Electronic Colloquium on Computational Complexity, Report No. 109 (2021)

# QRAT Polynomially Simulates Merge Resolution.

**Sravanthi Chede**
Department of Computer Science and Engineering, IIT Ropar, India
sravanthi.20csz0001@iitrpr.ac.in

**Anil Shukla**
Department of Computer Science and Engineering, IIT Ropar, India
anilshukla@iitrpr.ac.in

## Abstract

Merge Resolution (MRes [6]) is a refutational proof system for quantified Boolean formulas (QBF). Each line of MRes consists of clauses with only existential literals, together with information of countermodels stored as merge maps. As a result, MRes has strategy extraction by design. The QRAT [18] proof system was designed to capture QBF preprocessing. QRAT can simulate both the expansion-based proof system ∀Exp+Res and CDCL-based QBF proof system LD-Q-Res.

A family of false QBFs called SquaredEquality formulas were introduced in [6] and shown to be easy for MRes but need exponential size proofs in Q-Res, QU-Res, CP+∀red, ∀Exp+Res, IR-calc and reductionless LD-Q-Res. As a result none of these systems can simulate MRes. In this paper, we show a short QRAT refutation of the SquaredEquality formulas. We further show that QRAT strictly p-simulates MRes. Besides highlighting the power of QRAT system, this work also presents the first simulation result for MRes.

**2012 ACM Subject Classification** Theory of computation Proof complexity

**Keywords and phrases** Proof Complexity, QBF, Simulation, QRAT, Merge Resolution

## 1 Introduction

Quantified Boolean formulas (QBF) extend propositional logic with quantifications, there exists (∃) and for all (∀). QBF proof complexity deals with understanding the limitations and strength of various QBF solving approaches. In the literature, there exists mainly two solving approaches i.e. Conflict-Driven-Clause-Learning (CDCL) and expansion-based solving. Several QBF proof systems have been developed to capture these solving approaches. Q-resolution (Q-Res) [22] is the base of CDCL-based approach. It is further extended to QU-resolution (QU-Res) [16] and Long-Distance-resolution (LD-Q-Res) [2]. On the other hand, proof system ∀Exp+Res [19] is the base of expansion-based solving. It is further extended to powerful proof systems IR-calc [8] and IRM-calc [8]. The simulation orders of these proof systems are well studied in the literature [9, Figure 1].

Recently, a new proof system Merge resolution (MRes) [6] has been developed. It follows a different QBF-solving approach. In MRes, winning strategies for the universal player are explicitly represented within the proof in the form of deterministic branching programs, known as merge maps [6]. MRes builds partial strategies at each line of the proof such that the strategy at the last line (corresponding to the empty clause) forms the complete countermodel for the input QBF. As a result, MRes admits strategy extraction by design. While performing resolution steps, MRes merges the partial strategies of the two hypotheses carefully if their corresponding merge maps are isomorphic or consistent. Note that whether two merge maps are isomorphic or consistent can be checked efficiently. This allows those resolution steps to be performed in MRes which would have been blocked in LD-Q-Res.

To be precise, in LD-Q-Res universal variables $u$ could appear in both polarities in the hypotheses and get merged in the resolvent provided $u$ appears in the right of the pivot variable in the quantifier prefix. MRes relaxed this restriction by allowing resolution steps even if $u$ is on the left of the pivot variable provided the merge maps of $u$ in both the hypotheses

ISSN 1433-8092

are isomorphic. This makes MRes powerful as compared to reductionless LD-Q-Res [11, 23]. In fact there exists a family of false QBFs SquaredEquality formulas (Definition 7) with short refutations in MRes [6] but require exponential size refutations in Q-Res, QU-Res, CP+∀red [10], ∀Exp+Res, IR-calc [4, 5] and reductionless LD-Q-Res [6]. Therefore, none of these proof systems can simulate MRes.

Quantified Resolution Asymmetric Tautologies (QRAT) proof system is introduced in [18] to capture the preprocessing steps performed by several QBF-solvers. It has been shown in [18] that QRAT can efficiently simulate all the existing preprocessing steps used by present-day QBF solvers. Recently, it has been shown that QRAT can simulate both the expansion-based proof system ∀Exp+Res [21] and CDCL-based proof system LD-Q-Res [20]. Since QRAT allows resolution steps with universal variables as pivot, it simulates QU-Res as well [20]. It is also known that QRAT is strictly stronger than ∀Exp+Res, LD-Q-Res and QU-Res [20, Figure 2].

In this short paper, we extend the importance of QRAT among QBF proof systems by showing that QRAT even polynomially simulates MRes. We also show that refuting the SquaredEquality formulas in QRAT is easy. Thus the semantic structure of these formulas which makes it harder to refute in all other proof systems is not a restriction for QRAT. We explain these contributions in the following subsection.

## 1.1 Our contributions

**(a)** **Short QRAT refutation of SquaredEquality formulas:** SquaredEquality formulas, a variant of equality formulas [5], have been defined in [6] to show that MRes is strictly stronger than reductionless LD-Q-Res [11, 23]. The original equality formulas which are hard for Q-Res but easy for LD-Q-Res have been extended in a way that prohibits the resolution step in reductionless LD-Q-Res but not in MRes.

In this paper, we show that the SquaredEquality formulas have a short refutation in QRAT (Theorem 8). Also, since the original equality formulas are easy for LD-Q-Res and QRAT can simulate LD-Q-Res, the formulas are easy for QRAT as well.

