Electronic Colloquium on Computational Complexity, Revision 1 of Report No. 28 (2016)

Dependency Schemes in QBF Calculi: Semantics and Soundness

Olaf Beyersdorff and Joshua Blinkhorn

School of Computing, University of Leeds, UK

Abstract. We study the parametrisation of QBF resolution calculi by dependency schemes. One of the main problems in this area is to understand for which dependency schemes the resulting calculi are sound. Towards this end we propose a semantic framework for variable independence based on 'exhibition' by QBF models, and use it to express a property of dependency schemes called *full exhibition* that is known to be sufficient for soundness in Q-resolution. Introducing a generalised form of the long-distance resolution rule, we propose a complete parametrisation of classical long-distance Q-resolution, and show that full exhbition remains sufficient for soundness. We demonstrate that our approach applies to the current research frontiers by proving that the reflexive resolution path dependency scheme is fully exhibited.

1 Introduction

The excellent success of SAT solvers in the realm of propositional Boolean formulae has motivated much interest in the corresponding search problem for quantified Boolean formulae (QBF). The greater expressive capacity of QBF, afforded by its PSPACE-completeness [22], presents novel challenges in solving, and the array of emerging techniques is motivating a wealth of research in the closely-related field of proof complexity [3–8, 11–13, 24].

There is a natural correspondence between QBF practice and theory; when a solver concludes the falsity of an instance, the trace can be interpreted as a formal refutation. Understanding the refutational proof system that underpins a particular solving method, and thereby accounts for its correctness, motivates the proof-theoretic study of specific calculi. Recent work has led to a complete understanding of the relative strength of resolution-based QBF systems [3, 6], including Q-resolution (Q-Res) [14], universal Q-resolution (QU-Res) [24], and long-distance Q-resolution (LD-Q-Res) [1].

Implemented in the state-of-the-art solver DepQBF [15,16], one of the recent and exciting developments in QBF solving has seen the introduction of *dependency schemes*: algorithms that gather information on variable independence by prior appeal to the syntactic form of an instance. The quantifier prefix of a QBF (in prenex normal form) imposes a total order on the variables; due to the nesting of quantifier scopes, the value of a Boolean variable z can be dependent upon those values of variables to its left in the prefix. Naturally, this entails some

ISSN 1433-8092

restrictions on solving methods, and on the rules of the related formal systems. In general, however, z does not necessarily depend on all of the variables to its left. A dependency scheme attempts to replace the linear order of the prefix with a partial order that more accurately reflects the dependency structure of the formula, by identifiying such instances of *variable independence*. This approach allows some sets of instances to be solved more effeciently, despite the compuational overhead incurred in computing the dependency scheme [15].

Independence itself is presented as a semantic concept [15, 17]. The truth of a QBF Φ is witnessed by a Skolem-function model, in which each existential x is identified with a Boolean function f_x, such that substitution for each x produces a propositional tautology. The arguments to f_x are the universal variables U_x left of x in the quantifier prefix, but it may occur that some circuit computes f_x without using $u \in U_x$ as an input. In this case we say that *x is independent of u* – and a dual notion for false QBFs provides for independence of universals on existentials – even though the Skolem-function model is in general not unique.

This lack of uniqueness has consequences for soundness in QBF calculi. The impact of a dependency scheme in the proof system is to allow some logical steps which previously were prohibited; specifically, the $\forall$-reduction rule of Q-Res receives greater reign. This motivated the proposal of Q($\mathcal{D}$)-Res by Slivovsky and Szeider [21], a parametrization of the classical calculus by dependency schemes. Some schemes which were previously put forward in the literature, such as the triangle [18] and resolution path [23] dependency schemes, have proved too aggressive for soundness in Q($\mathcal{D}$)-Res, admitting refutations of true QBFs. The reflexive resolution path dependency scheme [21] is currently the strongest known scheme for which Q($\mathcal{D}$)-Res is sound, a result which was proved by means of a difficult transformation of a Q($\mathcal{D}$)-Res refutation into a Q-Res refutation [21].

What is currently absent in the literature is a deeper understanding of soundness based on classification of dependency schemes; moreover, the lack of general methods may frustrate future developments. It is natural to propose the parametrization by dependency schemes of stronger QBF calculi, of the other CDCL-based QBF resolution systems and QBF Frege [4], whereupon methods for proofs of soundness based on *properties* of dependency schemes will carry over. In this paper we demonstrate that semantic notions of independence are indeed equipped for this; our contributions are summarized below.

1. New QBF calculi parametrized by dependency schemes. We extend the parametrisation by dependency schemes to all the CDCL-based resolution calculi for QBF: with the new long-distance calculus LD-Q($\mathcal{D}$)-Res, with universal resolution QU($\mathcal{D}$)-Res, and with their combination LQU($\mathcal{D}$)-Res. Our new long-distance calculus presents the greatest challenge. Of the two inference rules employed classically, parametrization of $\forall$-reduction can be lifted straight from Q($\mathcal{D}$)-Res; here we investigate the additional effects of parametrizing the long-distance resolution rule as well, by relaxing the conditions under which so-called 'merged literals' can be introduced. Progressing from Q-resolution, we demonstrate that variable independence and merging have a more subtle interaction;

in $\mathsf{LD\text{-}Q}(\mathcal{D})\text{-}\mathsf{Res}$, we must supplant merged literals with *annotated literals*, which record existential pivots to prevent unsound ∀-reduction steps.

2. A semantic framework for independence and soundness. We unify some existing approaches in the literature towards a more fruitful understanding of the interplay between Q-resolution and dependency schemes. Building on the work of Samer [17] and Lonsing [15] we propose a semantic framework for variable independence. Central to the framework is a property of dependency schemes called *full exhibition*, which was shown to be sufficient for soundness in $\mathsf{Q}(\mathcal{D})\text{-}\mathsf{Res}$ by Slivovsky [20]. We further the potential of this approach to show that full exhibition is sufficient for soundness in all the dependency calculi we introduce. To that end, we deal with the semantic obstacles of long-distance resolution by incorporating techniques from strategy extraction due by Balabanov et al. [2].

3. Demonstrating full exhibition. We conclude by proving Slivovsky's conjecture [20, p. 37] that the reflexive resolution path dependency scheme $\mathcal{D}^{\mathrm{rrs}}$ is fully exhibited. Currently, $\mathcal{D}^{\mathrm{rrs}}$ is arguably the most important dependency scheme, capable of revealing more cases of independence than any other tractable scheme known to be sound for $\mathsf{Q}(\mathcal{D})\text{-}\mathsf{Res}$. As such, we show that everything currently known about soundness in this setting can be explained by full exhibition. On a technical level, the result is obtained by means of an algorithmic transformation of an arbitrary model for a true QBF Φ into a model that exhibits all the required independencies. We therefore reveal the possibility for QBF solving to implement long-distance techniques fully parametrized by $\mathcal{D}^{\mathrm{rrs}}$, or any other fully exhibited scheme.

Organisation of the paper. After providing the necessary fundamentals in Section 2, we present our semantic framework based on 'exhibition' in Section 3. In Section 4, we present the new long-distance calculus and corresponding soundness results, while Section 5 covers the proof that $\mathcal{D}^{\mathrm{rrs}}$ is fully-exhibited. Finally, some conclusions are offered in Section 6.

