Electronic Colloquium on Computational Complexity, Report No. 78 (2015)

A Generalized Method for Proving Polynomial Calculus Degree Lower Bounds*

Mladen Mikša
KTH Royal Institute of Technology

Jakob Nordström
KTH Royal Institute of Technology

May 4, 2015

Abstract

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 the techniques in [AR03] 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. As a corollary of this, 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.

1 Introduction

In one sentence, proof complexity studies how hard it is to certify the unsatifiability 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.

1.1 Resolution and Polynomial Calculus

Perhaps the most well-studied proof system in proof complexity is *resolution* [Bla37], 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 [Hak85, Urq87, CS88]). Many 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 result by Ben-Sasson and Wigderson [BW01] that any resolution proof must also consist of very many clauses. Research since [BW01] 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 slightly more general proof system *polynomial calculus resolution (PCR)*. This proof system was introduced by Clegg et al. [CEI96] in a slightly weaker form that is usually referred to as *polynomial calculus (PC)* and was later extended by Alekhnovich et al. [ABRW02].

*This is the full-length version of the paper with the same title to appear in *Proceedings of the 30th Annual Computational Complexity Conference (CCC '15)*.

ISSN 1433-8092

In PC and PCR clauses are translated to multilinear polynomials over some (fixed) field $\mathbb{F}$, and a CNF formula 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 F. Here the size of a proof is measured as the number of monomials in a proof when all polynomials 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 polynomial calculus and polynomial calculus resolution 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 [BW01], Impagliazzo et al. [IPS99] showed that strong lower bounds on the degree of PC proofs are sufficient to establish strong size lower bounds. The same proof goes through 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 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 [BW01], the paper [IPS99] 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 [Gri98, BGIP01]. For fields of arbitrary characteristic Alekhnovich and Razborov [AR03] developed a powerful 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. There are many formula families for which this is not true, however. One can have a family of constraint satisfaction problems where the constraint-variable incidence graph is an expander—say, for instance, for an unsatisfiable set of linear equations $\bmod 2$—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 $\bmod 2$ 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]

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 [CR79], has been to prove superpolynomial lower bounds on proof size for increasingly stronger 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 proof system by studying different combinatorial principles (encoded in CNF) and determining how hard they are to prove for this proof system.

[1] In a bit more detail, what is shown in [AR03] 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 that this set of polynomial equations is inconsistent in polynomial calculus 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 $\bmod 2$ has a low-degree consequence over $\mathbb{F}_2$, however—namely, the linear equation itself—and this is why [AR03] (correctly) fails to prove lower bounds in this case.

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* perhaps 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 [Raz02] 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 [Hak85] for formulas with $m = n + 1$ pigeons can be made to work also for onto-FPHP formulas, and more recent works by Raz [Raz04a] and Razborov [Raz03, Raz04b] 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 [Raz98], but for a different encoding than the standard translation from CNF, since translating wide clauses yields initial polynomials of high degree. Alekhnovich and Razborov [AR03] 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 [Rii93] the onto-FPHP formulas with $m = n + 1$ pigeons are easy for polynomial calculus. And while the encoding in [Raz98] also captures the functionality restriction in some sense, it has remained 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 [Raz15]).

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 $H = (U \dot{\cup} V, E)$ 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 [BW01, IPS99] to obtain exponential proof size lower bounds from linear width/degree lower bounds. A careful reading of the proofs in [AR03] 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 H. Razborov lists as one of the open problems in [Raz02] 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 [AR03] for proving polynomial calculus degree lower bounds, restricting our attention to the special case when the polynomials are obtained by the canonical translation of CNF formulas.

Instead of considering 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 construct a new graph G' by clustering several clauses and/or variables into single vertices, reflecting the structure of the combinatorial principle the CNF formula F is encoding. 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 such a clustering is already implicit in, for instance, the resolution lower bounds in [BW01] for Tseitin formulas (which is essentially just a special form of unsatisfiable linear equations) and graph PHP formulas, as well as in the graph PHP lower bound for polynomial calculus in [AR03].

We then show that if this clustering is done in the right way, the proofs in [AR03] still go through and yield strong polynomial calculus degree lower bounds when G' is a good enough expander.[2] It is clear that this cannot work in general—as already discussed above, any inconsistent system of linear equations $\bmod 2$ 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 clauses F' 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 ρ satisfies all of F' and such that for the clauses outside of F' they are either satisfied by ρ or left completely untouched by ρ. 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 [AR03] makes it possible to present previously obtained polynomial calculus degree lower bounds in [AR03, GL10, MN14] in a unified framework. Moreover, it allows us to prove the following new results:

1. If a bipartite graph $H = (U \dot\cup V, E)$ with $|U| = m$ and $|V| = n$ 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 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.

1.4 Organization of This Paper

After reviewing the necessary preliminaries in Section 2, we present our extension of the Alekhnovich–Razborov method in Section 3. In Section 4, 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 5 by discussing some possible directions for future research.

2 Preliminaries

Let us start by giving an overview of the relevant proof complexity background. This material is standard and we refer to, for instance, the survey [Nor13] for more details.

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 0 with true and 1 with false. We remark that this is the opposite of the standard convention in proof complexity, but it is a more natural choice in the context of polynomial calculus, where "evaluating to true" means "vanishing." A *clause* $C = a_1 \vee \cdots \vee a_k$ is a

[2] For a certain twist of the definition of expander that we do not describe in full detail here in order to keep the discussion at an informal, intuitive level. The formal description is given in Section 3.1.

disjunction of literals. A *CNF formula* $F = C_1 \wedge \cdots \wedge C_m$ is a conjunction of clauses. The *width* $W(C)$ of a clause C is the number of literals $|C|$ in it, and the width $W(F)$ of the formula F is the maximum width of any clause in the formula. We think of clauses and CNF formulas as sets, so that order is irrelevant and there are no repetitions. A k-CNF formula has all clauses of size at most k, where k is assumed to be some fixed constant.

In polynomial calculus resolution the goal is to prove the unsatisfiability of a CNF formula by reasoning with polynomials from a polynomial ring $\mathbb{F}[x, \overline{x}, y, \overline{y}, \ldots]$ (where x and $\overline{x}$ are viewed as distinct formal variables) over some fixed field $\mathbb{F}$. The results in this paper hold for all fields $\mathbb{F}$ regardless of characteristic. 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.

Definition 2.1 (Polynomial calculus resolution (PCR) [CEI96, ABRW02]). A *polynomial calculus resolution (PCR) refutation* $\pi : F \vdash \bot$ of a CNF formula F (also referred to as a *PCR proof* for F) over a field $\mathbb{F}$ is an ordered sequence of polynomials $\pi = (P_1, \ldots, P_\tau)$, expanded out as linear combinations of monomials, such that $P_\tau = 1$ and each line P_i, $1 \leq i \leq \tau$, is either

- a monomial $\prod_{x \in L^+} x \cdot \prod_{y \in L^-} \overline{y}$ encoding a clause $\bigvee_{x \in L^+} x \vee \bigvee_{y \in L^-} \overline{y}$ in F (a *clause axiom*);
- a *Boolean axiom* $x^2 - x$ or *complementarity axiom* $x + \overline{x} - 1$ for any variable x;
- 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 encode each negative literal $\overline{x}$ as $(1 - x)$, 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] 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 formula F, we define the size $S_{\mathcal{PCR}}(F \vdash \bot)$, degree $Deg_{\mathcal{PCR}}(F \vdash \bot)$, and length $L_{\mathcal{PCR}}(F \vdash \bot)$ of refuting F in PCR (and analogously for PC).

We write $Vars(C)$ and $Vars(m)$ 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. 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 encodings of clauses in F and the Boolean and complementarity axioms. It can be shown that such an ideal contains 1 if and only if F is unsatisfiable.

As mentioned above, we have $Deg_{\mathcal{PCR}}(F \vdash \bot) = Deg_{\mathcal{PC}}(F \vdash \bot)$ for any CNF formula F. This claim can essentially be verified by taking any PCR refutation of F and replacing all occurrences of $\overline{y}$ by $(1 - y)$ to obtain a valid PC refutation in the same degree. Hence, we can drop the subscript from the notation for the degree measure. 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).

Theorem 2.2 ([IPS99]). *Let F be an unsatisfiable CNF formula of width $W(F)$ over n variables. Then*

$$S_{\mathcal{PCR}}(F \vdash \bot) = \exp\left(\Omega\left(\frac{(Deg(F \vdash \bot) - W(F))^2}{n}\right)\right) \quad .$$

Thus, for k-CNF formulas it is sufficient to prove strong enough lower bounds on the PC degree of refutations to establish strong lower bounds on PCR proof size.

[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.

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 [AR03]. 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 (and hence, in constrast to resolution, the size of refutations, rather than the length, is the right measure to focus on). This is not hard to show, and in some sense is probably folklore, but since it does not seem to be too widely known we state it for the record and provide a proof.

Proposition 2.3. *Any unsatisfiable k-CNF formula F has a (multilinear) polynomial calculus refutation of length linear in the size of the formula F.*

Proof. We show by induction how to derive polynomials $P_j = 1 - \prod_{i=1}^{j}(1 - C_i)$ in length linear in j, where we identify the clause C_i in $F = \bigwedge_{i=1}^{m} C_i$ with the polynomial encoding of this clause. The end result is the polynomial $P_m = 1 - \prod_{i=1}^{m}(1 - C_i)$. As 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, it follows that P_m is syntactically equal to the polynomial 1.