Note that, all other known families of false QBFs used to establish the incomparability results among QBF proof systems are easy for QRAT: KBKF [22, Theorem 3.2] formulas are easy for QU-Res [16, Example 5.5] hence they are easy for QRAT. Similarly, QPARITY [9] formulas are easy for ∀Exp+Res [9, Lemma 15] and hence easy for QRAT. Variants of these formulas were used to show the incomparability results among proof systems known to be simulated by QRAT. Hence these formulas are also easy for QRAT. Thus the presented short QRAT refutation of the SquaredEquality formulas makes this formulas also easy for QRAT.

**(b)** **QRAT polynomially simulates MRes:** It has been shown that MRes can simulate reductionless LD-Q-Res [6]. However, none of the proof systems Q-Res, QU-Res, CP+∀red, ∀Exp+Res, IR-calc and reductionless LD-Q-Res are capable of simulating MRes. The difficulty for these proof systems lies in simulating the axiom steps of MRes. To be precise, MRes gets rid of all the universal variables from the input clauses just by maintaining the partial strategies for them. On the other hand, the above mentioned proof systems have different and restricted rules for handling the universal variables. For example, Q-Res and QU-Res use universal reduction(UR) rule, which allows dropping a universal variable only if it is not blocked. Similarly, expansion-based proof systems like ∀Exp+Res and IR-calc handle the universal variables by introducing the annotated existential variables only.

In this paper, we show how QRAT handles this hurdle and polynomially simulates MRes. We show this by proving that the downloaded clauses in MRes proofs are all Asymmetric Tautology (AT) (Definition 5) with respect to the input QBF (Lemma 11). Therefore, they can easily be added in QRAT. Since the resolution step can be easily simulated by QRAT (Observation 6), the remaining resolution steps in MRes refutation can also be simulated (Theorem 10).

**(c) Emphasizing the importance of QRAT among QBF proof systems:** QRAT has been shown to simulate varieties of QBF solving approaches. That is on one hand, QRAT can simulate the expansion-based system ∀Exp+Res and on the other, it can simulate the powerful CDCL-based system LD-Q-Res. Since MRes is based on an entirely different QBF-solving approach; by showing that QRAT can polynomially simulate MRes, the paper extends the importance of QRAT system.

QRAT is a possible candidate for the universal checking format which can verify all existing QBF-solving techniques [13]. Our simulation result is a small step in this direction. (The other possible candidate is the extended Frege for QBFs, denoted as, eFrege+∀red [13, Conjecture 1]). For the simulation order and incomparabilities involving QRAT and several QBF proof systems, refer Figure 1.

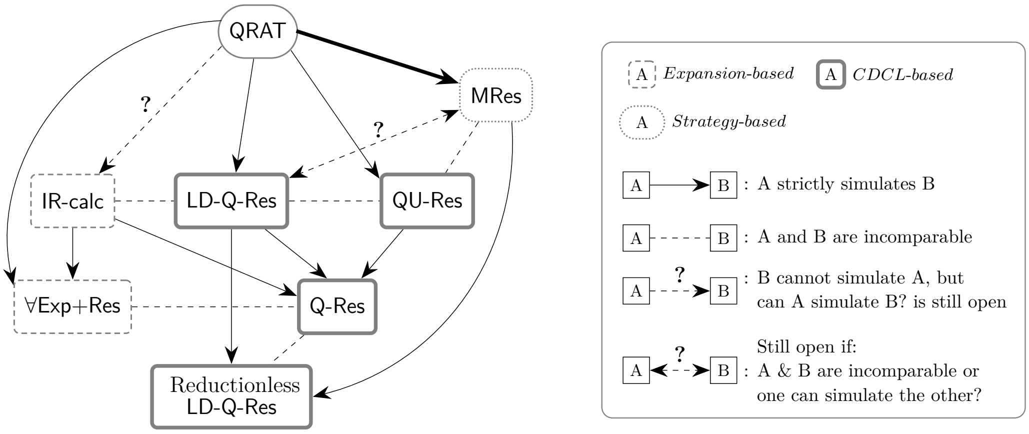
QRAT
MRes
?
?
IR-calc
LD-Q-Res
QU-Res
∀Exp+Res
Q-Res
Reductionless LD-Q-Res
A Expansion-based
A CDCL-based
A Strategy-based
A B : A strictly simulates B
A B : A and B are incomparable
A ? B : B cannot simulate A, but can A simulate B? is still open
A ? B : Still open if: A & B are incomparable or one can simulate the other?

**Figure 1** Simulation order of QBF proof systems, with our new result shown in bold. MRes simulation of reductionless LD-Q-Res is shown in [6]. QRAT simulation of ∀Exp+Res, LD-Q-Res and QU-Res are shown in [21, 20, 20] respectively. The incomparability result of reductionless LD-Q-Res and Q-Res was shown in [23]. For other known relations refer [9, Figure 1]

## 1.2 Organisation of the paper

In Section 2, we denote all important notations and preliminaries used in the paper. We define MRes in Section 2.1 and QRAT in Section 2.2. In Section 3, we define the SquaredEquality formulas and give a short QRAT proof for the same in (Theorem 8). We prove the QRAT simulation of MRes in Section 4. Finally, we conclude and discuss further open problems in Section 5.

## 2 Notations and Prerequisites

A clause $C$ is a disjunction of literals and a conjunctive normal form (CNF) $F$ is a conjunction of clauses. A clause $D$ is a sub-clause of $C$ if every literal of $D$ are also a literal of $C$. A unit clause is a clause with only one literal in it. We denote the empty clause by $\bot$. $\mathrm{vars}(C)$ is a set of all variables in $C$ and $var(F) = \cup_{C \in F}\ vars(C)$. $\overline{C}$ for a clause $C$, is a conjunction of negation of literals in $C$.

