Electronic Colloquium on Computational Complexity, Report No. 38 (2024)

Polynomial Calculus for Quantified Boolean Logic: Lower Bounds through Circuits and Degree

Olaf Beyersdorff
Friedrich Schiller University Jena, Germany

Kaspar Kasche
Friedrich Schiller University Jena, Germany

Luc Nicolas Spachmann
Friedrich Schiller University Jena, Germany

Abstract

We initiate an in-depth proof-complexity analysis of polynomial calculus ($\mathcal{Q}$-PC) for Quantified Boolean Formulas (QBF). In the course of this we establish a tight proof-size characterisation of $\mathcal{Q}$-PC in terms of a suitable circuit model (polynomial decision lists). Using this correspondence we show a size-degree relation for $\mathcal{Q}$-PC, similar in spirit, yet different from the classic size-degree formula for propositional PC by Impagliazzo, Pudlák and Sgall (1999).

We use the circuit characterisation together with the size-degree relation to obtain various new lower bounds on proof size in $\mathcal{Q}$-PC. This leads to incomparability results for $\mathcal{Q}$-PC systems over different fields.

2012 ACM Subject Classification Theory of computation → Proof complexity

Keywords and phrases proof complexity, QBF, polynomial calculus, circuits, lower bounds

Funding *Olaf Beyersdorff*: Carl-Zeiss Foundation and DFG grant BE 4209/3-1
Kaspar Kasche: Carl-Zeiss Foundation

1 Introduction

Proof complexity studies the problem to understand the minimal size of proofs of specific formulas in various formal proof systems. The field bears deep connections to computational complexity [29, 43], logic – mainly, but not only through the correspondence to bounded arithmetic [8, 28, 43] – and has practical significance due to the intricate relations to SAT solving [24]. In fact, proof complexity is the main theoretical framework to assess the strength and limitations of solvers.

While traditionally proof complexity concentrated on propositional logic, there has been intense work in the past two decades on proof complexity for further logics, most notably for *Quantified Boolean Formulas* (QBF) [9], but also for other non-classical logics such as modal and intuitionistic logics [19, 38, 50]. For QBF, one of the main drivers for the field has been significant advances in QBF solving [18, 47]. As in the propositional case, QBF proof complexity provides the theoretical tools to model, assess and guide QBF solving [12, 22, 40].

In propositional proof complexity, various proof systems have been studied intensively, including resolution, Frege systems, algebraic and geometric systems [43]. While resolution has arguably received most attention – and underpins modern SAT solving in the form of CDCL [2, 5, 45] – algebraic proof complexity has enjoyed a boost of interest in the past decade with many strong results shown for Nullensatz, polynomial calculus (PC), sum of squares (SOS), and very strong systems such as the ideal proof system (cf. e.g. [27, 31, 33, 34, 36, 37, 46]). Algebraic proof systems typically work with polynomials and the central system of polynomial calculus [26] is a refutational proof system demonstrating that a given set of polynomial equations does not admit a common solution.

ISSN 1433-8092

Similarly, in QBF proof complexity there are many results on various QBF resolution systems [4,9,11,14]. Yet, in stark contrast to the propositional case, information on algebraic proof systems for QBF is rather scarce. A version of polynomial calculus for QBF – called $\mathcal{Q}$-PC here – is straightforward to define [13] as there is a general framework how to lift a line-based propositional proof system P – fulfilling some modest closure properties – to a quantified system $\mathcal{Q}$-P by adding just one rule of universal reduction that allows to substitute a universal variable u from a formula F (under the condition that u is quantified rightmost in F) [13]. This system $\mathcal{Q}$-PC naturally works with polynomials as lines and provides a succinct way to prove the falsity of QBFs. Hence we view this algebraic system as a refutational system for QBFs. The existential and universal variables are therefore propositional and take 0/1 values in accordance with the QBF semantics, while intermediate values computed by the polynomials can be arbitrary field elements, making proofs more succinct.

So far, the only information on proof size in $\mathcal{Q}$-PC stems from the general semantic technique of cost through the size-cost-capacity theorem from [10] which allows to obtain lower bounds for QBF proof systems of bounded capacity (which applies to $\mathcal{Q}$-PC as well as to most QBF resolution systems). With the cost technique, QBFs become hard to prove whenever the universal player needs large winning strategies (measured as the number of different answers of the universal player in the game interpretation of QBFs) and these lower bounds simultaneously hold in all QBF systems to which this technique is applicable. Hence this method does not allow to separate QBF resolution from $\mathcal{Q}$-PC, for example.

One key motivation to study algebraic proof systems in the propositional case is their recently emerging connection to algebraic circuit complexity [33, 37, 46]. In general, a correspondence between progress for lower bounds for circuit and proof size has often been postulated (e.g. [6]), but formal connections for propositional proofs could not yet be established outside the algebraic domain. In fact, it could be argued that this correspondence perfectly works in the QBF setting: for QBF resolution – tightly corresponding to a version of decision lists [11] – and for QBF Frege systems where proof size is characterised by circuit size in Boolean circuits [13]. This is quite fruitful as it allows a direct transfer of known circuit lower bounds to proof complexity, e.g. from $\mathsf{AC}^0[p]$ to the corresponding system of $\mathcal{Q}$-$\mathsf{AC}^0[p]$-Frege [13, 51]. A similar transfer in the propositional case remains wide open.

Curiously, an analogous relation between algebraic circuits and algebraic QBF systems is missing, whereas exactly in this algebraic case, some connections are known propositionally [33, 37, 46], as mentioned above.

Our aim in this paper is to initiate a comprehensive analysis of the algebraic system $\mathcal{Q}$-PC. In the course of this investigation we achieve a circuit characterisation for $\mathcal{Q}$-PC. This leads to new lower bound techniques for proof size in $\mathcal{Q}$-PC, which we apply to show a number of new proof size lower bounds for this system.

1.1 Our contributions

A. Circuit characterisation for $\mathcal{Q}$-PC. Our first result is a tight circuit characterisation of $\mathcal{Q}$-PC proof size by circuit size in an appropriate circuit model. The circuit model in question is a generalisation of decision lists [48], which are lists of simple statements of the form

If (condition on existential variables) Then (assignment to universal variables).

The decision lists – termed PDLs here for *p*olynomial *d*ecision *l*ists – have polynomial equations in existential variables as conditions and compute a complete assignment to the

universal variables. Semantically, a PDL for a quantified set of polynomial equations Φ computes a countermodel for Φ in the two-player game semantics of QBFs.

We show that the minimal proof size for Φ (of bounded quantifier complexity) in $\mathcal{Q}$-PC is polynomially equivalent to the minimal size of a PDL for Φ. In fact, we show a more general result that applies to a whole class of QBF proof systems with bounded capacity [10] (and fulfilling some closure properties). The result is parameterised by the lines of the proof system, which in turn correspond to the conditions in the decision lists. This generalises a result for Q-Resolution [11] and lifts it to $\mathcal{Q}$-PC.

B. Size-degree relation for $\mathcal{Q}$-PC. Having the PDL characterisation in place, we can obtain a size-degree result, relating minimum proof size in $\mathcal{Q}$-PC to the minimal degree of polynomials in the refutation. This is similar in spirit to the size-degree method known for propositional PC [39], albeit the actual relation is different and includes the quantifier depth of the formula. Technically the result is shown via the degree-preserving transfer from $\mathcal{Q}$-PC to PDLs and back explained above, together with an additional size-degree relation that we show for PDLs. Again the result and proof technique are similar to a prior size-width result for Q-Resolution [11].

C. New lower bounds for $\mathcal{Q}$-PC. Having both the PDL characterisation and size-degree relation at hand opens the door to a number of new lower bounds for degree and hence size in $\mathcal{Q}$-PC.

Specifically, we show that the parity and more generally the modulo k functions mod_n^k on n variables as well as the majority function maj_n all require high-degree PDLs over all subfields of $\mathbb{C}$. Using a general construction from [13, 14] we can turn any Boolean function f into a QBF $\mathcal{Q}$-f that has f as its only countermodel. Together with our results above this implies that the $\mathcal{Q}$-mod_n^k and $\mathcal{Q}$-maj_n QBFs require both linear degree and exponential monomial size in $\mathcal{Q}$-PC.

In addition to using the size-degree method to prove lower bounds for PDLs and hence for $\mathcal{Q}$-PC proofs, we show that for finite fields of characteristic p, PDLs can be efficiently transformed into $\mathsf{AC}^0[p]$ circuits. This allows to directly transfer circuit lower bounds of [51] into $\mathcal{Q}$-PC proof lower bounds. As a result, either if F and G are both finite fields of different characteristics, or if F is finite and G is a subfield of $\mathbb{C}$, then the systems $\mathcal{Q}$-PC over F and G are incomparable.

In fact, all our lower bounds are very strong as they apply to a succinct model of QBF proof systems were propositional sub-derivations – for PC comprised of additions and multiplications of polynomials – can be abbreviated as semantic entailment steps that are checked with an NP oracle [17]. This implies that all our lower bounds and incomparability results also hold in the traditional proof model with 'unfolded' computations, but remain even valid in the mentioned stronger NP oracle model.

1.2 Organisation

We start in Section 2 by reviewing background material from proof complexity and QBF. Section 3 contains our characterisation of proof size in $\mathcal{Q}$-PC by polynomial decision lists. This is used in Section 4 to prove the size-degree relation for $\mathcal{Q}$-PC and applied to obtain a number of new exponential size lower bounds. In Section 5 we show how to transform PDLs over finite fields of characteristic p into $\mathsf{AC}^0[p]$ circuits and obtain incomparability results for $\mathcal{Q}$-PC systems over different fields. We conclude in Section 6 with open questions and a discussion on the wider implications of our circuit characterisation from Section 3 for the proof systems cutting planes and Frege.

2 Preliminaries

We assume familiarity with basic notions from computational complexity, cf. [1], as well as from logic, cf. [42], and algebra, cf. [44], but define all specific concepts needed in this paper.

We consider propositional formulas φ built from constants $0, 1$, the usual operators $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$, and propositional variables. A literal is a variable v or its negation $\overline{v}$. A clause is a disjunction of literals, and a formula is in Conjunctive Normal Form (CNF) if it is a disjunction of clauses. When V is a set of variables, a (partial) assignment to V is a (partial) function $\alpha : V \to \{0, 1\}$. We write $\langle V \rangle$ for the set of all complete assignments to V, and $\mathrm{vars}(\varphi)$ or $\mathrm{vars}(\alpha)$ for the set of all variables occuring in φ or α. For $V' \subseteq V$, we denote by $\alpha|_{V'}$ the restriction of α to only the variables in V'. We denote by $\varphi[\alpha]$ the formula φ where each variable $v \in \mathrm{vars}(\alpha)$ has been substituted by $\alpha(v)$, and by $\varphi[v_1/\theta_1, \ldots, v_k/\theta_k]$ the formula φ where variables v_i have been substituted by formulas θ_i.

Circuit classes. We recall the definitions of standard circuit classes used in this paper. The class AC^0 contains all languages recognizable by polynomial-size circuits over the Boolean basis $\neg$, $\vee$, $\wedge$ with bounded depth and unbounded fan-in. The class $\mathsf{AC}^0[p]$ uses bounded-depth circuits with MOD_p gates determining whether the sum of the inputs is 0 modulo p. $\mathsf{P/poly}$ uses circuits of polynomial size but arbitrary depth. For an in-depth account on circuit complexity we refer to [52].

Proof systems. According to [29] a *proof system* for a language $\mathcal{L}$ is a polynomial-time onto function $P : \{0, 1\}^* \to \mathcal{L}$. Each string $\varphi \in \mathcal{L}$ is a *theorem* and if $P(\pi) = \varphi$, π is a *proof* of φ in P. Given a polynomial-time function $P : \{0, 1\}^* \to \{0, 1\}^*$ the fact that $P(\{0, 1\}^*) \subseteq \mathcal{L}$ is the *soundness property* for $\mathcal{L}$ and the fact that $P(\{0, 1\}^*) \supseteq \mathcal{L}$ is the *completeness property* for $\mathcal{L}$. Proof systems for the language TAUT of propositional tautologies are called *propositional proof systems* and proof systems for the language TQBF of true QBF formulas are called *QBF proof systems*. Equivalently, propositional proof systems and QBF proof systems can be defined respectively for the languages UNSAT of unsatisfiable propositional formulas and FQBF of false QBF formulas, in this second case we call them *refutational*. The proofs of some proof systems consist of a sequence of statements called *lines*, where each line is derived according to certain rules. These proof systems are called *line-based*. A propositional *base system* is a line-based proof system with certain very natural restrictions that are formally defined in [10]. All propositional proof systems discussed in this paper are base systems.

