Logical Methods in Computer Science
Volume 20, Issue 3, 2024, pp. 22:1–22:28
https://lmcs.episciences.org/

Submitted Dec. 19, 2023
Published Sep. 10, 2024

QBF MERGE RESOLUTION IS POWERFUL BUT UNNATURAL

MEENA MAHAJAN [a,b] AND GAURAV SOOD [a,b]

[a] The Institute of Mathematical Sciences, Chennai, India

[b] Homi Bhabha National Institute, Training School Complex, Anushaktinagar, Mumbai, India
e-mail address: meena@imsc.res.in

Abstract. The Merge Resolution proof system (M-Res) for QBFs, proposed by Beyersdorff et al. in 2019, explicitly builds partial strategies inside refutations. The original motivation for this approach was to overcome the limitations encountered in long-distance Q-Resolution proof system (LD-Q-Res), where the syntactic side-conditions, while prohibiting all unsound resolutions, also end up prohibiting some sound resolutions. However, while the advantage of M-Res over many other resolution-based QBF proof systems was already demonstrated, a comparison with LD-Q-Res itself had remained open. In this paper, we settle this question. We show that M-Res has an exponential advantage over not only LD-Q-Res, but even over LQU$^+$-Res and IRM, the most powerful among currently known resolution-based QBF proof systems. Combining this with results from Beyersdorff et al. 2020, we conclude that M-Res is incomparable with LQU-Res and LQU$^+$-Res.

Our proof method reveals two additional and curious features about M-Res: (i) M-Res is not closed under restrictions, and is hence not a natural proof system, and (ii) weakening axiom clauses with existential variables provably yields an exponential advantage over M-Res without weakening. We further show that in the context of regular derivations, weakening axiom clauses with universal variables provably yields an exponential advantage over M-Res without weakening. These results suggest that M-Res is better used with weakening, though whether M-Res with weakening is closed under restrictions remains open. We note that even with weakening, M-Res continues to be simulated by eFrege$+\forall$red (the simulation of ordinary M-Res was shown recently by Chew and Slivovsky).

1. Introduction

Testing satisfiability of CNF formulas (the propositional SAT problem) is NP-complete and is hence believed to be hard in the worst case. Despite this, modern SAT solvers routinely solve industrial SAT instances with hundreds of thousands or even millions of variables in close to linear time [Var14, BN21, MSLM21]. Recently some mathematics problems, some of which were open for almost a century, have been solved by employing SAT solvers (see

Key words and phrases: QBF, proof complexity, resolution, weakening, restrictions.

A preliminary version of this article appeared in the proceedings of the 25th International Conference on Theory and Applications of Satisfiability Testing – SAT 2022 [MS22].

Current affiliation: G. Sood, School of Computing and Electrical Engineering, Indian Institute of Technology Mandi, Kamand, Himachal Pradesh, India. Current Email: gauravsood@iitmandi.ac.in.

LOGICAL METHODS IN COMPUTER SCIENCE
DOI:10.46298/LMCS-20(3:22)2024

[HK17] for a survey). This apparent disconnect between theory and practice has led to a more detailed study of the different solving techniques.

Most successful SAT solvers use a non-deterministic algorithm called conflict-driven clause learning (CDCL) [SS99, MMZ+01], which is inspired by and an improvement of the DPLL algorithm [DP60, DLL62]. The solvers use some heuristics to make deterministic or randomized choices for the non-deterministic steps of the CDCL algorithm. The CDCL algorithm (and the resulting solvers) can be studied by analysing a proof system called resolution. Resolution contains a single inference rule, which given clauses $x \vee A$ and $\overline{x} \vee B$, allows the derivation of clause $A \vee B$ [Bla37, Rob65]. To be more precise, from a run of the CDCL algorithm (or a solver) on an unsatisfiable formula, resolution refutations of the same length (as the run of the solver) can be extracted. This means that refutation size lower bounds on resolution translate to runtime lower bounds for the CDCL algorithm and the solvers based on it. See [MSLM21] for more on CDCL based SAT solvers and [BN21] for their connection to resolution.

With SAT solvers performing so well, the community has set sights on solving Quantified Boolean formulas (QBFs). Some of the variables in QBFs are quantified universally, allowing a more succinct but also explainable encoding of many constraints. As a result, QBF solving has many more practical applications (see [SBPS19] for a survey). However, it is PSPACE-complete [SM73] and hence believed to be much harder than SAT.

The main way of tackling QBFs in proof systems is by adapting resolution to handle universal variables. There are two major ways of doing this, which have given rise to two orthogonal families of proof systems. Reduction-based systems allow dropping a universal variable from a clause if some conditions are met — proof systems Q-Res and QU-Res [KKF95, Gel12] are of this type. In contrast, expansion-based systems eliminate universal variables at the outset by expanding the universal quantifiers into conjunctions, giving a purely propositional formula — proof systems $\forall\text{Exp} + \text{Res}$ and IR [JM15, BCJ19] are of this type. It was soon observed that, under certain conditions, producing a clause containing a universal variable in both polarities (to be interpreted in a special way, not as a tautology) is not only sound but also very useful for making proofs shorter [ZM02, ELW13]. This led to new proof systems of both types: reduction-based systems LD-Q-Res, LQU-Res and LQU$^+$-Res [BJ12, BWJ14], and expansion-based system IRM [BCJ19].

Since all these proof systems degenerate to resolution on propositional formulas, lower bounds for resolution continue to hold for these systems as well. However such lower bounds do not tell us much about the relative powers and weaknesses of these systems. QBF proof complexity aims to understand this. This is done by finding formula families which have polynomial-size refutations in one system but require super-polynomial size refutations in the other system. For example, among the reduction-based and expansion-based resolution systems, LQU$^+$-Res and IRM respectively are the most powerful and are known to be incomparable [BWJ14, BCJ19].

In this paper, we study a proof system called Merge Resolution (M-Res). This system was proposed in [BBM21] with the goal of circumventing a limitation of LD-Q-Res. The main feature of this system is that each line of the refutation contains information about partial strategies for the universal player in the standard two-player evaluation game associated with QBFs. These strategies are built up as the proof proceeds. The information about these partial strategies allows some resolution steps which are blocked in LD-Q-Res. This makes M-Res very powerful — it has short refutations for formula families requiring exponential-size refutations in Q-Res, QU-Res, $\forall\text{Exp} + \text{Res}$, and IR, and also in the system $\text{CP} + \forall\text{red}$

introduced in [BCMS18]. However, the authors of [BBM21] did not show any advantage over LD-Q-Res — the system that M-Res was designed to improve. They only showed advantage over a restricted version of LD-Q-Res, the system reductionless LD-Q-Res. In a subsequent paper [BBM$^+$24], limitations of M-Res were shown — there are formula families which have polynomial-size refutations in QU-Res, LQU-Res, LQU$^+$-Res and CP + $\forall$red, but require exponential-size refutations in M-Res. This, combined with the results from [BBM21], showed that M-Res is incomparable with QU-Res and CP + $\forall$red. More recently, it has been shown that eFrege + $\forall$red proof system p-simulates M-Res [CS24]. On the solving side, M-Res has recently been used to build a solver, though with a different representation for strategies [BPS21]. Some variants of M-Res have been studied in [CS23] from a theoretical viewpoint.

In this paper, we show that M-Res is indeed quite powerful, answering one of the main questions left open in [BBM21]. We show that there are formula families which have polynomial-size refutations in M-Res but require exponential-size refutations in LD-Q-Res. In fact, we show that there are formula families having short refutations in M-Res but requiring exponential-size refutations in LQU$^+$-Res and IRM — the most powerful resolution-based QBF proof systems. Combining this with the results in [BBM$^+$24], we conclude that M-Res is incomparable with LQU-Res and LQU$^+$-Res; see Theorem 3.8 and Theorem 3.14.

The power of M-Res is shown using modifications of two well-known formula families: KBKF-lq [BWJ14] which is hard for M-Res [BBM$^+$24], and QUParity [BCJ19] which we believe is also hard. The main observation is that the reason making these formulas hard for M-Res is the mismatch of partial strategies at some point in the refutation. This mismatch can be eliminated if the formulas are modified appropriately. The resultant formulas, called KBKF-lq-split and MParity, have polynomial-size refutations in M-Res but require exponential-size refutations in IRM and LQU$^+$-Res respectively.

We observe that the modification of KBKF-lq is actually a weakening of the clauses. This leads to an observation that weakening adds power to M-Res. Weakening is a rule that is sometimes augmented to resolution. This rule allows the derivation of $A \vee x$ from A, provided that A does not contain the literal $\overline{x}$. The weakening rule is mainly used to make resolution refutations more readable — it can not make them shorter [Ats04]. The same holds for all the known resolution-based QBF proof systems with the exception of M-Res — allowing weakening can make M-Res refutations exponentially shorter. We distinguish between two types of weakenings, namely existential clause weakening and strategy weakening. Both these weakenings were defined in the original paper [BBM21] in which M-Res was introduced. However, these weakenings were used only for Dependency-QBFs (DQBFs); in that setting they are necessary for completeness. The potential use of weakening for QBFs was not explicitly addressed. Here, we show that existential clause weakening adds exponential power to M-Res; see Theorem 4.2. We do not know whether strategy weakening adds power to M-Res. However, we show that it does add exponential power to regular M-Res; see Theorem 4.6. At the same time, weakening of any or both types does not make M-Res unduly powerful; we show in Theorem 4.15 that eFrege + $\forall$red polynomially simulates (p-simulates) M-Res even with both types of weakenings added. This is proven by observing that the p-simulation of M-Res in [CS24] can very easily be extended to handle weakenings.

Another observation from our main result is that M-Res is not closed under restrictions. Closure under restrictions is a very important property of proof systems. For a (QBF) proof system, it means that restricting a false formula by a partial assignment to some of the (existential) variables does not make the formula much harder to refute. Note that a

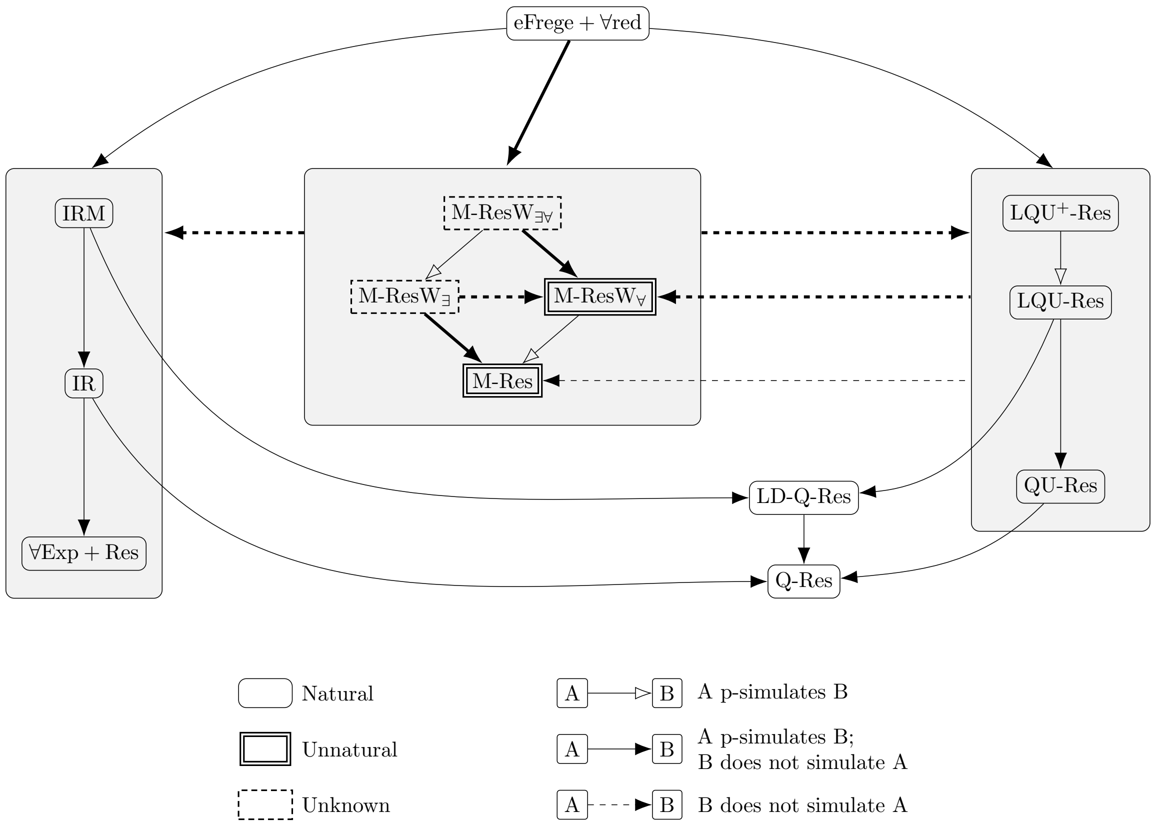

Figure 1. Relations among resolution-based QBF proof systems, with new results and observations highlighted using thicker lines. In addition, regular M-ResW$_\forall$ strictly p-simulates regular M-Res. Lines from a big grey box mean that the line is from every proof system within the box.

refutation of satisfiability of a formula implicitly encodes a refutation of satisfiability of all its restrictions, and it is reasonable to expect that such refutations can be extracted without paying too large a price. This is indeed the case for virtually all known proof systems to date. Algorithmically, CDCL-based solvers work by setting some variables and simplifying the formula [MSLM21]. Without closure under restrictions, setting a bad variable may make the job of refuting the formula exponentially harder. Because of this reason, proofs systems which are closed under restrictions have been called *natural proof systems* [BKS04]. We show in Theorem 4.16 that M-Res, with and without strategy weakening, is unnatural. We believe this would mean that it is hard to build QBF solvers based on it. On the other hand, we do not yet know whether it remains unnatural if existential clause weakening or both types of weakenings are added. We believe that this is the most important open question about M-Res — a negative answer can salvage it.

