F_i and $\bigwedge_{j<i} F_j \wedge E$. This yields an assignment α_s such that $\alpha_s\bigl(\bigwedge_{F_i \in \mathcal{F}'} F_i \wedge E\bigr) = 1$, and so Property 3.8 holds.

It remains to prove Claim 3.10, i.e., that if C is a clause with $\mu(C) = \sigma \leq s$, then C has width at least $\delta\sigma/\ell$. Towards this end, fix some left vertex set $\mathcal{F}_C \subseteq \mathcal{F} \setminus \{E\}$ witnessing that $\mu(C) = \sigma$. Then $|\mathcal{F}_C| = \sigma$ and $\bigwedge_{F \in \mathcal{F}_C} F \wedge E \vDash C$, but for all $\mathcal{F}' \subsetneq \mathcal{F}_C$ it holds that $\bigwedge_{F \in \mathcal{F}'} F \wedge E \nvDash C$ (where again the subset relation is with respect to the clause sets forming the left-hand side of the $(\mathcal{F}, \mathcal{V})_E$-graph). Claim 3.10 now follows from another, final claim.

Claim 3.11. *For all $V \in \partial(\mathcal{F}_C)$ it holds that $V \cap \mathit{Vars}(C) \neq \emptyset$.*

Since every variable occurs in at most ℓ sets $V \in \mathcal{V}$, Claim 3.11 implies that C has width at least $|\partial(\mathcal{F}_C)|/\ell \geq \delta|\mathcal{F}_C|/\ell = \delta\sigma/\ell$, which is what we want to prove. To establish Claim 3.11, we again appeal to the resolution edge game. Suppose towards contradiction that there exists a variable set $V \in \partial(\mathcal{F}_C)$ such that $V \cap \mathit{Vars}(C) = \emptyset$, and let $F_V \in \mathcal{F}_C$ be the unique neighbour of V. By the minimality of $\mathcal{F}_C$ we have $\bigwedge_{F \in \mathcal{F}_C \setminus \{F_V\}} F \wedge E \nvDash C$, or, expressed differently, there exists an assignment α such that $\alpha\bigl(\bigwedge_{F \in \mathcal{F}_C \setminus \{F_V\}} F \wedge E\bigr) = 1$ but $\alpha(C) = 0$. Use Satisfier's strategy for the edge game on (F_V, V) to modify α locally on V into an assignment α' such that $\alpha'(F_V \wedge E) = 1$. Since no variable in V appears in $\mathcal{F}_C \setminus \{F_V\}$ or C we have $\alpha'(\mathcal{F}_C \setminus \{F_V\}) = \alpha(\mathcal{F}_C \setminus \{F_V\}) = 1$ and



(a) Triangle graph with odd-weight labelling.

$$\begin{aligned} & (x \vee y) \\ \wedge\ & (\overline{x} \vee \overline{y}) \\ \wedge\ & (x \vee \overline{z}) \\ \wedge\ & (\overline{x} \vee z) \\ \wedge\ & (y \vee \overline{z}) \\ \wedge\ & (\overline{y} \vee z) \end{aligned}$$

(b) Corresponding Tseitin formula.

Fig. 2. Example Tseitin formula.

$\alpha'(C) = \alpha(C) = 0$. But this contradicts that $\mathcal{F}_C$ was chosen so that $\bigwedge_{F \in \mathcal{F}_C} F \wedge E \vDash C$. This concludes the proof, and Theorem 3.5 follows.

3.3 Two Applications of the Resolution Edge Game: Tseitin and Onto FPHP Formulas

We next show how Theorem 3.5 can be used to reprove some classic lower bounds for resolution. We employ the theorem to obtain lower bounds on width, after which the analogue of Theorem 2.3 for resolution from [9] can be used to turn these width lower bounds into size lower bounds.

Let us first consider *Tseitin formulas*, which provide a way of encoding (the negation of) the principle that "the sum of the vertex degrees in a graph is even." Let $G = (V, E)$ be a connected undirected graph of size $|V| = n$ and let $\chi : V \longrightarrow \{0, 1\}$ be a function labelling vertices in G by 0 or 1. Identify every edge $e \in E$ with a Boolean variable, and for every vertex $v \in V$ let $\mathit{PARITY}_{v,\chi}$ be the set of clauses encoding

$$\bigoplus_{e \ni v} e \equiv \chi(v) \pmod 2, \tag{3.5}$$

i.e., that the parity of the number of edges incident to v assigned to true should be equal to the vertex label $\chi(v)$, in the natural way by adding one clause for every assignment of the wrong parity ruling out that assignment. Then the *Tseitin formula* $\mathit{Ts}(G, \chi)$ is defined to be the CNF formula

$$\mathit{Ts}(G, \chi) = \bigwedge_{v \in V} \mathit{PARITY}_{v,\chi} . \tag{3.6}$$

See Figure 2 for an example, where Figure 2b displays the formula corresponding to the labelled graph in Figure 2a.

If the maximal vertex degree of G is d, then $\mathit{Ts}(G, \chi)$ is a d-CNF formula with at most $nd/2$ variables and at most $n \cdot 2^{d-1}$ clauses. Let us say that $\chi : V \longrightarrow \{0, 1\}$ has *odd weight* if $\sum_{v \in V} \chi(v) \equiv 1 \pmod 2$. Then it is not hard to verify that if G is a connected graph, it holds that $\mathit{Ts}(G, \chi)$ is unsatisfiable if and only if χ has odd weight. (For the if direction, which is the one we will need here, note that if we sum up all parity constraints (3.5), then the right-hand side is odd by assumption, but the left-hand side is even since every edge is counted twice.)

We can prove resolution width lower bounds for Tseitin formulas using the following definition of expansion.[5]

[5] We note that our definition of edge expanders is slightly non-standard in that one usually fixes $s = |V|/2$, meaning that the expansion factor δ will be the *isoperimetric number*

$$h(G) = \min \left\{ \frac{|E(V', V(G) \setminus V')|}{|V'|} \ : \ V' \subset V(G),\ |V'| \leq \left|V(G) \setminus V'\right| \right\} .$$

However, we obtain a more general result by considering the more relaxed notion of edge expansion in Definition 3.12.

Definition 3.12 (Edge expander). Let us say that a graph $G = (V, E)$ is an (s, δ)-*edge expander* if for every subset of vertices $V' \subseteq V(G)$ of size $|V'| \leq s$ it holds that $|E(V', V \setminus V')| \geq \delta|V'|$, where $E(U, W) = \{(u, w) \mid u \in U,\ w \in W\}$ denotes the set of edges between U and W.

Theorem 3.13 ([9, 46]). *If G is an (s, δ)-edge expander and χ has odd weight, then any resolution refutation of $Ts(G, \chi)$ requires width larger than $\delta s/2$.*

Proof sketch. We build an $(\mathcal{F}, \mathcal{V})_E$-graph for $Ts(G, \chi)$ in the following way:

- The left clause sets $F_v = PARITY_{v,\chi}$ consist of the clauses encoding the parity constraint for all vertices v.
- Every edge e forms a singleton right-hand set $V_e = \{e\}$ (so the overlap is $\ell = 1$).
- The filtering set $E = \emptyset$ is empty.

It is straightforward to verify that if G is an (s, δ)-edge expander, then $(\mathcal{F}, \mathcal{V})_E$ is a resolution expander with the same parameters. The edge expansion of G corresponds exactly to the boundary expansion of $(\mathcal{F}, \mathcal{V})_E$, and Satisfier can always win the resolution edge game on any (F_v, V_e) by flipping the value of the edge e incident to v to satisfy the parity constraint $F_v = PARITY_{v,\chi}$. Now the resolution width lower bound follows from an appeal to Theorem 3.5. □

Let us next consider *pigeonhole principle formulas.* In the interest of generality, and to be able to apply the tools developed in this section, we define these formulas over bounded-degree bipartite graphs $G = (U \mathbin{\dot\cup} V, E)$, meaning that the set of variables is $\{x_{u,v} \mid u \in U,\ v \in V,\ (u, v) \in E\}$. The axiom clauses appearing in different versions of PHP formulas are as follows:

$$\bigvee_{v \in N(u)} x_{u,v} \qquad u \in U \qquad \text{(pigeon axioms)} \tag{3.7a}$$

$$\overline{x}_{u,v} \vee \overline{x}_{u',v} \qquad v \in V,\ u, u' \in N(v),\ u \neq u', \qquad \text{(hole axioms)} \tag{3.7b}$$

$$\overline{x}_{u,v} \vee \overline{x}_{u,v'} \qquad u \in U,\ v, v' \in N(u),\ v \neq v' \qquad \text{(functionality axioms)} \tag{3.7c}$$

$$\bigvee_{u \in N(v)} x_{u,v} \qquad v \in V \qquad \text{(onto axioms)} \tag{3.7d}$$

The "plain vanilla" *graph pigeonhole principle formula $PHP(G)$* consists of clauses (3.7a) and (3.7b); the *graph functional pigeonhole principle formula $FPHP(G)$* contains the clauses of $PHP(G)$ and in addition clauses (3.7c); the *graph onto pigeonhole principle formula $Onto$-$PHP(G)$* contains $PHP(G)$ plus clauses (3.7d); and the *graph onto functional pigeonhole principle formula $Onto$-$FPHP(G)$*, finally, consists of all the clauses (3.7a)–(3.7d).

We obtain the standard versions of the PHP formulas by considering graph formulas as above over the complete bipartite graph $K_{n+1,n}$. In the opposite direction, for any bipartite graph G with $n + 1$ vertices on the left and n vertices on the right we can hit any version of the pigeonhole principle formula over $K_{n+1,n}$ with the restriction ρ_G setting $x_{u,v}$ to false for all $(u, v) \notin E(G)$ to recover the corresponding graph pigeonhole principle formula over G. When doing so, we can use the observation from Section 2 that restricting a formula can only decrease the size, width, or degree required to refute it.

For resolution, we can obtain lower bounds for all versions of PHP formulas simultaneously by considering onto-FPHP formulas. Since this is the most constrained version, any lower bound will also apply to other flavours of PHP formulas containing less axiom clauses. We can prove such lower bounds if G is an expander in the sense of Definition 3.1 such that there is a maximum matching from the left-hand side to the right-hand side. The following theorem can be obtained by applying the techniques in [9], but a similar result was also stated explicitly in [28].

Theorem 3.14. *If $G = (U \dot\cup V, E)$ is a bipartite (s, δ)-boundary expander such that there is a full matching of V into U,*[6] *then any resolution refutation of Onto-FPHP(G) requires width larger than $\delta s/2$.*

Proof. Construct an $(\mathcal{F}, \mathcal{V})_E$-graph for *Onto-FPHP*$(G)$ as follows:

- The left clause sets F_u are singleton sets with axioms (3.7a) for each pigeon u.
- The right clause sets $V_v = \{x_{u,v} \mid (u, v) \in E\}$ consist of all variables $x_{u,v}$ mentioning hole v (so again the overlap is $\ell = 1$).
- The filtering set E contains all hole axioms (3.7b), functional axioms (3.7c), and onto axioms (3.7d).

We claim that if G is an (s, δ)-boundary expander, then $(\mathcal{F}, \mathcal{V})_E$ is a resolution expander with the same parameters. The expansion part is clear—$(\mathcal{F}, \mathcal{V})_E$ is isomorphic to G by construction—but we need to argue that Satisfier wins the resolution edge game on $(\mathcal{F}, \mathcal{V})_E$.

Suppose that Adversary and Satisfier play the game on any edge (F_u, V_v). Then α has to correspond to a full matching of V into U, since the filtering set E is satisfied precisely for such assignments (which exist by our assumptions). If $\alpha(F_u) = \top$, then Satisfier is already winning and does not need to do anything. Otherwise, Satisfier sets $\alpha'(x_{u,v}) = \top$ and $\alpha'(x_{u',v}) = \bot$ for all $u' \neq u$ such that $(u', v) \in E$. Then $\alpha'(F_u \wedge E) = \top$, since α' just encodes a new full matching of V into U that includes pigeon u. Hence, Satisfier has a winning strategy, and the resolution width lower bound follows from Theorem 3.5. □

3.4 The Polynomial Calculus Edge Game and Degree Lower Bounds

The resolution edge game provides a unified framework in which many resolution lower bounds can be presented, but it is clear that we cannot hope to use it to obtain lower bounds on polynomial calculus degree. It is not hard to show that Tseitin formulas are easy over fields of characteristic 2, and as observed in [43] onto-FPHP formulas with $n + 1$ pigeons and n holes are easy to refute in any field. To get polynomial calculus degree lower bounds, we need to find winning strategies for the more challenging game presented next, where Satisfier has to choose an assignment first and Adversary can adapt to this choice.

Definition 3.15 (Polynomial calculus edge game). Given an $(\mathcal{F}, \mathcal{V})_E$-graph for the CNF formula $\mathcal{F}$ over variables $\mathcal{V}$ (with the clause set E being satisfiable), the *polynomial calculus (PC) edge game* on an edge (F_i, V_j) in $(\mathcal{F}, \mathcal{V})_E$ with respect to the *filtering set E* is the following game:

(1) Satisfier commits to a partial assignment $\rho : V_j \longrightarrow \{\top, \bot\}$ satisfying any clauses touched in E (i.e., $\rho(C) = \top$ for all clauses $C \in E$ with $V_j \cap \mathit{Vars}(C) \neq \emptyset$).
(2) Adversary provides a total assignment α such that $\alpha(E) = \top$.
(3) Satisfier wins if substituting ρ for V_j in α yields an assignment $\alpha' = \alpha[\rho/V_j]$ such that $\alpha'(F_i \wedge E) = \top$.

We say that Satisfier *wins the PC edge game on* $(\mathcal{F}, \mathcal{V})_E$ if there is a winning strategy for all edges in the graph.

Example 3.16. Let us return to the $(\mathcal{F}, \mathcal{V})_E$-graph in Figure 1. Recall that, as discussed in Example 3.3, in the resolution edge game Satisfier loses on edges (F_1, V_1) and (F_1, V_2) but wins on the edge (F_2, V_2).

In the harder PC edge game on (F_2, V_2) Satisfier loses. The filtering set $E = \{\overline{y} \vee z\}$ needs $\rho(y) = \bot$, but $F_2{\restriction}_{\{y=\bot\}} = \{x \vee \overline{z},\ \overline{x} \vee z\}$ requires that x and z should take the same value. Therefore, since

[6] This condition is not needed for PHP formulas without onto axioms, and the assumption can be weakened for onto formulas into requiring that the graph is expanding from the right to the left, but for simplicity we make the stronger requirement of a full matching in our statement of the lower bound.

Satisfier has to commit to ρ first, Adversary can choose $\alpha(z) = \neg\rho(x)$ to force a loss for Satisfier. Before, in the resolution edge game, Adversary had to satisfy the filtering set, and if y was assigned to false Satisfier could respond by setting x to the same value as z. But in the PC edge game Satisfier has to commit to ρ first, which means that there is no winning strategy.

On the edge (F_3, V_2) there is a winning strategy for Satisfier, however. If Satisfier chooses $\rho = \{x \mapsto \top, y \mapsto \bot\}$, then we have $\rho(F_3) = \rho(E) = \top$, which means that no matter what assignments Adversary makes to other variables, once the assignments from ρ are substituted for x and y we have that F_3 is satisfied and that every clause touched in E is also satisfied. (In this small example, E just consists of a single clause, but if there were other clauses in E, then every clause should either be satisfied by ρ or should contain no variable in V_2.)

Given the game for polynomial calculus in Definition 3.15, we can define PC expanders analogously to resolution expanders in Definition 3.4 and obtain degree lower bounds as a consequence of such constructions just as for resolution in Theorem 3.5. We do so in Definition 3.17 and Theorem 3.18 below, which are an attempt at summarizing the main technical contributions of this paper in as friendly a language as possible.

Definition 3.17 (Polynomial calculus expander). An $(\mathcal{F}, \mathcal{V})_E$-graph is an (s, δ, E)-*polynomial calculus expander* (or *PC expander* for short) if:

- Satisfier wins the polynomial calculus edge game on $(\mathcal{F}, \mathcal{V})_E$.
- The $(\mathcal{F}, \mathcal{V})_E$-graph is a bipartite (s, δ)-boundary expander, i.e., for all left vertex subsets $\mathcal{F}' \subseteq \mathcal{F} \setminus \{E\}$ with $|\mathcal{F}'| \leq s$ it holds that $|\partial(\mathcal{F}')| \geq \delta|\mathcal{F}'|$.

(Above, $|\mathcal{F}'|$ measures the number of vertices, i.e., the *number of clause subsets*, and not the total number of clauses.)

We say that a CNF formula $\mathcal{F}$ over variables $\mathcal{V}$ *admits an* (s, δ, E)*-PC expander* if there is a partition of the clauses $\mathcal{F} = E \cup \bigcup_{i=1}^{m} F_i$ and a division of the variables $\mathcal{V} = \bigcup_{j=1}^{n} V_j$ such that the resulting $(\mathcal{F}, \mathcal{V})_E$-graph is an (s, δ, E)-PC expander.

Theorem 3.18. *If a CNF formula $\mathcal{F}$ admits an (s, δ, E)-PC expander with overlap ℓ, then any polynomial calculus refutation of $\mathcal{F}$ over any field requires degree larger than $\delta s/(2\ell)$.*

We want to emphasize that the change of order of the players in the game in Definition 3.15 compared to Definition 3.2, which is the only difference between these two games, is also what is absolutely crucial to obtain Theorem 3.18. For the Tseitin formulas discussed above, it is clear that Satisfier can win the edge game if Adversary has to go first, since then the assignment to the edge can easily be flipped to satisfy the relevant parity constraint. It is equally clear that this is hopeless if Satisfier has to assign the edge first and Adversary can assign the rest of the edges afterwards, because then Adversary can make sure that the parity is violated. The fact that Satisfier can win when going second, but not when going first, is a way of explaining why Tseitin formulas are hard for resolution but not necessarily for polynomial calculus.

Another remark of a more technical nature regarding Theorem 3.18 is that the lower bounds hold for polynomial calculus over any field. This is both a strength and a weakness. Generalizing the example of Tseitin formulas, it is known that polynomial calculus over a field of characteristic p cannot efficiently refute contradictory CNF formulas based on counting modulo q for primes $p \neq q$ [12]. There is no way we can use the techniques developed in this paper to prove such results, however, since these lower bounds depend on the field.

