On Unification of QBF Resolution-Based Calculi

Original is to be published by Springer as a part of the MFCS '14 proceedings.

Olaf Beyersdorff[1], Leroy Chew[1], and Mikoláš Janota[2]

[1] School of Computing, University of Leeds, United Kingdom
[2] INESC-ID, Lisbon, Portugal

Abstract. Several calculi for quantified Boolean formulas (QBFs) exist, but relations between them are not yet fully understood. This paper defines a novel calculus, which is resolution-based and enables unification of the principal existing resolution-based QBF calculi, namely Q-resolution, long-distance Q-resolution and the expansion-based calculus ∀Exp+Res. All these calculi play an important role in QBF solving. This paper shows simulation results for the new calculus and some of its variants. Further, we demonstrate how to obtain winning strategies for the universal player from proofs in the calculus. We believe that this new proof system provides an underpinning necessary for formal analysis of modern QBF solvers.

1 Introduction

Traditionally, classifying a problem as NP-hard was ultimately understood as evidence for its infeasibility. Sharply contrasting this view, we have today fast algorithms for many important computational tasks with underlying NP-hard problems. One particularly compelling example of tremendous success is the area of SAT solving [25] where fast algorithms are being developed and tested for the classical NP-complete problem of satisfiability of propositional formulas (SAT). Modern SAT-solvers routinely solve industrial instances with even millions of variables. However, from a theoretical perspective, this success of SAT solvers is not well understood. The main theoretical approach to it comes via proof complexity. In particular, resolution and its subsystems have been very successfully analysed in terms of proof complexity and sharp bounds are known on the size and space for many important principles in resolution (cf. [6]). This is very important information as the main algorithmic approaches to SAT such as DPLL and CDCL are known to correspond to (systems of) resolution [2,7,14,26], and therefore bounds on size and space of proofs directly translate into bounds on running time and memory consumption of SAT solvers.

In the last decade, there has been ever-increasing interest to transfer the successful approach of SAT-solving to the more expressive case of *quantified propositional formulas (QBF)*. Due to its PSPACE completeness, QBF is far more expressive than SAT and thus applies to further fields such as formal verification or planning [27,4]. As for SAT, proof complexity provides the main

theoretical approach towards understanding the performance and limitations of QBF-solving. However, compared to proof complexity of classical propositional logic, QBF proof complexity is at a much earlier stage and also poses additional challenges. Currently, a handful of systems exist, and they correspond to different approaches in QBF-solving. In particular, Kleine Büning et al. [20] define a resolution-like calculus called *Q-resolution*. There are several extensions of Q-resolution; notably *long-distance Q-resolution* [1], which is believed to be more powerful than plain Q-resolution [10]. Q-resolution and its extensions are important as they model QBF solving based on CDCL [12]. Apart from CDCL, another main approach to QBF-solving is through expansion of quantifiers [5,3,16]. Recently, a proof system $\forall Exp{+}Res$ was introduced with the motivation to trace expansion-based QBF solvers [15]. $\forall$Exp+Res also uses resolution, but is conceptually very different from Q-resolution. The precise relation of $\forall$Exp+Res to Q-resolution is currently open (cf. [17]), but we conjecture that the two systems are incomparable as it has been shown that expansion-based solving can exponentially outperform DPLL-based solving.

In general, it is fair to say that relations between the different types of QBF systems mentioned above are currently not well understood. The objective of the present paper is to unify these approaches. Towards this aim we define a calculus that is able to capture the existing QBF resolution-based calculi and yet remains amenable to machine manipulation. Our main contributions are as follows. (1) We introduce two novel calculi IR-calc and IRM-calc, which are shown to be sound and complete for QBF. (2) IR-calc p-simulates Q-resolution and $\forall$Exp+Res, i.e., proofs in either Q-resolution or $\forall$Exp+Res can be efficiently translated into IR-calc. (3) The variant IRM-calc p-simulates long-distance Q-resolution. (4) We show how to extract winning strategies for the universal player from proofs in IR-calc and IRM-calc. Indeed, unified certification of QBF solvers or certification of solvers combining expansion and DPLL is of immense practical importance [13,1,10] and presents one of the main motivations for our research. To the best of our knowledge, constructions of strategies from expansion-based solvers were not known prior to this paper.

The rest of the paper is structured as follows. Section 2 overviews concepts and notation used throughout the paper. Section 3 introduces novel calculi and Section 4 shows how winning strategies for the universal player are constructed; this is used as an argument for soundness. Section 5 shows p-simulation results for the new calculi. Finally, Section 6 concludes the paper with a discussion. Due to space restrictions some proofs are sketched or omitted.

