# Classes of Hard Formulas for QBF Resolution

Agnes Schleitzer
Institut für Informatik, Friedrich-Schiller-Universität Jena, Germany

Olaf Beyersdorff
Institut für Informatik, Friedrich-Schiller-Universität Jena, Germany

## Abstract

To date, we know only a few handcrafted quantified Boolean formulas (QBFs) that are hard for central QBF resolution systems such as Q-Res and QU-Res, and only one specific QBF family to separate Q-Res and QU-Res.

Here we provide a general method to construct hard formulas for Q-Res and QU-Res. The construction uses simple propositional formulas (e.g. minimally unsatisfiable formulas) in combination with easy QBF gadgets ($\Sigma_2^b$ formulas without constant winning strategies). This leads to a host of new hard formulas, including new classes of hard random QBFs.

We further present generic constructions for formulas separating Q-Res and QU-Res, and for separating Q-Res and LD-Q-Res.

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

**Keywords and phrases** QBF, proof complexity, resolution, separations

**Digital Object Identifier** 10.4230/LIPIcs.SAT.2022.5

**Funding** *Agnes Schleitzer*: DFG grant BE 4209/3-1.
*Olaf Beyersdorff*: Carl-Zeiss Foundation and DFG grant BE 4209/3-1.

## 1 Introduction

The main objective in *proof complexity* is to study the size of proofs in different formal proof systems. Proof complexity has its origins in computational complexity [27] with many important connections to other fields, in particular to logic [26, 33] and solving [22]. For the latter, proof complexity provides the main theoretical tool to assess the strength of modern solving methods.

The main objective in proof complexity – and often also the most challenging – is to show *lower bounds* to the size of proofs and to obtain *separations* between different calculi. For this, *specific formula families* are needed on which the lower bounds are demonstrated. In propositional proof complexity and in particular for propositional resolution – arguably the best studied system, not least because of its tight connections to SAT solving [4, 8, 22, 36] – there is a vast literature on hard formulas stemming from diverse areas such as combinatorics (e.g. [21, 29]), graph theory [39], logic [32], random formulas [7], and many more [33, 37].

In comparison, *proof complexity of quantified Boolean formulas* (QBF) is at an earlier stage. As in the propositional domain, QBF resolution systems received key attention, of which Q-Resolution (Q-Res, [31]) and QU-Resolution (QU-Res, [40]) are the most important base systems. They augment the propositional resolution system by a simple universal reduction rule allowing to eliminate certain universal variables from clauses.

As in SAT, QBF resolution systems are intricately connected to QBF solving (cf. [18] for a recent overview), with Q-Res and its extension long-distance Q-Resolution (LD-Q-Res, [5]) corresponding to quantified conflict-driven clause learning (QCDCL) (cf. [14, 18, 34, 41]).

In contrast to the multitude of hard formulas for propositional resolution, we are somewhat short of interesting QBF families that are amenable to a proof-theoretic study. Only a handful of QBF families (and their modifications) have been used for lower bounds and separations

25th International Conference on Theory and Applications of Satisfiability Testing (SAT 2022).
Editors: Kuldeep S. Meel and Ofer Strichman; Article No. 5; pp. 5:1–5:18
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

in the QBF literature. The most prominent of these are arguably the KBKF formulas from the very first article [31] that introduced Q-Res. The other "notorious" QBF families are the equality formulas [11], the parity formulas [15], and the CR formulas [30]. Together these more or less comprise the formula toolbox of QBF proof complexity and are used for almost all of the known separations.

It would thus be desirable to have more interesting and natural QBFs that can be shown to be hard for Q-Res or QU-Res. More such QBFs would not only be valuable for proof complexity, but also for solving where they can be used as benchmarks to compare different solving techniques.[1]

It is also not so easy to tap into the fund of hard propositional formulas. While the existentially quantified version of each CNF that is hard for propositional resolution is trivially also hard for Q-Res and QU-Res, we are rather interested in "genuine" QBF hardness that stems from quantifier alternations and not from the propositional base system.[2]

**Our Contributions.** Our contributions can be summarised as follows.

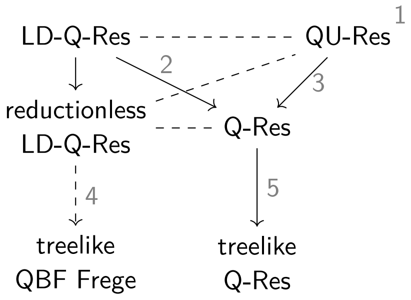
**(1)** **Hard QBFs for Q-Res and QU-Res.** We introduce a generic construction to obtain large classes of QBFs that are hard for Q-Res and QU-Res. The construction uses two key ingredients: (i) suitable propositional base formulas and (ii) simple QBF gadgets. The *propositional base formula* needs to have a sufficiently large set of clauses that we identify as "critical", e.g. all minimally unsatisfiable formulas meet that requirement. Otherwise, the base formulas can be quite simple (and in particular can be easy for propositional resolution). The *QBF gadget* must be a false $\Sigma_2^b$ formula without a constant winning strategy for the universal player in the evaluation game for QBFs. Otherwise, the gadgets can again be quite simple.
We then combine the propositional base formula with the QBF gadgets in a rather straightforward way to obtain $\Sigma_3^b$ QBFs that require exponential-size proofs in Q-Res and QU-Res. The lower bound follows by the size-cost lower-bound technique [11] that always yields "genuine" QBF lower bounds, i.e., our construction yields "genuinely" hard QBFs in the sense discussed above.
We illustrate our method with a couple of examples. These include the equality formulas [11] (which actually inspired our construction), new circle, equivalence, and XOR formulas, as well as a large class of random QBFs.

**(2)** **Separations between Q-Res and LD-Q-Res.** We show that our construction above yields QBFs that exponentially separate the systems Q-Res and LD-Q-Res, if the propositional base formulas are easy for propositional resolution and the QBF gadgets are easy for Q-Res. These conditions are met by all our examples above.
This should be welcome news as we previously knew of only very few formulas (essentially KBKF, equality, and parity) that separate Q-Res from LD-Q-Res [11, 15, 23, 28].

**(3)** **Separations between Q-Res and QU-Res.** To obtain separations between Q-Res and QU-Res, we first modify the $\Sigma_3^b$ prefix of the QBFs constructed in (1) to an unbounded "interleaved" prefix. These "interleaved" QBFs become easy for Q-Res (while still retaining hardness for treelike Q-Res), but a further "tail" construction (inspired by KBKF) modifies them into QBFs that become hard for Q-Res, yet easy for QU-Res.

