Journal of Artificial Intelligence Research 81 (2024) 741-769 Submitted 10/2023; published 11/2024

QCDCL vs QBF Resolution: Further Insights

Benjamin Böhm BENJAMIN.BOEHM@UNI-JENA.DE
Olaf Beyersdorff OLAF.BEYERSDORFF@UNI-JENA.DE
Friedrich Schiller University Jena, Germany

Abstract

We continue the investigation on the relations of QCDCL and QBF resolution systems. In particular, we introduce QCDCL versions that tightly characterise QU-Resolution and (a slight variant of) long-distance Q-Resolution. We show that most QCDCL variants – parameterised by different policies for decisions, unit propagations and reductions – lead to incomparable systems for almost all choices of these policies.

1. Introduction

SAT solving has revolutionised the way we practically handle computationally complex problems (Vardi, 2014) and emerged as a central tool for numerous applications (Biere et al., 2021). Modern SAT solving crucially relies on the paradigm of conflict-driven clause learning (CDCL) (Marques Silva et al., 2021), on which almost all current SAT solvers are based.

The main theoretical approach to understanding the success of SAT solving (and its limits) comes through proof complexity (Buss & Nordström, 2021). From seminal results (Beame et al., 2004; Pipatsrisawat & Darwiche, 2011; Atserias et al., 2011) we know that CDCL – viewed as a non-deterministic procedure – is exactly as powerful as propositional resolution, which is by far the best-understood propositional proof system (Krajíček, 2019; Buss & Nordström, 2021). However, we also know that practical CDCL using e.g. VSIDS is exponentially weaker than resolution (Vinyals, 2020). Moreover, any deterministic CDCL algorithm will be strictly weaker than resolution unless P = NP (Atserias & Müller, 2019). In any case, the mentioned results of (Beame et al., 2004; Pipatsrisawat & Darwiche, 2011; Atserias et al., 2011) imply that all formulas hard for resolution will be intractable for modern CDCL solvers (at least when disabling preprocessing).

Solving of quantified Boolean formulas (QBF) extends the success of SAT solving to the presumably computationally harder case of deciding QBFs, a PSPACE-complete problem. While QBF solving utilises quite different algorithmic approaches (Beyersdorff et al., 2021), which build on different proof systems, one of the central paradigms again rests on CDCL, lifted to QBFs in form of QCDCL (Zhang & Malik, 2002). In comparison to the propositional case, the main changes are (i) different decision strategies using information from the prefix, (ii) differently implemented unit propagation incorporating universal reductions (i.e., dropping trailing universal variables in clauses), and (iii) adapted methods for learning clauses using a QBF resolution system called long-distance Q-Resolution (Balabanov & Jiang, 2012).

The advances in QBF solving have also stimulated growing research in QBF proof complexity (Beyersdorff et al., 2023; Beyersdorff, 2022; Beyersdorff et al., 2020). As in the propositional case, QBF resolution systems have received great attention. However, in QBF there are a number of conceptually different resolution systems of varying strength (Beyersdorff et al., 2019; Balabanov et al., 2014; Beyersdorff et al., 2021). The core system is Q-Resolution, introduced in 1995 in (Kleine Büning et al., 1995). This system generalises propositional resolution to QBF by using the resolution rule for existential pivots and handling universal variables by universal reduction. A stronger calculus QU-Resolution (Van Gelder, 2012) also allows universal pivots in resolution steps (and this is perhaps the most natural QBF resolution system from a logical perspective (Beyersdorff et al., 2018, 2023)). Yet another generalisation is provided in the form of long-distance Q-Resolution (Balabanov & Jiang, 2012) which allows certain merging steps forbidden in Q-Resolution. Similarly to the simulation of CDCL by resolution, QCDCL traces can be efficiently transformed into long-distance Q-Resolution proofs and this was in fact the reason for creating that proof system.

A recent line of research has aimed at understanding the precise relationship between QCDCL and QBF resolution (Janota, 2016; Beyersdorff & Böhm, 2021; Böhm et al., 2022a, 2022b; Böhm & Beyersdorff, 2021). The findings so far reveal both similarities to the tight relation between CDCL and resolution in SAT as well as crucial differences. While the first work (Janota, 2016) on this topic showed that practical (deterministic) QCDCL is exponentially weaker than Q-Resolution, the paper (Beyersdorff & Böhm, 2021) demonstrated that QCDCL – even in its non-deterministic version – is incomparable to Q-resolution. This also implies that (non-deterministic) QCDCL is exponentially weaker than long-distance Q-Resolution. This is in sharp contrast to the equivalence of SAT and resolution in the propositional case (Beame et al., 2004; Pipatsrisawat & Darwiche, 2011; Atserias et al., 2011), as explained above.

These results were strengthened in (Böhm & Beyersdorff, 2021) by developing a lower-bound technique for QCDCL via a new notion of gauge, by which a number of lower bounds for QCDCL can be demonstrated (which not necessarily hinge on any QBF resolution hardness). Further, (Böhm et al., 2022a, 2022b) showed that several QCDCL variants, utilising e.g. cube learning, pure-literal elimination, and different decision strategies give rise to proof systems of different strength.

1.1 Our Contributions

In this paper we continue this recent line of research to try to understand to precisely determine the relationship of QCDCL variants and different QBF resolution systems. The central quest of our research here is to find different QCDCL variants that are as strong as QU-Resolution and long-distance Q-Resolution. While we do not claim that these new algorithms are of immediate practical interest, we believe it is important to theoretically gauge the full potential of QCDCL. Our results can be summarised as follows.

(a) New QCDCL Versions. We realise that there are at least three crucial QCDCL components that determine the strength of the algorithm. These are (i) whether decisions are made according to the prefix or not (decision policies LEV-ORD or ANY-ORD), (ii) whether unit propagation always or never includes universal reduction (reduction policies ALL-RED, NO-RED) or whether this can be freely chosen at each propagation (ANY-RED), and (iii) whether unit propagation can propagate only existential variables (as in practical QCDCL, propagation policy EXI-PROP) or whether also universal variables can be propagated (ALL-PROP).

While some of these policies were already defined and investigated in earlier works (Beyersdorff & Böhm, 2021; Böhm et al., 2022a, 2022b), the policies ANY-RED and

All-Prop are considered here for the first time. We note that a solver implementing the strategy All-Prop together with Lev-Ord and No-Red was recently presented in (Slivovsky, 2022) (in fact this motivated our definition of the policies Exi-Prop and All-Prop). We demonstrate that in principle, all the aforementioned policies can be combined to yield sound and complete QCDCL algorithms (Proposition 3.9). We denote these as e.g. $\mathsf{QCDCL}^{\textsc{Lev-Ord}}_{\textsc{All-Red},\textsc{Exi-Prop}}$ (this combination models standard QCDCL).

(b) Characterisation of QBF Proof Systems. In our main result we tightly characterise the proof systems QU-Resolution by $\mathsf{QCDCL}^{\textsc{Any-Ord}}_{\textsc{No-Red},\textsc{All-Prop}}$ as well as (a slight variant of) long-distance Q-Resolution by $\mathsf{QCDCL}^{\textsc{Any-Ord}}_{\textsc{Any-Red},\textsc{Exi-Prop}}$ (Proposition 4.10 and Theorem 5.6). These results are similar in spirit (and proof method) to the characterisation of propositional resolution by CDCL (Pipatsrisawat & Darwiche, 2011) and Q-Resolution by $\mathsf{QCDCL}^{\textsc{Any-Ord}}_{\textsc{No-Red},\textsc{Exi-Prop}}$ (Beyersdorff & Böhm, 2021). However, quite some technical care is needed for the simulations to go through with the modified policies, for which we use the new notion of a blockade (Definition 5.3).

The mentioned variant of long-distance Q-Resolution – called mLD-Q-Res (for modified long-distance Q-Resolution, Definition 4.9) – is defined such as to contain exactly those steps that are needed for clause learning in standard QCDCL. The original definition of long-distance Q-resolution also allows some merging steps that do not occur in clause learning (those that have merged literals left of the pivot in both clauses). We leave open whether mLD-Q-Res is indeed weaker or equivalent to long-distance Q-Resolution (cf. Section 6).

(c) Separations between QCDCL Variants. We clarify the joint simulation order of QBF resolution and QCDCL systems (cf. Figure 1 for an overview depicting known and new results). In general, the emerging picture shows that different choices of policies lead to incomparable systems (and could thus in principle be exploited for gains in practical solving over currently used QCDCL, cf. (Slivovsky, 2022; Böhm et al., 2022b)).

One set of results that we highlight concerns the new system $\mathsf{QCDCL}^{\textsc{Lev-Ord}}_{\textsc{Any-Red},\textsc{Exi-Prop}}$, which we show to be strictly stronger than standard QCDCL, yet still weaker than mLD-Q-Res (and incomparable to Q-Resolution). To show that the system is strictly stronger than standard QCDCL (= $\mathsf{QCDCL}^{\textsc{Lev-Ord}}_{\textsc{All-Red},\textsc{Exi-Prop}}$), we exhibit some new family of QBFs which we show to be hard under the All-Red or No-Red policies, yet tractable under Any-Red.

1.2 Organisation

The remainder of this article is organised as follows. We start by reviewing some notions from QBFs and QBF resolution systems in Section 2. In Section 3 we review the existing QCDCL models and define our variants. In Section 4 we investigate the simulation order of the QCDCL proof systems and show various separations. In Section 5 we obtain our main results, the characterisation of the proof systems QU-Res and mLD-Q-Res. We conclude in Section 6 with some open questions.

2. Preliminaries

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

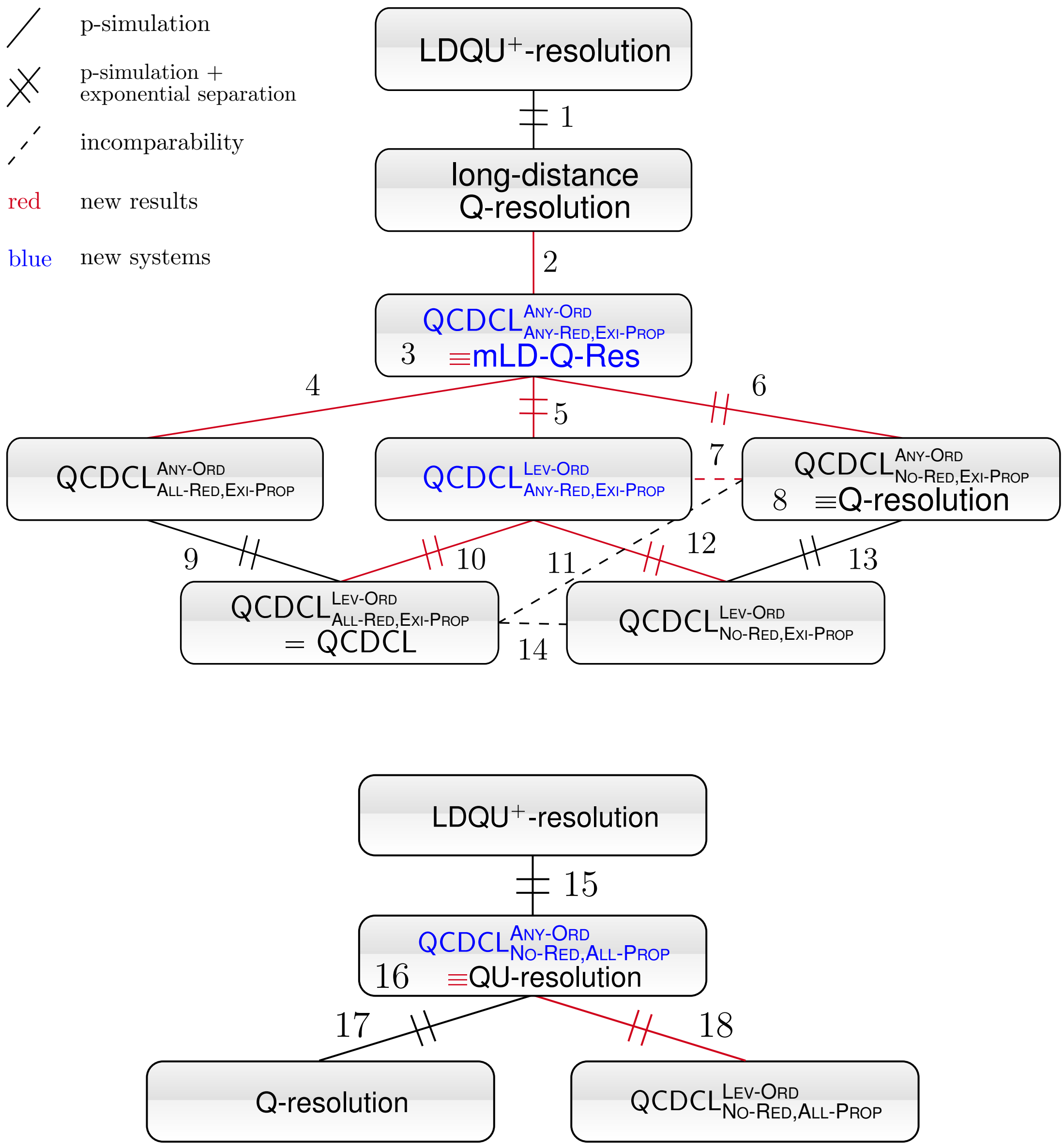

Figure 1: Hasse diagrams of the simulation order of QCDCL proof systems using propagation policies Exi-Prop (above) and All-Prop (below) together with corresponding QBF resolution proof systems. Blue names indicate new systems introduced here. Each relation is labelled by a number that represents a reference depicted in the table in Figure 2.