2 Preliminaries

A *literal* is a Boolean variable or its negation; we say that the literal x is *complementary* to the literal $\neg x$ and vice versa. If l is a literal, $\neg l$ denotes the complementary literal, i.e. $\neg\neg x = x$. A *clause* is a disjunction of zero or more literals. The empty clause is denoted by $\bot$, which is semantically equivalent to false. A formula in *conjunctive normal form* (CNF) is a conjunction of clauses.

$$\frac{}{C}\ \text{(Axiom)} \qquad \frac{C_1 \cup \{x\} \qquad C_2 \cup \{\neg x\}}{C_1 \cup C_2}\ \text{(Res)}$$

C is a clause in the matrix. Variable x is existential. If $z \in C_1$, then $\neg z \notin C_2$.

$$\frac{C \cup \{u\}}{C}\ (\forall\text{-Red})$$

Variable u is universal. If $x \in C$ is existential, then $\mathsf{lv}(x) < \mathsf{lv}(u)$.

Fig. 1. The rules of Q-Res [20]

Whenever convenient, a clause is treated as a set of literals and a CNF formula as a set of clauses. For a literal $l = x$ or $l = \neg x$, we write $\mathsf{var}(l)$ for x and extend this notation to $\mathsf{var}(C)$ for a clause C and $\mathsf{var}(\psi)$ for a CNF ψ.

A *proof system* (Cook, Reckhow [8]) for a language L over alphabet Γ is a polynomial-time computable partial function $f : \Gamma^\star \rightharpoonup \Gamma^\star$ with $rng(f) = L$. An *f-proof* of string y is a string x such that $f(x) = y$. In the systems that we consider here, proofs are sequences of clauses; a *refutation* is a proof deriving $\bot$. A proof system f for L *p-simulates* a system g for L if there exists a polynomial-time computable function t that translates g-proofs into f-proofs, i.e., for all $x \in \Gamma^\star$ we have $g(x) = f(t(x))$.

Quantified Boolean Formulas (QBFs) [19] extend propositional logic with quantifiers with the standard semantics that $\forall x.\Psi$ is satisfied by the same truth assignments as $\Psi[0/x] \wedge \Psi[1/x]$ and $\exists x.\Psi$ as $\Psi[0/x] \vee \Psi[1/x]$. Unless specified otherwise, we assume that QBFs are in *closed prenex* form with a CNF *matrix*, i.e., we consider the form $\mathcal{Q}_1 X_1 \ldots \mathcal{Q}_k X_k.\ \phi$, where X_i are pairwise disjoint sets of variables; $\mathcal{Q}_i \in \{\exists, \forall\}$ and $\mathcal{Q}_i \neq \mathcal{Q}_{i+1}$. The formula ϕ is in CNF and is defined only on variables $X_1 \cup \ldots \cup X_k$. The propositional part ϕ of a QBF is called the *matrix* and the rest the *prefix*. If a variable x is in the set X_i, we say that x is at *level* i and write $\mathsf{lv}(x) = i$; we write $\mathsf{lv}(l)$ for $\mathsf{lv}(\mathsf{var}(l))$. A closed QBF is *false* (resp. *true*), iff it is semantically equivalent to the constant 0 (resp. 1).

Often it is useful to think of a QBF $\mathcal{Q}_1 X_1 \ldots \mathcal{Q}_k X_k.\ \phi$ as a *game* between the *universal* and the *existential player*. In the i-th step of the game, the player $\mathcal{Q}_i$ assigns values to the variables X_i. The existential player wins the game iff the matrix ϕ evaluates to 1 under the assignment constructed in the game. The universal player wins iff the matrix ϕ evaluates to 0. A QBF is false iff there exists a *winning strategy* for the universal player, i.e. if the universal player can win any possible game.

2.1 Resolution-based Calculi for QBF

This section gives a brief overview of the main existing resolution-based calculi for QBF. *Q-resolution (Q-Res)*, by Kleine Büning et al. [20], is a resolution-like calculus that operates on QBFs in prenex form where the matrix is a CNF. The rules are given in Figure 1. *Long-distance resolution (LD-Q-Res)* appears

$$\frac{}{C}\ (\text{Axiom}) \qquad \frac{D \cup \{u\}}{D}\ (\forall\text{-Red}) \qquad \frac{D \cup \{u^*\}}{D}\ (\forall\text{-Red}^*)$$

C is a clause in the matrix. Literal u is universal and $\mathsf{lv}(u) \geq \mathsf{lv}(l)$ for all $l \in D$.

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

Variable x is existential. If for $l_1 \in C_1, l_2 \in C_2$, $\mathsf{var}(l_1) = \mathsf{var}(l_2) = z$ then $l_1 = l_2 \neq z^*$. U_1, U_2 contain only universal literals with $\mathsf{var}(U_1) = \mathsf{var}(U_2)$. For each $u \in \mathsf{var}(U_1)$ we require $\mathsf{lv}(x) < \mathsf{lv}(u)$. If for $w_1 \in U_1, w_2 \in U_2$, $\mathsf{var}(w_1) = \mathsf{var}(w_2) = u$ then $w_1 = \neg w_2$, $w_1 = u^*$ or $w_2 = u^*$. U is defined as $\{u^* \mid u \in \mathsf{var}(U_1)\}$.