Polynomial Calculus. Polynomial Calculus (PC) [26] is an algebraic proof system that can be used to show that a set of polynomials does not have common roots. For a set of variables $V = \{v_1, \ldots, v_n\}$ over a field F, its lines are polynomial equations $0 = \sum_{i=1}^{m} c_i \prod_{v \in V_i} v$ with $m \in \mathbb{N}, c_i \in F, V_i \subseteq V$. A PC refutation starts with a set of polynomials, derives new polynomials that are linear combinations of previous ones, and ends with the contradiction $0 = 1$. The size of a polynomial is its number of monomials, and the size of a PC refutation is the sum of the sizes of its polynomials.

Here, we view PC as a propositional proof system, and allow only the values 0 and 1 for each variable. To maintain the system's completeness for these restricted semantics, we introduce the Boolean axioms $v^2 - v = 0$ for each $v \in V$ that are available for use in every refutation. In order to represent literals and clauses efficiently, we introduce complementary variables $\overline{v}$ that are required to have the value $1 - v$, and introduce axioms $v + \overline{v} - 1 = 0$.

Perhaps counterintuitively, a variable that is 0 in a polynomial corresponds to a propositional variable that is true, and 1 corresponds to false. We recall that a polynomial equation is true if its polynomial equals 0. This way, a monomial corresponds to a clause containing

the same literals, and a propositional formula in CNF can be efficiently encoded as a system of monomial equations.

When p is a polynomial, v a variable, and $c \in \{0,1\}$, we denote by $p[v/c]$ the polynomial p where v has been replaced by c and $\overline{v}$ by $1-c$; this is defined analogously when assigning multiple variables at once.

Frege systems. Frege proof systems are the common 'textbook' proof systems for propositional logic based on axioms and rules [29]. The lines in a Frege proof are propositional formulas built from propositional variables x_i and Boolean connectives $\neg$, $\wedge$, and $\vee$. A Frege system comprises a finite set of axiom schemes and rules; however, the exact choice of the axiom schemes and rules does not matter as any two Frege systems are equivalent [29, 42]. A Frege *proof* is a sequence of formulas where each formula is either a substitution instance of an axiom, or can be inferred from previous formulas by a valid inference rule. If a Frege proof uses every derived formula at most once in another derivation, it is called *treelike*; a treelike Frege system is one where all proofs are required to have this property.

Given a circuit class $\mathscr{C}$, the $\mathscr{C}$-Frege system is restriction of Frege where the lines have to be circuits from $\mathscr{C}$. An exact definition of $\mathscr{C}$-Frege is contained in [41]. *Resolution* (Res) is a particular kind of Frege system introduced by [21, 49], where lines are clauses.

Cutting Planes. Cutting Planes (CP) [30] is a proof system for integer linear programming. Its lines are linear inequalities with integer coefficients, and new lines are derived either by linear combination of previous lines, or by dividing a previous line by a positive integer and rounding appropriately. The end of a refutation consists of the contradiction $0 \geqslant 1$. Cutting Planes can be used as a propositional proof system by introducing the Boolean axioms $v \geqslant 0$ and $v \leqslant 1$ for each variable v.

Quantified Boolean Formulas. A (closed prenex) *Quantified Boolean Formula* (QBF) is a formula where quantifiers are introduced to propositional logic. Each variable is quantified at the beginning of the formula, using either an existential or universal quantifier. We denote such formulas as $Q\,\varphi$, where φ is a propositional Boolean formula called *matrix*, and Q is its *quantifier prefix*. We typically use x_i for existentially quantified variables and u_i for universally quantified variables. When a system includes complementary variables $\overline{v}$ as in PC, those do not occur in the quantifier prefix. Their values are determined by the corresponding variables v instead.

Whether a QBF is true or not can be defined recursively depending on its first quantifier. The formula $\forall u\, Q\, \varphi$ is true if both $Q\,\varphi[u=0]$ and $Q\,\varphi[u=1]$ are true. The formula $\exists x\, Q\, \varphi$ is true if at least one of $Q\,\varphi[x=0]$ and $Q\,\varphi[x=1]$ is true. When a QBF proof system is derived from an algebraic system such as PC, it nonetheless has these Boolean semantics.

In a fully quantified prenex QBF, the quantifier prefix determines a total order of the variables. Given a variable y, we will sometimes refer to the variables preceding y in the prefix as variables *left* of y; analogously we speak of the variables *right* of y.

A QBF $Q_1 x_1 \cdots Q_k x_k\, \varphi$ can be seen as a game between two players: *universal* ($\forall$) and *existential* ($\exists$). In the i-th step of the game, the player Q_i assigns a value to the variable x_i. The existential player wins if φ evaluates to 1 under the assignment constructed in the game. The universal player wins if φ evaluates to 0. Given a universal variable u with index i, a *strategy for* u is a function from all variables of index $< i$ to $\{0,1\}$. A QBF is false if and only if there exists a *winning strategy* for the universal player, that is if the universal player has a strategy for all universal variables that wins any possible game [1, 35].

3 A general characterisation of $\mathcal{Q}$-P proof size by decision lists

We start by characterising proof size in $\mathcal{Q}$-PC by a suitable circuit model. In fact, we will show a more general result that applies to a class of QBF proof system which are lifted from a propositional base system P to the QBF system $\mathcal{Q}$-P. This lifting is done by adding the $\forall$red rule to the rules of P. The $\forall$red rule allows to derive $l[\mu]$ from a line l and a propositional assignment μ to universal variables, as long as vars(μ) occur after all existential variables in l in the quantifier prefix [13].

However, lower bounds for the resulting $P + \forall$red system are trivial to obtain from lower bounds for P by existentially quantifying all variables. We are not too interested in such bounds, but in 'genuine' QBF lower bounds that arise from quantifier alternation (cf. [17, 25] for a longer discussion and details). In other words, we want to filter out any propositional hardness in a QBF by ignoring purely propositional sub-derivations. For this, we introduce the Sem rule for semantic steps. It can derive a line l from a line r if $r \models l$. In general this inference step cannot be checked efficiently, but needs an NP oracle call [17]. Using Sem steps also removes the need for any other propositional inference rules as these can be carried out by Sem. We call the resulting system $\mathcal{Q}$-P.

▶ **Definition 1** ($\mathcal{Q}$-P). *Let P be a propositional base system. The system $\mathcal{Q}$-P is a refutational proof system for QBF that has the same lines as P, and rules* $\forall$red *and* Sem.

The system we are most interested in is $\mathcal{Q}$-PC where the lines are polynomials. We briefly review the semantics of this system. As specified in Definition 1, its only rules are $\forall$red and Sem. Its lines are polynomial equations with coefficients from a field F. The $\forall$red rule can be applied to a polynomial p to obtain $p[u/0]$ (which is p with variable u set to 0 and $\overline{u}$ set to 1) or $p[u/1]$ (which is p with variable u set to 1 and $\overline{u}$ set to 0) as long as u is quantified universally right of all existential variables in p. This also means that u cannot be a complementary variable $\overline{v}$. The Sem rule allows to derive polynomial equations that semantically follow from previous equations, the Boolean axioms, and the $v + \overline{v} = 1$ axioms. Semantically, the variables can only take the values 0 or 1, and $\overline{v}$ must always take the value $1 - v$. The propositional rules of PC can be added, but are not strictly needed and do not shorten proofs in the presence of Sem.

We now define the circuit model that we will use for the characterisation of $\mathcal{Q}$-P proof size. The model is a variant of decision lists [10, 48].

▶ **Definition 2** (P-UDL). *Let P be a base system and X, U sets of variables. A P-UDL of length k is a sequence $L = (p_1, \mu_1), (p_2, \mu_2), \ldots, (p_k, \mu_k)$ where $(\neg p_i)$ is a line of P,* vars$(p_i) \subseteq X$, *$\mu_i \in \langle U \rangle$ for each $i \in [k]$ and $p_k = \bot$. It computes a function $f_L : \langle X \rangle \to \langle U \rangle$ with $f_L(\alpha) = \mu_j$ for the smallest j such that $\alpha \models p_j$.*

Intuitively, a P-UDL checks as conditions p_j negations of lines of P in existential variables and outputs full assignment μ_j to universal variables.

Again the main instantiation of this definition for us is to choose P as PC. To ease notation we abbreviate PC-UDL by PDL for *p*olynomial *d*ecision *l*ists. The lines of PDLs are then polynomials. As lines in PC are polynomials p with the implicit meaning of $p = 0$, conditions in a PDL check that the polynomial is *not* zero. Hence a line p in a PDL becomes active, if the p does not evaluate to 0 under the assignment.

We use P-UDLs to compute countermodels for QBFs. More formally, we say that a P-UDL L is *correct* for a QBF $\mathcal{Q}.\varphi$ if it has all of the following properties:

- All variables in X are existential variables of $\mathcal{Q}$.

- All universal variables of $\mathcal{Q}$ are in U.
- Let $\alpha, \beta \in \langle X \rangle, u \in U$. If $f_L(\alpha)$ and $f_L(\beta)$ disagree on u, then α and β disagree on a variable $x \in X$ that occurs before u in $\mathcal{Q}$.
- For every $\alpha \in \langle X \rangle$, $\alpha \wedge f_L(\alpha)$ falsifies φ.

For P-UDL $L = (p_1, \mu_1), \ldots, (p_k \mu_k)$, we define the *size* of L $|L| = \sum_{i=1}^{k} size_P(\overline{p_i})$, where $size_P$ corresponds to the respective size measure in the base system P. Additionally, the length of L is defined as $\text{len}(L) = k$.

Our goal is to use P-UDLs to characterise the hardness of $\mathcal{Q}$-P proofs on QBFs of constant alternation depth, i.e. to show that there exists a short P-UDL for a QBF Φ if and only if there exists a short $\mathcal{Q}$-P proof of Φ.

From the definition of P-UDL, it is apparent that the size of P-UDL for a QBF Φ is always at least as big as the size of the smallest countermodel of Φ, since each line always assigns all universal variables. As such, P-UDL cannot possibly characterise proof size in $\mathcal{Q}$-P for all base systems P. In fact, we need three restrictions on $\mathcal{Q}$-P for the characterisation to work. Firstly, the negations of the lines of P must be able to succinctly represent assignments to variables. Secondly, the base system P must be *closed under disjunction*. This is a syntactic restriction which demands, that for two arbitrary lines l and p of P, the disjunction $l \vee p$ is also a valid line in P.

▶ **Definition 3** (Closed under disjunction). *Let l and p be lines in a proof system $\mathcal{P}$. We say $\mathcal{P}$ is closed under disjunction if $l \vee p$ is also a line in $\mathcal{P}$ with size $\mathcal{O}(|l| \cdot |p|)$.*

Thirdly, we require $\mathcal{Q}$-P to have limited *capacity*. Capacity is a measure introduced in [10], which counts how many different answers the universal player needs at most to respond to one line in a $\mathcal{Q}$-P proof π. Formally, the definition of capacity builds upon the definitions of *Response Maps* and *Response Map Sets*. A response map is a function specifying the universal answer to existential assignments to a single line in a proof.

▶ **Definition 4** (Response Map [10]). *Let P be a base system, let L be a line in $\mathcal{Q}$-P whose rightmost block U is universal, and let $X = \text{vars}(L) \setminus U$. A response map for L with respect to the quantifier prefix Q is a function $\mathcal{R} : \langle X \rangle \to \langle U \rangle$ satisfying the following for each $\alpha \in \langle X \rangle$:*

If $L[\alpha]$ is not a tautology, then $\mathcal{R}[\alpha]$ falsifies $L[\alpha]$.