The base case of the induction is the polynomial P_1 that is equal to C_1. To prove the induction step, we need to show how to derive

$$P_{j+1} = 1 - \prod_{i=1}^{j+1}(1 - C_i) = 1 - (1 - C_{j+1})(1 - P_j) = P_j + C_{j+1} - C_{j+1}P_j \tag{2.1}$$

from P_j and C_{j+1} in a constant number of steps. To start, we derive $C_{j+1}P_j$ from P_j, which can be done with a constant number of multiplications and additions since the width/degree of C_{j+1} is upper-bounded by the constant 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 will also need to use restrictions. A *restriction* ρ on F is a partial assignment to the variables of F. We use $\mathrm{Dom}(\rho)$ to denote the set of variables assigned by ρ. In a restricted formula $F\restriction_\rho$ all clauses satisfied by ρ are removed and all other clauses have falsified literals removed. For a PC refutation π restricted by ρ we have that if ρ satisfies a literal in a monomial, then that monomial is set to 0 and vanishes, and all falsified literals in a monomial get replaced by 1 and disappear. It is not hard to see that if π is a PC (or PCR) refutation of F, then $\pi\restriction_\rho$ is a PC (or PCR) refutation of $F\restriction_\rho$, and this restricted refutation has at most the same size, degree, and length as the original refutation.

3 A Generalization of the Alekhnovich–Razborov Method for CNFs

Many lower bounds in proof complexity are proved by arguing in terms of expansion. One common approach is to associate a bipartite graph $G(F)$ with the CNF formula 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). The method we present below, which is an extension of the techniques developed by Alekhnovich and Razborov [AR03] (but restricted to the special case of CNF formulas), is a variation on this theme. As already discussed, however, we will need a slightly more general graph construction where clauses and variables can be grouped into clusters. We begin by describing this construction.

3.1 A Generalized Clause-Variable Incidence Graph

The key to our construction of generalized clause-variable incidence graphs is to keep track of how clauses in a CNF formula are affected by partial assignments.

Definition 3.1 (Respectful assignments and variable sets). We say that a partial assignment ρ *respects* a CNF formula E, or that ρ is *E-respectful*, if for every clause C in E either $Vars(C) \cap \mathrm{Dom}(\rho) = \emptyset$ or ρ satisfies C. A set of variables V respects a CNF formula E if there exists an assignment ρ with $\mathrm{Dom}(\rho) = V$ that respects E.

Example 3.2. Consider the CNF formula $E = (x_1 \wedge x_2) \wedge (\overline{x}_1 \wedge x_3) \wedge (x_1 \wedge x_4) \wedge (\overline{x}_1 \wedge x_5)$ and 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 and x_5 to true respects E since it satisfies the clauses containing these variables, and hence V_2 is E-respectful. However, V_1 is not E-respectful since setting x_1 will affect all clauses in E but cannot satisfy both $x_1 \wedge x_4$ and $\overline{x}_1 \wedge x_5$.

Definition 3.3 (Respectful satisfaction). Let F and E be CNF formulas and let V be a set of variables. We say that F is *E-respectfully satisfiable by V* if there exists a partial assignment ρ with $\mathrm{Dom}(\rho) = V$ that satisfies F and respects E. Such an assignment ρ is said to *E-respectfully satisfy F*.

Using a different terminology, Definition 3.1 says that ρ is an *autarky* for E, meaning that ρ satisfies all clauses in E which it touches, i.e., that $E \restriction_\rho \subseteq E$ after we remove all satisfied clauses in $E \restriction_\rho$. Definition 3.3 ensures that the autarky ρ satisfies the formula F.

Recall that we identify a CNF formula $\bigwedge_{i=1}^{m} C_i$ with the set of clauses $\{C_i \mid i \in [m]\}$. In the rest of this section we will switch freely between these two perspectives. We also change to the notation $\mathcal{F}$ for the input CNF formula, to free up other letters that will be needed in notation introduced below.

To build a bipartite graph representing the CNF formula $\mathcal{F}$, we will group the formula into subformulas (i.e., subsets of clauses). In what follows, we write $\mathcal{U}$ to denote the part of $\mathcal{F}$ that will form the left vertices of the constructed bipartite graph, while E denotes the part of $\mathcal{F}$ which will not be represented in the graph but will be used to enforce respectful satisfaction. In more detail, $\mathcal{U}$ is a family of subformulas F of $\mathcal{F}$ where each subformula is one vertex on the left-hand side of the graph. We also consider the variables of $\mathcal{F}$ to be divided into a family $\mathcal{V}$ of subsets of variables V. In our definition, $\mathcal{U}$ and $\mathcal{V}$ do not need to be partitions of clauses and variables in $\mathcal{F}$, respectively. This is not too relevant for $\mathcal{U}$ because we will always define it as a partition, but it turns out to be useful in our applications to have sets in $\mathcal{V}$ share variables. The next definition describes the bipartite graph that we build and distinguishes between two types of neighbour relations in this graph.

Definition 3.4 (Bipartite $(\mathcal{U}, \mathcal{V})_E$-graph). Let E be a CNF formula, $\mathcal{U}$ be a set of CNF formulas, and $\mathcal{V}$ be a family of sets of variables V that respect E. Then the *(bipartite) $(\mathcal{U}, \mathcal{V})_E$-graph* is a bipartite graph with left vertices $F \in \mathcal{U}$, right vertices $V \in \mathcal{V}$, and edges between F and V if $Vars(F) \cap V \neq \emptyset$. For every edge (F, V) in the graph we say that F and V are *E-respectful neighbours* if F is E-respectfully satisfiable by V. Otherwise, they are *E-disrespectful neighbours*.

We will often write $(\mathcal{U}, \mathcal{V})_E$ as a shorthand for the graph defined by $\mathcal{U}$, $\mathcal{V}$, and E as above. We will also use standard graph notation and write $N(F)$ to denote the set of all neighbours $V \in \mathcal{V}$ of a vertex/CNF formula $F \in \mathcal{U}$. It is important to note that the fact that F and V are E-respectful neighbours can be witnessed by an assignment that falsfies other subformulas $F' \in \mathcal{U} \setminus \{F\}$.

We can view the formation of the $(\mathcal{U}, \mathcal{V})_E$-graph as taking the clause-variable incidence graph $G(\mathcal{F})$ of the CNF formula $\mathcal{F}$, throwing out a part of $\mathcal{F}$, which we denote E, and clustering the remaining clauses and variables into $\mathcal{U}$ and $\mathcal{V}$. The edge relation in the $(\mathcal{U}, \mathcal{V})_E$-graph follows naturally from this view, as we put an edge between two clusters if there is an edge between any two elements of these clusters. The only additional information we need to keep track of is which clause and variable clusters are E-respectful neighbours or not.

Definition 3.5 (Respectful boundary). For a $(\mathcal{U},\mathcal{V})_E$-graph and a subset $\mathcal{U}' \subseteq \mathcal{U}$, the *$E$-respectful boundary* $\partial_E(\mathcal{U}')$ of $\mathcal{U}'$ is the family of variable sets $V \in \mathcal{V}$ such that each $V \in \partial_E(\mathcal{U}')$ is an E-respectful neighbour of some clause set $F \in \mathcal{U}'$ but is not a neighbour (respectful or disrespectful) of any other clause set $F' \in \mathcal{U}' \setminus \{F\}$.

It will sometimes be convenient to interpret subsets $\mathcal{U}' \subseteq \mathcal{U}$ as CNF formulas $\bigwedge_{F\in\mathcal{U}'}\bigwedge_{C\in F} C$, and we will switch back and forth between these two interpretations as seems most suitable. We will show that a formula $\mathcal{F} = \bigwedge_{F\in\mathcal{U}}\bigwedge_{C\in F} C \wedge E = \mathcal{U} \wedge E$ is hard for polynomial calculus with respect to degree if the $(\mathcal{U},\mathcal{V})_E$-graph has a certain expansion property as defined next.

Definition 3.6 (Respectful boundary expander). A $(\mathcal{U},\mathcal{V})_E$-graph is said to be an (s,δ,ξ,E)-*respectful boundary expander*, or just an (s,δ,ξ,E)-*expander* for brevity, if for every set $\mathcal{U}' \subseteq \mathcal{U}$, $|\mathcal{U}'| \leq s$, it holds that $|\partial_E(\mathcal{U}')| \geq \delta|\mathcal{U}'| - \xi$.

Note that an (s,δ,ξ,E)-respectful boundary expander is a standard bipartite boundary expander except for two modifications:

- We measure expansion not in terms of the whole boundary but only in terms of the *respectful boundary*[4] as described in Definition 3.5.
- Also, the size of the boundary $|\partial_E(\mathcal{U}')|$ on the right does not quite have to scale linearly with the size of the vertex set $|\mathcal{U}'|$ on the left. Instead, we allow an *additive loss* ξ in the expansion. In our applications, we can usually construct graphs with good enough expansion so that we can choose $\xi = 0$, but for one of the results we present it will be helpful to allow a small slack here.

Before we state our main theorem we need one more technical definition, which is used to ensure that there do not exist variables that appear 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 3.7. The *overlap* of a variable x with respect to a family of variable sets $\mathcal{V}$ is $ol(x,\mathcal{V}) = |\{V \in \mathcal{V} : x \in V\}|$ and the overlap of $\mathcal{V}$ is $ol(\mathcal{V}) = \max_x\{ol(x,\mathcal{V})\}$, i.e., the maximum number of sets $V \in \mathcal{V}$ containing any particular variable x.

Given the above definitions, we can state the main technical result in this paper as follows.