Fig. 2. The rules of LD-Q-Res [1]

$$\frac{}{\{l^{\tau_l} \mid l \in C, l \text{ is existential}\} \cup \{\tau(l) \mid l \in C, l \text{ is universal}\}}\ (\text{Axiom})$$

C is a clause from the matrix and τ is an assignment to all universal variables.
τ_l are partial assignments obtained by restricting τ to variables u with $\mathsf{lv}(u) < \mathsf{lv}(l)$.

$$\frac{C_1 \cup \{x^\tau\} \qquad C_2 \cup \{\neg x^\tau\}}{C_1 \cup C_2}\ (\text{Res})$$

Fig. 3. The rules of ∀Exp+Res (adapted from [18])

originally in the work of Zhang and Malik [33] and was formalized into a calculus by Balabanov and Jiang [1]. It merges complementary literals of a universal variable u into the special literal u^*. These special literals prohibit certain resolution steps. In particular, different literals of a universal variable u may be merged only if $\mathsf{lv}(x) < \mathsf{lv}(u)$, where x is the resolution variable. The rules are given in Figure 2. Note that the rules do not prohibit resolving $w^* \vee x \vee C_1$ and $u^* \vee \neg x \vee C_2$ with $\mathsf{lv}(w) \leq \mathsf{lv}(u) < \mathsf{lv}(x)$ as long as $w \neq u$.

A different calculus ∀Exp+Res based on expansions was introduced in [18]. In Figure 3 we present an adapted version of this calculus so that it is congruent with the other resolution-based calculi (semantically it is the same as in [18]). The ∀Exp+Res calculus operates on clauses that comprise only existential variables from the original QBF; but additionally, each existential variable x is annotated with a substitution to those universal variables that precede x in the quantification order. For instance, the clause $x \vee b^{0/u}$ can be derived from the original clause $x \vee u$ under the prefix $\exists x \forall u \exists b$.

Besides the aforementioned resolution-based calculi, there is a system by Klieber et al. [23,22], which operates on pairs of sets of literals, rather than clauses; this system is in its workings akin to LD-Q-Res. Van Gelder defines an

extension of Q-Res, called *QU-resolution*, which additionally supports resolution over universal variables [32]. Another extension of Q-Res are *variable dependencies* [29,30,31] which enable more flexible ∀-reduction than traditional Q-Res. For proofs of true QBFs *term-resolution* was developed [11] or *models* in the form of Boolean functions [21] but those do not provide polynomially-verifiable proof systems. Some limitations of term-resolution were shown by Janota et al. [15]. A comparison of sequent calculi [24] and Q-Res was done by Egly [9].

3 Instantiation-based Calculi IR-calc and IRM-calc

We begin by setting up a framework allowing us to define our new calculi. The framework hinges on the concept of annotated clauses. An *extended assignment* is a partial mapping from the boolean variables to $\{0, 1, *\}$. An *annotated clause* is a clause where each literal is annotated by an extended assignment to universal variables. For an extended assignment σ to universal variables we write $l^{[\sigma]}$ to denote an annotated literal where $[\sigma] = \{c/u \in \sigma \mid \mathsf{lv}(u) < \mathsf{lv}(l)\}$. Two (extended) assignments τ and μ are called *contradictory* if there exists a variable $x \in \mathsf{dom}(\tau) \cap \mathsf{dom}(\mu)$ with $\tau(x) \neq \mu(x)$.

Further we define operations that let us modify annotations of a clause by *instantiation*. For (extended) assignments τ and μ, we write $\tau \trianglelefteq \mu$ for the assignment σ defined as follows: $\sigma(x) = \tau(x)$ if $x \in \mathsf{dom}(\tau)$, otherwise $\sigma(x) = \mu(x)$ if $x \in \mathsf{dom}(\mu)$. The operation $\tau \trianglelefteq \mu$ is referred to as *completion* because μ provides values for variables that are not defined in τ. The operation is associative and therefore we can omit parentheses. In contrast, it is *not* commutative. The following properties hold: (i) For non-contradictory μ and τ, we have $\mu \trianglelefteq \tau = \tau \trianglelefteq \mu = \mu \cup \tau$. (ii) $\tau \trianglelefteq \tau = \tau$.

We consider an auxiliary function $\mathsf{inst}(\tau, C)$, which for an extended assignment τ and an annotated clause C returns $\left\{l^{[\sigma \trianglelefteq \tau]} \mid l^{\sigma} \in C\right\}$.

Our first new system IR-calc operates on clauses annotated with usual assignments with range $\{0, 1\}$. The calculus introduces clauses from the matrix and allows to **i**nstantiate and **r**esolve clauses; hence the name IR-calc. It comprises the rules in Figure 4.