The proof of Theorem 3.18 is by a careful adaptation of [2], which we present in Section 4. We remark that there is no reason to require that the input should be polynomials obtained from translations of clauses of a CNF formula—the theorem statement can be generalized to apply to arbitrary sets of polynomials.

As we shall see in Section 5, Theorem 3.18 provides a common framework for presenting previous PC degree lower bounds for random k-CNF formulas [2, 8] (and any CNF formulas with expanding clause-variable incidence graphs), "vanilla" PHP formulas [2], ordering principle formulas [22], and subset cardinality formulas [33]. More importantly, it will also allow us to prove a new lower bound for CNF encodings of the functional pigeonhole principle.

4 A Generalization of the Alekhnovich–Razborov Method

We now proceed to describe a generalized constraint-variable incidence graph for polynomials and show how polynomial calculus degree lower bounds can be obtained from such graph constructions. Before doing so, let us recall that in what follows we only study degree in multilinear PC as explained in Section 2.2 after Theorem 2.3. When specializing the discussion to CNF formulas $\mathcal{F}$ we will identify $\mathcal{F}$ with the set of polynomials obtained by the canonical translation (2.2) of the clauses $C \in \mathcal{F}$, and will overload C to denote also the polynomial $p_{\mathcal{PC}}(C)$ when no confusion can arise. In the rest of this paper we will switch freely between these different perspectives.

We say that a (partial or total) assignment ρ, which is always assumed to be a $\{0, 1\}$-assignment even when we are operating in a larger field, *satisfies* a (multilinear) polynomial q if q vanishes under ρ (possibly after cancellation of like terms if ρ is a partial assignment). Note that this is equivalent to the condition that any total assignment ρ' consistent with ρ makes the polynomial q evaluate to 0. An assignment satisfies a set of polynomials Q if it satisfies all polynomials $q \in Q$, and similarly an assignment satisfies a family of sets of polynomials $\mathcal{U}$ if it satisfies all sets of polynomials in $\mathcal{U}$.

4.1 A Generalized Constraint-Variable Incidence Graph

The key to our construction of generalized constraint-variable incidence graphs is to keep track of how individual polynomials in a set of polynomials are affected by partial assignments. In what follows, the reader can think of P and Q as satisfiable subsets of polynomials, since this is the scenario in which the definitions below will be relevant.

Definition 4.1 (Respectful and semirespectful variable sets). We say that a partial assignment ρ *respects* a set of polynomials Q, or that ρ is *Q-respectful*, if for every polynomial $q \in Q$ either $\mathit{Vars}(q) \cap \mathrm{dom}(\rho) = \emptyset$ or ρ satisfies q, and furthermore Q does not contain any constant (i.e., degree-0) polynomial.[7]

A set of variables V respects a set of polynomials Q if there exists an assignment ρ with $\mathrm{dom}(\rho) = V$ that respects Q.

A set of variables V *semirespects* a set of polynomials Q, or is *Q-semirespectful*, if for any assignment σ to $\mathit{Vars}(Q) \setminus V$ there exists an assignment ρ to V such that ρ satisfies $Q\restriction_{\sigma}$.

Example 4.2. Consider the CNF formula $(x_1 \lor x_2) \land (\overline{x}_1 \lor x_3) \land (x_1 \lor x_4) \land (\overline{x}_1 \lor x_5)$ translated to the sets of polynomials $Q = \{x_1x_2, x_3 - x_1x_3, x_1x_4, x_5 - x_1x_5\}$ together with the subsets of variables $V_1 = \{x_1, x_2, x_3\}$ and $V_2 = \{x_4, x_5\}$. The assignment ρ_2 to V_2 setting $x_4 = x_5 = \top = 0$ respects Q since it satisfies the polynomials/clauses containing these variables, and hence V_2 is Q-respectful. However, V_1 is not Q-respectful since assigning x_1 will affect all polynomials in Q but cannot satisfy both x_1x_4 and $x_5 - x_1x_5$.

For the set of polynomials $Q' = \{x_1x_2, 1 - x_1 - x_2 + x_1x_2\}$ obtained by translating the CNF formula $(x_1 \lor x_2) \land (\overline{x}_1 \lor \overline{x}_2)$ it holds that both of the singleton variable sets $V_1 = \{x_1\}$ and $V_2 = \{x_2\}$ are Q'-semirespectful (as witnessed by setting $\rho(x_i) = \neg\sigma(x_{3-i}) = 1 - \sigma(x_{3-i})$ in Definition 4.1) but they are not Q'-respectful.

[7] This technical condition is just to avoid the uninteresting pathological case when Q already contains a trivially unsatisfiable polynomial—for such inputs we cannot expect to obtain any lower bounds anyway.

Definition 4.3 (Respectful and semirespectful satisfaction). Let P and Q be sets of polynomials and let V be a set of variables.

We say that P is *Q-respectfully satisfiable by V* if V respects Q and if this is shown by a Q-respectful partial assignment ρ with $\mathrm{dom}(\rho) = V$ that satisfies all polynomials in P. Such an assignment ρ is said to *Q-respectfully satisfy P*.

We say that P is *Q-semirespectfully satisfiable by V* if V semirespects $P \cup Q$.

We want to point out that for the special case of CNF formulas Definitions 4.1 and 4.3 have close connections to the concept of *autarkies* [29]. Namely, when P and Q are CNF formulas, what Definition 4.1 says is that the assignment ρ is Q-respectful precisely when it is an autarky for Q, meaning that ρ satisfies all clauses in Q which it touches, or equivalently that $Q{\restriction_\rho} \subseteq Q$ holds (after the satisfied clauses in $Q{\restriction_\rho}$ have been removed). Definition 4.3 says that ρ Q-respectfully satisfies P if, in addition to being an autarky for Q, it is also a satifying assignment for P.

To build a bipartite graph representing the set of polynomials $\mathcal{P}$, we will group the polynomials into (satisfiable) subsets. In what follows, we will let $\mathcal{U} = \{P_1, \ldots, P_m\}$ denote a collection of subsets of polynomials in $\mathcal{P}$, where each subset P_i corresponds to one vertex on the left-hand side of the graph. There will also be a set $Q = \mathcal{P} \setminus \bigcup_{i=1}^{m} P_i$ which is not represented as a vertex in the graph, but which will be used to enforce respectful or semirespectful satisfaction. Jumping ahead a bit, semirespectfulness will be relevant for resolution lower bounds whereas the stronger notion of respectfulness is what is needed for polynomial calculus. On the right-hand side, the variables of $\mathcal{P}$ will be divided into a family $\mathcal{V} = \{V_1, \ldots, V_n\}$ of subsets of variables, each V_j corresponding to a vertex. In our definition, $\mathcal{U}$ and $\mathcal{V}$ do not need to be partitions of polynomials and variables in $\mathcal{P}$, respectively. This is not too relevant for $\mathcal{U}$ because we will always construct it as a partition, but it turns out to be critical for some applications to have sets in $\mathcal{V}$ share variables. The next definition describes the bipartite graph that we build and distinguishes between different types of neighbour relations in this graph.

Definition 4.4 (Bipartite $(\mathcal{U}, \mathcal{V})_Q$-graph). Let Q be a set of polynomials, $\mathcal{U}$ be a family of sets of polynomials, and $\mathcal{V}$ be a family of sets of variables such that $\mathit{Vars}\big(\bigcup_{P \in \mathcal{U}} P \cup Q\big) = \bigcup_{V \in \mathcal{V}} V$. Then the *(bipartite) $(\mathcal{U}, \mathcal{V})_Q$-graph* is the bipartite graph with left vertices $P \in \mathcal{U}$, right vertices $V \in \mathcal{V}$, and edges $\{(P, V) \mid \mathit{Vars}(P) \cap V \neq \emptyset\}$. We say that the $(\mathcal{U}, \mathcal{V})_Q$-graph *represents* the set of polynomials $\mathcal{P} = \bigcup_{P \in \mathcal{U}} P \cup Q$ over the variables $\bigcup_{V \in \mathcal{V}} V$.

Suppose that (P, V) is an edge in the $(\mathcal{U}, \mathcal{V})_Q$-graph. Then we say that P and V are *Q-respectful neighbours* if P is Q-respectfully satisfiable by V and *Q-non-respectful neighbours* otherwise. Similarly, we say that P and V are *Q-semirespectful neighbours* if P is Q-semirespectfully satisfiable by V and *Q-non-semirespectful neighbours* otherwise. When P and V are (semi)respectful neighbours we also say that (P, V) is a *(semi)respectful edge.*

Example 4.5. For readers who wish to make the connection back to Section 3, we can again consider the toy formula in (3.1) with partition of clauses and division of variables as in (3.2a)–(3.2d) and (3.3a)–(3.3c). If we rewrite the clauses as polynomials, then the graph in Figure 1 turns into that in Figure 3. This $(\mathcal{F}, \mathcal{V})_E$-graph represents the set of polynomials in the sense of Definition 4.4.

Then the resolution edge game in Definition 3.2 is just another way to define semirespectful satisfaction, and respectful satisfaction corresponds to the PC edge game in Definition 3.15.[8] Thus, as shown in Example 3.3 we have that F_1 and V_1 are E-non-semirespectful neighbours, as are F_1 and V_2. The vertices F_2 and V_2 are E-semirespectful neighbours, but not E-respectful neighbours, as discussed in Example 3.16, whereas F_3 and V_2 are E-respectful neighbours.

[8] In fact, the correspondence between respectful satisfaction and the PC edge game is not perfect in that the definition of the game is slightly more relaxed. We will comment briefly on this difference and how to deal with it in the proofs of Theorem 4.11 and Lemma 4.30 below in order to establish Theorem 3.18.

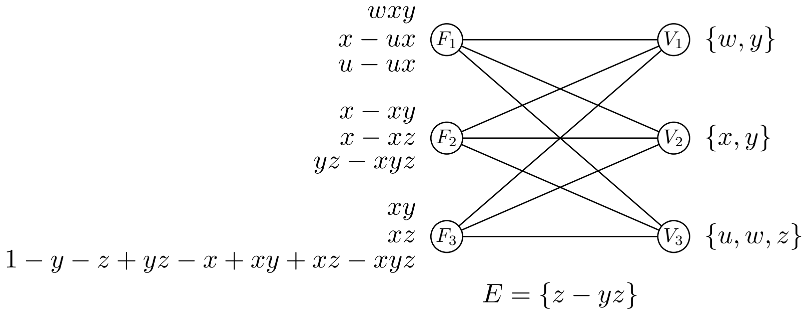


Fig. 3. Generalized incidence graph from Figure 1 with constraints written as polynomials.

Note that a set of polynomials $\mathcal{P}$ over variables $\mathcal{V}$ can be represented in many different ways by $(\mathcal{U}, \mathcal{V})_Q$-graphs depending on which subset structure is imposed on $\mathcal{P}$ and $\mathcal{V}$. Assuming that the structuring into subsets is clear from context, we will often write $(\mathcal{U}, \mathcal{V})_Q$ as a shorthand for the graph defined by $\mathcal{U}$, $\mathcal{V}$, and Q as above. When we write $\mathcal{U}' \subseteq \mathcal{U}$ or $\mathcal{V}' \subseteq \mathcal{V}$ we always view $\mathcal{U}'$ and $\mathcal{V}'$ as collections of subsets that respect this structuring, and $|\mathcal{U}'|$ and $|\mathcal{V}'|$ counts the number of subsets (not the number of individual polynomials or variables overall). We will also use standard graph notation and write $N(P)$ to denote the set of all neighbours $V \in \mathcal{V}$ of a vertex/set of polynomials $P \in \mathcal{U}$. It is important to observe that the fact that P and V are Q-respectful or Q-semirespectful neighbours might be witnessed by an assignment that falsifies some other set of polynomials $P' \in \mathcal{U} \setminus \{P\}$.

We remark that for a CNF formula $\mathcal{F}$ the concept of $(\mathcal{U}, \mathcal{V})_Q$-graphs is a fairly natural extension of the clause-variable incidence graph $G(\mathcal{F})$. The difference is that we throw out a part of $\mathcal{F}$, which we denote Q, and consider $G(\mathcal{F} \setminus Q)$. When the remaining clauses and variables are clustered into sets $\mathcal{U}$ and $\mathcal{V}$, there will be an edge in the $(\mathcal{U}, \mathcal{V})_Q$-graph between two clusters precisely when there is some edge between a pair of elements in these clusters. The only additional information we need to keep track of is which polynomial and variable clusters are (semi)respectful neighbours or not. In particular, for subsets $\mathcal{U}'$ of vertices on the left we will be interested in the unique neighbours on the right that are also (semi)respectful.

Definition 4.6 (Respectful and semirespectful boundary). Recall that for a bipartite graph $\mathcal{G} = (\mathcal{U} \mathbin{\dot\cup} \mathcal{V}, \mathcal{E})$ and a left vertex subset $\mathcal{U}' \subseteq \mathcal{U}$, the *boundary* $\partial(\mathcal{U}') \subseteq \mathcal{V}$ is the largest right vertex subset such that every $V \in \partial(\mathcal{U}')$ is a neighbour of exactly one $U \in \mathcal{U}'$.

The *Q-respectful boundary* $\partial_Q(\mathcal{U}')$ of $\mathcal{U}'$ consists of all $V \in \partial(\mathcal{U}')$ such that the unique edge to V emanating from $\mathcal{U}'$ is respectful. We obtain the *Q-semirespectful boundary* $\partial_Q^{\mathsf{SR}}(\mathcal{U}')$ by instead taking all $V \in \partial(\mathcal{U}')$ such that the unique edge to V emanating from $\mathcal{U}'$ is semirespectful.

Note that if $P_1, P_2 \in \mathcal{U}'$ are such that $V \in N(P_1) \cap N(P_2)$ where P_1 and V are Q-respectful neighbours but P_2 and V are Q-non-respectful neighbours, then V is *not* in the Q-respectful boundary of $\mathcal{U}'$ since it is a neighbour of two vertices in $\mathcal{U}'$. Thus, Q-non-respectful neighbours can never contribute positively to the respectful boundary but can shrink it.

The following easy observation will be helpful for us later.

Observation 4.7. *For any $(\mathcal{U}, \mathcal{V})_Q$-graph and any subsets $\mathcal{U}_1 \subseteq \mathcal{U}_2 \subseteq \mathcal{U}$, it always holds that $\partial_Q(\mathcal{U}_2) \cap N(\mathcal{U}_1) \subseteq \partial_Q(\mathcal{U}_1)$ and $\partial_Q^{\mathsf{SR}}(\mathcal{U}_2) \cap N(\mathcal{U}_1) \subseteq \partial_Q^{\mathsf{SR}}(\mathcal{U}_1)$.*

Proof. Clearly, for any subsets $\mathcal{U}_1 \subseteq \mathcal{U}_2 \subseteq \mathcal{U}$, it holds that $\partial(\mathcal{U}_2) \cap N(\mathcal{U}_1) \subseteq \partial(\mathcal{U}_1)$. Now consider any $V \in \partial(\mathcal{U}_1) \cap \partial(\mathcal{U}_2)$ with neighbour $P \in \mathcal{U}_1$. The question of whether P and V are (semi)respectful neighbours depends only on P, V, and Q, and not on $\mathcal{U}_1$ or $\mathcal{U}_2$, and hence the same subset containment holds for (semi)respectful boundary. □

The main technical result in this paper is that a set of polynomials $\mathcal{P}$ over variables $\mathcal{V}$ is hard for polynomial calculus with respect to degree if the polynomials can be partitioned as $\mathcal{P} = \bigcup_{P \in \mathcal{U}} P \cup Q$ and the variable set can be written as a union (but not necessarily a partition) $\mathcal{V} = \bigcup_{V \in \mathcal{V}} V$ in such a way that the resulting $(\mathcal{U}, \mathcal{V})_Q$-graph has *respectful boundary expansion* as defined next. We will also show that a weaker condition of *semirespectful boundary expansion* for the $(\mathcal{U}, \mathcal{V})_Q$-graph is sufficient to guarantee resolution width lower bounds for CNF formulas.

Definition 4.8 (Respectful and semirespectful boundary expanders). Let $\delta > 0$ and $\xi \geq 0$ be constants. A $(\mathcal{U}, \mathcal{V})_Q$-graph is said to be an (s, δ, ξ, Q)*-respectful boundary expander*, or just an (s, δ, ξ, Q)*-expander* for brevity, if for every left vertex set $\mathcal{U}' \subseteq \mathcal{U}$ of size $|\mathcal{U}'| \leq s$ it holds that $|\partial_Q(\mathcal{U}')| \geq \delta|\mathcal{U}'| - \xi$, and if in addition the technical condition holds that every $V \in \mathcal{V}$ is Q-respectful.

We say that a $(\mathcal{U}, \mathcal{V})_Q$-graph is an (s, δ, ξ, Q)*-semirespectful boundary expander* if for every set $\mathcal{U}' \subseteq \mathcal{U}$, $|\mathcal{U}'| \leq s$, we have $|\partial_Q^{\mathrm{SR}}(\mathcal{U}')| \geq \delta|\mathcal{U}'| - \xi$.

Note that disregarding the technical side conditions (which the reader can safely ignore for now), an (s, δ, ξ, Q)-(semi)respectful boundary expander is a standard bipartite boundary expander except for two modifications:

- Importantly, we measure expansion not in terms of the whole boundary but only in terms of the *(semi)respectful boundary* as described in Definition 4.6.
- Also, the size of the boundary $|\partial_Q(\mathcal{U}')|$ on the right does not have to scale quite linearly with the size of the vertex set $|\mathcal{U}'|$ on the left. Instead, we allow an *additive slack* ξ in the expansion. In most applications we will have $\xi = 0$, but for one of the results discussed in this paper we cannot obtain a good enough expander and so it will be helpful to allow a small slack.

For readers who wish to compare to Section 3, the respectful and semirespectful boundary expanders in Definition 4.8 correspond to the PC and resolution expanders in Definitions 3.17 and 3.4, respectively, only with the simplifying assumptions that all edges in the graph are (semi)respectful and that we have slack $\xi = 0$.