Theorem 3.8. *Let $\mathcal{F} = \bigwedge_{F\in\mathcal{U}}\bigwedge_{C\in F} C \wedge E = \mathcal{U} \wedge E$ be a CNF formula for which $(\mathcal{U},\mathcal{V})_E$ is an (s,δ,ξ,E)-expander with overlap $ol(\mathcal{V}) = d$, and suppose furthermore that for all $\mathcal{U}' \subseteq \mathcal{U}$, $|\mathcal{U}'| \leq s$, it holds that $\mathcal{U}' \wedge E$ is satisfiable. Then any polynomial calculus refutation of $\mathcal{F}$ requires degree strictly greater than $(\delta s - 2\xi)/(2d)$.*

In order to prove this theorem, it will be convenient to review some algebra. We do so next.

3.2 Some Algebra Basics

We will need to compute with polynomials modulo ideals, and in order to do so we need to have an ordering of monomials (which, as we recall, will always be multilinear).

Definition 3.9 (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:

[4] Somewhat intriguingly, we will not see any disrespectful neighbours in our applications in Section 4, but the concept of respectfulness is of crucial importance for the main technical result in Theorem 3.8 to go through. One way of seeing this is to construct a $(\mathcal{U},\mathcal{V})_E$-graph for an expanding set of linear equations $\bmod 2$, where $\mathcal{U}$ consists of the (CNF encodings of) the equations, $\mathcal{V}$ consists of one variable set for each equation containing exactly the variables in this equation, and E is empty. Then this $(\mathcal{U},\mathcal{V})_E$-graph has the same boundary expansion as the constraint-variable incidence graph, but Theorem 3.8 does not apply (which it should not do) since this expansion is not respectful.

- 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 $mm_1 \prec mm_2$.

Two terms $t_1 = \alpha_1 m_1$ and $t_2 = \alpha_2 m_2$ 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. For the rest of this section we only need that $\prec$ is some fixed but arbitrary admissible ordering, but the reader can think of the degree-lexicographical ordering without any particular loss of generality. We write $m_1 \preccurlyeq m_2$ to denote that $m_1 \prec m_2$ or $m_1 = m_2$.

Definition 3.10 (Leading, reducible, and irreducible terms). For a polynomial $P = \sum_i t_i$, the *leading term* $LT(P)$ of P 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.

The following fact is not hard to verify.

Fact 3.11. *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 $Q + R$, where $Q \in I$ and R is a linear combination of irreducible terms modulo I.*

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

Definition 3.12 (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 3.13. *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

$$P' = Q' + R' \tag{3.1}$$

for $Q' \in I_1$ and R' a linear combination of irreducible terms over I_1. Let

$$P \cdot R_{I_1}(P') = PR' = Q + R \tag{3.2}$$

for $Q \in I_2$ and R a linear combination of irreducible terms over I_2. Then

$$PP' = PQ' + PR' = PQ' + Q + R \tag{3.3}$$

where $PQ' + Q \in I_2$. By the uniqueness in Fact 3.11, we conclude that the equality $R_{I_2}(PP') = R = R_{I_2}(P \cdot R_{I_1}(P'))$ holds. □

Observation 3.14. *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 m_ρ be the product of all variables in t assigned by ρ and let $\alpha = m_\rho\restriction_\rho$, 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 $LT(\alpha^{-1} m_\rho Q) = t$, contradicting that t is irreducible. □

3.3 Proof Strategy

Let us now state the lemma on which we base the proof of Theorem 3.8.

Lemma 3.15 ([Raz98]). *Let $\mathcal{F}$ be any CNF formula and $D \in \mathbb{N}^+$ be a positive integer. Suppose that there exists a linear operator R on multilinear polynomials over $Vars(\mathcal{F})$ with the following properties:*

1. *$R(1) \neq 0$.*
2. *$R(C) = 0$ for (the translations to polynomials of) all axioms $C \in \mathcal{F}$.*
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 $\mathcal{F}$ (and hence any PCR refutation of $\mathcal{F}$) requires degree strictly greater than D.

The proof of Lemma 3.15 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 3.8, we construct a linear operator $R_{\mathcal{G}}$ that satisfies the conditions of Lemma 3.15 when the $(\mathcal{U}, \mathcal{V})_E$-graph $\mathcal{G}$ is an expander. First, let us describe how we make the connection between polynomials and the given $(\mathcal{U}, \mathcal{V})_E$-graph. We remark that in the rest of this section we will identify a clause C with its polynomial translation and will refer to C as a (polynomial) axiom.

Definition 3.16 (Term and polynomial neighbourhood). The *neighbourhood* $N(t)$ of a term t with respect to $(\mathcal{U}, \mathcal{V})_E$ 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.

To every polynomial we can now assign a family of variable sets $\mathcal{V}'$. But we are interested in the axioms that are needed in order to produce that 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 derivation that yielded polynomials P with $Vars(P) \subseteq \bigcup_{V \in \mathcal{V}'} V$. This is the intuition behind the next definition.[5]

Definition 3.17 (Polynomial support). For a given $(\mathcal{U}, \mathcal{V})_E$-graph and a family of variable sets $\mathcal{V}' \subseteq \mathcal{V}$, we say that a subset $\mathcal{U}' \subseteq \mathcal{U}$ is $(s, \mathcal{V}')$-*contained* if $|\mathcal{U}'| \leq s$ and $\partial_E(\mathcal{U}') \subseteq \mathcal{V}'$.

We define the *polynomial s-support $Sup_s(\mathcal{V}')$ of $\mathcal{V}'$ with respect to* $(\mathcal{U}, \mathcal{V})_E$, or just *$s$-support of $\mathcal{V}'$* for brevity, to be the union of all $(s, \mathcal{V}')$-contained subsets $\mathcal{U}' \subseteq \mathcal{U}$, and 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 $(\mathcal{U}, \mathcal{V})_E$-graph $\mathcal{G}$ will be clear from context. The next observation follows immediately from Definition 3.17.

Observation 3.18. *Support is monotone in the sense that if $t \subseteq t'$ are two terms, 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 3.12. We will show that under the assumptions in Theorem 3.8 it holds that this operator satisfies the conditions in Lemma 3.15. Let us first introduce some notation for the set of all polynomials that can be generated from some axioms $\mathcal{U}' \subseteq \mathcal{U}$.

[5] We remark that Definition 3.17 is a slight modification of the original definition of support in [AR03] that was proposed by Yuval Filmus [Fil14].

Definition 3.19. For a $(\mathcal{U},\mathcal{V})_E$-graph and $\mathcal{U}' \subseteq \mathcal{U}$, we write $\mathcal{I}_E(\mathcal{U}')$ to denote the ideal generated by the polynomial axioms in $\mathcal{U}' \wedge E$.[6]

Definition 3.20 ($(\mathcal{U},\mathcal{V})_E$-graph reduction). For a $(\mathcal{U},\mathcal{V})_E$-graph $\mathcal{G}$, the *$(\mathcal{U},\mathcal{V})_E$-graph reduction* $R_\mathcal{G}$ on a term t is defined as $R_\mathcal{G}(t) = R_{\mathcal{I}_E(Sup_s(t))}(t)$. 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 3.20, 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 $(\mathcal{U},\mathcal{V})_E$-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 3.25 below. To establish this lemma, we need to develop a better understanding of polynomial support.

3.4 Some Properties of Polynomial Support

A crucial technical property that we will need is that if a $(\mathcal{U},\mathcal{V})_E$-graph is a good expander in the sense of Definition 3.6, then for small enough sets $\mathcal{V}'$ all $(s,\mathcal{V}')$-contained subsets $\mathcal{U}' \subseteq \mathcal{U}$ as per Definition 3.17 are of at most half of the allowed size.

Lemma 3.21. *Let $(\mathcal{U},\mathcal{V})_E$ be an (s,δ,ξ,E)-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 $(\mathcal{U},\mathcal{V})_E$-graph to derive the inequality $|\partial_E(\mathcal{U}')| \geq \delta|\mathcal{U}'| - \xi$. In the other direction, we can obtain an upper bound on the size of $\partial_E(\mathcal{U}')$ by noting that for any $(s,\mathcal{V}')$-contained set $\mathcal{U}'$ it holds that $|\partial_E(\mathcal{U}')| \leq |\mathcal{V}'|$. 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 3.21 now allows us to conclude that for a small enough subset $\mathcal{V}'$ on the right-hand side of $(\mathcal{U},\mathcal{V})_E$ it holds that in fact the whole polynomial s-support $Sup_s(\mathcal{V}')$ of $\mathcal{V}'$ on the left-hand side is $(s/2,\mathcal{V}')$-contained.

Lemma 3.22. *Let $(\mathcal{U},\mathcal{V})_E$ be an (s,δ,ξ,E)-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 $(\mathcal{U},\mathcal{V})_E$ 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 $\mathcal{U}_1 \cup \mathcal{U}_2$ is also $(s,\mathcal{V}')$-contained. First, by Lemma 3.21 we have $|\mathcal{U}_1|, |\mathcal{U}_2| \leq s/2$ and hence $|\mathcal{U}_1 \cup \mathcal{U}_2| \leq s$. Second, it holds that $\partial_E(\mathcal{U}_1), \partial_E(\mathcal{U}_2) \subseteq \mathcal{V}'$, which implies that $\partial_E(\mathcal{U}_1 \cup \mathcal{U}_2) \subseteq \mathcal{V}'$, 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 3.21 shows that this set is $(s/2,\mathcal{V}')$-contained. This completes the proof. □

What the next lemma says is, roughly, that if we reduce a term t modulo an ideal generated by a not too large set of polynomials containing some polynomials outside of the support of t, then we can remove all such polynomials from the generators of the ideal without changing the irreducible component of t.