A **proof system** [15] for a non-empty language $L \subseteq \{0,1\}^*$ is a polynomial time computable function $f : \{0,1\}^* \rightarrow \{0,1\}^*$ such that $\mathrm{Range}(f) = L$. For string $x \in L$, we say a string $w \in \{0,1\}^*$ is an $f$-proof of $x$ if $f(w) = x$. A proof system $f$ for $L$ is complete iff for every $x \in L$ we have a corresponding $f$-proof for $x$. A proof system $f$ for $L$ is sound iff the existence of an $f$-proof for $x$ implies that $x \in L$.

A proof system $f$ p-simulates (polynomially simulates) another proof system $g$ (i.e., $f \leq_p g$) if both prove the same language $L$ and every $g$-proof of input $x \in L$ can be translated into an $f$-proof for the same input in time that is polynomial w.r.t size of the $g$-proof. Otherwise, we say that $f$ do not simulate $g$ ($f \not\leq_p g$). We say that a proof system $f$ strictly simulates a proof system $g$ if $f \leq_p g$ but $g \not\leq_p f$. Proof systems $f$ and $g$ are said to be incomparable, if none of them can simulate the other. $f$ and $g$ proof systems are said to be p-equivalent if both $f \leq_p g$ and $g \leq_p f$ hold.

Proof systems for $L = UNSAT/TAUT$ are called propositional proof systems. For example, the resolution proof system is defined as follows:

▶ **Definition 1.** ***Resolution proof system:*** *Resolution proof system [12, 24] is the most studied propositional proof system. The lines in this proof system are clauses. Given a CNF formula $F$, it can derive new clauses using the following inference rule, also known as the resolution rule:* $\frac{C \vee x \qquad D \vee \neg x}{C \vee D}$, *where $C$ and $D$ are clauses and $x$ is the pivot variable being resolved. The clause $C \vee D$ is called the resolvent. For the rest of this paper we denote this step as Res($(C \vee x), (D \vee \overline{x}), x$).*

Proof systems for $L$ = FQBFs/ TQBFs are said to be QBF proof systems where, FQBFs (TQBFs) denote the set of all false (true) QBFs. For example, Q-Res, QU-Res, etc. Before defining them we first define the QBFs.

**Quantified Boolean formulas:** QBFs are an extension of the propositional Boolean formulas where each variable is quantified with one of $\{\exists, \forall\}$, the symbols having their general semantic definition of existential and universal quantifier respectively.

In this paper, we assume that QBFs are in closed prenex form i.e., we consider the form $Q_1X_1...Q_kX_k.\ \phi(X)$ , where $X_i$ are pairwise disjoint sets of variables; $Q_i \in \{\exists,\ \forall\}$ and $Q_i \neq Q_{i+1}$, and $\phi(X)$ is in CNF form over $X = X_1 \cup \cdots \cup X_k$, called the matrix of the QBF. We denote QBFs as $Q.\phi$ in this paper, where $Q$ is the quantifier prefix.

If a variable $x$ is in the set $X_i$, we say that $x$ is at level $i$ and write $lv(x) = i$. Note that the quantifier of a literal is the quantifier of the corresponding variable. Given a literal $\ell$ with quantifier $Q_i$ and a literal $k$ with quantifier $Q_j$ , we say that $\ell$ occurs to left of $k$ and write $\ell \leq_Q k$ if $i \leq j$. Likewise, we say that $\ell$ occurs to right of $k$ and write $\ell >_Q k$ if $i > j$.

An assignment tree of a QBF $Q.\phi$ is a complete binary tree of depth $|vars(\phi)|$ where each level is dedicated to a variable in the order of the prefix. If the variable is existential, then the level is said to be an existential level and each node of this level is said to be an existential node. Similarly we have universal levels and universal nodes. If a level is dedicated for a variable (say $x$), every node in that level will have 2 outgoing edges for the level below marked with $x$ and $\overline{x}$ respectively except for the level corresponding to leaf nodes. Each node is labelled with either 1 or 0 eventually as follows: the path from root to leaf is a total assignment to all variables which if evaluate $\phi$ to 1 (resp. 0), the corresponding leaf node is labelled with 1 (resp. 0). Rest of the nodes are labelled from bottom up, where the existential nodes act as $OR$ gates and the universal nodes act as $AND$ gates. A **model** (resp. **countermodel**) is a sub-tree of the assignment tree where every existential node has exactly one child (resp. both children) and universal node has both the children (resp. exactly one child), furthermore every node in this tree is marked with 1 (resp. 0). True-QBFs have at least one model and false-QBFs have at least one countermodel.

**QBFs as a game:** QBFs are often seen as a game between the universal and the existential player i.e. in the $i^{th}$ step the player $Q_i$ assigns values to the variables $X_i$. At the end, the existential (resp. universal) player wins if substituting this total assignment of variables in $\phi$ evaluates to 1 (resp. 0).

For a QBF $Q.\phi$, a **strategy** of universal (resp. existential) player is a decision function that returns the assignment to all universal (resp. existential) variables of $Q$, where the decision for each $u$ depends only on the variables to the left of it in the quantifier prefix $Q$.

**Winning strategy** for a player is a strategy which makes this player win against every assignment of the other player.The term ‘winning strategy’ is often used instead of model (in case of the existential player) and countermodel (in case of the universal player). A QBF is false (true) iff there exists a winning strategy for the universal (existential) player [1].

We say that a QBF proof system $f$ admits **strategy extraction** if from every $f$-proof ($f$-refutation) of a true (false) QBF $Q.\phi$ one can extract a winning strategy for the existential (universal) player efficiently w.r.t. the size of the $f$-proof ($f$-refutation).

Strategy extraction is one of the most important lower bound techniques in QBF proof complexity. If a proof system $f$ admits strategy extraction, then every QBF with hard strategies must have long $f$-proofs.