Before we state our main theorems we need a final technical definition, which is used to ensure that no variable appears in too many variable sets in $\mathcal{V}$. We remark that the concept below is also referred to as the "maximum degree" in the literature, but since we already have degrees of polynomials and vertices in this paper we prefer a new term instead of overloading "degree" with a third meaning.

Definition 4.9 (Overlap). The *overlap* of a variable x with respect to a family of variable sets $\mathcal{V}$ is $ol(x, \mathcal{V}) = |\{V \in \mathcal{V} : x \in V\}|$, i.e., the number of sets $V \in \mathcal{V}$ containing x, and the overlap of $\mathcal{V}$ is $ol(\mathcal{V}) = \max_x\{ol(x, \mathcal{V})\}$.

4.2 Main Technical Results

We are now ready to state the main technical contributions of this paper. What remains of this section will then be spent on providing the proofs for all of these statements. We remark that when the input to the proof system is a CNF formula, we will use the notation $\mathcal{F}$ for this formula to emphasize that it will be viewed as a collection $\mathcal{F} = \bigwedge_{F \in \mathcal{U}} \bigwedge_{C \in F} C \wedge \bigwedge_{D \in Q} D = \mathcal{U} \wedge Q$ of CNF subformulas translated to sets of polynomials.

The most general form of our resolution width and polynomial calculus degree lower bounds, when the $(\mathcal{U}, \mathcal{V})_Q$-graph is an expander with slack $\xi > 0$, can be stated as follows.

Theorem 4.10. *Let $\mathcal{F} = \bigwedge_{F \in \mathcal{U}} \bigwedge_{C \in F} C \wedge \bigwedge_{D \in Q} D = \mathcal{U} \wedge Q$ be a CNF formula represented by a $(\mathcal{U}, \mathcal{V})_Q$-graph that is an (s, δ, ξ, Q)-semirespectful boundary expander with overlap $ol(\mathcal{V}) = \ell$. Suppose furthermore that for all $\mathcal{U}' \subseteq \mathcal{U}$, $|\mathcal{U}'| \leq s$, it holds that $\mathcal{U}' \wedge Q$ is satisfiable. Then any resolution refutation of $\mathcal{F}$ requires width strictly greater than $(\delta s - 2\xi)/(2\ell)$.*

Theorem 4.11. *Let $\mathcal{P} = \bigcup_{P \in \mathcal{U}} P \cup Q$ be a set of polynomials such that $|Vars(f)| \leq (\delta s - 2\xi)/(2\ell)$ for every polynomial $f \in \mathcal{P}$. Suppose that $\mathcal{P}$ is represented by an (s, δ, ξ, Q)-respectful boundary expander $(\mathcal{U}, \mathcal{V})_Q$ with overlap $ol(\mathcal{V}) = \ell$, which in addition is such that for all $\mathcal{U}' \subseteq \mathcal{U}$, $|\mathcal{U}'| \leq s$, it holds that $\bigcup_{P \in \mathcal{U}'} P \cup Q$ is satisfiable. Then any polynomial calculus refutation of $\mathcal{P}$ requires degree strictly greater than $(\delta s - 2\xi)/(2\ell)$.*

If we want polynomial calculus lower bounds for CNF formulas we can remove the condition that $|Vars(f)| \leq (\delta s - 2\xi)/(2\ell)$ for $f \in \mathcal{P}$ and simplify the statement of Theorem 4.11 as follows.

Corollary 4.12. *Let $\mathcal{F} = \bigwedge_{F \in \mathcal{U}} \bigwedge_{C \in F} C \wedge \bigwedge_{D \in Q} D = \mathcal{U} \wedge Q$ be a CNF formula represented by an (s, δ, ξ, Q)-respectful boundary expander $(\mathcal{U}, \mathcal{V})_Q$ with overlap $ol(\mathcal{V}) = \ell$, and suppose furthermore that for all $\mathcal{U}' \subseteq \mathcal{U}$, $|\mathcal{U}'| \leq s$, it holds that $\mathcal{U}' \wedge Q$ is satisfiable. Then any polynomial calculus refutation of $\mathcal{F}$ requires degree strictly greater than $(\delta s - 2\xi)/(2\ell)$.*

We can also prove corollaries of Theorems 4.10 and 4.11 stating that we do not need any separate condition saying that all sufficiently small families $\mathcal{U}'$ must be satisfiable when the $(\mathcal{U}, \mathcal{V})_Q$-graph has standard boundary expansion without any additive slack, i.e., when $\xi = 0$ (which is the most commonly occurring case in applications).

Corollary 4.13. *Let $\mathcal{F} = \bigwedge_{F \in \mathcal{U}} \bigwedge_{C \in F} C \wedge \bigwedge_{D \in Q} D = \mathcal{U} \wedge Q$ be a CNF formula represented by an $(s, \delta, 0, Q)$-semirespectful boundary expander $(\mathcal{U}, \mathcal{V})_Q$ with slack 0 and overlap $ol(\mathcal{V}) = \ell$. Then any resolution refutation of $\mathcal{F}$ requires width strictly greater than $\delta s/(2\ell)$.*

Corollary 4.14. *Let $\mathcal{P} = \bigcup_{P \in \mathcal{U}} P \cup Q$ be a set of polynomials such that $|Vars(f)| \leq \delta s/(2\ell)$ for every polynomial $f \in \mathcal{P}$ and suppose that $\mathcal{P}$ is represented by $(s, \delta, 0, Q)$-respectful boundary expander $(\mathcal{U}, \mathcal{V})_Q$ with slack 0 and overlap $ol(\mathcal{V}) = \ell$. Then any polynomial calculus refutation of $\mathcal{P}$ requires degree strictly greater than $\delta s/(2\ell)$.*

4.3 Proof of Resolution Width Lower Bounds

As a warm-up before attacking the more challenging polynomial calculus proof system, we establish Theorem 4.10 and Corollary 4.13 for resolution. These results are nothing but fairly straightforward generalizations of [9], albeit expressed in another language, but we will prove them using a slightly different technique that will prepare us for the polynomial calculus degree lower bounds that will follow. Our presentation for polynomial calculus will be self-contained, however, and will not depend on anything in this subsection, so the reader who so wishes can safely skip ahead to Section 4.4.

We are given a CNF formula $\mathcal{F} = \bigwedge_{F \in \mathcal{U}} \bigwedge_{C \in F} C \wedge \bigwedge_{D \in Q} D = \mathcal{U} \wedge Q$ represented by a $(\mathcal{U}, \mathcal{V})_Q$-graph. We want to associate clauses C in a derivation π from $\mathcal{F}$ with subsets of clauses of $\mathcal{F}$ that could have been used to derive C, and we will use the $(\mathcal{U}, \mathcal{V})_Q$-graph to do so as follows.

Definition 4.15 (Clausal neighbourhood). The *neighbourhood* $N(C)$ of a clause C with respect to $(\mathcal{U}, \mathcal{V})_Q$ is the family of all variable sets $V \in \mathcal{V}$ containing variables in C, or in formal notation $N(C) = \{V \in \mathcal{V} \mid Vars(C) \cap V \neq \emptyset\}$.

Definition 4.16 (Clausal support). Let C be a clause and s be a positive integer. Then we say that the left vertex set $\mathcal{U}' \subseteq \mathcal{U}$ in the $(\mathcal{U}, \mathcal{V})_Q$-graph is *$(s, C)$-semirespectfully contained*, or just *(s, C)-contained*, if $|\mathcal{U}'| \leq s$ and $\partial^{\mathrm{SR}}_Q(\mathcal{U}') \subseteq N(C)$.

The *clausal semirespectful s-support* $Sup_s^{\mathsf{SR}}(C)$ *of* C *with respect to* $(\mathcal{U}, \mathcal{V})_Q$, or just *clausal s-support of* C for brevity, is defined to be the union of all (s, C)-contained subsets $\mathcal{U}' \subseteq \mathcal{U}$.

Given a small-width resolution derivation π from $\mathcal{F}$, our plan is to show the following for all clauses $C \in \pi$:

(1) The clausal s-support $Sup_s^{\mathsf{SR}}(C)$ is not large.
(2) The clause set $Sup_s^{\mathsf{SR}}(C) \cup Q$ is enough to derive C.

If we can achieve this, then we are done. Since $\left|Sup_s^{\mathsf{SR}}(C)\right|$ is small, the clause set $Sup_s^{\mathsf{SR}}(C) \cup Q$ is satisfiable. The fact that C is derivable from a satisfiable set means that this clause is also satisfiable, and so no small-width derivation can derive the contradictory empty clause.

Let us first take care of item 1 on our list.

Lemma 4.17. *Suppose that* $(\mathcal{U}, \mathcal{V})_Q$ *is an* (s, δ, ξ, Q)*-semirespectful boundary expander with overlap* $ol(\mathcal{V}) = \ell$ *and let* C *be a clause of width* $W(C) \leq (\delta s - 2\xi)/(2\ell)$*. Then the clausal s-support* $Sup_s^{\mathsf{SR}}(C)$ *with respect to* $(\mathcal{U}, \mathcal{V})_Q$ *is* $(s/2, C)$*-contained.*

Proof. By definition, the clausal s-support $Sup_s^{\mathsf{SR}}(C)$ can be written on the form $Sup_s^{\mathsf{SR}}(C) = \bigcup_i \mathcal{U}_i$ where each $\mathcal{U}_i$ is (s, C)-contained.

Let $\mathcal{U}_i$ be any (s, C)-contained set. We know that $\partial_Q^{\mathsf{SR}}(\mathcal{U}_i) \subseteq N(C)$ and that $|\mathcal{U}_i| \leq s$. In view of this size bound, the expansion properties of the $(\mathcal{U}, \mathcal{V})_Q$-graph tell us that $\left|\partial_Q^{\mathsf{SR}}(\mathcal{U}_i)\right| \geq \delta|\mathcal{U}_i| - \xi$. Hence, it holds that

$$|\mathcal{U}_i| \leq (|N(C)| + \xi)/\delta \,. \tag{4.1}$$

Furthermore, we have

$$|N(C)| \leq |Vars(C)| \cdot ol(\mathcal{V}) \leq W(C) \cdot \ell \leq \delta s/2 - \xi \,. \tag{4.2}$$

Combining (4.1) and (4.2), we conclude that

$$|\mathcal{U}_i| \leq s/2 \,, \tag{4.3}$$

or, in words, that any (s, C)-contained set is also $(s/2, C)$-contained.

Next, let us argue for any pair of (s, C)-contained sets $\mathcal{U}_1, \mathcal{U}_2 \subseteq \mathcal{U}$ that their union $\mathcal{U}_1 \cup \mathcal{U}_2$ is also (s, C)-contained. We just showed that $|\mathcal{U}_i| \leq s/2$ for $i = 1, 2$ and hence $|\mathcal{U}_1 \cup \mathcal{U}_2| \leq s$. It also holds for $i = 1, 2$ that $\partial_Q^{\mathsf{SR}}(\mathcal{U}_i) \subseteq N(C)$, which implies that $\partial_Q^{\mathsf{SR}}(\mathcal{U}_1 \cup \mathcal{U}_2) \subseteq N(C)$, because the boundary of the union is at most the union of the boundaries. This establishes that $\mathcal{U}_1 \cup \mathcal{U}_2$ is (s, C)-contained, and hence $(s/2, C)$-contained. By induction on the number of (s, C)-contained sets we conclude that the support $Sup_s^{\mathsf{SR}}(C)$ is $(s/2, C)$-contained, which establishes the lemma. □

We continue on the list to item 2. Recall that we write $F \vDash C$ if F implies C, i.e., if any truth value assignment that satisies F must also satisfy C.

Lemma 4.18. *Let* $\mathcal{F}$ *be a CNF formula represented by a* $(\mathcal{U}, \mathcal{V})_Q$*-graph that is an* (s, δ, ξ, Q)*-semirespectful boundary expander with overlap* $ol(\mathcal{V}) = \ell$*, and suppose that* π *is a resolution derivation in width at most* $(\delta s - 2\xi)/(2\ell)$ *from* $\mathcal{F}$*. Then for every* $C \in \pi$ *it holds that* $Sup_s^{\mathsf{SR}}(C) \cup Q \vDash C$*.*

Proof. We prove that

$$Sup_s^{\mathsf{SR}}(C) \cup Q \vDash C \tag{4.4}$$

by forward induction over the clauses C in the resolution derivation π.

The base case is when C is an axiom. If $C \in Q$, then (4.4) obviously holds. Suppose that $C \notin Q$ and let the CNF subformula F be a left-hand vertex such that $C \in F$. We claim that $\{F\}$ is (s, C)-contained, from which it follows that $C \in Sup_s^{\mathsf{SR}}(C)$ and (4.4) holds. To argue this, consider any neighbour $V \in N(F)$. If $V \notin N(C)$, then $V \cap Vars(C) = \emptyset$. If $Q \vDash C$, then (4.4) holds and we are done, so suppose there is an assignment ρ that satisfies Q but falsifies C (and hence F). Clearly, there is no way to modify this assignment on V to satisfy C, so V is not a semirespectful neighbour of F. But this shows that $\partial_Q^{\mathsf{SR}}(\{F\}) \subseteq N(C)$, and so $\{F\}$ is (s, C)-contained as claimed.

For the induction step, suppose that C is derived from C_1 and C_2. By the induction hypothesis we have for $i = 1, 2$ that $Sup_s^{\mathsf{SR}}(C_i) \cup Q \vDash C_i$. By the soundness of resolution, it holds that

$$Sup_s^{\mathsf{SR}}(C_1) \cup Sup_s^{\mathsf{SR}}(C_2) \cup Q \vDash C\,. \tag{4.5}$$

We claim that it also holds that

$$\big(\big(Sup_s^{\mathsf{SR}}(C_1) \cup Sup_s^{\mathsf{SR}}(C_2)\big) \cap Sup_s^{\mathsf{SR}}(C)\big) \cup Q \vDash C\,, \tag{4.6}$$

from which the inductive step follows.

For brevity, let us write $\mathcal{S} = (Sup_s^{\mathsf{SR}}(C_1) \cup Sup_s^{\mathsf{SR}}(C_2))$. Note that by Lemma 4.17 we have $\big|Sup_s^{\mathsf{SR}}(C_i)\big| \leq s/2$ for $i = 1, 2$, and hence $|\mathcal{S}| \leq s$. To establish (4.6), we show the more general claim that for any $\mathcal{S} \subseteq \mathcal{U}$ of size $|\mathcal{S}| \leq s$ such that

$$\mathcal{S} \cup Q \vDash C \tag{4.7}$$

the implication

$$\big(\mathcal{S} \cap Sup_s^{\mathsf{SR}}(C)\big) \cup Q \vDash C \tag{4.8}$$

must also hold.

To establish this, we argue that if $\mathcal{S} \setminus Sup_s^{\mathsf{SR}}(C) \neq \emptyset$, then we can decrease the size of $\mathcal{S}$ while maintaining the implication (4.7). The assumption $\mathcal{S} \setminus Sup_s^{\mathsf{SR}}(C) \neq \emptyset$ implies, in particular, that $\mathcal{S}$ is not (s, C)-contained. Since the size constraint $|\mathcal{S}| \leq s$ is satisfied, it follows that $\partial_Q^{\mathsf{SR}}(\mathcal{S}) \nsubseteq N(C)$. Fix some $V \in \partial_Q^{\mathsf{SR}}(\mathcal{S}) \setminus N(C)$ with unique neighbour $F \in \mathcal{S}$ and assume towards contradiction that $(\mathcal{S} \setminus \{F\}) \cup Q \nvDash C$. This means that there is an assignment ρ that satisfies $(\mathcal{S} \setminus \{F\}) \cup Q$ but falsifies C. Since F and V are semirespectful neighbours we can modify ρ on V so that it satisfies $F \wedge Q$. The rest of $\mathcal{S}$ stays satisfied since no variable in V occurs in these clauses, and for the same reason C stays falsified. But this contradicts (4.7), and hence $(\mathcal{S} \setminus \{F\}) \cup Q \vDash C$.

By the induction principle, as long as $\mathcal{S} \setminus Sup_s^{\mathsf{SR}}(C) \neq \emptyset$ we can remove some F from $\mathcal{S}$ while maintaining $(\mathcal{S} \setminus \{F\}) \cup Q \vDash C$. When this process terminates, we have constructed a set $\mathcal{S}^* \subseteq \mathcal{S} \cap Sup_s^{\mathsf{SR}}(C)$ such that $\mathcal{S}^* \cup Q \vDash C$, meaning that (4.8) holds. This completes the induction step in our forward induction proof over π, and the lemma follows. □

Thanks to Lemmas 4.17 and 4.18, Theorem 4.10 now follows easily.

Proof of Theorem 4.10. Recall that $\mathcal{F} = \bigwedge_{F \in \mathcal{U}} \bigwedge_{C \in F} C \wedge \bigwedge_{D \in Q} D = \mathcal{U} \wedge Q$ is a CNF formula for which $(\mathcal{U}, \mathcal{V})_Q$ is an (s, δ, ξ, Q)-semirespectful boundary expander with overlap $ol(\mathcal{V}) = \ell$. We are also assuming that it holds for all $\mathcal{U}' \subseteq \mathcal{U}$ of size $|\mathcal{U}'| \leq s$ that $\mathcal{U}' \wedge Q$ is satisfiable.

Let π be any resolution derivation from $\mathcal{F}$ in which all clauses have width at most $(\delta s - 2\xi)/(2\ell)$. By Lemma 4.17 we have $\big|Sup_s^{\mathsf{SR}}(C)\big| \leq s$ for all $C \in \pi$. By Lemma 4.18 it holds that $Sup_s^{\mathsf{SR}}(C) \cup Q \vDash C$. Then it follows from the assumption in the theorem statement that $Sup_s^{\mathsf{SR}}(C) \cup Q$ is satisfiable, and hence C cannot be the contradictory empty clause. This establishes the theorem. □

In order to prove Corollary 4.13, one way to argue directly is to observe that for an $(s, \delta, 0, Q)$-semirespectful boundary expander the empty clause will have empty support, and so Lemma 4.18 shows that contradiction cannot be derived since this would imply that the set of clauses Q is contradictory (which would violate the definition of Q-semirespectful boundary expanders).