A response map set for a proof is a set containing response maps for each reducible line of the proof. We call line L *reducible* if its rightmost block is universally quantified.

▶ **Definition 5** (Response Map Set [10]). *Given a $\mathcal{Q}$-P refutation π of a QBF $Q.\varphi$ whose reducible lines are $L_1, \ldots, L_k$, a response map set for π is a set $\{\mathcal{R}_1, \ldots, \mathcal{R}_k\}$ in which each $\mathcal{R}_i$ is a response map for L_i with respect to Q.*

▶ **Definition 6** (Capacity [10]). *Let π be a $\mathcal{Q}$-P refutation of a QBF $Q.\varphi$. The capacity of a response map set $\{\mathcal{R}_1, \ldots, \mathcal{R}_k\}$ for π is $\max_{i \in [k]}\{|\text{rng}(\mathcal{R}_i)|\}$. The capacity of π is the minimal capacity of any response map set for π.*

In particular, we require that the capacity of $\mathcal{Q}$-P is at most polynomial in the size of π for all proofs π. We remark that $\mathcal{Q}$-PC and $\mathcal{Q}$-Res (as well as the QBF cutting planes system $\mathcal{Q}$-CP) all have bounded capacity [10].

We show that for proof systems with bounded capacity we can always assume that universal reductions reduce a full quantifier block of universal variables in just one step (instead of reducing the variables one by one).

▶ **Proposition 7.** *Let $\Phi = \mathcal{Q}.\varphi$ be a false QBF and π be a refutation of Φ in $\mathcal{Q}$-P. Then there exists a proof π', such that each reduction reduces a complete universal block with $|\pi'| \leqslant |\pi| \cdot \text{capacity}(P, \pi)$.*

Proof. Let $\pi = (c_1, \ldots, c_s)$ and $\mathcal{R} = (R_1, \ldots, R_n)$ be a response map set for π. Let R_i^k be an enumeration of the range of R_i. We iteratively construct a new proof with the desired properties. $\pi'_0 = \emptyset$ and

$$\pi'_{i+1} = \begin{cases} \pi'_i, c_i & \text{if } c_i \text{ is an axiom or} \\ & \text{derived with semantic entailment} \\ \pi'_i, c_j[R_j^1/U_\ell], \ldots, c_j[R_j^{k_j}/U_\ell], c_i & \text{if } c_i \text{ is derived from } c_j \text{ using} \\ & \text{universal reduction in } u \in U_\ell. \end{cases}$$

It remains to be shown that π'_s is a valid $\mathcal{Q}$-P refutation, i.e. each c'_i is either an axiom, entailed by $\varphi, c'_1, \ldots, c'_{i-1}$ or a reduction of c'_j for some $j < i$.

- c'_i was added by case (1) and is an axiom. Then there is nothing to show.
- c'_i was added by case (1) and was derived with a semantic entailment. Then $c'_i = c_k$ for some $c_k \in \pi$, which was also derived using semantic entailment. Since $\{c_1, \ldots, c_{k-1}\} \subseteq \{c'_1, \ldots, c'_{i-1}\}$, c'_i is a valid entailment.
- c'_i was added by case (2) as a reduction of some c_j. Since $\{c_1, \ldots, c_{k-1}\} \subseteq \{c'_1, \ldots, c'_{i-1}\}$, it is a valid reduction.
- c'_i was added by case (2) but is not a reduction. Per construction, $c'_i = c_k$ for some $c_k \in \pi$, which was derived by a universal reduction from c_j. Additionally, we know that $c_j, c_j[R_j^1/U_\ell], \ldots, c_j[R_j^{k_j}/U_\ell]$ are in π'. We show that $c_j[R_j^1/U], \ldots, c_j[R_j^{k_j}/U] \models c'_i$.
  Let $\alpha \in \langle X \rangle$ be an assignment with $\alpha \models \{c_j[R_j^1/U], \ldots, c_j[R_j^{k_j}/U]\}$. Per definition of response maps, $c_j[\alpha]$ is a tautology. As such, $c_j[\alpha][\beta]$ is true for each (partial) universal assignment β. Since α and β are disjoint, $c_j[\alpha][\beta] = c_j[\beta][\alpha]$ and $\alpha \models c_i[\beta]$. Since applying a universal assignment is equivalent to a universal reduction, every reduction of c_j, in particular $c'_i = c_k$, is semantically entailed from $\{c_j[R_j^1/U], \ldots, c_j[R_j^{k_j}/U]\}$.

◀

We recall the definition of the direct product of two decision lists [11]. The direct product combines two decision lists (which produce outputs for different universal variables, tentatively from different universal blocks) into one list.

▶ **Definition 8** (Direct product [11])**.** *Let $\mathcal{C}$ be the negation of a line in a proof system P. P must be closed under disjunction. Let X_1, U_1, X_2, U_2 be pairwise-disjoint sets of Boolean variables. Given two multi-output $\mathcal{C}$ decision lists $L := (\varepsilon_1, \mu_1), \ldots, (\varepsilon_s, \mu_s)$ and $M := (\delta_1, \nu_1), \ldots, (\delta_t, \nu_t)$, where $\text{vars}(\varepsilon_i) \subseteq X_1, \text{vars}(\mu_i) = U_1, \text{vars}(\delta_j) \subseteq X_1 \cup U_1 \cup X_2$ and $\text{vars}(v_j) = U_2$, the direct product $L \times M$ is the decision list*

$$\begin{array}{l} (\varepsilon_1 \wedge \delta_1[\mu_1], \mu_1 \wedge \nu_1), \cdots, (\varepsilon_s \wedge \delta_1[\mu_s], \mu_s \wedge \nu_1) \\ \qquad\qquad\qquad\qquad \vdots \\ (\varepsilon_1 \wedge \delta_t[\mu_1], \mu_1 \wedge \nu_t), \cdots, (\varepsilon_s \wedge \delta_t[\mu_s], \mu_s \wedge \nu_t). \end{array}$$

Note that the size of the direct product is the product of the sizes of the decision lists (up to a constant factor).

▶ **Proposition 9** ([11]). *Let X_1, U_1, X_2, U_2 be pairwise-disjoint sets of Boolean variables and let L and M be multi-output decision lists computing $f : \langle X_1 \rangle \to \langle U_1 \rangle$ and $g : \langle X_1 \cup U_1 \cup X_2 \rangle \to \langle U_2 \rangle$. Then $L \times M$ computes the function*

$$\begin{aligned} f \times g : \quad \langle X_1 \cup X_2 \rangle &\to \langle U_1 \cup U_2 \rangle \\ \tau &\mapsto f(\tau|_{X_1}) \wedge g(\tau \wedge f(\tau|_{X_1})). \end{aligned}$$

Proof. Let $L = (\varepsilon, \mu_1), \ldots, (\varepsilon_s, \mu_s), M = (\delta_1, \nu_1), \ldots, (\delta_t, \nu_t)$. Let $\tau \in \langle X_1 \cup X_2 \rangle$ and a and b be the least natural numbers such that $\tau|_{X_1}$ satisfies ε_a and τ satisfies $\delta_b[\mu_a]$. By definition of decision lists, $(L \times M)(\tau) = \mu_a \wedge \nu_b$.

$L(\tau|_{X_1}) = \mu_a$ by the definition of decision lists and therefore $\tau \wedge L(\tau|_{X_1}) = \tau \wedge \mu_a$. Aiming for contradiction, suppose that $M(\tau \wedge \mu_a) \neq \nu_b$. Since τ satisfies $\delta_b[\mu_a]$, $\tau \wedge \mu_a$ satisfies δ_b. Therefore. $\tau \wedge \mu_a$ satisfies some $\delta_{b'}$ with $b' < b$. It follows that τ satisfies $\delta_{b'}[\mu_a]$, contradicting the minimality of b. ◀

We are now ready to characterise proof size in $\mathcal{Q}$-P by the size of P-UDLs.

▶ **Theorem 10.** *Let P be a line-based, propositional proof system, where the lines are closed under disjunction, such that $\mathcal{Q}$-P has at most polynomial capacity. Then for QBFs Φ with bounded quantifier alternation depth, we can efficiently transform a P-*UDL *for Φ into a $\mathcal{Q}$-P refutation for Φ and vice versa. In particular, the minimal size of a P-*UDL *for Φ is polynomially equivalent to the size of the minimal $\mathcal{Q}$-P refutation for Φ.*

Proof sketch. This proof follows [11]. We will first give a proof sketch to summarise the ideas, as the complete proof is long and technical. *From $\mathcal{Q}$-P to P-*UDL. Let a QBF Φ of alternation depth d and a $\mathcal{Q}$-P refutation π of Φ be given. With Lemma 7, we can assume that π always uses universal reductions on entire blocks of universal variables. Using strategy extraction [3, 11, 32], we can extract a set of P-decision lists computing a countermodel of Φ from π. Each of these decision lists computes the universal strategy for one of the universal blocks and has a size of at most $|\pi|$. The direct product of all the decision lists gives the wanted P-UDL. Proposition 9 shows that the resulting decision list computes the correct function. Since there are d decision lists of size $|\pi|$, the computed P-UDL has size $\mathcal{O}(|\pi|^d)$.

*From P-*UDL *to $\mathcal{Q}$-P.* Let a QBF Φ of alternation depth d and a P-UDL L computing a countermodel of Φ be given. We define the *entailment sequence* $\mathcal{E}$ of L recursively in d.

- if $d = 1$, $\mathcal{E}(L) := \overline{\varepsilon_1} \vee \overline{\mu_1}, \ldots, \overline{\varepsilon_s} \vee \overline{\mu_s}$
- if $d \geqslant 2$, for each $i \in [s]$ define L_i as the list obtained from L by replacing the first $i-1$ existential terms by their X_1 components and setting all U_1 components to $\mu_i|_{U_1}$. $\mathcal{E}(L)$ is the sequence $\pi_1, \ldots, \pi_s$, where $\pi_i := (\overline{\varepsilon_i}|_{X_1} \vee \overline{\mu_i}|_{U_1}) \otimes \mathcal{E}(L_i[\varepsilon_i|_{X_1} \wedge \mu_i|_{U_i}])$.

Here, the $\otimes$-operator between a line of P and the entailment sequence defines an elementwise disjunction between the line and each element in the entailment sequence, i.e. $\ell \otimes (c_1, c_2, \ldots, c_r) = (\ell \vee c_1, \ell \vee c_2, \ldots, \ell \vee c_r)$. For a line ℓ, we call $\mathrm{red}(\ell)$ the line obtained by using universal reduction on ℓ by some specific universal assignment. Since negations of lines of P can succinctly represent assignments, $\overline{\mu_i}$ can be represented in P. If $\mathcal{E}(L) = (c_1, \ldots, c_r)$, then $(c_1, \mathrm{red}(c_1), c_2, \mathrm{red}(c_2), \ldots, c_r, \mathrm{red}(c_r))$ is a $\mathcal{Q}$-P refutation of Φ. ◀

Formal proof. This proof follows directly from [11], since the arguments are almost completely semantic. On a syntactic level, the only necessary conditions is the possibility of the direct product, which is given by Proposition 9 and the allowance of blockwise reduction, which can be assumed because of Proposition 7. For completeness, the proof is nonetheless given here.

Let $\Phi = \exists X_1 \forall U_1 \cdots \exists X_d \forall U_d \exists X_{d+1} \varphi$ be a QBF.

*From $\mathcal{Q}$-P to P-*UDL. Let $\pi = c_1 \dots c_s$ be a $\mathcal{Q}$-P refutation of Φ, which only uses blockwise reductions. Using strategy extraction, we will extract a sequence of multi-output P decision lists, one for each block, as follows: For each $i \in [d]$ and $j \in [s+1]$, we define $L_i^{s+1} := (\top, \alpha_i)$, where α_i is some fixed assignment to U_i; for each $j \in [s], L_i^j := (\overline{c_j}, \mu), L_i^{j+1}$ if c_j was derived by universal reduction due to $\mu \in \langle U_i \rangle$, and $L_i^j := L_i^{j+1}$ otherwise. By backwards induction on $j \in [s+1]$, applying Proposition 9, it is shown that

$$L_j := L_1^j \times (L_2^j \times \dots \times (L_{d-1}^j \times L_d^j) \dots)$$

is a UDL for $\mathcal{Q} \cdot \varphi \wedge \bigwedge_{k=1}^{j-1} c_k$. As such, L_1 is a P-UDL for Φ. Each L_i has size at most $|\pi|$, and the size of the direct product of decision lists is the product of the sizes of the decision lists. Therefore, $|L_1| = \mathcal{O}(|\pi|^d)$.

