Runtime vs. Extracted Proof Size: An Exponential Gap for CDCL on QBFs

Olaf Beyersdorff[1], **Benjamin Böhm**[1], **Meena Mahajan**[2]

[1]Institute of Computer Science, Friedrich Schiller University Jena, Germany
[2]The Institute of Mathematical Sciences (CI of Homi Bhabha National Institute), Chennai, India
olaf.beyersdorff@uni-jena.de, benjamin.boehm@uni-jena.de, meena@imsc.res.in

Abstract

Conflict-driven clause learning (CDCL) is the dominating algorithmic paradigm for SAT solving and hugely successful in practice. In its lifted version QCDCL, it is one of the main approaches for solving quantified Boolean formulas (QBF).

In both SAT and QBF, proofs can be efficiently extracted from runs of (Q)CDCL solvers. While for CDCL, it is known that the proof size in the underlying proof system propositional resolution matches the CDCL runtime up to a polynomial factor, we show that in QBF there is an exponential gap between QCDCL runtime and the size of the extracted proofs in QBF resolution systems. We demonstrate that this is not just a gap between QCDCL runtime and the size of *any* QBF resolution proof, but even the *extracted* proofs are exponentially smaller for some instances. Hence searching for a small proof via QCDCL (even with non-deterministic decision policies) will provably incur an exponential overhead for some instances.

1 Introduction

SAT solving has revolutionised the way we approach computationally hard problems (Vardi 2014). While SAT – determining whether a propositional formula is satisfiable – is the canonical NP-complete problem, modern SAT solvers successfully tackle huge instances of industrial problems from virtually all application domains (Biere et al. 2021). This algorithmic success has been extended to computationally even harder problems, in particular to the PSPACE-complete problem of solving quantified Boolean formulas (QBF), thus reaching to even further applications (Shukla et al. 2019).

SAT solving is dominated by the algorithmic paradigm of *conflict-driven clause learning* (CDCL) on which almost all contemporary SAT solvers are based (Marques Silva, Lynce, and Malik 2021). This approach was lifted to QBF in the form of QCDCL (Zhang and Malik 2002), one of the principal, but not the only (Biere 2004; Janota and Marques-Silva 2015), competitive QBF solving techniques, implemented e.g. in the state-of-the-art solvers DepQBF (Lonsing and Egly 2017) and Qute (Peitl, Slivovsky, and Szeider 2019).[1]

[1]These solvers go beyond plain QCDCL and use advanced techniques such as dependency learning that can improve performance. We will only consider plain QCDCL in this paper.

Both in SAT and in QBF there are intimate *connections between solving techniques and proof systems* (Buss and Nordström 2021; Beyersdorff et al. 2021). This manifests in the fact that each run of a solver on an unsatisfiable propositional formula (resp. a true or false QBF) can be understood as a proof of unsatisfiability (resp. truth or falsity) of the formula. While in principle, every solver gives thus rise to a proof system, CDCL corresponds to propositional resolution, which is arguably the most-studied and best-understood proof system in proof complexity.

One direction of this correspondence arises from the *efficient extraction of resolution proofs from CDCL traces* (Beame, Kautz, and Sabharwal 2004). A similar proof extraction also works from QCDCL to the QBF resolution system long-distance Q-Resolution (LD-Q-Res) (Zhang and Malik 2002; Balabanov and Jiang 2012).

These connections open the door towards *analysing the runtime of (Q)CDCL via proof complexity:* formulas without short resolution or LD-Q-Res proofs cannot possibly be solved efficiently by (Q)CDCL. Proof complexity – both in SAT and QBF – offers a wealth of techniques and crafted formulas on which exponential lower bounds for (QBF) resolution can be shown, implying analogous runtime lower bounds for the corresponding solvers (Buss and Nordström 2021; Beyersdorff 2022; Krajíček 2019).

Further, *proof extraction* is of huge practical importance as the extracted proofs can be used to certify answers of (potentially buggy) solvers (though practical proof logging employs more succinct proof formats than resolution (Wetzler, Heule, and Jr. 2014; Heule, Seidl, and Biere 2017)).

It is important, both theoretically and practically, to *understand how tight this proof extraction is*. This manifests in two related, yet different questions:

Q1: Solver runtime vs minimal proof size: Are there formulas that are hard for (Q)CDCL, but with short resolution (LD-Q-Res) proofs?

Q2: Solver runtime vs minimal *extracted* proof size: Are there formulas that are hard for (Q)CDCL, but where short resolution (LD-Q-Res) proofs can be extracted from (Q)CDCL runs?

Clearly, a positive answer for Q2 implies a positive answer for Q1 (hence answering Q2 positively is harder).

For SAT, Q1 received a *negative answer* in seminal work (Pipatsrisawat and Darwiche 2011; Atserias, Fichte, and

Thurley 2011), whereby resolution is polynomially equivalent to CDCL under a strong non-deterministic decision policy. Yet, the answer to Q1 is positive, once a practical decision scheme such as VSIDS is employed (Vinyals 2020).

For QBF, the situation is different: Q1 has a *positive answer* even for strong non-deterministic QCDCL models (Beyersdorff and Böhm 2023) (and hence also for practical QCDCL, which was known even earlier (Janota 2016)).

In contrast, Q2 has *not been considered before* (to the best of our knowledge), even though the question is arguably even more natural: when modelling solvers by proofs it makes sense to only consider extracted proofs that actually correspond to solver runs and disregard any other (possibly shorter) proofs of the same formula not stemming from a solver trace. One reason for this apparent neglect might be that the negative answer to Q1 for SAT (Pipatsrisawat and Darwiche 2011) obviously implies a negative answer to Q2 (we will argue in Section 6 that even for SAT there are subtle differences between Q1 and Q2). The situation is quite different for QBF as we show in this paper.

Our contributions. Our main result is a *positive answer* to Q2 for QBF. For this we construct specific QBF families that are exponentially hard for various QCDCL models, but with (exponentially long) QCDCL traces from which quadratic (in the size of the formula) QBF resolution proofs can be extracted. This considerably strengthens the previously known positive answer to Q1 (Beyersdorff and Böhm 2023).

Hence the obstacle for QCDCL is not to find the short proofs, it actually finds them, yet inevitably producing a huge overhead in the search. This overhead results from the fact that in contrast to CDCL, where clauses are learnt on conflicts, QCDCL also learns cubes (i.e. conjunctions of literals) from satisfying assignments. In QCDCL, clauses and cubes are used for unit propagations. Yet, as in CDCL, only clauses contribute to the extracted proof of false QBFs. Intuitively, we show that each QCDCL run on our target QBFs learns many cubes that do not appear in the extracted proof.

Technically, we achieve our results by (1) precisely modelling the QCDCL systems by proof systems in which the traces and learnt clauses/cubes are recorded, following the approach of (Beyersdorff and Böhm 2023); (2) showing a Master Theorem 4.8 that combines lower and upper bounds, employing a lower bound technique for QCDCL from (Böhm and Beyersdorff 2021); and (3) crafting new QBF families to which we apply our master theorem.

Our positive answer to Q2 is quite general in that it applies to *three different QCDCL models*, corresponding to the main QBF resolution systems Q-Res (Kleine Büning, Karpinski, and Flögel 1995), QU-Res (Van Gelder 2012) (implemented in (Slivovsky 2022)), and the previously mentioned system LD-Q-Res, underlying standard QCDCL.