If we do not want to go inside the proofs and tinker, we can instead show that if $(\mathcal{U}, \mathcal{V})_Q$ is an $(s, \delta, 0, Q)$-semirespectful boundary expander with slack $\xi = 0$, then this implies that for all $\mathcal{U}' \subseteq \mathcal{U}$

of size $|\mathcal{U}'| \leq s$ it holds that $\mathcal{U}' \wedge Q$ is satisfiable, after which we can appeal to Theorem 4.10. This is the next lemma.

Lemma 4.19. *Let $\mathcal{F}$ be a CNF formula represented by a $(\mathcal{U}, \mathcal{V})_Q$-graph that is an $(s, \delta, 0, Q)$-semirespectful boundary expander for $\delta > 0$. Then for any $\mathcal{U}' \subseteq \mathcal{U}$ $|\mathcal{U}'| \leq s$, it holds that the set of clauses $\bigcup_{F \in \mathcal{U}'} F \cup Q$ is satisfiable.*

Proof. Let $\mathcal{U}' \subseteq \mathcal{U}$ be any subset of size at most s. We will argue by induction on $|\mathcal{U}'|$.

For the base case, suppose that $|\mathcal{U}'| = 1$ and let $\mathcal{U}' = \{F\}$. Now, F must have a semirespectful neighbour V, because otherwise $\{F\}$ would be a non-expanding set. Fix any assignment ρ that satisfies Q (and that exists by assumption). Since V is a semirespectful neighbour of F we can flip ρ on V to satisfy $F \wedge Q$.

For the inductive step, we again have that since $|\mathcal{U}'| \leq s$ and $(\mathcal{U}, \mathcal{V})_Q$ is an $(s, \delta, 0, Q)$-expander it must hold that $\left|\partial_Q^{\mathrm{SR}}(\mathcal{U}')\right| \geq \delta|\mathcal{U}'| > 0$. Hence, by the definition of Q-semirespectful boundary there exists a variable set $V' \in \partial_Q^{\mathrm{SR}}(\mathcal{U}')$ and a set of clauses $F \in \mathcal{U}'$ such that V' is a Q-semirespectful neighbour of F but is not a neighbour of any other clause set in $\mathcal{U}' \setminus \{F\}$.

By the inductive hypothesis, there exists an assignment ρ that satisfies $(\mathcal{U}' \setminus \{F\}) \cup Q$. Just as above, we can flip ρ on V to satisfy $F \wedge Q$. Since the variables V do not occur in $\mathcal{U}' \setminus \{F\}$, all of these clauses are still satisfied by the modified assignment. The lemma now follows by the induction principle. □

4.4 Proof Strategy for Polynomial Calculus Lower Bound

We now proceed to polynomial calculus, for which we want to establish Theorem 4.11 and Corollaries 4.12 and 4.14. This will require substantially more work than the results above for resolution.

As is the case in [2], the proof of our main theorem is based on a foundational lemma from [37]. The idea is to give an overapproximation of what polynomials can be derived in degree at most D by defining an operator R on multilinear polynomials such that all degree-D consequences of the axioms are contained in the set $\{p \mid R(p) = 0\}$. The degree lower bound then follows by showing that $R(1) \neq 0$.

Lemma 4.20 ([37]). *Let $\mathcal{P}$ be any set of multilinear polynomials and $D \in \mathbb{N}^+$ be a positive integer. Suppose that there exists a linear operator R on multilinear polynomials over $Vars(\mathcal{P})$ of degree at most D with the following properties:*

(1) $R(1) \neq 0$.
(2) $R(f) = 0$ *for all axioms* $f \in \mathcal{P}$.
(3) *For every term t with $Deg(t) < D$ and every variable x it holds that $R(xt) = R(xR(t))$.*

Then any polynomial calculus refutation of $\mathcal{P}$ requires degree strictly greater than D.

The proof of Lemma 4.20 is not hard. The basic idea is that R will map all axioms to 0 by property 2, and further derivation steps in degree at most D will yield polynomials that also map to 0 by property 3 and the linearity of R. But then property 1 implies that no derivation in degree at most D can reach contradiction.

To prove Theorem 4.11, we construct a linear operator $R_{\mathcal{G}}$ that satisfies the conditions of Lemma 4.20 when the $(\mathcal{U}, \mathcal{V})_Q$-graph $\mathcal{G}$ is an expander. First, let us describe how we make the connection between polynomials and the given $(\mathcal{U}, \mathcal{V})_Q$-graph.

Definition 4.21 (Term and polynomial neighbourhood). The *neighbourhood* $N(t)$ of a term t with respect to $(\mathcal{U}, \mathcal{V})_Q$ is $N(t) = \{V \in \mathcal{V} \mid Vars(t) \cap V \neq \emptyset\}$, i.e., the family of all variable sets containing variables mentioned by t. The neighbourhood of a polynomial $p = \sum_i t_i$ is $N(p) = \bigcup_i N(t_i)$, i.e., the union of the neighbourhoods of all terms in p.

This associates a family of variable sets to every polynomial in the natural way. But we are more interested in going "in the other direction" to find out which axioms are needed in order to derive this polynomial. That is, given a family of variable sets $\mathcal{V}'$, we would like to identify the largest set of axioms $\mathcal{U}'$ that could possibly have been used in a small-degree derivation that yielded polynomials p with $Vars(p) \subseteq \bigcup_{V \in \mathcal{V}'} V$. This is the intuition behind the next definition.[9]

Definition 4.22 (Polynomial support). Let $\mathcal{V}' \subseteq \mathcal{V}$ be a family of variable sets in a $(\mathcal{U}, \mathcal{V})_Q$-graph and let s be a positive integer. Then we say that $\mathcal{U}'$ is $(s, \mathcal{V}')$*-respectfully contained*, or just $(s, \mathcal{V}')$*-contained*, if $|\mathcal{U}'| \leq s$ and $\partial_Q(\mathcal{U}') \subseteq \mathcal{V}'$.

We define the *polynomial respectful s-support $Sup_s(\mathcal{V}')$ of $\mathcal{V}'$ with respect to $(\mathcal{U}, \mathcal{V})_Q$*, or just *$s$-support of $\mathcal{V}'$* for brevity, to be the union of all $(s, \mathcal{V}')$-contained subsets $\mathcal{U}' \subseteq \mathcal{U}$. The s-support $Sup_s(t)$ of a term t is defined to be the s-support of $N(t)$.

We will usually just speak about "support" below without further qualifying this term, since the $(\mathcal{U}, \mathcal{V})_Q$-graph $\mathcal{G}$ will be clear from context. The next two observations follow immediately from Definition 4.22, but will be helpful to state explicitly.

Observation 4.23. *If $\mathcal{V}' \subseteq \mathcal{V}''$ and $\mathcal{U}'$ is $(s, \mathcal{V}')$-contained, then $\mathcal{U}'$ is also $(s, \mathcal{V}'')$-contained.*

Observation 4.24. *Let t and t' be two terms such that $Vars(t) \subseteq Vars(t')$. Then it holds that $Sup_s(t) \subseteq Sup_s(t')$.*

Once we have identified the axioms that are potentially involved in deriving p, we define the linear operator $R_{\mathcal{G}}$ as the reduction modulo the ideal generated by these axioms as in Definition 2.8. We will show that under the assumptions in Theorem 4.11 it holds that this operator satisfies the conditions in Lemma 4.20. Let us first introduce some notation for the set of all polynomials that can be generated from a subset of axioms $\mathcal{U}' \subseteq \mathcal{U}$.

Definition 4.25. For a $(\mathcal{U}, \mathcal{V})_Q$-graph and $\mathcal{U}' \subseteq \mathcal{U}$, we write $\mathcal{I}_Q(\mathcal{U}')$ to denote the ideal generated by the polynomial axioms in $\bigcup_{P \in \mathcal{U}'} P \cup Q$.

That is, $\mathcal{I}_Q(\mathcal{U}')$ is the smallest set I of multilinear polynomials that contains all axioms in $\bigcup_{P \in \mathcal{U}'} P \cup Q$ and that is closed under addition of $p_1, p_2 \in I$ and under multiplication of $p \in I$ by any multilinear polynomial over $Vars\big(\bigcup_{P \in \mathcal{U}} P \cup Q\big)$ (where as before the resulting product is implicitly multilinearized).

Definition 4.26 ($(\mathcal{U}, \mathcal{V})_Q$-graph reduction). For a $(\mathcal{U}, \mathcal{V})_Q$-graph $\mathcal{G}$, the *$(\mathcal{U}, \mathcal{V})_Q$-graph reduction* $R_{\mathcal{G}}$ of a term t is defined as $R_{\mathcal{G}}(t) = R_{\mathcal{I}_Q(Sup_s(t))}(t)$. For a polynomial p, we define $R_{\mathcal{G}}(p)$ to be the linear extension of the operator $R_{\mathcal{G}}$ defined on terms.

Looking at Definition 4.26, it is not clear that we are making progress. On the one hand, we have defined $R_{\mathcal{G}}$ in terms of standard reduction operators modulo ideals, which is nice since there is a well-developed machinery for such operators. On the other hand, it is not clear how to actually compute using $R_{\mathcal{G}}$. The problem is that if we look at a polynomial $p = \sum_i t_i$ and want to compute $R_{\mathcal{G}}(p)$, then as we expand $R_{\mathcal{G}}(p) = \sum_i R_{\mathcal{G}}(t_i)$ we end up reducing terms in one and the same polynomial modulo a priori completely different ideals. How can we get any sense of what p reduces to in such a case? The answer is that if our $(\mathcal{U}, \mathcal{V})_Q$-graph is a good enough expander, then this is not an issue at all. Instead, it turns out that we can pick a suitably large ideal containing the support of all the terms in p and reduce p modulo this larger ideal instead without changing anything. This key result is proven in Lemma 4.30 below. To establish this lemma, we need to develop a better understanding of polynomial support.

[9] We remark that for convenience we use a slightly modified, but almost equivalent, version of the original definition in [2]. This modification was proposed by Yuval Filmus and can also be found in the paper [20] mentioned in the introduction.

4.5 Some Properties of Polynomial Support and the Graph Reduction Operator

A crucial technical property that we will need is that if a $(\mathcal{U},\mathcal{V})_Q$-graph is a good expander in the sense of Definition 4.8, then for small enough sets $\mathcal{V}'$ all $(s,\mathcal{V}')$-contained subsets $\mathcal{U}' \subseteq \mathcal{U}$ as per Definition 4.22 are of at most half the allowed size.

Lemma 4.27. *Let $(\mathcal{U},\mathcal{V})_Q$ be an (s,δ,ξ,Q)-respectful boundary expander and let $\mathcal{V}' \subseteq \mathcal{V}$ be such that $|\mathcal{V}'| \leq \delta s/2 - \xi$. Then it holds that every $(s,\mathcal{V}')$-contained subset $\mathcal{U}' \subseteq \mathcal{U}$ is in fact $(s/2,\mathcal{V}')$-contained.*

Proof. As $|\mathcal{U}'| \leq s$ we can appeal to the expansion property of the $(\mathcal{U},\mathcal{V})_Q$-graph to derive the inequality $|\partial_Q(\mathcal{U}')| \geq \delta|\mathcal{U}'| - \xi$. In the other direction, we can obtain an upper bound on the size of $\partial_Q(\mathcal{U}')$ by noting that for any $(s,\mathcal{V}')$-contained set $\mathcal{U}'$ it holds that $\partial_Q(\mathcal{U}') \subseteq \mathcal{V}'$ and hence $|\partial_Q(\mathcal{U}')| \leq |\mathcal{V}'|$. If we combine these bounds and use the assumption that $|\mathcal{V}'| \leq \delta s/2 - \xi$, we can conclude that $|\mathcal{U}'| \leq s/2$, which proves that $\mathcal{U}'$ is $(s/2,\mathcal{V}')$-contained. □

Even more importantly, Lemma 4.27 now allows us to conclude that for a small enough subset $\mathcal{V}'$ on the right-hand side of $(\mathcal{U},\mathcal{V})_Q$ it holds that in fact the whole polynomial s-support $Sup_s(\mathcal{V}')$ on the left-hand side is $(s/2,\mathcal{V}')$-contained.

Lemma 4.28. *Let $(\mathcal{U},\mathcal{V})_Q$ be an (s,δ,ξ,Q)-respectful boundary expander and let $\mathcal{V}' \subseteq \mathcal{V}$ be such that $|\mathcal{V}'| \leq \delta s/2 - \xi$. Then the s-support $Sup_s(\mathcal{V}')$ of $\mathcal{V}'$ with respect to $(\mathcal{U},\mathcal{V})_Q$ is $(s/2,\mathcal{V}')$-contained.*

Proof. We show that for any pair of $(s,\mathcal{V}')$-contained sets $\mathcal{U}_1,\mathcal{U}_2 \subseteq \mathcal{U}$ their union $\mathcal{U}_1 \cup \mathcal{U}_2$ must also be $(s,\mathcal{V}')$-contained. First, by Lemma 4.27 we have $|\mathcal{U}_i| \leq s/2$ for $i = 1,2$ and hence $|\mathcal{U}_1 \cup \mathcal{U}_2| \leq s$. Second, for $i = 1,2$ it holds that $\partial_Q(\mathcal{U}_i) \subseteq \mathcal{V}'$, which implies that $\partial_Q(\mathcal{U}_1 \cup \mathcal{U}_2) \subseteq \partial_Q(\mathcal{U}_1) \cup \partial_Q(\mathcal{U}_2) \subseteq \mathcal{V}'$, because taking the union of two sets can only shrink the boundary. This establishes that $\mathcal{U}_1 \cup \mathcal{U}_2$ is $(s,\mathcal{V}')$-contained.

By induction on the number of $(s,\mathcal{V}')$-contained sets we can conclude that the support $Sup_s(\mathcal{V}')$ is $(s,\mathcal{V}')$-contained as well, after which one final application of Lemma 4.27 shows that this set is $(s/2,\mathcal{V}')$-contained. This completes the proof. □

We can now prove the key technical lemma that if a term t does not have too large support (which will be true if its degree is not too large), then it it is possible to reduce t modulo suitably chosen larger ideals, generated also by some polynomials outside of the support of t, without changing the result of the reduction operator. As a first step, we show that in such larger ideals there must exist particular kinds of axioms P, for which we later argue that they do not affect the reduction operator.

Lemma 4.29. *Let $\mathcal{G}$ be a $(\mathcal{U},\mathcal{V})_Q$-graph and let t be any term. Suppose that $\mathcal{U}' \subseteq \mathcal{U}$ is such that $\mathcal{U}' \supsetneq Sup_s(t)$ and $|\mathcal{U}'| \leq s$. Then there is a set of polynomials P and a variable set V such that $P \in \mathcal{U}' \setminus Sup_s(t)$ and $V \in \big(\partial_Q(\mathcal{U}') \cap N(P)\big) \setminus N(t)$.*

Proof. First note that $\mathcal{U}'$ cannot be $(s,N(t))$-contained, because by Definition 4.22 it would then hold that $\mathcal{U}' \subseteq Sup_s(t)$, contradicting the assumption in the lemma that $\mathcal{U}'$ is a strict superset of $Sup_s(t)$. We claim that the fact that $\mathcal{U}'$ is not $(s,N(t))$-contained implies that we can find a set of polynomials P with a neighbouring subset of variables V satisfying the conditions of the lemma. To argue this, note that since $|\mathcal{U}'| \leq s$ it follows from Definition 4.22 that the reason $\mathcal{U}'$ is not $(s,N(t))$-contained is that there exists some $P \in \mathcal{U}'$ and some set of variables $V \in N(P)$ such that $V \in \partial_Q(\mathcal{U}') \setminus N(t)$. Moreover, the assumption $\mathcal{U}' \supsetneq Sup_s(t)$ implies that such a P cannot be in $Sup_s(t)$. To see why, note that since the support $Sup_s(t)$ is the union of $(s,N(t))$-contained sets, which are sets $\mathcal{U}^*$ such that $\partial_Q(\mathcal{U}^*) \subseteq N(t)$, it certainly holds that

$\partial_Q(Sup_s(t)) \subseteq N(t)$ (since, as already argued in the proof of Lemma 4.28, taking unions can only shrink the boundary). Now, if $P \in Sup_s(t) \subseteq \mathcal{U}'$, then by using Observation 4.7 we conclude that $V \in \big(\partial_Q(\mathcal{U}') \cap N(Sup_s(t))\big) \subseteq \partial_Q(Sup_s(t)) \subseteq N(t)$, contradicting $V \notin N(t)$. Thus, we have shown the existence of a pair (P, V) such that $P \in \mathcal{U}' \setminus Sup_s(t)$ and $V \in \big(\partial_Q(\mathcal{U}') \cap N(P)\big) \setminus N(t)$, which proves the lemma. □

Using P and V provided by the previous lemma, we can prove the crucial technical property of the reduction operator that enlarging the ideal does not change the resulting reduced polynomial if the enlargement is done in the right way. Note that up to this point we have argued purely combinatorially about the structure of the $(\mathcal{U}, \mathcal{V})_Q$-graph, but in order to establish the next lemma we need to make use of the respectfulness of the edges in the graph.

Lemma 4.30. *Let $\mathcal{G}$ be a $(\mathcal{U}, \mathcal{V})_Q$-graph and let t be any term. Suppose that $\mathcal{U}' \subseteq \mathcal{U}$ is such that $\mathcal{U}' \supseteq Sup_s(t)$ and $|\mathcal{U}'| \leq s$. Then it holds that $R_{\mathcal{I}_Q(\mathcal{U}')}(t) = R_{\mathcal{I}_Q(Sup_s(t))}(t)$.*

Proof. If $\mathcal{U}' = Sup_s(t)$ the lemma trivially holds. Otherwise, we can use Lemma 4.29 to find a polynomial set $P \in \mathcal{U}' \setminus Sup_s(t)$ and a variable set $V \in \big(\partial_Q(\mathcal{U}') \cap N(P)\big) \setminus N(t)$. Fixing such P and V, our claim is that if P is removed from the generators of the ideal, it holds that $R_{\mathcal{I}_Q(\mathcal{U}')}(t) = R_{\mathcal{I}_Q(\mathcal{U}' \setminus \{P\})}(t)$. Given this claim we are done, since we can then argue by induction over the elements in $\mathcal{U}' \setminus Sup_s(t)$, removing them one by one, to arrive at the conclusion that $R_{\mathcal{I}_Q(\mathcal{U}')}(t) = R_{\mathcal{I}_Q(Sup_s(t))}(t)$, proving the lemma.