2 Preliminaries

Quantified Boolean Formulas. A *Quantified Boolean Formula* (QBF) Φ over a set $V = \{z_1, \ldots, z_n\}$ of n variables is a formula in quantified Boolean logic with quantifiers ranging over $\{0, 1\}$. We consider only formulas in *prenex conjunctive normal form* (PCNF), denoted $\Phi = \mathcal{Q} \,.\, \phi$, in which all variables are quantified either existentially or universally in the *quantifier prefix* $\mathcal{Q} = \mathcal{Q}_1 z_1 \cdots \mathcal{Q}_n z_n$, $\mathcal{Q}_i \in \{\exists, \forall\}$ for $i \in [n]$, and ϕ is a propositional conjunctive normal form (CNF) formula called the *matrix*. A CNF matrix is a conjunction of clauses, each clause is a disjunction of literals, and a literal is a variable or its negation. Whenever convenient, we refer to a clause as a set of literals and to a matrix as a set of clauses. We typically write x for existential variables, u and v for universals, and z for either. We denote the sets of existentially and universally quantified variables of Φ by $V_\exists = \{z_i \in V \mid \mathcal{Q}_i = \exists\}$ and $V_\forall = \{z_i \in V \mid \mathcal{Q}_i = \forall\}$ respectively. The prefix $\mathcal{Q}$ imposes a linear ordering on the variables of Φ that is captured by the binary relation $<_\Phi$, such that $z_i <_\Phi z_j$ holds whenever $i < j$, in which case we say that z_j *is right of* z_i, or that z_i *is left of* z_j. The sets of variables right and left of z are denoted $R_\Phi(z) = \{z' \in V \mid z <_\Phi z'\}$ and $L_\Phi(z) = \{z' \in V \mid z' <_\Phi z\}$.

Assignment Trees and Models. Assignment trees for PCNF were first introduced in [19]. We represent an assignment tree formally as a set of paths. Let Φ be a PCNF over variables $V = \{z_1, \dots, z_n\}$ and let $V_\forall = \{u_1, \dots, u_k\}$. A path is a set of literals $P = \{l_1, \dots, l_n\}$ with $\mathrm{var}(l_i) = z_i$ for all $i \in [n]$, and we write $P[z_i] = l_i$. If T is a set of paths with $P, Q \in T$, T is well-formed for Φ iff (1) for all $u \in V_\forall$, if $P[v] = Q[v]$ for all $v \in L_\Phi(u) \cap V_\forall$, then $P[x] = Q[x]$ for each $x \in L_\Phi(u) \cap V_\exists$, and (2) there is a unique path $P \in T$ with $U \subseteq P$ for each set of literals $U = \{l_1, \dots, l_k\}$ such that $\mathrm{var}(l_i) = u_i$ for $i \in [k]$. A set of paths that is well-formed for Φ is an *assignment tree* for Φ. We also use P to denote the total assignment $P : V \to \{\top, \bot\}$ given by $P(z_i) = \bot$ if $l_i = \neg z_i$ and $P(z_i) = \top$ if $l_i = z_i$, and extend this notation to literals with $P(\neg z_i) = \neg P(z_i)$, where $\top = \neg\bot$ and vice versa. An assignment tree for Φ is a *model* for Φ, typically denoted M, iff $P(C) = \top$ for all paths $P \in T$ and all clauses $C \in \phi$, where $P(C) = \top$ iff $P(l) = \top$ for some $l \in C$. A PCNF which has a model is *true*, otherwise it is *false*. An assignment tree is depicted as a tree with root r, as shown in Fig 1.

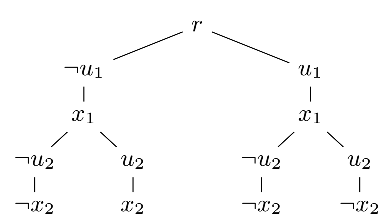

Fig. 1. An assignment tree T for a PCNF $\forall u_1 \exists x_1 \forall u_2 \exists x_2 \,.\, \phi$, with arbitrary matrix ϕ.

Dependency Schemes. The trivial dependency scheme $\mathcal{D}^{\mathrm{trv}}$ is a mapping which associates each PCNF $\Phi = \mathcal{Q}_1 z_1 \cdots \mathcal{Q}_n z_n \,.\, \phi$ over variables V to the trivial dependency relation $\mathcal{D}^{\mathrm{trv}}_\Phi = \{(z_i, z_j) \mid i < j \text{ and } \mathcal{Q}_i \neq Q_j\}$. A *proto-dependency scheme*[1] $\mathcal{D}$ is a function that maps each PCNF Φ to a binary relation $\mathcal{D}_\Phi \subseteq \mathcal{D}^{\mathrm{trv}}_\Phi$

[1] The term 'dependency scheme' was first introduced to denote a subset of proto-dependency schemes with a more technical definition [18]; for consistency with the literature we will use 'proto-dependency scheme' in technical portions of this paper.

called the *dependency relation*. If $(z_i, z_j) \in \mathcal{D}_\Phi$, then (z_i, z_j) is a $\mathcal{D}$-dependency and z_j is a $\mathcal{D}$-dependent of z_i, otherwise z_j is $\mathcal{D}$-independent of z_i. A proto-dependency scheme $\mathcal{D}'$ is said to be *at least as general* as another $\mathcal{D}$ if $\mathcal{D}'_\Phi \subseteq \mathcal{D}_\Phi$ for all PCNFs Φ, and is *strictly more general* if the inclusion is strict for some formula.

QBF Resolution Calculi. We give a brief overview of four resolution-based CDCL QBF calculi – see [6] for a more detailed survey. Their formal definitions are presented in Sections 3 and 4 as special cases of the corresponding 'dependency' systems. A refutational QBF calculus is *sound* iff the empty clause cannot be derived from any true formula, and is *complete*.

Q-resolution (Q-Res) introduced in [14] is the standard refutational calculus for PCNF. In addition to resolution over existential pivots with non-tautologous resolvents, the calculus has a universal reduction rule which allows a clause C to be derived from $C \cup \{u\}$, where u is a universal literal and all existential literals in C are left of u. *QU-resolution* (QU-Res) [24] is a natural extension of Q-Res that allows universal resolution pivots.

Long-distance resolution, which was introduced in [25] and formalised as the calculus LD-Q-Res [1], allows tautologous resolvents under certain conditions, using the special merged literal u^* to represent the tautology $\{u, \neg u\}$. The resulting system is exponentially stronger than Q-Res [12]. Finally, the calculus LQU-Res [3] combines naturally the features of QU-Res and LD-Q-Res, allowing merged literals and resolution over universal pivots.

3 Dependency Schemes, Q-resolution and Semantics

3.1 Dependency Schemes and Q-resolution

It is natural to try to strengthen a classical QBF calculus using a dependency scheme, and the starting point for the 'dependency version' is the identification of the trivial dependency relation in the rules of a calculus. Restrictions are inevitably imposed by the linear ordering of the quantifier prefix; dependency calculi can relax these restrictions, replacing the implicit reference to $\mathcal{D}^{\text{trv}}$ with an explicit reference to a more general dependency scheme $\mathcal{D}$.