Now, let us define few important QBF-proof systems:

**Q-Res:** [22] Q-Res is the extension of the resolution proof system for QBFs. It has two rules namely resolution and universal reduction. The resolution rule is the same as defined before in (Definition 1) i.e. $\text{Res}(C_a, C_b, x)$; the only restrictions being that the pivot variable $x$ should be an existential variable and that the resolvent clause should not be a tautology.

The **Universal Reduction** (UR) of Q-Res is the rule that allows dropping of universal literal $u$ from a clause $C$ in the QBF provided no existential literal $\ell \in C$ appears to the right of $u$ in the quantifier prefix. In this case, we say $u$ is not blocked in $C$, otherwise it is blocked.

**QU-Res:** [16] QU-Res is an extension of Q-Res which allows resolution on universal variables as well.

## 2.1 MRes proof system [6]

MRes is a proof system for false QBFs introduced in [6]. We describe MRes briefly in this section, please refer to [6] for its formal definition.

A MRes refutation of a QBF $Q.\phi$ consists of a sequence of lines where each line $L_i$ consists of a clause $C_i$ and a map (${M_i}^u$) for each universal variable $u \in Q$. At any given point in the proof, $C_i$ consists of only existential literals and ${M_i}^u$ gives the partial strategy of universal variable $u$ based on the existential variables which lie to the left of $u$ in the quantifier prefix $Q$.

The **Merge-maps** ${M_i}^u$ can be either directly $i \mapsto \{u/\overline{u}/*\}$ or it can be of the form $i \mapsto (x, a, b)$ (read as ‘if $x = 0$ then goto $a$ else goto $b$’) where $x$ is an existential variable and

$a, b < i$ are line indices from previous lines of the MRes proof.

Merge-maps can be represented as graphs where the node labels are line indices and the edges are labelled by existential literals. For example, for the merge map rule $i \mapsto (x, a, b)$, we have an edge $i \rightarrow a$ in the graph with label $\overline{x}$ and an edge $i \rightarrow b$ with label $x$. Let $i$ be a line index which is a leaf node in the graph, then it also has an additional label from $\{u, \overline{u}, *\}$ corresponding to the rule $i \mapsto \{u/\overline{u}/*\}$.

The following two properties can be easily checked on merge-maps:

**Isomorphism:** Two merge maps ${M_a}^u$ and ${M_b}^u$ are isomorphic (written ${M_a}^u \simeq {M_b}^u$) if and only if there exists a bijection mapping from the line numbers of one to those of another when represented as graphs. In other words, two isomorphic merge maps represent the same strategy.

**Consistency:** Two merge maps ${M_a}^u$ and ${M_b}^u$ are consistent (written ${M_a}^u \bowtie {M_b}^u$) if and only if for every common line index (say $i$) in both maps, it holds that ${M_a}^u(i) = {M_b}^u(i)$.

The following two functions are defined on merge-maps:

**Select(${M_a}^u, {M_b}^u$):** It is defined only when ${M_a}^u \simeq {M_b}^u$ or when either of them is trivial (i.e. $a \mapsto *$ or/& $b \mapsto *$ ). In such a case, it returns ${M_a}^u$ (if ${M_a}^u$ is not trivial), otherwise returns ${M_b}^u$.

**Merge(${M_a}^u, {M_b}^u, n, x$):** It is defined only when ${M_a}^u \bowtie {M_b}^u$ and $x$ is an existential variable and $n$ is a new line index strictly greater than both $a$ and $b$. In such a case, it returns a new merge-map which merges the same indice nodes into one node and adds a new node with the rule $n \mapsto (x, a, b)$. Note that the indices present only in one of the input maps (i.e. not-common) are retained in the new merged map as they were.

Now, we are ready to define the MRes proof system:

▶ **Definition 2** (**MRes proof system** [6]). *Let $\Phi := Q.\phi$ be a QBF with existential variables $X$ and universal variables $U$. MRes derivation of $\Phi$ is a sequence $\pi := L_1, ..., L_k$ of lines $L_i := (C_i, \{{M_i}^u : u \in U\})$ derived by one of the following steps:*

**(a)** ***Axiom**. There exists a clause in $C \in \phi$ such that $C_i$ is the existential sub-clause of $C$, and, for each $u \in U$, ${M_i}^u$ is the rule $i \mapsto$ the falsifying u-literal for $C$, if $u \notin C$ add the trivial rule $i \mapsto *$;* **or**,

**(b)** ***Resolution**. There exist integers $a, b < i$ and an existential pivot $x \in X$ such that $C_i$ = Res($C_a, C_b, x$), where one of the following must hold for every $u \in U$:*
*(i)* *${M_i}^u$ = select(${M_a}^u, {M_b}^u$) if defined;* ***or**,*
*(ii)* *$x <_Q u$ and ${M_i}^u$ = merge(${M_a}^u, {M_b}^u$), $i, x$).*

*The final line $L_k$ is the conclusion of $\pi$, and $\pi$ is a refutation of $\Phi$ iff $C_k = \bot$. In this case observe that $\{{M_k}^u : u \in U\}$ is a winning strategy for the universal player.*

It's known that MRes is sound and complete for false QBFs [6, Section 4.3]. We outline QRAT proof system in the next section.