While our results are purely theoretical, we believe that our findings will also be relevant to practitioners. In fact, our lower bounds imply that cube generation can provide a substantial bottleneck, even for false QBFs.

Organisation. We start in Section 2 by reviewing QBFs and relevant proof systems. Section 3 models different QCDCL paradigms as rigorous QCDCL proof systems, amenable to proof complexity analysis. In Section 4 we prove our master theorem for the combined upper and lower bounds, which we apply in Section 5 to answer Q2 for three QCDCL models. We conclude in Section 6 with a discussion. Due to space constraints, we omit some auxiliary results and proofs.

2 Preliminaries

Propositional and quantified formulas. Variables x and negated variables $\bar{x}$ are called *literals*. We denote the corresponding variable as $\text{var}(x) := \text{var}(\bar{x}) := x$.

A *clause* is a disjunction of literals, sometimes interpreted as a set of literals. A *unit clause* (ℓ) is a clause consisting of only one literal. The *empty clause* $(\bot)$ has zero literals. A clause C is *tautological* if $\{\ell, \bar{\ell}\} \subseteq C$ for some literal ℓ.

A *cube* is a conjunction of literals, sometimes viewed as a set of literals. We define a *unit cube* of a literal ℓ, denoted by $[\ell]$, and the *empty cube* $[\top]$ with 'empty literal' $\top$. A cube D is *contradictory* if $\{\ell, \bar{\ell}\} \subseteq D$ for some literal ℓ.

If C is a clause or a cube, we define $\text{var}(C) := \{\text{var}(\ell) : \ell \in C\}$. The negation of a clause $C = \ell_1 \vee \ldots \vee \ell_m$ is the cube $\neg C := \overline{C} := \bar{\ell}_1 \wedge \ldots \wedge \bar{\ell}_m$.

A *(total) assignment* σ of a set of variables V is a non-tautological set of literals such that for all $x \in V$ there is some $\ell \in \sigma$ with $\text{var}(\ell) = x$. A *partial assignment* σ of V is an assignment of a subset of V. A clause C is *satisfied* by σ if $C \cap \sigma \neq \emptyset$. A cube D is *falsified* by σ if $\neg D \cap \sigma \neq \emptyset$.

A *CNF* (conjunctive normal form) is a conjunction of clauses and a *DNF* (disjunctive normal form) is a disjunction of cubes. A CNF (DNF) is *satisfied* (*falsified*) by σ if all its clauses (cubes) are satisfied (falsified) by σ.

A *QBF* (quantified Boolean formula) $\Phi = \mathcal{Q} \cdot \phi$ consists of a propositional formula ϕ, called the *matrix*, and a *prefix* $\mathcal{Q}$. A *prefix* $\mathcal{Q} = Q_1 V_1 \ldots Q_s V_s$ consists of non-empty and pairwise disjoint sets of variables $V_1, \ldots, V_s$ and quantifiers $Q_1, \ldots, Q_s \in \{\exists, \forall\}$ with $Q_i \neq Q_{i+1}$ for $i \in [s-1]$. For a variable x in $\mathcal{Q}$, the *quantifier level* is $\text{lv}(x) := \text{lv}_\Phi(x) := i$, if $x \in V_i$. For $\text{lv}_\Phi(\ell_1) < \text{lv}_\Phi(\ell_2)$ we write $\ell_1 <_\Phi \ell_2$.

For a QBF $\Phi = \mathcal{Q} \cdot \phi$ with ϕ a CNF (DNF), we call Φ a *QCNF* (*QDNF*). We define $\mathfrak{C}(\Phi) := \phi$ (resp. $\mathfrak{D}(\Phi) := \phi$). Φ is an *AQBF* (augmented QBF), if $\phi = \psi \vee \chi$ with CNF ψ and DNF χ. We define $\text{var}(\Phi) := \bigcup_{C \in \Phi} \text{var}(C)$.

(Long-distance) Q(U)-resolution and Q(U)-consensus. Let C_1 and C_2 be two clauses (cubes) from a QCNF (QDNF) or AQBF Φ. Let ℓ be an existential (universal) literal with $\text{var}(\ell) \notin \text{var}(C_1) \cup \text{var}(C_2)$. The *resolvent* of $C_1 \vee \ell$ and $C_2 \vee \bar{\ell}$ over ℓ is defined as

$$(C_1 \vee \ell) \overset{\ell}{\bowtie}_\Phi (C_2 \vee \bar{\ell}) := C_1 \vee C_2$$

(resp. $(C_1 \wedge \ell) \overset{\ell}{\bowtie}_\Phi (C_2 \wedge \bar{\ell}) := C_1 \wedge C_2$).

Let $C := \ell_1 \vee \ldots \vee \ell_m$ be a clause from a QCNF or AQBF Φ such that $\ell_i \leqslant_\Phi \ell_j$ for all $i < j$, while $i, j \in [m]$. Let k be minimal such that $\ell_k, \ldots, \ell_m$ are universal. Then we can perform a *universal reduction* step and obtain

$$\text{red}_\Phi^\forall(C) := \ell_1 \vee \ldots \vee \ell_{k-1}.$$

Analogously, we perform *existential reduction* on cubes, which we denote as $\text{red}_\Phi^\exists(C)$.

If it is clear whether C is a clause or a cube, we can just write $\text{red}_\Phi(C)$ or even $\text{red}(C)$, if the QBF Φ is also obvious.

(Kleine Büning, Karpinski, and Flögel 1995) defined a Q-Res (Q-Con) proof π from a QCNF (QDNF) Φ of a clause (cube) C as a sequence $\pi = (C_i)_{i=1}^m$ of clauses (cubes) with $C_m = C$, and for each $i \in [m]$ one of the following holds:

- *Axiom:* $C_i \in \mathfrak{C}(\Phi)$; (resp. $C_i \in \mathfrak{D}(\Phi)$);
- *Resolution:* $C_i = C_j \overset{x}{\bowtie}_\Phi C_k$ with x existential (universal), $j, k < i$, and C_i non-tautological (non-contradictory);
- *Reduction:* $C_i = \text{red}^\forall_\Phi(C_j)$ (resp. $C_i = \text{red}^\exists_\Phi(C_j)$) for some $j < i$.

We call C the *root* of π.

(Balabanov and Jiang 2012) introduced an extension of Q-Res (Q-Con) proofs to LD-Q-Res (LD-Q-Con) proofs by replacing the resolution rule by

- *Resolution (long-distance):* $C_i = C_j \overset{x}{\bowtie}_\Phi C_k$ with x existential (universal) and $j, k < i$. The resolvent C_i is allowed to contain tautologies such as $u \vee \bar{u}$ (resp. contradictions $u \wedge \bar{u}$). If there is such a universal (existential) $u \in \text{var}(C_j) \cap \text{var}(C_k)$, then we require $x <_\Phi u$.

(Van Gelder 2012) presented a further extension QU-Res, (QU-Con), of Q-Res, (Q-Con), where we can resolve over arbitrary literals. Formally, it replaces the resolution rule by

- *Resolution (QU-Res)*: $C_i = C_j \overset{x}{\bowtie}_\Phi C_k$ with x existential *or* universal, $j, k < i$, and C_i non-tautological (non-contradictory).

A proof from Φ of the empty clause $(\bot)$ (resp. the empty cube $[\top]$) is called a *refutation* (*verification*) of Φ. In that case, Φ is called *false* (*true*).