[6] That is, $\mathcal{I}_E(\mathcal{U}')$ is the smallest set I of multilinear polynomials that contains all axioms in $\mathcal{U}' \wedge E$ and that is closed under addition of $P_1, P_2 \in I$ and by multiplication of $P \in I$ by any multilinear polynomial over $Vars(\mathcal{U} \wedge E)$ (where as before the resulting product is implicitly multilinearized).

Lemma 3.23. *Let $\mathcal{G}$ be a $(\mathcal{U},\mathcal{V})_E$-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 for any term t' with $N(t') \subseteq N(Sup_s(t)) \cup N(t)$ it holds that if t' is reducible modulo $\mathcal{I}_E(\mathcal{U}')$, it is also reducible modulo $\mathcal{I}_E(Sup_s(t))$.*

Proof. If $\mathcal{U}'$ is $(s, N(t))$-contained, then by Definition 3.17 it holds that $\mathcal{U}' \subseteq Sup_s(t)$ and there is nothing to prove. Hence, assume $\mathcal{U}'$ is not $(s, N(t))$-contained. We claim that this implies that we can find a subformula $F \in \mathcal{U}' \setminus Sup_s(t)$ with a neighbouring subset of variables $V \in \big(\partial_E(\mathcal{U}') \cap N(F)\big) \setminus N(t')$ in the respectful boundary of $\mathcal{U}'$ but not in the neighbourhood of t'. To argue this, note that since $|\mathcal{U}'| \leq s$ it follows from Definition 3.17 that the reason $\mathcal{U}'$ is not $(s, N(t))$-contained is that there exist some $F \in \mathcal{U}'$ and some set of variables $V \in N(F)$ such that $V \in \partial_E(\mathcal{U}') \setminus N(t)$. Moreover, the assumption $\mathcal{U}' \supseteq Sup_s(t)$ implies that such an F cannot be in $Sup_s(t)$. Otherwise there would exist an $(s, N(t))$-contained set $\mathcal{U}^*$ such that $F \in \mathcal{U}^* \subseteq Sup_s(t) \subseteq \mathcal{U}'$, from which it would follow that $V \in \partial_E(\mathcal{U}') \cap N(\mathcal{U}^*) \subseteq \partial_E(\mathcal{U}^*) \subseteq N(t)$, contradicting $V \notin N(t)$. We have shown that $F \notin Sup_s(t) \subseteq \mathcal{U}'$ and $V \in \partial_E(\mathcal{U}') \cap N(F)$, and by combining these two facts we can also deduce that $V \notin N(Sup_s(t))$, since otherwise V could not be contained in the boundary of $\mathcal{U}'$. In particular, this means that $V \notin N(t') \subseteq N(Sup_s(t)) \cup N(t)$, which establishes the claim made above.

Fixing F and V such that $F \in \mathcal{U}' \setminus Sup_s(t)$ and $V \in \big(\partial_E(\mathcal{U}') \cap N(F)\big) \setminus N(t')$, our second claim is that if F is removed from the generators of the ideal, it still holds that if t' is reducible modulo $\mathcal{I}_E(\mathcal{U}')$, then this term is also reducible modulo $\mathcal{I}_E(\mathcal{U}' \setminus \{F\})$. Given this second claim we are done, since we can then argue by induction over the elements in $\mathcal{U}' \setminus Sup_s(t)$ and remove them one by one to arrive at the conclusion that every term t' with $N(t') \subseteq N(Sup_s(t)) \cup N(t)$ that is reducible modulo $\mathcal{I}_E(\mathcal{U}')$ is also reducible modulo $\mathcal{I}_E(Sup_s(t))$, which is precisely what the lemma says.

We proceed to establish this second claim. The assumption that t' is reducible modulo $\mathcal{I}_E(\mathcal{U}')$ means that there exists a polynomial $P \in \mathcal{I}_E(\mathcal{U}')$ such that $t' = LT(P)$. Since P is in the ideal $\mathcal{I}_E(\mathcal{U}')$ it can be written as a polynomial combination $P = \sum_i P_i C_i$ of axioms $C_i \in \mathcal{U}' \wedge E$ for some polynomials P_i. If we could hit P with a restriction that satisfies (and hence removes) F while leaving t' and $(\mathcal{U}' \setminus \{F\}) \wedge E$ untouched, this would show that t' is the leading term of some polynomial combination of axioms in $(\mathcal{U}' \setminus \{F\}) \wedge E$. This is almost what we are going to do.

As our restriction ρ we choose any assignment with domain $\mathrm{Dom}(\rho) = V$ that E-respectfully satisfies F. Note that at least one such assignment exists since $V \in \partial_E(\mathcal{U}') \cap N(F)$ is an E-respectful neighbour of F by Definition 3.5. By the choice of ρ it holds that F is satisfied, i.e., that all axioms in F are set to 0. Furthermore, none of the axioms in $\mathcal{U}' \setminus \{F\}$ are affected by ρ since V is in the boundary of $\mathcal{U}'$.[7] As for axioms in E it is not necessarily true that ρ will leave all of them untouched, but by assumption ρ respects E and so any axiom in E is either satisfied (and zeroed out) by ρ or is left intact. It follows that $P\restriction_\rho$ can be be written as a polynomial combination $P\restriction_\rho = \sum_i \big(P_i\restriction_\rho\big) C_i$, where $C_i \in (\mathcal{U}' \setminus \{F\}) \wedge E$, and hence $P\restriction_\rho \in \mathcal{I}_E(\mathcal{U}' \setminus \{F\})$.

To see that t' is preserved as the leading term of $P\restriction_\rho$, note that ρ does not assign any variables in t' since $V \notin N(t')$. Hence, $t' = LT(P\restriction_\rho)$, as ρ can only make the other terms smaller with respect to $\prec$. This shows that there is a polynomial $P' = P\restriction_\rho \in \mathcal{I}_E(\mathcal{U}' \setminus \{F\})$ with $LT(P') = t'$, and hence t' is reducible modulo $\mathcal{I}_E(\mathcal{U}' \setminus \{F\})$. The lemma follows. □

We need to deal with one more detail before we can prove the key technical lemma that it is possible to reduce modulo suitably chosen larger ideals without changing the reduction operator, namely (again roughly speaking) that reducing a term modulo an ideal does not introduce any new variables outside of the generators of that ideal.

Lemma 3.24. *Suppose that $\mathcal{U}^* \subseteq \mathcal{U}$ for some $(\mathcal{U},\mathcal{V})_E$-graph and let t be any term. Then it holds that $N\big(R_{\mathcal{I}_E(\mathcal{U}^*)}(t)\big) \subseteq N(\mathcal{U}^*) \cup N(t)$.*

Proof. Let $P = R_{\mathcal{I}_E(\mathcal{U}^*)}(t)$ be the polynomial obtained when reducing t modulo $\mathcal{I}_E(\mathcal{U}^*)$ and let $V \in \mathcal{V}$ be any set such that $V \notin N(\mathcal{U}^*) \cup N(t)$. We show that $V \notin N(P)$.

[7] Recalling the remark after Definition 3.4, we note that we can ignore here if ρ happens to falsify axioms in $\mathcal{U} \setminus \mathcal{U}'$.

By the definition of $(\mathcal{U},\mathcal{V})_E$-graphs there exists an assignment ρ to all of the variables in V that respects E. Write $t = Q + P$ with $Q \in \mathcal{I}_E(\mathcal{U}^*)$ and P a linear combination of irreducible monomials as in Fact 3.11 and apply the restriction ρ to this equality. Note that $t\restriction_\rho = t$ as V is not a neighbour of t. Moreover, $Q\restriction_\rho$ is in the ideal $\mathcal{I}_E(\mathcal{U}^*)$ because ρ does not set any variables in $\mathcal{U}^*$ and every axiom in E sharing variables with V is set to 0 by ρ. Thus, t can be written as $t = Q' + P\restriction_\rho$, with $Q' \in \mathcal{I}_E(\mathcal{U}^*)$. As all terms in P are irreducible modulo $\mathcal{I}_E(\mathcal{U}^*)$, they remain irreducible after restricting P by ρ by Observation 3.14. Hence, it follows that $P\restriction_\rho = P$ by the uniqueness in Fact 3.11 and P cannot contain any variable from V. This in turn implies that every set $V \in N(P)$ is contained in $N(\mathcal{U}^*) \cup N(t)$. □

Now we can state the formal claim that enlarging the ideal does not change the reduction operator if the enlargement is done in the right way.

Lemma 3.25. *Let $\mathcal{G}$ be a $(\mathcal{U},\mathcal{V})_E$-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 $R_{\mathcal{I}_E(\mathcal{U}')}(t) = R_{\mathcal{I}_E(Sup_s(t))}(t)$.*

Proof. We prove that $R_{\mathcal{I}_E(\mathcal{U}')}(t) = R_{\mathcal{I}_E(Sup_s(t))}(t)$ by applying the contrapositive of Lemma 3.23. Recall that this lemma states that any term t' with $N(t') \subseteq N(Sup_s(t)) \cup N(t)$ that is reducible modulo $\mathcal{I}_E(\mathcal{U}')$ is also reducible modulo $\mathcal{I}_E(Sup_s(t))$. Since every term t' in $R_{\mathcal{I}_E(Sup_s(t))}(t)$ is irreducible modulo $\mathcal{I}_E(Sup_s(t))$ and since by applying Lemma 3.24 with $\mathcal{U}^* = Sup_s(t)$ we have that $N(t') \subseteq N(Sup_s(t)) \cup N(t)$, it follows that t' is also irreducible modulo $\mathcal{I}_E(\mathcal{U}')$. This shows that $R_{\mathcal{I}_E(\mathcal{U}')}(t) = R_{\mathcal{I}_E(Sup_s(t))}(t)$ as claimed, and the lemma follows. □

3.5 Putting the Pieces in the Proof Together

Now we have just a couple of lemmas left before we can prove Theorem 3.8, which as discussed above will be established by appealing to Lemma 3.15.