Our second system IRM-calc is an extension of IR-calc where we allow extended assignments with range $\{0, 1, *\}$. To introduce $*$ we include a new rule called *merging*. IRM-calc is defined in Figure 5. The resolution rule can now deal with $*$, but when $\sigma = \xi = \emptyset$ we have exactly the resolution rule from Figure 4.

Example 1. Consider the (true) QBF $\exists x \forall u w \exists b.\, (x \vee u \vee b) \wedge (\neg x \vee \neg u \vee b) \wedge (u \vee w \vee \neg b)$. In both calculi axioms yield $x \vee b^{0/u}$, $\neg x \vee b^{1/u}$, and $\neg b^{0/w,0/u}$. In IR-calc we resolve to get $b^{0/u} \vee b^{1/u}$. IRM-calc further derives $b^{*/u}$ by merging. Intuitively, $b^{0/u} \vee b^{1/u}$ means that the existential player must play so that for any assignment to w either $b = 1$ if $u = 0$, or $b = 0$ if $u = 1$. So for instance, the player might choose to play $b = 1$ if $w = 0$ and $u = 1$, and if $w = 1$ and $u = 0$. The clause $b^{*/u}$ can be seen as a shorthand for the clause $b^{0/u} \vee b^{1/u}$. Note that it would be *unsound* to derive the clause b (with no annotation). This would

mean that b must be 1 regardless of the moves of the universal player. However, b needs to be 0 when $u = w = 0$ due to the third axiom. ▲

Note that in ∀Exp+Res, propositional variables are introduced so that their annotations assign *all* relevant variables. Like so each literal corresponds to a value of a Skolem function in a specific point. In contrast, in IR-calc, variables are annotated "lazily", i.e. it enables us to reason about multiple points of Skolem functions at the same time. This is analogous to *specializaion* of free variables by constants in first-order logic (FOL). Similarly, resolution in IR-calc is analogous to resolution in Robinson's FOL resolution [28]. IRM-calc additionally enables "compressing" literals with contradictory annotations.

4 Soundness and Extraction of Winning Strategies

The purpose of this section is twofold: show how to obtain a *winning strategy* for the universal player given an IRM-calc proof, and, to show that IRM-calc is *sound* (and therefore also IR-calc). First we show how to obtain a winning strategy for the universal player from a proof. From this, the soundness of the calculus follows because a QBF is false if and only if such strategy exists.

The approach we follow is similar to the one used for Q-Res [13] or LD-Q-Res [10]. Consider a QBF $\Gamma = \exists E \forall U. \Phi$, where E and U are sets of variables and Φ is a QBF (potentially with further quantification). Let π be an IRM-calc refutation of Γ, and let ϵ be a total assignment to E. The assignment ϵ represents a move of the existential player. Reduce π to a refutation π_ϵ of $\forall U. \Phi|_\epsilon$. To obtain a response of the universal player, we construct an assignment μ to the variables U such that reducing π_ϵ by μ gives a refutation of $\Phi|_{\epsilon \cup \mu}$.

Let $\pi_{\epsilon,\mu}$ be the proof resulting from reducing π_ϵ by μ. The game continues with $\phi|_{\epsilon\cup\mu}$ and $\pi_{\epsilon,\mu}$. In each of these steps, two quantifier levels are removed from the given QBF and a refutation for each of the intermediate formulas is produced. This guarantees a winning strategy for the universal player because in the end the existential player will be faced with an unsatisfiable formula without universal variables. We follow this notation for the rest of the section.

$$\frac{}{\left\{x^{[\tau]} \mid x \in C, x \text{ is existential}\right\}} \text{ (Axiom)}$$

C is a non-tautological clause from the matrix. $\tau = \{0/u \mid u \text{ is universal in } C\}$, where the notation $0/u$ for literals u is shorthand for $0/x$ if $u = x$ and $1/x$ if $u = \neg x$.

$$\frac{x^\tau \vee C_1 \qquad \neg x^\tau \vee C_2}{C_1 \cup C_2} \text{ (Resolution)} \qquad \frac{C}{\mathsf{inst}(\tau, C)} \text{ (Instantiation)}$$

τ is an assignment to universal variables with $\mathsf{rng}(\tau) \subseteq \{0, 1\}$.

Fig. 4. The rules of IR-calc.

Axiom and instantiation rules as in IR-calc in Figure 4.

$$\frac{x^{\tau\cup\xi} \vee C_1 \qquad \neg x^{\tau\cup\sigma} \vee C_2}{\mathsf{inst}(\sigma, C_1) \cup \mathsf{inst}(\xi, C_2)} \text{ (Resolution)}$$

$\mathsf{dom}(\tau)$, $\mathsf{dom}(\xi)$ and $\mathsf{dom}(\sigma)$ are mutually disjoint. $\mathsf{rng}(\tau) = \{0, 1\}$

$$\frac{C \vee b^{\mu} \vee b^{\sigma}}{C \vee b^{\xi}} \text{ (Merging)}$$

$\mathsf{dom}(\mu) = \mathsf{dom}(\sigma)$. $\xi = \{c/u \mid c/u \in \mu, c/u \in \sigma\} \cup \{*/u \mid c/u \in \mu, d/u \in \sigma, c \neq d\}$

Fig. 5. The rules of IRM-calc.