[1] A track of crafted formulas was introduced into QBFEval 2020 and a tool to generate the mentioned QBF families was presented in [19].

[2] A formal framework for "genuine" QBF hardness was introduced in [17]. All the mentioned QBF examples – KBKF, equality, and parity – are genuinely hard in this sense.



Hard Formulas

| | |
|---|---|
| 1 | Section 3: Theorem 12 |

Separating Formulas

| | |
|---|---|
| 2 | Section 4: Corollary 26 |
| 3 | Section 5: Theorem 34 |
| 4 | Section 4: Proposition 29 |
| 5 | Section 5: Proposition 31 |

**Figure 1** The simulation order of QBF proof systems mentioned in this article and our contributions to formulas for lower bounds and separations. $A \longrightarrow B$ : $A$ simulates $B$ + exponential separation; $A$ - - - $B$ : $A$ and $B$ are incomparable; $A \dashrightarrow B$ : $B$ does not simulate $A$.

In comparison to our quite transparent method in (1) above, the technical details of these constructions are somewhat more involved. Yet again we obtain a large class of QBFs separating Q-Res and QU-Res. Previously, the KBKF formulas were the only known separating example [10,31,40]. Interestingly, all formulas we construct in (3) have unbounded quantifier complexity, which we know must be the case for a separation of QU-Res from Q-Res [12,25].

The simulation order of the proof systems mentioned in this paper as well as pointers to the relevant results are shown in Figure 1.

**Organisation.** We start in Section 2 with preliminaries on QBF and the relevant proof systems. Section 3 contains our generic construction of hard QBFs together with a couple of examples. QBFs separating LD-Q-Res from Q-Res and of QU-Res from Q-Res are constructed in Sections 4 and 5, respectively. We conclude in Section 6 with some open questions. Due to space constraints some details are omitted from this paper.

## 2 Preliminaries

A *CNF (conjunctive normal form)* is a conjunction of disjunctions of literals. The disjunctions are called *clauses*. A *literal* $l$ is a propositional variable $x$ or its negation $\overline{x}$, we write $\mathsf{vars}(l) = x$.

**QBFs.** A *quantified Boolean formula (QBF)* in *closed prenex form* $\phi = \mathcal{P} \cdot \varphi$ consists of a *quantifier prefix* $\mathcal{P}$ and a propositional formula $\varphi$, called the *matrix*. The prefix is a series of quantifiers $Q \in \{\forall, \exists\}$, each followed by a set of variables. For a *closed* QBF (which we only consider here), $\mathcal{P}$ quantifies exactly the variables occurring in $\varphi$. Thus, for $\mathcal{P} = Q_1X_1Q_2X_2 \dots Q_nX_n$, the matrix $\varphi$ is a formula in variables $\bigcup_{i\in[n]} X_i$ and we write $\mathsf{vars}(\mathcal{P} \cdot \varphi) = \mathsf{vars}(\varphi) = \bigcup_{i\in[n]} X_i$. As there are no free variables in a closed QBF, it is either *true* or *false*. We write $\mathsf{vars}_\exists(\varphi)$ for the set of existential variables in $\mathcal{P} \cdot \varphi$ and $\mathsf{vars}_\forall(\varphi)$ for those associated with $\forall$. A QCNF is a QBF with a CNF matrix.

An *assignment* assigns truth values to variables. We denote by $v^\alpha$ the value of a variable $v$ under an assignment $\alpha$. We write $\langle V \rangle$ for the set of all possible assignments to $V$, $\langle \chi \rangle = \langle \mathsf{vars}(\chi) \rangle$ for the assignments of a propositional formula $\chi$ and $\langle \phi \rangle = \langle \mathcal{P} \cdot \varphi \rangle = \langle \varphi \rangle$ for those of a QBF $\phi = \mathcal{P} \cdot \varphi$.

Closed QBFs can be viewed as a game between an existential and a universal player generating a total assignment [38]. The players assign truth values to all variables in the order of the quantifier prefix (the existential player chooses the values for existential variables, the

| | | |
|---|---|---|
| `Axiom` | $\dfrac{}{C}$ | $C$ is a non-tautologous clause in the matrix $\varphi$. |
| `Q-Res` | $\dfrac{C_1 \cup \{x\} \quad C_2 \cup \{\overline{x}\}}{C_1 \cup C_2}$ | $C_1 \cup C_2$ is non-tautologous; $x \in \mathsf{vars}_\exists(\phi)$. |
| `QU-Res` | $\dfrac{C_1 \cup \{x\} \quad C_2 \cup \{\overline{x}\}}{C_1 \cup C_2}$ | $C_1 \cup C_2$ is non-tautologous. |
| `LDQ-Res` | $\dfrac{C_1 \cup \{x\} \quad C_2 \cup \{\overline{x}\}}{C_1^* \cup C_2^* \cup U^*}$ | $l^* = l \vee \bar{l}$, $\{l^*\} = \{l, \bar{l}\}$ for any literal $l$;<br>$C_1^* = C_1 \setminus (C_1 \cap \overline{C_2})$; $C_2^* = C_2 \setminus (\overline{C_1} \cap C_2)$;<br>$U^* = \{u^* \mid u \in \mathsf{vars}(C_1 \cap \overline{C_2})\}$; $x \in \mathsf{vars}_\exists(\phi)$;<br>$C_1 \cup C_2$ does not contain any existential tautologies; any $u \in \mathsf{vars}(U^*)$ is quantified right of $x$ in $\mathcal{P}$. |
| `∀Red` | $\dfrac{C \cup \{u\}}{C}$ | $u \in \mathsf{vars}_\forall(\phi)$ and quantified right of each existential variable in $C$ regarding $\mathcal{P}$. |

**Figure 2** Rules of the QBF proof systems Q-Res, QU-Res and LD-Q-Res for a QBF $\phi = \mathcal{P}.\varphi$.

universal player those for universals). The existential player wins, if the generated assignment satisfies the matrix; otherwise the universal player wins. For a closed QBF, there is always a *winning strategy* for one of the two players. We call this game the *assignment game*.

A countermodel is a winning strategy for the universal player. While countermodels are often considered as a collection of functions (one for each universal variable), we prefer to understand them as a single function, whose output is an assignment to the universal variables (for further explanations see e.g. [12]). The range of a countermodel is therefore the number of different assignments to the universal variables that can be generated within the framework of the associated strategy. The range of a countermodel on a single universal block is analogously the number of different assignments to the variables of this block.