Number	Reference
1	(Balabanov et al., 2014)
2	Definition 4.9
3	Theorem 5.6
4	Proposition 4.10
5	Propositions 4.3, 4.5, 4.10
6	Corollary 4.11
7	Proposition 4.5
8	(Beyersdorff & Böhm, 2021)
9	(Beyersdorff & Böhm, 2021)
10	Proposition 4.8
11	(Beyersdorff & Böhm, 2021)
12	Proposition 4.8
13	(Beyersdorff & Böhm, 2021)
14	(Beyersdorff & Böhm, 2021), Proposition 4.6
15	(Balabanov et al., 2014), (Beyersdorff et al., 2018)
16	Theorem 5.6
17	(Van Gelder, 2012)
18	Propositions 4.3, 4.5

Figure 2: References for simulations and separations depicted in Figure 1

A *clause* is a disjunction of literals, but we will sometimes interpret them as sets of literals on which we can perform set-theoretic operations. A *unit clause* (ℓ) is a clause that consists of only one literal. The *empty clause* consists of zero literals, denoted $(\bot)$. We sometimes interpret $(\bot)$ as a unit clause with the 'empty literal' $\bot$. A clause C is called *tautological* if $\{\ell, \bar{\ell}\} \subseteq C$ for some literal ℓ. Alternatively, we will sometimes write $\ell^* \in C$ instead of $\{\ell, \bar{\ell}\} \subseteq C$.

A *cube* is a conjunction of literals and can also be 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-contradictory 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 $W \subseteq V$. A clause C is *satisfied* by an assignment σ if $C \cap \sigma \neq \emptyset$. A cube D is *falsified* by σ if $\neg D \cap \sigma \neq \emptyset$. A clause C that is not satisfied by σ can be *restricted* by σ, defined as $C|_\sigma := \bigvee_{\ell \in C, \bar{\ell} \notin \sigma} \ell$. Similarly we can restrict a non-falsified cube D as $D|_\sigma := \bigwedge_{\ell \in D \setminus \sigma} \ell$. Intuitively, an assignment sets all its literals to true.

If L is a set of literals (e.g., an assignment), we can get the negation of L, which we define as $\neg L := \overline{L} := \{\bar{\ell} | \, \ell \in L\}$.

A *CNF* (conjunctive normal form) is a conjunction of clauses and a *DNF* (disjunctive normal form) is a disjunction of cubes. We restrict a CNF ϕ by an assignment σ as $\phi|_\sigma := \bigwedge_{C \in \phi \text{ non-satisfied}} C|_\sigma$. Similarly, we restrict a DNF ϕ by an assignment σ as $\phi|_\sigma := \bigvee_{D \in \phi \text{ non-falsified}} D|_\sigma$. For a CNF (DNF) ϕ and an assignment σ, if $\phi|_\sigma = \emptyset$, then ϕ is *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* $\mathcal{Q} = \mathcal{Q}'_1 V_1 \dots \mathcal{Q}'_s V_s$ consists of non-empty and pairwise disjoint sets of variables $V_1, \dots, V_s$ and quantifiers $\mathcal{Q}'_1, \dots, \mathcal{Q}'_s \in \{\exists, \forall\}$ with $\mathcal{Q}'_i \neq \mathcal{Q}'_{i+1}$ for $i \in \{1, \dots, s-1\}$. 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 $\ell_1 <_\Phi \ell_2$, while $\ell_1 \leqslant_\Phi \ell_2$ means $\ell_1 <_\Phi \ell_2$ or $\ell_1 = \ell_2$.

For a QBF $\Phi = \mathcal{Q} \cdot \phi$ with ϕ a CNF, we call Φ a *QCNF*. We define $\mathfrak{C}(\Phi) := \phi$. The QBF Φ is an *AQBF* (augmented QBF), if $\phi = \psi \vee \chi$ with CNF ψ and DNF χ. Again we write $\mathfrak{C}(\Phi) := \psi$ and $\mathfrak{D}(\Phi) := \chi$. We will sometimes interpret QCNFs as sets of clauses and AQBFs as sets of clauses and cubes. If Φ is a QCNF or AQBF, we define $\text{var}(\Phi) := \bigcup_{C \in \Phi} \text{var}(C)$.

We restrict a QCNF $\Phi = \mathcal{Q} \cdot \phi$ by an assignment σ as $\Phi|_\sigma := \mathcal{Q}|_\sigma \cdot \phi|_\sigma$, where $\mathcal{Q}|_\sigma$ is obtained by deleting all variables from $\mathcal{Q}$ that appear in σ. Analogously, we restrict an AQBF $\Phi = \mathcal{Q} \cdot (\psi \vee \chi)$ as $\Phi|_\sigma := \mathcal{Q}|_\sigma \cdot (\psi|_\sigma \vee \chi|_\sigma)$.

(Long-distance) Q-resolution. Let C_1 and C_2 be two clauses. Let ℓ be an existential 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

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

Let $C := \ell_1 \vee \dots \vee \ell_m$ be a clause from a QCNF or AQBF Φ such that $\ell_i \leqslant_\Phi \ell_j$ for all $i < j$, while $i, j \in \{1, \dots, m\}$. Let k be minimal such that $\ell_k, \dots, \ell_m$ are universal. Then we can perform a *universal reduction* step and obtain

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

If it is clear that C is a clause, we can just write $\text{red}_\Phi(C)$ or even $\text{red}(C)$, if the QBF Φ is also obvious. We will write $\text{red}(\Phi) = \text{red}_\Phi(\Phi)$, if we reduce all clauses of Φ according to its prefix.

We can also perform *partial universal reduction*. Let K be a non-contradictory set of literals and let $C := \ell_1 \vee \dots \vee \ell_m$ be a clause from a QCNF Φ such that

$$\{\ell_k, \dots, \ell_m\} = \{\ell \in C |\ \ell \in K, \ell \text{ is universal and } x <_\Phi \ell \text{ for all existential } x \in C\}.$$

Then we can partially reduce C by K and obtain

$$\text{red}^\forall_{\Phi,K}(C) := \ell_1 \vee \dots \vee \ell_{k-1}.$$

Intuitively, we will reduce all reducible literals that are also contained in K.

As before, we simply write red_K instead of $\text{red}^\forall_{\Phi,K}$ if the context is clear.

As defined by (Kleine Büning et al., 1995), a Q-resolution proof π from a QCNF or AQBF Φ of a clause C is a sequence of clauses $\pi = (C_i)_{i=1}^m$, such that $C_m = C$ and for each C_i one of the following holds:

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

(Balabanov & Jiang, 2012) introduced an extension of Q-resolution proofs to long-distance Q-resolution proofs by replacing the resolution rule by

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

The work (Van Gelder, 2012) presented a further extension for Q-resolution, called QU-resolution, where we can also resolve over universal literals. Formally, it replaces the resolution rule by

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

In (Balabanov et al., 2014), long-distance Q-resolution and QU-resolution were combined into a new proof system: long-distance QU$^+$-resolution. The resolution rule is as follows:

- *Resolution (long-distance QU$^+$-Res):* $C_i = C_j \overset{x}{\bowtie} C_k$ with x existential *or* universal and $j, k < i$. The resolvent C_i is allowed to contain tautologies such as $u \vee \bar{u}$, if u is universal. If there is a such a universal $u \in \text{var}(C_j) \cap \text{var}(C_k)$, then we require $\text{index}(x) < \text{index}(u)$, where index( ) is a fixed total order on the variables of Φ such that $\text{index}(a_1) < \text{index}(a_2)$ whenever $a_1 <_\Phi a_2$ for variables a_1, a_2 of Φ.

A Q-resolution (resp. long-distance Q-resolution, QU-resolution or long-distance QU$^+$-resolution) proof from Φ of the empty clause ($\bot$) is called a *refutation* of Φ. In that case, Φ is called *false*. We will sometimes interpret π as a set of clauses.

For the sake of completeness, we note that the above described proof systems are refutational proof systems that cannot be used to prove the truth of a QBF. For that, we would need analogously defined proof systems that work on cubes instead of clauses. For these proof systems, it is common to use the notion *consensus* instead of resolution, as well as *verification* instead of *refutation*. However, as we will purely concentrate on false formulas in this paper, we omit defining these aspects in more detail.

A proof system P *p-simulates* a system Q, if every Q proof can be transformed in polynomial time into a P proof of the same formula. P and Q are *p-equivalent* (denoted $P \equiv_p Q$) if they p-simulate each other.

3. Our QCDCL Models

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

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

Definition 3.1 (trails)**.** *A* trail $\mathcal{T}$ *for a QCNF or AQBF Φ is a (finite) sequence of pairwise distinct literals from Φ, including the empty literals $\bot$ and $\top$. Each two literals in $\mathcal{T}$ have to correspond to pairwise distinct variables from Φ. In general, a trail has the form*

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

where the d_i *are* decision literals *and* $p_{(i,j)}$ *are* propagated literals*. Decision literals are written in **boldface**. We use a semicolon before each decision to mark the end of a decision level. If one of the empty literals* $\bot$ *or* $\top$ *is contained in* $\mathcal{T}$*, then it has to be the last literal* $p_{(r,g_r)}$*. In this case, we say that* $\mathcal{T}$ *has* run into a conflict*.*

Trails can be interpreted as non-contradictory sets of literals, and therefore as (partial) assignments. We write $x <_{\mathcal{T}} y$ *if* $x, y \in \mathcal{T}$ *and* x *is left of* y *in* $\mathcal{T}$*. Furthermore, we write* $x \leqslant_{\mathcal{T}} y$ *if* $x <_{\mathcal{T}} y$ *or* $x = y$*.*

As trails are produced gradually from left to right in an algorithm, we define $\mathcal{T}[i, j]$ *for* $i \in \{0, \ldots, r\}$ *and* $j \in \{0, \ldots, g_i\}$ *as the subtrail that contains all literals from* $\mathcal{T}$ *up to (and excluding)* $p_{(i,j)}$ *(resp.* d_i*, if* $j = 0$*) in the same order. Intuitively,* $\mathcal{T}[i, j]$ *is the state of the trail before we assigned the literal at the point* $[i, j]$ *(which is* $p_{(i,j)}$ *or* d_i*). We define* $\mathcal{T}[0, 0]$ *as the empty trail.*

For each point $[i, j]$ *in the trail there must exist a set of literals* $K_{(i,j)}$ *which we call the* reductive set at point $[i, j]$*. Intuitively,* $K_{(i,j)}$ *contains all literals that are reduced directly before the point* $[i, j]$*. The sets* $K_{(i,j)}$ *depend on the QCDCL variant (i.e., the reduction policy). Note that these sets are non-empty only if reduction is enabled.*

For each propagated literal $p_{(i,j)} \in \mathcal{T}$ *there has to be be a clause* $ante_{\mathcal{T}}(p_{(i,j)})$ *such that*

$$red_{K_{(i,j)}}(ante_{\mathcal{T}}(p_{(i,j)})|_{\mathcal{T}[i,j]}) = (p_{(i,j)}),$$

or a cube $ante_{\mathcal{T}}(p_{(i,j)})$ *such that*

$$red_{K_{(i,j)}}(ante_{\mathcal{T}}(p_{(i,j)})|_{\mathcal{T}[i,j]}) = [\bar{p}_{(i,j)}].$$

We call such a clause/cube the antecedent clause/cube *of* $p_{(i,j)}$*.*

Remark 3.2. *In classic QCDCL, all* $K_{(i,j)}$ *are set to* $var(\Phi) \cup \overline{var(\Phi)}$*.*

We state some general facts about trails and antecedent clauses/cubes.

Remark 3.3. *Let* $\mathcal{T}$ *be a trail,* $\ell \in \mathcal{T}$ *a propagated literal and* $A := ante_{\mathcal{T}}(\ell)$*.*

- *If* ℓ *is existential, then* $\ell \in A$ *and for each existential literal* $x \in A$ *with* $x \neq \ell$ *we need* $\bar{x} <_{\mathcal{T}} \ell$*.*
- *If* ℓ *is universal, then* $\bar{\ell} \in A$ *and for each universal literal* $u \in A$ *with* $u \neq \bar{\ell}$ *we need* $u <_{\mathcal{T}} \ell$*.*

Definition 3.4 (natural trails)**.** *We call a trail* $\mathcal{T}$ natural *for formula* Φ*, if for each* $i \in \{0, \ldots, r\}$ *the formula* $red_{K_{(i,0)}}(\Phi|_{\mathcal{T}[i,0]})$*, does not contain unit or empty constraints. Furthermore, the formula* $red_{K_{(i,j)}}\Phi|_{\mathcal{T}[i,j]}$ *must not contain empty constraints for each* $i \in \{1, \ldots, r\}$*,* $j \in \{1, \ldots, g_i\}$*, except* $[i, j] = [r, g_r]$*. Intuitively, this means that decisions are only made if there are no more propagations on the same decision level possible. Also, conflicts must be immediately taken care of.*

Remark 3.5. *Although it is allowed to define all sets* $K_{(i,j)}$ *differently, it might make sense from a practical perspective to weaken these possibilities. We point out three nuances of partial reduction in QCDCL that are interesting to consider:*

(i) We change the reductive set after each propagation or decision step. That means that all sets $K_{(i,j)}$ *might be different. This is the strongest possible version of partial reduction.*

(ii) We only update the reductive set after backtracking. That means the sets $K_{(i,j)}$ are constant for each trail. It will turn out that this version is enough for our characterisation of mLD-Q-Res *(cf. Theorem 5.6). Consequently, this version is as strong as version (i).*

(iii) We never change the reductive set. That means that the sets $K_{(i,j)}$ remain constant throughout the whole QCDCL proof. This version is enough for the separation between systems with and systems without partial reduction (cf. Theorem 4.8).