To reduce a refutation π by the existential assignment ϵ, we reduce the leaves of π by ϵ and repeat the steps of π with certain modifications. Instantiation steps are repeated with no discrimination. Merging is repeated in the reduced proof unless either of the merged literals is not in the reduced clause and then the clause is left as it is. Whenever a resolution step is possible, repeat it in the reduced proof. If it is not possible, the resolvent in the reduced proof is obtained from the antecedent that is not $\top$ and does *not* contain the pivot literal. If such does not exist, the resolvent is marked as $\top$ (effectively removing it from the proof). When producing a resolvent from a single antecedent, additional instantiation is required. This instantiation is the same one as done by the original resolution step but any $*$ is replaced by 0 (indeed, we can choose the constant arbitrarily). Like so, domains of annotations are preserved. In the end, any clauses marked as $\top$ are removed.

To obtain an assignment to the variables U, collect all the assignments μ to U appearing in annotations in π_ϵ; any variable not appearing in π_ϵ is given an arbitrary value. To obtain $\pi_{\epsilon,\mu}$, remove occurrences of U-variables from the annotation in the proof π_ϵ. This will leave us with a valid refutation because we will show in Lemma 3 that for each variable in U only a single value constant annotation can appear in the entire proof π_ϵ.

To show that this procedure is correct, we need to argue that the reduction returns a valid IRM-calc refutation π_ϵ, and that π_ϵ does not contain annotations giving contradictory values to variables in U. We start with the first claim.

Lemma 2. *The above reduction yields a valid IRM-calc refutation π_ϵ of $\forall U. \Phi|_\epsilon$.*

We omit the proof, which proceeds by induction on the derivation depth.

Lemma 3. *Let π be an IRM-calc refutation of a QBF formula starting with a block of universally quantified variables U. Consider the set of annotations μ on variables U that appear anywhere in π. Then μ is non-contradictory and does not contain instances of $*$.*

Proof. The proof proceeds by induction on the derivation depth. Let μ_C denote the set of annotations to variables in U appearing anywhere in the derivation of

C (i.e., we only consider the connected component of the proof dag with sink C). The induction hypothesis states:

(i) The set μ_C is non-contradictory.
(ii) For every literal $l^\sigma \in C$, it holds that $\mu_C \subseteq \sigma$.
(iii) $*/u \notin \mu_C$, for any $u \in U$.

Base Case. Condition (i) is satisfied by the axioms because we are assuming there are no complementary literals in clauses in the matrix. Condition (ii) is satisfied because all existential literals are at a higher level than the variables of U. Condition (iii) holds because we do not instantiate by $*/u$ in the axiom rule.

Instantiation. Let $u \in U$ and $C = \mathsf{inst}(c/u, C')$ in the proof π. By induction hypothesis, u either appears in the annotations of all the literals l^ξ in C' or it does not appear in any of them. In the first case, the instantiation step is ineffective. In the second case, c/u is added to all literals in C. By induction hypothesis u does not appear in any annotation of any clause in the sub-proof deriving C', and hence C is the first clause containing u.

Resolution. Let C be derived by resolving $x^{\tau \cup \xi} \vee C_1$ and $\neg x^{\tau \cup \sigma} \vee C_2$. Let $u \in U$, consider the following cases.

Case 1. For some $c \in \{0, 1\}$, $c/u \in \sigma$ and $u \notin \mathsf{dom}(\xi)$. By induction hypothesis, u does not appear in the annotations of C_1. Hence $\mathsf{inst}(\sigma, C_1)$ adds c/u to all the annotations in C_1.

Case 2. $c/u \in \tau$. By induction hypothesis, c/u appears in all annotations of C_1, C_2 and hence in all annotations of the resolvent.

Case 3. $u \notin \mathsf{dom}(\tau) \cup \mathsf{dom}(\sigma) \cup \mathsf{dom}(\xi)$. Then u does not appear as annotation anywhere in the derivation of either of the antecedents and neither it will appear in the resolvent.

Merging. Because of (i) we do not obtain $*$ for variables in U. □

Therefore we obtain winning strategies:

Theorem 4. *The construction above yields a winning strategy for the universal player.*

The soundness of IRM-calc follows directly from Theorem 4.