We proceed to establish the claim that $R_{\mathcal{I}_Q(\mathcal{U}')}(t) = R_{\mathcal{I}_Q(\mathcal{U}' \setminus \{P\})}(t)$. By Definition 2.8 we know that the polynomial $q = t - R_{\mathcal{I}_Q(\mathcal{U}')}(t)$ is contained in the ideal $\mathcal{I}_Q(\mathcal{U}')$. Equivalently, t can then be written as a sum $t = q + R_{\mathcal{I}_Q(\mathcal{U}')}(t)$, where $q \in \mathcal{I}_Q(\mathcal{U}')$ and $R_{\mathcal{I}_Q(\mathcal{U}')}(t)$ consists of terms irreducible modulo $\mathcal{I}_Q(\mathcal{U}')$, where this sum is unique by Fact 2.7. We are going to hit the equation $t = q + R_{\mathcal{I}_Q(\mathcal{U}')}(t)$ with a restriction that satisfies P while leaving t, $\mathcal{U}' \setminus \{P\}$, and Q untouched, which will result in an equation $t = q' + r'$ for which we will argue that $r' = R_{\mathcal{I}_Q(\mathcal{U}' \setminus \{P\})}(t)$ as well as $r' = R_{\mathcal{I}_Q(\mathcal{U}')}(t)$, implying $R_{\mathcal{I}_Q(\mathcal{U}' \setminus \{P\})}(t) = R_{\mathcal{I}_Q(\mathcal{U}')}(t)$.

As our restriction ρ we choose any assignment with domain $\mathrm{dom}(\rho) = V$ that Q-respectfully satisfies P. Note that there exists at least one such assignment since $V \in \partial_Q(\mathcal{U}') \cap N(P)$ is a Q-respectful neighbour of P by Definition 4.6. By the choice of ρ it holds that P is satisfied, i.e., that all axioms in P are set to 0. Furthermore, none of the axioms in $\mathcal{U}' \setminus \{P\}$ are affected by ρ since V is in the boundary of $\mathcal{U}'$.[10] As for axioms in Q it is not necessarily true that ρ will leave all of them untouched, but by assumption ρ respects Q and so any axiom in Q is either satisfied (and zeroed out) by ρ or is left intact. From this and the fact that $q \in \mathcal{I}_Q(\mathcal{U}')$ can be written as a polynomial combination $q = \sum_i p_i f_i$ of axioms $f_i \in \bigcup_{P' \in \mathcal{U}'} P' \cup Q$ for some polynomials p_i, it follows that $q{\restriction_\rho}$ can be written as a polynomial combination $q{\restriction_\rho} = \sum_i \big(p_i{\restriction_\rho}\big) f_i$, where $f_i \in \bigcup_{P' \in (\mathcal{U}' \setminus \{P\})} P' \cup \{q^* \in Q \mid Vars(q^*) \cap \mathrm{dom}(\rho) = \emptyset\}$. In other words, it holds that $q{\restriction_\rho} \in \mathcal{I}_Q(\mathcal{U}' \setminus \{P\})$. To see that t is preserved, note that ρ does not assign any variables in t since $V \notin N(t)$.[11]

Hence, after hitting the equation $t = q + R_{\mathcal{I}_Q(\mathcal{U}')}(t)$ with the restriction ρ we have $t = q' + r'$, where $q' = q{\restriction_\rho} \in \mathcal{I}_Q(\mathcal{U}' \setminus \{P\})$ and where $r' = R_{\mathcal{I}_Q(\mathcal{U}')}(t){\restriction_\rho}$ is a sum of terms irreducible modulo $\mathcal{I}_Q(\mathcal{U}')$, because restrictions preserve irreducibility by Observation 2.10. But if a term is irreducible modulo $\mathcal{I}_Q(\mathcal{U}')$ it is also irreducible modulo the smaller ideal $\mathcal{I}_Q(\mathcal{U}' \setminus \{P\})$. By

[10] Recalling the remark after Definition 4.4, we note that we can ignore here if ρ happens to falsify axioms in $\mathcal{U} \setminus \mathcal{U}'$.

[11] Note that in the PC edge game as described in Definition 3.15 it is *not* necessarily the case that ρ satisfies all polynomials in P, so that they vanish as described at the beginning of the paragraph. However, if playing ρ is a winning strategy for Satisfier, this means that Adversary cannot simultaneously satisfy $Q{\restriction_\rho} \subseteq Q$ and falsify $P{\restriction_\rho}$, which is to say that the implication $Q \vDash P{\restriction_\rho}$ must hold. Since all polynomials are multilinear, it can be shown that it follows from this that $P{\restriction_\rho}$ lies in the ideal generated by Q, which is sufficient to continue the argument as in the next paragraph of the proof.

the uniqueness in Fact 2.7 it then follows that $r' = R_{\mathcal{I}_Q(\mathcal{U}' \setminus \{P\})}(t)$. On the other hand, as $q' \in \mathcal{I}_Q(\mathcal{U}' \setminus \{P\})$ it follows that q' is also in the larger ideal $\mathcal{I}_Q(\mathcal{U}')$. Since as observed before r' is a sum of terms irreducible modulo $\mathcal{I}_Q(\mathcal{U}')$, we can again apply Fact 2.7 to deduce that $r' = R_{\mathcal{I}_Q(\mathcal{U}')}(t)$. Thus, we have $R_{\mathcal{I}_Q(\mathcal{U}')}(t) = r' = R_{\mathcal{I}_Q(\mathcal{U}' \setminus \{P\})}(t)$, which proves the lemma. □

4.6 Putting the Pieces in the Proof Together

We have just a few lemmas left before we can prove Theorem 4.11, which as discussed above will be established by appealing to Lemma 4.20. We first state a lemma saying that if a term does not have too high degree, then we can bound the size of its support.

LEMMA 4.31. *Let $(\mathcal{U}, \mathcal{V})_Q$ be an (s, δ, ξ, Q)-expander with overlap $ol(\mathcal{V}) = \ell$. Then for any term t with $Deg(t) \leq (\delta s - 2\xi)/(2\ell)$ it holds that $|Sup_s(t)| \leq s/2$.*

PROOF. Note that for any $(\mathcal{U}, \mathcal{V})_Q$-graph we have $|N(t)| \leq Deg(t) \cdot ol(\mathcal{V})$. It thus follows from the bound on the overlap $ol(\mathcal{V})$ in the statement of the lemma that the size of the neighbourhood $N(t)$ is bounded by $\delta s/2 - \xi$. An application of Lemma 4.28 now yields the desired bound $|Sup_s(t)| \leq s/2$. □

If we reduce a term modulo a set of polynomials, then the result of this reduction can only contain variables from the term and polynomials in question. The next lemma formalizes a slightly stronger version of this claim by saying that we can essentially ignore Q in the argument.

LEMMA 4.32. *Consider a $(\mathcal{U}, \mathcal{V})_Q$-graph such that all $V \in \mathcal{V}$ are Q-respectful and fix any left subset $\mathcal{U}^* \subseteq \mathcal{U}$ and term t. Then it holds that $N\big(R_{\mathcal{I}_Q(\mathcal{U}^*)}(t)\big) \subseteq N(\mathcal{U}^*) \cup N(t)$.*

PROOF. Let $r = R_{\mathcal{I}_Q(\mathcal{U}^*)}(t)$ be the polynomial obtained when reducing the term t modulo $\mathcal{I}_Q(\mathcal{U}^*)$, i.e., $t = q + r$ for some $q \in \mathcal{I}_Q(\mathcal{U}^*)$ and for r a linear combination of irreducible monomials as in Fact 2.7. Consider any variable set $V \in \mathcal{V}$ such that $V \notin N(\mathcal{U}^*) \cup N(t)$. We will show that $V \notin N(r)$.

By assumption there exists an assignment ρ to all of the variables in V that respects Q. Apply ρ to the equality $t = q + r$. Note that $t{\restriction}_\rho = t$ as V is not a neighbour of t. Moreover, $q{\restriction}_\rho$ is in the ideal $\mathcal{I}_Q(\mathcal{U}^*)$ because ρ does not assign values to any variables in $\mathcal{U}^*$ and every axiom in Q sharing variables with V is set to 0 by ρ. Thus, t can be written as $t = q' + r{\restriction}_\rho$ with $q' \in \mathcal{I}_Q(\mathcal{U}^*)$. As all terms in r are irreducible modulo $\mathcal{I}_Q(\mathcal{U}^*)$, they remain irreducible after restricting r by ρ by Observation 2.10. Hence, it follows that $r{\restriction}_\rho = r$ by the uniqueness in Fact 2.7, and r cannot contain any variable from V since it remains unaffected when these variables are assigned. This in turn implies that every set $V \in N(r)$ is contained in $N(\mathcal{U}^*) \cup N(t)$. □

As the final technical step, we study what happens to the reduction operator when a term is multiplied by a variable.

LEMMA 4.33. *Let $(\mathcal{U}, \mathcal{V})_Q$ be an (s, δ, ξ, Q)-respectful boundary expander with overlap $ol(\mathcal{V}) = \ell$. Then for any term t with $Deg(t) < \lfloor (\delta s - 2\xi)/(2\ell) \rfloor$, any term t' occurring in $R_{\mathcal{I}_Q(Sup_s(t))}(t)$, and any variable x not occurring in t, it holds that $R_{\mathcal{I}_Q(Sup_s(xt'))}(xt') = R_{\mathcal{I}_Q(Sup_s(xt))}(xt')$.*

PROOF. The outline of the proof is that we want to show $Sup_s(xt') \subseteq Sup_s(xt)$ and $|Sup_s(xt)| \leq s$, which will then allows us to apply Lemma 4.30. The inequality $|Sup_s(xt)| \leq s$ follows immediately from Lemma 4.31. To prove that $Sup_s(xt')$ is a subset of $Sup_s(xt)$, we use Lemma 4.32 to show that $Sup_s(xt') \cup Sup_s(xt)$ is $(s, N(xt))$-contained in the sense of Definition 4.22. It then follows from this definition that $Sup_s(xt') \cup Sup_s(xt)$ is contained in $Sup_s(xt)$, which, in particular, implies that $Sup_s(xt') \subseteq Sup_s(xt)$. Once we have reached this point, we can apply Lemma 4.30 with $\mathcal{U}' = Sup_s(xt)$ and t replaced by xt'.

To fill in the details of this outline, first observe that $t' \in R_{\mathcal{I}_Q(Sup_s(t))}(t)$ implies $t' \preccurlyeq t$ and hence $Deg(t') \leq Deg(t)$ by the definition of admissible ordering in Definition 2.5. Thus, we can apply Lemma 4.31 to deduce that $|Sup_s(xt')| \leq s/2$ and $|Sup_s(xt)| \leq s/2$, and hence the size condition $|Sup_s(xt') \cup Sup_s(xt)| \leq s$ for containment is satisfied.

In order to establish that $Sup_s(xt') \cup Sup_s(xt)$ is $(s, N(xt))$-contained, we also need to show that $\partial_Q\big(Sup_s(xt') \cup Sup_s(xt)\big) \subseteq N(xt)$. Note that all $V \in \mathcal{V}$ are Q-respectful by the technical side condition on (s, δ, ξ, Q)-expanders in Definition 4.8. From Lemma 4.32 with $\mathcal{U}^* = Sup_s(t)$ we get

$$N(t') \subseteq N(Sup_s(t)) \cup N(t) \tag{4.9}$$

and by the monotonicity in Observation 4.24 it holds that

$$Sup_s(t) \subseteq Sup_s(xt)\,. \tag{4.10}$$

Combining (4.9) and (4.10), we derive that

$$N(xt') = N(x) \cup N(t') \subseteq N(x) \cup N(Sup_s(t)) \cup N(t) = N(Sup_s(xt)) \cup N(xt)\,. \tag{4.11}$$

If we now consider the Q-respectful boundary of the set $Sup_s(xt') \cup Sup_s(xt)$, it holds that

$$\begin{aligned} \partial_Q\big(Sup_s(xt') \cup Sup_s(xt)\big) &\subseteq \big(\partial_Q\big(Sup_s(xt')\big) \setminus N(Sup_s(xt))\big) \cup \partial_Q(Sup_s(xt)) \\ &\subseteq \big(N\big(xt'\big) \setminus N(Sup_s(xt))\big) \cup N(xt) \\ &\subseteq N(xt)\,, \end{aligned} \tag{4.12}$$

where the first line follows from the boundary definition in Definition 4.6, the second line follows by the s-support property that $\partial_Q(Sup_s(xt)) \subseteq N(xt)$ (since the support is the union of $(s, N(xt))$-contained sets), and the last line follows from (4.11). Hence, $Sup_s(xt') \cup Sup_s(xt)$ is $(s, N(xt))$-contained.

This allows us to complete our proof outline by applying Lemma 4.30 to reach the desired conclusion that the equality $R_{\mathcal{I}_Q(Sup_s(xt'))}(xt') = R_{\mathcal{I}_Q(Sup_s(xt))}(xt')$ holds. □

Now we can prove our main technical theorem for polynomial calculus degree.

Proof of Theorem 4.11. Recall that the assumptions in the statement of the theorem are that we have a $(\mathcal{U}, \mathcal{V})_Q$-graph for a set of polynomials $\mathcal{P} = \bigcup_{P \in \mathcal{U}} P \cup Q$ such that $(\mathcal{U}, \mathcal{V})_Q$ is an (s, δ, ξ, Q)-respectful boundary expander with overlap $ol(\mathcal{V}) = \ell$ and for every $f \in \mathcal{P}$ we have $|Vars(f)| \leq (\delta s - 2\xi)/(2\ell)$. Furthermore, for all $\mathcal{U}' \subseteq \mathcal{U}$, $|\mathcal{U}'| \leq s$, it holds that $\bigcup_{P \in \mathcal{U}'} P \cup Q$ is satisfiable. We want to prove that no polynomial calculus derivation from $\mathcal{P}$ of degree at most $(\delta s - 2\xi)/(2\ell)$ can reach contradiction.

Note that all axioms in $\mathcal{P}$ have degree at most $(\delta s - 2\xi)/(2\ell)$, as this is the bound on the number of variables in each axiom. We want to show that the operator $R_{\mathcal{G}}$ in Definition 4.26 satisfies the conditions of Lemma 4.20, from which Theorem 4.11 immediately follows. We can note right away that the operator $R_{\mathcal{G}}$ is linear by construction.

To prove that $R_{\mathcal{G}}(1) = R_{\mathcal{I}_Q(Sup_s(1))}(1) \neq 0$, we start by observing that the size of the s-support of 1 is upper-bounded by $s/2$ according to Lemma 4.31. Using the assumption that for every subset $\mathcal{U}'$ of $\mathcal{U}$ such that $|\mathcal{U}'| \leq s$ it holds that the set of polynomials $\bigcup_{P \in \mathcal{U}'} P \cup Q$ is satisfiable, it follows that 1 is not in the ideal $\mathcal{I}_Q(Sup_s(1))$ and hence $R_{\mathcal{I}_Q(Sup_s(1))}(1) = 1 \neq 0$.

We next show that $R_{\mathcal{G}}(f) = 0$ for any axiom $f \in \bigcup_{P \in \mathcal{U}} P \cup Q$. By the assumption that $|Vars(f)| \leq (\delta s - 2\xi)/(2\ell)$ it holds that the degree of the term $t^* = \prod_{x \in Vars(f)} x$ is bounded by $(\delta s - 2\xi)/(2\ell)$, from which it follows by Lemma 4.31 that the size of the s-support of t^* is bounded by $s/2$. As t^* contains all the variables in f, the s-support $Sup_s(t^*)$ contains the s-support of every term in f by Observation 4.24 and we can use Lemma 4.30 to conclude that $R_{\mathcal{G}}(f) = R_{\mathcal{I}_Q(Sup_s(t^*))}(f)$. If $f \in Q$, this means we are done because $\mathcal{I}_Q(Sup_s(t^*))$ contains all of Q, implying that $R_{\mathcal{G}}(f) = 0$.

For $f \in \bigcup_{P \in \mathcal{U}} P$ we need to establish that f is always contained in $\bigcup_{P \in Sup_s(t^*)} P$, from which follows that f reduces to 0 under $R_{\mathcal{G}}$. To see that $f \in \bigcup_{P \in Sup_s(t^*)} P$ we prove that for any $P \in \mathcal{U}$ with $f \in P$ it holds that all neighbours in $N(P) \setminus N(t^*)$ have to be non-respectful, and so such a P (and hence f) always makes it into the support as an $(s, N(t^*))$-contained singleton set $\{P\}$. Taking a neighbour $V \in N(P) \setminus N(t^*)$ we have that $V \cap Vars(t^*) = V \cap Vars(f) = \emptyset$ and hence no assignment to V can zero out f, because V does not mention variables in f. Thus, all assignments to V leave f unsatisfied and, hence, leave $P \ni f$ unsatisfied as well, implying that V is a non-respectful neighbour of P according to Definitions 4.3 and 4.4.[12]

It remains to prove the last property in Lemma 4.20 stating that $R_{\mathcal{G}}(xt) = R_{\mathcal{G}}(xR_{\mathcal{G}}(t))$ for any term t such that $Deg(t) < \lfloor(\delta s - 2\xi)/(2\ell)\rfloor$. We can see that this holds by studying the following sequence of equalities:

$$
\begin{aligned}
R_{\mathcal{G}}(xR_{\mathcal{G}}(t)) &= R_{\mathcal{G}}\left(x \sum_{t' \in R_{\mathcal{G}}(t)} t'\right) && \left[\text{expanding out terms in } R_{\mathcal{G}}(t)\right] \\
&= \sum_{t' \in R_{\mathcal{G}}(t)} R_{\mathcal{G}}(xt') && \left[\text{by linearity of } R_{\mathcal{G}}\right] \\
&= \sum_{t' \in R_{\mathcal{G}}(t)} R_{\mathcal{I}_Q(Sup_s(xt'))}(xt') && \left[\text{by definition of } R_{\mathcal{G}}\right] \\
&= \sum_{t' \in R_{\mathcal{G}}(t)} R_{\mathcal{I}_Q(Sup_s(xt))}(xt') && \left[\text{by Lemma 4.33}\right] \\
&= R_{\mathcal{I}_Q(Sup_s(xt))}\left(\sum_{t' \in R_{\mathcal{G}}(t)} xt'\right) && \left[\text{by linearity of } R_{\mathcal{I}_Q(Sup_s(xt))}\right] \\
&= R_{\mathcal{I}_Q(Sup_s(xt))}(xR_{\mathcal{G}}(t)) && \left[\text{collecting terms in } R_{\mathcal{G}}(t)\right] \\
&= R_{\mathcal{I}_Q(Sup_s(xt))}(xR_{\mathcal{I}_Q(Sup_s(t))}(t)) && \left[\text{by definition of } R_{\mathcal{G}}\right] \\
&= R_{\mathcal{I}_Q(Sup_s(xt))}(xt) && \left[\text{by Observation 2.9}\right] \\
&= R_{\mathcal{G}}(xt) && \left[\text{by definition of } R_{\mathcal{G}}\right]
\end{aligned}
$$

Thus, $R_{\mathcal{G}}$ satisfies all the properties of Lemma 4.20, from which the theorem follows. □

In order to get a simpler version of Theorem 4.11 for CNF formulas as stated in Corollary 4.12, a crucial observation is that removing polynomials from $\mathcal{P}$ preserves respectful expansion. Let us state and prove this as a formal lemma.