Definition 3.6 (learnable constraints)**.** *Let $\mathcal{T}$ be a trail for Φ of the form* (1) *with $p_{(r,g_r)} \in \{\bot, \top\}$.*

In the case $p_{(r,g_r)} = \bot$, we have detected a clause conflict. Starting with the clause $ante_{\mathcal{T}}(p_{(r,g_r)})$ we reversely resolve with the antecedent clauses until we stop at some point. Literals that were propagated via cubes will be interpreted as decisions.

Analogously, if $p_{(r,g_r)} = \top$, we have detected a cube conflict. Starting with the cube $ante_{\mathcal{T}}(p_{(r,g_r)})$ we reversely resolve with the antecedent cubes until we stop at some point. Literals that were propagated via cubes will be interpreted as decisions.

If a resolution step cannot be performed at some point due to a missing pivot, we simply skip that antecedent. The constraint we so derive is a learnable constraint*. We denote the sequence 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 $\mathfrak{L}(\mathcal{T}) := \{red^{\exists}_{\Phi}(D) | \, D \subseteq \mathcal{T}$ and D satisfies $\mathfrak{C}(\Phi)\}$.

Note that only situations in which clauses are falsified or cubes are satisfied will be referred to as conflicts. In particular, although we can also learn cubes by satisfying all clauses, this is not considered a conflict.

Generally, we allow to learn an arbitrary constraint. However, for the characterisations, it suffices to concentrate on clause learning. Additionally, most of the time we will simply learn the clause which we obtain after propagation over every available literal in the trail. This clause can only consist of negated decision literals, and literals that were reduced during unit propagation. Since this is the last clause we can derive during clause learning in a trail $\mathcal{T}$, we will refer to that clause as the *rightmost clause in $\mathcal{L}(\mathcal{T})$*.

Definition 3.7 (QCDCL proof systems)**.** *Let* $\mathsf{D} \in \{$*LEV-ORD*, *ANY-ORD*$\}$ *a decision policy,* $\mathsf{R} \in \{$*ALL-RED*, *NO-RED*, *ANY-RED*$\}$ *a reduction policy and* $\mathsf{P} \in \{$*EXI-PROP*, *ALL-PROP*$\}$ *a propagation policy (all defined below). A* $\mathsf{QCDCL}^{\mathsf{D}}_{\mathsf{R},\mathsf{P}}$ *proof ι from a QCNF $\Phi = \mathcal{Q} \cdot \phi$ of a clause or cube C is a (finite) 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 for Φ_i that follows the policies D*,* R *and* P*, each $C_i \in \mathfrak{L}(\mathcal{T}_i)$ is one of the constraints we can learn from each trail and π_i is the proof from Φ_i of C_i we obtain by performing the steps described in Definition 3.6, where Φ_i are AQBFs that are defined recursively by setting $\Phi_1 := \mathcal{Q} \cdot (\mathfrak{C}(\Phi) \vee \varnothing)$ and*

$$\Phi_{j+1} := \begin{cases} \mathcal{Q} \cdot ((\mathfrak{C}(\Phi_j) \wedge C_j) \vee \mathfrak{D}(\Phi_j)) & \text{if } C_j \text{ is a clause,} \\ \mathcal{Q} \cdot (\mathfrak{C}(\Phi_j) \vee (\mathfrak{D}(\Phi_j) \vee C_j)) & \text{if } C_j \text{ is a cube,} \end{cases}$$

for $j = 1, \ldots, m-1$. If necessary, we set $\pi_i := \varnothing$.

We now explain the three types of policies:

- *Decision policies:*
  - LEV-ORD: *For each decision d_i we have that $lv_{\Psi|_{\mathcal{T}[i,0]}}(d_i) = 1$. I.e., decisions are level-ordered.*
  - ANY-ORD: *Decisions can be made arbitrarily in any order.*
- *Reduction policies:*
  - ALL-RED: *All $K_{(i,j)}$ are set to $var(\Phi) \cup \overline{var(\Phi)}$. This is the classic setting – we have to reduce all reducible literals during unit propagation.*
  - NO-RED: *All $K_{(i,j)}$ are set to $\emptyset$. We are not allowed to reduce during unit propagation at all. There is one exception: Combined with* ALL-PROP, *we are allowed (but not forced) to reduce universal unit clauses (existential unit cubes) and immediately obtain a conflict. This is due to reasons of completeness which will be explained later.*
  - ANY-RED: *The sets $K_{(i,j)}$ can be set arbitrarily. Hence, we can choose after each propagation or decisions step which literals are to be reduced next.*
- *Propagation policies:*
  - EXI-PROP: *Unit clauses can only propagate existential literals. Universal unit clauses will be reduced to the empty clause if allowed by the reduction policy.*

    Analogously, unit cubes can only propagate universal literals. Existential unit cubes will be reduced to the empty cube if allowed by the reduction policy.
  - ALL-PROP: *Universal unit clauses will lead to the propagation of the universal unit literal. Analogously, existential unit cubes will lead to the propagation of the existential unit literal.*

    This policy is nullified if combined with ALL-RED. *If combined with* NO-RED, *we are allowed to reduce universal unit clauses and existential unit cubes instead of doing a unit propagation. This is due to reasons of completeness.*

Having defined all policies, we can now denote trails that follow the policies D, R *and* P *as* $\mathsf{QCDCL}^{\mathsf{D}}_{\mathsf{R,P}}$ trails.

We require that $\mathcal{T}_1$ is a natural $\mathsf{QCDCL}^{\mathsf{D}}_{\mathsf{R,P}}$ trail and for each $2 \leqslant i \leqslant m$ there is a point $[a_i, b_i]$ such that $\mathcal{T}_i[a_i, b_i] = \mathcal{T}_{i-1}[a_i, b_i]$ and $\mathcal{T}_i \backslash \mathcal{T}_i[a_i, b_i]$ has to be a natural $\mathsf{QCDCL}^{\mathsf{D}}_{\mathsf{R,P}}$ trail for $\Phi_i|_{\mathcal{T}_i[a_i,b_i]}$. This process is called backtracking. *If $\mathcal{T}_{i-1}[a_i, b_i] = \emptyset$, then this is also called a* restart.

If $C = C_m = (\bot)$, then ι is called a $\mathsf{QCDCL}^{\mathsf{D}}_{\mathsf{R,P}}$ refutation *of Φ. If $C = C_m = [\top]$, then ι is called a $\mathsf{QCDCL}^{\mathsf{D}}_{\mathsf{R,P}}$* verification *of Φ. The proof ends once we have learned $(\bot)$ or $[\top]$.*

If C is a clause, we can stick together the long-distance Q-resolution *derivations from $\{\pi_1, \dots, \pi_m\}$ and obtain a* long-distance Q-resolution *proof from Φ of C, which we call $\mathfrak{R}(\iota)$.*

The size *of ι is defined as $|\iota| := \sum_{i=1}^{m} |\mathcal{T}_i|$. Obviously, we have $|\mathfrak{R}(\iota)| \in \mathcal{O}(|\iota|)$.*

Remark 3.8. *In contrast to earlier works, we allow (but not force) a QCDCL solver that uses* NO-RED *together with* ALL-PROP *to reduce universal unit clauses to the empty clause instead of using them for unit propagation. The following example will explain this tweak:*

Consider the QBF $\forall u \cdot (u)$ and assume, we would not be allowed to reduce universal unit clauses. Then we would need to propagate u as this is the only action available. We will not obtain a conflict and therefore learn the cube $[u]$. After backtracking, we must first propagate $\bar{u}$ via $[u]$, followed by a conflict on (u), which allows us to learn the empty clause.

However, if we would have first propagated u via (u), we would have got a cube conflict on $[u]$, from which we would not be able to learn something new. This might lead to unwanted loops, which should be avoided.

Additionally, this tweak ensures the completeness of the corresponding model on false formulas without the necessity to perform cube learning, which would be otherwise a very unnatural property.

We can show that all combinations of the above policies lead to sound and complete proof systems (and algorithms).

Proposition 3.9. *All defined QCDCL variants are sound and complete.*

Proof. It suffices to show completeness for the weakest combinations. Hence, we can use LEV-ORD and choose between ALL-RED and NO-RED, as both are subsumed by ANY-RED. For EXI-PROP, completeness was already shown in (Beyersdorff & Böhm, 2021). For ALL-PROP, we distinguish two cases:

(i) ALL-RED: Then we will never propagate universal literals via clauses, as they will always be directly reduced to the empty clause. Analogously, we will never propagate existential literals via cubes, as they will always be directly reduced to the empty cube. Hence, this system is the same as if we would have chosen EXI-PROP.

(ii) NO-RED: As described in Remark 3.8, we are not forced to do universal propagations via clauses or existential propagations via cubes. Therefore, the version with EXI-PROP is already simulated by this combination system.

The soundness follows from the soundness of long-distance QU^+-resolution (resp. long-distance QU^+-consensus) proofs, which can be extracted from all QCDCL variants defined here. □

While all $2 \times 3 \times 2 = 12$ combinations of policies are sound and complete, we point out that we will not consider all of them. On the one hand, combining the policies ALL-RED and ALL-PROP leads to two QCDCL variants that collapse to the respective versions with EXI-PROP. This is because ALL-RED prevents ALL-PROP from enabling propagations of universal literals via clauses and existential literals via cubes as universal unit clauses and existential unit cubes would be immediately reduced to empty constraints.

On the other hand, we also omit two further combinations using ALL-PROP, namely $\mathsf{QCDCL}^{\textsf{ANY-ORD}}_{\textsf{ANY-RED,ALL-PROP}}$ and $\mathsf{QCDCL}^{\textsf{LEV-ORD}}_{\textsf{ANY-RED,ALL-PROP}}$ since our simulation method does not work for systems with ANY-RED and ALL-PROP. In particular, the proof of Lemma 5.5 below, which is crucial for showing the characterizations, requires that ANY-RED is not combined with ALL-PROP due to some technicalities. We conjecture that the QCDCL variant $\mathsf{QCDCL}^{\textsf{ANY-ORD}}_{\textsf{ANY-RED,ALL-PROP}}$ might characterise a proof system such as LDQU^+-resolution, but we would need a completely different approach.

That said, we consider all remaining eight variants that are included in Figure 1. In particular, we analyse all possible variants with the EXI-PROP policy, which is also the standard policy in practical QCDCL.

4. The Simulation Order of QCDCL Proof Systems

While the policies ALL-RED and NO-RED were already introduced in previous work (Beyersdorff & Böhm, 2021), in which an incomparability between these two models was shown, it is natural to analyse their relation to our new policy ANY-RED. Obviously, ANY-RED covers (hence: simulates) both ALL-RED an NO-RED, as we can simply choose to reduce everything or nothing. We want to prove now that both ALL-RED and NO-RED are exponentially worse than ANY-RED on some family of QBFs. I.e., we want to show that there exist formulas where we need to reduce *some* but not *all* literals during unit propagation.

These formulas will be hand-crafted, consisting of two already well-known QCNFs, named $\texttt{MirrorCR}_n$, which is a modified version of the Completion Principle (Janota & Marques-Silva, 2015), and $\texttt{QParity}_n$ (Beyersdorff et al., 2019).

Definition 4.1 ((Böhm et al., 2022a)). *The QCNF* $\texttt{MirrorCR}_n$ *consists of the prefix*

$$\exists X \forall u \exists T,$$

where $X := \{x_{(1,1)}, \dots, x_{(n,n)}\}$ *and* $T := \{a_1, \dots, a_n, b_1, \dots, b_n\}$*, and the matrix*

$$\begin{array}{lllll} x_{(i,j)} \vee u \vee a_i & \bar{a}_1 \vee \dots \vee \bar{a}_n & x_{(i,j)} \vee \bar{u} \vee \bar{a}_i & a_1 \vee \dots \vee a_n & \\ \bar{x}_{(i,j)} \vee \bar{u} \vee b_j & \bar{b}_1 \vee \dots \vee \bar{b}_n & \bar{x}_{(i,j)} \vee u \vee \bar{b}_j & b_1 \vee \dots \vee b_n & \text{for } i, j \in \{1, \dots, n\}. \end{array}$$

The reason why we use $\texttt{MirrorCR}_n$ instead of $\texttt{CR}_n$ is because its matrix is unsatisfiable. That means that cube learning, which might have a positive effect on $\texttt{CR}_n$ (note that there are false QCNFs that become easy with cube learning (Böhm et al., 2022a)) is now completely unavailable. Additionally, we can now guarantee to always get a conflict once all variables from $\texttt{MirrorCR}_n$ got assigned.