Lemma 3.26. *Let $(\mathcal{U},\mathcal{V})_E$ be an (s,δ,ξ,E)-expander with overlap $ol(\mathcal{V}) = d$. Then for any term t with $Deg(t) \leq (\delta s - 2\xi)/(2d)$ it holds that $|Sup_s(t)| \leq s/2$.*

Proof. Because of the bound on the overlap $ol(\mathcal{V})$ we have that the size of $N(t)$ is bounded by $\delta s/2 - \xi$. An application of Lemma 3.22 now yields the desired bound $|Sup_s(t)| \leq s/2$. □

Lemma 3.27. *Let $(\mathcal{U},\mathcal{V})_E$ be an (s,δ,ξ,E)-expander with overlap $ol(\mathcal{V}) = d$. Then for any term t with $Deg(t) < \lfloor(\delta s - 2\xi)/(2d)\rfloor$, any term t' occurring in $R_{\mathcal{I}_E(Sup_s(t))}(t)$, and any variable x, it holds that $R_{\mathcal{I}_E(Sup_s(xt'))}(xt') = R_{\mathcal{I}_E(Sup_s(xt))}(xt')$.*

Proof. We prove the lemma by showing that $Sup_s(xt') \subseteq Sup_s(xt)$ and that $|Sup_s(xt)| \leq s$, which then allows us to apply Lemma 3.25. To prove that $Sup_s(xt')$ is a subset of $Sup_s(xt)$, we will show that $Sup_s(xt') \cup Sup_s(xt)$ is $(s, N(xt))$-contained in the sense of Definition 3.17. From this it follows that $Sup_s(xt') \subseteq Sup_s(xt') \cup Sup_s(xt) = Sup_s(xt)$.

Towards this goal, as $Deg(t') \leq Deg(t)$ we first observe that we can apply Lemma 3.26 to deduce that $|Sup_s(xt')| \leq s/2$ and $|Sup_s(xt)| \leq s/2$, and hence $|Sup_s(xt') \cup Sup_s(xt)| \leq s$, which satisfies the size condition for containment. It remains to show that $\partial_E\big(Sup_s(xt') \cup Sup_s(xt)\big) \subseteq N(xt)$. From Lemma 3.24 we have that $N(t') \subseteq N(Sup_s(t)) \cup N(t)$. As $N(xt') = N(x) \cup N(t')$ and $Sup_s(t) \subseteq Sup_s(xt)$ by the monotonicity in Observation 3.18, it follows that

$$N(xt') = N(x) \cup N(t') \subseteq N(x) \cup N(Sup_s(t)) \cup N(t) \subseteq N(Sup_s(xt)) \cup N(xt) \ . \tag{3.4}$$

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

$$\begin{aligned}
&\partial_E\left(Sup_s(xt') \cup Sup_s(xt)\right) = \\
&\qquad = \left(\partial_E\left(Sup_s(xt')\right) \setminus N\left(Sup_s(xt)\right)\right) \cup \left(\partial_E\left(Sup_s(xt)\right) \setminus N\left(Sup_s(xt')\right)\right) \\
&\qquad \subseteq \left(N\left(xt'\right) \setminus N\left(Sup_s(xt)\right)\right) \cup \left(N\left(xt\right) \setminus N\left(Sup_s(xt')\right)\right) \\
&\qquad \subseteq N\left(xt\right) \ ,
\end{aligned} \tag{3.5}$$

where the first line follows from the boundary definition in Definition 3.5, the second line follows by the property of s-support that $\partial_E(Sup_s(xt)) \subseteq N(xt)$, and the last line follows from (3.4). Hence, $Sup_s(xt') \cup Sup_s(xt)$ is $(s, N(xt))$-contained.

As discussed above, we can now apply Lemma 3.25 to reach the desired conclusion that the equality $R_{\mathcal{I}_E(Sup_s(xt'))}(xt') = R_{\mathcal{I}_E(Sup_s(xt))}(xt')$ holds. □

Now we can prove our main technical theorem.

Proof of Theorem 3.8. Recall that the assumptions of the theorem are that we have a $(\mathcal{U}, \mathcal{V})_E$-graph for a CNF formula $\mathcal{F} = \bigwedge_{F \in \mathcal{U}} F \wedge E$ such that $(\mathcal{U}, \mathcal{V})_E$ is an (s, δ, ξ, E)-expander with overlap $ol(\mathcal{V}) = d$ and that furthermore for all $\mathcal{U}' \subseteq \mathcal{U}$, $|\mathcal{U}'| \leq s$, it holds that $\bigwedge_{F \in \mathcal{U}'} F \wedge E$ is satisfiable. We want to prove that no polynomial calculus derivation from $\bigwedge_{F \in \mathcal{U}} F \wedge E = \mathcal{U} \wedge E$ of degree at most $(\delta s - 2\xi)/(2d)$ can reach contradiction.

First, if removing all axiom clauses from $\mathcal{U} \wedge E$ with degree strictly greater than $(\delta s - 2\xi)/(2d)$ produces a satisfiable formula, then the lower bound trivially holds. Otherwise, we can remove these large-degree axioms and still be left with a $(\mathcal{U}, \mathcal{V})_E$-graph that satisfies the conditions above. In order to see this, let us analyze what happens to the $(\mathcal{U}, \mathcal{V})_E$-graph if an axiom is removed from the formula.

Removing axioms from E only relaxes the conditions on respectful satisfiability while keeping all edges in the graph, so the conditions of the theorem still hold. In removing axioms from $\mathcal{U}$ we have two cases: either we remove all axioms from some subformula $F \in \mathcal{U}$ or we remove only a part of this subformula. In the former case, it is clear that we can remove the vertex F from the structure without affecting any of the conditions. In the latter case, we claim that any set $V \in \mathcal{V}$ that is an E-respectful neighbour of F remains an E-respectful neighbour of the formula F' in which large degree axioms have been removed. Clearly, the same assignments to V that satisfy F also satisfy $F' \subseteq F$. Also, V must still be a neighbour of F', for otherwise F' would not share any variables with V, which would imply that no assignment to V could satisfy F' and hence F. This would contradict the assumption that V is an E-respectful neighbour of F. Hence, we conclude that removal of large-degree axioms can only improve the E-respectful boundary expansion of the $(\mathcal{U}, \mathcal{V})_E$-graph.

Thus, let us focus on a $(\mathcal{U}, \mathcal{V})_E$-graph $\mathcal{G}$ that has all axioms of degree at most $(\delta s - 2\xi)/(2d)$. We want to show that the operator $R_{\mathcal{G}}$ from Definition 3.20 satisfies the conditions of Lemma 3.15, from which Theorem 3.8 immediately follows. We can note right away that the operator $R_{\mathcal{G}}$ is linear by construction.

To prove that $R_{\mathcal{G}}(1) = R_{\mathcal{I}_E(Sup_s(1))}(1) \neq 0$, we start by observing that the size of the s-support of 1 is upper-bounded by $s/2$ according to Lemma 3.26. Using the assumption that for every subset $\mathcal{U}'$ of $\mathcal{U}$, $|\mathcal{U}'| \leq s$, the formula $\mathcal{U}' \wedge E$ is satisfiable, it follows that 1 is not in the ideal $\mathcal{I}_E(Sup_s(1))$ and hence $R_{\mathcal{I}_E(Sup_s(1))}(1) \neq 0$.

We next show that $R_{\mathcal{G}}(C) = 0$ for any axiom clause $C \in \mathcal{U} \wedge E$ (where we recall that we identify a clause C with its translation into a linear combination of monomials). By the preprocessing step above it holds that the degree of C is bounded by $(\delta s - 2\xi)/(2d)$, from which it follows by Lemma 3.26 that the size of the s-support of every term in C is bounded by $s/2$. Since C is the polynomial encoding of a clause, the leading term $LT(C)$ contains all the variables appearing in C.[8] Hence, the s-support $Sup_s(LT(C))$ of the leading term contains the s-support of every other term in C by Observation 3.18 and we can use Lemma 3.25 to conclude that $R_{\mathcal{G}}(C) = R_{\mathcal{I}_E(Sup_s(LT(C)))}(C)$. If $C \in E$, this means we are done because $\mathcal{I}_E(Sup_s(LT(C)))$ contains all of E, implying that $R_{\mathcal{G}}(C) = 0$.

For $C \in \mathcal{U}$ we cannot immediately argue that C reduces to 0, since (in contrast to [AR03]) it is not immediately clear that $Sup_s(LT(C))$ contains C. The problem here is that we might worry that C is part of some subformula $F \in \mathcal{U}$ for which the boundary $\partial_E(F)$ is not contained in $N(LT(C)) = Vars(C)$, and hence there is no obvious reason why C should be a member of any $(s, N(LT(C)))$-contained subset of $\mathcal{U}$. However, in view of Lemma 3.25 (applied, strictly speaking, once for every term in C) we can choose some $F \in \mathcal{U}$ such that $C \in F$ and add it to the s-support $Sup_s(LT(C))$ to obtain a set

[8] We remark that this is the only place in the proof where we are using that C is (the encoding of) a clause.