*From P-*UDL *to $\mathcal{Q}$-P.* Let $L = (\varepsilon_1, \mu_1), \dots (\varepsilon_s, \mu_s)$ be a P-UDL computing a countermodel for Φ. We require some additional notation for this proof. Given a line b and a sequence of lines π of $\mathcal{Q}$-P, we define

$$b \otimes \pi = b \vee c_1, \dots b \vee c_s.$$

Additionally, for a block Z of Φ, the Z component of (ε_i, μ_i) is $(\varepsilon_i, \mu_i)|_Z$ To transform L into a $\mathcal{Q}$-P proof, we first define the *entailment sequence* $\mathcal{E}(L)$ of L. The entailment sequence is defined recursively on the alternation depth d of Φ.

- if $d = 1$, $\mathcal{E}(L) := \overline{\varepsilon_1} \vee \overline{\mu_1}, \dots, \overline{\varepsilon_s} \vee \overline{\mu_s}$
- if $d \geqslant 2$, for each $i \in [s]$ define L_i as the list obtained from L by replacing the first $i-1$ existential terms by their X_1 components and setting all U_1 components to $\mu_i|_{U_1}$. We define $\mathcal{E}(L)$ as the sequence $\pi_1, \dots, \pi_s$, where $\pi_i := (\overline{\varepsilon_i}|_{X_1} \vee \overline{\mu_i}|_{U_1}) \otimes \mathcal{E}(L_i[\varepsilon_i|_{X_1} \wedge \mu_i|_{U_i}])$.

First, we show that $|\mathcal{E}(L)| \in \mathcal{O}(|L|^d)$ by induction over d. In the base case $d = 1$, the entailment sequence consists only of the negations of the lines in L. As such, the statement holds. In the induction step, we build the disjunction of restrictions of our lines of L and the previous level of the entailment sequence. Since $\mathcal{Q}$-P is closed under disjunction, the size of disjunctions is at most the product of the sizes (upto some constant). The statement follows using the induction hypothesis.

Next, for a line ℓ of the entailment sequence we take the partial universal assignment v, which assigns the universal variables of the rightmost quantifier block occurring in ℓ. In particular, v maps u to true, if and only if $\overline{u}$ is in ℓ. This assignment is unique, because the construction of $\mathcal{E}(L)$ doesn't allow universal variables to appear in both polarities in a single line. We write $\mathrm{red}(\ell) = l[v]$ for the clause obtained by maximum reduction.

To obtain the final result, we will need to prove the following statement:

Let $\mathcal{E}(L) = c_1, \dots, c_r$ be the entailment sequence of L. The c_r is fully universal, and, for each $i \in [r]$,

$$\varphi \wedge \bigwedge_{j=1}^{i-1} \mathrm{red}(c_j) \models c_i.$$

Without loss of generality, we can assume that the X_{d+1} components of L are all empty and that the final existential query $\varepsilon_s = \top$. We proceed by induction on the alternation depth d of Φ. Let $i \in [r]$.

Base case $d = 1$. In this case $r = s, c_i = \overline{\varepsilon_i} \vee \overline{\mu_i}$, and $\mathrm{red}(c_i) = \overline{\varepsilon_i}$. Let τ be a total assignment falsifying $\overline{\varepsilon_i} \vee \overline{\mu_i}$. If the existential part $\tau_\exists$ satisfies $\bigvee_{k=1}^{-1} \varepsilon_k$, then it falsifies

$$\bigwedge_{k=1}^{i-1} \overline{\varepsilon_k} = \bigwedge_{k=1}^{i-1} \mathrm{red}(c_k).$$

Otherwise, since $\tau_\exists$ satisfies ε_i, and the universal part $\tau_\forall$ is equal to μ_i, τ falsifies φ by definition of countermodel. Since $\varepsilon_s = \top, c_s = \bot \vee \mu_s$ is fully universal.

Inductive step $d \geqslant 1$. For each $j \in [s]$, we put

$$\alpha_j = \varepsilon_j|_{X_1} \wedge \mu_j|_{U_1},$$

and claim that $L_j[\alpha_j]$ is a P-UDL for

$$Q_j = P[\alpha_j] \cdot \left(\varphi \wedge \bigwedge_{k=1}^{j-1} \overline{\varepsilon_k}|_{X_1}\right)[\alpha_j],$$

which is a QBF of alternation depth $d-1$. We prove this claim later.

Let p and q be natural numbers such that

$$c_i = \overline{\varepsilon_p}|_{x_1} \vee \overline{\mu_p}|_{U_1} \vee b_q$$

where $\mathcal{E}(L_p[\alpha_p]) = b_1, \ldots, b_{s_p}$. By the induction hypothesis,

$$\left(\varphi \wedge \bigwedge_{k=1}^{p-1} \overline{\varepsilon_k}|_{X_1}\right)[\alpha_p] \wedge \bigwedge_{k=1}^{q-1} \mathrm{red}(b_k) \models b_q,$$

from which it follows that

$$\varphi \wedge \bigwedge_{k=1}^{p-1} \overline{\varepsilon_k}|_{X_1} \wedge \bigwedge_{k=1}^{q-1} \mathrm{red}(\overline{\varepsilon_p}|_{X_1} \vee \overline{\mu_p}|_{U_1} \vee b_k) \tag{1}$$

entails $\overline{\varepsilon_p}|_{X_1} \vee \overline{\mu_p}|_{U_1} \vee b_q = c_i$.

We show that each conjunct in Equation 1 besides φ is $\mathrm{red}(c)$ for some c appearing in $\mathcal{E}(L)$ before c_i.

For each $k \in [q-1]$, the clause $\overline{\varepsilon_p}|_{X_1} \vee \overline{\mu_p}|_{U_1} \vee b_k$ appears in $\mathcal{E}(L)$ before c_i by definition. For each $k \in [p-1]$,

$$\overline{\varepsilon_k}|_{X_1} = \mathrm{red}(\overline{\varepsilon_k}|_{X_1} \vee \overline{\mu_k}|_{U_1} \vee f_k),$$

where f_k is the final line of $\mathcal{E}(L_k[\alpha_k])$, which is fully universal by the induction hypothesis, and the line $\overline{\varepsilon_k}|_{X_1} \vee \overline{\mu_k}|_{U_1} \vee f_k$ appears in L before c_i. Since $\varepsilon_s = \top, c_r = \bot \vee \overline{\mu_s}|_{U_1} \vee f_s$ is fully universal. This completes the induction step.

With this, we have proven the theorem. We have shown that if $\mathcal{E}(L) = c_1, \ldots, c_r$, the sequence π consisting of the lines of Φ followed by

$$c_1, \mathrm{red}(c_1), \ldots, c_r, \mathrm{red}(c_r)$$

is a $\mathcal{Q}$-P refutation of Φ with size $\mathcal{O}(|L|^d)$.

Now, we just have the proof the claim from above, i.e. that $L_j[\alpha_j]$ is a P-UDL for

$$Q_j = P[\alpha_j] \cdot \left(\varphi \wedge \bigwedge_{k=1}^{j-1} \overline{\varepsilon_k}|_{X_1}\right).$$

Fixing $j \in [s]$, we show that $L_j[\alpha_j]$ is a correct P-UDL by checking all conditions in Definition 2.

- Since L is a P-UDL and the first two conditions are purely syntactic, every restriction of L also fulfils these criteria.

- Let τ, ρ be assignments to the existential variables in Q_j. Suppose τ and φ agree on the first r existential blocks of Q_j for some $r \in [d-1]$. Since τ and φ agree on X_1 in particular, if either of them satisfies $\bigwedge_{k=1}^{j-1} \overline{\varepsilon_k}|_{X_1}[\alpha_j]$, then we have $L_j[\alpha_j](\tau) = L_j[\alpha_j](\rho)$ satisfying the condition trivially, so we assume otherwise. Notice that $L_j[\alpha_j](\tau) = L(\varepsilon_j|_{X_1} \wedge \tau)$ with the U_1 components removed, and likewise for ρ. Since $\varepsilon_j|_{X_1} \wedge \tau$ and $\varepsilon_j|_{X_1} \wedge \rho$ agree on the first $r+1$ existential blocks of Q, $L(\varepsilon_j|_{X_1} \wedge \tau)$ and $L(\varepsilon_j|_{X_1} \wedge \tau)$ agree on the first $r+1$ universal blocks of Q, thus $L_j[\alpha_j](\tau)$ and $L_j[\alpha_j](\rho)$ agree on the first r universal blocks of Q_j.
- Let τ be an assignment to the existential variables of Q_j, and let

  $$\sigma = \varepsilon_j \wedge \tau|_{\mathrm{vars}(\tau)\setminus\mathrm{vars}(\varepsilon_j)}.$$

  If τ falsifies $\bigwedge_{k=1}^{j-1} \overline{\varepsilon_k}|_{X_1}[\alpha_j]$, then $\tau \wedge L_j[\alpha_j](\tau)$ already falsifies the matrix of Q_j, so we assume otherwise. Then $L(\sigma) = \mu_j$, and since $\varepsilon_j|_{X_1} \wedge \tau$ agrees with σ on X_1, $L(\varepsilon_j|_{X_1} \wedge \tau)$ agrees with μ_j on U_1. It follows that

  $$L(\varepsilon_j|_{X_1} \wedge \tau) = \mu_j|_{U_1} \wedge L_j[\alpha_j](\tau),$$

  whereby $\alpha_j \wedge \tau \wedge L_j[\alpha_j](\tau)$ falsifies F, by definition of countermodel. Hence, $\tau \wedge L_j[\alpha_j](\tau)$ falsifies $F[\alpha_j]$, and therefore falsifies the matrix of Q_j.

◀

This reproves a characterisation of QBF resolution by decision lists from [11].

▶ **Corollary 11** ([11]). *Let $P =$ Res with clauses as lines. For a particular family of bounded-depth QBFs, the minimal size of* Res-UDL *proofs and the minimal* Res-UDL *sizes are polynomially equivalent.*

The main application for us is polynomial calculus (PC) for which this result is new. PC has a capacity of $\sqrt{n}$, where n is the size of the proof [10]. Additionally, PC is closed under disjunctions as the disjunction of two lines can be expressed as the product of the respective polynomials. The size of the disjunction is the product of the sizes of the original lines.

▶ **Corollary 12.** *For a particular family of bounded-depth QBFs, the minimal size of $\mathcal{Q}$-PC proofs and the minimal PDL sizes are polynomially equivalent.*

4 Size-degree bounds for polynomial calculus in QBF

Using the connection between $\mathcal{Q}$-PC and PDLs from Corollary 12, we now aim to show lower bounds for $\mathcal{Q}$-PC by proving lower bounds for PDLs. The latter task will be simplified by a relation between PDL size and PDL degree, which is measured as the maximal degree of the polynomial conditions in the PDL. As these polynomials are just defined in existential variables, it makes sense to define the *existential degree* for PDLs and $\mathcal{Q}$-PC refutations. This is in line with an analogous definition of existential width for Q-Resolution [11, 15].

▶ **Definition 13** (Existential degree of a $\mathcal{Q}$-PC refutation). *Let $f = \exists X_1 \forall U_1 \cdots \exists X_{d+1}.\varphi$, $\pi = (\pi_1, \ldots, \pi_s)$ be a $\mathcal{Q}$-PC refutation of f and $X = \bigcup_{i=1}^{d} X_i$. The* existential degree $\deg_\exists$ *of f is defined as* $\deg_\exists(\pi) = \max_{i\in[s]} \deg(\pi_i|_X)$.

Analogously, the degree of a PDL L is defined as the maximal degree of all queries in L. We can show a size-degree relation for PDLs.