Lemma 4.2 ((Böhm et al., 2022a)). *The matrix* $\mathfrak{C}(\texttt{MirrorCR}_n)$ *of* $\texttt{MirrorCR}_n$ *is unsatisfiable as a propositional formula.*

As $\texttt{MirrorCR}_n$ is simply an extension of the Completion Principle ($\texttt{CR}_n$), which is known to be easy for Q-resolution (Janota & Marques-Silva, 2015), we can simply reuse the exact same refutation from (Janota & Marques-Silva, 2015). Note that we do not need all axiom clauses to refute the formula.

Proposition 4.3 ((Böhm et al., 2022a)). *The QBFs* $\texttt{MirrorCR}_n$ *have polynomial-size* Q-resolution *refutations.*

Definition 4.4 ((Beyersdorff et al., 2019)). *The QCNF* $\texttt{QParity}_n(Y, w, S)$ *consists of the prefix* $\exists Y \forall w \exists S$*, where* $Y := \{y_1, \dots, y_n\}$ *and* $S := \{s_2, \dots, s_n\}$*, and the matrix*

$$\begin{array}{lllll} y_1 \vee y_2 \vee \bar{s}_2 & y_1 \vee \bar{y}_2 \vee s_2 & \bar{y}_1 \vee y_2 \vee s_2 & \bar{y}_1 \vee \bar{y}_2 \vee \bar{s}_2 & \\ y_i \vee s_{i-1} \vee \bar{s}_i & y_i \vee \bar{s}_{i-1} \vee s_i & \bar{y}_i \vee s_{i-1} \vee s_i & \bar{y}_i \vee \bar{s}_{i-1} \vee \bar{s}_i & \text{for } i \in \{2, \dots, n\}, \\ s_n \vee w & \bar{s}_n \vee \bar{w}. & & & \end{array}$$

When introduced in (Böhm et al., 2022a), it was shown that $\texttt{MirrorCR}_n$ is hard for all QCDCL models with level-ordered decisions considered in (Böhm et al., 2022a). We generalize this result and show that the lower bound for $\texttt{MirrorCR}_n$ indeed only depends on the decision policy used and also holds for our new models introduced here.

Proposition 4.5. *The QBFs* $\texttt{MirrorCR}_n(X, u, T)$ *need exponential-sized refutations in all our QCDCL variants with the* ***LEV-ORD*** *policy.*

Proof. In (Böhm et al., 2022a), it was shown that $\texttt{MirrorCR}_n$ is hard for the QCDCL variant $\mathsf{QCDCL}^{\textsf{Lev-Ord}}_{\textsf{All-Red},\textsf{Exi-Prop}}$ by proving that the QCDCL model generates so-called primitive Q-resolution refutations for that formula. This proof is completely independent from the applied reduction policy. Hence, one can easily show that the same result holds for No-Red and Any-Red, as well.

It remains to show that the policy All-Prop does not affect the hardness. This is because QCDCL with Lev-Ord will never be able to propagate universal literals for $\texttt{MirrorCR}_n$.

Assume, for the sake of contradiction, that QCDCL with Lev-Ord propagates a universal literal v in some trail $\mathcal{T}$. W.l.o.g. let this be the first trail and the first time in the trail where this happens. Its antecedent clause $\text{ante}_{\mathcal{T}}(v)$ must contain at least some T-literal ℓ_1, otherwise v would have been reduced away during the learning of $\text{ante}_{\mathcal{T}}(v)$. But then we need $\bar{\ell}_1 \in \mathcal{T}$ and $\bar{\ell}_1$ has to be assigned before v. It could not be done by decision, otherwise this would contradict the Lev-Ord rule. Therefore $\bar{\ell}_1$ was propagated and there exists its antecedent clause $A_1 := \text{ante}_{\mathcal{T}}(\bar{\ell}_1)$.

Since $\texttt{MirrorCR}_n$ fulfils the so-called XT-property (cf. (Böhm et al., 2022a)), A_1 cannot be a unit clause or a clause that consists of X- and T-literals, but no universal literals. Because the only universal variable of $\texttt{MirrorCR}_n$ will be assigned later, and we are not allowed to reduce universal literals for the propagation of T-literals, we conclude that A_1 cannot contain universal literals must therefore contain another T-literal t_2.

We can repeat the argument above and find the antecedent clause $A_2 := \text{ante}_{\mathcal{T}}(\bar{t}_2)$, another T-literal t_3, another clause A_3 and so on. Therefore we would detect an infinite amount of different T-literals, which is obviously impossible.

Hence we have shown that QCDCL cannot propagate universal literals for $\texttt{MirrorCR}_n$ and each QCDCL variant with All-Prop behaves the same as the corresponding version with Exi-Prop on that formula. ☐

With the QBFs $\texttt{QParity}_n$ one obtains one direction of the incomparability between classical QCDCL (here called $\mathsf{QCDCL}^{\textsf{Lev-Ord}}_{\textsf{All-Red},\textsf{Exi-Prop}}$) and Q-resolution, being easy for the former and hard for the latter system.

Theorem 4.6 ((Beyersdorff et al., 2015; Beyersdorff & Böhm, 2021))**.** *The QBFs* $\texttt{QParity}_n$ *need exponential-sized* Q-resolution *and* QU-resolution *refutations, but admit polynomial-sized* $\mathsf{QCDCL}^{\textsf{Lev-Ord}}_{\textsf{All-Red},\textsf{Exi-Prop}}$ *refutations.*

We combine the MirrorCR and QParity formulas into a new one, using auxiliary variables.

Definition 4.7. *The QBF* $\texttt{MiPa}_n$ *consists of the prefix* $\forall z \exists X \forall u \exists T \forall p \exists Y \forall w \exists S \forall v \exists r$ *such that* X, u, T *are the variables for* $\texttt{MirrorCR}_n(X,u,T)$*, and* Y, w, S *are the variables for* $\texttt{QParity}_n(Y,w,S)$*. The matrix of* $\texttt{MiPa}_n$ *contains the clauses*

$$z \vee \bar{r}, \quad \bar{z} \vee \bar{r}$$

$$\left.\begin{array}{l} C \vee p \vee v \vee r \\ C \vee p \vee \bar{v} \vee r \\ C \vee \bar{p} \vee v \vee r \\ C \vee \bar{p} \vee \bar{v} \vee r \end{array}\right\} \textit{for } C \in \mathfrak{C}(\texttt{MirrorCR}_n(X,u,T)),$$

$$\left.\begin{array}{l} p \vee D \\ \bar{p} \vee D \end{array}\right\} \textit{for } D \in \mathfrak{C}(\texttt{QParity}_n(Y,w,S)).$$

We show next that $\texttt{MiPa}_n$ needs indeed ANY-RED in order to admit polynomial-size refutations in QCDCL. The idea is that ALL-RED will always lead to refutations of $\texttt{MirrorCR}_n$, and NO-RED will alternatively lead to Q-resolution refutations of $\texttt{QParity}_n$, which are both of exponential size.

Theorem 4.8. *The QBFs* $\texttt{MiPa}_n$

(i) need exponential-size $\text{QCDCL}^{\textit{LEV-ORD}}_{\textit{ALL-RED,EXI-PROP}}$ *refutations,*

(ii) need exponential-size $\text{QCDCL}^{\textit{LEV-ORD}}_{\textit{NO-RED,EXI-PROP}}$ *refutations,*

(iii) but have polynomial-size $\text{QCDCL}^{\textit{LEV-ORD}}_{\textit{ANY-RED,EXI-PROP}}$ *refutations.*

Proof. For (i), since the formula has no unit clauses, we have to start by deciding the variable z. Because z occurs symmetrically in $\texttt{MiPa}_n$, we can assume that we set z to true. This always triggers the unit propagation of $\bar{r}$ via the clause $\bar{z} \vee \bar{r}$. After that, we are forced to assign the variables from X, $U := \{u\}$ and T along the quantification order. Since the matrix of $\texttt{MirrorCR}_n$ is unsatisfiable, and we need to reduce all literals if possible, we will detect a conflict at the same time as we would get the conflict in $\texttt{MirrorCR}_n$ itself. The proof we can extract from the trails is essentially a $\text{QCDCL}^{\text{LEV-ORD}}_{\text{ALL-RED,EXI-PROP}}$ refutation of $\texttt{MirrorCR}_n$, except that it additionally contains the variables z, p, v and r in some polarities. However, this does not change the fact that we can still not resolve two clauses that contain X-, U-, and T-variables over any X-variable. Therefore, if we shorten the proof by assigning r to false and z to true, we get a refutation of $\texttt{MirrorCR}_n$, in which we never resolve two clauses that contain X-, U-, and T-variables over an X-variable. This property is called *primitive* (cf. (Böhm & Beyersdorff, 2021)). Also in (Böhm & Beyersdorff, 2021), it was shown that primitive Q-resolution refutations of $\texttt{MirrorCR}_n$ need exponential size.

For (ii), we start in the same way as in (i), but we do not get a conflict once we assigned all variables of $\texttt{MirrorCR}_n$. Next, we need to decide p in some polarity, but nothing will happen for the moment. We then start assigning the variables of $\texttt{QParity}_n$ along the quantification order. Now we have to distinguish two cases:

Case 1: We get a conflict in $\texttt{QParity}_n$. But then, because of NO-RED, we can only extract Q-resolution derivations of learned clauses. And if we get enough conflicts in $\texttt{QParity}_n$, we can essentially extract a Q-resolution refutation of $\texttt{QParity}_n$, which has exponential size.

Case 2: We do not get a conflict in $\texttt{QPartity}_n$. This might happen when the universal player assigns the variable w the "wrong" way. Then the only unassigned variable is v. After deciding it in any polarity, we will always get a conflict in $\texttt{MirrorCR}_n$. If we find enough conflicts in $\texttt{MirrorCR}_n$, we can essentially extract an exponential-size fully reduced primitive Q-resolution refutation of $\texttt{MirrorCR}_n$ as in (i).

Note that it is possible to get both kind of conflicts. However, it is only important with what kind of conflicts we were able to derive the empty clause.

Finally, for (iii), we can construct a polynomial-size $\text{QCDCL}^{\text{LEV-ORD}}_{\text{ANY-RED,EXI-PROP}}$ proof by only reducing the literals w and $\bar{w}$. After deciding z, propagating $\bar{r}$, assigning all variables from X, u and T and deciding p arbitrarily, we can simply copy the polynomial-size $\text{QCDCL}^{\text{LEV-ORD}}_{\text{ALL-RED,EXI-PROP}}$ proof of $\texttt{QParity}_n$ (note that ALL-RED only applies to w and $\bar{w}$). At some point, we will derive the clause (p) or $(\bar{p})$, which can be reduced to the empty clause. □

One of the initial motivations of this paper was to find a way to p-simulate long-distance Q-resolution refutations of QCNFs by certain variants of QCDCL. However, it appears that not all resolution steps that are allowed in long-distance Q-resolution can be recreated with QCDCL proofs. In long-distance Q-resolution proofs that are extracted from QCDCL, one can easily observe that for each resolution step $C_1 \overset{\ell}{\bowtie} C_2$, at least one parent clause C_i has to be an antecedent clause for ℓ or $\bar{\ell}$ in the corresponding trail. In particular, there must be a partial assignment τ and a set of literals K such that $\text{red}_K(C_i|_\tau)$ becomes unit, i.e. $\text{red}_K(C_i|_\tau) = (\ell)$ (resp. $(\bar{\ell})$). This is not possible if there are tautologies left of ℓ in C_i that cannot be reduced.

Motivated by this observation, we introduce a new proof system similar to long-distance Q-resolution, but with the restriction that such a situation as described above is not allowed.

Definition 4.9. *A* long-distance Q-resolution *proof is called a* mLD-Q-Res proof*, if it does not contain a resolution step between two clauses D and E, such that $C = D \overset{x}{\bowtie} E$ for an existential variable x and there are universal variables u, w such that $u^* \in D$, $w^* \in E$ and $lv_\Phi(u), lv_\Phi(w) < lv_\Phi(x)$.*

With this definition in place, we can show that mLD-Q-Res proofs can be extracted from runs of most variants of QCDCL that we defined. Further, for some QCDCL paradigms, stricter simulations hold.

Proposition 4.10. *The following holds on false QCNFs:*

(i) Q-resolution *p-simulates* $\text{QCDCL}^{\textsf{Any-Ord}}_{\textsf{No-Red,Exi-Prop}}$.

(ii) QU-resolution *p-simulates* $\text{QCDCL}^{\textsf{Any-Ord}}_{\textsf{No-Red,All-Prop}}$.

(iii) mLD-Q-Res *p-simulates* $\text{QCDCL}^{\textsf{Any-Ord}}_{\textsf{Any-Red,Exi-Prop}}$.

Proof. Item (i) was already shown in (Beyersdorff & Böhm, 2021).