Figure 2 recalls the rules of Q($\mathcal{D}$)-Res, the dependency version of Q-Res, introduced in [21] to account for the behaviour of the QDPLL-based solver DepQBF [9, 15]. In Q-Res, the universal reduction rule allows a universal u to be dropped from a clause C containing only existential variables left of u. By comparison, Q($\mathcal{D}$)-Res allows u to be dropped whenever C contains no $\mathcal{D}$-dependents of u. Note that Q-Res and Q($\mathcal{D}^{\text{trv}}$)-Res are identical. Whether or not Q($\mathcal{D}$)-Res is sound depends on the strength of the dependency scheme. For example, in [21] it is shown that Q($\mathcal{D}$)-Res is sound for $\mathcal{D}^{\text{rrs}}$, but unsound for the strictly more general scheme $\mathcal{D}^{\text{res}}$.

It is natural to extend Q($\mathcal{D}$)-Res by allowing resolution over universal pivots. The resulting new system QU($\mathcal{D}$)-Res, also presented in Fig. 2, is the dependency version of QU-Res.

$\dfrac{}{C}$ (Axiom)	C is a clause in the matrix of Φ.
$\dfrac{D \cup \{l_u\}}{D}$ ($\forall$-Red)	Literal l_u is universal. If $l \in D$ and $\mathrm{var}(l) = v$, then $(u, v) \notin \mathcal{D}_{\Phi}$.
$\dfrac{C_1 \cup \{v\} \qquad C_2 \cup \{\neg v\}}{C_1 \cup C_2}$ (Res)	If $l \in C_1$, then $\neg l \notin C_2$. In $\mathsf{Q}(\mathcal{D})$-Res, variable v is existential; in $\mathsf{QU}(\mathcal{D})$-Res, v is existential or universal.

Fig. 2. The rules of $\mathsf{Q}(\mathcal{D})$-Res [21] and $\mathsf{QU}(\mathcal{D})$-Res

3.2 A Semantic Framework for Independence

We reformulate the definition of independence in terms of assignment trees from [15, 17]; we feel our notation is better suited to the aims of the current work. We first introduce the new idea of *complementary paths* in an assignment tree, whose universal literals differ for exactly one variable.

Definition 1 (Complementary path). *Let Φ be a QBF over variables V, let U be a non-tautologous set of literals such that $\mathrm{var}(U) = V_\forall$, let T be an assignment tree for Φ and let $P \in T$ be the unique path such that $U \subseteq P$. Then, for any $u \in V_\forall$, $P_u \in P$ is the unique path such that $U' \subseteq T$, where $U' = (U \setminus \{l\}) \cup \{\neg l\}$, $l \in U$ and $\mathrm{var}(l) = u$.*

It is fortunate that, throughout this paper, we need only consider the dependence of existentials on universals; this simplification, seen in the definition below, is the result of our dealing exclusively with refutational calculi, the rules of which remain unaffected by the dependence (or lack thereof) of universals on existentials.

Definition 2 (Independence of existentials from universals [15,17]). *Let Φ be a true QBF over variables V and let $u \in V_\forall$, $x \in V_\exists$. We say that x is independent of u in Φ if there exists a model M for Φ in which $P(x) = P_u(x)$ for all paths $P \in M$. For such a model M we write $M \prec (u, x)$, and we say that M* exhibits the independence *of x from u in Φ.*

Remark 3. It is not necessary for us to consider false QBFs in Definition 2, since, by definition, a false formula has no models. For a false formula, the condition for independence is satisfied vacuously, confirming our intuition that no existential variable can be dependent on any universal in such a formula.

As noted in [21], Definition 2 alone is too weak for soundness in $\mathsf{Q}(\mathcal{D})$-Res. The problem lies in the possibility for different models to exhibit different independencies, which are then used together in the same refutation. It is therefore natural to seek a model which exhibits all the independencies that may be used in a refutation.

Definition 4 (Fully exhibited dependency scheme). *Let $\mathcal{D}$ be a proto-dependency scheme. We say that $\mathcal{D}$ is* fully exhibited *iff for each true PCNF Φ there is a model M for Φ such that $M \prec (u,x)$ for each pair $(u,x) \notin D_\Phi$, with $u \in V_\forall$ and $x \in V_\exists$.*

In [20], it was proved that Q($\mathcal{D}$)-Res is sound for fully exhibited[2] $\mathcal{D}$, and this was combined with the fact that the standard dependency scheme $\mathcal{D}^{\text{std}}$ is fully exhibited (attributed to [10]). In the next section, we show that this approach scales up to the dependency versions of stronger QBF calculi.

4 Dependency Schemes and Long-Distance Q-resolution

In this section, we introduce the new long-distance calculi LD-Q($\mathcal{D}$)-Res and LQU($\mathcal{D}$)-Res, the respective dependency versions of LD-Q-Res and LQU-Res.

Long-distance Q-resolution was formalised as a calculus [1] to account for solving techniques due to [25], and the resulting system is exponentially stronger than Q-Res [12]. The salient feature of the system is that tautological clauses are allowed under certain conditions. Specifically, resolving clauses C_1 and C_2 over an existential pivot x, a 'merged literal' u^* appears in the resolvent clause C if $\neg u \in C_1$, $u \in C_2$ and $x <_\Phi u$. In successive resolution steps, a merged literal u^* may be merged again with another merged literal u^*, or with non-merged literals u and $\neg u$, provided that the existential pivot is left of u. Both merged and non-merged literals may be dropped from a clause by $\forall$-reduction under the usual conditions.

Whereas the parametrisation of $\forall$-reduction can be lifted directly from Q($\mathcal{D}$)-Res, parametrisation of long-distance resolution, which relaxes the conditions under which merging is allowed, presents a novel challenge.

4.1 Defining LD-Q($\mathcal{D}$)-Res and LQU($\mathcal{D}$)-Res

Although the method of generalising the reference to the trivial dependency scheme remains, more care must be taken when defining LD-Q($\mathcal{D}$)-Res. Parametrising long-distance resolution means relaxing the conditions under which merging may take place, which in turn entails some new notation. Replacing $x <_\Phi u$ with the condition $(u,x) \notin \mathcal{D}_\Phi$ in the dependency version, one must annotate merged literals with the corresponding *pivot set* X, producing an *annotated literal* $u^X \in C$, where X consists of all the existential variables over which u has been merged in the derivation of the clause C. Annotations are needed to keep track of the pivot sets to prevent unsound $\forall$-reduction steps; this will shortly be explained in greater detail.