Our results are summarized in Figure 1.

2. Preliminaries

The sets $\{1, 2, \ldots, n\}$ and $\{m, m+1, \ldots, n\}$ are abbreviated as $[n]$ and $[m, n]$ respectively. A literal is a variable or its negation; a clause is a disjunction of literals. We will interchangeably denote clauses as disjunctions of literals as well as sets of literals. A propositional formula in conjunctive normal form (cnf) is a conjunction of clauses, equivalently a set of clauses.

2.1. **Quantified Boolean Formulas.**

A Quantified Boolean Formula (QBF) in prenex conjunctive normal form (p-cnf), denoted $\Phi = \mathcal{Q}.\phi$, consists of two parts: (i) a quantifier prefix $\mathcal{Q} = Q_1 Z_1, Q_2 Z_2, \ldots, Q_n Z_n$ where the Z_i are pairwise disjoint sets of variables, each $Q_i \in \{\exists, \forall\}$, and $Q_i \neq Q_{i+1}$; and (ii) a conjunction of clauses ϕ with variables in $Z = Z_1 \cup \cdots \cup Z_n$. In this paper, when we say QBF, we mean a p-cnf QBF. The set of existential (resp. universal) variables of Φ, denoted X (resp. U), is the union of Z_i for which $Q_i = \exists$ (resp. $Q_i = \forall$).

The semantics of a QBF is given by a two-player evaluation game played on the QBF. In a run of the game, the existential player and the universal player take turns setting the existential and the universal variables respectively in the order of the quantification prefix. The existential player wins the run of the game if every clause is set to true. Otherwise the universal player wins. The QBF is true (resp. false) if and only if the existential player (resp. universal player) has a strategy to win all potential runs, i.e. a winning strategy. The winning strategy for the existential (resp. universal) player is called a model (resp. countermodel).

2.2. **Proof systems.**

Definition 2.1 [CR79]**.** For $L \subseteq \Sigma^*$, a proof system for L is a polynomial-time computable function $f : \Delta^* \longrightarrow \Sigma^*$ whose range is exactly L. For some $x \in \Delta^*$ and $y \in \Sigma^*$, if $y = f(x)$, then x is an f-proof of y, that is, a proof of y in the proof system f (a proof that $y \in L$). The size of the proof is the length of x.

The condition $\text{range}(f) = L$ is often stated in two parts:

- *Soundness:* For any $y \in \Sigma^*$ and $x \in \Delta^*$, if $f(x) = y$, then $y \in L$.
- *Completeness:* For every $y \in L$, there exists $x \in \Delta^*$ such that $f(x) = y$.

In this paper, we will be interested in the languages of True QBFs (TQBF) and False QBFs (FQBF). A proof system for the language FQBF is also called a refutational system and the proofs in this system are called refutations.

To compare the strength of different proof systems, we use the notion of simulations and p-simulations, Def 1.5 in [CR79]; see also Def. 1.5.4 in [Kra19].

Definition 2.2. Let P and Q be two proof systems for $QBFs$.

- We say that P *simulates* Q if there is a computable function g that transforms proofs in Q to proofs in P with at most a polynomial blow-up in size.
- If, in addition, g is polynomial-time computable, then we say that P *polynomially simulates (p-simulates)* Q.
- If P p-simulates Q but Q does not simulate P, then we say that P is *strictly stronger* than Q.

For a formula Φ and a partial assignment ρ to some of its variables, $\Phi{\restriction_{\rho}}$ denotes the restricted formula resulting from setting the specified variables according to ρ.

Definition 2.3. A QBF proof system P is *closed under restrictions* if for every false QBF Φ and every partial assignment ρ to some existential variables, the size of the smallest P-refutation of $\Phi\lceil_\rho$ is at most polynomial in the size of the smallest P-refutation of Φ.

Remark 2.4. Sometimes a stricter definition is used, requiring that a refutation of $\Phi\lceil_\rho$ be constructible in polynomial time from every refutation of Φ. We will prove that M-Res is not closed under restrictions for the weaker definition (and hence also for the stricter definition).

Definition 2.5 [BKS04]**.** A proof system is *natural* if it is closed under restrictions.

2.3. The Merge Resolution proof system.

Merge Resolution (M-Res) is a proof system for refuting false QBFs. Its original definition is rather technical and can be found in [BBM21]. Here, we first give an informal description and then reproduce the definition as presented in [BBM+24]. The reader already familiar with the proof system M-Res can skip this section altogether.

An M-Res refutation of a QBF $\Phi = Q.\phi$ is a sequence of lines. Each line consists of an ordered pair — the first part of the pair is a clause C over the existential variables; and the second part is a set of branching programs, $\{M^u \mid u \in U\}$, one branching program for each universal variable. These branching programs are called merge maps and represent partial strategies h^u for the corresponding universal variables; the internal nodes of the merge map M^u are labelled by the existential variables to the left of u in the quantifier prefix, and the leaves are labelled by 0 and 1. Merge maps with no branching nodes are called *trivial merge maps*, denoted by $*$. A trivial merge map computes an 'undefined' function.

The rules of the system maintain the invariant that at every line, the set of functions $\{h^u\}$ gives a partial strategy (for the universal player) that wins whenever the existential player plays from the set of assignments falsifying C. The goal is to derive the line with the empty clause; the corresponding strategy at this line will then be a countermodel.

Each line of the refutation is either obtained from an axiom (i.e. a clause of ϕ), or is obtained from two previous lines by resolution on the clauses. For axioms, the corresponding line is defined in a way that satisfies the desired invariant. At a resolution step, if the pivot is x, then for universal variable u right of x, the partial strategies are combined via a branching on x. To control the size blow-up, common parts of the merge maps are identified through line numbering and are reused in the new merge map. For u left of x, such a combination is disallowed as it would not be consistent with the semantics of the two-player evaluation game. Thus the resolution is blocked if for any such u, the antecedents have different non-trivial strategies. However, if both strategies are identical, or if one of them is trivial, then carrying the non-trivial strategy forward maintains the invariant. Deciding whether the strategies are identical may not be easy in general, but with the chosen representation of merge maps, isomorphism is easy to check.

We now give the formal definitions.

Definition 2.6. *Merge maps* are deterministic branching programs, specified by a sequence of instructions of one of the following two forms:

- $\langle$`Instruction i`$\rangle$: `b`, where $b \in \{*, 0, 1\}$.[1]
  Merge maps containing a single such instruction are called simple. In particular, if $b = *$, then they are called trivial.

[1]In [BBM21], the notation used is $b \in \{*, u, \overline{u}\}$; $u, \overline{u}, *$ denote $u = 1, u = 0$, undefined respectively.

- $\langle$`Instruction i`$\rangle$: `If x = 0 then go to` $\langle$`Instruction j`$\rangle$ `else go to` $\langle$`Instruction k`$\rangle$, for some $j, k < i$. In a merge map M for u, all queried variables x must precede u in the quantifier prefix.

  Merge maps with such instructions are called complex.

(All instruction numbers are positive integers.) The merge map M^u computes a partial strategy for the universal variable u starting at the largest instruction number (the leading instruction) and following the instructions in the obvious way. The value $*$ denotes an undefined value.

Definition 2.7. Two merge maps M_1 and M_2 are said to be *consistent*, denoted $M_1 \bowtie M_2$, if for every instruction number i appearing in both M_1, M_2, the instructions with instruction number i are identical.

When two merge maps, M_1 and M_2, are consistent, it is possible to build the merge map: `If x = 0 then go to` $\mathtt{M_1}$ `else go to` $\mathtt{M_2}$ without repeating the common parts of M_1 and M_2. To be more precise, the new merge map will contain all instructions of M_1 and M_2 and the following additional instruction: `If x = 0 then go to` $\langle$`leading instruction of` $\mathtt{M_1}\rangle$ `else go to` $\langle$`leading instruction of` $\mathtt{M_2}\rangle$.

Definition 2.8. Two merge maps M_1, M_2 are said to be *isomorphic*, denoted $M_1 \simeq M_2$, if there is a bijection between the instruction numbers in M_1 and M_2 transforming M_1 into M_2.

Note that the isomorphism only allows renumbering the instructions, not permuting the existence variables. Also note that isomorphic merge maps compute the same function.

Definition 2.9. The *proof system M-Res* has the following rules:

(1) *Axiom:* For a clause A in ϕ, let C be the existential part of A. For each universal variable u, let b_u be the value u must take to falsify A; if $u \notin \mathrm{var}(A)$, then $b_u = *$. For any natural number i, the line $(C, \{M^u : u \in U\})$ where each M^u is the simple merge map $\langle$`Instruction i`$\rangle$: $\mathtt{b_u}$ can be derived in M-Res.

(2) *Resolution:* From lines $L_a = (C_a, \{M^u_a : u \in U\})$ for $a \in \{0, 1\}$, in M-Res, the line $L = (C, \{M^u : u \in U\})$ can be derived, where for some $x \in X$,
   - $C = \mathrm{res}(C_0, C_1, x)$, and
   - for each $u \in U$: either (1) M^u_a is trivial and $M^u = M^u_{1-a}$ for some a; or (2) $M^u = M^u_0 \simeq M^u_1$; or (3) x precedes u, $M_1 \bowtie M_2$ and M^u has all the instructions of M^u_1 and M^u_2 in addition to the following leading instruction: `If x = 0 then go to` $\langle$`leading instruction of` $\mathtt{M^u_1}\rangle$ `else go to` $\langle$`leading instruction of` $\mathtt{M^u_2}\rangle$. The instruction number of this leading instruction is the number (position) of the line L in the derivation.

   With slight abuse of notation, we will call L the resolvent of L_0 and L_1 with pivot x, and denote this by $L = \mathrm{res}(L_0, L_1, x)$.

   Note that [BBM21] also requires that the positive literal of the pivot appears in the first argument, so $x \in C_0$ (i.e. the clause at line L_0) and $\overline{x} \in C_1$ (the clause at line L_1). However, this was only for syntactic convenience, and the way we formulate our arguments, this is not necessary.)

Note that the entire merge maps are not stored at each line, only the leading instruction specific to the line. Due to consistency when combining merge maps, this is enough information to build the entire map from the derivation. As noted in [BBM21] (Proposition

19), for lines within the same derivation, the corresponding merge maps are always consistent. Therefore, in the above definition, we don't have to explicitly do a consistency check.

Definition 2.10. An M-Res *refutation* is a derivation using the rules of M-Res and ending in a line with the empty existential clause. The size of the refutation is the number of lines.

(Note that the bit-size of representing the refutation is always polynomially bounded in the length of the refutation (the number of lines), so defining size as refutation length is acceptable.)

A refutation can be represented as a graph (with edges directed from the antecedents to the consequent, hence from the axioms to the final line). We denote the graph corresponding to refutation Π by G_Π.

Definition 2.11. Let Y be a subset of the existential variables X of Φ. We say that an M-Res refutation Π of Φ is Y*-regular* if for each $y \in Y$, there is no leaf-to-root path in G_Π that uses y as pivot more than once. An X-regular proof is simply called a *regular proof.* If G_Π is a tree, then we say that Π is a *tree-like proof.*

For concreteness, we reproduce a simple example from [BBM21]:

Example 2.12. For the QBF $\exists x, \forall u, \exists t. (x \vee u \vee t) \wedge (\overline{x} \vee \overline{u} \vee t) \wedge \left(x \vee u \vee \overline{t}\right) \wedge \left(\overline{x} \vee \overline{u} \vee \overline{t}\right)$, here is an M-Res refutation.

$$\dfrac{\dfrac{x \vee t, u = 0 \qquad \overline{x} \vee t, u = 1}{t, u = x} \qquad \dfrac{x \vee \overline{t}, u = 0 \qquad \overline{x} \vee \overline{t}, u = 1}{\overline{t}, u = x}}{\square, u = x}$$

(To be pedantic, each line should contain the merge map for u. For simplicity, we avoid it here, describing only the function computed by the merge map.)

3. Power of Merge Resolution

In this section, we prove that neither IRM nor LQU$^+$-Res simulates M-Res, and therefore M-Res has an advantage over these proof systems (as well as over LD-Q-Res, which they both simulate).

3.1. **Advantage over IRM.**

To show that M-Res is not simulated by IRM, we use a variant of the well-studied KBKF formula family. This family was first introduced in [KKF95], and along with multiple variants, has been a very influential example in showing many separations. In particular, it was used to prove that LD-Q-Res is strictly stronger than Q-Res [ELW13]. The variant KBKF-lq was defined in [BWJ14] and used to show that LD-Q-Res does not simulate QU-Res. In [BBM$^+$24], KBKF-lq was also shown to require exponentially large M-Res refutations. We reproduce the definitions of these formulas and provide some intuition about their meaning. We then define two further variants that will be useful for our purpose.

Definition 3.1 [KKF95]**.** KBKF_n is the QBF with the quantifier prefix $\exists d_1, e_1, \forall x_1, \ldots, \exists d_n, e_n, \forall x_n, \exists f_1, \ldots, f_n$ and with the following clauses:

$$\begin{array}{lll} A_0 = \{\overline{d_1}, \overline{e_1}\} & & \\ A_i^d = \{d_i, x_i, \overline{d_{i+1}}, \overline{e_{i+1}}\} & A_i^e = \{e_i, \overline{x_i}, \overline{d_{i+1}}, \overline{e_{i+1}}\} & \forall i \in [n-1] \\ A_n^d = \{d_n, x_n, \overline{f_1}, \ldots, \overline{f_n}\} & A_n^e = \{e_n, \overline{x_n}, \overline{f_1}, \ldots, \overline{f_n}\} & \\ B_i^0 = \{x_i, f_i\} & B_i^1 = \{\overline{x_i}, f_i\} & \forall i \in [n] \end{array}$$

We explain below why the KBKF formulas are false. This will also provide some intuition about the meaning of the formulas.