## 2.2 QRAT proof system [18]

The QRAT proof system was introduced to capture the state-of-the-art techniques used in current day QBF-solvers [18]. We give a brief summary of its rules. We need the following definitions:

▶ **Definition 3.** *For a CNF formula $F$, **unit propagation** (represented by $\vdash_1$ or unit-propagation($F$)) simplifies $F$ on unit clauses; that is for every unit clause $(\ell) \in F$, it assigns $\ell$ to 1 in all clauses of $F$. i.e. removes all clauses that contain the literal $\ell$ from the set $F$*

*and drops the literal $\overline{\ell}$ from all clauses in $F$. It keeps repeating this until no unit clause is left or an empty clause is derived.*

▶ **Definition 4** (Outer resolvent [18]). *Given two clauses $(C \vee \ell), (D \vee \overline{\ell})$ of a QBF $Q.\phi$, the* ***Outer Resolvent*** *$OR(Q,C,D,\ell)$ is the clause consisting of all literals in $C$ together with those literals of $D$ that occur to the left of $\ell$, i.e. $C \cup \{k \mid k \in D, k \leq_Q \ell\}$.*

▶ **Definition 5** (Asymmetric Tautology (AT)). *Clause $C$ is an AT w.r.t. to CNF $\phi$ iff $\phi \vdash_1 C$. Alternatively, $C$ is an AT w.r.t. $\phi$ iff $\bot \in$ unit-propagation$(\phi \wedge \overline{C})$. A clause $C$ is an AT w.r.t. a QBF $Q.\phi$ if it is an AT w.r.t. $\phi$.*

**QRAT-clause & QRAT-literal:** A clause $C \vee \ell$ is QRAT-clause w.r.t. a QBF $Q.\phi$ if for every $D \vee \overline{\ell} \in \phi$ the OR($Q$,$C$,$D$,$\ell$) is an AT w.r.t. $\phi$. We say that $\ell$ is the QRAT-literal in $C$.

If a clause $C$ contains an existential QRAT-literal, it has been shown in [18] that $C$ can be removed (called QRATE rule) or added (called QRATA rule) without affecting the satisfiability of the QBF. Also, if a clause $C$ contains a universal QRAT-literal $\ell$, then dropping $\ell$ from $C$ (called QRATU rule) is also a satisfiability preserving step. Note that a clause $C$, which is an AT w.r.t. a QBF $\Phi = Q.\phi$ is also a QRAT-clause on any literal belonging to $C$ w.r.t. $\Phi$ [18]. Additionally, QRAT allows elimination of any clause at any point in the proof [21].

The remaining rule in QRAT is EUR; to define it we need the following:

**Extended inner clause (EIC):** For a QBF $Q.\phi$ where $C \in \phi$ and $\ell \in C$, EIC$(Q, C, \ell)$ is the final clause obtained when repeatedly performing the following: for every existential literal $k \in C$ which is to the right of $\ell$ in $Q$, extend $C$ by all the right literals of $\ell$ in the clauses $D \in \phi$ with $\overline{k} \in D$ (also include $\overline{\ell}$ in $C$ if $\overline{\ell}$ also $\in$ such $D$).

**Extended Universal Reduction (EUR):** Given a QBF $Q.\phi$, for a clause $C \in \phi$ with a universal literal $\ell \in C$ such that $\overline{\ell} \notin$ EIC$(Q, C, \ell)$, the literal $\ell$ can be dropped from $C$ under the Extended Universal Reduction (EUR) rule of QRAT.

Given a QBF $\Phi = Q.\phi$, a sequence of clauses is called a QRAT refutation of $\Phi$, if they are derived using the above mentioned rules and the last clause in the sequence is $\bot$.

In this paper we show that even the full power of QRAT is not required to prove the SquaredEquality formulas or to simulate MRes. This restricted variant is referred to as QRAT (UR) in the literature [14] which allows all the QRAT rules but uses universal reduction (UR) instead of the powerful EUR rule. In fact, if we allow the definition that clauses with only universal literals are $\bot$; then QRAT simulation of MRes does not even require the **UR** rule, only the QRATU rule is sufficient.

Before moving on to the next section, we state the following observation which is useful for the upcoming proofs.

▶ **Observation 6** ([18]). *Consider a resolution step Res($(C \vee x), (D \vee \overline{x}), x$). QRAT can easily simulate the resolution steps by directly adding the resolvent clause to the QBF $Q.\phi$ as it is an AT w.r.t. $\phi$. This is true as unit propagation of the resolvent $(C \vee D)$ in $\phi$ derives $(x \wedge \overline{x}) = \bot$. Note that the above argument is valid for universal pivot variables as well. This implies, QRAT can simulate QU-Res.*

## 3 SquaredEquality Formulas [6]

In [7], it was stated that, there exists a family of false QBFs, the SquaredEquality formulas, with short proofs in MRes but requiring exponential size in Q-Res, QU-Res, CP+∀red, ∀Exp+Res, IR-calc and reductionless LD-Q-Res. In the next subsection, we show that these formulas are easy for QRAT proof system as well. We next present its definition.

▶ **Definition 7** (SquaredEquality Formulas [6]). *The squared equality family is the QBF family whose* $n^{th}$ *instance* $EQ^2(n) := Q(n).eq^2(n)$*, it has the prefix*

$$Q(n) := \exists\{x_1, y_1, ..., x_n, y_n\}\forall\{u_1, v_1, ..., u_n, v_n\}\exists\{t_{i,j} : i, j \in [n]\},$$

*and the matrix* $eq^2(n)$ *consisting of the clauses:* *Labels:*

$$\{x_i, y_j, u_i, v_j, t_{i,j}\}, \{x_i, \overline{y_j}, u_i, \overline{v_j}, t_{i,j}\}, \text{ for } i, j \in [n], \qquad C_{i,j}, C'_{i,j}$$

$$\{\overline{x_i}, y_j, \overline{u_i}, v_j, t_{i,j}\}, \{\overline{x_i}, \overline{y_j}, \overline{u_i}, \overline{v_j}, t_{i,j}\}, \text{ for } i, j \in [n], \qquad D_{i,j}, D'_{i,j}$$

$$(\overline{t_{i,j}} : i, j \in [n]). \qquad T$$

## 3.1 Short QRAT refutations of $EQ^2(n)$

In this section we give the first short QRAT refutation for SquaredEquality formulas.