We define *strategy size* in accordance with [9]:

▶ **Definition 1** (Strategy Size $\rho$ [9]). *Let $\phi$ be a false QBF. We refer to the smallest cardinality of the range of a countermodel for $\phi$ as the* strategy size *$\rho(\phi)$ of $\phi$.*

**Proof systems.** *Resolution* (Res) is a refutational proof system for propositional formulas with only two inference rules: For an input formula $\chi$, we can derive any $C \in \chi$ as an axiom and from two Clauses $C_1 \cup \{x\}$, $C_2 \cup \{\overline{x}\}$ we can derive the resolvent $C_1 \cup C_2$ by Resolution over the pivot $x$.

*Q-Res* [31] transfers Resolution from propositional logic to QBF. It uses the resolution rule (`Q-Res`) which only allows existential pivots and forbids tautologous resolvents. Universal variables are eliminated by universal reduction (`∀Red`). The rules are given in Figure 2.

*QU-Res* [40] extends the weaker system Q-Res by allowing resolution also over universal pivots in its resolution rule `QU-Res`. Nevertheless Q-Res is refutationally sound and complete.

*LD-Q-Res* [5] is an extension of Q-Res which allows long-distance resolution steps under certain conditions (see Figure 2 for the definition of the resolution rule `LDQ-Res`), allowing tautological resolvents. The `∀Red` rule is modified such that merged universal literals from long distance steps can also be reduced under the same conditions as usual universal variables.

The size of a proof $\pi$, denoted $|\pi|$, is the number of clauses in $\pi$. A proof system $S$ *p-simulates* a system $S'$, if every $S'$ proof can be transformed in polynomial time into an $S$ proof of the same formula.

# 3 Construction of Hard Formulas for QU-Res

We start by recalling the lower-bound technique for QU-Res via cost from [11].

▶ **Definition 2** (Cost). *We consider all countermodels for a false QBF $\phi$ and determine for each of them the largest range on a single universal block. The minimum over these cardinalities is the* cost *of $\phi$.*

For $\Sigma_3^b$ formulas (i.e., with only one universal block), cost coincides with strategy size (Definition 1). Cost is an absolute lower bound for proof size in QU-Res (and Q-Res):

▶ **Theorem 3** ( [11]). *Let $\phi$ be a false QCNF. Then QU-Res refutations of $\phi$ have size at least* $\operatorname{cost}(\phi)$.

The equality formulas from [11] have exponential cost and are therefore hard for QU-Res:

▶ **Definition 4** (Equality formulas [11]). *For $n \in \mathbb{N}$ we define the $n^{th}$equality formula as*

$$\mathrm{EQ}_n = \exists x_1 \dots x_n \forall u_1 \dots u_n \exists t_1 \dots t_n \cdot \left( \bigcup_{i\in[n]} \{\{x_i, u_i, \overline{t_i}\}, \{\overline{x_i}, \overline{u_i}, \overline{t_i}\}\} \right) \cup \{\{t_1, \dots, t_n\}\} . \quad (1)$$

We take the equality formulas as a starting point and then subsequently generalize their construction. The underlying principle of the equality formulas is to enforce a unique universal winning strategy of exponential size. In the case of equality, the winning strategy is to assign $u_i = x_i$. The formulas can be understood as being based on a simple propositional formula consisting of the clause $\{t_1, \dots, t_n\}$ and unit clauses $\{\overline{t_1}\}, \dots, \{\overline{t_n}\}$, into which this exponential size winning strategy is injected through adding the $x$ and $u$ variables.

Based on this intuition, we outline a general construction for hard QBFs, comprising the following steps:

- Find a family $(\chi_i)_{i\in\mathbb{N}}$ of propositional formulas whose $n^{\text{th}}$member $\chi_n$ has at least $n$ critical clauses (we define that notion in Definition 5).
- Find QBF gadgets (defined in Definition 9) that enforce exponential strategy size.
- Connect the two components such that any winning strategy has exponential range and forces the existential player to lose on the propositional formula.

## 3.1 Suitable Propositional Formulas

Let us first formally define the afore mentioned critical clauses:

▶ **Definition 5** (critical clauses). *For an unsatisfiable propositional formula $\chi$ we call a clause $C \in \chi$* critical, *if $\chi \setminus \{C\}$ is satisfiable. We call a set $\mathcal{C} \subseteq \chi$ critical, if any $C \in \mathcal{C}$ is critical.*

Note that for a minimally unsatisfiable formula, every subset of clauses is critical.

We now have a look at some suitable propositional formula families. We will denote the critical clauses by $\mathcal{C} = \{C_i \mid i \in [n]\}$ and by $\mathcal{D} = \{D_i \mid i \in [|\chi_n| - n]\}$ the remaining clauses. The subset of critical clauses can be chosen in more than one way, but for each example we make a specific choice that we will also use later in the construction of the hard QBFs.

The underlying propositional formulas from the equality formulas are:

▶ **Example 6** (Simple Contradiction). $\mathrm{SC}_n = \{D_1\} \cup \bigcup_{i\in[n]}\{C_i\}$ with $D_1 = \{t_1, \dots, t_n\}$ and $C_i = \{\overline{t_i}\}$ for $i \in [n]$. Note that $\mathrm{SC}_n$ is minimally unsatisfiable.

In addition, we consider two further running examples.

▶ **Example 7** (Implication Chain). $\mathrm{IC}_n = \bigcup_{i\in[n]}\{C_i\}$ with $C_i = \{t_{i-1}, \overline{t_i}\}$ for $i \in [1, n-2]$ and $C_{n-1} = \{\overline{t_0}\}$, $C_n = \{t_{n-2}\}$. In this minimally unsatisfiable formula we set $\mathcal{D} = \varnothing$.

▶ **Example 8** (Equivalence Chain). $\mathrm{EC}_n = \left(\bigcup_{i\in[n]}\{C_i, D_i\}\right) \cup \{D_{n+1}, D_{n+2}\}$ with $C_i = \{t_{i-1}, \overline{t_i}\}$, $D_i = \{\overline{t_{i-1}}, t_i\}$ for $i \in [n]$ and $D_{n+1} = \{t_0, t_n\}$, $D_{n+2} = \{\overline{t_0}, \overline{t_n}\}$. Note that even though the formula is minimally unsatisfiable, we can choose a large set $\mathcal{D}$.