Fact 3.2 [KKF95]**.** The KBKF formulas are false.

Proof. Consider the following strategy for the universal player: for each $i \in [n]$, set $x_i = d_i$. We will show that this strategy is a winning strategy for the universal player [2].

To see this, we show by induction on i that if the universal player plays according to this strategy, then the existential player must either set $d_i = 0$ or $e_i = 0$, or one of the A_j clauses for some $j < i$ (either A_j^d or A_j^e when $j > 0$) will be falsified. The base case $i = 1$ is immediate since A_0 has exactly the literals $\overline{d_1}, \overline{e_1}$. Consider $i \geq 2$, say $i = k + 1$. If some A_j clause for $j < k$ is already falsified, then there is nothing to prove. Otherwise, by the induction hypothesis, at least one of d_k, e_k is set to 0. Suppose $d_k = 0$. Then setting $x_k = d_k = 0$ reduces clause A_k^d to $\{\overline{d_{k+1}}, \overline{e_{k+1}}\}$. Otherwise, $d_k = 1$, and by induction, $e_k = 0$. Now, setting $x_k = d_k = 1$ reduces clause A_k^e to $\{\overline{d_{k+1}}, \overline{e_{k+1}}\}$. Either way, the existential player must set one of $\overline{d_{k+1}}, \overline{e_{k+1}}$ to 0, or falsify an A_k clause.

If the existential player has not yet lost the game after x_n is set, then we know that either d_n or e_n is set to 0. To satisfy A_n^d and A_n^e, the existential player must set $f_i = 0$ for some $i \in [n]$. This falsifies one of B_i^0, B_i^1, and the existential player loses the game. □

The KBKF-lq formulas are obtained from KBKF by adding some negated f literals to some clauses. This is done to make the formulas hard for the LD-Q-Res proof system by blocking resolution steps that would otherwise be allowed.

Definition 3.3 [BWJ14]**.** KBKF-lq_n is the QBF with the quantifier prefix $\exists d_1, e_1, \forall x_1, \ldots, \exists d_n, e_n, \forall x_n, \exists f_1, \ldots, f_n$ and with the following clauses:

$$\begin{array}{lll} A_0 = \{\overline{d_1}, \overline{e_1}, \overline{f_1}, \ldots, \overline{f_n}\} & & \\ A_i^d = \{d_i, x_i, \overline{d_{i+1}}, \overline{e_{i+1}}, \overline{f_1}, \ldots, \overline{f_n}\} & A_i^e = \{e_i, \overline{x_i}, \overline{d_{i+1}}, \overline{e_{i+1}}, \overline{f_1}, \ldots, \overline{f_n}\} & \forall i \in [n-1] \\ A_n^d = \{d_n, x_n, \overline{f_1}, \ldots, \overline{f_n}\} & A_n^e = \{e_n, \overline{x_n}, \overline{f_1}, \ldots, \overline{f_n}\} & \\ B_i^0 = \{x_i, f_i, \overline{f_{i+1}}, \ldots \overline{f_n}\} & B_i^1 = \{\overline{x_i}, f_i, \overline{f_{i+1}}, \ldots \overline{f_n}\} & \forall i \in [n-1] \\ B_n^0 = \{x_n, f_n\} & B_n^1 = \{\overline{x_n}, f_n\} & \end{array}$$

The proof of Fact 3.2 can easily be modified to show that the KBKF-lq formulas are false. (Consider any run of the game, where all variables are set, and the universal player has set each x_i to d_i. If all B clauses are satisfied, then working backwards we see that each f_i must have been set to 1. So the f literals cannot satisfy any of the A clauses. Now,

[2]Another winning strategy is: for all $i \in [n]$, set $x_i = \overline{e_i}$.

working backwards again through the A clauses, we see that if some d_i and e_i are not both set to 1, then an A_i clause is falsified, and otherwise A_0 is falsified.)

We now define two new formula families: KBKF-lq-weak and KBKF-lq-split.

Definition 3.4. KBKF-lq-weak has the same quantifier prefix as KBKF, and all the A-clauses of KBKF-lq, but it has the following clauses instead of B_i^0 and B_i^1:

$$\left.\begin{array}{ll} \text{weak-B}_i^0 & = d_i \vee B_i^0 \\ \text{weak-B}_i^1 & = \overline{d_i} \vee B_i^1 \end{array}\right\} \qquad \forall i \in [n]$$

Definition 3.5. KBKF-lq-split has all variables of KBKF-lq and one new variable t quantified existentially in the first block, so its quantifier prefix is $\exists t, \exists d_1, e_1, \forall x_1, \dots, \exists d_n, e_n, \forall x_n, \exists f_1, \dots, f_n$. It has all the A-clauses of KBKF-lq, but the following clauses instead of B_i^0 and B_i^1:

$$\left.\begin{array}{ll} \text{split-B}_i^0 & = t \vee B_i^0 \\ \text{split-B}_i^1 & = t \vee B_i^1 \\ T_i^0 & = \{\bar{t}, d_i\} \\ T_i^1 & = \{\bar{t}, \overline{d_i}\} \end{array}\right\} \qquad \forall i \in [n]$$

It is straightforward to see that both these formulas are false as well. With the universal player's strategy of setting $x_i = d_i$, each weak-B$_i^b$ clause is effectively B_i^b, so the strategy is a winning strategy for KBKF-lq-weak as well. In KBKF-lq-split, if t is set to 1, then one clause in each T_i^0, T_i^1 pair is falsified. Otherwise, the remaining formula is the same as KBKF-lq, which we know is false.

Lemma 3.6. *KBKF-lq-weak has polynomial-size M-Res refutations.*

Proof. We use the clause-names A_0, A_i^d, weak-B$_i^0$ etc. to denote the clause, merge map pair corresponding to the respective axioms.

Let L_i'' denote the M-Res-resolvent of weak-B$_i^0$ and weak-B$_i^1$. It has only one non-trivial merge map, setting $x_i = d_i$. Starting with A_0, resolve in sequence with A_1^e, A_1^d, A_2^e, A_2^d, and so on up to A_n^e, A_n^d to derive the line with all negated f literals and merge maps computing $x_i = d_i$ for each i. Now sequentially resolve this with L_1'', L_2'', up to L_n'' to obtain the empty clause. It can be verified that none of these resolutions are blocked, and the final merge maps compute the winning strategy $x_i = d_i$ for each i.

The refutation is pictorially depicted below.

$$\dfrac{\dfrac{A_0 \qquad A_1^e}{L_1^e} \qquad A_1^d}{L_1^d}$$

$$\vdots$$

$$\dfrac{\dfrac{\dfrac{L_{n-1}^d \qquad A_n^e}{L_n^e} \qquad A_n^d}{L'_1} \qquad \dfrac{\text{weak-B}_1^0 \qquad \text{weak-B}_1^1}{L''_1}}{L'_2}$$

$$\vdots$$

$$\dfrac{L'_n \qquad \dfrac{\text{weak-B}_n^0 \qquad \text{weak-B}_n^0}{L''_n}}{(\Box, \{x_1 = d_1, \dots, x_n = d_n\})}$$

Here $L_i^e, L_i^d, L'_i, L''_i$, for all $i \in [n]$, are the following lines, with only non-trivial merge maps written explicitly:

- $L_1^e = \left(\left\{\overline{d_1}, \overline{d_2}, \overline{e_2}, \overline{f_1}, \dots, \overline{f_n}\right\}, \{x_1 = 1\}\right)$
- $L_i^e = \left(\left\{\overline{d_i}, \overline{d_{i+1}}, \overline{e_{i+1}}, \overline{f_1}, \dots, \overline{f_n}\right\}, \{x_1 = d_1, \dots, x_{i-1} = d_{i-1}, x_i = 1\}\right)$ for all $i \in [2, n-1]$
- $L_n^e = \left(\left\{\overline{d_n}, \overline{f_1}, \dots, \overline{f_n}\right\}, \{x_1 = d_1, \dots, x_{n-1} = d_{n-1}, x_n = 1\}\right)$
- $L_i^d = \left(\left\{\overline{d_{i+1}}, \overline{e_{i+1}}, \overline{f_1}, \dots, \overline{f_n}\right\}, \{x_1 = d_1, \dots, x_i = d_i\}\right)$ for all $i \in [n-1]$
- $L'_i = \left(\left\{\overline{f_i}, \dots, \overline{f_n}\right\}, \{x_1 = d_1, \dots, x_n = d_n\}\right)$ for all $i \in [n]$
- $L''_i = \left(\left\{f_i, \overline{f_{i+1}}, \dots \overline{f_n}\right\}, \{x_i = d_i\}\right)$ for all $i \in [n-1]$
- $L''_n = (\{f_n\}, \{x_n = d_n\})$ □

Observe that the extra d_i variable in weak-B_i^0 and weak-B_i^1 (in contrast to B_i^0 and B_i^1 in KBKF-lq) allows us to resolve these two lines. This gives the clause L''_i whose merge map computes $x_i = d_i$. This merge map is isomorphic to the merge map for x_i in the line derived by resolving the A_i lines. This isomorphism allows the polynomial-size refutation.

Lemma 3.7. *KBKF-lq-split has polynomial-size M-Res refutations.*

Proof. For each $i \in [n]$ and $k \in \{0, 1\}$, resolving split-B_i^k and T_i^k yields weak-B_i^k. This gives us the KBKF-lq-weak formula family which, as shown in Lemma 3.6, has polynomial-size M-Res refutations. □

Theorem 3.8. *IRM does not simulate M-Res.*

Proof. The KBKF-lq-split formula family witnesses the separation. By Lemma 3.7, it has polynomial-size M-Res refutations. Restricting it by setting $t = 0$ gives the family KBKF-lq, which requires exponential size to refute in IRM, [BCJ19]. Since IRM is closed under restrictions (Lemma 11 in [BCJ19]), KBKF-lq-split also requires exponential size to refute in IRM. □

3.2. **Advantage over LQU$^+$-Res.** To show that LQU$^+$-Res does not simulate M-Res, we need a formula family which has polynomial-size refutations in M-Res but requires exponential-size refutations in LQU$^+$-Res. We define a new formula family called MParity, as a modification of the QParity formula family [BCJ19]. The polynomial-size M-Res refutation of MParity is obtained by mimicking the LD-Q-Res refutation of QParity with some modifications. We then show that MParity requires exponential-size LQU$^+$-Res refutations. Since LQU$^+$-Res polynomially simulates LD-Q-Res and LQU-Res, we get the non-simulation result with respect to these proof systems also.

Let us first give a brief history of QParity and other formulas based on it. QParity was first defined in [BCJ19] and was used to show that Q-Res does not p-simulate $\forall\mathrm{Exp}+\mathrm{Res}$ [BCJ19] and LD-Q-Res [Che17]. (A subsequent elegant argument in [BBH19] reproved its hardness for QU-Res and $\mathrm{CP}+\forall\mathrm{red}$.) The variant LQParity, also defined in [BCJ19], was used to show that LD-Q-Res does not p-simulate $\forall\mathrm{Exp}+\mathrm{Res}$. Finally, the variant QUParity, built by duplicating the universal variable of LQParity, was used to show that LQU$^+$-Res does not p-simulate $\forall\mathrm{Exp}+\mathrm{Res}$.

We give the definition of QParity, informally describe the variants LQParity and QUParity, and then define our new variant MParity. We will use the following notation. For variables o, o_1, o_2, let even-parity(o_1, o) and even-parity(o_1, o_2, o) be the following sets of clauses[3]:

$$\begin{aligned}\text{even-parity}(o_1,o) &= \{\overline{o_1}\vee o, o_1\vee\overline{o}\},\\ \text{even-parity}(o_1,o_2,o) &= \{\overline{o_1}\vee\overline{o_2}\vee\overline{o}, \overline{o_1}\vee o_2\vee o, o_1\vee\overline{o_2}\vee o, o_1\vee o_2\vee\overline{o}\}\end{aligned}$$

We note that even-parity of a list of variables is just the CNF representation of the constraint that the number of variables set to 'True' is even. That is, even-parity(o_1, o) is satisfied iff $o \equiv o_1 \pmod 2$, and even-parity(o_1, o_2, o) is satisfied iff $o \equiv o_1 + o_2 \pmod 2$.

Definition 3.9 [BCJ19]**.** QParity$_n$ is the QBF $\exists x_1,\dots,x_n,\forall z,\exists t_1,\dots,t_n.\ \left(\wedge_{i\in[n+1]}\zeta_i\right)$ where:

$$\begin{aligned}\zeta_1 &= \text{even-parity}(x_1,t_1);\\ \zeta_i &= \text{even-parity}(t_{i-1},x_i,t_i), \qquad \forall i\in[2,n];\\ \zeta_{n+1} &= \{t_n\vee z, \overline{t_n}\vee\overline{z}\}.\end{aligned}$$

With the same quantifier prefix, replacing each clause C of QParity that does not contain z with the two clauses $C\vee z$ and $C\vee\overline{z}$ gives the family LQParity.

To obtain QUParity, the universal variable is duplicated. That is, the block $\forall z$ is replaced with the block $\forall z_1, z_2$. Each clause of the form $C\vee z$ in LQParity is replaced with the clause $C\vee z_1\vee z_2$, and each clause of the form $C\vee\overline{z}$ is replaced with the clause $C\vee\overline{z_1}\vee\overline{z_2}$.