The rules of LD-Q($\mathcal{D}$)-Res are given in Fig. 3. We observe that LD-Q($\mathcal{D}^{\text{trv}}$)-Res is precisely the classical long-distance calculus LD-Q-Res, except that the merged literals of the latter are annotated. Since the dependency conditions of LD-Q($\mathcal{D}$)-Res are identical to the classical long-distance conditions if $\mathcal{D}$ is $\mathcal{D}^{\text{trv}}$, replacing

[2] Full exhibition is treated equivalently, as a property of models.

all annotated literals u^X in an LD-Q($\mathcal{D}^{\mathsf{trv}}$)-Res refutation with merged literals u^* produces an LD-Q-Res refutation, and vice versa – replacing all merged literals u^* in an LD-Q-Res refutation with annotated literals u^X produces an LD-Q($\mathcal{D}^{\mathsf{trv}}$)-Res refutation. Similarly as for Q($\mathcal{D}$)-Res, it is natural to extend LD-Q($\mathcal{D}$)-Res by allowing resolution over universal pivots. The resulting new system LQU($\mathcal{D}$)-Res, also given in Figure 3, is the dependency version of LQU-Res.

$$\frac{}{C}\ \text{(Axiom)}$$

C is a clause in the matrix of Φ.

$$\frac{D \cup \{u^X\}}{D}\ (\forall\text{-Red})$$

Varaible u is universal. If $l \in D$ and $\mathrm{var}(l) = z$, then $(u, z) \notin \mathcal{D}_\Phi$, and if $l = z^{X'}$ then $(u, x) \notin \mathcal{D}_\Phi$ for all $x \in X'$. If $X = \emptyset$ then literal u^X is either u or $\neg u$.

$$\frac{C_1 \cup U_1 \cup \{x\} \qquad C_2 \cup U_2 \cup \{\neg x\}}{C_1 \cup C_2 \cup U}\ \text{(Res)}$$

If for $l_1 \in C_1, l_2 \in C_2, \mathrm{var}(l_1) = \mathrm{var}(l_2)$, then $l_1 = l_2$ is not annotated. $\mathrm{var}(U_1) = \mathrm{var}(U_2) \subseteq V_\forall$, and $(x, u) \notin \mathcal{D}_\Phi$ for each $u \in \mathrm{var}(U_1)$. If for $u_1 \in U_1, u_2 \in U_2, \mathrm{var}(u_1) = \mathrm{var}(u_2) = u$, then $u_1 = \neg u_2$, or at least one of u_1, u_2 is annotated. U is defined as $\{u^X \mid u \in \mathrm{var}(U_1)\}$, where X is the union of $\{x\}$ with any annotations on u in $U_1 \cup U_2$. In LD-Q($\mathcal{D}$)-Res $\mathrm{var}(x)$ is existential. In LQU($\mathcal{D}$)-Res, $\mathrm{var}(x)$ is existential or universal.

Fig. 3. The rules of LD-Q($\mathcal{D}$)-Res

The purpose of annotating literals is to prevent unsound $\forall$-reduction steps, by checking that the pivot sets in the clause are $\mathcal{D}$-independent of the reduced universal variable. Annotations were never necessary in LD-Q-Res; the fact that a merged literal u^* in the clause is always right of its corresponding existential pivots is enough to ensure soundness. However, in LD-Q($\mathcal{D}$)-Res, we must explicitly forbid $\forall$-reduction of $v \in C$ if any $x \in X$ is not $\mathcal{D}$-independent of v, for any annotation X in the clause C. The following example shows that allowing v to be reduced under such conditions is unsound in general for a fully-exhibited proto-dependency scheme $\mathcal{D}$.

Example 5. Take the true QBF $\Psi = \forall u \exists x_1 \forall v \exists x_2 \exists x_3 \,.\, \phi$ with matrix

$$\phi = \{u, x_2, \neg x_3\}, \{\neg u, \neg x_2, \neg x_3\}, \{x_1, v, x_3\}, \{\neg x_1, \neg v, x_3\}$$

and the proto-dependency scheme $\mathcal{D}'$ defined by

$$\mathcal{D}'_{\Phi} = \begin{cases} \{(u,x_1),(v,x_2),(u,x_3),(v,x_3)\} & \text{if } \Phi = \Psi\,, \\ \mathcal{D}^{\text{trv}}_{\Phi} & \text{otherwise}\,. \end{cases}$$

First observe that $\mathcal{D}'$ is fully exhibited; Figure 4 depicts a model M for Ψ which exhibits the independence of x_2 on u.

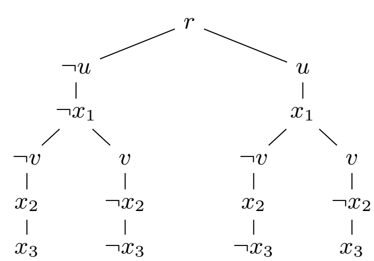

Fig. 4. A model M for Ψ for which $M \prec (u, x_2)$.

However, if we allow variable v to the reduced alonside the annotated literal $u^{\{x_2\}}$, noting that x_2 is not $\mathcal{D}'$-independent of v, we obtain the following refutation of Ψ.

$$\dfrac{\dfrac{\dfrac{\dfrac{\{u,x_2,\neg x_3\} \qquad \{\neg u,\neg x_2,\neg x_3\}}{\{u^{\{x_2\}},\neg x_3\}} \qquad \dfrac{\{x_1,v,x_3\} \qquad \{\neg x_1,\neg v,x_3\}}{\{v^{\{x_1\}},x_3\}}}{\{u^{\{x_2\}},v^{\{x_1\}}\}}}{\{u^{\{x_2\}}\}}}{\bot}$$

4.2 Soundness of LD-Q($\mathcal{D}$)-Res and LQU($\mathcal{D}$)-Res

In this subsection, we will prove that LD-Q($\mathcal{D}$)-Res is sound for a fully exhibited $\mathcal{D}$, and our method entails the following evaluation of annotated literals under assignment. We define *annotated literal functions*, which are based on the 'phase functions' and 'effective literals' introduced by Balabanov et al. [2].

Informally, it is demonstrated in [2] that any assignment σ to the existential variables 'induces' the phase of a merged literal u^* in an LD-Q-Res refutation of a PCNF Φ, such that for the purpose of strategy extraction it may be interpreted as either non-merged literal u or $\neg u$. In a given model M for Φ, every path P contains a particular assignment to the existential variables. Therefore, for any annotated literal u^X in a some LD-Q($\mathcal{D}$)-Res derivation from Φ, we can use the phase function to associate a non-annotated literal u or $\neg u$ with P. Not only does this allows us to evaluate annotated literals; the nature of the phase function ensures that the rules of LD-Q($\mathcal{D}$)-Res are logically correct for each path in a fully exhibiting model. For a given annotated literal, our annotated literal function uses the same method as Balabanov et al. to identify the correct phase induced by some existential assignment. However, since we are not concerned with strategy extraction, we are able to simplify the construction considerably compared to [2][3]. For that reason, we proceed as follows.

As a starting point, consider the following resolution step in an LD-Q-Res refutation of a QBF $\Phi = \mathcal{Q} \,.\, \phi$ over variables V, where $\mathrm{var}(l_u) = u \in V_\forall$ and $x \in V_\exists$.

$$\frac{C_1 \cup \{x\} \cup \{l_u\} \qquad C_2 \cup \{\neg x\} \cup \{\neg l_u\}}{C_1 \cup C_2 \cup \{u^*\}}$$

Let M be any model for the conjunction of the antecedent clauses prefixed by $\mathcal{Q}$, and let $P \in M$. For any universal v which is right of u, we must have $x <_\Phi u <_\Phi v$; therefore $P(x) = P_v(x)$, meaning that at least one of $C_1 \cup \{u\}$ and $C_2 \cup \{\neg u\}$ is satisfied by both P and P_v. In either case we can then choose a single literal u or $\neg u$ for u^* such that $C_1 \cup C_2 \cup \{u^*\}$ is satisfied by both P and P_v.