▶ **Theorem 8.** *The SqauredEquality formulas have* $\mathcal{O}(n^2)$*-size QRAT refutations.*

**Proof.** Let $n \in \mathbb{N}$,where $\mathbb{N}$ is the set of all natural numbers. We construct a refutation in 3 stages. In the first stage, we drop all the universal variables in the formulas for the reason that they are QRAT-literals. In the second stage we derive unit clauses of all $t_{i,j}$'s using 3 resolutions steps each. In the last stage we successively resolve these unit clauses with the clause $T$ and derive an empty clause.

**(a)** **Stage 1:** we prove the following lemma first.

▶ **Lemma 9.** *All universal literals are QRAT-literals in SquaredEquality formulas and can be dropped by QRATU rule.*

**Proof.** Observe that in all the 4 type of clauses (i.e. $C_{i,j}, C'_{i,j}, D_{i,j}, D'_{i,j}$) the existential literal $x_i$ is always in the same clause as the universal literal $u_i$ and the literal $\overline{x_i}$ is always in the same clause as the literal $\overline{u_i}$. Same is with the existential variable $y_j$ and universal variable $v_j$. Moreover $x_i, y_j$ are always on the left of $u_i, v_j$ in the quantifier prefix.

Consider the $C_{i,j}$ type of clauses, they contain the universal literal $u_i$. The outer resolvents of these clauses can be with either $D_{i,j}$ or $D'_{i,j}$ which contain the literal $\overline{u_i}$. All these outer resolvents will have both $x_i$ and $\overline{x_i}$, i.e. they are a tautology. Hence all the outer resolvents are ATs, which makes $u_i$ a QRAT-literal in $C_{i,j}$. A similar argument can be made for each one of the 4 clauses as the primary clause. Thus all the $u_i$ variables can be dropped from the formulas.

Now, consider again the $C_{i,j}$ type of clauses, they contain the universal literal $v_j$. The outer resolvents of these clauses can be with either $C'_{i,j}$ or $D'_{i,j}$ which contain the literal $\overline{v_j}$. All these outer resolvents will have both $y_j$ and $\overline{y_j}$, i.e. they are a tautology. Hence they are all ATs, that makes $v_j$ a QRAT-literal in $C_{i,j}$. Similar arguments can be made for each one of the 4 clauses. So all the $v_j$ variables can be dropped from the formulas. ◀

Now using Lemma 9, we drop all universal variables in $\mathcal{O}(n^2)$ and obtain the following clauses: Labels:

$$\{x_i, y_j, t_{i,j}\}, \{x_i, \overline{y_j}, t_{i,j}\}, \text{ for } i, j \in [n], \qquad C''_{i,j}, C'''_{i,j}$$

$$\{\overline{x_i}, y_j, t_{i,j}\}, \{\overline{x_i}, \overline{y_j}, t_{i,j}\}, \text{ for } i, j \in [n], \qquad D''_{i,j}, D'''_{i,j}$$

**(b)** **Stage 2:** For every $i, j \in [n]$, we use 3 resolution rules on the corresponding clauses $C''_{i,j}$, $C'''_{i,j}$, $D''_{i,j}$ & $D'''_{i,j}$ to obtain the unit clause $(t_{i,j})$ as follows:

$$P_{i,j} = Res(C''_{i,j}, C'''_{i,j}, y_j) = \{x_i, t_{i.j}\}$$
$$Q_{i,j} = Res(D''_{i,j}, D'''_{i,j}, y_j) = \{\overline{x_i}, t_{i.j}\}$$
$$R_{i,j} = Res(P_{i,j}, Q_{i,j}, x_i) = \{t_{i,j}\}$$

Resolution clauses are AT in QRAT (Observation 6), so the above clauses ($P_{i,j}, Q_{i,j}, R_{i,j}$ in this order) can be added directly. This stage can be done in $\mathcal{O}(n^2)$ resolution steps.

**(c)** **Stage 3:** For every $i, j \in [n]$ we have already derived all the $n^2$ unit clauses $R_{i,j}$'s, using these clauses along with the input clause $T$ we may derive the empty clause $\bot$ in $\mathcal{O}(n^2)$ steps.

This completes the proof. Observe that SquaredEquality formulas have $\mathcal{O}(n^2)$ clauses, hence the QRAT refutation is indeed linear in the size of the formula. ◀

Now we proceed to our simulation result in the next section.

## 4 QRAT polynomially simulates MRes

In MRes, in addition to finding a winning strategy for the universal player, the proof system also derives the empty clause $\bot$ through a sequence of sound rules. That is at the end, MRes proves that the given QBF is false in two different ways simultaneously. Firstly, by providing through sound rules, a sequence of clauses with only existential literals $C_1, C_2 \ldots, C_k = \bot$. Secondly, by providing a countermodel for the QBF through merge-maps.

While deriving the clauses $C_i$ in the sequence above, MRes consults the corresponding partial strategies presented in the hypothesis and makes sure that they meet certain criteria (Definition 2) to maintain soundness. Therefore, this sequence of clauses depends on the partial strategies that the MRes proof is building. However, it is sufficient for any proof system to produce either of these proof types to prove the falseness of any QBF.

So, even if a proof system $f$ can efficiently simulate through its sound rules the sequence of clauses $C_1, C_2, \ldots, C_k$ then we can say that $f$ polynomially simulates MRes. $f$ is not required to build or consult the partial strategies built by MRes.

The other way of simulating MRes by a proof system $f$ would be to simulate the process of building the partial strategies of the MRes proof.

We show that QRAT can efficiently simulate MRes by simulating it's sequence of clauses $C_1, C_2, \ldots, C_k$, as is mentioned in the first process.