It is easy to see why these formulas are false: in QParity, satisfying the ζ_i clauses for $i\in[n]$ forces t_n to take exactly the value $\sum_i x_i \mod 2$. Since z is universally quantified, setting z to the opposite value will falsify one of the ζ_{n+1} clauses. The tweaks to obtain QUParity and LQParity do not alter this; the same strategy (duplicated for z_1, z_2) remains a winning strategy for the universal player.

The short LD-Q-Res refutation of QParity (from [Che17, p. 54]) relies on the fact that most axioms do not have universal variable z. This enables steps in which a merged literal z^* is present in one antecedent but there is no literal over z in the other antecedent. LQParity is created from QParity by replacing each clause C not containing z by two clauses $C\vee z$ and $C\vee\overline{z}$. Since every axiom of LQParity (and hence also each derived clause) now has a literal

[3]In some prior papers considering this formula, these clause sets are denoted as xor. However in the wider circuit/Boolean-function-complexity community, xor or parity refer to odd parity. For clarity, we make the condition explicit by saying even-parity.

over z, we can no longer resolve clauses containing the merged literal z^* with any other clause. This forbids the creation of merged literals, which in turn, forbids all possible short refutations. The same problem seems to occur in M-Res also. In an M-Res refutation, we have merge maps instead of (starred and unstarred) universal literals, and resolution steps are allowed if the merge maps at the antecedents are isomorphic. However, we do not know of any way of converting the constant merge maps at the axioms to merge maps which pass the isomorphism checks in later steps of the refutation. We solve this problem by defining a new formula family called MParity. The MParity family is obtained from QUParity by modifying some clauses and adding some auxiliary clauses. The auxiliary clauses help in converting the constant merge maps at the axioms of the original clauses to merge maps that pass the isomorphism tests.

Definition 3.10. MParity_n is the following QBF:

$$\underset{i,j\in[n]}{\exists} a_{i,j}, \exists x_1,\dots,x_n, \forall z_1,z_2, \exists t_1,\dots,t_n.\ \left(\underbrace{\left(\bigwedge_{i\in[n+1]} \psi_i\right)}_{\psi} \wedge \underbrace{\left(\bigwedge_{i\in[n-1]} \delta_i\right)}_{\delta}\right)$$

where ψ and δ are defined as follows:

- for all $C \in \text{even-parity}\,(x_1, t_1)$, ψ_1 consists of the following clauses: $A^0_{1,C} = C \cup \{z_1, z_2, a_{1,n}\}$ and $A^1_{1,C} = C \cup \{\overline{z_1}, \overline{z_2}, a_{1,n}\}$,
- for all $i \in [2, n-1]$, for all $C \in \text{even-parity}\,(t_{i-1}, x_i, t_i)$, ψ_i consists of the following clauses: $A^0_{i,C} = C \cup \{z_1, z_2, a_{i,n}\}$ and $A^1_{i,C} = C \cup \{\overline{z_1}, \overline{z_2}, a_{i,n}\}$,
- for all $C \in \text{even-parity}\,(t_{n-1}, x_n, t_n)$, ψ_n consists of the following clauses: $A^0_{i,C} = C \cup \{z_1, z_2\}$ and $A^1_{i,C} = C \cup \{\overline{z_1}, \overline{z_2}\}$,
- ψ_{n+1} consists of the clauses $\{t_n, z_1, z_2\}$ and $\{\overline{t_n}, \overline{z_1}, \overline{z_2}\}$, and
- for all $i \in [n-1]$, δ_i consists of the following clauses:

$$B^0_{i,j} = \{\overline{a_{i,j}}, x_j, a_{i,j-1}\}, \qquad B^1_{i,j} = \{\overline{a_{i,j}}, \overline{x_j}, a_{i,j-1}\} \qquad \forall j \in \{n, n-1, \dots, i+2\}$$
$$B^0_{i,i+1} = \{\overline{a_{i,i+1}}, x_{i+1}\}, \qquad B^1_{i,i+1} = \{\overline{a_{i,i+1}}, \overline{x_{i+1}}\}$$

To see why these formulas are false, note that satisfying all the δ clauses requires all $a_{i,j}$, $1 \le i < j \le n$ to be set to 0. (Consider resolution on the x variables in these clauses.) At this setting, all the ψ clauses give back LQParity.

We can adapt the LD-Q-Res refutation of QParity to an M-Res refutation of MParity. We describe below exactly how this is achieved. The family MParity consists of two sets of clauses: ψ and δ. The proof has two stages. In the first stage, the $a_{i,j}$ variables are eliminated from the clauses in ψ using the clauses in δ. The role of these $a_{i,j}$ variables and the clauses of δ is to build up complex merge maps meeting the isomorphism condition, so that subsequent resolution steps are enabled. In the second phase, the LD-Q-Res refutation of QParity is mimicked, eliminating the t variables.

(In the proofs below, notice that each line contains a single merge map. This is done because the merge maps for z_1 and z_2 in every line are same. So, we write them only once to save space.)

For $i \in [n+1]$, let g_i be the parity function $\oplus_{j\ge i} x_j$, and let h_i denote its complement; thus h_i is even-parity on the variables $x_i, \dots, x_n$. (The parity of an empty set of variables is 0; thus $g_{n+1}=0$ and $h_{n+1}=1$.) Let M_i^1 (resp. M_i^0) be the smallest merge map which queries variables in the order $x_i,\ldots,x_n$ and computes the function g_i (resp. h_i). Note that both these merge maps have $2(n-i)+1$ internal nodes and two leaf nodes labelled 0 and 1.

The main idea is to replace the constant merge maps in the axioms of $A^0_{i,C}$ and $A^1_{i,C}$ by the merge maps M^0_{i+1} and M^1_{i+1} — the set of clause, merge map pairs so generated will be denoted by $\widetilde{\psi_i}$ (and are defined below). These merge maps will allow us to pass the isomorphism checks later in the proofs.

For $i\in[n]$, let $\widetilde{\psi_i}$ be the following sets of clause, merge map pairs:

$$\widetilde{\psi_i} = \left\{\left(C, M^b_{i+1}\right) \mid C\in \text{even-parity}(t_{i-1},x_i,t_i), b\in\{0,1\}\right\} \qquad \forall i\in[2,n]$$
$$\widetilde{\psi_1} = \left\{\left(C, M^b_{2}\right) \mid C\in \text{even-parity}(x_1,t_1), b\in\{0,1\}\right\}$$

Lemma 3.11. *Let the quantifier prefix be as in the definition of MParity. Then, for all $i\in[n]$, $\psi_i\wedge\delta_i\vdash_{\text{M-Res}}\widetilde{\psi_i}$. Moreover the size of these derivations is polynomial in n.*

Proof. At $i=n$, $\widetilde{\psi_n}$ is the same as ψ_n so there is nothing to prove.

Consider now an $i\in[n-1]$. For each $b\in\{0,1\}$ and each $C\in$ even-parity (t_{i-1},x_i,t_i) (if $i=1$, omit t_{i-1}), the clause $A^b_{i,C}\in\psi_i$ yields the line $(C\cup\{a_{i,n}\},M^{1-b}_{n+1})$. Resolving each of these with each of $B^d_{i,n}$ for $d\in\{0,1\}$, we obtain four clauses that can be resolved in two pairs to produce the lines $(C\cup\{a_{i,n-1}\},M^b_n)$. (See the derivation at the end of this proof.) Repeating this process successively for $j=n,n-1,\ldots,i+2$, using the clause pairs $B^d_{i,j}$ with the previously derived clauses, we can obtain each $(C\cup\{a_{i,j}\},M^b_{j+1})$. In each stage, the index j of the variable $a_{i,j}$ present in the clause decreases, while the merge map accounts for one more variable. Finally, when we use the clause pairs $B^d_{i,i+1}$, the $a_{i,i+1}$ variable is eliminated, variables $x_{i+1},\ldots,x_n$ are accounted for in the merge map, and we obtain the lines (C,M^b_{i+1}), corresponding to the clauses in $\widetilde{\psi_i}$.

The derivation at one stage is as shown below.

$$\dfrac{\dfrac{\left(C\cup\{a_{i,j}\},M^1_{j+1}\right) \qquad \overbrace{\left(\{\overline{a_{i,j}},x_j,a_{i,j-1}\},*\right)}^{B^0_{i,j}}}{\left(C\cup\{x_j,a_{i,j-1}\},M^1_{j+1}\right)} \qquad \dfrac{\left(C\cup\{a_{i,j}\},M^0_{j+1}\right) \qquad \overbrace{\left(\{\overline{a_{i,j}},\overline{x_j},a_{i,j-1}\},*\right)}^{B^1_{i,j}}}{\left(C\cup\{\overline{x_j},a_{i,j-1}\},M^0_{j+1}\right)}}{\left(C\cup\{a_{i,j-1}\},M^1_j\right)}$$

$$\dfrac{\dfrac{\left(C\cup\{a_{i,j}\},M^1_{j+1}\right) \qquad \overbrace{\left(\{\overline{a_{i,j}},\overline{x_j},a_{i,j-1}\},*\right)}^{B^1_{i,j}}}{\left(C\cup\{\overline{x_j},a_{i,j-1}\},M^1_{j+1}\right)} \qquad \dfrac{\left(C\cup\{a_{i,j}\},M^0_{j+1}\right) \qquad \overbrace{\left(\{\overline{a_{i,j}},x_j,a_{i,j-1}\},*\right)}^{B^0_{i,j}}}{\left(C\cup\{x_j,a_{i,j-1}\},M^0_{j+1}\right)}}{\left(C\cup\{a_{i,j-1}\},M^0_j\right)}$$

□

In the second phase, we successively eliminate the t variables in stages.

Lemma 3.12. *Let the quantifier prefix be as in the definition of MParity. Then the following derivations can be done in M-Res in size polynomial in n:*

(1) *For $i=n,n-1,\ldots,2$,* $\left(\{t_i\},M^1_{i+1}\right),\left(\{\overline{t_i}\},M^0_{i+1}\right),\widetilde{\psi_i}\vdash\left(\{t_{i-1}\},M^1_i\right),\left(\{\overline{t_{i-1}}\},M^0_i\right)$.

(2) $\left(\{t_1\},M^1_2\right),\left(\{\overline{t_1}\},M^0_2\right),\widetilde{\psi_1}\vdash\left(\Box,M^1_1\right)$.

Proof. For $i \geq 2$, the derivation is as follows:

$$\dfrac{\dfrac{\left(\{t_{i-1}, x_i, \overline{t_i}\}, M^1_{i+1}\right) \qquad \left(\{t_i\}, M^1_{i+1}\right)}{\left(\{t_{i-1}, x_i\}, M^1_{i+1}\right)} \qquad \dfrac{\left(\{t_{i-1}, \overline{x_i}, t_i\}, M^0_{i+1}\right) \qquad \left(\{\overline{t_i}\}, M^0_{i+1}\right)}{\left(\{t_{i-1}, \overline{x_i}\}, M^0_{i+1}\right)}}{\left(\{t_{i-1}\}, M^1_i\right)}$$

$$\dfrac{\dfrac{\left(\{\overline{t_{i-1}}, \overline{x_i}, \overline{t_i}\}, M^1_{i+1}\right) \qquad \left(\{t_i\}, M^1_{i+1}\right)}{\left(\{\overline{t_{i-1}}, \overline{x_i}\}, M^1_{i+1}\right)} \qquad \dfrac{\left(\{\overline{t_{i-1}}, x_i, t_i\}, M^0_{i+1}\right) \qquad \left(\{\overline{t_i}\}, M^0_{i+1}\right)}{\left(\{\overline{t_{i-1}}, x_i\}, M^0_{i+1}\right)}}{\left(\{\overline{t_{i-1}}\}, M^0_i\right)}$$

The derivation at the last stage is as follows:

$$\dfrac{\dfrac{\left(\{x_1, \overline{t_1}\}, M^1_2\right) \qquad \left(\{t_1\}, M^1_2\right)}{\left(\{x_1\}, M^1_2\right)} \qquad \dfrac{\left(\{\overline{x_1}, t_1\}, M^0_2\right) \qquad \left(\{\overline{t_1}\}, M^0_2\right)}{\left(\{\overline{x_1}\}, M^0_2\right)}}{\left(\Box, M^1_1\right)}$$

□

We can now conclude the following:

Lemma 3.13. *MParity has polynomial-size M-Res refutations.*

Proof. We first use Lemma 3.11 to derive all the $\widetilde{\psi_i}$. Next, we start with $\left(\{t_n\}, M^1_{n+1}\right)$ and $\left(\{\overline{t_n}\}, M^0_{n+1}\right)$, the lines corresponding to the clauses in ψ_{n+1}. From these lines and $\widetilde{\psi_n}$, we derive $\left(\{t_{n-1}\}, M^1_n\right)$ and $\left(\{\overline{t_{n-1}}\}, M^0_n\right)$, using Lemma 3.12. We continue in this manner deriving $\left(\{t_i\}, M^1_{i+1}\right)$ and $\left(\{\overline{t_i}\}, M^0_{i+1}\right)$ for $i = n-2, n-3, \ldots, 1$. From the lines $\widetilde{\psi_1}$, $\left(\{t_1\}, M^1_2\right)$ and $\left(\{\overline{t_1}\}, M^0_2\right)$, we derive $\left(\Box, M^1_1\right)$ using Lemma 3.12. □

Theorem 3.14. *LD-Q-Res does not p-simulate M-Res; and LQU-Res and LQU$^+$-Res are incomparable with M-Res.*

Proof. We showed in Lemma 3.13 that the MParity formulas have polynomial-size M-Res refutations. We will now show that MParity requires exponential-size LQU$^+$-Res refutations. We first note that QUParity requires exponential-size LQU$^+$-Res refutations [BCJ19]. We further note that LQU$^+$-Res is closed under restrictions (Proposition 2 in [BWJ14]). Since restricting the MParity formulas by setting $a_{i,j} = 0$, for all $i, j \in [n]$, gives the QUParity formulas, we conclude that MParity requires exponential-size LQU$^+$-Res refutations. Therefore LQU$^+$-Res does not simulate M-Res. Since LQU$^+$-Res p-simulates LD-Q-Res and LQU-Res, these two systems also do not simulate M-Res.