$\mathcal{U}' = Sup_s(LT(C)) \cup \{F\}$ of size $|\mathcal{U}'| \leq s/2 + 1 \leq s$ such that $R_{\mathcal{I}_E(Sup_s(LT(C)))}(C) = R_{\mathcal{I}_E(\mathcal{U}')}(C)$. Since $\mathcal{I}_E(\mathcal{U}')$ contains C as a generator we conclude that $R_{\mathcal{G}}(C) = R_{\mathcal{I}_E(\mathcal{U}')}(C) = 0$ also for $C \in \mathcal{U}$.[9]

It remains to prove the last property in Lemma 3.15 stating that $R_{\mathcal{G}}(xt) = R_{\mathcal{G}}(xR_{\mathcal{G}}(t))$ for any term t such that $Deg(t) < \lfloor(\delta s - 2\xi)/(2d)\rfloor$. We can see that this holds by studying the following sequence of equalities:

$$
\begin{aligned}
R_{\mathcal{G}}(xR_{\mathcal{G}}(t)) &= \sum_{t' \in R_{\mathcal{G}}(t)} R_{\mathcal{G}}(xt') && \text{[by linearity]} \\
&= \sum_{t' \in R_{\mathcal{G}}(t)} R_{\mathcal{I}_E(Sup_s(xt'))}(xt') && \text{[by definition of } R_{\mathcal{G}}\text{]} \\
&= \sum_{t' \in R_{\mathcal{G}}(t)} R_{\mathcal{I}_E(Sup_s(xt))}(xt') && \text{[by Lemma 3.27]} \\
&= R_{\mathcal{I}_E(Sup_s(xt))}(xR_{\mathcal{G}}(t)) && \text{[by linearity]} \\
&= R_{\mathcal{I}_E(Sup_s(xt))}(xR_{\mathcal{I}_E(Sup_s(t))}(t)) && \text{[by definition of } R_{\mathcal{G}}\text{]} \\
&= R_{\mathcal{I}_E(Sup_s(xt))}(xt) && \text{[by Observation 3.13]} \\
&= R_{\mathcal{G}}(xt) && \text{[by definition of } R_{\mathcal{G}}\text{]}
\end{aligned}
$$

Thus, $R_{\mathcal{G}}$ satisfies all the properties of Lemma 3.15, from which the theorem follows. □

Let us next show that if the slack ξ in Theorem 3.8 is zero, then the condition that $\mathcal{U}' \wedge E$ is satisfiable for sufficiently small $\mathcal{U}'$ is already implied by the expansion.

Lemma 3.28. *If a $(\mathcal{U}, \mathcal{V})_E$-graph is an $(s, \delta, 0, E)$-expander and $Vars(\mathcal{U} \wedge E) = \bigcup_{V \in \mathcal{V}} V$, then for any $\mathcal{U}' \subseteq \mathcal{U}$, $|\mathcal{U}'| \leq s$, the formula $\mathcal{U}' \wedge E$ 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 E-respectfully satisfies $\mathcal{U}'$. We do this by induction on the number of formulas in $\mathcal{U}'$. As the $(\mathcal{U}, \mathcal{V})_E$-graph is an $(s, \delta, 0, E)$-expander it follows that $|\partial_E(\mathcal{U}')| \geq \delta|\mathcal{U}'| > 0$ for any non-empty subset $\mathcal{U}'$ and hence there exists a formula $F \in \mathcal{U}'$ and a variable set V' such that V' is an E-respectful neighbour of F and is not a neighbour of any formula in $\mathcal{U}' \setminus \{F\}$. Therefore, there is an assignment ρ to the variables in V' that E-respectfully satisfies F. By the induction hypothesis there also exists an assignment ρ' that E-respectfully satisfies $\mathcal{U}' \setminus \{F\}$ and does not assign any variables in V' as $V' \notin N(\mathcal{U}' \setminus \{F\})$. Hence, by extending the assignment ρ' to the variables in V' according to the assignment ρ, we create an assignment to the union of variables in some subset of $N(\mathcal{U}')$ that E-respectfully satisfies $\mathcal{U}'$.

We now need to show how to extend this to an assignment satisfying also E. To this end, let $\rho_{\mathcal{U}'}$ be an assignment that E-respectfully satisfies $\mathcal{U}'$ and assigns the variables in $\bigcup_{V \in \mathcal{V}'} V$ for some $\mathcal{V}' \subseteq N(\mathcal{U}')$. By another induction 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 E-respectfully satisfies $\mathcal{U}'$ for every $\mathcal{V}' \subseteq \mathcal{V}'' \subseteq \mathcal{V}$. 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 E-respectfully satisfies $\mathcal{U}'$. As V' respects E, there is an assignment $\rho_{V'}$ to the variables V' that satisfies all affected clauses in E. 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. Consider therefore the subassignment $\rho^*_{V'}$ of $\rho_{V'}$ that assigns only the variables in $V' \setminus \left(\bigcup_{V \in \mathcal{V}''} V\right)$. We claim that extending $\rho_{\mathcal{V}''}$ by $\rho^*_{V'}$ creates an assignment that respects E. This is because every clause in E that has a variable in V' and was not already satisfied by $\rho_{\mathcal{V}''}$ cannot have variables in $V' \cap \left(\bigcup_{V \in \mathcal{V}''} V\right)$ (if so, $\rho_{\mathcal{V}''}$ would have been E-disrespectful) and hence every such clause must be satisfied by the subassignment $\rho^*_{V'}$.

[9] Actually, a slighly more careful argument reveals that C is always contained in $Sup_s(LT(C))$. This is so since for any $F \in \mathcal{U}$ with $C \in F$ it holds that any neighbours in $N(F) \setminus N(LT(C))$ have to be disrespectful, and so such an F always makes it into the support. However, the reasoning gets a bit more involved, and since we already needed to use Lemma 3.25 anyway we might as well apply it once more here.

Thus, we can find an assignment to all the variables $\cup_{V\in\mathcal{V}}V$ that E-respectfully satisfies $\mathcal{U}'$. As $\mathcal{V}$ includes all the variables in E it means that E is also fully satisfied. Hence, $\mathcal{U}' \wedge E$ is satisfiable and the lemma follows. □

This allows us to conclude this section by stating the following version of Theorem 3.8 for the most commonly occuring case with standard expansion without any slack.

Corollary 3.29. *Suppose that $(\mathcal{U},\mathcal{V})_E$ is an $(s,\delta,0,E)$-expander with overlap $ol(\mathcal{V}) = d$ such that $Vars(\mathcal{U}\wedge E) = \bigcup_{V\in\mathcal{V}} V$. Then any polynomial calculus refutation of the formula $\bigwedge_{F\in\mathcal{U}} F \wedge E$ requires degree strictly greater than $\delta s/(2d)$.*

Proof. This follows immediately by plugging Lemma 3.28 into Theorem 3.8. □

4 Applications

In this section, we demonstrate how to use the machinery developed in Section 3 to establish degree lower bounds for polynomial calculus. Let us warm up by reproving the bound from [AR03] for CNF formulas $\mathcal{F}$ whose clause-variable incidence graphs $G(\mathcal{F})$ are good enough expanders. We first recall the expansion concept used in [AR03] for ordinary bipartite graphs.

Definition 4.1 (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.

We can simply identify the $(\mathcal{U},\mathcal{V})_E$-graph with the standard clause-variable incidence graph $G(\mathcal{F})$ to recover the degree lower bound in [AR03] as stated next.

Theorem 4.2 ([AR03]). *For any CNF formula $\mathcal{F}$ and any constant $\delta > 0$ it holds that if the clause-variable incidence graph $G(\mathcal{F})$ is an (s,δ)-boundary expander, then the polynomial calculus degree required to refute $\mathcal{F}$ in polynomial calculus is $Deg(\mathcal{F}\vdash\bot) > \delta s/2$.*

Proof. To choose $G(\mathcal{F})$ as our $(\mathcal{U},\mathcal{V})_E$-graph, we set E to be the empty formula, $\mathcal{U}$ to be the set of clauses of $\mathcal{F}$ interpreted as one-clause CNF formulas, and $\mathcal{V}$ to be the set of variables partitioned into singleton sets. As E is an empty formula every set V respects it. Also, every neighbour of some clause $C \in \mathcal{U}$ is an E-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 $(s,\delta,0,E)$-expander, and hence by Corollary 3.29 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 $(\mathcal{U},\mathcal{V})_E$-graph is nontrivial, we show how the degree lower bound for the ordering principle formulas in [GL10] can be established using this framework. For an undirected (and in general non-bipartite) graph G, the *graph ordering principle formula* $GOP(G)$ says that there exists a totally ordered set of $|V(G)|$ elements where no element is minimal, since 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 consists of the following axiom clauses:

$$\overline{x}_{u,v} \vee \overline{x}_{v,w} \vee x_{u,w} \qquad u,v,w \in V(G), u \neq v \neq w \neq u \qquad \text{(transitivity)} \qquad (4.1a)$$

$$\overline{x}_{u,v} \vee \overline{x}_{v,u} \qquad u,v \in V(G), u \neq v \qquad \text{(anti-symmetry)} \qquad (4.1b)$$

$$x_{u,v} \vee x_{v,u} \qquad u,v \in V(G), u \neq v \qquad \text{(totality)} \qquad (4.1c)$$

$$\bigvee_{u\in N(v)} x_{u,v} \qquad v \in V(G) \qquad \text{(non-minimality)} \qquad (4.1d)$$

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 (4.1d) 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 4.3 (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 by Galesi and Lauria in [GL10] 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 4.4. With some modifications of the definition of E-respectful boundary in $(\mathcal{U}, \mathcal{V})_E$-graphs it would be possible to match the lower bound in [GL10], but it would also make the definitions more cumbersome and so we choose not to do so here.

