Electronic Colloquium on Computational Complexity, Revision 1 of Report No. 2 (2022)

Extending Merge Resolution to a Family of QBF-Proof Systems

Sravanthi Chede
Indian Institute of Technology Ropar, Rupnagar, India

Anil Shukla
Indian Institute of Technology Ropar, Rupnagar, India

Abstract

Merge Resolution (MRes [4]) is a recently introduced proof system for false QBFs. Unlike other known QBF proof systems, it builds winning strategies for the universal player (countermodels) within the proofs as merge maps. Merge maps are deterministic branching programs in which isomorphism checking is efficient, as a result MRes is a polynomial time verifiable proof system.

In this paper, we introduce a family of proof systems MRes-$\mathcal{R}$ in which the information of countermodels are stored in any pre-fixed complete representation $\mathcal{R}$. Hence, corresponding to each possible complete representation $\mathcal{R}$, we have a sound and refutationally complete QBF-proof system in MRes-$\mathcal{R}$. To handle these arbitrary representations, we introduce consistency checking rules in MRes-$\mathcal{R}$ instead of the isomorphism checking in MRes. As a result these proof systems are not polynomial time verifiable (Non-P). Consequently, the paper shows that using merge maps is too restrictive and with a slight change in rules, it can be replaced with arbitrary representations leading to several interesting proof systems.

We relate these new systems with the implicit proof system from the algorithm in [8], which was designed to solve DQBFs (Dependency QBFs) using MRes like clause-strategy pairs. We use the OBDD (Ordered Binary Decision Diagrams) representation suggested in the paper and deduce that 'Ordered' versions of the proof systems in MRes-$\mathcal{R}$ are indeed polynomial time verifiable.

On the lower bound side, we lift the lower bound result of regular MRes ([5]) by showing that the completion principle formulas (CR_n) from [16] which are shown to be hard for regular MRes in [5], are also hard for any regular proof system in MRes-$\mathcal{R}$. Thereby, the paper lifts the lower bound of regular MRes to an entire class of proof systems, which use some complete representations, including those undiscovered, instead of only merge maps. Thereby proving that the hardness of CR_n formulas is intact even after changing the weak isomorphism checking in MRes to the stronger consistency checking in MRes-$\mathcal{R}$.

2012 ACM Subject Classification Theory of computation → Proof complexity

Keywords and phrases Proof complexity, QBFs, Merge Resolution, Simulation, Lower Bound

Acknowledgements We would like to thank Gaurav Sood and Leroy Chew for important discussions, comments and suggestions regarding this paper.

ISSN 1433-8092

1 Introduction

Proof complexity is a sub-branch of computational complexity in which the main focus is to understand the complexity of proving (refuting) theorems (contradictions) in various proof systems. Informally, a proof system is a polynomial time computable function which maps proofs to theorems. Several propositional proof systems like resolution [21], Cutting planes [12], and Frege [15] have been developed for proving (refuting) propositional formulas. The relative strength of these proof systems has been well studied [22]. Several proof systems which are not polynomial time verifiable (unless $NP = \text{co-}NP$) have also been well studied. For example, semantic resolution [17] and semantic cutting planes [14].

Quantified Boolean formulas (QBFs) extend propositional logic by quantifying every variable by $\exists$ (there exists) and $\forall$ (for all). There are two major approaches for QBF-proof systems, namely, the CDCL (Conflict-Driven Clause Learning)-based and expansion-based systems. The basic systems in these approaches are Q-Res [18] and $\forall$Exp+Res [16] respectively. The Q-Res system was later extended to LD-Q-Res in [2] to allow a certain type of tautological clauses on universal variables in the proofs (which were always discarded in Q-Res) using merged literals. These merged literals have been shown to be interpreted as partial strategies rather than tautologies. These strategies were represented explicitly in [4] to form a new proof system called the MRes system. MRes (Merge Resolution) proof system [4] follows a different QBF solving approach. It builds partial strategies as 'merge maps' at each line of the proof such that the strategy at the last line forms the countermodel for the input QBF. Before applying the refutation rules, MRes needs the strategies of the hypothesis to be isomorphic. As isomorphism checking is known to be efficient in merge maps, MRes is a polynomial time verifiable proof system.

In this paper, we extend MRes to a family of sound and refutationally complete QBF proof systems MRes-$\mathcal{R}$. We observe that the representation of strategies in the proofs as merge maps is not relevant for the soundness and completeness of the proof system. Strategies can be depicted by any complete representation (Sec:2) and by slightly modifying the refutation rules to include arbitrary representations, the soundness and completeness of the proof system remains intact. To be precise, we change the isomorphism checking rule in MRes to 'consistency' checking rule (Sec: 3.1) defined initially for Dependency Quantified Boolean Formulas (DQBFs, [20]) in [8]. This leads to the definition of a new proof system for each complete representation. All these new proof systems together form the family of proof systems denoted by MRes-$\mathcal{R}$. Since the consistency checking rules are computationally hard, the proof systems in MRes-$\mathcal{R}$ are not polynomial time verifiable (**Non-P proof systems**). In literature, many interesting Non-P QBF proof systems have been studied, for instance, semantic cutting planes for QBFs (SemCP+$\forall$red) [7] and QBF proof systems modulo NP [10].

The paper also studies in detail the strength and limitations of these new proof systems. In [4], the authors demonstrated how MRes allows a few forbidden resolution steps of LD-Q-Res. Similarly, we show that because of the introduction of such powerful consistency checking rules, proof systems in MRes-$\mathcal{R}$ also allow a few forbidden resolution steps of MRes (ref. Example 8). We also show a Lower bound on a restricted version of proof systems in MRes-$\mathcal{R}$. We explain our contributions in detail in the following section:

1.1 Our Contributions

1. **Introducing a new family of non-polynomial time verifiable proof systems MRes-$\mathcal{R}$ for QBFs:** As already stated, proof systems in MRes-$\mathcal{R}$ use consistency checking instead of isomorphism rules of MRes. Informally, an 'isomorphism' check confirms whether two strategies are exactly the same or not. On the other hand, a

'consistency' check confirms whether or not two strategies can give a non-contradicting output for every possible assignment of input variables. Precisely, an output of '$*$' (trivial strategy) doesn't contradict with any output of the other strategy, while an output of '1' from one strategy and '0' from another is considered contradicting. MRes allows select operation (Sec:2.1) on isomorphic strategies. Whereas, proof systems in MRes-$\mathcal{R}$ allow union operation (Definition 2) on consistent strategies (i.e it retains both strategies and outputs the non-trivial assignment (if possible) from their outputs). For the further explanations of the MRes and MRes-$\mathcal{R}$ proof systems, refer Section 2.1 and 3 respectively.

For proving refutational completeness of the MRes-$\mathcal{R}$ family, we consider the MRes-$\mathcal{M}$ proof system $\in$ MRes-$\mathcal{R}$ which uses merge maps as the representation. We then prove that every valid rule of MRes is also valid in MRes-$\mathcal{M}$ (Theorem 6). We then show how to convert a MRes-$\mathcal{M}$ proof into a proof of any $\mathcal{P} \in$ MRes-$\mathcal{R}$ (Claim 7). This conversion is non-efficient, but still guarantees the completeness of systems in MRes-$\mathcal{R}$.

The soundness proof of any $\mathcal{P}$ ($\in$ MRes-$\mathcal{R}$) proof system follows from proving that every line of the $\mathcal{P}$-refutation gives a partial falsifying strategy for the universal player. Hence at the last line, we prove that it gives a countermodel for the input QBF (Lemma 4).

In [4], resolution steps where the strategy of the universal variables left of the pivot being same were shown to be forbidden in LD-Q-Res but allowed in MRes. We show in example 8, of resolution steps where the strategy of universal variables left of the pivot being consistent but not isomorphic to be forbidden in MRes but allowed in MRes-$\mathcal{R}$.

2. **Relating MRes-$\mathcal{R}$ with the implicit proof system from [8]**: In [8], the authors introduce an algorithm to solve DQBFs, which works with clause-strategy pairs like the MRes system. They also give a representation called the OBDDs (Ordered Binary Decision Diagrams [8, Definition 3]) which can support the consistency check in polynomial time. We observed that the implicit proof system from this algorithm is closely related to the MRes-$\mathcal{R}$ systems. More specifically, this algorithm outputs 'ordered'-MRes-$\mathcal{R}$ proofs using OBDDs as the representation (denoted by MRes-$\mathcal{O}$). We also show in Proposition 12 that Ordered MRes-$\mathcal{R}$ systems are polynomial time verifiable. For this, we first show how any MRes-$\mathcal{R}$ proof can be efficiently converted into a proof of MRes-$\mathcal{T}$ (the MRes-$\mathcal{R}$ system using a $\mathcal{T}$-representation defined in Section 4.1). Then, in Theorem 11, we give a method to convert efficiently an ordered MRes-$\mathcal{T}$ proof into an ordered MRes-$\mathcal{O}$ proof. Hence making the former system polynomial time verifiable. By transitivity from Theorem 9, it implies that all ordered systems in MRes-$\mathcal{R}$ are also polynomial time verifiable.
3. **Proving a lower bound for Regular MRes-$\mathcal{R}$:** A Lower bound for a proof system is a family of problems which are hard (exponential) to refute in that particular system. We establish one such lower bound for a restricted version of all proof systems in MRes-$\mathcal{R}$ (i.e regular MRes-$\mathcal{R}$, Sec:3.2) with a family of QBFs called Completion Principle Formulas (CR_n). The CR_n formulas (ref. Definition 13) were first introduced in [16], to show that level-ordered Q-Res cannot p-simulate the $\forall$Exp+Res proof system. It has been shown recently in [5], that CR_n formulas are even hard for regular MRes. In this paper, we lift this lower bound to all regular proof systems in MRes-$\mathcal{R}$ (Sec:5).