In [BBM$^+$24] it is shown that M-Res does not simulate QU-Res. (The separating formula is in fact KBKF-lq.) Since LQU-Res and LQU$^+$-Res p-simulate QU-Res [BWJ14] and the simulation order is transitive, it follows that M-Res does not simulate LQU-Res and LQU$^+$-Res.

Hence LQU-Res and LQU$^+$-Res are incomparable with M-Res. □

Remark 3.15. In these proofs, note that the hardness for LQU$^+$-Res and IRM was proven using restrictions. But the same did not apply to M-Res — a restricted formula being hard for M-Res does not mean that the original formula is also hard. This means that M-Res is not closed under restrictions, and is hence unnatural.

Remark 3.16. Another observation is that the clauses of the KBKF-lq-weak formula family are weakenings of the clauses of KBKF-lq. Since KBKF-lq requires exponential-size M-Res refutations but KBKF-lq-weak has polynomial-size M-Res refutations, we conclude that weakening adds power to M-Res.

4. Role of weakenings, and unnaturalness

4.1. **Weakenings.** Let $(C, \{M^u \mid u \in U\})$ be a line of an M-Res refutation. Then it can be weakened in two different ways [BBM21]:

- Existential clause weakening: $C \vee x$ can be derived from C, provided it does not contain the literal $\overline{x}$. The merge maps remain the same. Similarly, $C \vee \overline{x}$ can be derived if $x \notin C$.
- Strategy weakening: A trivial merge map $(*)$ can be replaced by a constant merge map (0 or 1). The existential clause remains the same.

Adding these weakenings to M-Res gives the following three proof systems:

- M-Res with existential clause weakening (M-ResW$_\exists$),
- M-Res with strategy weakening (M-ResW$_\forall$), and
- M-Res with both existential clause and strategy weakening (M-ResW$_{\exists\forall}$).

Proposition 4.1 [BBM21]**.** *All the proof systems M-ResW$_\exists$, M-ResW$_\forall$, and M-ResW$_{\exists\forall}$ are complete and sound.*

Proof. By definition, M-ResW$_{\exists\forall}$ p-simulates M-ResW$_\exists$ and M-ResW$_\forall$, both of which p-simulate M-Res. Completeness thus follows from the completeness of M-Res. It suffices to show that M-ResW$_{\exists\forall}$ is sound. This follows from Lemma 31 in [BBM21], where a generalized version of this system, allowing simultaneous existential clause weakening and universal strategy weakening in a single rule application, is shown to be sound for the Herbrand-form Dependency QBFs (H-form DQBFs, discussed in Section 6 of [BBM21]). Since QBFs are a special case of H-form DQBFs, soundness of M-ResW$_{\exists\forall}$ for QBFs follows. □

In the remainder of this subsection, we will study the relation among these systems.
First, we note that existential clause weakening adds exponential power.

Theorem 4.2. *M-ResW$_\exists$ is strictly stronger than M-Res.*

Proof. Since M-ResW$_\exists$ is a generalization of M-Res, M-ResW$_\exists$ p-simulates M-Res.

The KBKF-lq formulas can be transformed into the KBKF-lq-weak formulas in M-ResW$_\exists$ using a linear number of applications of the existential weakening rule. The transformed KBKF-lq-weak formulas have polynomial-size M-Res (and hence M-ResW$_\exists$) refutations, Lemma 3.6. Thus the KBKF-lq formulas have polynomial-size M-ResW$_\exists$ refutations. Since the KBKF-lq formulas require exponential-size M-Res refutations [BBM$^+$24], we get the desired separation. □

Next we observe that a lower bound for M-Res, Theorem 3.17 from [BBM$^+$24], can be lifted to M-ResW$_\forall$.

Lemma 4.3. *KBKF-lq requires exponential-size refutations in M-ResW$_\forall$.*

Proof. We observe that the M-Res lower bound for KBKF-lq in [BBM+24] works with a minor modification. In [BBM+24, Lemma 3.19], item 3 says that $M^{x_i} = *$. However a weaker condition $M^{x_i} \in \{*, 0, 1\}$ is sufficient for the lower bound. With this modification, we observe that the remaining argument carries over, and the lower bound also works for M-ResW$_\forall$.

For the convenience of the reader, we reproduce this proof from [BBM+24] in Appendix A with the modification incorporated. □

Theorem 4.4. *M-ResW$_\forall$ does not simulate M-ResW$_\exists$; and M-ResW$_{\exists\forall}$ is strictly stronger than M-ResW$_\forall$.*

Proof. We showed that the KBKF-lq formulas require refutations of exponential size in M-ResW$_\forall$ (Lemma 4.3) but have polynomial-size refutations in M-ResW$_\exists$ and M-ResW$_{\exists\forall}$ (proof of Theorem 4.2). Therefore M-ResW$_\forall$ does not simulate M-ResW$_\exists$ and M-ResW$_{\exists\forall}$. Since M-ResW$_{\exists\forall}$ p-simulates M-ResW$_\forall$, M-ResW$_{\exists\forall}$ is strictly stronger than M-ResW$_\forall$. □

The next logical question is whether strategy weakening adds power to M-Res. We do not know the answer. However, we can answer this for the regular versions of these systems.

Definition 4.5. A refutation is called regular if each variable is resolved at most once along every leaf-to-root path. A proof system is called regular if it only allows regular refutations.

Theorem 4.6. *Regular M-ResW$_\forall$ is strictly stronger than regular M-Res.*

To prove this theorem, we will use a variant of the Squared-Equality (Eq^2) formula family, called Squared-Equality-with-Holes (H-Eq^2). Squared-Equality, defined in [BBM21], is a two-dimensional version of the Equality (Eq) formula family [BBH19], and has short regular tree-like M-Res refutations. It was used to show that the systems Q-Res, QU-Res, reductionless LD-Q-Res, $\forall$Exp + Res, IR and CP + $\forall$red do not p-simulate M-Res. We recall the definitions of Equality and Squared-Equality formulas below:

Definition 4.7. *Equality* (Eq(n)) is the following QBF formula family:

$$\underset{i\in[n]}{\exists} x_i, \underset{i\in[n]}{\forall} u_i, \underset{i\in[n]}{\exists} t_i.\ \left(\bigwedge_{i\in[n]} A_i\right) \wedge B$$

where

- $B = \vee_{i\in[n]}\overline{t_i}$,
- For $i \in [n]$, A_i contains the following two clauses:

$$x_i \vee u_i \vee t_i, \qquad\qquad \overline{x_i} \vee \overline{u_i} \vee t_i.$$

To see that the Equality formulas are false, observe that $u_i = x_i$ for each $i \in [n]$ is a winning strategy for the universal player. If the universal player plays according to this strategy, then after x and u are set, for each $i \in [n]$, one of the clauses in A_i becomes t_i. These reduced clauses, along with B, cannot all be satisfied by the existential plater.

Definition 4.8. *Squared-Equality* (Eq^2(n)) is the following QBF family:

$$\underset{i\in[n]}{\exists} x_i, \underset{j\in[n]}{\exists} y_j, \underset{i\in[n]}{\forall} u_i, \underset{j\in[n]}{\forall} v_j, \underset{i,j\in[n]}{\exists} t_{i,j}.\ \left(\bigwedge_{i,j\in[n]} A_{i,j}\right) \wedge B$$

where

- $B = \vee_{i,j\in[n]}\overline{t_{i,j}}$,

- For $i, j \in [n]$, $A_{i,j}$ contains the following four clauses:

$$x_i \bigvee y_j \bigvee u_i \bigvee v_j \bigvee t_{i,j}, \qquad x_i \bigvee \overline{y_j} \bigvee u_i \bigvee \overline{v_j} \bigvee t_{i,j},$$
$$\overline{x_i} \bigvee y_j \bigvee \overline{u_i} \bigvee v_j \bigvee t_{i,j}, \qquad \overline{x_i} \bigvee \overline{y_j} \bigvee \overline{u_i} \bigvee \overline{v_j} \bigvee t_{i,j}$$

For the Squared Equality formulas, the following strategy for the universal player is a winning strategy: for all $i \in [n]$, set $u_i = x_i$; and for all $j \in [n]$, set $v_j = y_j$.

We observe that the short M-Res refutation of $\mathrm{Eq}^2(n)$ crucially uses the isomorphism of merge maps. For each $i, j \in [n]$, the four clauses in $A_{i,j}$ are resolved to derive the line $(t_{i,j}, \{u_i = x_i, v_j = y_j\})$. These lines are then resolved with the line $\left(\vee_{i,j\in[n]}\overline{t_{i,j}}, \{*, \cdots, *\}\right)$ to derive the line $(\Box, \{u_i = x_i, v_j = y_j \mid \forall i, j \in [n]\})$. The resolutions over the $t_{i,j}$ variables are possible only because the merge maps are isomorphic. If we modify the clauses of Eq^2 such that the merge maps produced from different $A_{i,j}$ are non-isomorphic, then the refutation described above is forbidden. This is the motivation behind the Squared-Equality-with-Holes ($\mathrm{H\text{-}Eq}^2$) formula family defined below. It is constructed from Eq^2 by removing some of the universal variables from the $A_{i,j}$ clauses. The resulting QBF family remains false but different $A_{i,j}$ lead to different merge maps. We believe that this QBF family is hard for M-Res, but we have only been able to prove the hardness for regular M-Res, and hence the separation is between the regular versions of M-Res and M-ResW$_\forall$.

The variant identifies regions in the $[n] \times [n]$ grid, and changes the clause sets $A_{i,j}$ depending on the region that (i, j) belongs to. We can use any partition of $[n] \times [n]$ into two regions R_0, R_1 such that each region has at least one position in each row and at least one position in each column; call such a partition a *covering partition*. One possible choice for R_0 and R_1 is the following: $R_0 = ([1, n/2] \times [1, n/2]) \cup ([n/2 + 1, n] \times [n/2 + 1, n])$ and $R_1 = ([1, n/2] \times [n/2 + 1, n]) \cup ([n/2 + 1, n] \times [1, n/2])$. We will call R_0 and R_1 the two regions of the matrix.

Definition 4.9. Let R_0, R_1 be a covering partition of $[n] \times [n]$.

Squared-Equality-with-Holes ($\mathrm{H\text{-}Eq}^2(n)(R_0, R_1)$) is the following QBF family:

$$\underset{i\in[n]}{\exists} x_i, y_i, \underset{j\in[n]}{\forall} u_j, v_j, \underset{i,j\in[n]}{\exists} t_{i,j}. \left(\bigwedge_{i,j\in[n]} A_{i,j}\right) \wedge B$$

where

- $B = \vee_{i,j\in[n]}\overline{t_{i,j}}$,
- For $(i, j) \in R_0$, $A_{i,j}$ contains the following four clauses:

$$x_i \vee y_j \vee u_i \vee v_j \vee t_{i,j}, \qquad x_i \vee \overline{y_j} \vee u_i \vee t_{i,j},$$
$$\overline{x_i} \vee y_j \vee v_j \vee t_{i,j}, \qquad \overline{x_i} \vee \overline{y_j} \vee t_{i,j}$$

- For $(i, j) \in R_1$, $A_{i,j}$ contains the following four clauses:

$$x_i \vee y_j \vee t_{i,j}, \qquad x_i \vee \overline{y_j} \vee \overline{v_j} \vee t_{i,j},$$
$$\overline{x_i} \vee y_j \vee \overline{u_i} \vee t_{i,j}, \qquad \overline{x_i} \vee \overline{y_j} \vee \overline{u_i} \vee \overline{v_j} \vee t_{i,j}$$

(We do not always specify the regions explicitly but merely say $\mathrm{H\text{-}Eq}^2$.)

Lemma 4.10. *$H\text{-}Eq^2(n)$ requires exponential-size refutations in regular M-Res.*

Before proving this, we show how to obtain Theorem 4.6.

Proof of Theorem 4.6. Since regular M-ResW$_\forall$ is a generalization of regular M-Res, it p-simulates regular M-Res.

Using strategy weakening, we can get Eq^2 from H-Eq^2 in a linear number of steps. Since Eq^2 has polynomial-size refutations in regular M-Res, we get polynomial-size refutations for H-Eq^2 in regular M-ResW$_\forall$. On the other hand, Lemma 4.10 gives an exponential lower bound for H-Eq^2 in regular M-Res. Therefore regular M-ResW$_\forall$ is strictly stronger than regular M-Res. □

It remains to prove Lemma 4.10. This is a fairly involved proof, but in broad outline and in many details it is similar to the lower bound for Eq^2 in reductionless LD-Q-Res [BBM21].

The size bound is trivially true for $n = 1$, so we assume that $n > 1$. Let Π be a Regular M-Res refutation of H-Eq$^2(n)$. Since a tautological clause cannot occur in a regular M-Res refutation, we assume that Π does not have a line whose clause part is tautological.

Let us first fix some notation. Let $X = \{x_1, \dots, x_n\}$, $Y = \{y_1, \dots, y_n\}$, $U = \{u_1, \dots, u_n\}$, $V = \{v_1, \dots, v_n\}$, and $T = \{t_{i,j} \mid i, j \in [n]\}$. For lines L_1, L_2, etc., the respective clauses and merge maps will be denoted by C_1, C_2 and M_1, M_2 etc. For a line L in Π, Π_L denotes the sub-derivation of Π ending in L. Viewing Π as a directed acyclic graph, we can talk of leaves and paths in Π. For a line L of Π, let $\textsc{Uci}(L) = \{(i, j) \mid A_{i,j} \cap \text{leaves}(\Pi_L) \neq \emptyset\}$. (The abbreviation Uci stands for UsedConstraintsIndex.)