For (ii), because of All-Prop, we might propagate (and resolve) over universal literals, which can be handled by QU-resolution. It remains to show that No-Red prevents the derivation of tautological clauses. This holds because we only use antecedent clauses for clause learning. Let us assume we learn a tautological clause C from a $\text{QCDCL}^{\textsf{Any-Ord}}_{\textsf{No-Red,All-Prop}}$ trail $\mathcal{T}$. Then there would be two antecedent clauses $D := \text{ante}_\mathcal{T}(\ell_1)$ and $E := \text{ante}_\mathcal{T}(\ell_2)$ such that there exists a universal literal u with $u \neq \ell_1$, $\bar{u} \neq \ell_2$, $u \in D$ and $\bar{u} \in E$. We need $\bar{u} \in \mathcal{T}$ for D to become unit and at the same time we need $u \in \mathcal{T}$ for E to become unit, which is not possible. Therefore, we will never derive tautological clauses.

Let us now prove (iii). By definition, we can extract long-distance Q-resolution proof from $\text{QCDCL}^{\textsf{Any-Ord}}_{\textsf{Any-Red,Exi-Prop}}$ trails (note that we only propagate existential literals, hence we also only resolve over existential variables during clause learning). It remains to show that the kind of resolution step that is forbidden in mLD-Q-Res (but allowed in long-distance Q-resolution) will never occur during clause learning.

Assume it does. Then we have derived a clause C by resolving two clauses D and E over some literal x (hence $C = D \overset{x}{\bowtie} E$), such that there exists universal tautologies $u^* \in D$ and $w^* \in E$ with $u^* \neq w^*$ and $\text{lv}(u^*), \text{lv}(w^*) < \text{lv}(x)$. Then at least one of these parent clauses needs to be an antecedent clause for a trail $\mathcal{T}$, say $D = \text{ante}_\mathcal{T}(x)$.

But then D can never become the unit clause (x), because we cannot reduce u^* since it is blocked by x, and we cannot falsify it by the previous trail assignment since it is a tautology. This is a contradiction that shows that all resolution and reduction steps are allowed in mLD-Q-Res.

□

We could formulate analogous results on true QCNFs using the notation of consensus proofs. However, we will omit this as all separations and characterisations will be performed on false QCNFs and resolution proofs.

One can easily show that the separation between Q-resolution and long-distance Q-resolution transfers to a separation between Q-resolution and mLD-Q-Res.

Corollary 4.11. mLD-Q-Res *p-simulates* Q-resolution. *Furthermore,* mLD-Q-Res *is exponentially stronger than* Q-resolution.

Proof. The simulation follows by definition. The separation follows by Theorem 4.6 and Proposition 4.10 (iii). □

In fact, all currently known upper bounds for long-distance Q-resolution can be easily transformed into mLD-Q-Res upper bounds. However, we leave open the question whether long-distance Q-resolution is stronger than or equivalent to mLD-Q-Res.

5. Characterisations of QU-resolution and mLD-Q-Res

In this section, we show that all the simulations in Proposition 4.10 can be tightened to equivalences.

Characterising Q-resolution by $\mathsf{QCDCL}^{\textsc{Any-Ord}}_{\textsc{No-Red},\textsc{Exi-Prop}}$ was already undertaken in (Beyersdorff & Böhm, 2021). Here we characterise both mLD-Q-Res and QU-resolution by the specific variants of QCDCL mentioned in Proposition 4.10. However, we leave open whether we can replace mLD-Q-Res with long-distance Q-resolution in this characterization. This will depend on whether it is possible to polynomially transform the 'forbidden' resolution steps that can occur in long-distance Q-resolution, but cannot be created by QCDCL, into mLD-Q-Res steps.

The characterisations follow the same idea as in (Beyersdorff & Böhm, 2021), in which Q-resolution was characterised. One crucial difference is that we now want to use the Any-Red policy, i.e., in each step we have to decide what literals to reduce.

As already mentioned in Remark 3.5, it suffices to update the reductive sets only after a conflict. That means that for characterising mLD-Q-Res, it is enough to fix the literals that are going to be reduced throughout the whole trail. Thus, we introduce the notion of *L-reductive trails.*

Definition 5.1 (L-reductive trails)**.** *Let L be a set of literals. A trail $\mathcal{T}$ is called* L-reductive*, if for each propagation step in $\mathcal{T}$ the literals that were selected to be reduced are exactly the literals in L. Formally, this means that for each $p_{(i,j)}$ there is an antecedent clause (resp. cube) $ante_{\mathcal{T}}(p_{(i,j)})$ such that $red_L(ante_{\mathcal{T}}(p_{(i,j)})|_{\mathcal{T}[i,j]}) = (p_{(i,j)})$ (resp. $[\bar{p}_{(i,j)}]$).*

Before starting with a new L-reductive trail, we always need to consider the choice of the reductive set L. As we know from (Beyersdorff & Böhm, 2021) and Proposition 4.10, tautologies can only be created when the corresponding literal got reduced somewhere in the trail. In fact, since $\mathsf{QCDCL}^{\textsc{Any-Ord}}_{\textsc{No-Red},\textsc{Exi-Prop}}$ already characterises Q-resolution

(Beyersdorff & Böhm, 2021), we can conclude that in some sense the only purpose of reductions during unit propagation is to create tautological clauses. Therefore we will distinguish between the tautological and the non-tautological part of a clause.

Definition 5.2. *Let C be a clause. Let $G(C) := \{u \in C : u$ is universal and $\bar{u} \in C\}$. This set is the* tautological part *of C. The* non-tautological part *$H(C)$ of C is defined as $H(C) := C \backslash G(C)$.*

For each QU-resolution proof π and $C \in \pi$ we have $G(C) = \emptyset$.

Our next notion is similar to the concepts of *unreliable* (Beyersdorff & Böhm, 2021) and *1-empowering* (Pipatsrisawat & Darwiche, 2011).

Definition 5.3 (Blockades)**.** *Let $\mathsf{S} \in \{\mathsf{QCDCL}^{\textsf{Any-Ord}}_{\textsf{Any-Red,Exi-Prop}}, \mathsf{QCDCL}^{\textsf{Any-Ord}}_{\textsf{No-Red,All-Prop}}\}$ and C be a clause. A tuple $(\mathcal{U}, \alpha, \ell, K)$, where $\mathcal{U}$ is a trail, ℓ is a literal, α is a non-contradictory set of literals and K is a set of universal literals, is called a* blockade of C with respect to S *for a QCNF $\Phi = \mathcal{Q} \cdot \phi$, if $\mathcal{U}$ is a K-reductive* S *trail with decisions α, such that $\ell \in C$, $\alpha \subseteq \overline{C} \backslash \{\bar{\ell}\}$, $K \subseteq G(C)$ and $\alpha \cap K = \emptyset$.*

For $\mathsf{S} = \mathsf{QCDCL}^{\textsf{Any-Ord}}_{\textsf{Any-Red,Exi-Prop}}$, we additionally require that ℓ is an existential literal and α consists of only existential literals.

Example 5.4. *Blockades occur when we are not able to choose all decisions from a pre-defined non-contradictory set α. For example, consider the QCNF*

$$\exists x, y \forall u, v \exists z \; (\bar{y} \vee \bar{z}) \wedge (\bar{x} \vee \bar{u} \vee z) \wedge (x \vee y \vee v \vee z) \wedge (y \vee \bar{v} \vee z).$$

Assume that we use $\mathsf{QCDCL}^{\textsf{Any-Ord}}_{\textsf{Any-Red,All-Prop}}$. Then the clause $C := \bar{x} \vee \bar{y} \vee u \vee \bar{u} \vee z$ has a blockade $(\mathcal{U}, \alpha, \ell, K)$ with $\mathcal{U} := (\mathbf{y}, \bar{z}, \bar{x})$, where $ante_{\mathcal{U}}(\bar{z}) = \bar{y} \vee \bar{z}$, $ante_{\mathcal{U}}(\bar{x}) = \bar{x} \vee \bar{u} \vee z$, as well as $\ell := \bar{x} \in C$, $\alpha := \{y\} \subseteq \overline{C} \backslash \{\bar{\ell}\}$ and $K := \{\bar{u}\}$.

Intuitively, this means that although the clause C is not directly contained in the formula, we are still able to detect the implication $(\alpha \wedge \overline{K} \rightarrow \ell) = (y \wedge u) \rightarrow \bar{x}$ (which is equivalent to $\bar{y} \vee \bar{u} \vee \bar{x} \subseteq C$) as a composition of decisions and unit propagations. It turns out that, instead of learning C directly, it is enough to detect a blockade in order to make use of C for unit propagations in later trails.

The next lemma shows, that we can recall trails (and blockades in particular), that were detected and stored at an earlier point, and restore all propagations they contained. This will be important for the characterisations, as we will go through the given proof, find blockades or conflicts for all clauses in that proof, and recall the corresponding trails (by using this Lemma) an all their containing propagations whenever the clauses are needed for another resolution step. In that way, we can virtually store previous trails and recall them later again as natural trails.

Lemma 5.5. *Let $\Phi = \mathcal{Q} \cdot \phi$ and $\Psi = \mathcal{Q} \cdot \psi$ be QCNFs such that $\psi \subseteq \phi$.*

Let $\mathsf{S} \in \{\mathsf{QCDCL}^{\textsf{Any-Ord}}_{\textsf{Any-Red,Exi-Prop}}, \mathsf{QCDCL}^{\textsf{Any-Ord}}_{\textsf{No-Red,All-Prop}}\}$ and let $\mathcal{U}$ be a K-reductive S *trail (for* **No-Red** *we set $K = \emptyset$) for the QCNF Ψ with decisions β. Let $\mathcal{T}$ be a natural L-reductive* S *trail ($L = \emptyset$ for* **No-Red***) with decisions α for the QCNF Φ such that $K \subseteq L$, $\beta \subseteq \mathcal{T}$ and $\alpha \cap L = \emptyset$. If $\mathcal{T}$ does not run into a clause conflict, then all propagated literals from $\mathcal{U}$ are also contained in $\mathcal{T}$.*

Proof. Assume that $\mathcal{T}$ does not run into a clause conflict, but there are some propagated literals from $\mathcal{U}$ that are not contained in $\mathcal{T}$. Let $p_{(a,b)}$ be the literal that is leftmost in $\mathcal{U}$

with this property and define $A := \text{ante}_{\mathcal{U}}(p_{(a,b)})$. Since there are no cubes present, we conclude that A must be a clause, regardless of whether $p_{(a,b)}$ is existential or universal.

Because $p_{(a,b)}$ is leftmost, all other propagated literals before $p_{(a,b)}$ in $\mathcal{U}$ are already contained in $\mathcal{T}$. Since $\mathcal{U}$ was K-reductive, we know that $\text{red}_K(A|_{\mathcal{U}[a,b]}) = (p_{(a,b)})$. Because of $K \subseteq L$ and $\mathcal{U}[a,b] \subseteq \mathcal{T}$ we have either $\text{red}_L(A|_{\mathcal{T}}) \in \{(p_{(a,b)}), (\bot)\}$, or $A|_{\mathcal{T}}$ becomes true. Note that we can set $K := L := \varnothing$ for the rest of our argumentation in the case where $p_{(a,b)}$ is universal.

The first case would contradict our assumption (since $\mathcal{T}$ is natural), therefore we have to assume that $A|_{\mathcal{T}}$ becomes true. This means that we can find a literal $p_{(a,b)} \neq u \in A \cap \mathcal{T}$. If u was existential, then we would need $\bar{u} \in \mathcal{U}[a,b]$. But this would also imply $\bar{u} \in \mathcal{T}$ which contradicts the fact that $u \in \mathcal{T}$. Hence u must be universal.

If u was a decision in $\mathcal{T}$, then we would have $u \in \alpha$. Because of $\alpha \cap L = \varnothing$ we conclude $u \notin L$ and also $u \notin K$. In order to make u vanish in $\text{red}_K(A|_{\mathcal{U}[a,b]})$, we need $\bar{u} \in \mathcal{U}[a,b]$, hence also $\bar{u} \in \mathcal{T}$. However, this is a contradiction because we already assumed $u \in \mathcal{T}$.

Therefore, u must have been propagated by an antecedent clause $\text{ante}_{\mathcal{T}}(u)$. But then we have $K = \varnothing$, hence $u \notin K$ and $\bar{u} \in \mathcal{U}[a,b] \subseteq \mathcal{T}$, which is a contradiction again because of $u \in \mathcal{T}$. ☐

In the next theorem we will prove the main result: There exist two QCDCL variants that can p-simulate mLD-Q-Res and QU-resolution respectively. The other direction was already proven in Proposition 4.10, therefore we essentially prove the equivalence of these systems.

Theorem 5.6. *The following holds:*

- $\text{QCDCL}^{\text{Any-Ord}}_{\text{Any-Red,Exi-Prop}}$ *p-simulates* mLD-Q-Res.
- $\text{QCDCL}^{\text{Any-Ord}}_{\text{No-Red,All-Prop}}$ *p-simulates* QU-resolution.