Generalising, let M be a model for Φ. We observe that any resolution step producing u^* from complementary literals gives rise to a well-defined function $f^*_u : M \to \{u, \neg u\}$, with rule

$$f^*_u(P) = \begin{cases} l_u\,, & \text{if } P(x) = \bot\,, \\ \neg l_u\,, & \text{if } P(x) = \top\,. \end{cases}$$

We observe two features of such a definition. First, f^*_u simply reads the truth value of $P(x)$, selects the antecedent clause in which the pivot variable x is falsified, and takes the universal literal from that clause. In this way, any path $P \in M$ made to satisfy $C_1 \cup C_2 \cup f^*_u(P)$. Second, if v is any universal right of u, then $P(x) = P_v(x)$; hence v satisfies the *complementary property* $f^*_u(P) = f^*_u(P_v)$ for all $P \in M$.

Moreover, the above discussion does not consider 'successive merging'. Moving forward to the annotated literals of LD-Q($\mathcal{D}$)-Res, we therefore present a recursive definition based on the preceeding discussion.

Definition 6 (Annotated literal function). *Let $u^X \in U$ be a literal introduced by merging universal literals $l_1 \in U_1$ and $l_2 \in U_2$ in a resolution step*

$$\frac{C_1 \cup U_1 \cup \{x\} \qquad C_2 \cup U_2 \cup \{\neg x\}}{C_1 \cup C_2 \cup U}$$

of an LD-Q($\mathcal{D}$)-Res *refutation of a formula Φ. Let X_1, X_2 be the resolution sets of l_1, l_2 respectively, and let M be a model for Φ. Then the annotated literal*

[3] We can prove what we need to from the definition of such functions; we need not represent them explicitly as circuits as in [2].

function $f_u^X : M \to \{u, \neg u\}$ *for* M *is given by*

$$f_u^X(P) = \begin{cases} l_1\,, & \text{if } P(x) = \bot \text{ and } X_1 = \emptyset\,,\\ f_u^{X_1}(P)\,, & \text{if } P(x) = \bot \text{ and } X_1 \neq \emptyset\,,\\ l_2\,, & \text{if } P(x) = \top \text{ and } X_2 = \emptyset\,,\\ f_u^{X_2}(P)\,, & \text{if } P(x) = \top \text{ and } X_2 \neq \emptyset\,,\end{cases}$$

where $X = X_1 \cup X_2 \cup \{x\}$*.*

The following lemma states that the complementary property holds for annotated literal functions.

Lemma 7. *Let* Φ *be a QBF over variables* V*, let* $u, v \in V_\forall$*, let* $X \subseteq V_\exists$ *and let* M *be a model for* Φ *for which* $M \prec (v, x)$ *for all* $x \in X$*. Then any annotated literal function* f_u^X *for* M *satisfies* $f_u^X(P) = f_u^X(P_v)$ *for all paths* $P \in T$*.*

Proof. The lemma follows from the observation that, throughout the recursive definition of the annotated literal function f_u^X, complementary paths P and P_v always map to the same case, since $P(x) = P_v(x)$ for all $x \in X$ and for all $P \in M$. ☐

Evaluation of annotated literals. We defined annotated literal functions for a model M specifically so that any $P \in M$ satisfying both antecedents of a resolution step also satisfies the resolvent. For that reason, we define u^X to have the same truth value as the concrete literal $f_u^X(P)$ when evaluated under α, the assignment represented by path P; that is, we define $(u^X)|_\alpha = (f_u^X(P))|_\alpha$. Representing assignments by paths, this would be written $P(u^X) = P(f_u^X(P))$, but we emphasize that the apparent circularity here is simply the result of the notational convenience of using the symbol P to represent both an assignment tree path and its corresponding assignment. The expression $P(f_u^X(P))$ is always well-defined because f_u^X can be computed for any given model M, so $f_u^X(P)$ is a well-defined non-annotated literal, which can then be evaluated under P in the usual way. We are now in a position to prove the following theorem.

Theorem 8. *Let* $\mathcal{D}$ *be a fully exhibited proto-dependency scheme. Then* LD-Q($\mathcal{D}$)-Res *is sound.*

Proof. Let $\Phi = \mathcal{Q}\,.\,\phi$ be a QBF over variables V, suppose that $\pi = \{C_1, \ldots, C_l\}$ is a LD-Q($\mathcal{D}$)-Res refutation of Φ, and let

$$\phi_i = \begin{cases} \phi & \text{if } i = 0\,,\\ \phi \wedge C_1 \wedge \cdots \wedge C_i & \text{otherwise}\,,\end{cases}$$

for $i = 1, \ldots, l$. Since $\mathcal{D}$ is fully exhibited, if Φ is true there exists a model M for Φ for which $M \prec \bar{\mathcal{D}}^{\mathrm{rrs}}_\Phi(u)$ for all $u \in V_\exists$. We prove by induction on i that if Φ is true, M is a model for $\mathcal{Q}\,.\,\phi_i$, so $\mathcal{Q}\,.\,\phi_i$ is true. Hence at step $i = l$, we deduce that $\Phi = \mathcal{Q}\,.\,\phi_l$ is true, a clear contradiction since ϕ_l contains the empty clause

C_l. Since $\mathcal{Q} \,.\, \phi = \mathcal{Q} \,.\, \phi_0$, if Φ is true then M is a model for $\mathcal{Q} \,.\, \phi_0$, thus the base case $i = 0$ is established. We only need confirm that if M is a model for $\mathcal{Q} \,.\, \phi_i$, then M is a model for $\mathcal{Q} \,.\, \phi_{i+1}$, for $i \in [l-1]$.

Suppose that $C_{i+1} = C_1 \cup C_2 \cup U$ is the resolvent of clauses $C_j = C_1 \cup U_1 \cup \{x\}$ and $C_k = C_2 \cup U_2 \cup \{\neg x\}$ for $j, k < i+1$, and let P be an arbitrary path in M. By the inductive hypothesis, P satisfies C_j and C_k. Assume without loss of generality that $P(x) = \bot$. Then P satisfies $C_1 \cup U_1$. If P satisfies C_1 then P satisfies C_{i+1}. Otherwise, $P(l_u) = \top$ for some (annotated or non-annotated) literal $l_u \in U_1$ with $\text{var}(l_u) = u$. The recursive definition of the annotated literal function ensures that $P(l_u) = \top \Rightarrow P(u^X) = \top$ for some annotated literal $u^X \in U$, and so P satisfies C_{i+1}. Therefore M is a model for $\mathcal{Q} \,.\, \phi_{i+1}$.

On the other hand, suppose that C_{i+1} was obtained from $C_j, j < i+1$, by $\forall$-reduction on a non-annotated universal literal u. Then $C_{i+1} = C_j \backslash \{l_u\}$, where $\text{var}(l_u) = u$, $(u, x) \notin \mathcal{D}_\Phi$ for all $x \in C_j \cap V_\exists$ and for all $x \in X$, where X is the union of the resolution sets of all universal literals in C_j. Suppose that there exists some path P in M which satisfies C_j but falsifies C_{i+1}. Let $z \in C_{i+1}$; then $P(z) = \bot$, and since $M \prec_\Phi (u, x)$ for all $(u, x) \notin \mathcal{D}_\Phi$, we have $P_u(z) = P(z) = \bot$ whenever z is a non-annotated literal. On the other hand, suppose that $z = v^X$, where $v \neq u$; then, since $M \prec_\Phi (u, x)$ for all $x \in X$, Lemma 7 gives $f_v^X(P) = f_v^X(P_u) = l_v$ with $\text{var}(l_v) = v$, which implies $P^u(v^X) = P(v^X) = \bot$. Also, since $P(u) = \top$, we have $P_u(u) = \bot$, and we deduce that $P_u(C_j) = \bot$, contradicting that M is a model for $\mathcal{Q}.\phi_i$. It follows that P satisfies C_{i+1}, and that M is a model for $\mathcal{Q} \,.\, \phi_{i+1}$.