We first show some structural properties about Π. The first property excludes using many axioms in certain derivations.

Lemma 4.11. *For line $L = (C, M)$ of Π, and $i, j \in [n]$, if $t_{i,j} \in C$, then $\textsc{Uci}(L) = \{(i, j)\}$.*

Proof. Since the literal $t_{i,j}$ only occurs in clauses in $A_{i,j}$, so leaves$(L) \cap A_{i,j} \neq \emptyset$, hence $\textsc{Uci}(L) \supseteq \{(i, j)\}$.

Now suppose $|\textsc{Uci}(L)| > 1$. Let (i', j') be an arbitrary element of $\textsc{Uci}(L)$ distinct from (i, j). Pick a leaf of Π_L using a clause in $A_{i',j'}$, and let p be a path from this leaf to L and then to the final line of Π. Both $t_{i,j}$ and $t_{i',j'}$ are necessarily used as pivots on this path. Assume that $t_{i,j}$ is used as a pivot later (closer to the final line) than $t_{i',j'}$; the other case is symmetric. Let $L_c = \text{res}(L_a, L_b, t_{i',j'})$ and $L_f = \text{res}(L_d, L_e, t_{i,j})$ respectively be the positions where $t_{i',j'}$ and $t_{i,j}$ are used as resolution pivots on this path (here L_a and L_d are the lines of path p, hence $t_{i',j'} \in C_a$ and $t_{i,j} \in C_d$). Then C_b has the negated literal $\overline{t_{i',j'}}$; hence $B \in \text{leaves}(L_b)$. Since $\overline{t_{i,j}} \in B$ but $\overline{t_{i,j}} \notin L_d$, $t_{i,j}$ is used as a resolution pivot in the derivation Π_{L_d}. This contradicts the fact that Π is regular. □

The next property is the heart of the proof, and shows that paths with B at the leaf must have a suitable wide clause.

Lemma 4.12. *On every path from $\left(\vee_{i,j\in[n]}\overline{t_{i,j}}, \{*, \cdots, *\}\right)$ (the line for axiom clause B) to the final line, there exists a line $L = (C, M)$ such that either $X \subseteq var(C)$ or $Y \subseteq var(C)$.*

Proof. With each line $L_l = (C_l, M_l)$ in Π, we associate an $n \times n$ matrix N_l in which $N_l[i, j] = 1$ if $\overline{t_{i,j}} \in C_l$ and $N_l[i, j] = 0$ otherwise.

Let $p = L_1, \dots, L_k$ be a path from $\left(\vee_{i,j\in[n]}\overline{t_{i,j}}, \{*, \cdots, *\}\right)$ to the final line in Π. Since Π is regular, each $\overline{t_{i,j}}$ is resolved away exactly once, so no clause on p has any positive $t_{i,j}$ literal. Let l be the least integer such that N_l has a 0 in each row or a 0 in each column. Note that $l \geq 2$ since N_1 has no zeros. Consider the case that N_l has a 0 in each row; the argument for the other case is identical. We will show in this case that $X \subseteq \text{var}(C_l)$. We will use the following claim:

Claim 4.13. In each row of N_l, there is a 0 and a 1 such that the 0 and 1 are in different regions (i.e. one is in R_0 and the other in R_1).

Proof. We already know that N_l has a 0 in each row. We will first prove that N_l also has a 1 in each row. Aiming for contradiction, suppose that N_l has a full 0 row r. Since $l \geq 2$, N_{l-1} exists. Note that, by definition of resolution, there can be at most one element that changes from 1 in N_{l-1} to 0 in N_l. Since N_{l-1} does not have a 0 in every column, it does not contain a full 0 row. Hence, the unique element that changed from 1 in N_{l-1} to 0 in N_l must be in row r. Thus all other rows of N_{l-1} already contain the one 0 of that row in N_l. Since $n \geq 2$, N_{l-1} also has at least one 0 in row r; thus N_{l-1} has a 0 in each row, contradicting the minimality of l.

Since R_0 and R_1 form a covering partition, it cannot be the case that all the 0s and 1s of any row are in the same region R_b; that would imply that R_{1-b} does not cover the row. (Proof of Claim 4.13 complete.) □

We want to prove that $X \subseteq var(C_l)$. Suppose, to the contrary, there exists $i \in [n]$ such that $x_i \notin \text{var}(C_l)$. We know that there exist $j_1, j_2 \in [n]$ such that $N_l[i, j_1] = 0$ and $N_l[i, j_2] = 1$; and either $(i, j_1) \in R_0$ and $(i, j_2) \in R_1$, or $(i, j_1) \in R_1$ and $(i, j_2) \in R_0$. Without loss of generality, we may assume that $(i, j_1) \in R_0$ and $(i, j_2) \in R_1$.

We know that on path p, there is a resolution with pivot t_{i,j_1} before L_l and a resolution with pivot t_{i,j_2} after L_l. Let the former resolution be $L_c = \text{res}(L_a, L_b, t_{i,j_1})$ where L_b is on path p, and let the latter resolution be $L_f = \text{res}(L_d, L_e, t_{i,j_2})$ where L_e is on path p. Since Π is a regular refutation, $t_{i,j_1} \in C_a$, $\overline{t_{i,j_1}} \in C_b$ and $t_{i,j_2} \in C_d$, $\overline{t_{i,j_2}} \in C_e$. Thus along path p these lines appear in the relative order $B, L_b, L_c, L_l, L_e, L_f, (\Box, \{u_i = x_i, v_j = y_j \mid \forall i, j \in [n]\})$.

Claim 4.14. $\overline{x_i} \in C_c$.

Proof. By Lemma 4.11, $\text{UCI}(L_d) = \{(i, j_2)\}$, or equivalently leaves$(L_d) \subseteq A_{i,j_2}$. Since $(i, j_2) \in R_1$, no clause in A_{i,j_2} has literal u_i. Hence $M_d^{u_i} \in \{*, 1\}$. Furthermore, if $M_d^{u_i} = *$, then $x_i \in C_d$. Since the pivot for resolving L_d and L_e is t_{i,j_2}, this would imply that $x_i \in C_f$.

By a similar argument, we can conclude that (i) leaves$(L_a) \subseteq A_{i,j_1}$, (ii) $M_a^{u_i} \in \{*, 0\}$, and (iii) if $M_a^{u_i} = *$, then $\overline{x_i} \in C_c$.

If $M_d^{u_i} = *$ and $M_a^{u_i} = *$, then $x_i \in C_f$ and $\overline{x_i} \in C_c$. So x_i must be used twice as a pivot on path p, contradicting regularity.

If $M_d^{u_i} = *$ and $M_a^{u_i} = 0$, then $x_i \in C_f$ and Π_{L_a} uses some clause containing x_i to make the merge map for u_i non-trivial. Thus $x_i \in \Pi_{L_a}$, $x_i \notin L_l$ by assumption, $x_i \in L_f$. Hence x_i is used twice as pivot, contradicting regularity.

Hence $M_d^{u_i} = 1$. Since the resolution at line L_f is not blocked, $M_e^{u_i} \in \{*, 1\}$. But L_e is derived after, and using, L_a. Since merge maps do not get simpler along a path, $M_a^{u_i} \in \{*, 1\}$. It follows that $M_a^{u_i} = *$. Hence $\overline{x_i} \in C_c$. (Proof of Claim 4.14 complete.) □

Since $\overline{x_i} \notin C_l$, x_i has been used as a resolution pivot between L_c and L_l on path p. Let $L_w = \text{res}(L_u, L_v, x_i)$ be the position on path p where x_i is used as pivot (since the refutation is regular, such a position is unique). Let L_v be the line on path p. By regularity of the refutation, $x_i \in L_u$ and $\overline{x_i} \in L_v$.

As observed at the outset, L_w is on path p and so does not contain a positive t literal. Since C_w is obtained via pivot x_i, this implies that C_u also does not contain a positive t literal. Since all axioms contain at least one t variable but only B contains negated t literals, so $B \in \text{leaves}(L_u)$.

Let q be a path that starts from a leaf using B, passes through L_u to L_w, and then continues along path p to the final clause. Since the refutation is regular, $N_v = N_u = N_w$. Hence $N_v[i, j_1] = 0$ i.e. $\overline{t_{i,j_1}} \notin C_v$. This implies that t_{i,j_1} is used as resolution pivot before L_v on path q.

We already know that t_{i,j_2} is used as a pivot after line L_l on path p, and hence on path q. Arguing analogous to Claim 4.14 for path p but with respect to path q, we observe that $\overline{x_i}$ belongs to at least one leaf of L_u. Since $x_i \in C_u$ and since the refutation is regular, x_i is not used as a resolution pivot before C_u on path q. This implies that $\overline{x_u} \in C_u$. We already know that $x_i \in C_u$, since it contributed the pivot at L_w. This means that C_u is a tautological clause, a contradiction. □

We can finally prove Lemma 4.10. This part is identical to the corresponding part of the proof of Theorem 28 in [BBM21]; we include it here for completeness.

Proof of Lemma 4.10. For each $a = (a_1, \dots, a_n) \in \{0,1\}^n$, consider the assignment σ_a to the existential variables which sets $x_i = y_i = a_i$ for all $i \in [n]$, and $t_{i,j} = 1$ for all $i, j \in [n]$. Call such an assignment a symmetric assignment. Given a symmetric assignment σ_a, walk from the final line of Π towards the leaves maintaining the following invariant: for each line $L = (C, \{M^u \mid u \in U \cup V\})$, σ_a falsifies C. Let p_a be the path followed. By Lemma 4.12, this path will contain a line $L = (C, \{M^u \mid u \in U \cup V\})$ such that either $X \subseteq var(C)$ or $Y \subseteq var(C)$. Let us define a function f from symmetric assignments to the lines of Π as follows: $f(a) = (C, \{M^u \mid u \in U \cup V\})$ is the last line (i.e. nearest to the leaves) on p_a such that either $X \subseteq var(C)$ or $Y \subseteq var(C)$. Note that, for any line L of Π, there can be at most one symmetric assignment a such that $f(a) = L$. This means that there are at least 2^n lines in Π. This gives the desired lower bound. □

4.2. **Simulation by eFrege + ∀red.** It was recently shown that eFrege + ∀red p-simulates all known resolution-based QBF proof systems; in particular, it p-simulates M-Res [CS24]. We observe that this p-simulation can be extended in a straightforward manner to handle both the weakenings in M-Res. Hence we obtain a p-simulation of M-ResW$_\exists$, M-ResW$_\forall$ and M-ResW$_{\exists\forall}$ by eFrege + ∀red.

Theorem 4.15. *eFrege + ∀red strictly p-simulates M-ResW$_\exists$, M-ResW$_\forall$ and M-ResW$_{\exists\forall}$.*

Proof. The separation follows from the separation of the propositional proof systems resolution and eFrege [Urq87]. We prove the p-simulation below.

It suffices to prove that eFrege + ∀red p-simulates M-ResW$_{\exists\forall}$. The proof is essentially same as that of the p-simulation of M-Res in [CS24], but with two additional cases for the two weakenings. So, we will briefly describe that proof and then describe the required modifications.

Let Π be an M-ResW$_{\exists\forall}$ refutation Π of a QBF Φ. The last line of this refutation gives a winning strategy for the universal player; let us call this strategy S. We will first prove that there is a short eFrege derivation $\Phi \vdash \neg S$. Then, as mentioned in [CS24], the technique of [BBCP20, Che21] can be used to derive the empty clause from $\neg S$ using universal reduction.

We will now describe an eFrege derivation $\Phi \vdash \neg S$. Let $L_i = (C_i, \{M_i^u \mid u \in U\})$ be the i$^{\text{th}}$ line of Π. We create new extension variables: $s_{i,j}^u$ is the variable for the j$^{\text{th}}$ node of M_i^u. If node j is a leaf of M_i^u labelled by constant c, then $s_{i,j}^u$ is defined to be c. Otherwise, if

$M_i^u(j) = (x, a, b)$, then $s_{i,j}^u$ is defined as $s_{i,j}^u \triangleq \left(x \wedge s_{i,a}^u\right) \vee \left(\overline{x} \wedge s_{i,b}^u\right)$. The extension variables for u will be to its left in the quantifier prefix.

We will prove that for each line L_i of Π, we can derive the formula $F_i \triangleq \wedge_{u \in U_i}(u \leftrightarrow s_{i,r(u,i)}^u) \rightarrow C_i$; where $r(u,i)$ is the index of the root of merge map M_i^u, and U_i is the set of universal variables for which M_i^u is non-trivial.

Our proof will proceed by induction on the lines of the refutation.