   To establish the lower bound, we mostly follow the ideas of the lower bound proof of regular MRes from [5, Theorem 9]. In [5], the major part of the proof relied on the fact that MRes uses isomorphism, so in a resolution step, they could rule out the variables not in one hypothesis merge map as also not to be present in the other. However, this is not the case in MRes-$\mathcal{R}$. So we provide a new Claim (ref. Claim 16) with proof that even though MRes-$\mathcal{R}$ insists on consistency rather than isomorphism, the clauses in CR_n make it such that the above property holds. Fig 4 shows the exact illustration of the proof idea.

This result implies that the lower bound result in [5] for regular-MRes is not because of the strictness of isomorphism checking in MRes, but it persists independent of the strategy representations.

We sum up our contributions in Fig 1. The figure also shows current p-simulation order among QBF proof systems with the contributions of this paper being in bold.

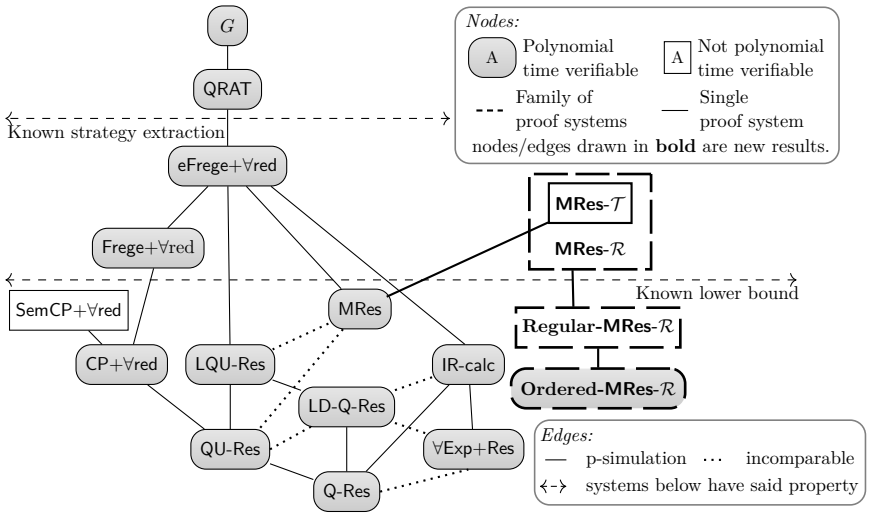


Figure 1 *Various QBF proof systems and p-simulations.* Regular MRes-$\mathcal{R}$ are below the 'known lower bound' dashed line by Theorem 18. MRes-$\mathcal{T}$ p-simulates MRes by Proposition 10. Ordered MRes-$\mathcal{R}$ systems are polynomial time verifiable due to Proposition 12. For other known simulations refer[11, Fig.1], in-comparability results refer [19, 5, 6]

2 Notations and Preliminaries

For a Boolean variable x, its literals can be x and $\overline{x}$. A clause C is a disjunction of literals and a conjunctive normal form (CNF) formula F is a conjunction of clauses. We denote the empty clause by $\bot$. $vars(C)$ is a set of all variables in C and $\text{width}(C) = |vars(C)|$.

Given a language $L \subseteq \{0,1\}^*$ and a string $x \in L$, a **proof system** f for L is an inference system, which is capable of showing that x is indeed in L. To do this, f derives a sequence of lines inferred via a set of predefined rules in a step by step fashion either from the hypothesis (i.e x) or from previously inferred lines. This sequence of lines are called an f-proof of the fact that $x \in L$. 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 $x \in L$. By definition, a proof system must be sound and complete for the language L. In addition, it must be polynomial time computable (verifiable). That is, given a sequence of lines, it must be check-able whether every line is derived by a valid rule of the system in time polynomial w.r.t the size of the input sequence, in which case, it is said to be a valid f-proof. A non-polynomial time verifiable proof system (**Non-P proof system**) for a language L is a proof system but without needing to be polynomial time verifiable.

Quantified Boolean formulas: QBFs are an extension of the propositional Boolean formulas where each variable is quantified with one of $\{\exists, \forall\}$, with their general semantic meaning of existential and universal quantifier respectively. In this paper, we assume that QBFs are in closed prenex form with CNF matrix i.e., we consider the form $Q_1X_1...Q_kX_k.\phi(X_1 \cup \cdots \cup X_k)$, where X_i are pairwise disjoint sets of variables; $Q_i \in \{\exists, \forall\}$ and $Q_i \neq Q_{i+1}$, and the matrix ϕ is in CNF form. We denote QBFs as $\mathcal{F} := Q.\phi$ in this paper, where Q is the quantifier prefix. If $x \in X_i$ then we denote $Q(x)$ to be equal to Q_i. For a variable x if $Q(x) = \exists$ (resp. $Q(x) = \forall$), we call x an existential (resp. universal) variable. If a variable x

is in the set X_i, any $y \in X_j$ where $j < i$ ($j > i$), we say that y occurs to the left (right) of x in the quantifier prefix and write $y \leq_Q x$ ($y \geq_Q x$). The set of existential variables to the left of a universal variable u will be denoted by $L_Q(u)$ in this paper.

Let $C \in \phi$ and $Q(u) = \forall$, then the 'falsifying u-literal' is defined to be 0 if $u \in C$, and 1 if $\overline{u} \in C$ and $*$ if $u \notin vars(C)$. Also, the existential subclause of C is the clause formed by only the existential literals from C. If S is any set of variables, a complete assignment of S will be an assignment which assigns all variables in S to either 1 or 0. Similarly, a partial assignment is an assignment which assigns a subset of variables in S to either 1 or 0 and the rest are denoted as having an assignment of '$*$'. We denote $\langle S \rangle$ and $\langle\langle S \rangle\rangle$ as the sets of all possible complete assignments and partial assignments of S respectively.

For a QBF $Q.\phi$, a **strategy** of universal player is a decision function that returns the assignment to all universal variables of Q, where the decision for each u depends only on the variables in $L_Q(u)$. If H^u is the strategy for the universal variable u then, $vars(H^u)$ is the subset of variables from $L_Q(u)$ which are actually used in building the strategy H^u. **Winning strategy** for the universal player is a strategy which for every possible assignment of existential variables, gives an assignment to all universal variables such that it falsifies the QBF. A QBF is false iff there exists a winning strategy for the universal player [1]. We say that a QBF proof system f admits **strategy extraction** if for any given valid f-proof of a false QBF $\mathcal{F}$, one can compute a winning strategy for the universal player in the time polynomial to the size of the f-proof. As said earlier, strategies are basically decision functions. For the portrayal of the same, many representations can be used like truth tables, directed acyclic graphs (DAGs), merge maps, etc. A **complete representation** is the one in which every possible finite decision function can be represented.

Resolution [21] is the most studied redundancy rule in both SAT and QBF worlds, we define the same as: $\frac{C \vee x \quad D \vee \overline{x}}{C \vee D}$, where C, D are clauses and x is the pivot variable. We denote this step as '$res(C \vee x, D \vee \overline{x}, x)$' throughout the paper.
Next, we define a few QBF proof systems that we require in this paper.

Q-Res [18] is one of a basic QBF proof system. It is an extension of the resolution proof system for QBFs. It allows the resolution rule defined in Section 2 with the pivot variable being existential. For dealing with the universal variables, it defines a 'universal reduction' rule which allows dropping of a universal variable u from a clause C, provided no existential variable $x \in C$ appears to the right of u.

LD-Q-Res [2] was developed as an improvement to the Q-Res system. The main 'Long distance rule' used in this system is: $\frac{(C_1 \cup U_1 \cup \{x\}) \quad (C_2 \cup U_2 \cup \{\overline{x}\})}{C_1 \cup C_2 \cup U}$, where x is the $\forall$ pivot variable, $U_1 \& U_2$ are literals of all the universal variables common in both the hypothesis such that if $u_1 (\in U_1)$ and $u_2 (\in U_2)$ are literals of the same variable z then either $u_1 = \overline{u_2}$ or $u_1/u_2 = z^*$. Here $U = \{u^* | u \in U_1\}$.

2.1 MRes

MRes is a proof system for false QBFs introduced in [4], we describe MRes briefly in this section. For a false QBF $Q.\phi$, an MRes refutation will be a sequence of lines of the form $L_i = (C_i, \{M_i^u\})$; where C_i is a clause of only existential-literals and $\{M_i^u\}$ is the set of merge maps of each universal variable $u \in Q$. The **merge map** M_i^u is a decision branching graph with definite strategies $\{0, 1, *\}$ at the leaves nodes ('$*$' is used when no strategy for u exists till that line) and the intermediate nodes branch on some existential variable (say x) $\in L_Q(u)$. That is, if $L_i = res(L_a, L_b, x)$ for some $a, b < i$, then M_i^u will get connected to M_a^u with an edge label of $\overline{x}$ and to M_b^u with an edge label of x.
An important property used in MRes rules is: **Isomorphism:** Two merge maps are isomorphic iff there exists a bijection mapping from the nodes of one to that of another.