▶ **Theorem 10.** *QRAT polynomially simulates MRes.*

**Proof.** Given an MRes refutation $\pi = L_1, ..., L_k$ for a false QBF $\Phi = Q.\phi$ with $X$ (resp. $U$) as the set of existential (resp. universal) variables, where each $L_i = (C_i, \{{M_i}^u : u \in U\})$, we effectively compute a QRAT refutation $\Pi$ for the QBF $\Phi = Q.\phi$ as follows:

**(a)** **Axiom Steps:** For every axiom step in $\pi$ (say $L_i$), the following Lemma holds:

▶ **Lemma 11.** *The existential sub-clause of a clause $C \in \phi$ is AT w.r.t the QBF $\Phi$.*

**Proof.** Let $C = \{e_1, .., e_n, u_1, ..., u_m\}$ be an input clause in the QBF $Q.\phi$, where $e_1, .., e_n$ are existential literals and $u_1, .., u_m$ are universal literals with arbitrary order in $Q$. The existential sub-clause $C_i = \{e_1, .., e_n\}$ is an AT w.r.t $Q.\phi$: since $\overline{C_i} \wedge C \vdash_1 (u_1, ..., u_m) = \bot$. The clause $(u_1, ..., u_m)$ is the empty clause since all its literals are universal. ◀

Using Lemma 11, we can directly add the existential sub-clause $C_i$ belonging to the axiom step $L_i$ of MRes proof $\pi$ in the QRAT proof $\Pi$. This can be done for all the axiom steps in order as they appear in $\pi$.

**(b) Resolution Steps:** Resolution step in MRes is executed provided some conditions on hypothesis merge maps are met (Definition 2). So while simulating, QRAT only needs to simulate the resolution steps where soundness part is already taken care of by MRes via maintaining partial strategies through merge-maps.

The resolvent clause is known to be AT w.r.t. QBF $\Phi$, so can be directly added to $\Pi$ (Observation 6). When QRAT simulates the last line of $\pi$ through resolution, we get the corresponding $\perp$ in $\Pi$ as well.

This completes the simulation. Observe that $\Pi$ is a valid QRAT refutation of the input formula $\Phi$. Note that, the size of the QRAT proof is linear in the size of the corresponding MRes proof. Also observe that the sequence of clauses $C_1, ..., C_k$ in the lines of the MRes are in itself a valid QRAT proof! ◀

On the other hand, we observe that MRes is not powerful enough to efficiently simulate the QRAT proof system. To be precise, we have:

▶ **Observation 12.** *MRes cannot simulate QRAT.*

**Proof.** There exists a family of false QBFs KBKF-lq$[n]$ [3, Definition 3] which are shown to be hard for MRes in [7, Theorem 19], but easy for QU-Res [3, Theorem 2]. Since QRAT simulates QU-Res [20], these formulas are easy for QRAT. So this concludes that MRes cannot simulate QRAT. ◀

## 5 Conclusions and future work

QRAT proof system is capable of efficiently simulating both the expansion-based QBF-solving approach, i.e., ∀Exp+Res [21] and the CDCL-based QBF-solving approach LD-Q-Res [20]. It is also known that QRAT can simulate all the existing preprocessing techniques used by current QBF-solvers [18]. In this paper, we show that QRAT can even strictly simulate the new proof system which builds partial strategies into proofs, that is, the MRes proof system [6]. Thus extending the importance of QRAT among QBF proof systems. Also, we have given a short QRAT refutation for the SquaredEquality formulas introduced in [6].

Work in this domain still has many interesting open problems; we would like to mention a few of the same:

Although MRes was inspired from LD-Q-Res, it is still open if they are incomparable or if one can simulate the other.

QRAT simulation of ∀Exp+Res has been proven in [21], but it is still open whether or not QRAT can simulate it's powerful variant the IR-calc proof system? Note that IR-calc cannot simulate QRAT proof system since the former is incomparable with LD-Q-Res and QRAT simulates LD-Q-Res [20]. For the complexity landscape of these systems, refer Figure 1 in this paper.

Given a false QBF $\Phi = Q.\phi$, MRes builds a winning strategy of the universal player by design. In case, $\Phi$ has a computationally hard winning strategy, MRes refutation of $\Phi$ is also going to be large. As a result, proving lower bound for such QBFs in MRes is easy. On the other hand, QRAT does not admit strategy extraction for false QBFs unless P = PSPACE [14]. That is, there exists a family of false QBFs (the Select Formulas from [14, Section 4.1] which are easy for QRAT but have computationally hard universal winning strategies provided P $\neq$ PSPACE. As a result, establishing lower bound results for false QBFs in QRAT using the strategy extraction technique is not possible. In fact, proving a lower bound result in QRAT is still open. It should be noted that QRAT admits strategy extraction for true QBFs [17], therefore strategy extraction can still be used to prove QRAT lower bounds for true QBFs.

## References

1 Sanjeev Arora and Boaz Barak. *Computational Complexity - A Modern Approach*. Cambridge University Press, 2009. URL: `http://www.cambridge.org/catalogue/catalogue.asp?isbn=9780521424264`.

2 Valeriy Balabanov and Jie-Hong R. Jiang. Unified QBF certification and its applications. *Formal Methods in System Design*, 41(1):45–65, August 2012.

3 Valeriy Balabanov, Magdalena Widl, and Jie-Hong R. Jiang. QBF resolution systems and their proof complexities. In Carsten Sinz and Uwe Egly, editors, *Theory and Applications of Satisfiability Testing - SAT 2014 - 17th International Conference, Held as Part of the Vienna Summer of Logic, VSL 2014, Vienna, Austria, July 14-17, 2014. Proceedings*, volume 8561 of *Lecture Notes in Computer Science*, pages 154–169. Springer, 2014.