## 3.2 QBF Gadgets

We now define the second ingredient of our construction, the QBF gadgets:

▶ **Definition 9** (QBF Gadget). *A* QBF gadget *is a false $\Sigma_2^b$ QBF $\phi = \mathcal{P} \cdot \varphi$ with only non-constant winning strategies, i.e., there is no strategy to falsify $\phi$ that uses only one fixed assignment to the variables in the universal block.*

In fact, it is not necessary to restrict gadgets to $\Sigma_2^b$ formulas, but it is sufficient for our purposes and simplifies constructions and proofs.

The equality formulas can be understood to use the equality gadget:

▶ **Example 10** (Equality Gadget). $\mathrm{EQ} = \exists x \forall u \cdot \{\{x, u\}, \{\overline{x}, \overline{u}\}\}$.

Note that the gadget is equivalent to $\exists x \forall u \cdot x \not\leftrightarrow u$, so the unique winning strategy for the universal player is $u = x$. Therefore it is a QBF gadget.

To see more clearly, how the equality formulas are composed from the gadget and the propositional base formulas $\mathrm{SC}_n$, we could restate (1) as

$$\exists x_1 \cdots x_n \forall u_1 \cdots u_n \exists t_1 \cdots t_n \cdot \left(\bigwedge_{i=1}^{n} ((x_i \leftrightarrow u_i) \rightarrow \bar{t}_i)\right) \wedge \left(\bigvee_{i=1}^{n} t_i\right). \tag{2}$$

The formulas (1) are then simply a transformation of (2) into CNF. Note that the gadget is not inserted into all clauses, but only into the chosen set of critical clauses of $\mathrm{SC}_n$.

The equality gadget is arguably the simplest QBF gadget and except for $\exists x \forall u \cdot x \leftrightarrow u$ the only one in two variables. Nevertheless, it is easy to construct many further gadgets. As an example, we consider the XOR gadget $\exists x^1 x^2 \forall u \cdot (x^1 \oplus x^2) \not\leftrightarrow u$, which has the unique winning strategy $u = x^1 \oplus x^2$.

▶ **Example 11** (XOR Gadget). $\mathrm{XOR} = \exists x^1 x^2 \forall u \cdot$
$\{\{x^1, x^2, u\}, \{x^1, \overline{x^2}, \bar{u}\}, \{\overline{x^1}, x^2, \bar{u}\}, \{\overline{x^1}, \overline{x^2}, u\}\}$.

It is also possible to construct gadgets with more than one universal variable, e.g. by using functions with more than one (logical) output variable (e.g. a half adder). We will use this approach to get random gadgets in Section 3.5.

## 3.3 Hard Formulas for QU-Res

We now want to combine the described propositional formulas with QBF gadgets.

We need a QBF gadget for each clause in a sufficiently large set of critical clauses. As we intend to construct families of hard QBFs, for any $n \in \mathbb{N}$ we first collect a sequence of $n$ QBF gadgets whose variables are pairwise disjoint. The simplest way to obtain such a sequence is to choose $n$ instances of the same gadget for each $n \in \mathbb{N}$. Another possibility would be to insert different gadgets into the critical clauses, e.g. we could choose them from the previously mentioned examples.

We define the product $\varphi \times C$ of a formula $\varphi$ and a clause $C$ as $\varphi \times C := \{D \cup C \mid D \in \varphi\}$. Our first main result follows:

▶ **Theorem 12.** *Let $\Phi_n = (\phi_i)_{i\in[n]} = (\exists X_i \forall U_i \cdot \varphi_i)_{i\in[n]}$ be a sequence of variable disjoint QBF gadgets and $\chi_n$ a propositional formula with a set $\mathcal{C} = \{C_1, \dots, C_n\}$ of critical clauses and a set $\mathcal{D}$ of remaining clauses. Set $T_n = \mathsf{vars}(\chi_n)$ and let $\chi_n$ have no common variables with $\bigcup_{i\in[n]}(X_i \cup U_i)$. Then*

$$\chi_n^\Phi = \exists X_1 \dots X_n \forall U_1 \dots U_n \exists T_n \left[ \bigcup_{i\in[n]} \{\varphi_i \times C_i\} \right] \cup \mathcal{D}$$

*requires QU-Res refutations of size at least $2^n$.*

It should be intuitively clear that the following holds:

▶ **Lemma 13.** *Let $\Phi_n$, $\chi_n$, and $\chi_n^\Phi$ be as described in Theorem 12. Then any winning strategy for $\chi_n^\Phi$ is a combination of winning strategies of the used gadgets in $\Phi_n$.*

**Proof of Theorem 12.** We know from Lemma 13 that any winning strategy $S$ for $\chi_n^\Phi$ is composed of winning strategies for the single gadgets. As the $n$ gadgets in $\chi_n^\Phi$ do not have constant winning strategies and are variable disjoint, the combination of winning strategies must have range at least $2^n$, i.e., $\chi_n^\Phi$ has cost $\geq 2^n$. By Theorem 3 this implies QU-Res refutations of size at least $2^n$. ◀

In this way, we get a large collection of formulas that are hard for QU-Res (and hence also for Q-Res). The constructed formulas all have a $\Sigma_3^b$ prefix, which is the result of using $\Sigma_2^b$ gadgets. The $\Sigma_3^b$ case is probably also the most natural setting as the size-cost technique from Theorem 3 essentially works for $\Sigma_3^b$ formulas. However, as mentioned, the restriction to $\Sigma_2^b$-gadgets is not necessary (we then only have to give some thought on how to suitably compose the prefix and define the non-constant property) This also allows the construction of formulas with more complex prefixes (incl. unrestricted).

## 3.4 Examples

Let us look at some example formulas which can be constructed using the propositional base formulas and the equality gadget, all of them exponentially hard for QU-Res.

▶ **Example 14** (Equality Formulas [11]). The equality formulas (Definition 4) arise from applying the equality gadgets to the simple contradiction formulas: $\mathrm{EQ}_n = \mathrm{SC}_n^{\mathrm{EQ}}$.