Operations needed for MRes rules are defined as follows: **Select** operation on two isomorphic merge maps, outputs one of them. Or if one of them is trivial (i.e '$*$'), outputs the other. **Merge**(M^u_a, M^u_b, n, x) operation, defined when $a, b < n$, returns a new merge map where the root node is connected to M^u_a with $\overline{x}$ and M^u_b with x.

Now we define the MRes proof system:
For a false QBF $Q.\phi$, the MRes proof $\Pi := L_1, L_2, ..., L_m$ where every line L_i is derived using either an 'Axiom' step or a 'Resolution' step. In the axiom step, C_i will be the existential subclause of some $C \in \phi$ and every M^u_i will be a leaf node with the falsifying u-literal of C. In the resolution step, C_i is obtained from $res(C_a, C_b, x)$ where x is an $\exists$-variable and $a, b < i$, also each M^u_i must either be select(M^u_a, M^u_b) or if $x <_Q u$ then can be merge(M^u_a, M^u_b, i, x). Π is a refutation iff $C_m = \bot$.
G_Π is the derivation graph corresponding to Π with edges directed from the hypothesis to the resolvent (i.e from the axioms to the final line). For some given line L, Π_L is defined as the sub-derivation of Π deriving the line L.

3 MRes-$\mathcal{R}$: A new family of proof systems for false QBFs

Inspired from MRes, we define a family MRes-$\mathcal{R}$ where every proof system $\mathcal{P}$ ($\in$ MRes-$\mathcal{R}$) has its own complete representation to represent the strategies. We use the idea of consistency checking used in MRes-$\mathcal{R}$ from [8], which uses the same for DQBFs. For simplicity, we use the same notations from [8] whenever possible. We begin by defining some important notations and operations needed before formally defining the MRes-$\mathcal{R}$ systems.

3.1 Important notations used in MRes-$\mathcal{R}$

To begin, let us define what consistency means for any two assignments of a set of variables. Then we will extend it for two strategies.

▶ **Definition 1** ([8]). *Let X be any set of variables and $\varepsilon, \delta \in \langle\langle X \rangle\rangle$. We say that ε and δ are consistent, denoted by $\varepsilon \simeq \delta$, if for every $x \in X$ for which $\varepsilon(x), \delta(x) \neq *$ we have $\varepsilon(x) = \delta(x)$.*
Let H_u and H'_u be individual strategy functions for the universal variable u, we say that H_u and H'_u are **consistent** (written $H_u \simeq H'_u$) when $H_u(\varepsilon) \simeq H'_u(\varepsilon)$ for each $\varepsilon \in \langle\langle L_Q(u) \rangle\rangle$. In other words H_u and H'_u are consistent, if the u-assignments given by $H_u(\varepsilon)$ and $H'_u(\varepsilon)$ are consistent for every possible $L_Q(u)$-assignment ε.

By a change in notation, we can see (partial) assignments as both functions and sets of literals, i.e. an assignment ε corresponds to the set of literals it satisfies. For example, $\{x_1, x_2, \overline{x_3}, \overline{x_4}\}$ represents an assignment which sets 1 to the variables x_1 and x_2 and 0 to x_3 and x_4. In this notation as sets of literals, a union ($\cup$) of assignments ε, δ is defined when $\varepsilon \simeq \delta$ and it is equal to $\varepsilon \cup \delta$.
We now define a **union operation** ('$\circ$') on two consistent strategies H_u and H'_u.
▶ **Definition 2** ([8]). *Given two consistent strategies H_u and H'_u (i.e., $H_u \simeq H'_u$), we define the union strategy H''_u of H_u and H'_u, denoted by $H''_u = H_u \circ H'_u$, as:*

$$H''_u(\varepsilon) = H_u(\varepsilon) \cup H'_u(\varepsilon) \text{ for each } \varepsilon \in \langle L_Q(u) \rangle.$$

For example, if H_u and H'_u are defined as below, then $H''_u = H_u \circ H'_u$ will be:

$$H_u = \begin{cases} 1 & : & x \\ * & : & \overline{x} \end{cases} \qquad H'_u = \begin{cases} * & : & x \\ 0 & : & \overline{x} \end{cases} ; \qquad H''_u = \begin{cases} 1 \cup * = 1 & : & x \\ * \cup 0 = 0 & : & \overline{x} \end{cases}$$

We now define a **if-else operation** ('$\bowtie$') on any two strategies H_u and H'_u.
▶ **Definition 3** ([8]). *Given any two strategies H_u and H'_u and an existential variable x, we define the if-else operation of H_u and H'_u on x to give the strategy H''_u, denoted by $H''_u = H_u \overset{x}{\bowtie} H'_u$, for every $\varepsilon \in \langle L_Q(u) \rangle$ as follows:*

$$H''_u(\varepsilon) = \begin{cases} H_u(\varepsilon) & : & \varepsilon(x) = 1 \\ H'_u(\varepsilon) & : & \varepsilon(x) = 0 \end{cases}$$

For example, if H_u and H'_u be defined as below, then $H''_u = H_u \overset{x}{\bowtie} H'_u$ will be:

$$H_u = \begin{cases} 1 & : & y \\ * & : & \overline{y} \end{cases} \qquad H'_u = 0 \qquad ; \qquad H''_u = \begin{cases} 1 & : & xy \\ * & : & x\overline{y} \\ 0 & : & \overline{x} \end{cases}$$

3.2 Definition of MRes-$\mathcal{R}$

Let $\Phi = Q.\phi$ be a QBF with existential variables X and universal variables U. A MRes-$\mathcal{R}$ derivation of L_m from Φ is sequence $\pi = L_1, ..., L_m$ of lines where each $L_i = (C_i, \{H_i^u : u \in U\})$ in which at least one of the following holds for $i \in [m]$:

a. **Axiom**. There exists a clause $C \in \phi$ such that C_i is the existential subclause of C, and for each $u \in U$, H_i^u is the strategy function mapping u to the falsifying u-literal for C or,
b. **Resolution**. There exist integers $a, b < i$ and an existential pivot $x \in X$ such that $C_i = res(C_a, C_b, x)$ and for each $u \in U$:
   i. if $x <_Q u$, then $H_i^u = H_b^u \overset{x}{\bowtie} H_a^u$
   ii. else if $x >_Q u$, then $H_i^u = H_a^u \circ H_b^u$.

π is a refutation of Φ iff $C_m = \bot$. Size of π is the number of lines i.e $|\pi| = m$.

Regular MRes-$\mathcal{R}$: Let S be a subset of existential variables X of a QBF $\mathcal{F}$. We say that a $\mathcal{P}$-refutation π of $\mathcal{F}$ ($\mathcal{P} \in$MRes-$\mathcal{R}$) is S-regular if for every $x \in S$, there is no leaf to root path in G_π that uses x as pivot more than once. A X-regular proof is simply a regular proof.

Ordered MRes-$\mathcal{R}$: Let X be the set of all existential variables of a false QBF $\mathcal{F}$ and $\leq_X$ be a fixed ordering of variables in X. We say that a $\mathcal{P}$-refutation π of $\mathcal{F}$ (where $\mathcal{P} \in$MRes-$\mathcal{R}$) is ordered if it is regular and for each leaf-to-root path in G_π, the pivots follow $\leq_X$.

3.3 Soundness of MRes-$\mathcal{R}$:

Soundness of MRes-$\mathcal{R}$ is proved by the next lemma, it follows closely to that of MRes in [4].

▶ **Lemma 4.** *Let $\pi = L_1, \ldots, L_m$ be a $\mathcal{P}$ refutation ($\mathcal{P} \in$ MRes-$\mathcal{R}$) of QBF Φ. Then, strategy functions $\{H_m^u : u \in U\}$ in the conclusion line L_m will form a countermodel for Φ.*

Proof. Given $\pi := L_1, \ldots, L_m$ be a $\mathcal{P}$-refutation of QBF $\Phi = Q.\phi$. Each L_i=$(C_i, \{H_i^u : u \in U\})$ and X, U are sets of all existential and universal variables in Q respectively. For $i \in [m]$,

- let $\alpha_i := \{\overline{l} : l \in C_i\}$ be the smallest assignment falsifying Ci ,
- let $A_i := \{\alpha \in \langle X \rangle : C_i \cap \alpha = \emptyset\}$ be all complete assignments to X consistent with α_i,
- for each $\alpha \in A_i$, let $l_i^u(\alpha) := H_i^u(\alpha)$ and $H_i(\alpha) := \{l_i^u(\alpha) : u \in U\} \setminus \{*\}$.

Induction statement:By induction on $i \in [m]$, we show, for each $\alpha \in A_i$, that the restriction of ϕ by $\alpha \cup H_i(\alpha)$ contains the empty clause.

Proof: For the base case $i = 1$, let $\alpha \in A_1$. As L_1 is introduced as an axiom, there exists a clause $C \in \phi$ such that C_1 is the existential subclause of C, and each H_1^u is the function mapping u to the falsifying u-literal for C. Hence, for each $u \in U$, $l_1^u(\alpha)$ is the falsifying u-literal for C, so $C[\alpha \cup H_1(\alpha)] = \emptyset$.

For the inductive step, let $i \geq 2$ and let $\alpha \in A_i$. The case where L_i is introduced as an axiom is identical to the base case, so we assume that L_i was derived by resolution. Then there exist integers $a, b < i$ and an existential pivot $x \in X$ such that $C_i = res(C_a, C_b, x)$. Suppose that $\overline{x} \in \alpha$ (a similar argument holds when $x \in \alpha$), each $u \in U$ has to satisfy either:

(i) $x <_Q u$ and $H_i^u = H_b^u \overset{x}{\bowtie} H_a^u$: In which case, $l_i^u(\alpha) = l_a^u(\alpha)$.
(ii) $x >_Q u$ and $H_i^u = H_a^u \circ H_b^u$: In which case, $l_i^u(\alpha) = \{l_a^u(\alpha) \cup l_b^u(\alpha)\}$.