The base case is when L_i is an axiom; and the inductive step will have three cases depending on which rule is used to derive L_i: (i) resolution, (ii) existential clause weakening, or (iii) strategy weakening. The proof for the base case and the resolution step case is as given in [CS24]. We give proofs for the other two cases below:

- Existential clause weakening: Let line $L_b = (C_b, \{M_b^u \mid u \in U\})$ be derived from line $L_a = (C_a, \{M_a^u \mid u \in U\})$ using existential clause weakening. Then $C_b = C_a \vee x$ for some existential literal x such that $\overline{x} \notin C_a$, and $M_b^u = M_a^u$ for all $u \in U$. By the induction hypothesis, we have derived the formula $F_a \triangleq \wedge_{u \in U_a}(u \leftrightarrow s_{a,r(u,a)}^u) \rightarrow C_a$. We have to derive the formula $F_b \triangleq \wedge_{u \in U_b}(u \leftrightarrow s_{b,r(u,b)}^u) \rightarrow C_b = \wedge_{u \in U_b}(u \leftrightarrow s_{b,r(u,b)}^u) \rightarrow C_a \vee x$. Since $M_b^u = M_a^u$ for each u, there is a short eFrege + $\forall$red derivation of the formula $s_{a,j}^u \leftrightarrow s_{b,j}^u$ for each $u \in U_i$, and each node j of M_a^u. This allows us to replace variable $s_{a,j}^u$ by $s_{b,j}^u$ in F_a. As a result, we get the formula $F_b' \triangleq \wedge_{u \in U_b}(u \leftrightarrow s_{b,r(u,b)}^u) \rightarrow C_a$. Now, using an inference of the form $p \rightarrow q \vdash p \rightarrow q \vee r$, we obtain the formula F_b.
- Strategy weakening: Let line $L_b = (C_b, \{M_b^u \mid u \in U\})$ be derived from line $L_a = (C_a, \{M_a^u \mid u \in U\})$ using strategy weakening for a variable v. Then $C_b = C_a$, $M_b^u = M_a^u$ for all $u \in U \setminus \{v\}$, and $M_a^v = *$, M_b^v is a constant, say d. Similar to the above case, we start with the inductively obtained F_a and replace each $s_{a,j}^u$ with $s_{b,j}^u$ to obtain a formula $F_b' \triangleq \wedge_{u \in U_b \setminus \{v\}}(u \leftrightarrow s_{b,r(u,b)}^u) \rightarrow C_b$. With a final inference of the form $p \rightarrow q \vdash p \wedge r \rightarrow q$, we can then add $(v \leftrightarrow s_{b,r(v,b)}^v)$ to the conjunction to obtain F_b. □

4.3. **Unnaturalness.** In this section, we observe that M-Res and M-ResW$_\forall$ are unnatural proof systems, i.e. they are not closed under restrictions.

Theorem 4.16. *M-Res and M-ResW$_\forall$ are unnatural proof systems.*

Proof. The KBKF-lq-split formula family has polynomial-size refutations in M-Res (and M-ResW$_\forall$), as seen in Lemma 3.7. The restriction of this family obtained by setting $t = 0$ is exactly the KBKF-lq formula family, which, as shown in Lemma 4.3, is exponentially hard for M-ResW$_\forall$ and hence also for M-Res. □

5. Conclusion and future work

M-Res was introduced in [BBM21] to overcome the weakness of LD-Q-Res. It was shown that M-Res has advantages over many proof systems, but the advantage over LD-Q-Res was not demonstrated. In this paper, we have filled this gap. We have shown that M-Res has advantages over not only LD-Q-Res, but also over more powerful systems, LQU$^+$-Res and IRM. While we also know that M-Res cannot simulate LQU$^+$-Res, it remains open whether M-Res can actually simulate LD-Q-Res or even IRM. We have also looked at the role of weakening and shown that it adds power to M-Res.

On the negative side, we have shown that M-Res with and without strategy weakening is unnatural — which could well make it useless in practice. It is possible, but yet unproven, that M-Res can be made natural by adding existential weakening or both weakenings. This, in our opinion, is the most important open question about M-Res. However, it is worth noting that even for propositional SAT solving, the most successful solvers are based on determinizations of CDCL, which is not natural (closed under restrictions), and so unnaturalness may not be a real hindrance after all.

Acknowledgment

We thank Olaf Beyersdorff, Joshua Blinkhorn, and Tomáš Peitl for interesting discussions about the power of M-Res. Part of this work was done at Schloss Dagstuhl Leibniz Centre for Informatics during seminar 20061 (SAT and Interactions).

References

[Ats04] Albert Atserias. On sufficient conditions for unsatisfiability of random formulas. *J. ACM*, 51(2):281–311, 2004. doi:10.1145/972639.972645.

[BBCP20] Olaf Beyersdorff, Ilario Bonacina, Leroy Chew, and Ján Pich. Frege systems for quantified Boolean logic. *J. ACM*, 67(2):9:1–9:36, 2020. doi:10.1145/3381881.

[BBH19] 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. doi:10.23638/LMCS-15(1:13)2019.

[BBM21] Olaf Beyersdorff, Joshua Blinkhorn, and Meena Mahajan. Building strategies into QBF proofs. *J. Autom. Reason.*, 65(1):125–154, 2021. Preliminary version in the proceedings of the STACS 2019, LIPIcs vol. 126, 14:1–14:18. doi:10.1007/s10817-020-09560-1.

[BBM+24] Olaf Beyersdorff, Joshua Blinkhorn, Meena Mahajan, Tomáš Peitl, and Gaurav Sood. Hard QBFs for Merge Resolution. *ACM Trans. Comput. Theory*, 16(2), March 2024. Preliminary version in the proceedings of FSTTCS 2020, LIPIcs vol. 182, 12:1–12:15. doi:10.1145/3638263.

[BCJ19] Olaf Beyersdorff, Leroy Chew, and Mikolás Janota. New resolution-based QBF calculi and their proof complexity. *ACM Trans. Comput. Theory*, 11(4):26:1–26:42, 2019. doi:10.1145/3352155.

[BCMS18] Olaf Beyersdorff, Leroy Chew, Meena Mahajan, and Anil Shukla. Understanding cutting planes for QBFs. *Inf. Comput.*, 262:141–161, 2018. doi:10.1016/j.ic.2018.08.002.

[BJ12] Valeriy Balabanov and Jie-Hong R. Jiang. Unified QBF certification and its applications. *Formal Methods Syst. Des.*, 41(1):45–65, 2012. doi:10.1007/s10703-012-0152-6.

[BKS04] Paul Beame, Henry A. Kautz, and Ashish Sabharwal. Towards understanding and harnessing the potential of clause learning. *J. Artif. Intell. Res.*, 22:319–351, 2004. doi:10.1613/jair.1410.

[Bla37] Archie Blake. *Canonical expressions in Boolean algebra*. PhD thesis, University of Chicago, 1937.

[BN21] Sam Buss and Jakob Nordström. Proof complexity and SAT solving. In Armin Biere, Marijn Heule, Hans van Maaren, and Toby Walsh, editors, *Handbook of Satisfiability*, pages 233–350. IOS Press, Netherlands, 2nd edition, May 2021. doi:10.3233/FAIA200990.

[BPS21] Joshua Blinkhorn, Tomás Peitl, and Friedrich Slivovsky. Davis and Putnam meet Henkin: Solving DQBF with resolution. 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 30–46. Springer, 2021. doi:10.1007/978-3-030-80223-3_4.

[BWJ14] 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. doi:10.1007/978-3-319-09284-3_12.

[Che17] Leroy Nicholas Chew. *QBF proof complexity*. PhD thesis, University of Leeds, 2017. URL: https://etheses.whiterose.ac.uk/18281/.

[Che21] 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.

[CR79] Stephen A. Cook and Robert A. Reckhow. The relative efficiency of propositional proof systems. *Journal of Symbolic Logic*, 44(1):36–50, 1979.

[CS23] Sravanthi Chede and Anil Shukla. Extending merge resolution to a family of QBF-proof systems. In Petra Berenbrink, Patricia Bouyer, Anuj Dawar, and Mamadou Moustapha Kanté, editors, *40th International Symposium on Theoretical Aspects of Computer Science, STACS 2023, March 7-9, 2023, Hamburg, Germany*, volume 254 of *LIPIcs*, pages 21:1–21:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023. URL: https://doi.org/10.4230/LIPIcs.STACS.2023.21, doi:10.4230/LIPICS.STACS.2023.21.

[CS24] Leroy Chew and Friedrich Slivovsky. Towards uniform certification in QBF. *Log. Methods Comput. Sci.*, 20(1), 2024. Preliminary version in the proceedings of STACS 2022, LIPIcs vol. 219, 22:1–22:23. doi:10.46298/LMCS-20(1:14)2024.

[DLL62] Martin Davis, George Logemann, and Donald W. Loveland. A machine program for theorem-proving. *Commun. ACM*, 5(7):394–397, 1962. doi:10.1145/368273.368557.

[DP60] Martin Davis and Hilary Putnam. A computing procedure for quantification theory. *J. ACM*, 7(3):201–215, 1960. doi:10.1145/321033.321034.

[ELW13] Uwe Egly, Florian Lonsing, and Magdalena Widl. Long-distance resolution: Proof generation and strategy extraction in search-based QBF solving. In Kenneth L. McMillan, Aart Middeldorp, and Andrei Voronkov, editors, *Logic for Programming, Artificial Intelligence, and Reasoning - 19th International Conference, LPAR-19, Stellenbosch, South Africa, December 14-19, 2013. Proceedings*, volume 8312 of *Lecture Notes in Computer Science*, pages 291–308. Springer, 2013. doi:10.1007/978-3-642-45221-5_21.

[Gel12] Allen Van Gelder. Contributions to the theory of practical quantified boolean formula solving. In Michela Milano, editor, *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. doi:10.1007/978-3-642-33558-7_47.

[HK17] Marijn J. H. Heule and Oliver Kullmann. The science of brute force. *Commun. ACM*, 60(8):70–79, 2017. doi:10.1145/3107239.

[JM15] Mikolás Janota and João Marques-Silva. Expansion-based QBF solving versus Q-resolution. *Theor. Comput. Sci.*, 577:25–42, 2015. doi:10.1016/j.tcs.2015.01.048.

[KKF95] Hans Kleine Büning, Marek Karpinski, and Andreas Flögel. Resolution for quantified boolean formulas. *Inf. Comput.*, 117(1):12–18, 1995. doi:10.1006/inco.1995.1025.

[Kra19] Jan Krajíček. *Proof Complexity*. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 2019. doi:10.1017/9781108242066.

[MMZ$^+$01] Matthew W. Moskewicz, Conor F. Madigan, Ying Zhao, Lintao Zhang, and Sharad Malik. Chaff: Engineering an efficient SAT solver. In *Proceedings of the 38th Design Automation Conference, DAC 2001, Las Vegas, NV, USA, June 18-22, 2001*, pages 530–535. ACM, 2001. doi:10.1145/378239.379017.

[MS22] Meena Mahajan and Gaurav Sood. QBF merge resolution is powerful but unnatural. In Kuldeep S. Meel and Ofer Strichman, editors, *25th International Conference on Theory and Applications of Satisfiability Testing, SAT 2022, August 2-5, 2022, Haifa, Israel*, volume 236 of *LIPIcs*, pages 22:1–22:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPICS.SAT.2022.22.

[MSLM21] Joao Marques-Silva, Ines Lynce, and Sharad Malik. Conflict-driven clause learning SAT solvers. In Armin Biere, Marijn Heule, Hans van Maaren, and Toby Walsh, editors, *Handbook of Satisfiability*, pages 133–182. IOS Press, Netherlands, 2nd edition, May 2021. doi:10.3233/FAIA200987.

[Rob65] John Alan Robinson. A machine-oriented logic based on the resolution principle. *J. ACM*, 12(1):23–41, 1965. doi:10.1145/321250.321253.

[SBPS19] Ankit Shukla, Armin Biere, Luca Pulina, and Martina Seidl. A survey on applications of quantified Boolean formulas. In *31st IEEE International Conference on Tools with Artificial*

Intelligence, ICTAI 2019, Portland, OR, USA, November 4-6, 2019, pages 78–84. IEEE, 2019. doi:10.1109/ICTAI.2019.00020.

[SM73] Larry J. Stockmeyer and Albert R. Meyer. Word problems requiring exponential time: Preliminary report. In Alfred V. Aho, Allan Borodin, Robert L. Constable, Robert W. Floyd, Michael A. Harrison, Richard M. Karp, and H. Raymond Strong, editors, *Proceedings of the 5th Annual ACM Symposium on Theory of Computing, April 30 - May 2, 1973, Austin, Texas, USA*, pages 1–9. ACM, 1973. doi:10.1145/800125.804029.

[SS99] João P. Marques Silva and Karem A. Sakallah. GRASP: A search algorithm for propositional satisfiability. *IEEE Trans. Computers*, 48(5):506–521, 1999. doi:10.1109/12.769433.

[Urq87] Alasdair Urquhart. Hard examples for resolution. *J. ACM*, 34(1):209–219, 1987. doi:10.1145/7531.8928.

[Var14] Moshe Y. Vardi. Boolean satisfiability: theory and engineering. *Commun. ACM*, 57(3):5, 2014. doi:10.1145/2578043.

[ZM02] Lintao Zhang and Sharad Malik. Conflict driven learning in a quantified Boolean satisfiability solver. In Lawrence T. Pileggi and Andreas Kuehlmann, editors, *Proceedings of the 2002 IEEE/ACM International Conference on Computer-aided Design, ICCAD 2002, San Jose, California, USA, November 10-14, 2002*, pages 442–449. ACM / IEEE Computer Society, 2002. doi:10.1145/774572.774637.

Appendix A. Complete proof of Lemma 4.3

As mentioned in subsection 4.1, the proof of the M-Res lower bound for KBKF-lq formulas [BBM$^+$24, Theorem 3.17] can be generalized to M-ResW$_\forall$ by doing a minor modification to the proof. In this section, we sketch the proof of [BBM$^+$24, Theorem 3.17], highlighting the appropriate modification needed to handle strategy weakening. Most of the text is reproduced directly from [BBM$^+$24].

Recall the definition of the KBKF-lq formulas from Definition 3.3. Note that the existential part of each clause in KBKF-lq$_n$ is a Horn clause (at most one positive literal), and except A_0, is even strict Horn (exactly one positive literal).