3 A Framework for QCDCL Systems

First, we formalise QCDCL procedures as proof systems in order to analyse their complexity. We follow the approach initiated in (Beyersdorff and Böhm 2023; Böhm and Beyersdorff 2021; Böhm, Peitl, and Beyersdorff 2022a,b; Böhm and Beyersdorff 2023).

We store all relevant information of a QCDCL run in *trails*. Since QCDCL uses several runs and potentially also restarts, a QCDCL proof typically consists of many trails.

A *trail* $\mathcal{T}$ for a QCNF or AQBF Φ is a sequence of literals of Φ, including $\bot$ and $\top$. In general, a trail has the form

$$\mathcal{T} = (p_{(0,1)}, \ldots, p_{(0,g_0)}; \mathbf{d_1}, p_{(1,1)}, \ldots, p_{(1,g_1)}; \ldots; \mathbf{d_r}, p_{(r,1)}, \ldots, p_{(r,g_r)}), \tag{3.1}$$

such that the d_i are *decision literals* and $p_{(i,j)}$ are *propagated literals*. We write $x <_\mathcal{T} y$ if $x, y \in \mathcal{T}$ and x is left of y in $\mathcal{T}$. Trails can be thought of as non-tautological sets of literals, and therefore as (partial) assignments. A trail $\mathcal{T}$ has *run into conflict* if $\bot \in \mathcal{T}$ or $\top \in \mathcal{T}$.

Simply put, our QCDCL proofs can be viewed as sequences of trails. These trails cannot be created arbitrarily, but have to follow special rules, depending on the model.

We consider three different variants of QCDCL, each with a different underlying proof system, meaning that each variant generates proofs in its corresponding proof system. We use the policy notation from (Böhm and Beyersdorff 2023) to preserve consistency with previous works.

- $\mathsf{M_{LD}} := \mathsf{QCDCL}^{\text{LEV-ORD}}_{\text{ALL-RED,EXI-PROP}}$, which can be interpreted as the classic QCDCL variant that generates LD-Q-Res and LD-Q-Con proofs. All decisions have to follow quantification order (LEV-ORD). Reductions during unit propagation are always performed when possible (ALL-RED). Clauses can only propagate existential while cubes can only propagate universal literals (EXI-PROP).
- $\mathsf{M_Q} := \mathsf{QCDCL}^{\text{LEV-ORD}}_{\text{NO-RED,EXI-PROP}}$ is defined almost as $\mathsf{M_{LD}}$, but reductions during unit propagations are turned off (NO-RED). $\mathsf{M_Q}$ generates Q-Res or Q-Con proofs.
- $\mathsf{M_{QU}} := \mathsf{QCDCL}^{\text{LEV-ORD}}_{\text{NO-RED,ALL-PROP}}$ is an extension of $\mathsf{M_Q}$ where clauses can also propagate universal literals and analogously cubes propagate existentially (ALL-PROP). This generates QU-Res or QU-Con proofs.

Decisions can only be made when no more propagations are possible. Conflicts have a higher priority than propagations of literals. Hence, we never skip conflicts or propagations. For each propagated literal $p_{(i,j)}$ in a trail $\mathcal{T}$ the formula must contain a clause or a cube that caused this propagation by becoming a unit clause/cube. We denote such an antecedent clause/cube by $\text{ante}_\mathcal{T}(p_{(i,j)})$.

After a trail has run into a conflict, or if all variables were assigned, we start the learning process.

Definition 3.1 (learnable constraints)**.** *Let $\mathcal{T}$ be a trail for Φ of the form* (3.1) *with $p_{(r,g_r)} \in \{\bot, \top\}$. Starting with $\text{ante}_\mathcal{T}(\bot)$ (resp. $\text{ante}_\mathcal{T}(\top)$) we reversely resolve over the antecedent clauses (cubes) that propagated the existential (universal) variables, until we stop at some arbitrarily chosen point. Each antecedent and resolvent is reduced as soon as possible, regardless of the choice of policies. The clause (cube) we so derive is a* learnable constraint*. We denote the set of learnable constraints by $\mathfrak{L}(\mathcal{T})$.*

We can also learn cubes from trails that did not run into conflict. If $\mathcal{T}$ is a total assignment of the variables from Φ, then we define the set of learnable constraints as the set of cubes $\mathfrak{L}(\mathcal{T}) := \{\text{red}^\exists_\Phi(D) | \, D \subseteq \mathcal{T}$ and D satisfies $\mathfrak{C}(\Phi)\}$.

Definition 3.2 (QCDCL proof systems)**.** *Let* S *be one of* $\mathsf{M_{LD}}$, $\mathsf{M_Q}$, $\mathsf{M_{QU}}$. *An* S *proof ι from a QCNF $\Phi = \mathcal{Q} \cdot \phi$ of a clause or cube C is a sequence of triples*

$$\iota := [(\mathcal{T}_i, C_i, \pi_i)]_{i=1}^m,$$

where $C_m = C$, each $\mathcal{T}_i$ is a trail following the policies of S*, each $C_i \in \mathfrak{L}(\mathcal{T}_i)$ is one of the constraints we can learn from the trail, and π_i is the derivation of C_i we get by performing the steps in Definition 3.1. We define $\mathfrak{R}(\iota)$ as the extracted proof of C that we get by sticking together suitable π_i. For $C = (\bot)$, ι is an* S refutation *of Φ. For $C = [\top]$, ι is an* S verification *of Φ.*

The trail size *of ι is defined as trail-size$(\iota) := |\iota| := \sum_{i=1}^m |\mathcal{T}_i|$ and the* extracted proof size *of ι is defined as the size of the extracted proof, i.e. extr-size$(\iota) := |\mathfrak{R}(\iota)|$.*

Obviously, we have extr-size$(\iota) \leqslant$ trail-size(ι).

On true formulas, these three variants generate consensus (Q-Con, LD-Q-Con or QU-Con) verifications. As such proofs are only defined on QDNFs, they are formally not verifications of the QCNF Φ, but of a QDNF consisting of cubes that satisfy $\mathfrak{C}(\Phi)$. These cubes (often called *initial cubes*) correspond to the cubes that can be learned whenever a trail does not run into conflict (cf. 3.1). From now on, we will refer to these verifications as *verifications of* Φ.

All combinations of the above policies lead to sound and complete proof systems (and algorithms).

Theorem 3.3 ((Böhm and Beyersdorff 2023)). *All defined QCDCL variants are sound and complete.*

4 Combining Lower and Upper Bounds

In this section we provide a general framework for showing a lower bound for trail size combined with an upper bound for the extracted proof size of QCDCL systems. For the lower bound we employ the gauge technique from (Böhm and Beyersdorff 2021) which we review first.

The Gauge Lower Bound Technique

To ease notation, we will assume that prefixes of Σ_3^b QCNFs have the form $\exists X \forall U \exists T$, for sets of literals X, U, T, and we will use the notions of X-, U- and T-variables and -literals. Further, we define certain types of clauses:

- X-clauses consist of X-literals only (analogously we define U-clauses and T-clauses),
- XT-clauses consist of at least one X- and at least one T-literal, but no U-literals,
- XUT-clauses consist of at least one X-, U- and T-literal.