It follows that $l_i^u \neq l_a^u$ only if $l_a^u = *$, and hence $H_a(\alpha) \subseteq H_i(\alpha)$. Since $C_a \setminus \{x\} \subseteq C_i$, we have $\alpha \in A_a$, so the restriction of ϕ by $\alpha \cup H_i(\alpha)$ contains the empty clause by the inductive hypothesis that $\alpha \cup H_a(\alpha)$ contains the empty clause. ◀

Since α_m is the empty assignment, we have $A_m = \langle X \rangle$. We therefore prove the lemma at the final step $i = m$, as we show that $\{H_m^u : u \in U\}$ is a countermodel for Φ. ◀

3.4 Completeness of MRes-$\mathcal{R}$:

Completeness of MRes-$\mathcal{R}$ is proved by the following Theorem 6 and Claim 7. We will need the following remark from the paper introducing MRes [4]. For proof of Claim 7, refer Appx: A.

▶ Remark 5. [4, Proposition 10] Any two isomorphic merge maps compute the same function.

▶ **Theorem 6.** *MRes-$\mathcal{M}$ (MRes-$\mathcal{R}$ using merge maps as representation) p-simulates MRes.*

Proof. Given a QBF Φ and its MRes-proof $\pi = L_1, ..., L_m$, where every $L_i = \{C_i, \{M_i^u : u \in U\}\}$. We build an MRes-$\mathcal{M}$ proof $\Pi = L'_1, ..., L'_m$ for Φ, where each $L'_i = \{C'_i, \{H_i^u : u \in U\}\}$. For every line L_i in π starting from $i = 1$ to m, if L_i is an axiom step then directly $C'_i = C_i$ and $H_i^u = M_i^u$ for all $u \in U$. Otherwise, if L_i is an resolution step i.e for some $a, b < i$, $C_i = res(C_a, C_b, x)$; then set $C'_i = C_i$ and for each $u \in U$ if $x <_Q u$ then set $H_i^u = H_b^u \overset{x}{\bowtie} H_a^u$ else set $H_i^u = H_a^u \circ H_b^u$. These are sound steps as resolution in MRes can be either:

(i) $x >_Q u$ and $M_i = \text{select}(M_a^u, M_b^u)$; in this case we set $H_i^u = H_a^u \circ H_b^u$ which holds given the Remark 5 and that isomorphism $\Rightarrow$ consistency.

(ii) $x <_Q u$ and $M_i = \text{merge}(M_a^u, M_b^u, i, x)$; in this case we set $H_i^u = H_b^u \overset{x}{\bowtie} H_a^u$ which is same as the merge function of MRes.

(iii) $x <_Q u$ and $M_i = \text{select}(M_a^u, M_b^u)$; in this case we set $H_i^u = H_b^u \overset{x}{\bowtie} H_a^u$ which is allowed as MRes did the isomorphism test on M_a^u and M_b^u, but we do not need it for $\bowtie$ in MRes-$\mathcal{R}$.

In case-(iii) above it remains to note that adding a $\bowtie$ to two isomorphic maps or when one of them is $*$, doesn't add any new strategy: it just dilutes the strategy represented by the corresponding merge map. That is, we are adding an if-else condition where both the outcomes are same or one of them is $*$. Hence doesn't affect future consistency checks which may arise in the proof. (For further clarity, one is suggested to look at Appx:Example 19 but is not needed for the proof). Finally, the constructed Π is a valid MRes-$\mathcal{M}$ proof of Φ. ◀

▶ **Claim 7.** *Every MRes-$\mathcal{M}$-proof can be transformed into an MRes-$\mathcal{R}$-proof for any representation R in exponential time.*

These guarantee the completeness of proof systems in MRes-$\mathcal{R}$ as MRes is complete and any MRes-proof can be transformed into a MRes-$\mathcal{M}$-proof (by Theorem 6) which in-turn can be transformed as any MRes-$\mathcal{R}$-proof (by Claim 7).

Next, we present an example (Example 8) of MRes-$\mathcal{R}$ allowing few resolution steps which are forbidden in MRes. Such examples may be useful for the separation results between the proof systems in MRes-$\mathcal{R}$ and the existing MRes proof system.

Table 1 $\mathcal{P}$-refutation, where $\mathcal{P} \in$ MRes-$\mathcal{R}$, of the false QBF in Example 8

Line	Rule	C_i	H_i^u
L_1	axiom	$\{y, x\}$	0
L_2	axiom	$\{y, \overline{x}\}$	$*$
L_3	$res(L_1, L_2, x)$	$\{y\}$	$H_2^u \overset{x}{\bowtie} H_1^u$
L_4	axiom	$\{\overline{y}, x\}$	$*$
L_5	axiom	$\{\overline{y}, \overline{x}\}$	1
L_6	$res(L_4, L_5, x)$	$\{\overline{y}\}$	$H_5^u \overset{x}{\bowtie} H_4^u$
L_7	$res(L_3, L_6, y)$	$\{\}$	$H_3^u \circ H_6^u$

▶ **Example 8.** Consider any proof system $\mathcal{P}$ in MRes-$\mathcal{R}$ which uses some complete R representation for strategies. The following Table 1 is a $\mathcal{P}$-refutation of the false QBF : $\exists x \forall u \exists y\ (y \vee x \vee u) \wedge (y \vee \overline{x}) \wedge (\overline{y} \vee x) \wedge (\overline{y} \vee \overline{x} \vee \overline{u})$

The strategies H_3^u and H_6^u in function format are as follows:

$$H_3^u = \begin{cases} 0 & : \quad x = 0 \\ * & : \quad x = 1 \end{cases} \qquad H_6^u = \begin{cases} * & : \quad x = 0 \\ 1 & : \quad x = 1 \end{cases}$$

One can see that these strategies are consistent (but not isomorphic), hence the resolution of L_3, L_6 on y is allowed in the $\mathcal{P}$-refutation. But the corresponding resolution would be blocked in MRes since the corresponding merge maps M_3^u, M_6^u will not be isomorphic.

4 MRes-$\mathcal{T}$ proof system

In this section, we will define a particular proof system MRes-$\mathcal{T}$ from the family of MRes-$\mathcal{R}$ proof systems. The importance of this system is that any $\mathcal{P}$-refutation ($\mathcal{P} \in$ MRes-$\mathcal{R}$) can be efficiently converted into an MRes-$\mathcal{T}$-refutation. That is, all proof systems in MRes-$\mathcal{R}$ can be p-simulated by this MRes-$\mathcal{T}$ system. Later in this section, we will discuss how this system relates to the implicit proof system of the algorithm defined in [8].

4.1 Definition of MRes-$\mathcal{T}$

For a false QBF $\mathcal{F}$, the sequence of lines $\pi := (C_1, T_1), \ldots, (C_m, T_m)$ is an MRes-$\mathcal{T}$ refutation if $C_m = \bot$ and each T_i is built based on the derivation of C_i from parents C_j, C_k as follows:

$$T_i := \begin{cases} \text{Axiom node as in MRes} & \text{Axiom step of MRes-}\mathcal{R} \\ \text{Merge node over } T_j, T_k \text{ as in MRes} & \text{If-else step (`}C_j \overset{x}{\bowtie} C_k\text{') of MRes-}\mathcal{R} \\ \#\text{ node (defined below) on } T_j, T_k & \text{Union step (`}C_j \circ C_k\text{') of MRes-}\mathcal{R} \end{cases}$$

The **# node** is defined assuming both its inputs are consistent, and it outputs the result of a union operation on them; Precisely, it's truth table is shown in Fig. 2.

A	B	A # B
1	1	1
0	0	0
*	0/1	0/1
0/1	*	0/1
*	*	*

Figure 2 Truth table for # operator (It assumes inputs to be consistent).

Note that $A = 1, B = 0$ and vice-versa cannot happen in a valid MRes-$\mathcal{R}$ proof owing to the definition of union('$\circ$'). Therefore, the corresponding rows are omitted from the truth table in Fig 2. For an illustrative example of an MRes-$\mathcal{T}$-proof, see Appx:Example 20.

Observe that the proposed T representation is complete. That is, any valid finite function can be represented by a T graph. This follows since, merge maps are a subset of T-graphs (i.e without # nodes) which are just branching programs, but known to be complete for all valid functions. Since T representations are complete, it implies MRes-$\mathcal{T} \in$ MRes-$\mathcal{R}$.

4.2 Conversion of MRes-$\mathcal{R}$ proofs into MRes-$\mathcal{T}$ proofs

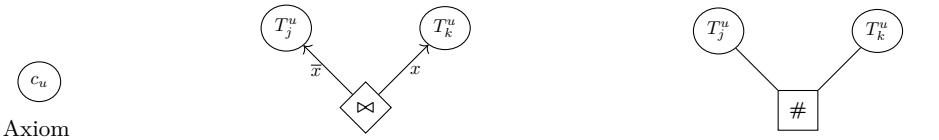
In this section, we show how to convert a $\mathcal{P}$-proof π ($\mathcal{P} \in$ MRes-$\mathcal{R}$) into a MRes-$\mathcal{T}$ proof π'. Let $\pi = (C_1, R_1), \ldots, (C_m, R_m)$ be a $\mathcal{P}$ proof of a QBF $\mathcal{F}$. We show how to convert π into a MRes-$\mathcal{T}$ proof $\pi' = (C_1, T_1), \ldots, (C_m, T_m)$ of the same QBF $\mathcal{F}$. Note that here T_i is not the representation of R_i, but T_i is capturing how R_i has been constructed from some hypothesis R_j, R_k with $j, k < i$ using rules from Section 3.2. For this we do not need to interpret R_i's, but we can extract the required information from the clauses C_j, C_k and C_i of π (illustration of the same is given in Fig 3).