▶ **Example 15** (Circle Formulas). Consider now the application of equality gadgets to the implication chain formulas. For $n > 1$ we obtain the QBFs

$$\begin{aligned}\mathrm{IC}_n^{\mathrm{EQ}} =& \exists x_1, \dots, x_n \forall u_1, \dots, u_n \exists t_0, \dots, t_{n-2} \cdot \\ & \left( \bigcup_{i=1}^{n-2} \{\{u_i, x_i, t_{i-1}, \overline{t_i}\}, \{\overline{u_i}, \overline{x_i}, t_{i-1}, \overline{t_i}\}\} \right) \\ & \cup \{\{u_{n-1}, x_{n-1}, \overline{t_0}\}, \{\overline{u_{n-1}}, \overline{x_{n-1}}, \overline{t_0}\}, \{u_n, x_n, t_{n-2}\}, \{\overline{u_n}, \overline{x_n}, t_{n-2}\}\} .\end{aligned}$$

▶ **Example 16** (Equivalence Formulas). Instead of the implication chain, we can also use the equivalence chain EC. Applying equality gadgets on these formulas, we get

$$\mathrm{EC}_n^{\mathrm{EQ}} = \exists x_1 \dots x_n \forall u_1 \dots u_n \exists t_0 \dots t_n \cdot \left( \bigcup_{i\in[n]} \{C_{i,1}, C_{i,2}, D_i\} \right) \cup \{D_{n+1}, D_{n+2}\}$$

with clauses $C_{i,1} = \{x_i, u_i, t_{i-1}, \overline{t_i}\}$, $C_{i,2} = \{\overline{x_i}, \overline{u_i}, t_{i-1}, \overline{t_i}\}$, $D_i = \{\overline{t_{i-1}}, t_i\}$ for $i \in [n]$ and $D_{n+1} = \{t_0, t_n\}$, $D_{n+2} = \{\overline{t_0}, \overline{t_n}\}$.

We would argue that the circle and equivalence formulas are almost as canonical and intuitive as the already familiar equality formulas.

▶ **Example 17** (XOR Formulas). We combine the XOR gadgets (Example 11) with SC:

$$\begin{aligned}\mathrm{SC}_n^{\mathrm{XOR}} =& \exists x_1^1 x_1^2 \dots x_n^1 x_n^2 \forall u_1 \dots u_n \exists t_1 \dots t_n \cdot \\ & \left[ \bigcup_{i \in [n]} \left\{ \{x_i^1, x_i^2, u_i, \overline{t_i}\}, \{x_i^1, \overline{x_i^2}, \overline{u_i}, \overline{t_i}\}, \{\overline{x_i^1}, x_i^2, \overline{u_i}, \overline{t_i}\}, \{\overline{x_i^1}, \overline{x_i^2}, u_i, \overline{t_i}\} \right\} \right] \\ & \cup \{t_1, \dots, t_n\}.\end{aligned}$$

## 3.5 Random Formulas

Using our construction, it is also quite straightforward to obtain various random QBFs. For this we construct gadgets from Boolean functions. We need the following notion:

▶ **Definition 18** ($F$-satisfying Assignment). *For $X = \{x_1, \dots, x_a\}$, $U = \{u_1, \dots, u_b\}$ and a function $F : \langle X \rangle \to \langle U \rangle$ we call an assignment $\alpha \in \langle X \cup U \rangle$ $F$*-satisfying *iff $F(x_1^\alpha \dots x_a^\alpha) = u_1^\alpha \dots u_b^\alpha$.*

▶ **Definition 19** ($F_{a,b}$-Gadget). *An $F_{a,b}$-gadget is built from a non-constant Boolean function $F : \{0,1\}^a \to \{0,1\}^b$ as follows: We introduce sets of variables $X = \{x_1, \dots, x_a\}$ and $U = \{u_1, \dots, u_b\}$. Consider $F$ as function from $\langle X \rangle$ to $\langle U \rangle$. For any $F$-satisfying assignment $\alpha$ we add the clause $\{v \mid v^\alpha = 0\} \cup \{\overline{v} \mid v^\alpha = 1\}$. We call the following QBF an $F_{a,b}$-gadget:*

$$\mathrm{RG}_{a,b}^F = \exists x_1 \dots x_a \forall u_1, \dots u_b \cdot \{\{v \mid v^\alpha = 0\} \cup \{\overline{v} \mid v^\alpha = 1\} \mid \alpha \text{ is } F\text{-satisfying}\}.$$

It is easy to check that $F_{a,b}$-gadgets satisfy the required properties:

▶ **Lemma 20.** *Let $\mathrm{RG}_{a,b}^F$ be an $F_{a,b}$-gadget based on a Boolean function $F : \{0,1\}^a \to \{0,1\}^b$ as described in Definition 19. Then $\mathrm{RG}_{a,b}^F$ is a QBF gadget.*

**Proof.** Obviously, any such QBF is a $\Sigma_2^b$ formula. To argue for its falsity, let us consider the assignment game: First, the existential player assigns the $X$-variables. Let $\alpha$ be the $F$-satisfying extension of the chosen assignment to $X \cup U$, i.e., $F(x_1^\alpha \dots x_a^\alpha) = u_1^\alpha \dots u_b^\alpha$. The strategy of the universal player is now to assign $U$ according to $\alpha$. This will falsify the clause $\{v \mid v^\alpha = 0\} \cup \{\overline{v} \mid v^\alpha = 1\}$ and thus the whole QBF. Thus the strategy following $F$ is apparently a winning strategy. The non-constancy is also clear as the function $F$ is not constant: Suppose, there was a constant winning strategy and $\{l_1^u, \dots, l_b^u\}$ was its negation pattern on $\{u_1, \dots, u_b\}$ (i.e. $l_i^u = \overline{u_i}$ iff $u_i$ is assigned 0 in the strategy and $l_i^u = u_i$ else). A winning strategy always falsifies a clause, so for every possible assignment to the existential variables, there needs to be a clause containing the inverse negation pattern of this assignment and $\{\overline{l_1^u}, \dots, \overline{l_b^u}\}$. Since every clause is based on a $F$-satisfying assignment (by definition), we see that $F$ is constant, which violates the assumptions. ◀

There are $(2^b)^{(2^a)} - 2^b$ different non-constant functions with $a$ inputs and $b$ outputs. Each of them leads to an $F_{a,b}$-gadget. Such a gadget uses $2^a$ clauses, containing $a + b$ literals each.

For the construction of random formulas, we need multiple gadgets. A possible procedure to construct sequences of random gadgets is to set lower and upper bounds for $a, b$, for each $i \in [n]$ choose parameters $a_i, b_i$ randomly within the bounds and then obtain a $F_{a_i,b_i}$-gadget from a randomly chosen non-constant function $F : \{0,1\}^{a_i} \to \{0,1\}^{b_i}$ (repeating this process for each index $n \in \mathbb{N}$).