Corollary 5. *The calculi IR-calc and IRM-calc are sound.*

5 Completeness and Simulations of Known QBF Systems

In this section we prove that our calculi simulate the main existing resolution-based QBF proof systems. As a by-product, this also shows completeness of our proof systems IR-calc and IRM-calc. We start by simulating Q-resolution, which is even possible with our simpler calculus IR-calc.

Theorem 6. *IR-calc p-simulates Q-Res.*

Proof (Sketch). Let $C_1, \ldots, C_k$ be a Q-Res proof. We translate the clauses into $D_1, \ldots, D_k$, which will form the skeleton of a proof in IR-calc.

- For an axiom C_i in Q-Res we introduce the same clause D_i by the axiom rule of IR-calc, i.e., we remove all universal variables and add annotations.
- If C_i is obtained via $\forall$-reduction from C_j, then $D_i = D_j$.
- Consider now the case that C_i is derived by resolving C_j and C_k with pivot variable x. Then $D_j = x^\tau \vee K_j$ and $D_k = x^\sigma \vee K_k$. We instantiate to get $D'_j = \mathsf{inst}(\sigma, D_j)$ and $D'_k = \mathsf{inst}(\tau, D_k)$. Define D'_i as the resolvent of D'_j and D'_k. In order to obtain D_i we must ensure that there are no identical literals with different annotations. For this consider the set $\zeta = \{c/u \mid c/u \in t, l^t \in D'_i\}$ and define $D_i = \mathsf{inst}(\zeta, D'_i)$. This guarantees that we will always have fewer literals in D_i than in C_i, and we get a refutation.

We have to prove that the resolution steps are valid, by showing that τ and σ are not contradictory and ζ does not contain contradictory annotations. This follows from the next claim, which can be proven by induction (omitted here).

Claim. For all existential literals l we have $l \in C_i$ iff $l^t \in D_i$ for some annotation t. Additionally, if $0/u \in t$ for a literal u, then $u \in C_i$ (where for a variable x, we equivalently denote the annotation $1/x$ by $0/\neg x$). □

Despite its simplicity, IR-calc is powerful enough to also simulate the expansion based proof system ∀Exp+Res from [18].

Theorem 7. *IR-calc p-simulates ∀Exp+Res.*

Proof. Let $C_1, \ldots, C_k$ be an ∀Exp+Res proof. We transform it into an IR-calc proof $D_1, \ldots, D_k$ as follows. If C_i is an axiom from clause C and assignment τ we construct D_i by taking the axiom in IR-calc of C and then instantiating with $\mathsf{inst}(\tau, C)$. If C_i is derived by resolving C_j, C_k over variable x^τ, then D_i is derived by resolving D_j, D_k over variable x^τ. This yields a valid IR-calc proof because $l^t \in D_i$ iff $l^t \in C_i$, which is preserved under applications of both rules. □

We now come to the simulation of a more powerful system than Q-resolution, namely LD-Q-Res from [1]. We show that this system is simulated by IRM-calc. The proof uses a similar, but more involved technique as in Theorem 6.

Theorem 8. *IRM-calc p-simulates LD-Q-Res.*

Proof (Sketch). Consider an LD-Q-Res refutation $C_1, \ldots, C_n$. We construct clauses $D_1, \ldots, D_n$, which will form the skeleton of the IRM-calc proof. The construction will preserve the following four invariants for $i = 1, \ldots, n$.

(1) For an existential literal l, it holds that $l \in C_i$ iff $l^t \in D_i$ for some t.
(2) The clause D_i has no literals l^{t_1} and l^{t_2} such that $t_1 \neq t_2$.
(3) If $l^t \in D_i$ with $0/u \in t$, then $u \in C_i$ or $u^* \in C_i$, likewise if $l^t \in D_i$ with $1/u \in t$, then $\neg u \in C_i$ or $u^* \in C_i$.
(4) If $l^t \in D_i$ with $*/u \in t$, then $u^* \in C_i$.

The actual construction proceeds as follows. If C_i is an axiom, D_i is constructed by the axiom rule from the same clause. If C_i is a $\forall$-reduction of C_j with $j < i$, then we set D_i equal to D_j. If C_i is obtained by a resolution step from C_j and C_k with $j < k < i$, the clause D_i is obtained by a resolution step from D_j and D_k, yielding clause K, and by performing some additional steps on K. Firstly, we let $\theta = \{c/u \mid c \in \{0,1\}, c/u \in t, l^t \in K\} \cup \{0/u \mid */u \in t, l^t \in K\}$ and perform instantiation on K by substitutions in θ, in any order, to derive K'. Like so, all annotations in K' have the same domain. We merge all pairs of literals $l^\sigma, l^\tau \in K'$ with $\tau \neq \sigma$ (in any order) to derive D_i.

To show that this construction yields a valid IRM-calc refutation, we first need to prove the invariants above. This proceeds by induction on i. We omit the base case and the $\forall$-reduction and just sketch the case of a resolution step.