The same argument applies to an annotated literal u^X. Since $M \prec_\Phi (u, x)$ for all $x \in X$, the special case of Lemma 7 with $v = u$ gives $f_u^X(P) = f_u^X(P_u)$, hence $P(u^X) = \top$ implies $P_u(u^X) = \bot$. This completes the proof. $\square$

Since the proof of Theorem 8 makes no use of the fact that the pivot is existential, it suffices also to show the soundness of $\mathsf{LQU}(\mathcal{D})$-$\mathsf{Res}$, the 'dependency version' of LQU-Res, for fully exhibited $\mathcal{D}$.

Theorem 9. *Let $\mathcal{D}$ be a fully exhibited proto-dependency scheme. Then* $\mathsf{LQU}(\mathcal{D})$-$\mathsf{Res}$ *is sound.*

Also, since $\mathsf{LQU}(\mathcal{D})$-$\mathsf{Res}$ clearly simulates $\mathsf{QU}(\mathcal{D})$-Res simply by disallowing long-distance resolution steps, we obtain same result for $\mathsf{QU}(\mathcal{D})$-Res.

Theorem 10. *Let $\mathcal{D}$ be a fully exhibited proto-dependency scheme. Then* $\mathsf{QU}(\mathcal{D})$-Res *is sound.*

Theorems 8, 9 and 10 together constitute the generalisation to all the CDCL QBF calculi of Slivovsky's result [20] that $\mathsf{Q}(\mathcal{D})$-Res is sound for fully exhibited $\mathcal{D}$. Whereas full exhbition is a sufficient condition for each calculus, it is not a necessary condition for any of them, witnessed by the following counter example.

Example 11. Consider the formula $\Psi = \forall u_1 \forall u_2 \exists x_1 \exists x_2 \,.\, \psi$ with matrix

$$\begin{aligned}\psi \;=\; \big\{ &\{u_1, x_1, \neg x_2\}_1, \{u_1, \neg x_1, x_2\}_2, \{\neg u_1, u_2, x_1, \neg x_2\}_3, \{\neg u_1, u_2, \neg x_1, x_2\}_4, \\ &\{\neg u_1, \neg u_2, x_1, x_2\}_5, \{\neg u_1, \neg u_2, \neg x_1, \neg x_2\}_6 \big\} ,\end{aligned}$$

and the dependency scheme $\mathcal{D}'$ defined by $\mathcal{D}'(\Phi) = \{(u_1, x_1), (u_2, x_2)\}$ if $\Phi = \Psi$ and $\mathcal{D}'(\Phi) = \mathcal{D}^{\text{trv}}(\Phi)$ otherwise. It can be verified that Ψ is true, but there is no model for Ψ which exhibits both independencies (u_1, x_2) and (u_2, x_1) simultaneously, and hence $\mathcal{D}'$ is not fully exhibited. However, there is no $\mathsf{LQU}(\mathcal{D}')$-$\mathsf{Res}$ refutation of Ψ. One may resolve clauses 1 and 3 over u_1 to obtain $\{u_2, x_1, \neg x_2\}$, and resolve over clauses 2 and 4 to obtain $\{u_2, \neg x_1, x_2\}$. Beyond these two steps, no more $\mathsf{LQU}(\mathcal{D})$-$\mathsf{Res}$ steps can be made.

5 Demonstrating Full Exhibition

In this section, we demonstrate that the reflexive resolution path dependency scheme $\mathcal{D}^{\text{rrs}}$ [21] is fully exhibited, thereby proving a conjecture of Slivovsky [20, p.37]. This result provides for a better understanding of soundness in Q-resolution with dependency shemes; since $\mathcal{D}^{\text{rrs}}$ is the most general scheme known to be sound in $\mathsf{Q}(\mathcal{D})$-Res, what is already known about soundness for that calculus can subsequently be explained entirely by full exhibition.

The scheme $\mathcal{D}^{\text{rrs}}$ uses the notion of 'resolution paths' introduced in [23], which define connections through the matrix with respect to a particular set of variables. For convenience, we represent the connections used in $\mathcal{D}^{\text{rrs}}$ as a binary relation $\mathcal{C}_\Phi$.

Definition 12. *Let $\Phi = \mathcal{Q} \,.\, \phi$ be a PCNF over variables V and let l, l' be literals such that $(\text{var}(l), \text{var}(l')) \in \mathcal{D}^{trv}_\Phi$. Then $(l, l') \in \mathcal{C}_\Phi$ iff there is a sequence of clauses $C_1, \ldots, C_n \in \phi$ and a sequence of literals $l_1, \ldots, l_{n-1} \in V_\exists \cap \mathcal{R}_\Phi(\text{var}(l))$ such that $l_i \in C_i$, $\neg l_i \in C_{i+1}$ and $\text{var}(l_i) \neq \text{var}(l_{i+1})$ for $i \in [n-1]$.*

Definition 13 (Reflexive resolution path dependency scheme [21]). *The reflexive resolution path dependency scheme $\mathcal{D}^{rrs}$ maps each PCNF Φ to the dependency relation*

$$\mathcal{D}^{rrs}_\Phi = \{(z_1, z_2) \in \mathcal{D}^{trv}_\Phi \mid (z_1, z_2), (\neg z_1, \neg z_2) \notin \mathcal{C}_\Phi \text{ or } (z_1, \neg z_2), (\neg z_1, z_2) \notin \mathcal{C}_\Phi\}.$$

We first introduce some auxiliary notation for this section. We extend the notion of exhibition of independence from pairs to sets of pairs; if $S = \{(z_1, z'_1), \ldots, (z_n, z'_n)\}$ we take $M \prec S$ to mean $M \prec (z_i, z'_i)$ for $i \in [n]$. Also, for a PCNF Φ over variables V and $u \in V_\forall$, we adopt the notation $\bar{\mathcal{D}}^{\text{rrs}}_\Phi(u) = \{(u, x) \mid x \in V_\exists \text{ and } (u, x) \notin \mathcal{D}^{\text{rrs}}_\Phi\}$.

The proof is obtained by showing that an arbitrary model for a true PCNF can be transformed to exhibit all the required independencies. We begin by defining an operation $\text{ref}_u(P)$, which reforms a model path P based on the assignments of its complementary path with respect to a given universal variable u. We then prove that the resulting path does not falsify any clauses.