We also want to randomly choose the propositional base formulas. Each clause of a minimally unsatisfiable formula is critical, so we focus on generating minimally unsatisfiable formulas. A full characterization of minimally unsatisfiable 2-CNFs was recently given in [3] (see also [1, 2]). We can use this characterization to obtain the propositional part of our construction (thereby restricting ourselves to 2-CNFs). This includes the $\mathrm{IC}_n$ formulas (the implication chain formulas), but not the $\mathrm{SC}_n$ formulas (simple contradiction formulas).

The work [1] also describes a generation procedure for special minimally unsatisfiable formulas that are 2-CNFs with deficiency one (exactly one clause more than the number of variables). Using the approach described there with a small modification (allowing $C_1$ and $C_2$ to contain more than one literal) enables us to generate unsatisfiable deficiency one formulas (which are not necessarily 2-CNFs):

▶ **Lemma 21.** *Consider the following construction method:*
*Start with $F_0 := \{\bot\}$. Repeat the following steps for $i = 1, \dots, n$:*

- *Choose a clause $C \in F_{i-1}$ at random (set $C := \{\}$ if $F_{i-1} = \bot$).*
- *Choose $C_1$ and $C_2$ with $C_1 \cup C_2 = C$.*
- *Build $F_i = F_{i-1} \setminus \{C\} \cup \{C_1 \cup \{v\}\} \cup \{C_2 \cup \{\overline{v}\}\}$ for some $v \notin \mathsf{vars}(F_{i-1})$.*

*The formulas constructed according to this method are minimally unsatisfiable.*

The proof can be done by a simple induction. Now $\mathrm{SC}_n$ can be obtained in this way.

Combining random QBF gadgets (according to Lemma 20) with random minimally unsatisfiable formulas, we get random QBFs, which are hard for QU-Res by Theorem 12:

▶ **Proposition 22.** *Let $\Phi_n = (\phi_i)_{i\in[n]}$ be a sequence of random $(a_i, b_i)$-gadgets, $\chi_n$ a random minimally unsatisfiable formula with $n$ clauses and $T_n = \mathsf{vars}(\chi_n)$. Then any QU-Res refutation of $\chi_n^\Phi$ (constructed as in Theorem 12) has length at least $2^n$.*

Let us briefly compare our random QBFs with the hard random formulas presented in [11]. The formulas in [11] resemble our formulas, but with one major difference: the QBFs in [11] are only false and hard with high probability. In contrast, we construct QBFs that are always hard and false by design. The random formulas from [11] can be understood to be based on the SC formulas. To this they add a random construction that is akin to a QBF gadget, but only yields one with high probability. Note that in our construction here, we can choose both the propositional base formulas and the QBF gadgets randomly.

Finally, let us give a specific construction for random QBFs.

▶ **Example 23** (Random SC)**.** To keep the example as simple as possible, we again resort to the SC formulas. As we assemble the gadgets, we will set $a$ and $b$ fixed at $a = 2, b = 1$, instead of randomly choosing these parameters. Thus, all gadgets will be random $F_{1,2}$-gadgets. There are $2^4 - 2 = 16$ such gadgets (resp. functions) from which we can choose. We construct $\mathrm{SC}_n^{\mathrm{RG}}$ as follows: Let $(F_i)_{i\in[n]}$ be a sequence of randomly chosen non-constant functions $F_i : \{0,1\}^2 \to \{0,1\}$ for $i \in [n]$ and $\mathrm{RG}_n = (\mathrm{RG}_{2,1}^{F_i})_{i\in[n]}$ the sequence of the associated gadgets in variables $x_i^1, x_i^2$ and $u_i$ each, i.e. $\mathrm{RG}_{2,1}^{F_i} = \exists x_i^1 x_i^2 \forall u_i \cdot \varphi_i$. We build

$$\mathrm{SC}_n^{\mathrm{RG}} = \exists x_1^1 x_1^2 \dots x_n^1 x_n^2 \forall u_1 \dots u_n \exists t_1 \dots t_n \cdot \left( \bigcup_{i\in[n]} \{\varphi_i \times \{\overline{t_i}\}\} \right) \cup \{\{t_1, \dots, t_n\}\}.$$

These formulas have $4n$ clauses with four literals each (three from the gadget and one from a critical clause in $\mathrm{SC}_n$) and the additional clause with all the positive $t$ literals.

Their hardness follows directly from Proposition 22 and the construction of $\mathrm{SC}_n^{\mathrm{RG}}$:

▶ **Corollary 24.** *Any QU-Res refutation of $\mathrm{SC}_n^{\mathrm{RG}}$ has size at least $2^n$.*

## 4 Formulas Separating Q-Res and LD-Q-Res

We now prove that most of our constructed QBFs, including all the explicit examples and the random formulas, separate Q-Res and LD-Q-Res. This requires just one further natural condition, namely that the propositional base formulas have polynomial-size propositional resolution refutations and the QBF gadgets have polynomial-size Q-Res refutations.

In fact, instead of LD-Q-Res we can even use a weaker system, so-called reductionless LD-Q-Res [13,20,35], which is a strict fragment of LD-Q-Res [13]. This system allows merging as in LD-Q-Res but no universal reduction, i.e., any refutation ends with a purely universal clause. In other words, it includes LD-Q-Res refutations in which all universal reductions occur at the end of the derivation.