▶ **Theorem 14.** *Let f be a multi-output Boolean function with n input variables. If f is computed by a PDL of size s, it is also computed by a PDL of degree $O(\sqrt{n \log s})$.*

In essence, the proof of this theorem is the same as the original size-degree result for propositional polynomial calculus by Impagliazzo, Pudlák and Sgall [39] (cf. also [7, 23] for similar arguments). Equivalently, a function f that can only be computed by PDLs of degree at least k needs PDLs of size $\exp(\Omega(\frac{k^2}{n}))$.

For the proof of this theorem we require the following lemma.

▶ **Lemma 15.** *$\forall d \geqslant 0, n \geqslant 0, b \geqslant 0$ if a PDL L on n input variables has fewer than $a(n,d)^b$ monomials of degree greater than d, then it can be transformed into an equivalent PDL M of degree at most $d + b$, where $a(n,d) = \left(1 - \frac{d}{2n}\right)^{-1}$.*

Proof. We prove the statement for every d by double induction over b and n. Let $d \geqslant 0$ be fixed. For a PDL L, Let L^*_d denote the set of *fat* monomials of L, i.e. those of degree greater than d.

The base case $b = 0$ is trivial, as the condition $|L^*_d| < a(n,d)^b = 1$ ensures that L already has degree at most $d + b = d$. Similarly, the base case $n \leqslant d$ is also trivial, as the degree of a Boolean polynomial cannot be larger then the number of variables.

For the inductive step, consider a PDL L for which $|L^*_d| < a(n,d)^b$. Since the number of occurrences of literals in the monomials of L^*_d is greater than $d|L^*_d|$ and there are $2n$ literals, by the pigeonhole principle, there is a literal c that occurs in more than $\frac{d|L^*_d|}{2n}$ monomials of L^*_d. Therefore, the list $L|_{\overline{c}}$ has fewer than

$$|L^*_d| - \frac{d|L^*_d|}{2n} = \frac{|L^*_d|}{a(n,d)^b}$$

fat monomials. In other words,

$$|(L|_{\overline{c}})^*_d| < \frac{|L^*_d|}{a(n,d)} < a(n,d)^{b-1} < a(n-1,d)^{b-1}.$$

Thus, by induction hypothesis, $L|_{\overline{c}}$ can be transformed into an equivalent PDL L_1 of degree at most $d + b + 1$. However, the list $L|_c$ has $n - 1$ variables, and it holds that

$$|(L|_c)^*_d| < |L^*_d| < a(n,d)^b < a(n-1,d)^b,$$

so by induction hypothesis it can also be transformed into an equivalent list L_2 of degree at most $d + b$.

Now consider the list M that consists of $\overline{c} \otimes L_1$ followed by L_2. Because L_1 is equivalent to $L|_{\overline{c}}$ and L_2 is equivalent to $L|_c$, M is equivalent to L. The degree of M is at most $d+b$. ◀

Proof of Theorem 14. We will apply Lemma 15 with $d = b = \left\lceil \sqrt{2n \ln s} \right\rceil$. Let $a = a(n,d)$. We will show that $s < a^b$, which will allow us to use Lemma 15 to obtain the statement. By $\ln(1 + x) \leqslant x$, we have

$$\ln a = -\ln(1 - \frac{d}{2n}) \geqslant \frac{d}{2n}$$

and hence

$$\log_a s = \frac{\ln s}{\ln a} \leqslant \frac{2n \ln s}{d} < \sqrt{2n \ln s} < b.$$

◀

We can use this size-degree relation for PDLs to obtain a size-degree relation on $\mathcal{Q}$-PC.

▶ **Theorem 16.** *Let f be a QBF with n variables of alteration depth d such that every $\mathcal{Q}$-PC proof has existential degree at least k. Then every $\mathcal{Q}$-PC proof has size at least* $\exp(\Omega(\frac{k^2}{d^3 n}))$.

Proof. Let $f = \exists X_1 \forall U_1 \cdots \exists X_{d+1}.\varphi$, $X = \bigcup_{i=1}^{d} X_i$ the set of existential variables except the last block and $|X| = v$. Additionally, let π be a shortest $\mathcal{Q}$-PC refutation of f. Using Corollary 12, π can be transformed into a PDL L of size at most $|\pi|^d$. With Theorem 14, L can then be transformed into a PDL M with degree at most $\mathcal{O}(\sqrt{dv \log|\pi|})$. Transforming M back into a $\mathcal{Q}$-PC proof with Corollary 12 results in a proof π' with degree at most $k = \mathcal{O}(d\sqrt{dv \log|\pi'|})$. Solving this equation for $|\pi|$ yields the result. ◀

The proof is similar to Theorem 6.2 in [11]. In contrast to the size-degree relation from [39], Theorem 16 includes the quantifier depth of the QBF, but not the initial degree of the QBF.

In the rest of this section, we will explore specific lower bounds for the degree of PDLs. We will first show degree lower bounds for PDLs computing specific functions and then turn this into $\mathcal{Q}$-PC size lower bounds for related QBFs.

4.1 Parity and mod functions require PDLs of high degree

Let $\mathrm{par}_n(x_1, \ldots, x_n) = \bigoplus_{i=1}^{n} x_n$ be the parity function. The lower bounds for this function only hold for PDLs over certain fields; in fact, par_n is just the sum of input variables in fields of characteristic 2, and is therefore trivial for PDLs over those fields. However, it seems to be hard in fields of characteristic 0. We prove a lower bound for $\mathbb{C}$ and its subfields.

▶ **Proposition 17.** *A PDL with polynomials over a subfield of $\mathbb{C}$ computing par_n has degree at least $\frac{n}{2}$.*

Proof. We consider the first line of the PDL and assume without loss of generality that it has output 1.[1]

Let p be the first line's polynomial and d its degree. Let $X = \{x_1, \ldots, x_n\}$. We can assume that there is a $w \in \langle X \rangle$ with $p(w) \neq 0$ or we could omit the first line. However, to avoid giving any wrong answers, for every $a \in \langle X \rangle$ with $\mathrm{par}_n(a) = 0$, it must hold that $p(a) = 0$.

We compute the complex conjugate p^* by conjugating every coefficient. Because we only evaluate the polynomials on real numbers (specifically 0 and 1), we know that $p^*(a) = (p(a))^*$ for every $a \in \langle X \rangle$. We define $q := p \cdot p^*$ and note that $\deg(q) \leqslant 2d$ and for all $a \in \langle X \rangle$, $q(a) = p(a) \cdot p(a)^* \in \mathbb{R}^{\geqslant 0}$.

For a polynomial r, define the function

$$s(r) := \sum_{\substack{a \in \langle X \rangle \\ \mathrm{par}(a)=1}} r(a) - \sum_{\substack{a \in \langle X \rangle \\ \mathrm{par}(a)=0}} r(a)$$

[1] If it has output 0, we can invert all outputs in the PDL and replace every occurence of x_1 with $1 - x_1$. This does not change its degree, and the resulting PDL computes $\neg\mathrm{par}_n(\overline{x_1}, x_2, \ldots, x_n) = \mathrm{par}_n(x_1, \ldots, x_n)$.

which is linear with respect to r. If r is a monomial that does not contain the variable x,

$$\begin{aligned} s(r) &= \sum_{\substack{a\in\langle X\setminus\{x\}\rangle \\ b\in\langle\{x\}\rangle \\ \mathrm{par}(a,b)=1}} r(a,b) - \sum_{\substack{a\in\langle X\setminus\{x\}\rangle \\ b\in\langle\{x\}\rangle \\ \mathrm{par}(a,b)=0}} r(a,b) = \sum_{\substack{a\in\langle X\setminus\{x\}\rangle \\ b\in\langle\{x\}\rangle \\ \mathrm{par}(a,b)=1}} r(a) - \sum_{\substack{a\in\langle X\setminus\{x\}\rangle \\ b\in\langle\{x\}\rangle \\ \mathrm{par}(a,b)=0}} r(a) \\ &= \left(\sum_{\substack{a\in\langle X\setminus\{x\}\rangle \\ \mathrm{par}(a)=1}} r(a) + \sum_{\substack{a\in\langle X\setminus\{x\}\rangle \\ \mathrm{par}(a)=0}} r(a) \right) - \left(\sum_{\substack{a\in\langle X\setminus\{x\}\rangle \\ \mathrm{par}(a)=0}} r(a) + \sum_{\substack{a\in\langle X\setminus\{x\}\rangle \\ \mathrm{par}(a)=1}} r(a) \right) \\ &= 0. \end{aligned}$$

Because s is linear, $s(r) = 0$ also holds for any polynomial r of degree $\leqslant n$.

To obtain a value of $s(q)$, we use that

$$\sum_{\substack{a\in\langle X\rangle \\ \mathrm{par}(a)=1}} q(a) > 0 \quad \text{and} \quad \sum_{\substack{a\in\langle X\rangle \\ \mathrm{par}(a)=0}} q(a) = 0$$

(in the left equation, none of the summands are negative; one of them is $q(w) = |p(w)|^2 > 0$). Consequently, $s(q) > 0$ and $\deg(q) \geqslant n$, so using $\deg(q) \leqslant 2d$ we conclude that $d \geqslant \frac{n}{2}$. ◀

To turn this lower bound into a $\mathcal{Q}$-PC bound we need QBFs based on the parity function. For this we describe a general transformation originating from [13,14] to construct QBFs that are false, but force the universal player to use a unique strategy by computing a particular function f. We define the QBF $\mathcal{Q}$-f:

▶ **Definition 18** ($\mathcal{Q}$-f). *Let $f : \langle X\rangle \to \langle U\rangle$ be a function that is computed by a* P/poly *circuit C. Then $\mathcal{Q}$-$f := \exists X \forall U \exists T\, \varphi$ where φ is the Tseitin transformation of the circuit $U \neq C(X)$, and T is the corresponding set of auxiliary Tseitin variables.*

In our case above, f is par_n and C is a simple P/poly circuit for par_n. The existential player wins if and only if the circuit $U \neq C(X)$ yields true, after the assignments to X and U were chosen by the respective players. The universal player therefore has the unique strategy of playing $U = f(X)$.

From Proposition 17 together with the size-degree relation for PDLs (Theorem 14) and the efficient transformation into $\mathcal{Q}$-PC (Theorem 10) we obtain:

▶ **Corollary 19.** *$\mathcal{Q}$-PC refutations over a subfield of $\mathbb{C}$ of $\mathcal{Q}$-par_n have size* $\exp(\Omega(n))$.

We can generalise this to the modulo k functions. Let

$$\mathrm{mod}_n^k(x_1,\ldots,x_n) = \begin{cases} 1 & \text{if } \sum_{i=1}^n x_i \equiv 0 \mod k \\ 0 & \text{otherwise.} \end{cases}$$

With a similar, but somewhat more technical proof than for Proposition 17 we can show:

▶ **Proposition 20.** *A PDL with polynomials over a subfield of $\mathbb{C}$ computing mod_n^k has degree at least $\frac{1}{2}\left\lfloor \frac{n}{(k-1)} \right\rfloor$.*

Proof. Let $X = \{x_1,\ldots,x_n\}$. We assume without loss of generality that $(k-1) \mid n$: otherwise, fix an appropriate number of inputs to be 0 to obtain a PDL with fewer variables where that is the case, without an increase in degree and with no change to $\left\lfloor \frac{n}{k-1} \right\rfloor$.

We consider the first line of the PDL and use it to obtain a polynomial p with the property: if $\sum_{i=1}^{n} x_i \equiv 0 \mod k$, then $p(x_1, \ldots, x_n) = 0$. If the first line has output 0, let p be its polynomial, which has the property due to correctness of the PDL. If the first line has output 1, let g be its polynomial and $p(x_1, x_2 \ldots, x_n) = g(\overline{x_1}, x_2, \ldots, x_n)$. We check the property: suppose $\sum_{i=1}^{n} x_i \equiv 0 \mod k$. Then $\overline{x_1} + \sum_{i=2}^{n} x_i \not\equiv 0 \mod k$, so the output 1 is wrong for the input $\overline{x_1}, x_2, \ldots, x_n$. Because the PDL is correct, $g(\overline{x_1}, x_2, \ldots, x_n) = 0 = p(x_1, x_2 \ldots, x_n)$, so the property holds. Let $d = \deg(p)$ be the degree of the first line.