In detail: Let $\Phi = \mathcal{Q} \cdot \phi$ be a QCNF in n variables and $\pi = D_1, \ldots, D_m$ be a mLD-Q-Res *(*QU-resolution*) refutation of Φ. Then we can construct a* $\text{QCDCL}^{\text{Any-Ord}}_{\text{Any-Red,Exi-Prop}}$ *(*$\text{QCDCL}^{\text{Any-Ord}}_{\text{No-Red,All-Prop}}$*) refutation ι of Φ with $|\iota| \in \mathcal{O}(n \cdot |\pi|)$. Furthermore, all trails from ι can be constructed such that they run into clause conflicts, meaning that we will only learn clauses in ι.*

Note that the fact that we only learn clauses in the QCDCL proof not only strengthens the result (a possibly weaker QCDCL system suffices for the simulations), but it also simplifies the simulation itself as several auxiliary results below will rely on the assumption that no cubes are learned. Before giving the full proof of Theorem 5.6, which involves the aforementioned auxiliary results, we will sketch the proof idea.

Proof sketch of Theorem 5.6. Going through a given mLD-Q-Res (QU-resolution) refutation π, starting at the axioms, for each $C \in \pi$ we create specific natural trails (where some of them will later be part of the $\text{QCDCL}^{\text{Any-Ord}}_{\text{Any-Red,Exi-Prop}}$ or $\text{QCDCL}^{\text{Any-Ord}}_{\text{No-Red,All-Prop}}$ proof) in which all decisions are negated literals from C, until one of the following events occur:

- We get a conflict and learn a subclause of C.
- We obtain a blockade of C.

When this happens, we either assign the label "subclause" or the label "blockade" to C. When a clause was derived via a resolution or reduction step in π, we simply recall the blockades of its parent clauses by applying Lemma 5.5 to create a blockade for the resolvent or a conflict. If a parent clause does not have a blockade, the clause itself (or a subclause) must have been learned directly and can therefore be used as an antecedent clause for the trail that either becomes a blockade for the resolvent, or that runs into a conflict from which we can learn a subclause of the resolvent.

Since a clause $C \in \pi$ can be derived via resolution (say $C = D \bowtie E$) or reduction (say $C = \text{red}(D)$), we have to consider all possible cases:

(i) resolution, both D and E are labelled "blockade" (cf. Lemma 5.8)

(ii) resolution, D is labelled "blockade", E is labelled "subclause", or vice versa (cf. Lemma 5.9)

(iii) resolution, both D and E are labelled "subclause" (cf. Lemma 5.10)

(iv) reduction, D is labelled "blockade" (cf. Lemma 5.11)

(v) reduction, D is labelled "subclause" (cf. Lemma 5.12)

At the end, each clause in π is either labelled "subclause" or "blockade". In particular, this holds for the empty clause. Because, by definition, there cannot be a blockade of the empty clause (we need at least one literal), the empty clause must be labelled "subclause", which means we have learned the empty clause. □

We now proceed with the full proof of Theorem 5.6. Before doing so, we need a couple of technical lemmas.

The aforementioned creation of specific natural trails is determined by their decisions as well as reductive sets $K_{(i,j)}$. In particular, it is not important in which order propagations are performed as long as they are valid. The following remark explains the way we create these trails in more detail.

Remark 5.7. *Let $\Phi = \mathcal{Q} \cdot \phi$ be a QCNF and L be a set of universal literals. Let α be a non-contradictory set of literals. Then we can construct a trail $\mathcal{T}$ for Φ by choosing α in a specific order as decision literals and propagating literals as soon as the corresponding antecedent clauses become L-reductive unit clauses. We are allowed to update the set L after each propagation. We can even undertake these automatic construction steps after backtracking. However, it is possible that we propagate a literal from α in the same polarity before deciding it. In this case we have to skip the decision. Also, we could reach a conflict before deciding all literals, then we abort the trail as usual.*

If we propagate a literal from $\bar{\alpha}$, then we also abort.

For the next lemmas, we will construct natural trails from a given set α of decision literals. Our goal will to obtain a blockade or a conflict. We will always assume that we start with all existential literals from α before deciding universal literals. In particular, for the simulation of mLD-Q-Res, where we use Exi-Prop instead of All-Prop, all blockades $(\mathcal{U}, \alpha, \ell, K)$ will consist of existential decisions α, but universal reductions K. Therefore we can guarantee $\alpha \cap K = \varnothing$.

The first lemma handles the case where we want to simulate a resolution step between two clauses such that we have already detected blockades for both of them (i.e., both parental clauses are labelled "blockade"). This corresponds to Case (i) from the proof sketch of Theorem 5.6.

Lemma 5.8. *Let $\Phi = \mathcal{Q} \cdot \phi$ be a QCNF. Let further $C \vee x$ and $D \vee \bar{x}$ two clauses such that $C \vee D = C \vee x \overset{x}{\bowtie} D \vee \bar{x}$ is a valid* mLD-Q-Res *(*QU-resolution*) step. Suppose that there exists a blockade of $C \vee x$ for $\Psi = \mathcal{Q} \cdot \psi$ and a blockade of $D \vee \bar{x}$ for $\Gamma = \mathcal{Q} \cdot \gamma$ with $\psi, \gamma \subseteq \phi$. Then there exists a* $\mathsf{QCDCL}^{\textit{Any-Ord}}_{\textit{Any-Red,Exi-Prop}}$ *(*$\mathsf{QCDCL}^{\textit{Any-Ord}}_{\textit{No-Red,All-Prop}}$*) proof*

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

from Φ with a constant c, such that $C_c \subseteq C \vee D$ or there exists a blockade of $C \vee D$ for $\mathcal{Q} \cdot (\phi \cup \{C_1, \ldots, C_c\})$.

Proof. Let the blockade of $C \vee x$ for Ψ be $(\mathcal{U}_1, \alpha_1, \ell_1, K_1)$. Analogously let $(\mathcal{U}_2, \alpha_2, \ell_2, K_2)$ be the blockade of $D \vee \bar{x}$ for Γ with respect to corresponding QCDCL model.

Case 1: $\ell_1 = x$ and $\ell_2 = \bar{x}$.

We construct a natural $G(C \vee D)$-reductive trail $\mathcal{T}$ with decisions $\alpha := \alpha_1 \cup \alpha_2 \subseteq \overline{C \vee D}$. If we receive a blockade, we are done. Note that the set of decisions that were actually made and $G(C \vee D)$ is always disjoint.

If we run into a conflict, then we can start clause learning and learn the rightmost clause E in $\mathcal{L}_{\mathcal{T}}$. Then E can only contain literals from $\bar{\alpha} \subseteq C \vee D$ or $G(C \vee D) \subseteq C \vee D$. In this case we are also done.

Suppose that we do not get a blockade and do not run into a conflict. Then we have $\alpha \subseteq \mathcal{T}$. By Lemma 5.5, each propagation from $\mathcal{U}_1$ as well as $\mathcal{U}_2$ is contained in $\mathcal{T}$. Note that we have $K_1 \cup K_2 \subseteq G(C \vee D)$. But then we would have $x, \bar{x} \in \mathcal{T}$, which is a contradiction.

Case 2: $\ell_1 = x$ and $\ell_2 \neq \bar{x}$ (or analogously $\ell_1 \neq x$ and $\ell_2 = \bar{x}$).

We construct a natural $G(C \vee D)$-reductive trail $\mathcal{T}$ with decisions $\alpha := (\alpha_1 \cup \alpha_2 \cup \{\bar{\ell}_2\}) \backslash \{x\} \subseteq \overline{C \vee D}$ (note that x might be contained in α_2). Similar to Case 1, we are done if we get a blockade or run into a conflict.

Otherwise we would have $\alpha \subseteq \mathcal{T}$. By Lemma 5.5, we conclude $\ell_1 = x \in \mathcal{T}$. This means $\alpha_2 \subseteq \mathcal{T}$. Again, by Lemma 5.5 we would get $\ell_2 \in \mathcal{T}$, which is a contradiction to $\bar{\ell}_2 \in \mathcal{T}$. Hence we always get a blockade or a conflict.

Case 3: $\ell_1 \neq x$ and $\ell_2 \neq \bar{x}$.

W.l.o.g. let C not contain a universal tautology $u \vee \bar{u}$ with $\text{lv}(u) < \text{lv}(x)$. We can make this assumption because the resolution step is valid for mLD-Q-Res (and also for QU-resolution).

We construct a natural K_1-reductive trail $\mathcal{T}$ with decisions $\alpha := \alpha_1 \cup \{\bar{\ell}_1\}$, but we will decide $\bar{x}$ at the end (if x or $\bar{x}$ does not get propagated before). If we run into a conflict without deciding $\bar{x}$, then we can again learn the rightmost clause E in $\mathcal{L}_{\mathcal{T}}$ which is a subclause of C and therefore a subclause of $C \vee D$. Assume we get a blockade of $C \vee x$ with a literal $\ell \in \bar{\alpha} \subseteq C \vee x$. If $\ell \neq x$, then this is a blockade of C (and also a blockade of $C \vee D$) since $\bar{x}$ was not decided, yet. If $\ell = x$, then we have propagated x before deciding $\bar{x}$. But then we can go to Case 2 with the trails $\mathcal{T}$ and $\mathcal{U}_2$.

If we do not get a blockade and do not run into a conflict without deciding $\bar{x}$, and if we actually decide $\bar{x}$ at the end, we will show that we will run into a conflict afterwards. Assume not. Then we have $\alpha_1 \subseteq \alpha \subseteq \mathcal{T}$. By Lemma 5.5, we conclude that $\ell_1 \in \mathcal{T}$, which is a contradiction to $\bar{\ell}_1 \in \alpha \subseteq \mathcal{T}$. Therefore we run into a conflict.

Then we again learn the rightmost clause E in $\mathcal{L}_{\mathcal{T}}$, which is now a subclause of $\bar{\alpha} \vee K_1 \subseteq \bar{\alpha} \vee G(C)$ with $x \in E$ (because $\bar{x}$ was the last decision and the last decision always contributes to the conflict). If $\bar{x}$ was the r^{th} decision, we backtrack back to $\mathcal{T}[r, 0]$ (right before the decision $\bar{x}$ was made). Because $\bar{x}$ was the last decision, we have $\alpha \backslash \{\bar{x}\} \subseteq \mathcal{T}[r, 0]$.

Our precondition at the beginning was that C does not contain a universal tautology left of x. In particular, for all $u \in G(C)$ we have $\mathrm{lv}(u) \geqslant \mathrm{lv}(x)$. We conclude

$$\mathrm{red}_{K_1}(E|_{\mathcal{T}[r,0]}) = (x).$$

Finally, we propagate x, receive the new trail $\mathcal{T}'$ (which is $\mathcal{T}[r, 0]$ plus x) and go into Case 2 again. Note that $\mathcal{T}'$ is still a K_1-reductive trail, even after backtracking.

The number of backtracking steps and restarts are obviously bounded, hence $f_n \in \mathcal{O}(1)$.

Note that for QU-resolution, we construct $\varnothing$-reductive trails because $G(C \vee D) = \varnothing$, which means that we can activate NO-RED. $\square$

In the next lemma, we want to simulate a resolution step between a clause for which we have already detected a blockade, and a second clause that is subsumed by a previously learned clause or an axiom (i.e., one clause is labelled "blockade" while the other is labelled "subclause"). This corresponds to Case (ii) from the proof sketch of Theorem 5.6.

Lemma 5.9. *Let $\Phi = \mathcal{Q} \cdot \phi$ be a QCNF. Let further $C \vee x$ and $D \vee \bar{x}$ two clauses such that $C \vee D = C \vee x \overset{x}{\bowtie} D \vee \bar{x}$ is a valid* mLD-Q-Res *(*QU-resolution*) step. Suppose that there exists a blockade of $C \vee x$ for $\Psi = \mathcal{Q} \cdot \psi$ with $\psi \subseteq \phi$. Suppose also there exists a subclause $D' \subseteq D \vee \bar{x}$ with $D' \in \phi$. Then there exists a* $\mathsf{QCDCL}^{\textsf{ANY-ORD}}_{\textsf{ANY-RED,EXI-PROP}}$ *(*$\mathsf{QCDCL}^{\textsf{ANY-ORD}}_{\textsf{NO-RED,ALL-PROP}}$*) proof*

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

from Φ with a constant c, such that $C_c \subseteq C \vee D$ or there exists a blockade of $C \vee D$ for $\mathcal{Q} \cdot (\phi \cup \{C_1, \ldots, C_c\})$.

Proof. Let the blockade of $C \vee x$ be $(\mathcal{U}_1, \alpha_1, \ell_1, K_1)$.

Case 1: $\ell_1 = x$.

Construct a natural $G(C \vee D)$-reductive trail $\mathcal{T}$ with decisions $\alpha := (\alpha_1 \cup \bar{D}) \backslash G(C \vee D)$ such that we decide existential literals first (again, this is only important for mLD-Q-Res). If we get a blockade, we are done, as for mLD-Q-Res we could have only decided existential literals from $\alpha \cup \bar{D} \subseteq \overline{C \vee D}$. If we run into a conflict, we can learn the rightmost clause E in $\mathcal{L}_{\mathcal{T}}$ which is a subclause of $C \vee D$.

Assume now that we do not get a blockade and not run into a conflict. Then we have $\alpha \subseteq \mathcal{T}$. By Lemma 5.5, all propagations from $\mathcal{U}_1$ are contained in $\mathcal{T}$, in particular $\ell_1 = x \in \mathcal{T}$. Consider the clause

$$A := \mathrm{red}_{G(C \vee D)}(D'|_{\mathcal{T}}).$$

The negations of all literals from $D \backslash G(C \vee D)$ are contained in $\mathcal{T}$. Hence A can only consist of literals from $G(C \vee D)$. But these literals can be reduced away. Therefore

$A = (\bot)$ and we would be able to run into a conflict, which is a contradiction. All in all we run into a conflict or receive a blockade of $C \vee D$.