For our lower bounds we will make use of a technique that only holds for a particular (but still sufficiently large) class of formulas which is determined by the so-called *XT-property*. Intuitively, this property ensures that there cannot be any direct connections between the inner and the outer quantifier block in Σ_3^b QCNFs.

Definition 4.1 ((Beyersdorff and Böhm 2023)). *We say that Φ fulfils the* XT-property*, if $\mathfrak{C}(\Phi)$ contains no XT-clauses, no T-clauses that are unit (or empty) and no two T-clauses from $\mathfrak{C}(\Phi)$ are resolvable.*

This property does not only hold for the initial formula, but also for all clauses that can be derived via LD-Q-Res.

Lemma 4.2 ((Beyersdorff and Böhm 2023)). *If Φ is a Σ_3^b QCNF that fulfils the XT-property, then it is not possible to derive XT-clauses or new T-clauses via* LD-Q-Res *from Φ.*

The lower bound technique requires two further notions – one which is quite natural for proofs that are extracted from QCDCL runs (*fully reduced*), and one which is closely related to the XT-property (*primitive*).

Definition 4.3 (fully reduced proofs (Böhm and Beyersdorff 2021; Böhm, Peitl, and Beyersdorff 2022a)). *A* LD-Q-Res *refutation π of a QCNF Φ is* fully reduced*, if for each clause $C \in \pi$ that contains universal literals that are reducible, the reduction step is performed immediately and C is not used otherwise in the proof.*

Definition 4.4 (primitive proofs (Böhm and Beyersdorff 2021; Böhm, Peitl, and Beyersdorff 2022a)). *A* LD-Q-Res *proof π from a Σ_3^b formula is* primitive*, if there are no two XUT-clauses in π that are resolved over an X-variable.*

Note that every fully reduced primitive LD-Q-Res proof has to be a Q-Res proof. Therefore, from now on we will only mention fully reduced primitive Q-Res proofs. Furthermore, it is easy to see that each QCDCL variant generates fully reduced proofs by Definition 3.1.

Now that we have defined the class of formulas and the class of proofs for which the lower bound technique is applicable, we introduce the measure that determines the lower bound itself. Intuitively, the *gauge* of a Σ_3^b QCNF is the minimal number of X-literals that will be necessarily piled up in any derivation of an X-clause in which it is only allowed to resolve over T-literals.

Definition 4.5 ((Böhm and Beyersdorff 2021)). *Let Φ be a Σ_3^b QCNF with prefix $\exists X \forall U \exists T$. We define W_Φ as the set of all* Q-Res *proofs π from Φ of X-clauses C_π, such that π consists of resolutions over T-literals and reductions only. We define*

$$gauge(\Phi) := \min\{|C_\pi| \ : \ C_\pi \text{ is the root of some } \pi \in W_\Phi\}.$$

Combining these notions and conditions we obtain the *gauge lower bound method*.

Theorem 4.6 ((Böhm and Beyersdorff 2021)). *Each fully reduced primitive* Q-Res *refutation of a Σ_3^b QCNF Φ that fulfils the XT-property has size $2^{\Omega(gauge(\Phi))}$.*

To obtain an exponential lower bound for a QCDCL variant via this technique, we will construct Σ_3^b QCNFs that fulfil the XT-property and have linear gauge, such that the QCDCL variant generates primitive proofs on these QBFs.

The Master Theorem

We can now approach our main Theorem 4.8. Throughout the paper we will construct QBFs by combining two QCNFs into one single QCNF by concatenating the two quantifier prefixes and conjoining both matrices.

Definition 4.7. *Let $\Phi = \mathcal{Q} \cdot \phi$ and $\Psi = \mathcal{R} \cdot \psi$ be two QCNFs. Let $\Psi' = \mathcal{R}' \cdot \psi'$ be the QCNF that is obtained after renaming the variables from Ψ such that $var(\Phi) \cap var(\Psi') = \emptyset$. Then we define the* disjoint composition *of Φ and Ψ as the QCNF $dc(\Phi, \Psi) := \mathcal{Q}\mathcal{R}' \cdot \phi \wedge \psi'$.*

From now on, we will assume that variables from ϕ and variables from ψ' do not share the same quantifier level in $\mathcal{Q}\mathcal{R}'$. In particular, Φ will always end with an existential quantifier and Ψ will start with a universal quantifier.

The next theorem is our main technical result which will be used for all following main results in the paper.

Theorem 4.8. *Let Φ_n and Ψ_n be two formulas with the following properties:*

1. *Φ_n is a Σ_3^b QCNF that fulfils the XT-property, is false, and has M_{LD} (resp. M_Q and M_{QU}) refutations with trail size s.*
2. *(for M_{QU}) Each* QU-Res *refutation of Φ_n is a* Q-Res *refutation. I.e., it is not possible to resolve two clauses that were derived from Φ_n over a universal variable.*

3. *Ψ_n is true, and for each M_{LD} (resp. M_{Q} and M_{QU}) proof ι_E of some cube E from $dc(\Phi_n, \Psi_n)$ with $var(E) \subseteq var(\Phi_n)$ we have $|\iota_E| \geqslant r$.*

Then for each M_{LD} (resp. M_{Q} and M_{QU}) refutation θ of $dc(\Phi_n, \Psi_n)$ we have trail-size$(\theta) \in \min(2^{\Omega(gauge(\Phi_n))}, r)$, but there exists a M_{LD} (resp. M_{Q} and M_{QU}) refutation ι of $dc(\Phi_n, \Psi_n)$ with extr-size$(\iota) \leqslant s$.

Before proving the theorem, let us provide some intuition on this result and the way we will apply it later in Section 5.

One can obtain an exponential separation between the trail size and the extracted size of a proof by choosing Φ_n and Ψ_n in such a way that s is polynomial in n, gauge(Φ_n) is linear in n, and r is exponential in n.

Intuitively, in order to refute dc(Φ_n, Ψ_n), we can consider two possibilities:

1. We never learn a cube E with var$(E) \subseteq$ var(Φ_n). Then each learned cube contains some variables from Ψ_n. We will show in the proof of Theorem 4.8 that we will then generate fully reduced primitive proofs and therefore the gauge lower bound applies (Theorem 4.6), resulting in an exponential lower bound.
2. We learn at least one cube E with var$(E) \subseteq$ var(Φ_n). But then the derivation of this cube E itself needs QCDCL proofs of size r, which is exponential by assumption.

For the polynomial upper bound on the extracted proof size, we are allowed to learn and derive any cube that is useful. Since the derivations of these cubes do not appear in the extracted proof, they may even have exponential size without increasing the size of the extracted proof. As we assume that Φ_n is easy to refute in the considered QCDCL variant, we can reproduce this refutation, as verifying Ψ_n comes for free when only measuring the extracted proof size.

A visualisation of this separation is depicted in Figure 1.

Proof of Theorem 4.8. Obviously, dc(Φ_n, Ψ_n) is a false formula. It suffices to show that each M_{LD} (resp. M_{Q} and M_{QU}) refutation θ of dc(Φ_n, Ψ_n), in which each learned cube contains some literals from Ψ_n, has trail-size$(\theta) = 2^{\Omega(\text{gauge}(\Phi_n))}$. We show that such θ generates fully reduced primitive Q-Res refutations of Φ_n. The lower bound then follows by Theorem 4.6.