We can assume there is a $w \in \langle X \rangle$ with $p(w) \neq 0$, otherwise the first line's polynomial would always be 0 and we could omit the first line. However, to avoid giving any wrong answers, for every $a \in \langle X \rangle$ with $\mathrm{mod}_n^k(a) = 1$, it must hold that $p(a) = 0$.

We compute the complex conjugate p^* by conjugating every coefficient. Because we only evaluate the polynomials on real numbers (specifically 0 and 1), we know that $p^*(a) = (p(a))^*$ for every $a \in \langle X \rangle$. We define $q := p \cdot p^*$ and note that $\deg(q) \leqslant 2d$ and for all $a \in \langle X \rangle$, $q(a) = p(a) \cdot p(a)^* \in \mathbb{R}^{\geqslant 0}$.

We partition the variables in X into buckets $B_1, \ldots, B_{n/(k-1)}$ of size $k-1$ and fix an ordering within every bucket. Let $\omega_i : \langle B_i \rangle \to \{0, 1\}$ be a function with $\omega_i(\beta) = 1$ if and only if, according to the ordering, β contains all false variables before all true variables. Note that there are exactly k assignments to B_i where $\omega_i = 1$. Let $\omega : \langle X \rangle \to \{0, 1\}, \omega = \Pi_{i=1}^{n/(k-1)} \omega_i$. We assume without loss of generality that $\omega(w) = 1$, if this is not the case, we reorder the variables in each bucket to make it so.

For an assignment a, let $\sigma(a)$ be its number of true variables. Let $\rho(a) = -1$ if $a \equiv 0 \mod k$ and $\rho(a) = \frac{1}{k-1}$ otherwise, and note that $\sum_{i=0}^{k-1} \rho(i) = 0$. For a polynomial r, define the function

$$s(r) := \sum_{a \in \langle X \rangle} \rho(\sigma(a)) \omega(a) r(a)$$

which is linear with respect to r.

If r is a monomial which contains no variable from the bucket B_j,

$$\begin{aligned}
s(r) &= \sum_{a\in\langle X\rangle} \rho(\sigma(a))\omega(a)r(a) \\
&= \sum_{i=0}^{k-1} \rho(i) \sum_{\substack{a\in\langle X\rangle \\ \sigma(a)\equiv i \mod k}} \omega(a)r(a) \\
&= \sum_{i=0}^{k-1} \rho(i) \sum_{b\in\langle B_j\rangle} \sum_{\substack{a\in\langle X\setminus B_j\rangle \\ \sigma(a)+\sigma(b)\equiv i \mod k}} \omega(a,b)r(a,b) \\
&= \sum_{i=0}^{k-1} \rho(i) \sum_{b\in\langle B_j\rangle} \omega_j(b) \sum_{\substack{a\in\langle X\setminus B_j\rangle \\ \sigma(a)+\sigma(b)\equiv i \mod k}} \omega(a)r(a) \\
&= \sum_{i=0}^{k-1} \rho(i) \sum_{c_1=0}^{k-1} \sum_{\substack{a\in\langle X\setminus B_j\rangle \\ \sigma(a)+c_1\equiv i \mod k}} \omega(a)r(a) \\
&= \sum_{i=0}^{k-1} \rho(i) \sum_{c_2=i-k+1}^{i} \sum_{\substack{a\in\langle X\setminus B_j\rangle \\ \sigma(a)\equiv c_2 \mod k}} \omega(a)r(a) \\
&= \left(\sum_{i=0}^{k-1} \rho(i)\right) \cdot \sum_{c_2=0}^{k-1} \sum_{\substack{a\in\langle X\setminus B_j\rangle \\ \sigma(a)\equiv c_2 \mod k}} \omega(a)r(a) = 0.
\end{aligned}$$

In the fifth equality, we replaced $\sigma(b)$ with c_1, using the fact that for every sum c, there is exactly one $b \in \langle B_j\rangle$ with $\sigma(b) = c_1$ and $\omega_j(b) = 1$. In the sixth equality, we replaced $i - c_1$ by c_2.

We conclude that any monomial r for which $s(r) \neq 0$ must include at least one variable from each bucket, so its degree is at least $\frac{n}{k-1}$. Because s is linear in its argument, this fact extends from monomials to polynomials.

To obtain the value of $s(q)$, we use that $s(q) = \sum_{a\in\langle X\rangle} \rho(\sigma(a))\omega(a)q(a)$ and observe that all summands are nonnegative real numbers: if $\sigma(a) \equiv 0 \mod k$ then $\mathrm{mod}_n^k(a) = 1$, so $q(a) = p(a)\cdot p(a)^* = 0$. Otherwise, $\rho(\sigma(a)) = \frac{1}{k-1}, \omega(a) \in \{0,1\}, q(a) \in \mathbb{R}^{\geqslant 0}$, so their product is a nonnegative real number. One of the summands is $\rho(\sigma(w))\omega(w)q(w) = \frac{1}{k-1}\cdot 1 \cdot q(w) > 0$, so the sum cannot be zero. Since $s(q) > 0$, we know that $\deg(q) \geqslant \frac{n}{k-1}$, so using $\deg(q) \leqslant 2d$ we conclude that $d \geqslant \frac{1}{2}\cdot\frac{n}{k-1}$. ◀

Consequently, the $\mathcal{Q}$-mod_n^k QBFs are hard for $\mathcal{Q}$-PC over $\mathbb{C}$.

▶ **Corollary 21.** *$\mathcal{Q}$-PC refutations over a subfield of $\mathbb{C}$ of $\mathcal{Q}$-mod_n^k have size $\exp(\Omega(\frac{n}{k}))$.*

4.2 Majority PDLs have high degree

Next we want to show degree lower bounds for PDLs that compute the majority function, which is defined as

$$\mathrm{maj}_n(x_1,\ldots,x_n) = \begin{cases} 1 & \text{if } \sum_{i=1}^n x_i \geqslant \frac{n}{2} \\ 0 & \text{otherwise.} \end{cases}$$

In contrast to the par_n and mod_n^k functions, this lower bound does not depend on the underlying field. We start by proving a useful lemma about PDLs:

▶ **Lemma 22.** *Let $X = \{x_1, \ldots, x_n\}$, $\alpha \in \langle X \rangle$ a complete assignment and p a polynomial with variables in X that is not the constant 0. There is an assignment $\beta \in \langle X \rangle$ such that α and β only differ in at most $\deg(p)$ variables, and $p(\beta) \neq 0$.*

Proof. We start by taking p and constructing an equivalent polynomial q that contains no variables according their polarity in α: when $\alpha(x) = 0$, we replace $\overline{x}$ by $1 - x$, and when $\alpha(x) = 1$, we replace x by $1 - \overline{x}$. Let m be one of the lowest-degree monomials in q, and β the assignment that satisfies every literal in m, and assigns every other variable according to α. Note that α and β only differ in at most $\deg(m) \leqslant \deg(q) = \deg(p)$ variables. Because m has minimal degree, every other monomial in q contains a variable $v \notin \mathrm{vars}(m)$, so $\beta(v) = \alpha(v)$. Because v only occurs in the polarity opposite of $\alpha(v)$, β does not satisfy that other monomial. Therefore, $0 \neq m(B) = q(B) = p(B)$. ◀

With this, we can show the lower bound fairly easily:

▶ **Proposition 23.** *A PDL computing $\mathrm{maj}_n(x_1, \ldots, x_n)$ has degree at least $\frac{n}{2}$.*

Proof. Let L be such a PDL, d its degree, and p its first polynomial. We also define $X := \{x_1, \ldots, x_n\}$. If the first line of L has output 1, let A be the assignment that assigns all $x_i = 0$, and apply Lemma 22. Because $p(\beta) \neq 0$ and L is correct, we obtain $\mathrm{maj}_n(\beta) = 1$. So β has to assign 1 to at least $\frac{n}{2}$ variables and thus differs from A in at least $\frac{n}{2}$ variables. Therefore, $\frac{n}{2} \leqslant \deg(p) \leqslant d$. If the first line has output 0 instead, let A be the assignment that assigns all $x_i = 1$. Lemma 22 yields an assignment β with $p(\beta) \neq 0$, so $\mathrm{maj}_n(\beta) = 0$ and β has to assign 0 to at least $\frac{n}{2}$ variables. Therefore, $\frac{n}{2} \leqslant \deg(p) \leqslant d$. ◀

▶ **Corollary 24.** *$\mathcal{Q}$-PC refutations of the majority formulas have size $\exp(\Omega(n))$.*

4.3 Limits of the size-degree method

While the size-degree technique is useful for showing several lower bounds as demonstrated above, it does not capture all nuances of PDL size. We will illustrate this by constructing two functions, each having n^2 variables and requiring PDLs of degree n. For these values, Theorem 14 does not yield any useful lower bound. Indeed, one of the functions requires PDLs of size $\exp(\Omega(n))$, while for the other one, size $O(n)$ is sufficient.

▶ **Theorem 25.** *Given a function f, the minimal size of its PDLs is not solely determined by its number of variables and minimal PDL degree. In particular, there are families of functions f_n and g_n, each with n^2 input variables and minimal PDL degree n, such that f_n has PDLs of size $O(n)$ and g_n requires PDLs of size $\exp(\Omega(n))$.*

In order to prove this, we introduce examples for those functions g and f. We define the square-majority function as

$$\mathrm{sqm}_n(x_1, \ldots, x_{n^2}) = \begin{cases} 1 & \text{if } \sum_{i=1}^{n^2} x_i \geqslant n \\ 0 & \text{otherwise.} \end{cases}$$

▶ **Lemma 26.** *The sqm_n function can be computed by PDLs of degree n, but not by PDLs of smaller degree. It requires PDLs of size $\exp(\Omega(n))$.*

Proof. The sqm_n function can be computed by a PDL of degree n in the following way: for every subset $S \subset \{x_1, \ldots, x_{n^2}\}, |S| = n$, have a line with polynomial $\prod_{s \in S} s$ and output 1. At the end, there is a line with polynomial 1 and output 0. This is correct: if an assignment contains at least n true variables, there will be some S consisting of only true variables. The corresponding polynomial will be 1 and the PDL will output 1. Otherwise, each S will contain at least one false variable, so the corresponding polynomials will be 0 and the PDL will have output 0.

To show the lower bounds on degree and size, we assume that there is a PDL L with size r and degree k computing $\mathrm{sqm}(x_1, \ldots, x_{n^2})$. We can modify L by setting $x_{2n+1} = x_{2n+2} = \cdots = x_{n^2} = 0$ to obtain a PDL L' with size $\leqslant r$ and degree $\leqslant k$ that computes the function $\mathrm{sqm}_n(x_1, \ldots, x_{2n}, 0, \ldots, 0) = \mathrm{maj}_{2n}(x_1, \ldots, x_{2n})$. According to Proposition 23 it has degree at least n, so $k \geqslant n$. According to Theorem 14, it has size $\exp(\Omega(2n))$, so $r \in \exp(\Omega(n))$. ◀

The lower bound on the size of PDLs for sqm already cannot be shown by Theorem 14. This raises the question of whether our size-degree relation is simply too weak, and whether we could obtain a stronger one that can capture the complexity of the sqm function. It turns out that this is not the case: there are functions with the same degree and number of variables, but smaller decision lists. The complexity of the sqm function cannot be derived from its degree and number of variables alone.

To obtain one such function, let $X = \{x_i^j \mid i, j \in \{1, \ldots, n\}\}$ and define

$$f_n(X) := \bigvee_{i=1}^{n} \bigwedge_{j=1}^{n} x_i^j.$$

▶ **Lemma 27.** *The function f_n defined above can be computed by PDLs of degree n and size $O(n)$, but not by PDLs of smaller degree.*

Proof. This can be solved with the following decision list: for every $i = 1, \ldots, n$, have a line that says: if $\prod_{j=1}^{n} x_i^j \neq 0$, then the output is 1. The last line says that if $1 \neq 0$, then the output is 0. This decision list has degree n and size $n + 1$; its correctness is obvious. We will go on to show that a degree of n is actually necessary.