▶ **Theorem 9.** *Any $\mathcal{P}$-proof ($\mathcal{P} \in$ MRes-$\mathcal{R}$) can be converted efficiently into an MRes-$\mathcal{T}$ proof.*

Proof. For a false QBF $\mathcal{F}$, proofs of proof systems belonging to MRes-$\mathcal{R}$ can have arbitrary representations for the strategies computed. However, the rules allowed to construct a strategy R_i using any strategies R_j and R_k (where $j, k < i$) are fixed. They must follow the rules mentioned in Section 3.2. MRes-$\mathcal{T}$ proof π' captures these rules only.

To be precise, given a $\mathcal{P}$-proof π of $\mathcal{F}$ where $\pi = (C_1, P_1), (C_2, P_2), ..., (C_m, P_m)$, we construct MRes-$\mathcal{T}$-proof π' as follows: From the clause part of the proof π i.e $C_1, ..., C_m$ (in

this sequence) based on what step is being followed (axiom, or resolution where pivot is on left, or resolution where pivot is on right), we build the corresponding T-graphs as shown in the Figure 3. After following this procedure for all lines in π, the sequence of lines so formed i.e $\pi' = (C_1, T_1), (C_2, T_2), \ldots, (C_m, T_m)$ is a MRes-$\mathcal{T}$ proof as the clauses $C_1, ..., C_m$ are the same as in the original MRes-$\mathcal{R}$ proof hence we know that C_m is definitely $\bot$ and that $T_1, ..., T_m$ are built using the same rules as used when building the $\mathcal{P}$-proof π. ◀



(a) Axiom

(b) if-else node (res when x is left of u in prefix)

(c) # node (i.e union node) (res when x is right of u in prefix)

Figure 3 Rules to construct T-graphs. In Figure 3a, c_u is the falsifying strategy of u for the axiom clause C_i. In Figure 3b, $C_i = res(C_j, C_k, x)$ and x is left of u in prefix i.e $T_i^u = T_k^u \overset{x}{\bowtie} T_j^u$. In Figure 3c, $C_i = res(C_j, C_k, x)$ and x is right of u in prefix i.e $T_i^u = T_j^u \circ T_k^u$.

Observe that due to Theorem 9, MRes-$\mathcal{T}$ p-simulates any MRes-$\mathcal{R}$ proof system, and therefore, it also p-simulates the MRes-$\mathcal{M} \in$ MRes-$\mathcal{R}$ proof system, which is known to simulate the MRes proof system (Theorem 6). Thus we have the following:

▶ **Proposition 10.** *MRes-$\mathcal{T}$ p-simulates MRes.*

4.3 MRes-$\mathcal{T}$ versus Implicit proof system in [8]

The authors in [8] give an algorithm to work with DQBFs and a representation for strategies (OBDDs) to make the consistency check in the algorithm efficient. In this section we discuss how the implicit proof system from this algorithm relates to our newly defined proof systems.

The algorithm in [8] is designed to eliminate pivots in any fixed order by taking all possible resolvents at every stage. Hence, one can clearly see that the algorithm works implicitly on the ordered-MRes-$\mathcal{R}$ system which uses OBDDs for the representation (denoted as MRes-$\mathcal{O}$). As the consistency check and union operation on OBDDs are shown to be efficient ([8, 13]), it makes the corresponding ordered-MRes-$\mathcal{O}$ systems polynomial time verifiable.

We note that the 'ordered'-$\mathcal{T}$-representation is much similar to an OBDD with an extra # node. We can efficiently convert an 'ordered'-$\mathcal{T}$ strategy into an OBDD representation as follows: At every # node, recursively from leaf-to-root, perform the union operation on the hypothesis strategies assuming them to be OBDDs and then replace the # node with the resultant OBDD-representation. This will end finally with a complete OBDD representation of the initial strategy represented by the ordered-$\mathcal{T}$-representation. When clubbed with ordered-MRes-$\mathcal{O}$ being polynomial time verifiable, it implies the following theorem. For an example of this conversion, see Appx D.

▶ **Proposition 11.** *Ordered MRes-$\mathcal{T}$ is polynomial time verifiable proof system.*

Proposition 11 along with the algorithm in Theorem 9 provided for conversion of any MRes-$\mathcal{R}$ proof into MRes-$\mathcal{T}$, deduces the following:

▶ **Proposition 12.** *Ordered MRes-$\mathcal{R}$ is a family of polynomial time verifiable proof systems.*

Observe that regular MRes-$\mathcal{T}$ cannot be guaranteed to be polynomial time verifiable. This is because, a regular MRes-$\mathcal{T}$ would need an FBDD (Free BDD) to use as a representation for strategies and according to [13], FBDDs cannot guarantee polynomial time verifiability. Therefore, polynomial time verification, even with the usage of OBDDs, stops at ordered-proofs itself, which are a restriction of regular proofs which in-turn are a restricted version of general proofs. Refer Fig 1 for the complete picture of all the above discussed proof systems.

5 Lower Bound for Regular MRes-$\mathcal{R}$

▶ **Definition 13** (Completion Principle Formulas (CR_n) [16]).

$$CR_n = \underset{i,j\in[n]}{\exists} x_{ij}, \forall z, \underset{i\in[n]}{\exists} a_i, \underset{j\in[n]}{\exists} b_j \left(\bigwedge_{i,j\in[n]} (A_{ij} \wedge B_{ij}) \right) \wedge L_A \wedge L_B$$

where, $A_{ij} = x_{ij} \vee z \vee a_i$ $\quad B_{ij} = \overline{x_{ij}} \vee \overline{z} \vee b_j$

$L_A = \overline{a_1} \vee \cdots \vee \overline{a_n}$ $\quad L_B = \overline{b_1} \vee \cdots \vee \overline{b_n}$

We define the sets $A := \{a_1, a_2, ..., a_n\}$ *and* $B := \{b_1, b_2, ..., b_n\}$ *for ease of usage.*

In this section, we prove that the CR_n formulas are hard to refute in regular proof systems of MRes-$\mathcal{R}$. The lower bound result follows from a stronger result that we prove below in Theorem 14. We use the ideas from [5] and try to maintain the same notations wherever possible for simplicity. The proof setup is depicted in Figure 4. The basic idea of the proof is: As every clause in CR_n has a variable from the set $A \cup B$, but the refutation should derive a $\bot$ at the final line; there must be a 'section' of the proof (See shaded region S' in Fig 4) which only has X variables in all its clauses. This section also includes the final line. The set of clauses at the 'border' (See the bold line S in Fig 4) of this section of the proof is shown to be wide (in terms of number of literals) in Lemma 15. Using this and the argument that the conjunction of clauses in S itself forms a false CNF formula, we show in Theorem 14 that the number of clauses in S is exponential in n. This directly implies the lower bound.

To establish the width bound, we note that the pivots which are used while deriving clauses in S are variables from $A \cup B$ and that they are all to the right of z. Meaning that the corresponding resolutions must all be union steps i.e the incoming strategies must be consistent (not isomorphic as is the case in MRes). This especially makes it difficult to directly lift the lower bound proof of MRes from [5]. However we overcome this issue in Claim 16 by arguing how L_A, L_B are the only clauses with trivial strategies and how any other clause which resolves with these will mask this trivial-ness with its own definitive strategy. Further, by analysing what axiom clauses cannot be used in the derivation of the clauses in S, we show that many variables cannot be resolved before these lines. Hence, these variables will still be present in the clause $\in S$, making it wide. We now clearly state the Theorem, Lemma and Claims explained above and give the proof for those with vital changes from the proofs in [5], for remaining detailed proofs see Appx:C.

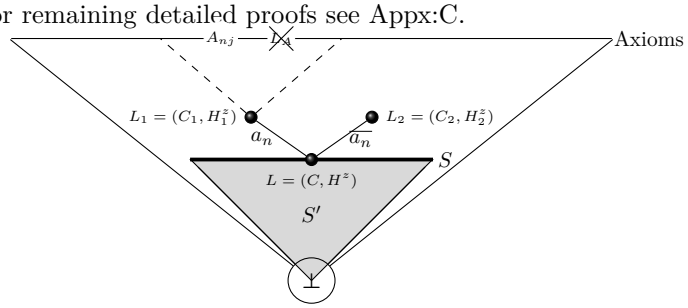


Figure 4 *Lower bound proof illustration.* Given a CR_n formula and it's $\mathcal{P}$-proof Π ($\mathcal{P} \in$ MRes-$\mathcal{R}$), this figure shows the graph G_Π. Claim 16 proves $x_{ij} \notin var(H_2^z)$ for $i \in [n-1], j \in [n]$. Claim 17 shows $|vars(C_2)| \geq n-1$. Lemma 15 shows $|vars(C)| \geq n-1$. Theorem 14 proves that $|S| \geq 2^{n-1}$.