Assume, for the sake of contradiction, that there is such a M_{LD} (resp. M_{Q} and M_{QU}) refutation θ of dc(Φ_n, Ψ_n) such that the extracted proof $\mathfrak{R}(\theta)$ is not a fully reduced primitive Q-Res refutation of Φ_n. By the definition of a disjoint composition, $\mathfrak{R}(\theta)$ is a refutation of Φ_n. By condition 2 of the theorem, $\mathfrak{R}(\theta)$ does not contain a resolution step over a universal variable, i.e. all resolutions are over existential variables. There must be a resolution step in $\mathfrak{R}(\theta)$ between two XUT-clauses over an X-literal. Consider the first trail $\mathcal{T}$ in θ in which such a resolution, say $C \overset{x}{\bowtie} D$ with $x \in C$ and $\bar{x} \in D$, appeared in the learning phase.

Then one of these two clauses must have been an antecedent clause for the pivot, say ante$_{\mathcal{T}}(x) = C$. The clause C must contain at least one T-literal, say $t_1 \in C$. Then we need $\bar{t}_1 <_{\mathcal{T}} x$ and therefore there exists an antecedent clause $A_1 :=$ ante$_{\mathcal{T}}(\bar{t}_1)$. Because of the XT-property, A_1 cannot be a unit clause, hence it must be either a non-unit T-clause, or a clause with a U-literal. If A_1 is a non-unit T-clause, then we can find another $\bar{t}_1 \neq t_2 \in A_1$, for which we would find another antecedent clause $A_2 :=$ ante$_{\mathcal{T}}(\bar{t}_2)$. We can repeat this argument, until at some point the antecedent clause $A_j =$ ante$_{\mathcal{T}}(\bar{t}_j)$ for some T-literal $\bar{t}_j$ contains a U-literal, say $u \in A_j$.

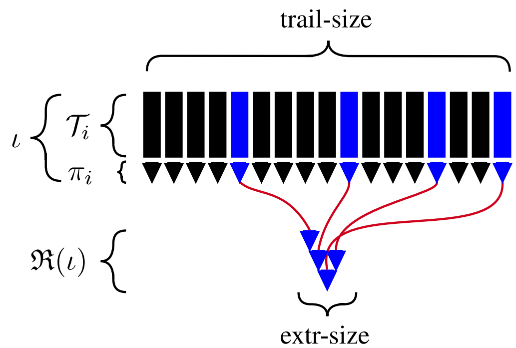

Figure 1: Visualisation of the separations following from Theorem 4.8: The rectangles symbolise the trails of a proof, the triangles represent the derivations of the learned constraints. Black rectangles and triangles denote trails and derivations for learned cubes, while blue rectangles and triangles denote trails and derivations of learned clauses. As the last learned constraint is empty, i.e. the empty clause, all derivations of learned clauses can be stuck together to obtain a refutation of the original formula. The derivations of learned cubes will not be used for the extracted refutation The separation between the measures trail-size and extr-size occurs when the number of black trails is exponential while the number of blue trails is polynomial.

Because $u <_{\Phi_n} t_j$, we need $\bar{u} <_{\mathcal{T}} \bar{t}_j <_{\mathcal{T}} x$ in order to propagate $\bar{t}_j$. Because our decisions need to be level-ordered, we conclude that $\bar{u}$ was propagated. It is not possible for $\bar{u}$ to have been propagated by a clause, because otherwise we could perform a universal resolution step in $\mathfrak{R}(\theta)$. Therefore $\bar{u}$ must have been propagated by a cube $F :=$ ante$_{\mathcal{T}}(\bar{u})$ (note that $u \in F$). From now on, we assume that $\bar{u}$ is the first U-literal that was propagated in $\mathcal{T}$ by a cube and let F be the corresponding antecedent cube (we do not need the connection to A_j anymore).

By our assumption from the beginning of the proof, F contains some literals from Ψ_n. We can safely assume that at least one of these is universal, otherwise all literals from Ψ_n would have been reduced away during the learning of F. Let $w \in F$ be such a universal literal from Ψ_n. Then we need $w <_{\mathcal{T}} \bar{u}$ because we cannot reduce universally in cubes. That means w was also propagated by some constraint $G :=$ ante$_{\mathcal{T}}(w)$, which is either a clause or a cube. We distinguish these two cases.

Case 1. G is a cube. This cube G cannot contain any U-literals from Φ_n, otherwise they must have been propagated via a cube (not possible because $\bar{u}$ was the first propagated U-literal via a cube in $\mathcal{T}$) or decided (not possible because decisions need to be level-ordered) before $\bar{u}$ and x in $\mathcal{T}$. We conclude that only existential literals from Φ_n appear in G.

Let π_G be the LD-Q-Con (resp. QU-Con) subproof of G from $\mathrm{dc}(\Phi_n, \Psi_n)$. We can restrict π_G to an LD-Q-Con (resp. QU-Con) proof ρ from Φ_n by just deleting all literals from Ψ_n (and, if necessary, delete redundant cubes). But then ρ is a LD-Q-Con (resp. QU-Con) proof of G', where G' is a cube that only contains existential literals that are also contained in Φ_n. If we reduce G' existentially, we obtain a verification of Φ_n, contradicting the falsity of Φ_n.

Case 2. G is a clause. This case is only relevant for M_{QU}. Again we obtain a contradiction. We omit the proof.

We thus obtain contradictions in both cases. Hence, our assumption that $\mathfrak{R}(\theta)$ was not fully reduced primitive is false. We conclude that $|\theta| \in 2^{\Omega(\mathrm{gauge}(\Phi_n))}$ by Theorem 4.6.

The upper bound can be shown by the following construction: Given an M_{LD} (resp. M_{Q} and M_{QU}) refutation ι' of Φ_n with trail size s, we can essentially reproduce all trails from ι' for a M_{LD} (resp. M_{Q} and M_{QU}) refutation ι of $\mathrm{dc}(\Phi_n, \Psi_n)$. The only difference is that, whenever a cube is learned in ι', we need to verify Ψ_n to learn this cube in ι. However, as this verification of Ψ_n does not appear in $\mathfrak{R}(\iota)$, we conclude $\mathfrak{R}(\iota) = \mathfrak{R}(\iota')$ and therefore $\mathfrak{R}(\iota)$ is of size at most s. $\square$

5 Separations for QCDCL Models

We will now put this general idea into action and construct separations between the trail size and the extracted proof size for each of the three QCDCL variants M_{LD}, M_{Q} and M_{QU}, corresponding to their respective underlying proof systems LD-Q-Res, Q-Res and QU-Res.

QCDCL Based on LD-Q-Resolution

We start with M_{LD} which corresponds to standard QCDCL as used in modern state-of-the art QBF solvers (Lonsing and Egly 2017; Peitl, Slivovsky, and Szeider 2019).

First, we need to find a true QCNF that is hard for M_{LD} and fulfils the properties of Ψ_n from Theorem 4.8. We recall the lower bound on true formulas from (Böhm, Peitl, and Beyersdorff 2022b) which uses the notion of the *twin formula* of a QCNF as well as the *reversion*.

Definition 5.1 (twin formulas, (Böhm, Peitl, and Beyersdorff 2022b)). *Let $\Lambda = \exists X \forall U \exists T \cdot \mathfrak{C}(\Lambda)$ be a QCNF. Let $U = \{u_1, \dots, u_m\}$ and let $v_1, \dots, v_m$ be variables not occuring in Λ. Then the* twin formula *of Λ is the QCNF $\mathtt{Twin}\Lambda$ defined as*

$$\mathtt{Twin}\Lambda := \exists X \forall (U \cup \{v_1, \dots, v_m\}) \exists T \cdot \mathfrak{C}(\Lambda) \wedge \bigwedge_{C \in \mathfrak{C}(\Lambda)} C[u_1/v_1, \dots, u_m/v_m],$$

where u_i/v_i indicates that all occurrences of u_i are substituted by v_i.