Let L be a PDL computing f_n. We distinguish two cases based on the output of the first line of L. They are very similar, but not quite symmetric.

In the first case, the first line in L has polynomial p and output 1. Let A be the assignment that assigns every variable to 0. According to Lemma 22, there is an assignment B that assigns at most $\deg(p)$ variables to 1, with $p(B) \neq 0$. This means that $f(B) = 1$, so at least n variables in B have to be set to 1 (specifically, all the x_i^j for some particular i). Accordingly, $n \leqslant \deg(p) \leqslant \deg(L)$.

In the second case, the first line in L has polynomial p and output 0. Let A be the assignment that assigns every variable to 1. According to Lemma 22, there is an assignment B that assigns at most $\deg(p)$ variables to 0, with $p(B) \neq 0$. This means that $f(B) = 0$, so at least n variables in B have to be set to 0 (specifically, at least one x_i^j for every i). Accordingly, $n \leqslant \deg(p) \leqslant \deg(L)$. ◀

Proof of Theorem 25. Let f_n be the function defined above, and $g_n = \mathrm{sqm}_n$. The statement follows directly from Lemmas 26 and 27. ◀

5 Converting PDLs into Boolean circuits

We will now obtain a different lower bound method for $\mathcal{Q}$-PC proof size that directly imports circuit-size lower bounds. For this we show a connection between PDLs over finite fields and

$AC^0[p]$ circuits. To utilize this, we start by reducing PDLs over a finite field to PDLs over a prime-order finite field.

▶ **Lemma 28.** *Let p be a prime, $k, l, m \in \mathbb{N}^{\geqslant 1}$ and L a PDL of length l and size m over the finite field $\mathbb{F}_{p^k}$. Then L can be converted into a PDL of length $k \cdot l$ and size $k \cdot m$ over the finite field $\mathbb{F}_p$ that computes the same function.*

Proof. When viewing the field $\mathbb{F}_{p^k}$ additively, it is isomorphic to the group $(C_p)^k$, where C_p is the cyclic group with p elements. Take a polynomial equation

$$0 = \sum_{i=1}^{n} a_i m_i \Leftrightarrow 0 = \sum_{\substack{i=1 \\ m_i \neq 0}}^{n} a_i$$

where $a_i \in \mathbb{F}_{p^k}$ are coefficients and m_i are unit monomials. This uses the fact that the PDL is evaluated on boolean assignments, so the m_i can only become 1 or 0.

This equation no longer uses multiplication in $\mathbb{F}_{p^k}$, so we can use the mentioned isomorphism to convert each coefficient a_i to $(c_{i,1}, c_{i,2}, \ldots, c_{i,k}) \in (C_p)^k$. The polynomial equation becomes

$$(0, 0, \ldots, 0) = \sum_{\substack{i=1 \\ m_i \neq 0}}^{n} (c_{i,1}, c_{i,2}, \ldots, c_{i,k})$$

which is equivalent to

$$\bigwedge_{j=1}^{k} \left(0 = \sum_{\substack{i=1 \\ m_i \neq 0}}^{n} c_{i,j} \right).$$

Using the fact that C_p is isomorphic to $\mathbb{F}_p$ regarding addition, we can convert each coefficient $c_{i,j}$ to $a_{i,j} \in \mathbb{F}_p$, and the polynomial equation becomes

$$\bigwedge_{j=1}^{k} \left(0 = \sum_{\substack{i=1 \\ m_i \neq 0}}^{n} a_{i,j} \right) \Leftrightarrow \bigwedge_{j=1}^{k} \left(0 = \sum_{i=1}^{n} a_{i,j} m_i \right).$$

Using this procedure, we can convert an arbitrary polynomial equation in $\mathbb{F}_{p^k}$ into the conjunction of k polynomial equations in $\mathbb{F}_p$, each of shorter or equal size. To convert a PDL, we replace each line with a sequence of l lines, whose polynomial equations are obtained by the construction above, and whose output is the output of the original line. ◀

We will now efficiently convert PDLs over $\mathbb{F}_p$ for primes p into $AC^0[p]$ circuits.

▶ **Proposition 29.** *Let $f : \{0,1\}^n \to \{0,1\}^s$ be a function that is computed by a PDL of size m over a finite field $\mathbb{F}_p$. f can be computed by an $\mathsf{AC}^0[p]$ circuit of depth 6 with only $O(pm + s)$ AND or OR gates.*

Proof. We construct the circuit iteratively starting at the inputs and ending at the output. At each layer, we describe the semantics of the newly-added circuit gates.

- Start with v input gates for the variables.
- Add v NOT gates, each negating one of the variables. All literals are now represented in the circuit.

- Consider each monomial $c \cdot \prod_i t_i$ where $c \in \mathbb{F}_p$ and the t_i are literals. Add c identical AND gates with all the t_i as inputs. The sum of the outputs of these gates will be equivalent to the value of the monomial. Because each $c < p$, the total number of these gates is smaller than pm.
- For each line, add a MOD_p gate over all the AND gates of its monomials. The sum of its inputs will be equivalent to the sum of the values of its monomials, so its output indicates whether the polynomial equation holds in $\mathbb{F}_p$.
- For each line, add a NOT gate negating the MOD_p gate.
- For each line, add an AND gate that checks if all polynomial equations of previous line hold, but the equation of the current line does not. Its output indicates whether this line's output value is active.
- Finally, for each output variable z add an OR gate connecting the AND gates of those lines whose output sets $z = 1$. Its output indicates whether the PDL sets $z = 1$, it is the output of the circuit for variable z.

This circuit has depth 6 and at most $O(pm + s)$ AND or OR gates. ◀

Using Lemma 28 we can extend this result to fields $\mathbb{F}_{p^k}$:

▶ **Corollary 30.** *Let $f : \{0,1\}^n \to \{0,1\}^s$ be a function that is computed by a PDL of size m over a finite field $\mathbb{F}_{p^k}$. Then f can be computed by an $\mathsf{AC}^0[p]$ circuit of depth 6 with only $O(pmk + s)$ AND or OR gates.*

This corollary allows us to lift lower bounds for $\mathsf{AC}^0[p]$ circuits to PDLs over finite fields. We cite such a circuit lower bound:

▶ **Theorem 31** (Smolensky [51])**.** *Let p be a prime and r not a power of p. An $\mathsf{AC}^0[p]$ circuit of depth k that computes mod_n^r requires $\exp(\Omega(n^{\frac{1}{2k}}))$ AND or OR gates.*

Using Corollary 30 we can apply this result to PDLs:

▶ **Corollary 32.** *Let p, r be distinct primes. A PDL over the finite field $\mathbb{F}_{p^k}$ that computes mod_n^r needs size $\exp(\Omega(\sqrt[12]{n} - \log(pk)))$.*

We also want to apply these results to another class of functions that we call balance:

$$\mathrm{balance}_{2n}(x_1, x_2, \ldots, x_{2n}) = \begin{cases} 1 \text{ if } \sum x_i = n \\ 0 \text{ if } \sum x_i \neq n. \end{cases}$$

We now show that PDLs for the balance function can be transformed into $AC^0[p]$ circuits for the function mod_n^q with arbitrary q.

▶ **Proposition 33.** *If there is a PDL of size m over $\mathbb{F}_p$ that computes $\mathrm{balance}_{2n}$, then mod_n^q can be computed by an $\mathsf{AC}^0[p]$ circuit of depth 8 that uses only $O(pnm)$ AND or OR gates.*

Proof. Let $R = \{q \cdot i \mid i \in \mathbb{Z}, 0 \leqslant q \cdot i \leqslant n\} = q\mathbb{Z} \cap [0; n]$, and note that $|R| = \left\lceil \frac{n+1}{q} \right\rceil$. When exactly k of the variables $x_1, \ldots, x_n$ are true, then $\mathrm{mod}_n^q(x_1 \ldots, x_n) = 1$ if and only if $k \in R$.

Using Proposition 29 we can obtain a circuit that computes $\mathrm{balance}_{2n}$ with $pm + l$ AND gates and one OR gate. We instantiate this circuit once for each $k \in R$, replacing the $2n$ inputs with n actual inputs $x_1, \ldots, x_n$, as well as k constants 0 and $n - k$ constants 1. The output of such an instance is 1 if and only if k of the actual inputs are 1. We use a single OR gate over all these outputs to obtain $\mathrm{mod}_n^q(x_1 \ldots, x_n)$. The total number of AND or OR gates is $|R|(pm + l) + 1 = \left\lceil \frac{n+1}{q} \right\rceil (pm + l) + 1 \leqslant (n + 1)(p + 1)m + 1$. ◀

▶ **Corollary 34.** *The balance formulas require exponential-sized PDLs over finite fields.*

We can now show that many of the systems $\mathcal{Q}$-PC over different fields are incomparable.

▶ **Theorem 35.** *Let F be a finite field, and G either a finite field with different characteristic, or a subfield of $\mathbb{C}$. Then the $\mathcal{Q}$-PC systems over F and G are incomparable.*

Proof. Because PC systems and PDLs are polynomially equivalent, we can use exponential separations between the respective PDLs to obtain the result. Let p and q be the characteristics of F and G, respectively. The function mod_n^p is trivial for PDLs over F and requires only linear size. However, it requires exponential-sized PDLs over G, either due to Corollary 21 (if G is a finite field) or due to Corollary 32 (if G is a subfield of $\mathbb{C}$).

If G is a finite field, the function mod_n^q similarly requires exponential-size PDLs over F and only linear-size PDLs over G. If G is a subfield of $\mathbb{C}$, the $\mathrm{balance}_n$ functions can be trivially computed in linear size by PDLs over G, but require exponential-size PDLs over F due to Corollary 34. ◀

6 Beyond polynomial calculus – the wider picture

While we concentrated on $\mathcal{Q}$-PC in this paper, it is interesting to explore the wider consequences of our P-UDL characterisation in Theorem 10 for further proof systems $\mathcal{Q}$-P. Besides PC and Res, the most studied base systems are arguably *Cutting Planes* (CP) and the various $\mathscr{C}$-Frege systems, where $\mathscr{C}$ is some circuit class, e.g. $\mathsf{AC}^0, \mathsf{NC}^1$ or $\mathsf{P/poly}$.

While $\mathcal{Q}$-CP has polynomial capacity (the capacity is 1) [10], the lines are not closed under disjunction as we show in the next lemma.

▶ **Lemma 36.** *Inequalities are not closed under disjunction.*

Proof. We use two variables x and y and look at the inequalities $x - y \geqslant 1$ and $-x + y \geqslant 1$. The disjunction of those is true when (x, y) is $(1, 0)$ or $(0, 1)$ and false if it is $(1, 1)$ or $(0, 0)$. Suppose it is expressed by the inequality $ax + by \geqslant c$. Because this is true for $(x, y) = (1, 0)$, we know $a \geqslant c$, and because it is true for $(x, y) = (0, 1)$, we know $b \geqslant c$. But because it is false for $(x, y) = (1, 1)$, we know $a + b < c$, and because it is false for $(x, y) = (0, 0)$, we get $0 < c$. By adding $a + b < c$ and $0 < c$ and subtracting $a \geqslant c$ and $b \geqslant c$, we get $0 < 0$, which is a contradiction. ◀

Hence we cannot use Theorem 10 to obtain a CP-UDL characterisation. We leave open, whether the result itself does not hold or if it only requires a different proof.

For $\mathcal{Q}$-$\mathscr{C}$-Frege, we cannot use Theorem 10 either. Here the lines are closed under disjunction, but the capacity is exponential. However, it is known that the circuit class $\mathscr{C}$ itself, which is a strictly stronger model than $\mathscr{C}$-UDL, tightly characterises $\mathcal{Q}$-$\mathscr{C}$-Frege [20]. As such, the equivalence between $\mathscr{C}$-UDLs and $\mathcal{Q}$-$\mathscr{C}$-Frege fails.

Interestingly, if we consider treelike $\mathcal{Q}$-$\mathscr{C}$-Frege systems, the $\mathscr{C}$-UDL characterisation does hold, even though the capacity is still superpolynomial. Intuitively, this could be explained by the fact that limited capacity is only required for blockwise reductions, and such reductions can (possibly) be obtained by combining reductions along a path in the proof-tree. We prove the result even without using Theorem 10, as a direct proof is straightforward to obtain using a previous characterisation of treelike $\mathcal{Q}$-$\mathscr{C}$-Frege from [16].