Definition 14 (Reformed path). *Let M be a model for a PCNF Φ over variables V, let $P \in M$, let $u \in V_\forall$ and put $l_u = P[u]$. The* reformed path $\text{ref}_u(P)$ *of P with respect to u is given by*

$$\text{ref}_u(P)[z] = \begin{cases} P_u[z] & \text{if } z \in V_\exists, P_u[z] = l_z, \text{ and } (\neg l_u, \neg l_z) \notin \mathcal{C}_\Phi\,, \\ P[z] & \text{otherwise}\,. \end{cases}$$

Lemma 15. *Let M be a model for a PCNF $\Phi = \mathcal{Q}\,.\,\phi$ over variables V, let $P \in M$ and let $u \in V_\forall$. Then* $\mathrm{ref}_u(P)(C) = \top$ *for all $C \in \phi$.*

Proof. Towards a contradiction, suppose that $\mathrm{ref}_u(P)(C) = \bot$ for some $C \in \phi$, and assume without loss of generality that $\mathrm{ref}_u(P)[u] = \neg u$ and $P_u[u] = u$. Since M is a model for Φ and $P \in M$, there is some existential literal $l \in C$ for which $P[\mathrm{var}(l)] = l$ but $\mathrm{ref}_u(P)[\mathrm{var}(l)] = \neg l$, so by Def. 14 we have $P_u[\mathrm{var}(l)] = \neg l$ and $(u, l) \notin \mathcal{C}_\Phi$. The latter implies that $(u, \neg l') \notin \mathcal{C}_\Phi$ for all existential literals $l' \in C$ such that $l' \neq l$. Hence, by Def. 14, if $P_u[\mathrm{var}(l')] = l'$, then $\mathrm{ref}_u[\mathrm{var}(l')] = l'$; but $\mathrm{ref}_u[\mathrm{var}(l')] = \neg l'$, so we must have $P_u[\mathrm{var}(l')] = \neg l'$. It follows that P_u falsifies all existential literals in C. Since $\mathrm{ref}_u(P)$ and P_u agree on all universal variables except u, and literal $u = P[u] \notin C$ (because $(u, l) \notin \mathcal{C}_\Phi$), P_u also falsifies all universal literals in C. Therefore $P_u(C) = \bot$, contradicting the premise that M is a model for Φ. □

We proceed to define $\mathrm{ref}_u(M)$, the extension of the reformation operation from paths to models, in which all pairs of complementary paths P and P_u in some model M are reformed. The resulting models enjoys the useful properties stated in the subsequent lemma.

Definition 16 (Reformed model). *Let M be a model for a PCNF Φ over variables V, let $u \in V_\forall$, let $G = \{P \in M \mid P[u] = \neg u\}$ and let $M' = (M \setminus G) \cup \hat{G}$, where $\hat{G} = \{\mathrm{ref}_u(P) \mid P \in G\}$. Then the* reformed model *of M with respect to u is $\mathrm{ref}_u(M) = (M' \setminus G') \cup \hat{G}'$, where $G' = \{P \in M' \mid P[u] = u\}$ and $\hat{G}' = \{\mathrm{ref}_u(P) \mid P \in G'\}$.*

Lemma 17. *Let M be a model for a PCNF Φ over variables V, and let $u \in V$. Then*

(a) $\mathrm{ref}_u(M)$ *is a model for Φ,*
(b) $\mathrm{ref}_u(M) \prec \bar{\mathcal{D}}^{rrs}_\Phi(u)$, *and*
(c) *if $M \prec (u', x)$, with $u' \in V_\forall$ and $(u', u) \in R_\Phi$, then* $\mathrm{ref}_u(M) \prec (v, x)$.

Proof. We let M' be defined as in Def. 16, and use the alias $M'' = \mathrm{ref}_u(M)$. Let $P \in M$ such that $P[u] = \neg u$ and let U be the set of universal literals in P. We denote by $P' \in M'$ and $P'' \in M''$ the unique paths such that $U \subseteq P'$ and $U \subseteq P''$. Observe that, by Def. 16, $P'' = P' = \mathrm{ref}_u(P)$, $P''_u = \mathrm{ref}_u(P'_u)$ and $P'_u = P_u$.

(a) We prove that M' is a model for Φ. By Lemma 15, every path in M' satisfies ϕ. To show that M' is well-formed for Φ, let $v \in V_\forall$, let $U' = L_\Phi(v) \cap V_\forall$ and let $S' \in M'$ such that $S'[v'] = P'[v']$ for all $v' \in U'$. Let $x \in V_\exists$ such that $x \in L_\phi(v)$, and let $S \in M$ such that $S' = \mathrm{ref}_u(S)$. Since M is well-formed and $S_u[v'] = P_u[v']$ for all $v' \in U'$, we must have $S_u[x] = P_u[x]$ for all $x \in V_\exists \cap L_\Phi(v)$. Then $S'[x] = P'[x]$ by Def. 14. By construction, every universal assignment defines a unique path in M', so M' is well-formed for Φ. A similar argument shows that M'' is a model for Φ.

(b) Let $x \in V_\exists$ such that $(u, x) \notin \mathcal{D}^{\mathrm{rrs}}_\Phi$. It is sufficient to show that $P''[x] = P''_u[x]$ follows from Def. 14; to do this we consider two cases. (1) Suppose that $P[x] = P_u[x] = l_x$. Then $P'[x] = P'_u[x] = l_x$ and $P''[x] = P''_u[x] = l_x$. (2) Suppose intsead that $P[x] = l_x$ and $P_u[x] = \neg l_x$. Since $(u, x) \notin \mathcal{D}^{\mathrm{rrs}}_\Phi$, we must have either $(u, l_x) \notin \mathcal{C}_\Phi$ or $(\neg u, \neg l_x) \notin \mathcal{C}_\Phi$. If $(u, l_x) \notin \mathcal{C}_\Phi$, we have both $P'[x] = P'_u[x] = \neg l_x$ and $P''[x] = P''_u[x] = \neg l_x$. On the other hand, if $(u, l_x) \in \mathcal{C}_\Phi$, $P'[x] = l_x$ and $P'_u[x] = \neg l_x$, whereupon $(\neg u, \neg l_x) \notin \mathcal{C}_\Phi$ yields $P''[x] = P''_u[x] = l_x$.

(c) Suppose that $M \prec (u', x)$, with $u' \in V_\forall$ and $(u', u) \in R_\Phi$, and put $Q = P_{u'}$. Similarly as for the path P above, let $U_{u'}$ be the set of universal literals in Q, and denote by $Q' \in M'$ and $Q'' \in M''$ the unique paths such that $U_{u'} \subseteq Q'$ and $U_{u'} \subseteq Q''$. Observe that, again by Def. 16, $Q'' = Q' = \mathrm{ref}_u(Q)$, $Q''_u = \mathrm{ref}_u(Q'_u)$ and $Q'_u = Q_u$. Since we assume $P[u] = \neg u$, to deduce $M'' \prec (u', x)$ we must show that $P''[x] = Q''_u[x]$ and that $P''_u[x] = Q''_u[x]$. The observation that $P[x] = Q[x]$ and $P_u[x] = Q_u[x]$, in combination with Def. 16, leads easily to the result. Firstly, this guarantees that $\mathrm{ref}_u(P)[x] = \mathrm{ref}_u(Q)[x]$, that is $P'[x] = Q'[x]$, therefore $P''[x] = Q''[x]$. Secondly, using $P'[x] = Q'[x]$, it also guarantees that $\mathrm{ref}_u(P'_u)[x] = \mathrm{ref}_u(Q'_u)[x]$, that is $P''_u[x] = Q''_u[x]$. □

The main result of this section follows quickly.

Theorem 18. *$\mathcal{D}^{rrs}$ is fully exhibited.*

Proof. Let M_0 be a model for a PCNF Φ over variables V, let $V_\forall = \{u_1, \dots, u_n\}$ with $u_i <_\Phi u_{i+1}$ for $i \in [n-1]$, and let $M_{i+1} = \mathrm{ref}_{u_i}(M_i)$ for $i \in [n-1]$. We claim that M_n is a model for Φ such that $M_n \prec \bar{\mathcal{D}}^{\mathrm{rrs}}_\Phi(u)$ for all $u \in V_\forall$.