Definition 5.2 ((Böhm, Peitl, and Beyersdorff 2022b)). *If $\Lambda = \mathcal{Q}_1 V_1 \mathcal{Q}_2 V_2 \dots \mathcal{Q}_k V_k \cdot \bigwedge_{j=1}^m C_j$ is a QCNF with $\mathcal{Q}_i \in \{\exists, \forall\}$ and disjoint sets of variables V_i for $i = 1, \dots, k$, then the* reversion $\mathtt{Rev}(\Lambda)$ *of Λ is the QCNF*

$$\mathcal{Q}'_1 V_1 \mathcal{Q}'_2 V_2 \dots \mathcal{Q}'_k V_k \forall w \exists c_1, \dots, c_m \cdot (\bar{c}_1 \vee \dots \vee \bar{c}_m) \wedge \bigwedge_{j=1}^m \bigwedge_{\ell \in C_j} (\bar{\ell} \vee w \vee c_j) \wedge (\bar{\ell} \vee \bar{w} \vee c_j)$$

where $\mathcal{Q}'_i = \forall$ if $\mathcal{Q}_i = \exists$, and $\mathcal{Q}'_i = \exists$ if $\mathcal{Q}_i = \forall$, and $w, c_1, \dots, c_m$ are new variables not contained in Λ.

It is easy to see that the reversion flips the truth value:

Lemma 5.3 ((Böhm, Peitl, and Beyersdorff 2022b)). *If Λ is a QCNF, then $\mathtt{Rev}(\Lambda)$ is true if and only if Λ is false.*

The next theorem is a generalization of one of the main results from (Böhm, Peitl, and Beyersdorff 2022b). Instead of proving a lower bound on verifications of a particular formula Ψ (as done in that paper), we consider derivations of any cube from a formula $\mathrm{dc}(\Gamma, \Psi)$, such that this cube does not contain literals from Ψ. Hence, by choosing Γ as the empty formula, one obtains the result from (Böhm, Peitl, and Beyersdorff 2022b).

Theorem 5.4. *Let Λ be a false Σ_3^b QCNF with the prefix $\exists X \forall U \exists T$ and let Γ be an arbitrary QCNF. Let ι_E be an M_{LD} proof of some cube E from $dc(\Gamma, \mathtt{Rev}(\mathtt{Twin}\Lambda))$ such that $var(E) \subseteq var(\Gamma)$. Additionally, let all clauses $C \in \mathfrak{C}(\Lambda)$ contain at least one U- and one T-literal. If the QCNF $\mathtt{Twin}\Lambda$ needs fully reduced primitive Q-Res refutations of size s, then $|\iota_E| \geqslant s$.*

The proof of the above theorem follows along the same lines as the corresponding theorem from (Böhm, Peitl, and Beyersdorff 2022b).

Next, we want to find specific formulas to which Theorem 5.4 can be applied. We recall the well-known equality formulas, introduced in (Beyersdorff, Blinkhorn, and Hinde 2019), as well as a modification from (Böhm, Peitl, and Beyersdorff 2022b). This modification adds a U-literal to some clauses such that each clause now contains at least one U- and one T-literal, which is a precondition for Theorem 5.4.

Definition 5.5 ((Beyersdorff, Blinkhorn, and Hinde 2019; Böhm, Peitl, and Beyersdorff 2022b)). *The QCNF $\mathtt{Eq}_n$ consists of the prefix $\exists x_1, \dots, x_n \forall u_1, \dots, u_n \exists t_1, \dots, t_n$ and the matrix $x_i \vee u_i \vee t_i$, $\bar{x}_i \vee \bar{u}_i \vee t_i$, $\bar{t}_1 \vee \dots \vee \bar{t}_n$ for $i = 1, \dots, n$.*

The QCNF $\mathtt{ModEq}_n$ consists of the prefix $\exists x_1, \dots, x_n \forall u_1, \dots, u_n, p \exists t_1, \dots, t_n$ and the matrix $x_i \vee u_i \vee t_i$, $\bar{x}_i \vee \bar{u}_i \vee t_i$, $p \vee \bar{t}_1 \vee \dots \vee \bar{t}_n$, $\bar{p} \vee \bar{t}_1 \vee \dots \vee \bar{t}_n$ for $i = 1, \dots, n$.

Many properties of $\mathtt{Eq}_n$ carry over to $\mathtt{ModEq}_n$ or even $\mathtt{TwinModEq}_n$. This is important for obtaining exponential lower bounds via Theorem 4.8.

Proposition 5.6 ((Beyersdorff, Blinkhorn, and Hinde 2019; Beyersdorff and Böhm 2023; Böhm and Beyersdorff 2021; Böhm, Peitl, and Beyersdorff 2022a,b)). *$\mathtt{Eq}_n$ needs QU-Res refutations of size $2^{\Omega(n)}$ but has M_{LD} refutations of quadratic trail size. Furthermore, $\mathtt{Eq}_n$ and $\mathtt{ModEq}_n$ fulfil the XT-property and $gauge(\mathtt{Eq}_n) = gauge(\mathtt{TwinModEq_n}) = n$. Hence, $\mathtt{Eq}_n$ and $\mathtt{TwinModEq_n}$ need exponential-size fully reduced primitive Q-Res refutations.*

Using Theorem 5.4 and Proposition 5.6, we obtain:

Corollary 5.7. *Let* $\mathtt{RTME}_n := \mathtt{Rev}(\mathtt{TwinModEq}_n)$. *Then for each QCNF* Γ_n, *all* $\mathsf{M_{LD}}$ *proofs of any cube* E *with* $var(E) \subseteq var(\Gamma_n)$ *from the disjoint composition* $dc(\Gamma_n, \mathtt{RTME}_n)$ *have exponential trail size.*

Definition 5.8. *We define the QCNF* $\mathtt{ERTME}_n$ *as the disjoint composition* $\mathtt{ERTME}_n := dc(\mathtt{Eq}_n, \mathtt{RTME}_n)$.

After applying our Master Theorem 4.8 by setting $\Phi_n := \mathtt{Eq}_n$ and $\Psi_n := \mathtt{RTME}_n$, where s is quadratic by Proposition 5.6 and r is exponential by Corollary 5.7, we conclude:

Corollary 5.9. *For each* $\mathsf{M_{LD}}$ *refutation* θ_n *of* $\mathtt{ERTME}_n$ *we have trail-size*$(\theta_n) \in 2^{\Omega(n)}$, *but there exists an* $\mathsf{M_{LD}}$ *refutation* ι_n *of* $\mathtt{ERTME}_n$ *with extr-size*$(\iota_n) \in O(n^2)$.

QCDCL Based on Q-Resolution

For the separation on the QCDCL model $\mathsf{M_Q}$ we need a formula with linear gauge that is still easy for $\mathsf{M_Q}$. Since $\mathsf{M_Q}$ generates Q-Res proofs, $\mathtt{Eq}_n$ will not work because it is hard for Q-Res (and even QU-Res) (Beyersdorff, Blinkhorn, and Hinde 2019). Therefore we introduce the simplicity formulas $\mathtt{Sim}_n$, which are similar to $\mathtt{Eq}_n$, but now all universal literals occur in only one polarity.