Theorem 4.4 ([GL10]). *For a 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 $(\mathcal{U}, \mathcal{V})_E$-graph for $GOP(G)$, we let E consist of all transitivity axioms (4.1a), anti-symmetry axioms (4.1b), and totality axioms (4.1c). The non-minimality axioms (4.1d) 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 $F_u = \bigvee_{v \in N(u)} x_{v,u} \in \mathcal{U}$ are variable sets V_v where v is either equal to u or a neighbour of u in G. We can prove that each $V_v \in N(F_u)$ is an E-respectful neighbour of F_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 F_u as well as all the affected axioms in E. 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 E-respectfully satisfy F_u. Observe that this also shows that all $V_v \in \mathcal{V}$ respect E as required by Definition 3.4.

By the analysis above, it holds that the boundary $\partial(V')$ of some vertex set V' in G yields the E-respectful boundary $\partial_E\big(\bigcup_{u \in V'} F_u\big) \supseteq \{V_v \mid v \in \partial(V')\}$ in $(\mathcal{U}, \mathcal{V})_E$. Thus, the expansion parameters for $(\mathcal{U}, \mathcal{V})_E$ are the same as those for G and we can conclude that $(\mathcal{U}, \mathcal{V})_E$ is an $(s, \delta, 0, E)$-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 3.29 the degree of refuting $GOP(G)$ is greater than $\delta s/4$. □

With the previous theorem in hand, we can prove (a version of) the main result in [GL10], namely that there exists a family of 5-CNF formulas witnessing that the lower bound on size in terms of degree in Theorem 2.2 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 $\Omega\big(\sqrt{N}\big)$. This follows by plugging expanders with suitable parameters into Theorem 4.4. By standard calculations (see, for example, [HLW06]) 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 4.3 with high probability. By Theorem 4.4, graph ordering principle formulas on such graphs yield 5-CNF formulas over $\Theta\big(n^2\big)$ variables that require degree $\Omega(n)$. Since these formulas have polynomial calculus refutations in size $\mathrm{O}\big(n^3\big)$ (just mimicking the resolution refutations constructed in [Stå96]), this shows that the bound in Theorem 2.2 is essentially tight. The difference between this bound and [GL10] is that

since a weaker form of expansion is required in [GL10] it is possible to use 3-regular graphs, yielding families of 3-CNF formulas.

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 4.1 into respectful boundary expansion as in Definition 3.6. For a bipartite graph $G = (U \dot\cup V, E)$ the axioms appearing in the different versions of the graph pigeonhole principle formulas are as follows:

$$\bigvee_{v \in N(u)} x_{u,v} \qquad u \in U \qquad \text{(pigeon axioms)} \tag{4.2a}$$

$$\overline{x}_{u,v} \vee \overline{x}_{u',v} \qquad v \in V,\ u, u' \in N(v),\ u \neq u', \qquad \text{(hole axioms)} \tag{4.2b}$$

$$\overline{x}_{u,v} \vee \overline{x}_{u,v'} \qquad u \in U,\ v, v' \in N(u),\ v \neq v' \qquad \text{(functionality axioms)} \tag{4.2c}$$

$$\bigvee_{u \in N(v)} x_{u,v} \qquad v \in V \qquad \text{(onto axioms)} \tag{4.2d}$$

The "plain vanilla" *graph pigeonhole principle formula* PHP_G is the CNF formula over variables $\{x_{u,v} \mid (u,v) \in E\}$ consisting of clauses (4.2a) and (4.2b); the *graph functional pigeonhole principle formula* $FPHP_G$ contains the clauses of PHP_G and in addition clauses (4.2c); the *graph onto pigeonhole principle formula* $Onto\text{-}PHP_G$ contains PHP_G plus clauses (4.2d); and the *graph onto functional pigeonhole principle formula* $Onto\text{-}FPHP_G$ consists of all the clauses (4.2a)–(4.2d).

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 will use the observation from Section 2 that restricting a formula can only decrease the size and degree required to refute it.

As mentioned in Section 1, it was established already in [AR03] 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 [AR03] works also for the onto version $Onto\text{-}PHP_G$—by constructing an appropriate $(\mathcal{U}, \mathcal{V})_E$-graph. In addition, we can generalize the result in [AR03] slightly by allowing some additive slack $\xi > 0$ in the expansion in Theorem 3.8. This works as long as we have the guarantee that no too small subformulas are unsatisfiable.

Theorem 4.5. *Suppose that $G = (U \dot\cup V, E)$ is a bipartite graph with $|U| = n$ and $|V| = n - 1$ and that $\delta > 0$ is a constant such that*

- *for every set $U' \subseteq U$ of size $|U'| \leq s$ there is a matching of U' into V, and*
- *for every set $U' \subseteq U$ of size $|U'| \leq s$ it holds that $|\partial(U')| \geq \delta |U'| - \xi$.*

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

Proof sketch. The $(\mathcal{U}, \mathcal{V})_E$-graph for PHP_G is formed by taking $\mathcal{U}$ to be the set of pigeon axioms (4.2a), E to consist of the hole axioms (4.2b) and onto axioms (4.2d), and $\mathcal{V}$ to be the collection of variable sets $V_v = \{x_{u,v} \mid u \in N(v)\}$ partitioned with respect to the holes $v \in V$. It is straightforward to check that this $(\mathcal{U}, \mathcal{V})_E$-graph is isomorphic to the graph G and that all neighbours in $(\mathcal{U}, \mathcal{V})_E$ are E-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 $\mathcal{U}' \wedge E$ is satisfiable as long as $|\mathcal{U}'| \leq s$. Hence, we can apply Theorem 3.8 to derive the claimed bound. We refer to the upcoming full-length version of [MN14] for the omitted details. □

Theorem 4.5 is the only place in this paper where we use non-zero slack for the expansion. The reason that we need slack is so that we can establish lower bounds for another type of formulas, namely

the subset cardinality formulas studied in [Spe10, VS10, MN14]. A brief (and somewhat informal) description of these formulas is as follows. We start with a 4-regular bipartite graph to which we add an extra edge between two non-connected vertices. 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 again we refer to [MN14] for the details).

Definition 4.6 (Subset cardinality formulas [VS10, MN14]). Suppose that $G = (U \dot\cup V, E)$ is a bipartite graph that is 4-regular except that one extra edge has been added between two unconnected vertices on the left and right. Then the *subset cardinality formula* $SC(G)$ over G has variables $x_e, e \in E$, and clauses:

- $x_{e_1} \vee x_{e_2} \vee x_{e_3}$ for every triple e_1, e_2, e_3 of edges incident to any $u \in U$,
- $\overline{x}_{e_1} \vee \overline{x}_{e_2} \vee \overline{x}_{e_3}$ 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 4.1.

Definition 4.7 (Bipartite expander). A bipartite graph $G = (U \dot\cup V, E)$ 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, [HLW06]).

Theorem 4.8 ([MN14]). *Suppose that $G = (U \dot\cup V, E)$ 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 $S_{\mathcal{PCR}}(SC(G') \vdash \bot) = \exp\big(\Omega(n)\big)$.*

Proof sketch. The proof is by reducing to graph PHP formulas and applying Theorem 4.5 (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 4.5 (and Theorem 2.2) 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 4.5 (but adding the axioms (4.2c) and removing axioms (4.2d)). The naive idea would be to modify our $(\mathcal{U}, \mathcal{V})_E$-graph slightly by substituting the functionality axioms for the onto axioms in E while keeping $\mathcal{U}$ and $\mathcal{V}$ the same. This does not work, however—although the sets $V_v \in \mathcal{V}$ are E-respectful, the only assignment that respects E is the one that sets all variables $x_{u,v} \in V_v$ to false. Thus, it is not possible to satisfy any

of the pigeon axioms, meaning that there are no E-respectful neighbours in $(\mathcal{U}, \mathcal{V})_E$. In order to obtain a useful $(\mathcal{U}, \mathcal{V})_E$-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 4.9. *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 $(\mathcal{U}, \mathcal{V})_E$-graph from $FPHP_G$ as follows. We let the set of clauses E consist of all hole axioms (4.2b) and functionality axioms (4.2c). We define the family $\mathcal{U}$ to consist of the pigeon axioms (4.2a) interpreted as singleton CNF formulas. 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

$$V_v = \left\{ x_{u',v'} \middle| u' \in N(v) \text{ and } v' \in N(u') \right\} \ . \tag{4.3}$$

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 $(\mathcal{U}, \mathcal{V})_E$ as defined above satisfies the conditions in Corollary 3.29.

Note first that every variable set V_v respects the clause set E since setting all variables in V_v to false satisfies all clauses in E mentioning variables in V_v. It is easy to see from (4.3) that when a hole v is a neighbour of a pigeon u, the variable set V_v is also a neighbour in the $(\mathcal{U}, \mathcal{V})_E$-graph of the corresponding pigeon axiom $F_u = \bigvee_{v \in N(u)} x_{u,v}$. These are the only neighbours of the pigeon axiom F_u, as each V_v contains only variables mentioning pigeons in the neighbourhood of v. In other words, G and $(\mathcal{U}, \mathcal{V})_E$ share the same neighbourhood structure.