▶ **Theorem 14.** *Every $(A \cup B)$-regular refutation of CR_n in any proof system belonging to MRes-$\mathcal{R}$ has size $2^{\Omega(n)}$.*

For $\mathcal{P} \in$ MRes-$\mathcal{R}$, let Π be a $\mathcal{P}$-refutation of CR_n (for $n > 2$). Let the set of lines S, S' be defined as follows:

S': This set consists of all the lines $L = (C, H^z)$ from Π such that $vars(C) \cap \{A \cup B\} = \emptyset$ and there exists a path from L to $\bot$ in G_Π consisting of lines only from S'.

S: This set contains all the lines $L \in S'$ such that $L = Res(L_1, L_2, v)$ where $L_1, L_2 \notin S'$. Observe that the pivot variable v must belong to $\{A \cup B\}$.

▶ **Lemma 15** ([5]). *For all $L = (C, H^z) \in S$, width(C) $\geq n-1$.*

Proof. Observe that L is not an axiom as all axioms of CR_n have a variable from $A \cup B$ and so they cannot belong to S. So, let $L = res(L_1, L_2, v)$ where $L_1, L_2 \notin S'$. Since two lines not belonging in S' resolve to make the resultant $\in S'$, the pivot (i.e v) should be from $A \cup B$. Assume $v \in A$, a similar argument can be made when $v \in B$. Without loss of generality, assume that $v = a_n$; and $a_n \in C_1$ and $\overline{a_n} \in C_2$ (Note: a_n is used only for ease in dividing the set A into partitions. Nowhere in the proof we use the fact that a_n is the last variable in A.)

Since Π is $(A \cup B)$-regular, a_n does not occur as a pivot in the sub-derivation Π_{L_1}. It implies that the axiom clause L_A cannot be used in deriving L_1, because otherwise C_1 will have both a_n & $\overline{a_n}$ making it a tautology. That implies, axioms with other positive literals a_i's cannot be used in Π_{L_1} as the negated literals $\overline{a_i}$'s are only available in L_A which in-turn cannot be used in Π_{L_1}. Positive literals of a_i's only $\in A_{ij}$ for all $j \in [n]$. Hence, axioms A_{ij} for $i \in [n-1], j \in [n]$ also cannot be used in deriving the line L_1. We know x_{ij} only occur in A_{ij}; so H_1^z has no x_{ij} variable for $i \in [n-1], j \in [n]$. Also, H_1^z is not a trivial strategy as some A_{nj} for $j \in [n]$ has been used because $a_n \in C_1$. Fix this j for the rest of the proof.

Since the pivot a_n at the resolution step obtaining line L is to the right of z, by the rules of MRes-$\mathcal{R}$, H_1^z and H_2^z are consistent. In Claim 16, we prove that even though MRes-$\mathcal{R}$ only insists on consistency, it still holds that for each $i \in [n-1]$, and each $j \in [n]$, $x_{ij} \notin \text{var}(H_2^z)$. Using this result we prove in Claim 17 below, that C_2 will have at least $n-1$ variables (including $\overline{a_n}$). Therefore, at least $n-2$ variables from C_2 belong in C.

At this point, if we can show that at least one extra variable is present in C_1 but not in C_2, then $|C| \geq n-1$ and the proof is complete. To prove this, there will be two cases, in the first case, we easily prove that $x_{nj} \in C_1$ but not in C_2. In the second case (which is lengthy), where x_{nj} is not in C_1, we prove that one extra x variable must belong to $C_1 \setminus C_2$.

Case-1 Since the clause A_{nj} (as fixed above) was used in Π_{L_1}, the literal x_{nj} is introduced into the proof and resolution of x_{nj} is not possible before L_1. This is because, the clause B_{nj} needed to resolve it, brings with it literal b_j which needs to be resolved before L_1 (as L_1 cannot have any $A \cup B$ literals other than a_n). To resolve this b_j, one needs to introduce the clause L_B, but L_B brings all $\overline{b}$'s into the resultant which cannot be further resolved as the B-clauses needed for the same do not have consistent strategies anymore. That is, because of the use of A_{nj} the resolvent has a 0 strategy for some assignment to X variables, but B-clauses have a constant strategy of 1 hence these strategies will not be consistent to resolve further. This x_{nj} cannot $\in C_2$ as the corresponding axiom clause needed for the same has a_n in it, which would make C_2 a tautology.

Case-2 One can forcefully take a longer way to get rid of x_{nj} from the clause C_1. That will be by resolving L_B with B_{*p}'s for $p \in [n]-j$ and with B_{nj}. Then, resolve with A_{nj} to get rid of x_{nj}. One should note that in this process instead of 1 we now have $n-1$ variables other than a_n in C_1. The final C_1 in this process will be $\{a_n,\ \overline{x_{*p}}$'s for $p \in [n]-j\}$.
Now for C_2 to already have all these literals, one should start from L_B and use the same B_{*p}'s for $p \in [n]-j$ and a B_{*j} as well to clear out all b's from the clause. The resultant clause will be $\{\overline{x_{*p}}$'s for $p \in [n]\}$. Now one needs to introduce the $\overline{a_n}$ literal from L_A but no pivot variable is available for direct resolution. Even if one cleverly uses some A_{kj} for $k \in [n-1]$ to resolve and remove x_{kj} which was anyhow not in C_1; later resolving with L_A blocks any further resolutions to remove remaining a's as the strategies will no longer be consistent. So, one will have to resolve first with all A_{*p}'s, which will remove all x_{*p}'s as well. But further when resolving with L_A, both literals of all a's will be present, which is a tautology. Hence atleast one extra literal will belong to C_1 which will be passed to C.

Using the three results above, we can derive that width$(C) \geq n-1$. ◀

▶ **Claim 16.** *For $i \in [n-1]$, and each $j \in [n]$, $x_{ij} \notin var(H_2^z)$.*

Proof. At the point of use of this claim in the proof of Lemma 15, we definitely know that for $i \in [n-1]$ & $j \in [n]$; $x_{ij} \notin H_1^z$. That is, if f_1 is the function representing the strategy H_1^z, then for any assignment σ of x_{nj}'s and $i \in [n-1], j \in [n]$, it implies that:

$$f_1(\sigma, x_{ij} = 0) = f_1(\sigma, x_{ij} = 1) \tag{1}$$

Let f_2 be the function representing the strategy H_2^z. Since a_n is to the right of z, we know that H_1^z and H_2^z are consistent, i.e for any assignment σ' (an extension of σ) and for $i \in [n-1], j \in [n]$, it implies that:

$$f_2(\sigma', x_{ij} = 0) \simeq f_1(\sigma', x_{ij} = 0) \tag{2}$$

$$f_2(\sigma', x_{ij} = 1) \simeq f_1(\sigma', x_{ij} = 1) \tag{3}$$

Only remaining question is if $f_2(\sigma', x_{ij} = 0) = f_2(\sigma', x_{ij} = 1)$? Observe that if this equality holds, then f_2 will be independent of x_{ij}'s, which implies that $x_{ij} \notin H_2^z$ for $i \in [n-1], j \in [n]$. Now, we are heading towards proving the equality holds. Note that if none of the terms in equation 2 and equation 3 give a '$*$' for any assignment of X, the equality in question definitely holds. So, now we prove that none of them can give a '$*$' for any given assignment.

The only axiom clauses of CR_n with trivial strategies are L_A, L_B and these axioms only contain variables of $A \cup B$, which are all to the right of z. Hence if any other clause is to be resolved with these clauses, the pivot has to be in $A \cup B$ i.e. a union step needs to be performed. At this point the trivial-ness of L_A (or L_B) is masked and does not show up in the final strategy of the resultant line; this is because union of any strategy with a trivial strategy will be the strategy itself. The only case by which a '$*$' can be in the resulting strategy is if L_A is resolved with L_B, which can clearly not happen as they have no common variable. Since C_1, C_2 are definitely not the axiom clauses L_A (or L_B), using the above argument it is simply not possible for the functions f_1 (or f_2) to output a '$*$' for any input assignment provided. This means the equality in question above holds; meaning that H_2^z also doesn't depend on x_{ij}'s when $i \in [n-1], j \in [n]$ i.e $x_{ij} \notin vars(H_2^z)$. ◀

▶ **Claim 17** ([5]). *Either for all $i \in [n-1]$, C_2 has a variable of the form x_{i*}, or for all $j \in [n]$, C_2 has a variable of the form x_{*j}*

From the above discussions and due to Theorem 14, we have the following:

▶ **Theorem 18.** *Every MRes-$\mathcal{R}$-regular refutation of CR_n has size $2^{\Omega(n)}$.*

6 Conclusion and Future work

This paper extends MRes proof system into a family of non-P proof systems MRes-$\mathcal{R}$ and provides a motivation example of forbidden steps of MRes being allowed in MRes-$\mathcal{R}$. This paper also deduces that 'ordered' versions of proof systems in MRes-$\mathcal{R}$ are polynomial time verifiable and gives a lower bound for 'regular' versions of proof systems in MRes-$\mathcal{R}$. Still several open problems remain in the scope of this paper. We point some of them as follows:

The relative strength of proof systems in MRes-$\mathcal{R}$ and MRes is still unclear. Since proof systems in MRes-$\mathcal{R}$ use strong consistency checking rules as compared to the isomorphism rule in MRes, we believe that there exists a family of QBFs which are easy for proof systems in MRes-$\mathcal{R}$ but hard for MRes.