Case 2: $\ell_1 \neq x$.

Case 2.1: $C \vee x$ does not contain a universal tautology $u \vee \bar{u}$ with $\text{lv}(u) < \text{lv}(x)$ (we are always in this case if we consider QU-resolution).

Note that in this case for all literals $w \in G(C)$ we have $\text{lv}(w) > \text{lv}(x)$. This case is similar to Case 3 of the previous Lemma. We construct a natural $G(C)$-reductive trail $\mathcal{T}$ with decisions $\alpha := \alpha_1 \cup \{\bar{\ell}_1\}$, whereby we decide $\bar{x}$ at the end (if $\bar{x} \in \alpha_1$). If we run into a conflict before deciding $\bar{x}$, we can learn the rightmost clause E in $\mathcal{L}_\mathcal{T}$, which is a subclause of C. Assume we get a blockade with a literal $\ell \in \bar{\alpha} \subseteq C \vee x$. If $\ell \neq x$, then this is a blockade of C and also $C \vee D$. However, if $\ell = x$, then we can go to Case 1 and replace the blockade that consists of $\mathcal{U}_1$ with the blockade consisting of $\mathcal{T}$.

Suppose we do not run into a conflict or get a blockade before deciding $\bar{x}$. If we somehow propagate $\bar{x}$, we have $\alpha_1 \subseteq \alpha \subseteq \mathcal{T}$ and by Lemma 5.5 we conclude $\ell_1 \in \mathcal{T}$. However, this contradicts $\bar{\ell}_1 \in \mathcal{T}$.

By not running into a conflict or getting a blockade before deciding $\bar{x}$, we are able to actually decide $\bar{x}$ at the end. We would run into a conflict by the same argument as above. We learn the rightmost clause E in $\mathcal{L}_\mathcal{T}$, which is a subclause of $\bar{\alpha} \vee G(C)$ with $x \in E$. We backtrack back to $\mathcal{T}[r, 0]$ (right before deciding $\bar{x}$). As above, we have $\alpha \backslash \{\bar{x}\} \subseteq \mathcal{T}[r, 0]$.

By our precondition, we conclude

$$\text{red}_{G(C)}(E|_{\mathcal{T}[r,0]}) = (x).$$

We propagate x and receive another blockade of $C \vee x$ such that we can go into Case 1 again.

Case 2.2: $D \vee \bar{x}$ does not contain a universal tautology $u \vee \bar{u}$ with $\text{lv}(u) < \text{lv}(x)$.

We can assume that we only consider mLD-Q-Res and $\text{QCDCL}^{\text{Any-Ord}}_{\text{Any-Red,Exi-Prop}}$.

Now we have $\text{lv}(v) > \text{lv}(x)$ for all $v \in G(D)$. We construct a natural $G(D)$-reductive trail $\mathcal{T}$ with decisions

$$\alpha := (\alpha_1 \cup \{\bar{\ell}_1\} \cup \overline{H(D)}) \backslash \{\bar{x}\}$$

such that we decide the existential literals first. Note that α is non-contradictory because $\alpha_1 \cup \{\bar{\ell}_1\}$ consists of existential literals only and $\overline{C \vee D} \supseteq \alpha$ can only have universal tautologies. Also, we still have $\alpha \cap L = \varnothing$ because of $G(D) \cap \overline{H(D)} = \varnothing$.

If we run into a conflict or get a blockade, we are done again. Otherwise, we have decided or propagated all literals from α. I.e., $\alpha \subseteq \mathcal{T}$. Consider the clause

$$A := \text{red}_L(D'|_\mathcal{T}).$$

Because of $\bar{x} \notin \overline{H(D)}$, we have $\overline{H(D)} \subseteq \alpha \subseteq \mathcal{T}$ and therefore $A \subseteq G(D) \vee \bar{x}$. By our precondition (all literals from $G(D)$ are right of x), we conclude $A = (\bar{x})$ since we can reduce all universal literals from $D'|_\mathcal{T}$. That means we have to propagate $\bar{x}$ in $\mathcal{T}$, hence $\bar{x} \in \mathcal{T}$. But then we have $\alpha_1 \subseteq \mathcal{T}$. By Lemma 5.5, all propagated literals from $\mathcal{U}_1$ have to be contained in $\mathcal{T}$, in particular $\ell_1 \in \mathcal{T}$. However, this is a contradiction to $\bar{\ell}_1 \in \mathcal{T}$.

That means we always have to run into a conflict or get a blockade, as we desired. ☐

The next lemma covers the last case for the simulation of resolution steps: Both parent clauses are now subsumed by previously learned clauses or axioms (i.e., both clauses are labelled "subclause"). This corresponds to Case (iii) from the proof sketch of Theorem 5.6.

Lemma 5.10. *Let $\Phi = \mathcal{Q} \cdot \phi$ be a QCNF. Let further $C \vee x$ and $D \vee \bar{x}$ two clauses such that $C \vee D = C \vee x \overset{x}{\bowtie} D \vee \bar{x}$ is a valid* mLD-Q-Res *(*QU-resolution*) resolution step. Suppose there exist subclauses $C' \subseteq C \vee x$ and $D' \subseteq D \vee \bar{x}$ with $C', D' \in \phi$. Then there exists a* $\mathsf{QCDCL}^{\textsf{Any-Ord}}_{\textsf{Any-Red,Exi-Prop}}$ *(*$\mathsf{QCDCL}^{\textsf{Any-Ord}}_{\textsf{No-Red,All-Prop}}$*) proof*

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

from Φ with a constant c, such that $C_c \subseteq C \vee D$ or there exists a blockade of $C \vee D$ for $\mathcal{Q} \cdot (\phi \cup \{C_1, \ldots, C_c\})$.

Proof. We construct a natural $G(C \vee D)$-reductive trail $\mathcal{T}$ with decisions $\alpha := \overline{H(C \vee D)}$ such that existential decisions are made first. If we get a blockade or run into a conflict, we are done. So suppose we neither get a blockade, nor run into a conflict. Then we have $\alpha \subseteq \mathcal{T}$. W.l.o.g. let $C \vee x$ not contain a universal tautology $u \vee \bar{u}$ with $\text{lv}(u) < \text{lv}(x)$. Consider the clause

$$A := \text{red}_{G(C \vee D)}(C'|_{\mathcal{T}}).$$

The clause $C'|_{\mathcal{T}}$ can only consist of x or universal literals from $G(C \vee D)$ since the rest got negated by α. We now want to prove that all universal literals in $C'|_{\mathcal{T}}$ can be reduced. In detail, for all universal literals $w \in C'|_{\mathcal{T}}$ we need $\text{lv}(w) > \text{lv}(x)$. Suppose we have a universal literal $v \in C'|_{\mathcal{T}} \subseteq C \vee x$ with $\text{lv}(v) < \text{lv}(x)$. We already concluded that this literal v has to be contained in $G(C \vee D)$. Because we do not have universal tautologies in C left of x, we conclude $\bar{v} \notin C$. But then we need $\bar{v} \in D$ since $v \in G(C \vee D)$. However, such a resolution step is not allowed in mLD-Q-Res (not even in long-distance Q-resolution).

That means all universal literals from $C'|_{\mathcal{T}}$ can be reduced, hence $A \in \{(x), (\bot)\}$. The case $A = (\bot)$ is impossible because we assumed we do not run into a conflict. Therefore $A = (x)$ and we have to propagate x in $\mathcal{T}$. I.e., $x \in \mathcal{T}$.

Now, we consider the clause

$$B := \text{red}_{G(C \vee D)}(D'|_{\mathcal{T}}).$$

Similarly to the situation before, B can only consist of universal literals from $G(C \vee D)$. Note that the $\bar{x}$ that was potentially contained in D' is now vanished. All literals from $D'|_{\mathcal{T}}$ can be reduced, hence $B = (\bot)$. Then $\mathcal{T}$ would run into a conflict, which is a contradiction. ☐

The following lemma shows how we can simulate a reduction step on a clause for which we have detected a blockade (i.e., the clause is labelled "blockade"). This corresponds to Case (iv) from the proof sketch of Theorem 5.6.

Lemma 5.11. *Let $\Phi = \mathcal{Q} \cdot \phi$ be a QCNF. Let $D = C \vee u_1 \vee \ldots \vee u_s$ such that $red(D) = C$ is a valid reduction step in* mLD-Q-Res *(*QU-resolution*). Suppose there exists a blockade of D for $\Psi = \mathcal{Q} \cdot \psi$ with $\psi \subseteq \phi$. Then there exists a* $\mathsf{QCDCL}^{\textsf{Any-Ord}}_{\textsf{Any-Red,Exi-Prop}}$ *(*$\mathsf{QCDCL}^{\textsf{Any-Ord}}_{\textsf{No-Red,All-Prop}}$*) proof*

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

from Φ with a constant c, such that $C_c \subseteq C$ or there exists a blockade of C for $\mathcal{Q} \cdot (\phi \cup \{C_1, \ldots, C_c\})$.

Proof. Let the blockade of D be $(\mathcal{U}_1, \alpha_1, \ell_1, K_1)$. In the case of mLD-Q-Res, we demand that α_1 and ℓ_1 is existential, hence this is also a blockade for C and we are done.

In the case for QU-resolution, we construct a natural K_1-reductive trail $\mathcal{T}$ with decisions $\alpha := \alpha_1 \cup \{\bar{\ell_1}\}$ such that the literals $\bar{u}_i$ are decided last for those contained in α. If we get a conflict, we can learn the clause $\text{red}(\bar{\alpha}) \subseteq \text{red}(D) = C$ and we are done. So suppose we get a blockade $(\mathcal{T}, \beta, \ell, K_1)$ with $\beta \subseteq \alpha$. If $\ell \neq u_i$ for each $i = 1, \ldots, m$, then we also have $\bar{u}_i \notin \beta$ for each i because the $\bar{u}_i$ can only be decided last. But then we have a blockade of C and we are done. However, if $\ell = u_i$ for some $i \in \{1, \ldots m\}$, then instead of propagating u_i, we can simply run into a conflict and learn a subclause of $\text{red}(\bar{\beta} \vee u) \subseteq \text{red}(C \vee u) = C$.

If we neither get a blockade, nor run into a conflict, then we have $\alpha \subseteq \mathcal{T}$ and we can make all propagations from $\mathcal{U}_1$ by Lemma 5.5, hence we get $\beta \in \mathcal{T}$. This is a contradiction to $\bar{\ell} \in \alpha \subseteq \mathcal{T}$. □

The next lemma is the last one before we are able to completely prove Theorem 5.6. It handles the case where we want to simulate a reduction step on a clause D that is subsumed by a previously learned clause or axiom (i.e., the clause is labelled "subclause"). This corresponds to Case (v) from the proof sketch of Theorem 5.6. Note that this case is somewhat special: As learned clauses are already fully universally reduced, there is nothing to show if D is subsumed by a previously learned clause. Hence, we can assume that D is subsumed by an axiom. But then we can also assume that D is an axiom itself, otherwise we could simply shorten the given mLD-Q-Res (QU-resolution) refutation that is to be simulated. In particular, we can assume that D is non-tautological.

Lemma 5.12. *Let $\Phi = \mathcal{Q} \cdot \phi$ be a QCNF. Let $D = C \vee u_1 \vee \ldots \vee u_m$ be non-tautological such that red$(D) = C$. Suppose $D \in \phi$. Then there exists a* $\mathsf{QCDCL}^{\textsf{Any-Ord}}_{\textsf{Any-Red,Exi-Prop}}$ *(resp.* $\mathsf{QCDCL}^{\textsf{Any-Ord}}_{\textsf{No-Red,All-Prop}}$*) proof*

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

from Φ with a constant c, such that $C_c \subseteq C$ or there exists a blockade of C for $\mathcal{Q} \cdot (\phi \cup \{C_1, \ldots, C_c\})$.

Proof. We construct a natural ($\emptyset$-reductive) trail $\mathcal{T}$ with decisions $\alpha = \bar{D}$, such that all the $\bar{u}_i$ are decided last. If we run into a conflict, we can learn a subclause of $\text{red}(\bar{\alpha}) = \text{red}(D) = C$.

Assume that we get a blockade $(\mathcal{T}, \beta, \ell, \emptyset)$. If $\ell \neq u_i$ for each i, then we also have $\bar{u}_i \notin \beta$ for each i because the $\bar{u}_i$ are decided last. In that case, this is also a blockade for C and we are done. In the case where $\ell = u_i$ for some i, we can again simply run into a conflict instead of propagating u_i, hence learning a subclause of $\text{red}(\bar{\alpha}) = \text{red}(D) = C$.

Suppose that none of this occurs. Then we have $\alpha \subseteq \mathcal{T}$. But then we would falsify D, hence we would have the opportunity to run into a conflict with the aid of D, which is a contradiction. □

Finally, we are able to formally prove Theorem 5.6.