For this consider C_j, C_k being resolved in LD-Q-Res to obtain C_i. As only the resolved variable is removed, which is removed completely due to condition (2), D_i fulfills (1). By induction hypothesis we know that there can be at most two copies of each variable when we derive K. Their annotations have the same domain in K', because instantiation by θ applies the entire domain of all annotations in the clause to all its literals. It then follows that all copies of identical literals are merged into one literal in D_i. Therefore (2) holds for D_i.

To prove (3) consider the case where $l^t \in D_i$ with $0/u \in t$. The case with $1/u \in t$ is analogous. We know that $0/u$ appearing in D_i means that $0/u$ must appear in K' as merging cannot produce a new annotation $0/u$. Existence of $0/u$ in K' means that either $*/u$ appears in K or $0/u$ appears in K. No new annotations are created in a resolution step, so either $*/u$ or $0/u$ must appear in one or more of D_j, D_k. By induction hypothesis this means that u or u^* appears in $C_j \cup C_k$, hence also in C_i.

To show condition (4), let $l^t \in D_i$ with $*/u \in t$. Then either $*/u$ is present in K', or $0/u$ and $1/u$ are present in K' and will be merged. In the first case it is clear that some $*/u$ annotation appears in K and thus in D_j or in D_k, in which case from (4) of the induction hypothesis u^* must appear in C_i. In the second case it is possible that $0/u$ in K' was obtained from $*/u$ in K. Thus as already argued, u^* must appear in C_i. If instead $1/u$, $0/u$ are both present in K then they must come from the original clauses D_j, D_k. If they both appear in the same clause D_j, then by condition (3) it must be the case that u^* appears in C_j and thus in C_i. If, however, they appear in different clauses, then by (3) either of the clauses C_j, C_k contains u^* or they contain literals over u of opposite polarity. Both situations merge the literals to $u^* \in C_i$.

We now show that these invariants imply that we indeed obtain a valid IRM-calc proof. We only need to consider the resolution steps. Suppose $x^{t_1} \in D_j$ and $\neg x^{t_2} \in D_k$ where C_j and C_k are resolved on x to get C_i in the LD-Q-Res proof. To perform the resolution step between D_j and D_k we need to ensure that we do not have $c/u \in t_1$, $d/u \in t_2$ where $c \neq d$ or $c = d = *$. Assume on the contrary that $*/u \in t_1$ and $c/u \in t_2$. By (4) we have $u^* \in C_j$, and by (3) some literal of u is in C_k. But as $\mathsf{lv}(u) < \mathsf{lv}(x)$ the LD-resolution of C_j and C_k on variable x is forbidden, giving a contradiction. Similarly, if there is $0/u \in t_1$ and $1/u \in t_2$,

then either we get the same situation or we have two opposite literals of u in the different clauses C_j, C_k. In either case the resolution of C_j, C_k is forbidden. Hence the IRM-calc proof is correct.

It is not difficult to see that the IRM-calc proof is indeed a refutation and all steps of the construction can be performed in polynomial time, thus we obtain a p-simulation. □

6 Conclusion

This paper introduces two novel calculi for quantified Boolean formulas. Both of these calculi are anchored in a common framework of *annotated clauses*. The first calculus, IR-calc, provides the rules of resolution and instantiation of clauses. The second calculus, IRM-calc, additionally enables *merging* literals with contradictory annotations. The paper demonstrates that the simple calculus IR-calc already p-simulates Q-resolution and the expansion-based system ∀Exp+Res. The extended version IRM-calc additionally p-simulates long-distance Q-resolution. The paper further demonstrates that refutations in the introduced calculi enable generation of winning strategies of the universal player—a favorable property from a practical perspective [1].

The contribution of the paper is both practical and theoretical. From a practical perspective, a calculus unifying the existing calculi for QBF enables a uniform certification of off-the-shelf QBF solvers. From a theoretical perspective, a unifying calculus provides an underpinning necessary for complexity characterizations of existing solvers as well as for furthering our understanding of the strengths of the underlying proof systems.

Acknowledgments. This work was supported by FCT grants ATTEST (CMU-PT-/ELE/0009/2009), POLARIS (PTDC/EIA-CCO/123051/2010), INESC-ID's multiannual PIDDAC funding PEst-OE/EEI/LA0021/2013, grant no. 48138 from the John Templeton Foundation, and a Doctoral Training Grant from EPSRC (2nd author).