Another direction is to establish a lower bound for proof systems in MRes-$\mathcal{R}$. It is open whether KBKF-lq formulas [3] (shown to be hard for the MRes proof system in [5]), are hard or easy for proof systems in MRes-$\mathcal{R}$. Note that, by slightly modifying the formula to KBKF-lq-split [19] it has been shown to be easy for MRes and hence making them easy for MRes-$\mathcal{R}$ as well.

References

1 Sanjeev Arora and Boaz Barak. Computational Complexity - A Modern Approach. Cambridge University Press, 2009.

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 Theory and Applications of Satisfiability Testing - SAT 2014, volume 8561 of LNCS, pages 154–169. Springer, 2014.

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

5 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, volume 182 of LIPIcs, pages 12:1–12:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020.

6 Olaf Beyersdorff, Leroy Chew, and Mikolás Janota. Proof complexity of resolution-based QBF calculi. In Ernst W. Mayr and Nicolas Ollinger, editors, 32nd International Symposium on Theoretical Aspects of Computer Science, STACS 2015, March 4-7, 2015, Garching, Germany, volume 30 of LIPIcs, pages 76–89. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2015.

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

8 Joshua Blinkhorn, Tomás Peitl, and Friedrich Slivovsky. Davis and Putnam meet Henkin: Solving DQBF with resolution. In Theory and Applications of Satisfiability Testing - SAT 2021, volume 12831 of LNCS, pages 30–46. Springer, 2021.

9 Randal E. Bryant. Graph-based algorithms for boolean function manipulation. IEEE Trans. Computers, 35(8):677–691, 1986. doi:10.1109/TC.1986.1676819.

10 Leroy Chew. Hardness and optimality in QBF proof systems modulo NP. In Chu-Min Li and Felip Manyà, editors, Theory and Applications of Satisfiability Testing - SAT 2021 - 24th International Conference, Barcelona, Spain, July 5-9, 2021, Proceedings, volume 12831 of Lecture Notes in Computer Science, pages 98–115. Springer, 2021. doi:10.1007/978-3-030-80223-3_8.

11 Leroy Chew and Friedrich Slivovsky. Towards uniform certification in QBF. In Petra Berenbrink and Benjamin Monmege, editors, 39th International Symposium on Theoretical Aspects of Computer Science, STACS 2022, March 15-18, 2022, Marseille, France (Virtual Conference), volume 219 of LIPIcs, pages 22:1–22:23. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022.

12 William J. Cook, Collette R. Coullard, and György Turán. On the complexity of cutting-plane proofs. Discrete Applied Mathematics, 18(1):25–38, 1987.

13 Adnan Darwiche and Pierre Marquis. A knowledge compilation map. J. Artif. Intell. Res., 17:229–264, 2002. doi:10.1613/jair.989.

14 Yuval Filmus, Pavel Hrubes, and Massimo Lauria. Semantic versus syntactic cutting planes. In 33rd Symposium on Theoretical Aspects of Computer Science, STACS, volume 47 of LIPIcs, pages 35:1–35:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016.

15 Gottlob Frege. Begriffsschrift, eine der arithmetischen nachgebildete Formelsprache der reinen Denkens, Halle 1879. 1967. English translation in: from Frege to Gödel, a source book in mathematical logic (J. van Heijenoord editor), Harvard University Press, Cambridge.

16 Mikolás Janota and João Marques-Silva. Expansion-based QBF solving versus Q-resolution. Theor. Comput. Sci., 577:25–42, 2015.

17 Stasys Jukna. Exponential lower bounds for semantic resolution. In Proof Complexity and Feasible Arithmetics, Proceedings of a DIMACS Workshop, volume 39, pages 163–172. DIMACS/AMS, 1996.

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

19 Meena Mahajan and Gaurav Sood. QBF merge resolution is powerful but unnatural. CoRR, abs/2205.13428, 2022.

20 G. Peterson, J. Reif, and S. Azhar. Lower bounds for multiplayer noncooperative games of incomplete information. Computers & Mathematics with Applications, 41(7):957–992, 2001. doi:https://doi.org/10.1016/S0898-1221(00)00333-3.

21 John Alan Robinson. Theorem-proving on the computer. J. ACM, 10(2):163–174, 1963.

22 Nathan Segerlind. The complexity of propositional proofs. Bull. Symb. Log., 13(4):417–481, 2007.

Appendix-1

A Missing proof and example from Section: 3.4

This example considers the situation corresponding to the case-(iii) of Theorem 6. That is, two isomorphic merge maps can be combined with an if-else and the resulting strategy will still output the same as input merge maps. Or when one of the input merge map being $*$, makes the resulting strategy diluted in the sense that for half the assignments it gives a $*$ and for others the same as the non-trivial input merge map.

▶ **Example 19.** Let $M_1^u = M_2^u = 1$ be leaf nodes in MRes proof. It implies that corresponding $H_1^u = 1$ and $H_2^u = 1$ in MRes-$\mathcal{R}$ proof. Now say MRes performs a resolution on pivot variable x which is to the left of u, resulting in $M_3^u = select(M_1^u, M_2^u)$. Whereas the corresponding MRes-$\mathcal{R}$ rule needs to be a $H_3^u = H_1^u \overset{x}{\bowtie} H_2^u$ from case(iii) (ref. Theorem 6). That is, H_3^u in function form would be defined as follows:

$$H_3^u = \begin{cases} 1 & : \quad x \\ 1 & : \quad \overline{x} \end{cases}$$

Notice how this is just a diluted way of writing the strategy $H_3^u = 1$. Hence when in the next line of MRes if a $M_4^u = 1$ which is isomorphic to M_3^u is encountered; the corresponding $H_4^u = 1$ in MRes-$\mathcal{R}$ will still remain to be consistent with H_3^u (though they might seem to be structurally different).
In the same example if $M_2^u = *$ (i.e. trivial), the strategy H_3^u would have been:

$$H_3^u = \begin{cases} 1 & : \quad x \\ * & : \quad \overline{x} \end{cases}$$

Notice how this is another way of diluting the strategy and is still consistent with $H_4^u = 1$.

Claim 7. *Every MRes-$\mathcal{M}$-proof can be transformed into an MRes-$\mathcal{R}$-proof for any representation R in exponential time.*

Proof. Given a QBF Φ and its MRes-$\mathcal{M}$-proof $\pi = L_1, ..., L_m$, where every line $L_i = \{C_i, \{M_i^u : u \in U\}\}$. We intend to build an MRes-$\mathcal{R}$-proof $\Pi = L'_1, ..., L'_m$ for Φ, where each $L'_i = \{C'_i, \{H_i^u : u \in U\}\}$.

For every line L_i in π, we keep the clause part intact while we convert the merge maps into plain functions. Further as R is a complete representation, these functions should have a corresponding representation in R; we extensively search for the same. This search terminates at some point owing to R being a complete representation. (This is the place where we used the property that R is a complete representation). The result is an MRes-$\mathcal{R}$-proof for Φ. This process is not polynomial in time but regardless still proves completeness for the family of proof systems MRes-$\mathcal{R}$. ◀

B Missing example from Section: 4

▶ **Example 20.** Let $\Phi := \exists x, y, \forall u, \exists a, b\ (x, \overline{y}, \overline{u}, a) \wedge (\overline{x}, \overline{y}, a) \wedge (\overline{x}, \overline{y}, \overline{u}, \overline{a}) \wedge (x, \overline{y}, \overline{a}) \wedge (\overline{x}, y, \overline{u}, b) \wedge (x, y, u, b) \wedge (y, \overline{b})$. The MRes-$\mathcal{T}$ proof of Φ is shown below in Table 2:

The final T-graph of winning strategy for the only universal variable u from Example 20 is shown in Figure 5. One can see that this graph is a hybrid structure of both branching programs and circuits. Since it has both 'branching' nodes ($\bowtie$ nodes) and 'circuit' nodes (# nodes).

Table 2 A MRes-$\mathcal{T}$ refutation of the false QBF in Example 20

Line	Rule	C_i	T_i^u	Type of node
L_1	axiom	$\{x, \overline{y}, a\}$	1	Leaf
L_2	axiom	$\{\overline{x}, \overline{y}, a\}$	*	Leaf
L_3	$res(L_1, L_2, x)$	$\{\overline{y}, a\}$	$T_2^u \overset{x}{\bowtie} T_1^u$	if-else
L_4	axiom	$\{\overline{x}, \overline{y}, \overline{a}\}$	1	Leaf
L_5	axiom	$\{x, \overline{y}, \overline{a}\}$	*	Leaf
L_6	$res(L_5, L_4, x)$	$\{\overline{y}, \overline{a}\}$	$T_4^u \overset{x}{\bowtie} T_5^u$	if-else
L_7	$res(L_3, L_6, a)$	$\{\overline{y}\}$	$T_3^u \circ T_6^u$	#
L_8	axiom	$\{\overline{x}, y, b\}$	1	Leaf
L_9	axiom	$\{x, y, b\}$	0	Leaf
L_{10}	$res(L_9, L_8, x)$	$\{y, b\}$	$T_8^u \overset{x}{\bowtie} T_9^u$	if-else
L_{11}	axiom	$\{y, \overline{b}\}$	*	Leaf
L_{12}	$res(L_{10}, L_{11}, b)$	$\{y\}$	$T_{10}^u \circ T_{11}^u$	#
L_{13}	$res(L_{12}, L_7, y)$	$\{\}$	$T_7^u \overset{y}{\bowtie} T_{12}^u$	if-else

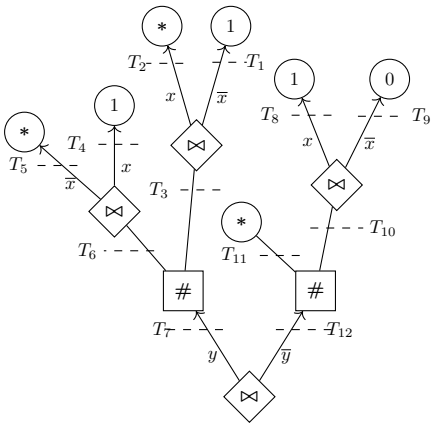


Figure 5 T_{13}^u graph for Example 20

C Missing proofs from Section: 5

Theorem *14.* *Every $(A \cup B)$-regular refutation of CR_n in any proof system belonging to MRes-$\mathcal{R}$ has size $2^{\Omega(n)}$.*

Proof. For $\mathcal{P} \in$ MRes-$\mathcal{R}$, let Π be a $\mathcal{P}$-refutation of CR_n (for $n > 2$). Let the set of lines S, S' be defined as follows:

***S′*:** This set consists of all the lines $L = (C, H^z)$ from Π such that $vars(C) \cap \{A \cup B\} = \emptyset$ and there exists a path from L to $\bot$ in G_Π consisting of lines only from S'.