Theorem 5.6. *The following holds:*

- $\mathsf{QCDCL}^{\mathsf{Any\text{-}Ord}}_{\mathsf{Any\text{-}Red,Exi\text{-}Prop}}$ *p-simulates* mLD-Q-Res.
- $\mathsf{QCDCL}^{\mathsf{Any\text{-}Ord}}_{\mathsf{No\text{-}Red,All\text{-}Prop}}$ *p-simulates* QU-resolution.

Proof. Let $\mathsf{S} \in \{\mathsf{QCDCL}^{\mathsf{Any\text{-}Ord}}_{\mathsf{Any\text{-}Red,Exi\text{-}Prop}}, \mathsf{QCDCL}^{\mathsf{Any\text{-}Ord}}_{\mathsf{No\text{-}Red,All\text{-}Prop}}\}$. The procedure for both simulations is the same. The plan is going through the whole proof π for each clause D in π either find a blockade, or learn a subclause of D during the construction of S-trails.

Suppose we already considered the clauses $D_1, \ldots, D_{i-1}$ for an $i \in \{1, \ldots, m\}$ and constructed S proofs $\iota_1, \ldots, \iota_{i-1}$. In detail, we have $\iota_j = [(\mathcal{T}^{(j)}_q, C^{(j)}_q, \pi^{(j)}_q)]^{c_j}_{q=1}$ for each $j \in \{1, \ldots, i-1\}$ and some constants c_j. Define

$$\phi_j := \phi \cup \bigcup_{k=1}^{j-1} \{C^{(k)}_1, \ldots, C^{(k)}_{c_k}\}.$$

One could interpret ϕ_j as the knowledge base right before considering the clause D_j in π (and right after going through D_{j-1}, if $j > 1$). In particular, we want to show that if for each $h \in \{1, \ldots, i-1\}$ there exists a blockade of D_h for $\mathcal{Q}\cdot\phi_{h+1}$ or there exists a subclause $D'_h \subseteq D_h$ with $D'_h \in \phi_{h+1}$, then we can construct an S proof $\iota_i = [(\mathcal{T}^{(i)}_q, C^{(i)}_q, \pi^{(i)}_q)]^{c_i}_{q=1}$ from $\mathcal{Q}\cdot\phi_i$ such that there exists a blockade of D_i for $\mathcal{Q}\cdot\phi_{i+1}$ or there exists a subclause $D'_i \subseteq D_i$ with $D'_i \in \phi_{i+1}$. Note that we will only add the trail to our S proof if we learned a clause. The blockade itself will never actually be added to the proof.

If D_i was an axiom (for example if $i = 1$), then we already have a subclause of D_i which is contained in ϕ_i (in fact, D_i itself). In this case we set $\phi_{i+1} := \phi_i$ and do nothing. The proof ι_{i+1} can be defined as the empty proof (or simply left out).

Suppose D_i was the resolvent of two previous clauses D_a and D_b. By induction, we know that

- there exists a blockade of D_a for $\mathcal{Q}\cdot\phi_{a+1}$ or there exists a subclause $D'_a \subseteq D_a$ with $D'_a \in \phi_{a+1}$, and
- there exists a blockade of D_b for $\mathcal{Q}\cdot\phi_{b+1}$ or there exists a subclause $D'_b \subseteq D_b$ with $D'_b \in \phi_{b+1}$.

Each possibility is covered by some earlier Lemma: Lemma 5.8 or Lemma 5.9 or Lemma 5.10. In each case we can construct an S proof ι_i from $\mathcal{Q}\cdot\phi_i$ such that $D'_i := C^{(i)}_{c_i} \subseteq D_i$ or we get a blockade of D_i for $\mathcal{Q}\cdot\phi_{i+1}$. Note that we always have $\phi_{a+1} \subseteq \phi_i$ and $\phi_{b+1} \subseteq \phi_i$.

Now suppose D_i was derived by a reduction of some previous clause D_a, i.e., $D_i = \mathrm{red}(D_a)$. By induction, we either know that

- there exists a blockade of D_a for $\mathcal{Q}\cdot\phi_{a+1}$, or
- there exists a subclause $D'_a \subseteq D_a$ with $D'_a \in \phi_{a+1}$.

The first case is covered by Lemma 5.11. In the second case either D'_a is non-tautological (this case is covered by Lemma 5.12), or D'_a is tautological and hence actually a previously learned clause.

In the latter case we already have $D'_a = \mathrm{red}(D'_a) \subseteq \mathrm{red}(D_a) = D_i$ by the definition of clause learning. Hence we do not have to construct any trails and therefore ι_i can again be viewed as the empty proof.

Finally we construct proofs ι_i as long as we have not learned the empty clause, yet. In the worst case we will learn $(\bot)$ during ι_m since it is impossible to find a blockade of $D_m = (\bot)$ and therefore we will always learn a subclause of D_m.

We receive a refutation ι by sticking together all constructed subproofs ι_i. As usual, we restart between two subproofs ι_i and ι_{i+1}. Note that all ι_i have, by construction, linear size and therefore $|\iota| \in \mathcal{O}(n \cdot |\pi|)$. $\square$

Proposition 4.10 and Theorem 5.6 yield the following characterisations:

Corollary 5.13. *It holds*

(i) $\mathsf{QCDCL}^{\textit{Any-Ord}}_{\textit{Any-Red,Exi-Prop}} \equiv_p$ mLD-Q-Res,

(ii) $\mathsf{QCDCL}^{\textit{Any-Ord}}_{\textit{No-Red,All-Prop}} \equiv_p$ QU-Res.

Remark 5.14. *Note that our simulations require a particular learning scheme, in which we almost always restart after each conflict. This is also the reason why we get an improved simulation complexity of $\mathcal{O}(n \cdot |\pi|)$ compared to $\mathcal{O}(n^3 \cdot |\pi|)$ from (Beyersdorff & Böhm, 2021), in which arbitrary (asserting) learning schemes were allowed (where we do not necessarily restart every time).*

Performing our simulation under arbitrary asserting learning schemes might require some additional analysis on asserting clauses under the Any-Ord *and* Any-Red *rules, as a clause learned from a K_1-reductive trail might not be asserting in K_2-reductive trails anymore. However, if it was clear how to guarantee asserting clauses in our systems, we would be able to obtain similar results as in (Beyersdorff & Böhm, 2021), that is:*

- *For each clause C in the given* mLD-Q-Res *(*QU-resolution*) refutation and an arbitrary asserting learning scheme, we need $\mathcal{O}(n^2)$ trails and backtracking steps until we either learn a subclause of C, or we receive a blockade for C.*
- *Under any arbitrary asserting learning scheme, we can perform the simulation in time $\mathcal{O}(n^3 \cdot |\pi|)$. In particular, we do not need to restart after each conflict.*

6. Conclusion

Proving theoretical characterisations of QCDCL variants successfully used in practice is an important and compelling endeavour. While we contributed to this line of research, a number of open questions remain, both theoretically and practically. In particular, in light of Figure 1, it seems worthwhile to explore whether some of the QCDCL models shown to be theoretically better than standard QCDCL can be used for practical solving.

In our quest to modify QCDCL to match the strength of its underlying system long-distance Q-resolution, we introduced the new proof system mLD-Q-Res, which not only characterises a strong version of QCDCL, but also simulates all related variants. This allows to use proof-theoretic results for mLD-Q-Res whenever considering the strength of QCDCL solvers. Yet, we leave open whether mLD-Q-Res is strictly weaker than or equivalent to long-distance Q-resolution. Both possible outcomes would be interesting, as either long-distance Q-resolution does not characterise QCDCL, or there are modifications of QCDCL that unleash the full strength of long-distance Q-resolution.

Additionally, we exhibited a QCDCL version characterising QU-resolution. One could try to combine these two characterisations to obtain an even stronger family of QCDCL variants in the spirit of LDQU$^+$-resolution. We point out again that the QCDCL variant $\mathsf{QCDCL}^{\text{ANY-ORD}}_{\text{ANY-RED,ALL-PROP}}$ might be a suitable candidate for that.

Further, cube learning, which can hugely impact the running time even on false formulas (Böhm et al., 2022a), was not considered here. Hence, verifying true formulas as well as the proof-theoretic characterisation of modifications to QCDCL such as dependency learning (Peitl et al., 2019) are further topics for future research.

Acknowledgments

An extended abstract of this work appeared in the proceedings of the conference SAT'23 (Böhm & Beyersdorff, 2023). Research was supported by a grant from the Carl-Zeiss Foundation and DFG grant BE 4209/3-1.

References

Atserias, A., Fichte, J. K., & Thurley, M. (2011). Clause-learning algorithms with many restarts and bounded-width resolution. *J. Artif. Intell. Res.*, *40*, 353–373.

Atserias, A., & Müller, M. (2019). Automating resolution is NP-hard. In *IEEE Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 498–509. IEEE Computer Society.

Balabanov, V., & Jiang, J.-H. R. (2012). Unified QBF certification and its applications. *Form. Methods Syst. Des.*, *41*(1), 45–65.

Balabanov, V., Widl, M., & Jiang, J.-H. R. (2014). QBF resolution systems and their proof complexities. In *Proc. Theory and Applications of Satisfiability Testing (SAT)*, pp. 154–169.

Beame, P., Kautz, H. A., & 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., & Schuster, P. (Eds.), *Mathematics for Computation (M4C)*, pp. 353–391. World Scientific.

Beyersdorff, O., Blinkhorn, J., & Hinde, L. (2018). Size, cost, and capacity: A semantic technique for hard random QBFs. In *Proc. Conference on Innovations in Theoretical Computer Science (ITCS'18)*, pp. 9:1–9:18.

Beyersdorff, O., Blinkhorn, J., & Mahajan, M. (2021). Building strategies into QBF proofs. *J. Autom. Reason.*, *65*(1), 125–154.

Beyersdorff, O., Blinkhorn, J., Mahajan, M., & Peitl, T. (2023). Hardness characterisations and size-width lower bounds for QBF resolution. *ACM Trans. Comput. Log.*, *24*(2), 10:1–10:30.

Beyersdorff, O., & Böhm, B. (2021). Understanding the Relative Strength of QBF CDCL Solvers and QBF Resolution. In *12th Innovations in Theoretical Computer Science Conference (ITCS 2021)*, Vol. 185 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 12:1–12:20.

Beyersdorff, O., Bonacina, I., Chew, L., & Pich, J. (2020). Frege systems for quantified Boolean logic. *J. ACM*, *67*(2), 9:1–9:36.

Beyersdorff, O., Chew, L., & Janota, M. (2019). New resolution-based QBF calculi and their proof complexity. *ACM Transactions on Computation Theory*, *11*(4), 26:1–26:42.

Beyersdorff, O., Chew, L., & Janota, M. (2015). Proof complexity of resolution-based QBF calculi. In *Proc. Symposium on Theoretical Aspects of Computer Science (STACS'15)*, pp. 76–89. LIPIcs.

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

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

Böhm, B., & Beyersdorff, O. (2021). Lower bounds for QCDCL via formula gauge. In Li, C.-M., & Manyà, F. (Eds.), *Theory and Applications of Satisfiability Testing (SAT)*, pp. 47–63, Cham. Springer International Publishing.

Böhm, B., & Beyersdorff, O. (2023). QCDCL vs QBF resolution: Further insights. In Mahajan, M., & Slivovsky, F. (Eds.), *26th International Conference on Theory and Applications of Satisfiability Testing (SAT)*, Vol. 271 of *LIPIcs*, pp. 4:1–4:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Böhm, B., Peitl, T., & 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)*, pp. 1781–1787. ijcai.org.

Böhm, B., Peitl, T., & Beyersdorff, O. (2022b). Should decisions in QCDCL follow prefix order?. In Meel, K. S., & Strichman, O. (Eds.), *25th International Conference on Theory and Applications of Satisfiability Testing (SAT)*, Vol. 236 of *LIPIcs*, pp. 11:1–11:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Buss, S., & Nordström, J. (2021). Proof complexity and SAT solving. In Biere, A., Heule, M., van Maaren, H., & Walsh, T. (Eds.), *Handbook of Satisfiability*, Frontiers in Artificial Intelligence and Applications, pp. 233–350. IOS Press.

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

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

Kleine Büning, H., Karpinski, M., & Flögel, A. (1995). Resolution for quantified Boolean formulas. *Inf. Comput.*, *117*(1), 12–18.

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

Marques Silva, J. P., Lynce, I., & Malik, S. (2021). Conflict-driven clause learning SAT solvers. In Biere, A., Heule, M., van Maaren, H., & Walsh, T. (Eds.), *Handbook of Satisfiability*, Frontiers in Artificial Intelligence and Applications. IOS Press.

Peitl, T., Slivovsky, F., & Szeider, S. (2019). Dependency learning for QBF. *J. Artif. Intell. Res.*, *65*, 180–208.

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

Slivovsky, F. (2022). Quantified CDCL with universal resolution. In Meel, K. S., & Strichman, O. (Eds.), *25th International Conference on Theory and Applications of Satisfiability Testing (SAT)*, Vol. 236 of *LIPIcs*, pp. 24:1–24:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Van Gelder, A. (2012). Contributions to the theory of practical quantified Boolean formula solving. In *Proc. Principles and Practice of Constraint Programming (CP)*, pp. 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)*.

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