References

1. Balabanov, V., Jiang, J.H.R.: Unified QBF certification and its applications. Formal Methods in System Design 41(1), 45–65 (2012)
2. Beame, P., Kautz, H.A., Sabharwal, A.: Towards understanding and harnessing the potential of clause learning. J. Artif. Intell. Res. (JAIR) 22, 319–351 (2004)
3. Benedetti, M.: Evaluating QBFs via symbolic Skolemization. In: LPAR (2004)
4. Benedetti, M., Mangassarian, H.: QBF-based formal verification: Experience and perspectives. JSAT 5(1-4), 133–191 (2008)
5. Biere, A.: Resolve and expand. In: SAT. pp. 238–246 (2004)
6. Buss, S.R.: Towards NP-P via proof complexity and search. Ann. Pure Appl. Logic 163(7), 906–917 (2012)
7. Buss, S.R., Hoffmann, J., Johannsen, J.: Resolution trees with lemmas: Resolution refinements that characterize DLL algorithms with clause learning. Logical Methods in Computer Science 4(4) (2008)

8. Cook, S.A., Reckhow, R.A.: The relative efficiency of propositional proof systems. J. Symb. Log. 44(1), 36–50 (1979)
9. Egly, U.: On sequent systems and resolution for QBFs. In: SAT. pp. 100–113 (2012)
10. Egly, U., Lonsing, F., Widl, M.: Long-distance resolution: Proof generation and strategy extraction in search-based QBF solving. In: LPAR (2013)
11. Giunchiglia, E., Narizzano, M., Tacchella, A.: Clause/term resolution and learning in the evaluation of quantified Boolean formulas. JAIR 26(1), 371–416 (2006)
12. Giunchiglia, E., Marin, P., Narizzano, M.: Reasoning with quantified boolean formulas. In: Handbook of Satisfiability, pp. 761–780. IOS Press (2009)
13. Goultiaeva, A., Van Gelder, A., Bacchus, F.: A uniform approach for generating proofs and strategies for both true and false QBF formulas. In: IJCAI (2011)
14. Hertel, P., Bacchus, F., Pitassi, T., Van Gelder, A.: Clause learning can effectively p-simulate general propositional resolution. In: AAAI (2008)
15. Janota, M., Grigore, R., Marques-Silva, J.: On QBF proofs and preprocessing. In: LPAR. pp. 473–489 (2013)
16. Janota, M., Klieber, W., Marques-Silva, J., Clarke, E.M.: Solving QBF with counterexample guided refinement. In: SAT. pp. 114–128 (2012)
17. Janota, M., Marques-Silva, J.: ∀Exp+Res does not P-Simulate Q-resolution. International Workshop on Quantified Boolean Formulas (2013)
18. Janota, M., Marques-Silva, J.: On propositional QBF expansions and Q-resolution. In: SAT. pp. 67–82 (2013)
19. Kleine Büning, H., Bubeck, U.: Theory of quantified boolean formulas. In: Handbook of Satisfiability, pp. 735–760. IOS Press (2009)
20. Kleine Büning, H., Karpinski, M., Flögel, A.: Resolution for quantified Boolean formulas. Inf. Comput. 117(1), 12–18 (1995)
21. Kleine Büning, H., Subramani, K., Zhao, X.: Boolean functions as models for quantified boolean formulas. J. Autom. Reasoning 39(1), 49–75 (2007)
22. Klieber, W., Janota, M., Marques-Silva, J., Clarke, E.M.: Solving QBF with free variables. In: Schulte, C. (ed.) CP. vol. 8124, pp. 415–431. Springer (2013)
23. Klieber, W., Sapra, S., Gao, S., Clarke, E.M.: A non-prenex, non-clausal QBF solver with game-state learning. In: SAT (2010)
24. Krajíček, J., Pudlák, P.: Quantified propositional calculi and fragments of bounded arithmetic. Mathematical Logic Quarterly 36(1), 29–46 (1990)
25. Marques Silva, J.P., Lynce, I., Malik, S.: Conflict-driven clause learning SAT solvers. In: Handbook of Satisfiability. IOS Press (2009)
26. Pipatsrisawat, K., Darwiche, A.: On the power of clause-learning SAT solvers as resolution engines. Artif. Intell. 175(2), 512–525 (2011)
27. Rintanen, J.: Asymptotically optimal encodings of conformant planning in QBF. In: AAAI. pp. 1045–1050. AAAI Press (2007)
28. Robinson, J.A.: A machine-oriented logic based on the resolution principle. J. ACM 12(1), 23–41 (1965)
29. Samer, M., Szeider, S.: Backdoor sets of quantified Boolean formulas. J. Autom. Reasoning 42(1), 77–97 (2009)
30. Slivovsky, F., Szeider, S.: Variable dependencies and Q-Resolution. International Workshop on Quantified Boolean Formulas (2013)
31. Van Gelder, A.: Variable independence and resolution paths for quantified Boolean formulas. In: Lee, J.H.M. (ed.) CP. vol. 6876, pp. 789–803. Springer (2011)
32. Van Gelder, A.: Contributions to the theory of practical quantified Boolean formula solving. In: Milano, M. (ed.) CP. vol. 7514, pp. 647–663. Springer (2012)
33. Zhang, L., Malik, S.: Conflict driven learning in a quantified Boolean satisfiability solver. In: ICCAD. pp. 442–449 (2002)