▶ **Theorem 37.** *For a circuit class $\mathscr{C}$ and a QBF family, the minimal sizes of $\mathcal{Q}$-$\mathscr{C}$-$\mathsf{Frege}_{\mathrm{tree}}$ proofs and the minimal $\mathscr{C}$-UDL sizes are polynomially bounded by each other.*

Proof. Let Φ be a QBF and L a $\mathscr{C}$-UDL computing a countermodel f. Then $|\mathsf{rng}(f)| \leqslant |L|$. Additionally, a circuit computing only a single universal variable can be extracted from L with size $\mathcal{O}(L)$. [16] show, that every lower bound of $\mathcal{Q}$-$\mathscr{C}$-$\mathsf{Frege}_{\text{tree}}$ is either based on one of these two criteria or propositional hardness.

Let $C_1, \ldots, C_k$ be a set of circuits computing a countermodel of Φ. Let $u_1, \ldots, u_s$ be the range of the countermodel. Construct a $\mathscr{C}$-UDL as follows: Build a line for every u_i, such that the query is true if and only if the output of all circuits correspond to the respective assignment in u_i. The resulting UDL has size $\mathcal{O}(s \cdot k \cdot \sum_j |C_j|)$. From [16] the theorem follows. ◀

Interestingly, this result even holds for QBFs with unbounded quantifier alternation. We leave open whether similar characterisations can be obtained for daglike systems $\mathcal{Q}$-P on formulas with unbounded alternation depth.

References

1 Sanjeev Arora and Boaz Barak. *Computational Complexity – A Modern Approach*. Cambridge University Press, 2009.

2 Albert Atserias, Johannes Klaus Fichte, and Marc Thurley. Clause-learning algorithms with many restarts and bounded-width resolution. *J. Artif. Intell. Res.*, 40:353–373, 2011.

3 Valeriy Balabanov and Jie-Hong R. Jiang. Unified QBF certification and its applications. *Form. Methods Syst. Des.*, 41(1):45–65, 2012.

4 Valeriy Balabanov, Magdalena Widl, and Jie-Hong R. Jiang. QBF resolution systems and their proof complexities. In *Proc. Theory and Applications of Satisfiability Testing (SAT)*, pages 154–169, 2014.

5 Paul Beame, Henry A. Kautz, and Ashish Sabharwal. Towards understanding and harnessing the potential of clause learning. *J. Artif. Intell. Res.*, 22:319–351, 2004.

6 Paul Beame and Toniann Pitassi. Propositional proof complexity: Past, present, and future. In G. Paun, G. Rozenberg, and A. Salomaa, editors, *Current Trends in Theoretical Computer Science: Entering the 21st Century*, pages 42–70. World Scientific Publishing, 2001.

7 Eli Ben-Sasson and Avi Wigderson. Short proofs are narrow - resolution made simple. *J. ACM*, 48(2):149–169, 2001.

8 Olaf Beyersdorff. On the correspondence between arithmetic theories and propositional proof systems – a survey. *Mathematical Logic Quarterly*, 55(2):116–137, 2009.

9 Olaf Beyersdorff. Proof complexity of quantified Boolean logic – a survey. In Marco Benini, Olaf Beyersdorff, Michael Rathjen, and Peter Schuster, editors, *Mathematics for Computation (M4C)*, pages 353–391. World Scientific, Singapore, 2022.

10 Olaf Beyersdorff, Joshua Blinkhorn, and Luke Hinde. Size, cost, and capacity: A semantic technique for hard random QBFs. *Logical Methods in Computer Science*, 15(1), 2019.

11 Olaf Beyersdorff, Joshua Blinkhorn, Meena Mahajan, and Tomás Peitl. Hardness characterisations and size-width lower bounds for QBF resolution. *ACM Trans. Comput. Log.*, 24(2):10:1–10:30, 2023.

12 Olaf Beyersdorff and Benjamin Böhm. Understanding the relative strength of QBF CDCL solvers and QBF resolution. *Log. Methods Comput. Sci.*, 19(2), 2023.

13 Olaf Beyersdorff, Ilario Bonacina, Leroy Chew, and Jan Pich. Frege systems for quantified Boolean logic. *J. ACM*, 67(2):9:1–9:36, 2020.

14 Olaf Beyersdorff, Leroy Chew, and Mikolás Janota. New resolution-based QBF calculi and their proof complexity. *ACM Transactions on Computation Theory*, 11(4):26:1–26:42, 2019.

15 Olaf Beyersdorff, Leroy Chew, Meena Mahajan, and Anil Shukla. Are short proofs narrow? QBF resolution is *not* so simple. *ACM Trans. Comput. Log.*, 19(1):1:1–1:26, 2018.

16 Olaf Beyersdorff and Luke Hinde. Characterising tree-like Frege proofs for QBF. *Inf. Comput.*, 268, 2019.

17 Olaf Beyersdorff, Luke Hinde, and Ján Pich. Reasons for hardness in QBF proof systems. *ACM Transactions on Computation Theory*, 12(2):10:1–10:27, 2020.

18 Olaf Beyersdorff, Mikolás Janota, Florian Lonsing, and Martina Seidl. Quantified Boolean formulas. In Armin Biere, Marijn Heule, Hans van Maaren, and Toby Walsh, editors, *Handbook of Satisfiability*, Frontiers in Artificial Intelligence and Applications, pages 1177–1221. IOS Press, 2021.

19 Olaf Beyersdorff and Oliver Kutz. Proof complexity of non-classical logics. In N. Bezhanishvili and V. Goranko, editors, *Lectures on Logic and Computation - ESSLLI 2010 /ESSLLI 2011, Selected Lecture Notes*, pages 1–54. Springer-Verlag, Berlin Heidelberg, 2012.

20 Olaf Beyersdorff and Ján Pich. Understanding Gentzen and Frege systems for QBF. In *Proc. ACM/IEEE Symposium on Logic in Computer Science (LICS)*, 2016.

21 A. Blake. *Canonical expressions in boolean algebra.* PhD thesis, University of Chicago, 1937.

22 Benjamin Böhm, Tomás Peitl, and Olaf Beyersdorff. QCDCL with cube learning or pure literal elimination - what is best? In Luc De Raedt, editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1781–1787. ijcai.org, 2022.

23 Nader H. Bshouty. A subexponential exact learning algorithm for DNF using equivalence queries. *Inf. Process. Lett.*, 59(1):37–39, 1996.

24 Sam Buss and Jakob Nordström. Proof complexity and SAT solving. In Armin Biere, Marijn Heule, Hans van Maaren, and Toby Walsh, editors, *Handbook of Satisfiability*, Frontiers in Artificial Intelligence and Applications, pages 233–350. IOS Press, 2021.

25 Hubie Chen. Proof complexity modulo the polynomial hierarchy: Understanding alternation as a source of hardness. *ACM Transactions on Computation Theory*, 9(3):15:1–15:20, 2017.

26 Matthew Clegg, Jeff Edmonds, and Russell Impagliazzo. Using the Groebner basis algorithm to find proofs of unsatisfiability. In *Proc. 28th ACM Symposium on Theory of Computing*, pages 174–183, 1996.

27 Jonas Conneryd, Susanna F. de Rezende, Jakob Nordström, Shuo Pang, and Kilian Risse. Graph colouring is hard on average for polynomial calculus and Nullstellensatz. In *64th IEEE Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1–11. IEEE, 2023.

28 Stephen A. Cook and Phuong Nguyen. *Logical Foundations of Proof Complexity.* Cambridge University Press, 2010.

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

30 William Cook, Collette R. Coullard, and György Turán. On the complexity of cutting-plane proofs. *Discrete Applied Mathematics*, 18(1):25–38, 1987.

31 Susanna F. de Rezende, Mika Göös, Jakob Nordström, Toniann Pitassi, Robert Robere, and Dmitry Sokolov. Automating algebraic proof systems is NP-hard. In Samir Khuller and Virginia Vassilevska Williams, editors, *53rd Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, pages 209–222. ACM, 2021.

32 Uwe Egly, Florian Lonsing, and Magdalena Widl. Long-distance resolution: Proof generation and strategy extraction in search-based QBF solving. In *Proc. Logic for Programming, Artificial Intelligence, and Reasoning (LPAR)*, pages 291–308, 2013.

33 Michael A. Forbes, Amir Shpilka, Iddo Tzameret, and Avi Wigderson. Proof complexity lower bounds from algebraic circuit complexity. *Theory Comput.*, 17:1–88, 2021.

34 Nicola Galesi, Joshua A. Grochow, Toniann Pitassi, and Adrian She. On the algebraic proof complexity of tensor isomorphism. In Amnon Ta-Shma, editor, *38th Computational Complexity Conference (CCC)*, volume 264 of *LIPIcs*, pages 4:1–4:40. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023.

35 Alexandra Goultiaeva, Allen Van Gelder, and Fahiem Bacchus. A uniform approach for generating proofs and strategies for both true and false QBF formulas. In *IJCAI*, pages 546–553, 2011.

36 Nashlen Govindasamy, Tuomas Hakoniemi, and Iddo Tzameret. Simple hard instances for low-depth algebraic proofs. In *63rd IEEE Annual Symposium on Foundations of Computer Science (FOCS)*, pages 188–199. IEEE, 2022.

37 Joshua A. Grochow and Toniann Pitassi. Circuit complexity, proof complexity, and polynomial identity testing: The ideal proof system. *J. ACM*, 65(6):37:1–37:59, 2018.

38 Pavel Hrubeš. On lengths of proofs in non-classical logics. *Ann. Pure Appl. Logic*, 157(2–3):194–205, 2009.

39 Russell Impagliazzo, Pavel Pudlák, and Jirí Sgall. Lower bounds for the polynomial calculus and the Gröbner basis algorithm. *Comput. Complex.*, 8(2):127–144, 1999.

40 Mikolás Janota and Joao Marques-Silva. Expansion-based QBF solving versus Q-resolution. *Theor. Comput. Sci.*, 577:25–42, 2015.

41 Emil Jeřábek. *Weak pigeonhole principle, and randomized computation*. PhD thesis, Faculty of Mathematics and Physics, Charles University, Prague, 2005.

42 Jan Krajíček. *Bounded Arithmetic, Propositional Logic, and Complexity Theory*, volume 60 of *Encyclopedia of Mathematics and Its Applications*. Cambridge University Press, Cambridge, 1995.

43 Jan Krajíček. *Proof complexity*, volume 170 of *Encyclopedia of Mathematics and Its Applications*. Cambridge University Press, 2019.

44 Serge Lang. *Algebra (3. ed.)*. Addison-Wesley, 1993.

45 Knot Pipatsrisawat and Adnan Darwiche. On the power of clause-learning SAT solvers as resolution engines. *Artif. Intell.*, 175(2):512–525, 2011.

46 Toniann Pitassi and Iddo Tzameret. Algebraic proof complexity: progress, frontiers and challenges. *SIGLOG News*, 3(3):21–43, 2016.

47 Luca Pulina and Martina Seidl. The 2016 and 2017 QBF solvers evaluations (QBFEVAL'16 and QBFEVAL'17). *Artif. Intell.*, 274:224–248, 2019.

48 Ronald L. Rivest. Learning decision lists. *Machine Learning*, 2(3):229–246, 1987.

49 John Alan Robinson. A machine-oriented logic based on the resolution principle. *Journal of the ACM*, 12:23–41, 1965.

50 Sarah Sigley and Olaf Beyersdorff. Proof complexity of modal resolution. *J. Autom. Reason.*, 66(1):1–41, 2022.

51 R. Smolensky. Algebraic methods in the theory of lower bounds for boolean circuit complexity. In *Proc. ACM Symposium on Theory of Computing (STOC)*, pages 77–82. ACM Press, 1987.

52 Heribert Vollmer. *Introduction to Circuit Complexity – A Uniform Approach*. Texts in Theoretical Computer Science. Springer Verlag, Berlin Heidelberg, 1999.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il