▶ **Theorem 25.** *For $n \in \mathbb{N}$ let $\Phi_n$ be sequences of QBF gadgets with polynomial-size Q-Res refutations and $\chi_n$ propositional formulas with polynomial-size resolution refutations. Let $\Phi_n = (\phi_i)_{i\in[n]} = (\exists X_i \forall U_i \cdot \varphi_i)_{i\in[n]}$ and $\chi_n = \mathcal{C} \cup \mathcal{D}$ with critical clauses $\mathcal{C} = \{C_1, \dots, C_n\}$, additional clauses $\mathcal{D}$, $T_n = \mathsf{vars}(\chi_n)$ and $\mathsf{vars}(\chi_n) \cap \left(\bigcup_{i\in[n]}\{X_i \cup U_i\}\right) = \varnothing$. Then $\chi_n^\Phi$ (as in Theorem 12) has polynomial-size refutations in reductionless LD-Q-Res.*

**Proof.** We consider the formula $\chi_n^\Phi$. Let $R_n$ be polynomial-size resolution refutations of $\chi_n$ and $S_1, \dots, S_n$ polynomial-size LD-Q-Res refutations[3] of the gadgets $\phi_1, \dots, \phi_n$. Let $S_i'$ be as $S_i$, but without the final universal reduction steps. Let $U_i^*$ be the set of (possibly merged) universal variables in the last clause of the resulting derivation. We can enlarge every clause in $S_i'$ by $C_i$ and get a derivation $S_i^*$ of $C_i \cup U_i^*$ from $\exists X_i \forall U_i \exists T_n \cdot \varphi_i \times C_i$. Now we can enlarge every $C_i$ in $R_n$ by $U_i^*$. This extension runs through the entire proof[4] and we obtain a reductionless LD-Q-Res derivation $R_n^*$ of $\bigcup_{i\in[n]} U_i^*$, which we can complete to a refutation by universal reduction. The composition of the proof is shown in Figure 3. ◀

By Theorem 12 (the formulas are hard for QU-Res) and Theorem 25 (which provides short LD-Q-Res refutations) the following holds:

▶ **Corollary 26.** *The formulas $\chi_n^\Phi$ from Theorem 25 separate QU-Res from (reductionless) LD-Q-Res.*

Note that all examples from Section 3.4 satisfy the required conditions and are therefore separating formulas. Furthermore the random formulas from Section 3.5 are based on either propositional 2-CNFs, which are known to have short resolution refutations, or a deficiency one formula constructed with the procedure described there, which at the same time provides a polynomial-size resolution refutation (viewed backwards, each step of the algorithm can be transformed into a resolution step with the newly introduced variable as a pivot). Thus all the random formulas separate QU-Res from reductionless LD-Q-Res.

For the next insight we need a result from [16]:

▶ **Theorem 27** ( [16]). *For any QBF $\phi$, if $\pi$ is a treelike P+$\forall$red proof of $\phi$ (where P is a propositional proof system), then $|\pi| \geq \rho(\phi)$ (where $\rho(\phi)$ is the strategy size from Definition 1).*

[3] Note that for $\Sigma_2^b$-formulas the systems Q-Res and LD-Q-Res are equivalent. A Q-Res refutation of such a formula is just a resolution refutation of the restriction of the formula to its existential variables with some reductions, which can be moved towards the beginning of the proof (since the universal block is rightmost). Allowing merging, we can move the reductions to the end without any problems.

[4] There can not be any conflicts in form of tautologous resolvents, since the $U_i^*$ are pairwise variable disjoint.

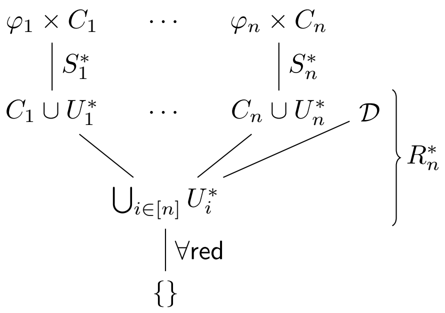


**Figure 3** Polynomial-size LD-Q-Res refutations for $\chi_n^\Phi$.

This implies that all the formulas we have constructed so far, including the random QBFs, are hard for all tree-like P+∀red systems.

▶ **Corollary 28.** *If $\chi_n^\Phi$ is a QBF as described in Theorem 12, then any refutation of $\chi_n^\Phi$ in treelike P+∀red systems has length at least $2^n$.*

This leads to an interesting fact:

▶ **Proposition 29.** *Treelike reductionless LD-Q-Res is not simulated by treelike QBF extended Frege systems (EF+∀red).*

**Proof.** The polynomial-size reductionless LD-Q-Res refutations shown in the proof of Theorem 25 are treelike, as long as the resolution refutation of the propositional formula and the reductionless LD-Q-Res refutation of the gadgets are (it is easy to find examples for both). Since EF+∀red is the extension of propositional extended Frege by universal reduction and all the formulas we constructed have exponential strategy size, the results immediately follow from Theorems 25 and 27. ◀

This is surprising because reductionless LD-Q-Res itself is not a very strong proof system; certainly the treelike variant is not either. Reductionless LD-Q-Res does not even simulate Q-Res (the two systems are in fact incomparable [35]). This is interesting to contrast with the recent simulation of LD-Q-Res (and even stronger systems) by QBF Frege [24]. The simulation there is quite non-trivial and highly dag-like. Proposition 29 above means that it cannot be strengthened to a tree-preserving simulation.

## 5 Construction of Separating Formulas between Q-Res and QU-Res

We now want to construct QBFs that separate Q-Res and QU-Res. As an intermediate step, we will build QBFs that are easy for Q-Res but have exponential strategy size. We will use the equality QBFs from the previous sections as running example, and, in fact, only change the prefix (and add some conditions on the underlying propositional formulas for the general case). We will then use such false QBFs with exponential strategy size and short Q-Res refutations to construct a large class of formulas to separate Q-Res from QU-Res.

### 5.1 Formulas with Exponential Strategy Size and Short Q-Res Refutations

First we will look at Example 14 from Section 3.4 and show how to obtain formulas from it that are easy for Q-Res but still have exponential strategy size. The key point here is the prefix – while we leave the matrix unchanged, we re-sort the $\Sigma_3^b$ prefix into an unrestricted

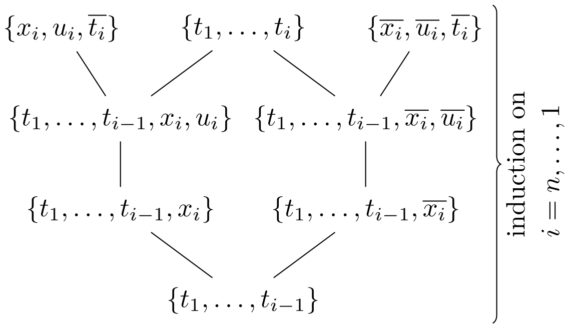

$\{x_i, u_i, \overline{t_i}\}$
$\{t_1, \dots, t_i\}$
$\{\overline{x_i}, \overline{u_i}, \overline{t_i}\}$
$\{t_1, \dots, t_{i-1}, x_i, u_i\}$
$\{t_1, \dots, t_{i-1}, \overline{x_i}, \overline{u_i}\}$
$\{t_1, \dots, t_{i-1}, x_i\}$
$\{t_1, \dots, t_{i-1}, \overline{x_i}\}$
$\{t_1, \dots, t_{i-1}\}$
induction on
$i = n, \dots, 1$

**Figure 4** Polynomial-size Q-Res refutation of ${}^{\mathrm{il}}\mathrm{SC}_n^{\mathrm{EQ}}$.

prefix. Roughly speaking, we do this by arranging the "crucial" variables of each critical clause into a separate existential block to the right of the variables of the associated gadget, and the remaining propositional variables into the leftmost existential block. In most of the examples already given, it is intuitively easy to identify the "crucial" variables of a clause; in the general case, this is somewhat more involved[5], as is to determine the appropriate order of the critical clauses (i.e., of their variables in the prefix), which is not arbitrary. We therefore only verify the desired properties for Example 14 from Section 3.4.