Definition 5.10 (Simplicity formula)**.** *The QCNF* $\mathtt{Sim}_n$ *consists of the prefix* $\exists x_1, \ldots, x_n \forall u_1, \ldots, u_n \exists t_1, \ldots, t_n$ *and the matrix* $x_i \vee u_i \vee t_i$, $\bar{x}_i \vee u_i \vee t_i$, $\bar{t}_1 \vee \ldots \vee \bar{t}_n$ *for* $i = 1, \ldots, n$.

One can easily construct short $\mathsf{M_Q}$ and $\mathsf{M_{QU}}$ refutations for $\mathtt{Sim}_n$.

Proposition 5.11. $\mathtt{Sim}_n$ *has* $\mathsf{M_Q}$ *and* $\mathsf{M_{QU}}$ *refutations with quadratic trail size.*

Yet, the gauge properties of $\mathtt{Eq}_n$ carry over to $\mathtt{Sim}_n$.

Proposition 5.12. *The QCNF* $\mathtt{Sim}_n$ *fulfils the XT-property and gauge*$(\mathtt{Sim}_n) = n$. *Hence, each fully reduced primitive* Q-Res *refutation of* $\mathtt{Sim}_n$ *has exponential size.*

Now that we have found our candidate for Φ_n in Theorem 4.8, let us construct a suitable formula for Ψ_n. For this, we need to find a true formula that is hard to verify in $\mathsf{M_Q}$. We will again make use of the idea of a reversion of a formula that is already hard for Q-Res (and QU-Res).

Proposition 5.13. *The true QCNF* $\mathtt{Rev}(\mathtt{Eq}_n)$ *needs exponential-trail-size* QU-Con *verifications.*

Next, we have to show that deriving a cube from $\mathrm{dc}(\Gamma, \Psi)$ without variables from Ψ is as hard as verifying Ψ.

Proposition 5.14. *For each QCNF* Γ *and* Ψ, *from each* $\mathsf{M_Q}$ *(resp.* $\mathsf{M_{QU}}$*) proof* ι_E *of some cube* E *from* $dc(\Gamma, \Psi)$ *with* $var(E) \subseteq var(\Gamma)$ *we can extract a* Q-Con *(resp.* QU-Con*) verification* ρ *of* Ψ *with* $|\rho| \leqslant |\iota_E|$.

From Propositions 5.13 and 5.14 we conclude:

Corollary 5.15. *For each QCNF* Γ_n, *the disjoint composition* $dc(\Gamma_n, \mathtt{Rev}(\mathtt{Eq}_n))$ *needs exponential-trail-size* $\mathsf{M_Q}$ *and* $\mathsf{M_{QU}}$ *proofs of any cube* E *with* $var(E) \subseteq var(\Gamma_n)$.

We combine $\mathtt{Sim}_n$ with $\mathtt{Rev}(\mathtt{Eq}_n)$ and obtain our formula for the separation.

Definition 5.16. *We define the QCNF* $\mathtt{SRE}_n$ *as the disjoint composition* $\mathtt{SRE}_n := dc(\mathtt{Sim}_n, \mathtt{Rev}(\mathtt{Eq}_n))$.

Applying Theorem 4.8 with $\Phi_n := \mathtt{Sim}_n$ and $\Psi_n := \mathtt{Rev}(\mathtt{Eq}_n)$ where s is quadratic by Proposition 5.11 and r is exponential by Corollary 5.15, we conclude:

Corollary 5.17. *For each* $\mathsf{M_Q}$ *refutation* θ_n *of* $\mathtt{SRE}_n$ *we have trail-size*$(\theta_n) \in 2^{\Omega(n)}$, *but there exists an* $\mathsf{M_Q}$ *refutation* ι_n *of* $\mathtt{SRE}_n$ *with extr-size*$(\iota_n) \in O(n^2)$.

QCDCL Based on QU-Resolution

For the last separation in the QCDCL model $\mathsf{M_{QU}}$ – recently implemented as a QBF solver (Slivovsky 2022) – we can use the same formulas as for $\mathsf{M_Q}$ and Q-Res, as we only have to prove that no genuine QU-Res proofs can be generated.

Lemma 5.18. *Each* QU-Res *refutation of* $\mathtt{Sim}_n$ *is a* Q-Res *refutation.*

Proof. All universal variables from $\mathtt{Sim}_n$ occur only in one polarity, hence we cannot resolve over them. □

As for Corollary 5.17, we conclude:

Corollary 5.19. *For each* $\mathsf{M_{QU}}$ *refutation* θ_n *of* $\mathtt{SRE}_n$ *we have trail-size*$(\theta_n) \in 2^{\Omega(n)}$, *but there exists a* $\mathsf{M_{QU}}$ *refutation* ι_n *of* $\mathtt{SRE}_n$ *with extr-size*$(\iota_n) \in O(n^2)$.

6 Conclusion

Separating the two measures *trail size* and *extracted proof size* yields interesting consequences for QCDCL: lower bounds on trail size do not necessarily carry over to extracted proof size. This indicates that traditional proof systems such as Q-Res (and their extensions) are not perfectly suited to model QCDCL runs. Instead the systems need to track all non-redundant learned constraints (clauses and cubes).

Our separations only hold if decisions follow the quantifier prefix (using LEV-ORD). Allowing arbitrary decisions strengthens the three QCDCL variants to the point that they polynomially simulate the underlying proof systems Q-Res, QU-Res and LD-Q-Res (Böhm and Beyersdorff 2023). Besides changing the order of decisions, there are other approaches trying to avoid problems caused by the asymmetrical behaviour of clauses and cubes by changing the encoding (Zhang 2006; Goultiaeva and Bacchus 2013; Tu, Hsu, and Jiang 2015).

In conclusion, our results depict another crucial difference between SAT and QBF solving. In fact, as SAT solving only works on clauses, trail size and extracted proof size in SAT differ by exactly a linear factor, leading to a precise negative answer to Q2. Q1 also has a negative answer in SAT (Pipatsrisawat and Darwiche 2011), yet the precise overhead of CDCL trail size over minimal resolution size is still subject to ongoing research: it is at most cubic (Beyersdorff and Böhm 2023), but at least linear as very recently shown (Vinyals et al. 2023). This confirms again that comparing trail size and extracted proof size (Q2) is more challenging than comparing trail size and minimal proof size (Q1), not only for QBF as done here, but also for SAT.

Acknowledgments

This work was done in part while the authors were visiting the Simons Institute for the Theory of Computing during the Extended Reunion on Satisfiability in 2023.

Research was supported by grants from the Carl-Zeiss Foundation, DFG grant BE 4209/3-1, and a joint DAAD/DST grant.

References

Atserias, A.; Fichte, J. K.; and Thurley, M. 2011. Clause-Learning Algorithms with Many Restarts and Bounded-Width Resolution. *J. Artif. Intell. Res.*, 40: 353–373.

Balabanov, V.; and Jiang, J.-H. R. 2012. Unified QBF Certification and Its Applications. *Form. Methods Syst. Des.*, 41(1): 45–65.