***S*:** This set contains all the lines $L \in S'$ such that $L = Res(L_1, L_2, v)$ where $L_1, L_2 \notin S'$. Observe that the pivot variable v must belong to $\{A \cup B\}$.

Let $F = \bigwedge_{(C,H^z)\in S} C$. Note that F is a false CNF formula because there exists a sub-derivation $\widehat{\Pi} = \{C | \exists L = (C, H^z) \in S'\}$ which derives a $\bot$ given F. The variables in F are only x_{ij}'s where $i, j \in [n]$, therefore it consists of n^2 variables. In Lemma 15 we prove that each clause in F has width $\geq n-1$. That is each clause can be falsified by setting atleast n-1 variables to 0. Hence the number of complete assignments of X that can falsify a clause $\in F$ will be

at most $2^{n^2-(n-1)}$. Since F is a false CNF formula, all assignments to X should falsify some clause of F. Therefore, the number of clauses in F should be $\geq 2^{n-1}$. This implies that the number of lines in S is at least 2^{n-1}. Therefore, the number of lines in Π must also be exponential in n. ◀

Claim 17. *([5]) Either for all $i \in [n-1]$, C_2 has a variable of the form x_{i*}, or for all $j \in [n]$, C_2 has a variable of the form x_{*j}*

Proof. At this point in the proof of Lemma 15, we definitely know that $\overline{a_n} \in C_2$, and for all $i \in [n-1]$, for all $j \in [n]$, $x_{ij} \notin \text{var}(H_2^z)$. We prove this claim by contradiction. Suppose the claim is wrong i.e, there exists some $u \in [n-1]$ where for all $l \in [n]$ $x_{ul} \notin var(C_2)$ and some $v \in [n]$ where for all $k \in [n]$ $x_{kv} \notin var(C_2)$.

Let ρ be the minimum partial assignment falsifying C_2. Then we know that :

▷ ρ sets $a_n = 1$, leaves all other variables in $A \cup B$ unset, since they $\notin C_2$.

▷ ρ does not set any x_{ul} or x_{kv}, since by our assumptions they all are not in C_2.

Now, extend ρ to assignment α by setting:

▷ $a_u = b_v = 0$ and rest all unset variables from $A \cup B$ to 1.

▷ Also except x_{uv}, set $x_{u*} = 1$ and $x_{*v} = 0$.

Observe that the assignment α satisfies all axiom clauses except A_{uv} and B_{uv} and does not falsify any axiom.

Now extend α to α_0 and α_1 by setting $x_{uv} = 0$ and 1 respectively.

The extension α_0 satisfies one more axiom i.e. B_{uv}; similarly α_1 satisfies one more axiom i.e. A_{uv}. Note that they still do not falsify the remaining axiom. That is, α_0 does not falsify A_{uv} and similarly, α_1 does not falsify B_{uv}.

α_0 and α_1 agree everywhere except on x_{ij} , and since $x_{ij} \notin \text{var}(H_2^z)$, it follows that $H_2^z(\alpha_0) = H_2^z(\alpha_1)$, say this value is equal d.

From the proved Induction in Lemma 4, the partial strategy of universal player at every line combined with the extension of the existential assignment falsifying its clause part, should falsify some axiom of the QBF. Also, α_0 and α_1 falsify C_2, since they extend ρ. Hence, it is a contradiction that $(\alpha_{\overline{d}}, d)$ satisfies all axioms. Therefore, the claim needs to be true. ◀

D Missing example from Section: 4.3

In [8], the authors describe a way to use 2-valued OBDD functions to represent 3-valued strategy functions (including the don't care (*)). They represent each strategy function H_u as a pair of 2 functions $(H_u^\top, H_u^\bot)$ which can then represented by 2 OBDDs. This pair is simply defined as follows:

$$H_u^\top(\varepsilon) = \begin{cases} 1 & : \quad if H_u(\varepsilon) = 1 \\ 0 & : \quad otherwise \end{cases} \qquad H_u^\bot(\varepsilon) = \begin{cases} 1 & : \quad H_u(\varepsilon) = 0 \\ 0 & : \quad otherwise \end{cases}$$

In [8, Proposition 1], the authors also give the following formulas to perform union and if-else operation on OBDDs efficiently:

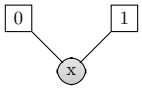
1. $(G_u \circ H_u)^\top = G_u^\top \vee H_u^\top$; $(G_u \circ H_u)^\bot = G_u^\bot \vee H_u^\bot$
   The algorithm for '$\vee$' on OBDDs (called bounded disjunction) is very well-known, it was introduced in [9]. (We will not discuss it here, rather will directly use it in Example 21).
2. $(G_u \overset{x}{\bowtie} H_u)^\top = G_u^\top \overset{x}{\bowtie} H_u^\top$; $(G_u \circ H_u)^\bot = G_u^\bot \overset{x}{\bowtie} H_u^\bot$

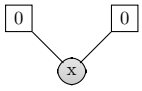
Next, we will see how to convert a $\mathcal{T}$-represented strategy into an OBDD-represented strategy. For this we use the graph of T_{13}^u from previously shown Example 20, the graph of the same is already shown in Figure 5 for ease of cross-checking.

▶ **Example 21.** For the input $\mathcal{T}$-strategy, we use T^u_{13} from Example 20 (shown in Figure 5). As explained in Section 4.3, we follow recursively from leaf-to-root in this graph, converting it into OBDD pairs and applying $\circ$ and $\bowtie$ operations (as stated above) wherever applicable. Note that in OBDDs, we fix that the left edge of non-leaf node represents the positive edge and the right one is the negative edge (for ease in drawing).

$T^{\top}_1 = 1$, $T^{\bot}_1 = 0$

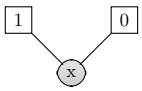
$T^{\top}_2 = 0$, $T^{\bot}_2 = 0$

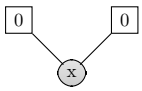
$T^{\top}_3 =$ 

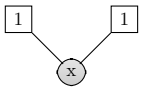
, $T^{\bot}_3 =$ 

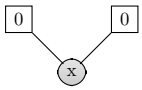
$T^{\top}_4 = 1$, $T^{\bot}_4 = 0$

$T^{\top}_5 = 0$, $T^{\bot}_5 = 0$

$T^{\top}_6 =$ 

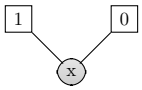
, $T^{\bot}_6 =$ 

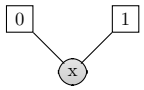
$T^{\top}_7 =$ 

, $T^{\bot}_7 =$ 

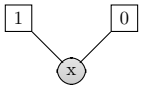
$T^{\top}_8 = 1$, $T^{\bot}_8 = 0$

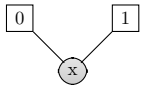
$T^{\top}_9 = 0$, $T^{\bot}_9 = 1$

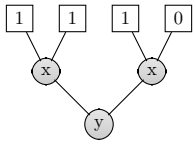
$T^{\top}_{10} =$ 

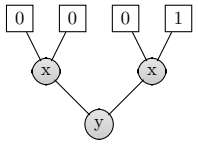
, $T^{\bot}_{10} =$ 

$T^{\top}_{11} = 0$, $T^{\bot}_{11} = 0$

$T^{\top}_{12} =$ 

, $T^{\bot}_{12} =$ 

$T^{\top}_{13} =$ 

, $T^{\bot}_{13} =$ 

This completes the conversion of $\mathcal{T}$-graph to OBDD-pairs. To cross-check let us compute the resulting function from $(T_{13}^{\top}, T_{13}^{\perp})$. The process to extract the strategy (H_u) back from the OBDD-pair $(H_u^{\top}, H_u^{\perp})$:

$$H_u(\varepsilon) = \begin{cases} 1 & : \quad if H_u^{\top}(\varepsilon) = 1 \\ 0 & : \quad if H_u^{\perp}(\varepsilon) = 1 \\ * & : \quad otherwise \end{cases}$$

So the truth-table of the resultant function (say T'_{13}) from $(T_{13}^{\top}, T_{13}^{\perp})$ is as follows:

x	y	T'_{13}
0	0	0
0	1	1
1	0	1
1	1	1

One can cross check that this is exactly the function computed by T_{13}^{u} in Figure 5.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il