Moreover, we claim that every neighbour V_v of F_u is an E-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, F_u is satisfied by $\rho_{u,v}$. All axioms in E *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 E 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 (4.3) 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 (4.3) and is hence assigned to false. Thus, the assignment $\rho_{u,v}$ E-respectfully satisfies F_u, and so F_u and V_v are E-respectful neighbours as claimed.

Since our constructed $(\mathcal{U}, \mathcal{V})_E$-graph is isomorphic to the original graph G and all neighbour relations are respectful, the expansion parameters of G trivially carry over to respectful expansion in $(\mathcal{U}, \mathcal{V})_E$. This is just another way of saying that $(\mathcal{U}, \mathcal{V})_E$ is an $(s, \delta, 0, E)$-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 3.29. 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 2.2. □

It is not hard to show (again we refer to [HLW06] 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 $FPHP_n^{n+1} = FPHP_{K_{n+1,n}}$ with a restriction ρ_G setting $x_{u,v}$ to false for all $(u, v) \notin E$ to obtain $FPHP_n^{n+1}{\restriction_{\rho_G}} = FPHP_G$. 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 4.10. *Any polynomial calculus or polynomial calculus resolution refutation of (the standard CNF encoding of) the functional pigeonhole principle $FPHP_n^{n+1}$ requires size $\exp(\Omega(n))$.*

5 Concluding Remarks

In this work, we extend the techniques developed by Alekhnovich and Razborov [AR03] for proving degree lower bounds on refutations of CNF formulas in polynomial calculus. Instead of looking at the clause-variable incidence graph $G(F)$ of the formula F as in [AR03], we allow clustering of clauses and variables and reason in terms of the incidence graph G' defined on these clusters. We show that the CNF formula 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.

This provides us with a unified framework within which we can reprove previously established degree lower bounds in [AR03, GL10, MN14]. 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 [Raz02]. 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 [Raz15] 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 which 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 prove a lower bound for refuting the formulas encoding that a graph does not contain an independent set of size k, which were proven hard for resolution in [BIS07]? Or what about the formulas stating that a graph is k-colorable, for which resolution lower bounds were established in [BCMM05]?

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? For instance, do the formulas become easier if we have 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 [Raz04a] and Razborov [Raz01, Raz03, Raz04b], but as far as we are aware they remain wide open 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 [AR03] 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 F expansion of the clause-variable incidence graph $G(F)$ 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 we were quite surprised by this ourselves we want to elaborate briefly on this.

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 E 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, [Gre00]), the techniques in [AR03] 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 (as discussed in the

footnote after Definition 3.6) 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 in such a way that the effects of assignments to a group of variables can always be contained in a small neighbourhood of clauses/polynomials, which the assignments (mostly) satisify, 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 [AR03] are closer to being orthogonal rather than parallel. It would be desirable to gain a deeper understanding of what is going on here. In particular, in comparison to [AR03], 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 very interesting to find more general criteria of hardness that could capture both our approach and that of [AR03], and ideally provide a unified view of these lower bound techniques.

Acknowledgements

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 [AR03]. Finally, we are thankful for the feedback provided by the anonymous *CCC '15* referees and participants of the Dagstuhl workshop 15171 *Theory and Practice of SAT Solving* in April 2015.

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 Swedish Research Council grants 621-2010-4797 and 621-2012-5645.

References

[ABRW02] Michael Alekhnovich, Eli Ben-Sasson, Alexander A. Razborov, and Avi Wigderson. Space complexity in propositional calculus. *SIAM Journal on Computing*, 31(4):1184–1211, 2002. Preliminary version appeared in *STOC '00*.

[AR03] Michael Alekhnovich and Alexander A. Razborov. Lower bounds for polynomial calculus: Non-binomial case. *Proceedings of the Steklov Institute of Mathematics*, 242:18–35, 2003. Available at http://people.cs.uchicago.edu/~razborov/files/misha.pdf. Preliminary version appeared in *FOCS '01*.

[BCMM05] Paul Beame, Joseph C. Culberson, David G. Mitchell, and Cristopher Moore. The resolution complexity of random graph k-colorability. *Discrete Applied Mathematics*, 153(1-3):25–47, December 2005.

[BGIP01] Samuel R. Buss, Dima Grigoriev, Russell Impagliazzo, and Toniann Pitassi. Linear gaps between degrees for the polynomial calculus modulo distinct primes. *Journal of Com-*

puter and System Sciences, 62(2):267–289, March 2001. Preliminary version appeared in *CCC '99*.

[BIS07] Paul Beame, Russell Impagliazzo, and Ashish Sabharwal. The resolution complexity of independent sets and vertex covers in random graphs. *Computational Complexity*, 16(3):245–297, October 2007.

[Bla37] Archie Blake. *Canonical Expressions in Boolean Algebra*. PhD thesis, University of Chicago, 1937.

[BW01] Eli Ben-Sasson and Avi Wigderson. Short proofs are narrow—resolution made simple. *Journal of the ACM*, 48(2):149–169, March 2001. Preliminary version appeared in *STOC '99*.

[CEI96] Matthew Clegg, Jeffery Edmonds, and Russell Impagliazzo. Using the Groebner basis algorithm to find proofs of unsatisfiability. In *Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC '96)*, pages 174–183, May 1996.

[CR79] Stephen A. Cook and Robert Reckhow. The relative efficiency of propositional proof systems. *Journal of Symbolic Logic*, 44(1):36–50, March 1979.

[CS88] Vašek Chvátal and Endre Szemerédi. Many hard examples for resolution. *Journal of the ACM*, 35(4):759–768, October 1988.

[Fil14] Yuval Filmus. On the Alekhnovich–Razborov degree lower bound for the polynomial calculus. Manuscript. Available at http://www.cs.toronto.edu/~yuvalf/AlRa.pdf, 2014.

[GL10] Nicola Galesi and Massimo Lauria. Optimality of size-degree trade-offs for polynomial calculus. *ACM Transactions on Computational Logic*, 12:4:1–4:22, November 2010.

[Gre00] Frederic Green. A complex-number Fourier technique for lower bounds on the mod-m degree. *Computational Complexity*, 9(1):16–38, January 2000.

[Gri98] Dima Grigoriev. Tseitin's tautologies and lower bounds for Nullstellensatz proofs. In *Proceedings of the 39th Annual IEEE Symposium on Foundations of Computer Science (FOCS '98)*, pages 648–652, November 1998.

[Hak85] Armin Haken. The intractability of resolution. *Theoretical Computer Science*, 39(2-3):297–308, August 1985.

[HLW06] Shlomo Hoory, Nathan Linial, and Avi Wigderson. Expander graphs and their applications. *Bulletin of the American Mathematical Society*, 43(4):439–561, October 2006.

[IPS99] Russell Impagliazzo, Pavel Pudlák, and Jiří Sgall. Lower bounds for the polynomial calculus and the Gröbner basis algorithm. *Computational Complexity*, 8(2):127–144, 1999.

[MN14] Mladen Mikša and Jakob Nordström. Long proofs of (seemingly) simple formulas. In *Proceedings of the 17th International Conference on Theory and Applications of Satisfiability Testing (SAT '14)*, volume 8561 of *Lecture Notes in Computer Science*, pages 121–137. Springer, July 2014.

[Nor13] Jakob Nordström. Pebble games, proof complexity and time-space trade-offs. *Logical Methods in Computer Science*, 9:15:1–15:63, September 2013.

[Raz98] Alexander A. Razborov. Lower bounds for the polynomial calculus. *Computational Complexity*, 7(4):291–324, December 1998.

[Raz01] Alexander A. Razborov. Improved resolution lower bounds for the weak pigeonhole principle. Technical Report TR01-055, Electronic Colloquium on Computational Complexity (ECCC), July 2001.

[Raz02] Alexander A. Razborov. Proof complexity of pigeonhole principles. In *5th International Conference on Developments in Language Theory, (DLT '01), Revised Papers*, volume 2295 of *Lecture Notes in Computer Science*, pages 100–116. Springer, July 2002.

[Raz03] Alexander A. Razborov. Resolution lower bounds for the weak functional pigeonhole principle. *Theoretical Computer Science*, 1(303):233–243, June 2003.

[Raz04a] Ran Raz. Resolution lower bounds for the weak pigeonhole principle. *Journal of the ACM*, 51(2):115–138, March 2004. Preliminary version appeared in *STOC '02*.

[Raz04b] Alexander A. Razborov. Resolution lower bounds for perfect matching principles. *Journal of Computer and System Sciences*, 69(1):3–27, August 2004. Preliminary version appeared in *CCC '02*.

[Raz15] Alexander Razborov. Possible research directions. List of open problems (in proof complexity and other areas) available at http://people.cs.uchicago.edu/~razborov/teaching/, 2015.

[Rii93] Søren Riis. *Independence in Bounded Arithmetic*. PhD thesis, University of Oxford, 1993.

[Spe10] Ivor Spence. sgen1: A generator of small but difficult satisfiability benchmarks. *Journal of Experimental Algorithmics*, 15:1.2:1.1–1.2:1.15, March 2010.

[Stå96] Gunnar Stålmarck. Short resolution proofs for a sequence of tricky formulas. *Acta Informatica*, 33(3):277–280, May 1996.

[Urq87] Alasdair Urquhart. Hard examples for resolution. *Journal of the ACM*, 34(1):209–219, January 1987.

[VS10] Allen Van Gelder and Ivor Spence. Zero-one designs produce small hard SAT instances. In *Proceedings of the 13th International Conference on Theory and Applications of Satisfiability Testing (SAT '10)*, volume 6175 of *Lecture Notes in Computer Science*, pages 388–397. Springer, July 2010.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de