Beame, P.; Kautz, H. A.; and Sabharwal, A. 2004. Towards Understanding and Harnessing the Potential of Clause Learning. *J. Artif. Intell. Res. (JAIR)*, 22: 319–351.

Beyersdorff, O. 2022. Proof Complexity of Quantified Boolean Logic – a Survey. In Benini, M.; Beyersdorff, O.; Rathjen, M.; and Schuster, P., eds., *Mathematics for Computation (M4C)*, 353–391. World Scientific.

Beyersdorff, O.; Blinkhorn, J.; and Hinde, L. 2019. Size, Cost, and Capacity: A Semantic Technique for Hard Random QBFs. *Logical Methods in Computer Science*, 15(1).

Beyersdorff, O.; and Böhm, B. 2023. Understanding the Relative Strength of QBF CDCL Solvers and QBF Resolution. *Log. Methods Comput. Sci.*, 19(2).

Beyersdorff, O.; Janota, M.; Lonsing, F.; and Seidl, M. 2021. Quantified Boolean Formulas. In Biere, A.; Heule, M.; van Maaren, H.; and Walsh, T., eds., *Handbook of Satisfiability*, Frontiers in Artificial Intelligence and Applications, 1177–1221. IOS Press.

Biere, A. 2004. Resolve and Expand. In Hoos, H. H.; and Mitchell, D. G., eds., *Theory and Applications of Satisfiability Testing SAT*, volume 3542 of *Lecture Notes in Computer Science*, 59–70. Springer.

Biere, A.; Heule, M.; van Maaren, H.; and Walsh, T., eds. 2021. *Handbook of Satisfiability*, Frontiers in Artificial Intelligence and Applications. IOS Press.

Böhm, B.; and Beyersdorff, O. 2021. Lower Bounds for QCDCL via Formula Gauge. In Li, C.-M.; and Manyà, F., eds., *Theory and Applications of Satisfiability Testing (SAT)*, 47–63. Cham: Springer International Publishing.

Böhm, B.; and Beyersdorff, O. 2023. QCDCL vs QBF Resolution: Further Insights. In Mahajan, M.; and Slivovsky, F., eds., *26th International Conference on Theory and Applications of Satisfiability Testing (SAT 2023)*, volume 271 of *Leibniz International Proceedings in Informatics (LIPIcs)*, 4:1–4:17. Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik. ISBN 978-3-95977-286-0.

Böhm, B.; Peitl, T.; and Beyersdorff, O. 2022a. QCDCL with Cube Learning or Pure Literal Elimination - What is Best? In Raedt, L. D., ed., *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence (IJCAI)*, 1781–1787. ijcai.org.

Böhm, B.; Peitl, T.; and Beyersdorff, O. 2022b. Should Decisions in QCDCL Follow Prefix Order? In Meel, K. S.; and Strichman, O., eds., *25th International Conference on Theory and Applications of Satisfiability Testing (SAT)*, volume 236 of *LIPIcs*, 11:1–11:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Buss, S.; and Nordström, J. 2021. Proof Complexity and SAT Solving. In Biere, A.; Heule, M.; van Maaren, H.; and Walsh, T., eds., *Handbook of Satisfiability*, Frontiers in Artificial Intelligence and Applications, 233–350. IOS Press.

Goultiaeva, A.; and Bacchus, F. 2013. Recovering and Utilizing Partial Duality in QBF. In Järvisalo, M.; and Gelder, A. V., eds., *Theory and Applications of Satisfiability Testing - SAT 2013 - 16th International Conference, Helsinki, Finland, July 8-12, 2013. Proceedings*, volume 7962 of *Lecture Notes in Computer Science*, 83–99. Springer.

Heule, M. J. H.; Seidl, M.; and Biere, A. 2017. Solution Validation and Extraction for QBF Preprocessing. *J. Autom. Reason.*, 58(1): 97–125.

Janota, M. 2016. On Q-Resolution and CDCL QBF Solving. In *Proc. International Conference on Theory and Applications of Satisfiability Testing (SAT)*, 402–418.

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

Kleine Büning, H.; Karpinski, M.; and Flögel, A. 1995. Resolution for Quantified Boolean Formulas. *Inf. Comput.*, 117(1): 12–18.

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

Lonsing, F.; and Egly, U. 2017. DepQBF 6.0: A Search-Based QBF Solver Beyond Traditional QCDCL. In *Proc. International Conference on Automated Deduction (CADE)*, 371–384.

Marques Silva, J. P.; Lynce, I.; and Malik, S. 2021. Conflict-Driven Clause Learning SAT Solvers. In Biere, A.; Heule, M.; van Maaren, H.; and Walsh, T., eds., *Handbook of Satisfiability*, Frontiers in Artificial Intelligence and Applications. IOS Press.

Peitl, T.; Slivovsky, F.; and Szeider, S. 2019. Dependency Learning for QBF. *J. Artif. Intell. Res.*, 65: 180–208.

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

Shukla, A.; Biere, A.; Pulina, L.; and Seidl, M. 2019. A Survey on Applications of Quantified Boolean Formulas. In *Proc. IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, 78–84.

Slivovsky, F. 2022. Quantified CDCL with Universal Resolution. In Meel, K. S.; and Strichman, O., eds., *25th International Conference on Theory and Applications of Satisfiability Testing (SAT)*, volume 236 of *LIPIcs*, 24:1–24:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Tu, K.; Hsu, T.; and Jiang, J. R. 2015. QELL: QBF Reasoning with Extended Clause Learning and Levelized SAT

Solving. In Heule, M.; and Weaver, S. A., eds., *Theory and Applications of Satisfiability Testing - SAT 2015 - 18th International Conference, Austin, TX, USA, September 24-27, 2015, Proceedings*, volume 9340 of *Lecture Notes in Computer Science*, 343–359. Springer.

Van Gelder, A. 2012. Contributions to the Theory of Practical Quantified Boolean Formula Solving. In *Proc. Principles and Practice of Constraint Programming (CP)*, 647–663.

Vardi, M. Y. 2014. Boolean satisfiability: theory and engineering. *Commun. ACM*, 57(3): 5.

Vinyals, M. 2020. Hard Examples for Common Variable Decision Heuristics. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*.

Vinyals, M.; Li, C.; Fleming, N.; Kolokolova, A.; and Ganesh, V. 2023. Limits of CDCL Learning via Merge Resolution. In Mahajan, M.; and Slivovsky, F., eds., *26th International Conference on Theory and Applications of Satisfiability Testing (SAT 2023)*, volume 271 of *Leibniz International Proceedings in Informatics (LIPIcs)*, 27:1–27:19. Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik. ISBN 978-3-95977-286-0.

Wetzler, N.; Heule, M.; and Jr., W. A. H. 2014. DRAT-trim: Efficient Checking and Trimming Using Expressive Clausal Proofs. In Sinz, C.; and Egly, U., eds., *Theory and Applications of Satisfiability Testing (SAT*, volume 8561 of *Lecture Notes in Computer Science*, 422–429. Springer.

Zhang, L. 2006. Solving QBF by Combining Conjunctive and Disjunctive Normal Forms. In *Proceedings, The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference, July 16-20, 2006, Boston, Massachusetts, USA*, 143–150. AAAI Press.

Zhang, L.; and Malik, S. 2002. Conflict driven learning in a quantified Boolean Satisfiability solver. In *Proc. IEEE/ACM International Conference on Computer-aided Design (ICCAD)*, 442–449.