Lemma 4.34. *Let $\mathcal{P} = \bigcup_{P \in \mathcal{U}} P \cup Q$ be a set of polynomials represented by an (s, δ, ξ, Q)-respectful boundary expander $(\mathcal{U}, \mathcal{V})_Q$ with overlap $ol(\mathcal{V}) = \ell$. Then removing any axiom f from $\mathcal{P}$ yields a subset of polynomials $\mathcal{P}' = \bigcup_{P \in \mathcal{U}'} P \cup Q'$ representable by (s, δ, ξ, Q')-respectful boundary expander $(\mathcal{U}', \mathcal{V})_{Q'}$ with overlap $ol(\mathcal{V}) = \ell$.*

Proof. We analyse what happens to the $(\mathcal{U}, \mathcal{V})_Q$-graph if an axiom f is removed from $\mathcal{P}$. First, note that $\mathcal{V}$ does not change and hence the overlap remains the same. Removing axioms from Q

[12] Here we need to make the argument slightly more precise if we wish to use the PC edge game in Definition 3.15 to establish Theorem 3.18. If it holds that $Q \vDash f$, then (as noted in a footnote in the proof of Lemma 4.30) we have that f lies in the ideal generated by Q, and so $R_{\mathcal{G}}(f) = 0$ as desired. If $Q \nvDash f$, then by definition there is an assignment α that satisfies Q but falsifies f. Since $V \cap Vars(f) = \emptyset$, there is no assignment ρ to V that Satisfier could play to guarantee that f is satisfied if Adversary would respond with α, and so Satisfier does not win the PC game on the edge (P, V). But this contradicts the assumption in Theorem 3.18 that Satisfier should have a winning strategy for all edges, and so this scenario can never arise.

only relaxes the conditions on respectful satisfiability while keeping all edges in the graph, so the conditions on the expansion still hold. In removing axioms from $\mathcal{U}$ we have two cases: either removing an axiom from a set of polynomials $P \in \mathcal{U}$ yields an empty set or we are left with a non-empty set of polynomials $P' = P \setminus \{f\}$. In the former case, it is clear that we can remove the vertex P to obtain a graph $(\mathcal{U} \setminus \{P\}, \mathcal{V})_Q$ that still satisfies the same expansion conditions. In the latter case, we claim that any set $V \in \mathcal{V}$ that is a Q-respectful neighbour of P remains a Q-respectful neighbour of the set of polynomials P'. Clearly, the same assignments to V that satisfy P also satisfy $P' \subseteq P$. In particular, this shows that V must still be a neighbour of P', for otherwise P' would not share any variables with V, which would imply that no assignment to V could satisfy P' and hence P. This would contradict the assumption that V is a Q-respectful neighbour of P. Hence, we conclude that removing any axiom f from $(\mathcal{U}, \mathcal{V})_Q$ yields a $(\mathcal{U}', \mathcal{V})_{Q'}$-graph which is an (s, δ, ξ, Q')-expander. □

Proof of Corollary 4.12. Recall that our assumption is that $\mathcal{F} = \bigwedge_{F \in \mathcal{U}} \bigwedge_{C \in F} C \wedge \bigwedge_{D \in Q} D = \mathcal{U} \wedge Q$ is a CNF formula that can be represented by an (s, δ, ξ, Q)-respectful boundary expander $(\mathcal{U}, \mathcal{V})_Q$ with overlap $ol(\mathcal{V}) = \ell$. Also, $\mathcal{F}$ has the property that for all $\mathcal{U}' \subseteq \mathcal{U}$, $|\mathcal{U}'| \leq s$, it holds that $\mathcal{U}' \wedge Q$ is satisfiable. We want to prove that no polynomial calculus derivation from $\mathcal{F}$ in degree at most $(\delta s - 2\xi)/(2\ell)$ can reach contradiction.

We translate the clauses of $\mathcal{F}$ into polynomials and remove the polynomial axioms $f \in \mathcal{P}$ that have degree greater than $(\delta s - 2\xi)/(2\ell)$. Note that the width of a clause, i.e., the number of variables that it mentions, is equal to the degree of its polynomial translation, so this step removes all $f \in \mathcal{P}$ for which $|Vars(f)| > (\delta s - 2\xi)/(2\ell)$. We observe that removing all axiom clauses from $\mathcal{U} \wedge Q$ of width strictly greater than $(\delta s - 2\xi)/(2\ell)$ could leave a subformula that is satisfiable, but if so the lower bound trivially holds. Otherwise, any polynomial calculus refutation can only use axioms left in our subformula, and Lemma 4.34 tells us that this subformula is representable by an (s, δ, ξ, Q')-expander. Now the lower bound follows from Theorem 4.11. □

When the expansion slack ξ in the (s, δ, ξ, Q)-respectful boundary expander is equal to 0, we do not need any separate condition that all sufficiently small families $\mathcal{U}'$ need to be satisfiable. The next lemma is similar in spirit to Lemma 4.19, which was the lemma we used to establish the analogous statement for resolution width lower bounds in Corollary 4.13.

Lemma 4.35. *Let $\mathcal{P} = \bigcup_{P \in \mathcal{U}} P \cup Q$ be a set of polynomials represented by a $(\mathcal{U}, \mathcal{V})_Q$-graph that is an $(s, \delta, 0, Q)$-respectful boundary expander for $\delta > 0$. Then for any $\mathcal{U}' \subseteq \mathcal{U}$, $|\mathcal{U}'| \leq s$, it holds that the subset of polynomials $\bigcup_{P \in \mathcal{U}'} P \cup Q$ is satisfiable.*

Proof. Let $\mathcal{U}' \subseteq \mathcal{U}$ be any subset of size at most s. First, we show that we can find a subset $\mathcal{V}' \subseteq N(\mathcal{U}')$ and an assignment ρ to the set of variables $\bigcup_{V \in \mathcal{V}'} V$ such that ρ Q-respectfully satisfies $\mathcal{U}'$.

Let us argue by induction on $|\mathcal{U}'|$. As the $(\mathcal{U}, \mathcal{V})_Q$-graph is an $(s, \delta, 0, Q)$-expander it holds that $|\partial_Q(\mathcal{U}')| \geq \delta|\mathcal{U}'| > 0$ for any non-empty subset $\mathcal{U}'$. Hence, by the definition of Q-respectful boundary there exists a variable set $V' \in \partial_Q(\mathcal{U}')$ and a set of polynomials $P \in \mathcal{U}'$ such that V' is a Q-respectful neighbour of P but is not a neighbour of any other set of polynomials in $\mathcal{U}' \setminus \{P\}$. Therefore, there is an assignment ρ to the variables in V' that Q-respectfully satisfies P. By the induction hypothesis there also exists an assignment ρ' that Q-respectfully satisfies $\mathcal{U}' \setminus \{P\}$. Note that ρ' does not assign any variables in V' as $V' \notin N(\mathcal{U}' \setminus \{P\})$. Hence, by taking the union $\rho \cup \rho'$ we obtain an assignment to the variables in some subset of $N(\mathcal{U}')$ that Q-respectfully satisfies $\mathcal{U}'$.

We now need to show how to extend this to an assignment satisfying Q as well. To this end, let $\rho_{\mathcal{U}'}$ be an assignment to the variables in $\bigcup_{V \in \mathcal{V}'} V$ for some $\mathcal{V}' \subseteq N(\mathcal{U}')$ that Q-respectfully satisfies $\mathcal{U}'$. By another inductive argument over the size $|\mathcal{V}'' \setminus \mathcal{V}'|$ of families $\mathcal{V}'' \supseteq \mathcal{V}'$, we show that there is an assignment $\rho_{\mathcal{V}''}$ to the variables $\bigcup_{V \in \mathcal{V}''} V$ that Q-respectfully satisfies $\mathcal{U}'$ for every $\mathcal{V}''$ with $\mathcal{V}' \subseteq \mathcal{V}'' \subseteq \mathcal{V}$. When $\mathcal{V}'' = \mathcal{V}'$, we just take the assignment $\rho_{\mathcal{U}'}$. We want to show that for any $V' \in \mathcal{V} \setminus \mathcal{V}''$ we can extend $\rho_{\mathcal{V}''}$ to the variables in V' so that the new assignment Q-respectfully satisfies $\mathcal{U}'$. As V' respects Q, there is an assignment $\rho_{V'}$ to the variables V' that satisfies all affected polynomials in Q. We would like to combine $\rho_{V'}$ and $\rho_{\mathcal{V}''}$ into one assignment, but this requires some care since the intersection of the domains $V' \cap \left(\bigcup_{V \in \mathcal{V}''} V\right)$ could be non-empty. Therefore, we add to $\rho_{\mathcal{V}''}$ only the subassignment $\rho^*_{V'}$ of $\rho_{V'}$ that assigns the variables in $V' \setminus \left(\bigcup_{V \in \mathcal{V}''} V\right)$ and hence does not share any variables with $\rho_{\mathcal{V}''}$.

We claim that extending $\rho_{\mathcal{V}''}$ by $\rho^*_{V'}$ creates an assignment $\rho_{\mathcal{V}'' \cup \{V'\}}$ that respects Q. Consider a polynomial $f \in Q$ affected by $\rho_{\mathcal{V}'' \cup \{V'\}}$. If f mentions a variable in $\mathcal{V}''$, then f must already be satisfied by $\rho_{\mathcal{V}''}$, since $\rho_{\mathcal{V}''}$ respects Q. Otherwise, f does not mention any variable from $\left(\bigcup_{V \in \mathcal{V}''} V\right)$, but has to mention at least one variable from V'. Therefore, f must be satisfied by the Q-respectful assignment $\rho_{V'}$ and in particular by its subassignment $\rho^*_{V'}$ that assigns variables in $V' \setminus \left(\bigcup_{V \in \mathcal{V}''} V\right)$. It follows that every polynomial $f \in Q$ affected by $\rho_{\mathcal{V}'' \cup \{V'\}}$ must be satisfied and hence $\rho_{\mathcal{V}'' \cup \{V'\}}$ respects Q.

We have shown that we can find an assignment to all the variables $\cup_{V \in \mathcal{V}} V$ that Q-respectfully satisfies $\mathcal{U}'$. Since $\mathcal{V}$ includes all the variables in Q, and since Q does not contain any constant polynomials according to Definition 4.1, this means that Q is also fully satisfied. Hence, $\bigcup_{P \in \mathcal{U}'} P \cup Q$ is satisfiable and the lemma follows. □

Proof of Corollary 4.14. Suppose that $\mathcal{P} = \bigcup_{P \in \mathcal{U}} P \cup Q$ is a set of polynomials such that $|Vars(f)| \leq \delta s/(2\ell)$ for every polynomial $f \in \mathcal{P}$. Suppose furthermore that $\mathcal{P}$ is represented by $(s, \delta, 0, Q)$-respectful boundary expander $(\mathcal{U}, \mathcal{V})_Q$ with slack $\xi = 0$ and overlap $ol(\mathcal{V}) = \ell$. To see that any polynomial calculus refutation of $\mathcal{P}$ requires degree strictly greater than $\delta s/(2\ell)$, just plug Lemma 4.35 into Theorem 4.11. □

5 Applications

In this section, we demonstrate how to use the machinery developed in Section 4 to establish degree lower bounds for polynomial calculus. Let us warm up by reproving the bound from [2] for CNF formulas $\mathcal{F}$ whose clause-variable incidence graphs $G(\mathcal{F})$ are good expanders in the sense of Definition 3.1. In this case, we can simply identify the $(\mathcal{U}, \mathcal{V})_Q$-graph with the standard clause-variable incidence graph $G(\mathcal{F})$ to recover the degree lower bound in [2] as stated next.

Theorem 5.1 ([2]). *Let $\mathcal{F}$ be a CNF formula such that the clause-variable incidence graph $G(\mathcal{F})$ is an (s, δ)-boundary expander for some $\delta > 0$. Then the polynomial calculus degree required to refute $\mathcal{F}$ is $Deg(\mathcal{F} \vdash \bot) > \delta s/2$.*

Proof. To choose $G(\mathcal{F})$ as our $(\mathcal{U}, \mathcal{V})_Q$-graph, we set Q to be the empty set, $\mathcal{U}$ to be the family of singleton sets of clauses of $\mathcal{F}$ translated to polynomials, and $\mathcal{V}$ to be the set of variables partitioned into singleton sets. As Q is an empty set every set V respects it. Also, every neighbour of some clause/polynomial $C \in \mathcal{U}$ is a Q-respectful neighbour because we can set the neighbouring variable so that the clause $C \in \mathcal{U}$ is satisfied. Under this interpretation $G(\mathcal{F})$ is an $(s, \delta, 0, Q)$-expander, and hence by Corollary 4.14 the degree of refuting $\mathcal{F}$ is greater than $\delta s/2$. □

As a second application, which is more interesting in the sense that the $(\mathcal{U}, \mathcal{V})_Q$-graph is nontrivial, we show how the degree lower bound for the ordering principle formulas in [22] can be presented in this framework. For an undirected (and, in general, non-bipartite) graph G, the *graph ordering principle formula $GOP(G)$* claims that there exists a totally ordered set of $|V(G)|$ elements where no element is minimal, as witnessed by the fact that every element/vertex v has a neighbour $u \in N(v)$ which is smaller according to the ordering. Formally, the CNF formula $GOP(G)$ is defined over variables $x_{u,v}$, $u, v \in V(G)$, $u \neq v$, where the intended meaning of the variables is that $x_{u,v}$ is true if $u < v$ according to the ordering, and it consists of the following axiom clauses:

$$\overline{x}_{u,v} \vee \overline{x}_{v,w} \vee x_{u,w} \qquad u, v, w \in V(G), u \neq v \neq w \neq u \qquad \text{(transitivity)} \tag{5.1a}$$

$$\overline{x}_{u,v} \vee \overline{x}_{v,u} \qquad u, v \in V(G), u \neq v \qquad \text{(asymmetry)} \tag{5.1b}$$

$$x_{u,v} \vee x_{v,u} \qquad u, v \in V(G), u \neq v \qquad \text{(totality)} \tag{5.1c}$$

$$\bigvee_{u \in N(v)} x_{u,v} \qquad v \in V(G) \qquad \text{(non-minimality)} \tag{5.1d}$$

We remark that the graph ordering principle on the complete graph K_n on n vertices is the *(linear) ordering principle formula LOP_n* (also known as a *least number principle formula*, or *graph tautology* in the literature), for which the non-minimality axioms (5.1d) have width linear in n. By instead considering graph ordering formulas for graphs G of bounded degree, one can bring the initial width of the formulas down so that the question of degree lower bounds becomes meaningful.

To prove degree lower bounds for $GOP(G)$ we need the following extension of boundary expansion to the case of non-bipartite graphs.

Definition 5.2 (Non-bipartite boundary expander). A graph $G = (V, E)$ is an (s, δ)*-boundary expander* if for every subset of vertices $V' \subseteq V(G)$, $|V'| \leq s$, it holds that $|\partial(V')| \geq \delta|V'|$, where the *boundary* $\partial(V') = \{v \in V(G) \setminus V' : |N(v) \cap V'| = 1\}$ is the set of all vertices in $V(G) \setminus V'$ that have a unique neighbour in V'.

We want to point out that the definition of expansion used in [22] is slightly weaker in that they do not require boundary expansion but just vertex expansion (measured as $|N(V') \setminus V'|$ for vertex sets V' with $|V'| \leq s$), and hence their result is slightly stronger than what we state below in Theorem 5.3. With some modifications of the definition of Q-respectful boundary in $(\mathcal{U}, \mathcal{V})_Q$-graphs it would be possible to match the lower bound in [22], but it would also make the definitions more cumbersome and so we choose not to do so here.

Theorem 5.3 ([22]). *For any non-bipartite graph G that is an (s, δ)-boundary expander it holds that $Deg(GOP(G) \vdash \bot) > \delta s/4$.*

Proof. To form the $(\mathcal{U}, \mathcal{V})_Q$-graph for $GOP(G)$, we let the set of polynomials Q consist of all transitivity axioms (5.1a), asymmetry axioms (5.1b), and totality axioms (5.1c). The non-minimality axioms (5.1d) viewed as singleton sets form the family $\mathcal{U}$, while $\mathcal{V}$ is the family of variable sets V_v for each vertex v containing all variables that mention v, i.e., $V_v = \{x_{u,w} \mid u, w \in V(G),\ u = v \text{ or } w = v\}$.