By induction on $i \in [n]$, we prove that M_i is a model for Φ such that $M_i \prec \bigcup_{j=1}^{i} \bar{\mathcal{D}}^{\mathrm{rrs}}_\Phi(u_j)$, and hence at step $i = n$ we prove the claim and the theorem. For the base case $i = 1$, observe that M_1 is model for Φ by Lemma 17(a), and that $M_1 \prec \bar{\mathcal{D}}^{\mathrm{rrs}}_\Phi(u_1)$ by Lemma 17(b). For the inductive step, let $i \in [n-1]$ and suppose that M_i is a model for Φ and that $M_i \prec \bigcup_{j=1}^{i} \bar{\mathcal{D}}^{\mathrm{rrs}}_\Phi(u_j)$. Then $M_{i+1} = \mathrm{ref}_{u_i}(M_i)$ is a model for Φ by Lemma 17(a), $M_{i+1} \prec \bigcup_{j=1}^{i} \bar{\mathcal{D}}^{\mathrm{rrs}}_\Phi(u_j)$ by Lemma 17(c), and $M_{i+1} \prec \bar{\mathcal{D}}^{\mathrm{rrs}}_\Phi(u_{i+1})$ by Lemma 17(b). Therefore $M_{i+1} \prec \bigcup_{j=1}^{i+1} \bar{\mathcal{D}}^{\mathrm{rrs}}_\Phi(u_j)$. □

Our concluding result now follows immediately from Theorems 8, 9 and 10.

Corollary 19. QU($\mathcal{D}^{\mathsf{rrs}}$)-Res*,* LD-Q($\mathcal{D}^{\mathsf{rrs}}$)-Res *and* LQU($\mathcal{D}^{\mathsf{rrs}}$)-Res *are sound proof systems.*

6 Conclusions and Open Problems

As we have shown, the parametrization by dependency schemes can be extended to all four CDCL QBF calculi, and the property of full exhibition – which is possessed by the reflexive resolution path dependency scheme – is sufficient for soundness in each case. Showing by counterexample that full-exhibition is not a necessary condition, our work leads naturally to the open problem of finding a characterization for soundness in this setting.

Acknowledgments. This research was supported by grant no. 48138 from the John Templeton Foundation and EPSRC grant EP/L024233/1.

References

1. Balabanov, V., Jiang, J.R.: Unified QBF certification and its applications. Formal Methods in System Design 41(1), 45–65 (2012)
2. Balabanov, V., Jiang, J.R., Janota, M., Widl, M.: Efficient extraction of QBF (counter)models from long-distance resolution proofs. In: Conference on Artificial Intelligence (AAAI). pp. 3694–3701 (2015)
3. Balabanov, V., Widl, M., Jiang, J.R.: QBF resolution systems and their proof complexities. In: International Conference on Theory and Applications of Satisfiability Testing (SAT). pp. 154–169 (2014)
4. Beyersdorff, O., Bonacina, I., Chew, L.: Lower bounds: From circuits to QBF proof systems. In: Proc. ACM Conference on Innovations in Theoretical Computer Science (ITCS'16). pp. 249–260. ACM (2016)
5. Beyersdorff, O., Chew, L., Janota, M.: On unification of QBF resolution-based calculi. In: International Symposium on Mathematical Foundations of Computer Science (MFCS). pp. 81–93 (2014)
6. Beyersdorff, O., Chew, L., Janota, M.: Proof complexity of resolution-based QBF calculi. In: International Symposium on Theoretical Aspects of Computer Science (STACS). Leibniz International Proceedings in Informatics (LIPIcs), vol. 30, pp. 76–89 (2015)
7. Beyersdorff, O., Chew, L., Mahajan, M., Shukla, A.: Feasible interpolation for QBF resolution calculi. In: Proc. International Colloquium on Automata, Languages, and Programming (ICALP'15). pp. 180–192. Springer (2015)
8. Beyersdorff, O., Chew, L., Mahajan, M., Shukla, A.: Are short proofs narrow? QBF resolution is not simple. In: Proc. Symposium on Theoretical Aspects of Computer Science (STACS'16) (2016)
9. Biere, A., Lonsing, F.: Integrating dependency schemes in search-based QBF solvers. In: International Conference on Theory and Applications of Satisfiability Testing (SAT). pp. 158–171. Springer (2010)
10. Bubeck, U.: Model-based Transformations for Quantified Boolean Formulas. Ph.D. thesis (2010)
11. Egly, U.: On sequent systems and resolution for QBFs. In: Theory and Applications of Satisfiability Testing (SAT'12). pp. 100–113 (2012)
12. Egly, U., Lonsing, F., Widl, M.: Long-distance resolution: Proof generation and strategy extraction in search-based QBF solving. In: International Conference on Logic for Programming, Artificial Intelligence and Reasoning (LPAR). pp. 291–308 (2013)
13. Janota, M., Marques-Silva, J.: Expansion-based QBF solving versus Q-resolution. Theorical Compututer Science 577, 25–42 (2015)
14. Kleine Büning, H., Karpinski, M., Flögel, A.: Resolution for quantified boolean formulas. Information and Computation 117(1), 12–18 (1995)
15. Lonsing, F.: Dependency Schemes and Search-Based QBF Solving: Theory and Practice. Ph.D. thesis, Johannes Kepler University (2012)
16. Lonsing, F., Egly, U.: Incrementally computing minimal unsatisfiable cores of QBFs via a clause group solver API. In: International Conference on Theory and Applications of Satisfiability Testing (SAT). pp. 191–198 (2015)

17. Samer, M.: Variable dependencies of quantified csps. In: International Conference on Logic for Programming, Artificial Intelligence and Reasoning (LPAR). pp. 512–527 (2008)
18. Samer, M., Szeider, S.: Backdoor sets of quantified boolean formulas. Journal of Automated Reasoning 42(1), 77–97 (2009)
19. Samulowitz, H., Bacchus, F.: Using SAT in QBF. In: International Conference on Principles and Practice of Constraint Programming (CP). pp. 578–592 (2005)
20. Slivovsky, F.: Structure in #SAT and QBF. Ph.D. thesis, Vienna University of Technology (2015)
21. Slivovsky, F., Szeider, S.: Soundness of Q-resolution with dependency schemes. TCS 612, 83–101 (2016)
22. Stockmeyer, L.J., Meyer, A.R.: Word problems requiring exponential time: Preliminary report. In: Annual Symposium on Theory of Computing. pp. 1–9. ACM (1973)
23. Van Gelder, A.: Variable independence and resolution paths for quantified boolean formulas. In: International Conference on Principles and Practice of Constraint Programming (CP). pp. 789–803. Springer (2011)
24. Van Gelder, A.: Contributions to the theory of practical quantified Boolean formula solving. In: CP. pp. 647–663 (2012)
25. Zhang, L., Malik, S.: Conflict driven learning in a quantified boolean satisfiability solver. In: International Conference on Computer-aided Design (ICCAD). pp. 442–449 (2002)

ECCC ISSN 1433-8092

http://eccc.hpi-web.de