4 Olaf Beyersdorff and Joshua Blinkhorn. Lower bound techniques for QBF expansion. *Theory Comput. Syst.*, 64(3):400–421, 2020.

5 Olaf Beyersdorff, Joshua Blinkhorn, and Luke Hinde. Size, cost, and capacity: A semantic technique for hard random qbfs. *Log. Methods Comput. Sci.*, 15(1), 2019.

6 Olaf Beyersdorff, Joshua Blinkhorn, and Meena Mahajan. Building strategies into QBF proofs. *J. Autom. Reason.*, 65(1):125–154, 2021.

7 Olaf Beyersdorff, Joshua Blinkhorn, Meena Mahajan, Tomás Peitl, and Gaurav Sood. Hard QBFs for merge resolution. In *40th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2020, December 14-18, 2020, BITS Pilani, K K Birla Goa Campus, Goa, India (Virtual Conference)*, volume 182 of *LIPIcs*, 2020.

8 Olaf Beyersdorff, Leroy Chew, and Mikoláš Janota. On unification of QBF resolution-based calculi. In *Mathematical Foundations of Computer Science (MFCS)*, pages 81–93, 2014.

9 Olaf Beyersdorff, Leroy Chew, and Mikoláš Janota. Proof complexity of resolution-based QBF calculi. In *Proceedings of the 32nd International Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 76–89. LIPIcs, 2015.

10 Olaf Beyersdorff, Leroy Chew, Meena Mahajan, and Anil Shukla. Understanding cutting planes for qbfs. *Inf. Comput.*, 262:141–161, 2018.

11 Nikolaj Bjørner, Mikolás Janota, and William Klieber. On conflicts and strategies in QBF. In Ansgar Fehnker, Annabelle McIver, Geoff Sutcliffe, and Andrei Voronkov, editors, *20th International Conferences on Logic for Programming, Artificial Intelligence and Reasoning - Short Presentations, LPAR 2015, Suva, Fiji, November 24-28, 2015*, volume 35 of *EPiC Series in Computing*, pages 28–41. EasyChair, 2015.

12 A. Blake. *Canonical expressions in Boolean algebra*. PhD thesis, University of Chicago, 1937.

13 Leroy Chew. Hardness and optimality in QBF proof systems modulo NP. In *Theory and Applications of Satisfiability Testing - SAT 2021*, volume 12831 of *Lecture Notes in Computer Science*, pages 98–115. Springer, 2021.

14 Leroy Chew and Judith Clymo. How QBF expansion makes strategy extraction hard. In Nicolas Peltier and Viorica Sofronie-Stokkermans, editors, *Automated Reasoning - 10th International Joint Conference, IJCAR 2020, Paris, France, July 1-4, 2020, Proceedings, Part I*, volume 12166 of *Lecture Notes in Computer Science*, pages 66–82. Springer, 2020. URL: `https://doi.org/10.1007/978-3-030-51074-9_5`.

15 Stephen A. Cook and Robert A. Reckhow. The relative efficiency of propositional proof systems. *J. Symb. Log.*, 44(1):36–50, 1979. `doi:10.2307/2273702`.

16 Allen Van Gelder. Contributions to the theory of practical quantified boolean formula solving. In *Principles and Practice of Constraint Programming - 18th International Conference, CP 2012, Québec City, QC, Canada, October 8-12, 2012. Proceedings*, volume 7514 of *Lecture Notes in Computer Science*, pages 647–663. Springer, 2012.

17 Marijn Heule, Martina Seidl, and Armin Biere. Efficient extraction of skolem functions from QRAT proofs. In *Formal Methods in Computer-Aided Design, FMCAD 2014, Lausanne, Switzerland, October 21-24, 2014*, pages 107–114. IEEE, 2014.

18 Marijn J. H. Heule, Martina Seidl, and Armin Biere. Solution validation and extraction for QBF preprocessing. *J. Autom. Reason.*, 58(1):97–125, 2017.

19 Mikolás Janota and Joao Marques-Silva. Expansion-based QBF solving versus Q-resolution. *Theoretical Computer Science*, 577:25–42, 2015.

20 Benjamin Kiesl, Marijn J. H. Heule, and Martina Seidl. A little blocked literal goes a long way. In Serge Gaspers and Toby Walsh, editors, *Theory and Applications of Satisfiability Testing - SAT 2017 - 20th International Conference, Melbourne, VIC, Australia, August 28 - September 1, 2017, Proceedings*, volume 10491 of *Lecture Notes in Computer Science*, pages 281–297. Springer, 2017.

21 Benjamin Kiesl and Martina Seidl. QRAT polynomially simulates ∀ \text -exp+res. In Mikolás Janota and Inês Lynce, editors, *Theory and Applications of Satisfiability Testing - SAT 2019 - 22nd International Conference, SAT 2019, Lisbon, Portugal, July 9-12, 2019, Proceedings*, volume 11628 of *Lecture Notes in Computer Science*, pages 193–202. Springer, 2019.

22 Hans Kleine Büning, Marek Karpinski, and Andreas Flögel. Resolution for quantified Boolean formulas. *Information and Computation*, 117(1):12–18, 1995.

23 Tomás Peitl, Friedrich Slivovsky, and Stefan Szeider. Proof complexity of fragments of long-distance q-resolution. In Mikolás Janota and Inês Lynce, editors, *Theory and Applications of Satisfiability Testing - SAT 2019 - 22nd International Conference, SAT 2019, Lisbon, Portugal, July 9-12, 2019, Proceedings*, volume 11628 of *Lecture Notes in Computer Science*, pages 319–335. Springer, 2019.

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

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il