For a vertex u, the neighbours of a non-minimality axiom $P_u = \bigvee_{v \in N(u)} x_{v,u} \in \mathcal{U}$ are variable sets V_v where v is either equal to u or a neighbour of u in G. We can prove that each $V_v \in N(P_u)$ is an Q-respectful neighbour of P_u (although the particular neighbour V_u will not contribute in the proof of the lower bound). If $v \neq u$, then setting all the variables $x_{v,w} \in V_v$ to true and all the variables $x_{w,v} \in V_v$ to false (i.e., making v into the minimal element of the set) satisfies P_u as well as all the affected axioms in Q. If $v = u$, we can use a complementary assignment to the one above (i.e., making $v = u$ into the maximal element of the set) to Q-respectfully satisfy P_u. Observe that this also shows that all $V_v \in \mathcal{V}$ respect Q as required by Definition 4.4.

By the analysis above, it holds that the boundary $\partial(V')$ of some vertex set V' in G yields the Q-respectful boundary $\partial_Q\left(\bigcup_{u \in V'} P_u\right) \supseteq \{V_v \mid v \in \partial(V')\}$ in $(\mathcal{U}, \mathcal{V})_Q$. Thus, the expansion parameters for $(\mathcal{U}, \mathcal{V})_Q$ are the same as those for G and we can conclude that $(\mathcal{U}, \mathcal{V})_Q$ is an $(s, \delta, 0, Q)$-expander.

Finally, we note that while $\mathcal{V}$ is *not* a partition of the variables of $GOP(G)$, the overlap is only $ol(\mathcal{V}) = 2$ since every variable $x_{u,v}$ occurs in exactly two sets V_u and V_v in $\mathcal{V}$. Hence, by Corollary 4.14 the degree of refuting $GOP(G)$ is greater than $\delta s/4$. □

With the previous theorem in hand, we can obtain (a slightly weaker version of) the main result in [22], namely that there exists a family of 5-CNF formulas witnessing that the lower bound on size in terms of degree in Theorem 2.3 is essentially optimal. That is, there are formulas over N variables that can be refuted in polynomial calculus (in fact, in resolution) in size polynomial in N but require degree $\Omega(\sqrt{N})$. This follows by plugging expanders with suitable parameters into Theorem 5.3. By standard calculations (see, for example, [26]) one can show that there exist constants $\gamma, \delta > 0$ such that randomly sampled graphs on n vertices with degree at most 5 are $(\gamma n, \delta)$-boundary expanders in the sense of Definition 5.2 with high probability. By Theorem 5.3, graph ordering principle formulas on such graphs yield 5-CNF formulas over $\Theta(n^2)$ variables that require degree $\Omega(n)$. Since these formulas have polynomial calculus refutations in size $O(n^3)$ (just mimicking the resolution refutations constructed in [45]), this shows that the bound in Theorem 2.3 is essentially tight. The difference between this bound and [22] is that since Galesi and Lauria only require a weaker version of expansion they can use 3-regular graphs, yielding families of 3-CNF formulas (which is optimal, since any unsatisfiable 2-CNF formula can be refuted in width 2 in resolution).

Let us now turn our attention back to bipartite graphs and consider different flavours of pigeonhole principle formulas. We will focus on formulas over bounded-degree bipartite graphs, where we will convert standard bipartite boundary expansion as in Definition 3.1 into respectful boundary expansion as in Definition 4.8. Recall from Section 3.3 that given a bipartite graph $G = (U \dot{\cup} V, E)$, the *graph pigeonhole principle formula* $PHP(G)$ has axiom clauses (3.7a) and (3.7b); the *graph functional pigeonhole principle formula* $FPHP(G)$ contains the clauses of $PHP(G)$ plus axioms (3.7c); the *graph onto pigeonhole principle formula* $Onto\text{-}PHP(G)$ contains $PHP(G)$ plus axioms (3.7d); and the *graph onto functional pigeonhole principle formula* $Onto\text{-}FPHP(G)$ contains all the axiom clauses (3.7a)–(3.7d).

As mentioned in Section 1, it was established already in [2] that good bipartite boundary expanders G yield formulas $PHP(G)$ that require large polynomial calculus degree to refute. We can reprove this result in our language—and, in fact, observe that the lower bound in [2] works also for the onto version $Onto\text{-}PHP(G)$—by constructing an appropriate $(\mathcal{U}, \mathcal{V})_Q$-graph. In addition, we can generalize the result in [2] slightly by allowing some additive slack $\xi > 0$ in the expansion. This works as long as we have the guarantee that no too small subformulas are unsatisfiable.

Theorem 5.4 ([33]). *Let $G = (U \dot{\cup} V, E)$ be a bipartite graph with $|U| = n$, $|V| = n - 1$, and no isolated vertices in V. Suppose that $\delta > 0$ and $\xi \geq 0$ are constants such that for every set $U' \subseteq U$ of size $|U'| \leq s$ it holds that*

- *there is a matching of U' into V and*
- *$|\partial(U')| \geq \delta|U'| - \xi$.*

Then $Deg(Onto\text{-}PHP(G) \vdash \bot) > \delta s/2 - \xi$.

Sketch. The $(\mathcal{U}, \mathcal{V})_Q$-graph for $PHP(G)$ is formed by taking $\mathcal{U}$ to be the set of pigeon axioms (3.7a), Q to consist of the hole axioms (3.7b) and onto axioms (3.7d), and $\mathcal{V}$ to be the collection of variable sets $V_v = \{x_{u,v} \mid u \in N(v)\}$, i.e., the variables are partitioned with respect to the holes $v \in V$. It is straightforward to check that this $(\mathcal{U}, \mathcal{V})_Q$-graph is isomorphic to the graph G and that all neighbours in $(\mathcal{U}, \mathcal{V})_Q$ are Q-respectful (for $\bigvee_{v \in N(u)} x_{u,v} \in \mathcal{U}$ and V_v for some $v \in N(u)$, apply the partial assignment sending pigeon u to hole v and ruling out all other pigeons in $N(v) \setminus \{u\}$ for v). Moreover, using the existence of matchings for all sets of pigeons U' of size $|U'| \leq s$ we can prove that every subformula $\mathcal{U}' \wedge Q$ is satisfiable as long as $|\mathcal{U}'| \leq s$. (We note that we might need to send some pigeons to several holes in order to satisfy the onto axioms in Q, but this is in order since there are no functionality axioms forbidding pigeons to fly to multiple holes.) Hence, we can apply Theorem 4.11 to derive the claimed bound. □

Theorem 5.4 is the only place in this paper where we use non-zero slack for the expansion. We want to make clear that this slack parameter is not really essential for our pigeonhole principle lower bounds per se. Rather, the reason that we need slack is so that we can establish lower bounds for another family of formulas, namely the *subset cardinality formulas* studied in [33, 44, 47, 48], by reducing them to pigeonhole principle formulas on expanders where we have to allow for some slack. We discuss these formulas next.

A brief (and somewhat informal) description of the subset cardinality formulas is as follows. We start with a 4-regular bipartite graph to which we add an extra edge between a pair of non-connected vertices on the left and right. We then write down clauses stating that each degree-4 vertex on the left has at least 2 of its edges set to true, while the single degree-5 vertex has a strict majority of 3 incident edges set to true. On the right-hand side of the graph we encode the opposite, namely that all vertices with degree 4 have at least 2 of its edges set to false, while the vertex with degree 5 has at least 3 edges set to false. A simple counting argument yields that the CNF formula consisting of these clauses must be unsatisfiable. Formally, we have the following definition (which strictly speaking is a slightly specialized case of the general construction, but we refer to [33] for the details).

Definition 5.5 (Subset cardinality formulas). Suppose that $G = (U \mathbin{\dot\cup} V, E)$ is a bipartite graph with $|U| = |V|$ that is 4-regular on both sides except that one extra edge has been added between a pair of otherwise unconnected vertices $u \in U$ and $v \in V$ on the left and right. Then the *subset cardinality formula $SC(G)$* over G has variables x_e, $e \in E$, and clauses:

- $x_{e_1} \lor x_{e_2} \lor x_{e_3}$ for every triple e_1, e_2, e_3 of edges incident to any $u \in U$,
- $\overline{x}_{e_1} \lor \overline{x}_{e_2} \lor \overline{x}_{e_3}$ for every triple e_1, e_2, e_3 of edges incident to any $v \in V$.

To prove lower bounds on refutation degree for these formulas we use the standard notion of vertex expansion on bipartite graphs, where all neighbours on the left are counted and not just unique neighbours as in Definition 3.1.

Definition 5.6 (Bipartite expander). We say that a bipartite graph $G = (U \mathbin{\dot\cup} V, E)$ is a *bipartite (s, δ)-expander* if for each vertex set $U' \subseteq U$, $|U'| \leq s$, it holds that $|N(U')| \geq \delta |U'|$.

The existence of such expanders with appropriate parameters can again be established by straightforward calculations (as in, for instance, [26]).

Theorem 5.7 ([33]). *Suppose that $G = (U \mathbin{\dot\cup} V, E)$ is a 4-regular bipartite $\left(\gamma n, \frac{5}{2} + \delta\right)$-expander for $|U| = |V| = n$ and some constants $\gamma, \delta > 0$, and let G' be obtained from G by adding an arbitrary edge between two unconnected vertices in U and V. Then refuting the formula $SC(G')$ requires degree $Deg(SC(G') \vdash \bot) = \Omega(n)$, and hence size $S_{\mathcal{PCR}}(SC(G') \vdash \bot) = \exp\big(\Omega(n)\big)$.*

sketch. The proof is by reducing to graph PHP formulas and applying Theorem 5.4 (which of course also holds with onto axioms removed). We fix some complete matching in G, which is guaranteed to exist in regular bipartite graphs, and then set all edges in the matching as well as the extra added edge to true. Now the degree-5 vertex v^* on the right has only 3 neighbours and the constraint for v^* requires all of these edges to be set to false. Hence, we set these edges to false as well which makes v^* and its clauses vanish from the formula. The restriction leaves us with n vertices on the left which require that at least 1 of the remaining 3 edges incident to them is true, while the $n - 1$ vertices on the right require that at most 1 out of their incident edges is true. That is, we have restricted our subset cardinality formula to obtain a graph PHP formula.

As the original graph is a $(\gamma n, \frac{5}{2} + \delta)$-expander, a simple calculation can convince us that the new graph is a boundary expander where each set of vertices U' on the left with size $|U'| \leq \gamma n$ has boundary expansion $|\partial(U')| \geq 2\delta|U'| - 1$. Note the additive slack of 1 compared to the usual expansion condition, which is caused by the removal of the degree-5 vertex v^* from the right. Now we can appeal to Theorem 5.4 (and then Theorem 2.3) to obtain the lower bounds claimed in the theorem. □

Let us conclude this section by presenting our new lower bounds for the functional pigeonhole principle formulas. As a first attempt, we could try to reason as in the proof of Theorem 5.4 (but adding the axioms (3.7c) and removing axioms (3.7d)). The naive idea would be to modify our $(\mathcal{U}, \mathcal{V})_Q$-graph slightly by substituting the functionality axioms for the onto axioms in Q while keeping $\mathcal{U}$ and $\mathcal{V}$ the same. This does not work, however—although the sets $V_v \in \mathcal{V}$ are Q-respectful, the only assignment to the variables in V_v that respects Q is the one that sets all variables $x_{u,v} \in V_v$ to false. This is because setting any $x_{u,v}$ to true would affect functional axioms that mention u and that cannot be satisfied by setting only the variables in V_v. Thus, it is not possible to satisfy any of the pigeon axioms, meaning that there are no Q-respectful neighbours in $(\mathcal{U}, \mathcal{V})_Q$. In order to obtain a useful $(\mathcal{U}, \mathcal{V})_Q$-graph, we instead need to redefine $\mathcal{V}$ by enlarging the variable sets V_v, using the fact that $\mathcal{V}$ is not required to be a partition. Doing so in the appropriate way yields the following theorem.

Theorem 5.8. *Suppose that $G = (U \dot\cup V, E)$ is a bipartite (s, δ)-boundary expander with left degree bounded by d. Then it holds that refuting FPHP(G) in polynomial calculus requires degree strictly greater than $\delta s/(2d)$. It follows that if G is a bipartite $(\gamma n, \delta)$-boundary expander with constant left degree and $\gamma, \delta > 0$, then any polynomial calculus (PC or PCR) refutation of FPHP(G) requires size* $\exp(\Omega(n))$.

Proof. We construct a $(\mathcal{U}, \mathcal{V})_Q$-graph from *FPHP*$(G)$ as follows. We let the set of polynomials Q consist of all hole axioms (3.7b) and functionality axioms (3.7c). We define the family $\mathcal{U}$ to consist of the pigeon axioms (3.7a) interpreted as singleton polynomials. For the variables we let $\mathcal{V} = \{V_v \mid v \in V\}$, where for every hole $v \in V$ the set V_v is defined by

$$V_v = \left\{x_{u',v'} \,\middle|\, u' \in N(v) \text{ and } v' \in N(u')\right\}. \tag{5.2}$$

That is, to build V_v we start with the hole v on the right, consider all pigeons u' on the left that can go into this hole, and finally include in V_v for all such u' the variables $x_{u',v'}$ for all holes v' incident to u'. We want to show that $(\mathcal{U}, \mathcal{V})_Q$ as defined above satisfies the conditions in Corollary 4.14.

Note first that every variable set V_v respects the set Q since setting all variables in V_v to false satisfies all clauses/polynomials in Q mentioning variables in V_v. It is easy to see from (5.2) that when a hole v is a neighbour of a pigeon u, the variable set V_v is also a neighbour in the $(\mathcal{U}, \mathcal{V})_Q$-graph of the corresponding pigeon axiom $P_u = \bigvee_{v \in N(u)} x_{u,v}$. These are the only neighbours of the pigeon axiom P_u, as each V_v contains only variables mentioning pigeons in the neighbourhood of v. In other words, G and $(\mathcal{U}, \mathcal{V})_Q$ share the same neighbourhood structure.

Moreover, we claim that every neighbour V_v of P_u is a Q-respectful neighbour. To see this, consider the assignment $\rho_{u,v}$ that sets $x_{u,v}$ to true and the remaining variables in V_v to false. Clearly, P_u is satisfied by $\rho_{u,v}$. All axioms in Q *not* containing $x_{u,v}$ are either satisfied by $\rho_{u,v}$ or left untouched, since $\rho_{u,v}$ assigns all other variables in its domain to false. Any hole axiom $\overline{x}_{u,v} \vee \overline{x}_{u',v}$ in Q that *does* contain $x_{u,v}$ is satisfied by $\rho_{u,v}$ since $x_{u',v} \in V_v$ for $u' \in N(v)$ by (5.2) and this variable is set to false by $\rho_{u,v}$. In the same way, any functionality axiom $\overline{x}_{u,v} \vee \overline{x}_{u,v'}$ containing $x_{u,v}$ is satisfied since the variable $x_{u,v'}$ is in V_v by (5.2) and is hence assigned to false. Thus, the assignment $\rho_{u,v}$ Q-respectfully satisfies P_u, and so P_u and V_v are Q-respectful neighbours as claimed.

Since our constructed $(\mathcal{U}, \mathcal{V})_Q$-graph is isomorphic to the original bipartite graph G and all neighbour relations are respectful, the expansion parameters of G trivially carry over to respectful expansion in $(\mathcal{U}, \mathcal{V})_Q$. This is just another way of saying that $(\mathcal{U}, \mathcal{V})_Q$ is an $(s, \delta, 0, Q)$-respectful boundary expander.

To finish the proof, note that the overlap of $\mathcal{V}$ is at most d. This is so since a variable $x_{u,v}$ appears in a set $V_{v'}$ only when $v' \in N(u)$. Hence, for all variables $x_{u,v}$ it holds that they appear in at most $|N(u)| \leq d$ sets in $\mathcal{V}$. Now the conclusion that any polynomial calculus refutation of *FPHP*(G) requires degree greater than $\delta s/(2d)$ can be read off from Corollary 4.14. In addition, the exponential lower bound on the size of a refutation of *FPHP*(G) when G is a $(\gamma n, \delta)$-boundary expander G with constant left degree follows by plugging the degree lower bound into Theorem 2.3. □

It is not hard to show (again we refer to [26] for the details) that there exist bipartite graphs with left degree 3 which are $(\gamma n, \delta)$-boundary expanders for $\gamma, \delta > 0$ and hence our size lower bound for polynomial calculus refutations of *FPHP*(G) can be applied to them. Moreover, if $|U| = n + 1$ and $|V| = n$, then we can identify some bipartite graph G that is a good expander and hit $\mathit{FPHP}_n^{n+1} = \mathit{FPHP}(K_{n+1,n})$ with a restriction ρ_G setting $x_{u,v}$ to false for all $(u, v) \notin E$ to obtain $\mathit{FPHP}_n^{n+1}\restriction_{\rho_G} = \mathit{FPHP}(G)$. Since restrictions can only decrease refutation size, it follows that size lower bounds for *FPHP*(G) apply also to FPHP_n^{n+1}, yielding the second lower bound claimed in Section 1.3.

Theorem 5.9. *Any polynomial calculus or polynomial calculus resolution refutation of (the standard CNF encoding of) the functional pigeonhole principle* FPHP_n^{n+1} *requires size* $\exp(\Omega(n))$.

6 Concluding Remarks

In this work, we extend the techniques developed by Alekhnovich and Razborov [2] for proving degree lower bounds on refutations of CNF formulas in polynomial calculus. Instead of looking at the clause-variable incidence graph $G(\mathcal{F})$ of the formula $\mathcal{F}$ as in [2], we allow clustering of clauses and variables and reason in terms of the incidence graph G' defined on these clusters. We show that the (canonical translation to polynomials of the) CNF formula $\mathcal{F}$ requires high degree to be refuted in polynomial calculus whenever this clustering can be done in a way that "respects the structure" of the formula and so that the resulting graph G' has certain expansion properties. We also prove similar lower bounds for more general sets of polynomials not obtained as translations of CNF formulas.

This provides us with a unified framework within which we can reprove previously established degree lower bounds in [2, 22, 33]. More importantly, this also allows us to obtain a degree lower bound on the functional pigeonhole principle defined on expander graphs, solving an open problem from [39]. It immediately follows from this that the (standard CNF encodings of) the usual functional pigeonhole principle formulas require exponential proof size in polynomial calculus resolution, resolving a question on Razborov's problems list [42] which had (quite annoyingly) remained open. This means that we now have an essentially complete understanding of how the different variants of pigeonhole principle formulas behave with respect to polynomial calculus in the standard setting with $n + 1$ pigeons and n holes. Namely, while onto-FPHP formulas are easy, both FPHP formulas and onto-PHP formulas are exponentially hard in n even when restricted to bounded-degree expanders.