We use the following shorthand notation. Sets of variables: $D = \{d_1, \ldots, d_n\}$, $E = \{e_1, \ldots, e_n\}$, $F = \{f_1, \ldots, f_n\}$, and $X = \{x_1, \ldots, x_n\}$. Sets of literals: For $Y \in \{D, E, X, F\}$, set $Y^1 = \{u \mid u \in Y\}$ and $Y^0 = \{\overline{u} \mid u \in Y\}$. Sets of clauses:

$$\begin{array}{lcllcl}
\mathcal{A}_0 & = & \{A_0\} & & & \\
\mathcal{A}_i & = & \{A_i^d, A_i^e\} \quad \forall i \in [n] & \mathcal{B}_i & = & \{B_i^0, B_i^1\} \quad \forall i \in [n] \\
\mathcal{A}_{[i,j]} & = & \cup_{k\in[i,j]}\mathcal{A}_k \quad \forall i,j \in [0,n], i \le j & \mathcal{B}_{[i,j]} & = & \cup_{k\in[i,j]}\mathcal{B}_k \quad \forall i,j \in [n], i \le j \\
\mathcal{A} & = & \mathcal{A}_{[0,n]} & \mathcal{B} & = & \mathcal{B}_{[1,n]}
\end{array}$$

Lemma 4.3 asserts that KBKF-lq requires exponential-size to refute in M-ResW$_\forall$. The proof follows the following high-level idea. Let Π be a M-ResW$_\forall$ refutation of KBKF-lq. Since every axiom of KBKF-lq contains a variable from F while the final clause of Π is empty, there is a maximal "component" (say $\mathcal{S}$) of Π leading to and including the final line, where all clauses are F-free. The clauses in this component only contain variables in D and E. We show that the "boundary" ($\partial\mathcal{S}$) of this component is large. To show that the boundary is large, [BBM$^+$24] identify a property of merge maps called *self-dependence* which captures the right complexity; a merge map for x_i has this self-dependence property if it depends on at least one of d_i, e_i. We show that all merge maps at the final line must have self-dependence, whereas at the boundary lines none of the merge maps have self-dependence. We use this to then conclude that there must be exponentially many lines.

To show that self-dependence is not possible outside the F-free component, we show that from a line with F-variables and at least one self-dependent strategy, the F-variables can never be removed.

Elaborating on the roadmap of the argument: Let Π be an M-ResW$_\forall$ refutation of KBKF-lq$_n$. Each line in Π has the form $L = (C, M^{x_1}, \ldots, M^{x_n})$ where C is a clause over D, E, F, and each M^{x_i} is a merge map computing a strategy for x_i.

Define $\mathcal{S}$ to be the set of those lines in Π where the clause part has no F variable and furthermore the line has a path in G_Π to the final empty clause via lines where all the clauses also have no F variables. Let $\partial\mathcal{S}$, called the boundary of $\mathcal{S}$, denote the set of leaves in the subgraph of G_Π restricted to $\mathcal{S}$; these are lines that are in $\mathcal{S}$ but their parents are not in $\mathcal{S}$. Note that by definition, for each $L = (C, \{M^{x_i} \mid i \in [n]\}) \in \mathcal{S}$, $\mathrm{var}(C) \subseteq D \cup E$. No line in $\mathcal{S}$ (and in particular, no line in $\partial\mathcal{S}$) is an axiom since all axiom clauses have variables from F.

Recall that the variables of KBKF-lq$_n$ can be naturally grouped based on the quantifier prefix: for $i \in [n]$, the ith group has d_i, e_i, x_i, and the $(n+1)$th group has the F variables. By construction, the merge map for x_i does not depend on variables in later groups, as is indeed required for a countermodel. We say that a merge map for x_i has *self-dependence* if it does depend on d_i and/or e_i.

We show that every merge map at every line in $\mathcal{S}$ is non-trivial (Lemma A.5). Further, we show that at every line on the boundary of $\mathcal{S}$, i.e. in $\partial\mathcal{S}$, no merge map has self-dependence (Lemma A.6). Using this, we conclude that $\partial\mathcal{S}$ must be exponentially large, since in every countermodel the strategy of each variable must have self-dependence (Proposition A.7).

In order to show that lines in $\partial\mathcal{S}$ do not have self-dependence, we first establish several properties of the sets of axiom clauses used in a sub-derivation (Lemmas A.1, A.2, A.3 and A.4).

For a line $L \in \Pi$, let Π_L be the minimal sub-derivation of L, and let G_{Π_L} be the corresponding subgraph of G_Π with sink L. Let $\textsc{Uci}(\Pi_L) = \{i \in [0, n] \mid \mathrm{leaves}(G_{\Pi_L}) \cap \mathcal{A}_i \neq \emptyset\}$. (Uci stands for UsedConstraintsIndex). Note that we are only looking at the clauses in $\mathcal{A}$ to define Uci.

Lemma A.1 [BBM+24, Lemma 3.18]**.** *For every line $L = (C, \{M^{x_i} \mid i \in [n]\})$ of Π, $|C \cap F^1| \leq 1$. Furthermore, $\textsc{Uci}(\Pi_L) = \emptyset \Leftrightarrow C \cap F^1 \neq \emptyset$. (Here, $F^1 = \{f_1, f_2, \ldots, f_n\}$ is the set of positive literals over the variable set $\{f_1, f_2, \ldots, f_n\}$.)*

Lemma A.2 (adapted from [BBM+24, Lemma 3.19])**.** *A line $L = (C, \{M^{x_i} \mid i \in [n]\})$ of Π with $\textsc{Uci}(\Pi_L) = \emptyset$ has these properties:*

(1) *$var(C) \subseteq F$; for all $i \in [n]$, $M^{x_i} \in \{*, 0, 1\}$;*
(2) *For some $j \in [n]$, $f_j \in C$ and $M^{x_j} \in \{0, 1\}$; such a j is unique;*
(3) *For the unique j from (2), for $1 \leq i < j$, $f_i \notin var(C)$ and $M^{x_i} \in \{*, 0, 1\}$;*
(4) *For $j < i \leq n$, if $f_i \notin var(C)$, then $M^{x_j} \in \{0, 1\}$.*

This statement differs from [BBM+24, Lemma 3.19] **in exactly one respect:** in item 3 we have a weaker conclusion (already shown in item 1) that $M^{x_i} \in \{*, 0, 1\}$, whereas in [BBM+24, Lemma 3.19] it was further proved that $M^{x_i} = *$. Since strategy weakening allows us to replace a * by 0 or 1, we cannot draw this conclusion. However, the stronger conclusion was not used to prove the subsequent item 4, so nothing changes in the proof.

In subsequent lemmas, whenever we use this lemma, we need to show that the weaker conclusion suffices. It turns out that subsequent lemmas use this item essentially to say that the merge map is not complex; it is trivial $*$ or simple $0, 1$. This conclusion remains valid with the modified version.

Lemma A.3 [BBM+24, Lemma 3.20]**.** *Let $L = (C, \{M^{x_i} \mid i \in [n]\})$ be a line of Π with $\textsc{Uci}(\Pi_L) \neq \emptyset$. Then $\textsc{Uci}(\Pi_L)$ is an interval $[a, b]$ for some $0 \leq a \leq b \leq n$. Furthermore, (in the items below, a, b refer to the endpoints of this interval), it has the following properties:*

(1) *For $k \in [n] \cap [a, b]$, $M^{x_k} \neq *$.*
(2) *If $a \geq 1$, then $|\{d_a, e_a\} \cap C| = 1$. If $a = 0$, then C does not have any positive literal.*
(3) *If $b < n$, then $\overline{d_{b+1}}, \overline{e_{b+1}} \in C$.*
(4) *For all $k \in [n] \setminus [a, b]$, (i) $d_k, e_k \notin var(M^{x_k})$, and (ii) if $M^{x_k} = *$ then $\overline{f_k} \in C$.*

Lemma A.4 [BBM+24, Lemma 3.21]**.** *For any line $L = (C, \{M^{x_i} \mid i \in [n]\})$ in Π, and any $k \in [n]$, if $\{d_k, e_k\} \cap var(M^{x_k}) \neq \emptyset$, then $\textsc{Uci}(\Pi_L) = [a, n]$ for some $a \leq k - 1$.*

Lemma A.5 [BBM+24, Lemma 3.22]**.** *For all $L \in \mathcal{S}$, for all $k \in [n]$, $M^{x_k} \neq *$.*

Lemma A.6 [BBM+24, Lemma 3.23]**.** *For all $L \in \partial\mathcal{S}$, for all $k \in [n]$, $d_k, e_k \notin var(M^{x_k})$.*

We will also use the following property of KBKF-lq formulas. It implies that in every countermodel, the strategy for every variable has self-dependence. This is used, towards the end of the proof of Lemma 4.3, to show that merge maps for countermodels must be complex and large.

Proposition A.7 [BBM+24, Proposition 3.24]**.** *Let h be any countermodel for KBKF-lq$_n$. Let α be any assignment to D, and β be any assignment to E. For each $i \in [n]$, if $\alpha_j \neq \beta_j$ for all $1 \leq j \leq i$, then $h^{x_i}\big((\alpha, \beta)\restriction_{L_Q(x_i)}\big) = \alpha_i$. In particular, if $\alpha_j \neq \beta_j$ for all $j \in [n]$, then the countermodel computes $h(\alpha, \beta) = \alpha$.*

Now we have all the required information; we put it together to obtain the lower bound.

Proof of Lemma 4.3. Let Π be a refutation of KBKF-lq$_n$ in M-ResW$_\forall$. Let $\mathcal{S}, \partial\mathcal{S}$ be as defined in the beginning of this section. Let the final line of Π be $L_\Box = (\Box, \{M_\Box^{x_i} \mid i \in [n]\})$, and for $i \in [n]$, let h_i be the functions computed by the merge map $M_\Box^{x_i}$. By soundness of M-ResW$_\forall$, the functions $\{h_i\}_{i \in [n]}$ form a countermodel for KBKF-lq$_n$.

For each $a \in \{0, 1\}^n$, consider the assignment α to the variables of $D \cup E$ where $d_i = a_i$, $e_i = \overline{a_i}$. Call such an assignment an anti-symmetric assignment. Given such an assignment, walk from $L_\Box$ towards the leaves of Π as far as is possible while maintaining the following invariant at each line $L = (C, \{M^{x_i} \mid i \in [n]\})$ along the way:

(1) α falsifies C, and
(2) for each $i \in [n]$, $h_i(\alpha) = M^{x_i}(\alpha)$.

Clearly this invariant is initially true at $L_\Box$, which is in $\mathcal{S}$. If we are currently at a line $L \in \mathcal{S}$ where the invariant is true, and if $L \notin \partial\mathcal{S}$, then consider how L is derived. If it is obtained by using strategy weakening on some L', then $C' = C$ and so L' is also in $\mathcal{S}$. Then by Lemma A.5, each strategy in L' is already non-trivial, so no weakening is possible. **(Note: This argument, that there is no strategy weakening inside S, is the only addition needed to adapt the lower bound for M-Res to M-ResW$_\forall$.)**

Hence it must be the case that L is derived using resolution. Say L is obtained from lines L', L''. The resolution pivot in this step is not in F, since that would put L in $\partial\mathcal{S}$. So both L' and L'' are in $\mathcal{S}$, and the pivot is in $D \cup E$. Let the pivot be in $\{d_\ell, e_\ell\}$ for some $\ell \in [n]$. Depending on the pivot value, exactly one of C', C'' is falsified by α; say C' is falsified. By Lemma A.5, for each $i \in [n]$, both $(M')^{x_i}$ and $(M'')^{x_i}$ are non-trivial. By definition of the M-Res rule,

- For $i < \ell$, $(M')^{x_i}$ and $(M'')^{x_i}$ are isomorphic (otherwise the resolution is blocked), and $M^{x_i} = (M')^{x_i} = (M'')^{x_i}$.
- For $i \geq \ell$, there are two possibilities:
  (1) $(M')^{x_i}$ and $(M'')^{x_i}$ are isomorphic, and $M^{x_i} = (M')^{x_i}$.
  (2) M^{x_i} is a merge of $(M')^{x_i}$ and $(M'')^{x_i}$ with the pivot variable queried. By definition of the merge operation, since C' is falsified by α, $M^{x_i}(\alpha) = (M')^{x_i}(\alpha)$.

Thus in all cases, for each i, $h_i(\alpha) = M^{x_i}(\alpha) = (M')^{x_i}(\alpha)$. Hence L' satisfies the invariant.

We have shown that as long as we have not encountered a line in $\partial\mathcal{S}$, we can move further. We continue the walk until a line in $\partial\mathcal{S}$ is reached. We denote the line so reached by $P(\alpha)$. Thus P defines a map from anti-symmetric assignments to $\partial\mathcal{S}$.

We now show that the map P is one-to-one. Suppose, to the contrary, $P(\alpha) = P(\beta) = (C, \{M^{x_i} \mid i \in [n]\})$ for two distinct anti-symmetric assignments obtained from $a, b \in \{0,1\}^n$ respectively. Let j be the least index in $[n]$ where $a_j \neq b_j$. By Lemma A.6, M^{x_j} depends only on $\{d_i, e_i \mid i < j\}$, and α, β agree on these variables. Thus we get the equalities $a_j = h_j(\alpha) = M^{x_j}(\alpha) = M^{x_j}(\beta) = h_j(\beta) = b_j$, where the first and last equalities follow from Proposition A.7, the third equality from Lemma A.6 and choice of j, and the second and fourth equalities by the invariant satisfied at $P(\alpha)$ and $P(\beta)$ respectively. This contradicts $a_j \neq b_j$.

We have established that the map P is one-to-one. Hence, $\partial\mathcal{S}$ has at least as many lines as anti-symmetric assignments, so $|\Pi| \geq |\partial\mathcal{S}| \geq 2^n$. □