A natural next step would be to see if this generalized framework can also be used to attack other interesting formula families that are known to be hard for resolution but for which there are currently no lower bounds in polynomial calculus. In particular, can our framework or some modification of it be used to obtain average-case lower bounds for graph problems like independent set, vertex cover, or colourability,[13] which were proven hard for resolution in [5, 6]? The area of Ramsey theory has a lot of interesting problems in combinatorics and these problems can be studied from the perspective of proof complexity, via establishing lower bounds on the lengths of proofs for such problems. There already exist some lower bounds for resolution [14, 32, 35], and we would like to prove the same bounds in polynomial calculus. While these papers use somewhat different strategies, they are all based on some type of a width lower bound and hence might yield to degree lower bound techniques. A combinatorial characterization of resolution width was given in [3] and used to prove a lower bound on the dense ordering principle. Is it possible to find a similar characterization for the degree in polynomial calculus and use it to prove lower bounds?

Returning to the pigeonhole principle, we now understand how different encodings behave in polynomial calculus when we have $n + 1$ pigeons and n holes. But what happens when we increase the number of pigeons m? For instance, do the formulas become easier if we have $m = n^2$ pigeons and n holes? (This is the point where lower bound techniques based on degree break down.) What about arbitrary many pigeons? In resolution these questions are fairly well understood, as witnessed by the works of Raz [36], Razborov [38, 40, 41], and de Rezende et al. [19], but as far as we are aware they remain wide open for polynomial calculus. Another question is how hard onto functional PHP formulas are for polynomial calculus when the difference $m - n$ between the number of pigeons and holes is divisible by the characteristic of the underlying field. This problem was studied for the weaker proof system Nullstellensatz by Beame and Riis [7], but we are not aware of any lower bounds for polynomial calculus.

Finally, we want to point out an intriguing contrast between our work and that of Alekhnovich and Razborov. As discussed in the introduction, the main technical result in [2] is that when the incidence graph of a set of polynomial equations is expanding and the polynomials are immune, i.e., have no low-degree consequences, then refuting this set of equations is hard with respect to polynomial calculus degree. Since clauses of width w have maximal immunity w, it follows that for a CNF formula $\mathcal{F}$ expansion of the clause-variable incidence graph $G(\mathcal{F})$ is enough to imply hardness. A natural way of interpreting our work would be to say that we simply extend this result to a slightly more general constraint-variable incidence graph. On closer inspection, however, this analogy seems to be misleading, and since the authors have to confess to still being somewhat intrigued by this, we want to elaborate briefly.

For the functional pigeonhole principle, the pigeon and functional axioms for a pigeon u taken together imply the polynomial equation $\sum_{v \in N(u)} x_{u,v} = 1$ (summing over all holes $v \in N(u)$ to which the pigeon u can fly). Since this is a degree-1 consequence, it shows that the pigeonhole axioms in FPHP formulas have *lowest possible immunity* modulo the set Q consisting of hole and functionality axioms. Nevertheless, our lower bound proof still works, and only needs expansion of the constraint-variable graph although the immunity of the constraints is non-existent.

On the other hand, the constraint-variable incidence graph of a random set of parity constraints is expanding asymptotically almost surely, and since over fields of characteristic distinct from 2 parity constraints have high immunity (see, for instance, [23]), the techniques in [2] can be used to prove strong degree lower bounds in such a setting. However, it seems that our framework of respectful boundary expansion is inherently unable to establish this result. The problem is that it is not possible to group variables together in such a way as to ensure respectful neighbourhood relations. At a high level, it seems that the main ingredient needed for our technique to work is that clauses/polynomials and variables can be grouped together so that the effects of assignments to a

[13] As the final version of this manuscript was being prepared, average-case lower bounds for graph colouring of random graphs were published in [17], improving on worst-case lower bounds previously obtained in [4, 31]. The proof techniques in [17] follow [2] at a high level, but it is unclear whether they can be expressed in terms of clause-variable incidence graphs as in the current paper.

group of variables can always be contained in a small neighbourhood of clauses/polynomials, which the assignments (mostly) satisfy, and do not propagate beyond this neighbourhood. Functional pigeonhole principle formulas over bounded-degree graphs have this property, since assigning a pigeon u to a hole v only affects the neighbouring holes of u and the neighbouring pigeons of v, respectively. There is no such way to contain the effects locally when one starts satisfying individual equations in an expanding set of parity constraints, however, regardless of the characteristic of the underlying field.

In view of this, it seems that our techniques and those of [2] are closer to being orthogonal than parallel. It would be desirable to gain a deeper understanding of what is going on here. In particular, in comparison to [2], which gives clear, explicit criteria for hardness (is the graph expanding? are the polynomials immune?), our work is less explicit in that it says that hardness is implied by the existence of a "clustered clause-variable incidence graph" with the right properties, but gives no guidance as to if and how such a graph might be built. It would be satisfying if some more general criterion of hardness could be found that would capture both our approach and that of [2], and that would ideally provide a unified view of these lower bound techniques.

Acknowledgments

We are grateful to Ilario Bonacina, Yuval Filmus, Nicola Galesi, Massimo Lauria, Alexander Razborov, and Marc Vinyals for numerous discussions on proof complexity in general and polynomial calculus degree lower bounds in particular. We want to give a special thanks to Massimo Lauria for several insightful comments on an earlier version of this work, which allowed us to simplify the construction (and improve the parameters in the results) considerably, and to Alexander Razborov for valuable remarks on a preliminary version of this manuscript that, in particular, helped to shed light on the similarities with and differences from the techniques in [2]. We thank Leszek Kołodziejczyk for bringing the manuscript [49] to our attention. We also want to acknowledge useful discussions at the Dagstuhl workshop 15171 *Theory and Practice of SAT Solving* in April 2015 and at the Simons Institute for the Theory of Computing at UC Berkeley during the semester program *Lower Bounds in Computational Complexity* in the autumn of 2018. We wish to thank Yassine Ghannane, Rui Ji, Shuo Pang, Kilian Risse, and Dmitry Sokolov for helpful comments during the preparations of the final version of this manuscript. Finally, we are most thankful for the very detailed feedback provided by the anonymous reviewers, which helped improve the exposition considerably.

The authors were funded by the European Research Council under the European Union's Seventh Framework Programme (FP7/2007–2013) / ERC grant agreement no. 279611. The second author was also supported by the Knut and Alice Wallenberg grant KAW 2016.0066, the Swedish Research Council grants 621-2010-4797, 621-2012-5645, and 2016-00782, and the Independent Research Fund Denmark grant 9040-00389B.

References

[1] Michael Alekhnovich, Eli Ben-Sasson, Alexander A. Razborov, and Avi Wigderson. 2002. Space Complexity in Propositional Calculus. *SIAM J. Comput.* 31, 4 (April 2002), 1184–1211. Preliminary version in *STOC '00*.

[2] Michael Alekhnovich and Alexander A. Razborov. 2003. Lower Bounds for Polynomial Calculus: Non-Binomial Case. *Proceedings of the Steklov Institute of Mathematics* 242 (2003), 18–35. Available at http://people.cs.uchicago.edu/~razborov/files/misha.pdf. Preliminary version in *FOCS '01*..

[3] Albert Atserias and Víctor Dalmau. 2008. A Combinatorial Characterization of Resolution Width. *J. Comput. System Sci.* 74, 3 (May 2008), 323–334. Preliminary version in *CCC '03*.

[4] Albert Atserias and Joanna Ochremiak. 2019. Proof Complexity Meets Algebra. *ACM Transactions on Computational Logic* 20 (Feb. 2019), 1:1–1:46. Preliminary version in *ICALP '17*.

[5] Paul Beame, Joseph C. Culberson, David G. Mitchell, and Cristopher Moore. 2005. The Resolution Complexity of Random Graph k-Colorability. *Discrete Applied Mathematics* 153, 1-3 (Dec. 2005), 25–47.

[6] Paul Beame, Russell Impagliazzo, and Ashish Sabharwal. 2007. The Resolution Complexity of Independent Sets and Vertex Covers in Random Graphs. *Computational Complexity* 16, 3 (Oct. 2007), 245–297. Preliminary version in *CCC '01*.

[7] Paul Beame and Søren Riis. 1998. More on the Relative Strength of Counting Principles. In *Proof Complexity and Feasible Arithmetics (DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Vol. 39)*. American Mathematical Society, 13–35. Available at https://homes.cs.washington.edu/~beame/papers/riis.pdf.

[8] Eli Ben-Sasson and Russell Impagliazzo. 2010. Random CNF's are Hard for the Polynomial Calculus. *Computational Complexity* 19, 4 (2010), 501–519. Preliminary version in *FOCS '99*.

[9] Eli Ben-Sasson and Avi Wigderson. 2001. Short Proofs are Narrow—Resolution Made Simple. *J. ACM* 48, 2 (March 2001), 149–169. Preliminary version in *STOC '99*.

[10] Armin Biere, Marijn J. H. Heule, Hans van Maaren, and Toby Walsh (Eds.). 2021. *Handbook of Satisfiability* (2nd ed.). Frontiers in Artificial Intelligence and Applications, Vol. 336. IOS Press.

[11] Archie Blake. 1937. *Canonical Expressions in Boolean Algebra.* Ph. D. Dissertation. University of Chicago.

[12] Samuel R. Buss, Dima Grigoriev, Russell Impagliazzo, and Toniann Pitassi. 2001. Linear Gaps between Degrees for the Polynomial Calculus Modulo Distinct Primes. *J. Comput. System Sci.* 62, 2 (March 2001), 267–289. Preliminary version in *CCC '99*.

[13] Samuel R. Buss and Jakob Nordström. 2021. Proof Complexity and SAT Solving. See [10], Chapter 7, 233–350.

[14] Lorenzo Carlucci, Nicola Galesi, and Massimo Lauria. 2016. On the Proof Complexity of Paris-Harrington and Off-Diagonal Ramsey Tautologies. *ACM Transactions on Computational Logic* 17, 4, Article 26 (Nov. 2016), 26:1–26:25 pages.

[15] Vašek Chvátal and Endre Szemerédi. 1988. Many Hard Examples for Resolution. *J. ACM* 35, 4 (Oct. 1988), 759–768.

[16] Matthew Clegg, Jeffery Edmonds, and Russell Impagliazzo. 1996. Using the Groebner Basis Algorithm to Find Proofs of Unsatisfiability. In *Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC '96)*. 174–183.

[17] Jonas Conneryd, Susanna F. de Rezende, Jakob Nordström, Shuo Pang, and Kilian Risse. 2023. Graph Colouring Is Hard on Average for Polynomial Calculus and Nullstellensatz. In *Proceedings of the 64th Annual IEEE Symposium on Foundations of Computer Science (FOCS '23)*. 1–11.

[18] Stephen A. Cook and Robert A. Reckhow. 1979. The Relative Efficiency of Propositional Proof Systems. *Journal of Symbolic Logic* 44, 1 (March 1979), 36–50. Preliminary version in *STOC '74*.

[19] Susanna F. de Rezende, Jakob Nordström, Kilian Risse, and Dmitry Sokolov. 2020. Exponential Resolution Lower Bounds for Weak Pigeonhole Principle and Perfect Matching Formulas over Sparse Graphs. In *Proceedings of the 35th Annual Computational Complexity Conference (CCC '20) (Leibniz International Proceedings in Informatics (LIPIcs), Vol. 169)*. 28:1–28:24.

[20] Yuval Filmus. 2019. Another Look at Degree Lower Bounds for Polynomial Calculus. *Theoretical Computer Science* 796 (Dec. 2019), 286–293.

[21] Nicola Galesi and Massimo Lauria. 2010. On the Automatizability of Polynomial Calculus. *Theory of Computing Systems* 47, 2 (Aug. 2010), 491–506.

[22] Nicola Galesi and Massimo Lauria. 2010. Optimality of Size-Degree Trade-offs for Polynomial Calculus. *ACM Transactions on Computational Logic* 12, 1, Article 4 (Nov. 2010), 4:1–4:22 pages.

[23] Frederic Green. 2000. A Complex-Number Fourier Technique for Lower Bounds on the Mod-m degree. *Computational Complexity* 9, 1 (Jan. 2000), 16–38.

[24] Dima Grigoriev. 1998. Tseitin's Tautologies and Lower Bounds for Nullstellensatz Proofs. In *Proceedings of the 39th Annual IEEE Symposium on Foundations of Computer Science (FOCS '98)*. 648–652.

[25] Armin Haken. 1985. The Intractability of Resolution. *Theoretical Computer Science* 39, 2-3 (Aug. 1985), 297–308.

[26] Shlomo Hoory, Nathan Linial, and Avi Wigderson. 2006. Expander Graphs and Their Applications. *Bull. Amer. Math. Soc.* 43, 4 (Oct. 2006), 439–561.

[27] Russell Impagliazzo, Pavel Pudlák, and Jiří Sgall. 1999. Lower Bounds for the Polynomial Calculus and the Gröbner Basis Algorithm. *Computational Complexity* 8, 2 (1999), 127–144.

[28] Dmitry Itsykson, Vsevolod Oparin, Mikhail Slabodkin, and Dmitry Sokolov. 2016. Tight Lower Bounds on the Resolution Complexity of Perfect Matching Principles. *Fundamenta Informaticae* 145, 3 (Aug. 2016), 229–242. Preliminary version in *CSR '15*.

[29] Hans Kleine Büning and Oliver Kullmann. 2021. Minimal Unsatisfiability and Autarkies. See [10], Chapter 14, 571–633.

[30] Jan Krajíček. 2019. *Proof Complexity.* Encyclopedia of Mathematics and Its Applications, Vol. 170. Cambridge University Press.

[31] Massimo Lauria and Jakob Nordström. 2017. Graph Colouring is Hard for Algorithms Based on Hilbert's Nullstellensatz and Gröbner Bases. In *Proceedings of the 32nd Annual Computational Complexity Conference (CCC '17) (Leibniz International Proceedings in Informatics (LIPIcs), Vol. 79)*. 2:1–2:20.

[32] Massimo Lauria, Pavel Pudlák, Vojtěch Rödl, and Neil Thapen. 2017. The Complexity of Proving That a Graph is Ramsey. *Combinatorica* 37, 2 (April 2017), 253–268. Preliminary version in *ICALP '13*.

[33] Mladen Mikša and Jakob Nordström. 2014. Long Proofs of (Seemingly) Simple Formulas. In *Proceedings of the 17th International Conference on Theory and Applications of Satisfiability Testing (SAT '14) (Lecture Notes in Computer Science, Vol. 8561)*. Springer, 121–137.

[34] Mladen Mikša and Jakob Nordström. 2015. A Generalized Method for Proving Polynomial Calculus Degree Lower Bounds. In *Proceedings of the 30th Annual Computational Complexity Conference (CCC '15) (Leibniz International Proceedings in Informatics (LIPIcs), Vol. 33)*. 467–487.

[35] Pavel Pudlák. 2012. A Lower Bound on the Size of Resolution Proofs of the Ramsey Theorem. *Inform. Process. Lett.* 112, 14–15 (Aug. 2012), 610–611.

[36] Ran Raz. 2004. Resolution Lower Bounds for the Weak Pigeonhole Principle. *J. ACM* 51, 2 (March 2004), 115–138. Preliminary version in *STOC '02*.

[37] Alexander A. Razborov. 1998. Lower Bounds for the Polynomial Calculus. *Computational Complexity* 7, 4 (Dec. 1998), 291–324.

[38] Alexander A. Razborov. 2001. *Improved Resolution Lower Bounds for the Weak Pigeonhole Principle.* Technical Report TR01-055. Electronic Colloquium on Computational Complexity (ECCC).

[39] Alexander A. Razborov. 2002. Proof Complexity of Pigeonhole Principles. In *5th International Conference on Developments in Language Theory, (DLT '01), Revised Papers (Lecture Notes in Computer Science, Vol. 2295)*. Springer, 100–116.

[40] Alexander A. Razborov. 2003. Resolution Lower Bounds for the Weak Functional Pigeonhole Principle. *Theoretical Computer Science* 1, 303 (June 2003), 233–243.

[41] Alexander A. Razborov. 2004. Resolution Lower Bounds for Perfect Matching Principles. *J. Comput. System Sci.* 69, 1 (Aug. 2004), 3–27. Preliminary version in *CCC '02*.

[42] Alexander A. Razborov. 2014. Possible research directions. List of open problems (in proof complexity and other areas) available at http://people.cs.uchicago.edu/~razborov/teaching/.

[43] Søren Riis. 1993. *Independence in Bounded Arithmetic.* Ph. D. Dissertation. University of Oxford.

[44] Ivor Spence. 2010. sgen1: A Generator of Small but Difficult Satisfiability Benchmarks. *Journal of Experimental Algorithmics* 15, Article 1.2 (March 2010), 1.2:1–1.2:15 pages.

[45] Gunnar Stålmarck. 1996. Short Resolution Proofs for a Sequence of Tricky Formulas. *Acta Informatica* 33, 3 (May 1996), 277–280.

[46] Alasdair Urquhart. 1987. Hard Examples for Resolution. *J. ACM* 34, 1 (Jan. 1987), 209–219.

[47] Allen Van Gelder and Ivor Spence. 2010. Zero-One Designs Produce Small Hard SAT Instances. In *Proceedings of the 13th International Conference on Theory and Applications of Satisfiability Testing (SAT '10) (Lecture Notes in Computer Science, Vol. 6175)*. Springer, 388–397.

[48] Marc Vinyals, Jan Elffers, Jesús Giráldez-Cru, Stephan Gocht, and Jakob Nordström. 2018. In Between Resolution and Cutting Planes: A Study of Proof Systems for Pseudo-Boolean SAT Solving. In *Proceedings of the 21st International Conference on Theory and Applications of Satisfiability Testing (SAT '18) (Lecture Notes in Computer Science, Vol. 10929)*. Springer, 292–310.

[49] Łukasz Wołochowski. 2013. *An Application of Expanders to the Study of Proofs of the Pigeonhole Principle in Algebraic Propositional Proof Systems.* Master's thesis. University of Warsaw.

Received February 20XX; revised March 20YY; accepted April 20ZZ