We start with the equality formulas. These were already modified in the desired way to the *interleaved equality formulas* [11], which have the same matrix as the equality formulas, but with an interleaved prefix (this also inspired our general construction). We denote the interleaved variant of a $\Sigma_3^b$-QBF $\chi_n^\Phi$ by ${}^{\mathrm{il}}\chi_n^\Phi$. We will give short Q-Res refutations.

▶ **Example 30** (Interleaved Equality [11])**.** We build ${}^{\mathrm{il}}\mathrm{SC}_n^{\mathrm{EQ}}$ from $\mathrm{SC}_n^{\mathrm{EQ}}$ by reordering the prefix in a natural way according to the indices:

$$\begin{aligned}\mathrm{SC}_n^{\mathrm{EQ}} &= \exists x_1 \dots x_n \forall u_1 \dots u_n \exists t_1, \dots, t_n \cdot \psi\\ {}^{\mathrm{il}}\mathrm{SC}_n^{\mathrm{EQ}} &= (\exists x_1 \forall u_1 \exists t_1) \dots (\exists x_n \forall u_n \exists t_n) \cdot \psi\\ \psi &= \bigcup_{i \in [n]} \{\{\overline{t_i}, x_i, u_i\}, \{\overline{t_i}, \overline{x_i}, \overline{u_i}\}\} \cup \{t_1, \dots, t_n\}.\end{aligned}$$

The Q-Res refutations are shown in Figure 4, they closely follow the resolution proof of SC. Note, that all the universal reductions in the refutation comply with the rules thanks to the variable order in the prefix.

It is readily verified that the interleaved formulas inherit exponential strategy size from their $\Sigma_3^b$ origins. While the winning strategies of the universal player are no longer unique for the interleaved formulas, the existential player can nevertheless continue to force a game that corresponds to the winning strategy of the associated $\Sigma_3^b$ formulas, i.e., $u_i = x_i$ for all $i \in [n]$. Thus, the interleaved formulas retain exponential strategy size.

Interleaved versions of the equivalence and XOR formulas can easily be build following this pattern and have the same properties.

Although we need the interleaved formulas mainly as a basis for separating Q-Res and QU-Res, they also have some noteworthy property, which follows from Theorem 27 together with the fact that all these formulas have exponential strategy size:

[5] They are in fact the pivots of certain resolution steps in special resolution refutations of the propositional formula.

▶ **Proposition 31.** *The formulas from Example 30 (and all other formulas with short Q-Res refutations and exponential strategy size) separate treelike from dag-like Q-Res.*

## 5.2 Separating Formulas

In the second step we will use the QBFs with short Q-Res refutations and exponential strategy size to construct separating formulas between Q-Res and QU-Res. Our method is inspired by the structure of the KBKF formulas [31]. We first define the concept of target clauses.

▶ **Definition 32** (Target Clauses). *For a false QBF $\phi = \mathcal{P} \cdot \varphi$ let $F$ be a set of clauses such that the existential player has a strategy to never lose on clauses from $\phi \setminus F$ in any assignment game (regardless of the strategy chosen by the universal player), i.e., the existential player will always lose on clauses in $F$. We call $F$ a set of* target clauses.

Notice that $F$ is in general not unique. It is always possible to choose $F = \varphi$. Based on this, the construction is remarkably simple:

▶ **Definition 33** (Tail Construction). *Let $\phi = \mathcal{P} \cdot \varphi$ be a false QBF with universal variables $\mathsf{vars}_\forall(\phi) = \{u_1, \ldots, u_n\}$ and $\{e_1, \ldots, e_n\} \cap \mathsf{vars}(\phi) = \varnothing$. Let further $F$ be a set of target clauses for $\phi$. Then we call*

$$\begin{aligned}\phi^* =& \mathcal{P}^* \cdot \varphi^* \\ =& \mathcal{P}\exists e_1 \ldots e_n \cdot \left( \bigcup_{C \in \varphi \setminus F} \{C\} \right) \cup \left( \bigcup_{C \in F} \{C \cup \{\overline{e_i} : i \in [n]\}\} \right) \cup \left( \bigcup_{i \in [n]} \{\{u_i, e_i\}, \{\overline{u_i}, e_i\}\} \right)\end{aligned}$$

*the* tailed version *$\phi^*$ of $\phi$.*

Although the choice of $F = \varphi$ will not significantly increase the size of the resulting formula, i.e., we always have $|\phi^*| = O(|\phi|)$, it makes sense to choose $F$ as small as possible. These tailed formulas have exactly the properties we aim for (if we choose a suitable $\phi$):

▶ **Theorem 34.** *Let $\phi^*_n$ be tailed versions of formulas $\phi_n$ as described in Definition 33, where $\phi_n$ requires super-polynomial strategy size, but has polynomial-size Q-Res refutations. Then $\phi^*_n$ separates Q-Res from QU-Res, i.e., $\phi^*_n$ requires super-polynomial size Q-Res refutations, but has polynomial-size QU-Res refutations.*

We will split the proof of Theorem 34 into two parts, first showing hardness for Q-Res of the constructed formula and afterwards constructing short QU-Res proofs.

To show this, we modify $\phi^*$ once more, similarly as described in [6] for the KBKF formulas. That is, we use new variables $v_1, \ldots, v_n$ and put them into the formula as copies of the universal variables $u_1, \ldots, u_n$. While Balabanov, Widl, and Jiang create $\forall u_i v_i$ from each $\forall u_i$ in the prefix, we group the universal copies in a (possibly additional) universal quantification block to the right of $\mathcal{P}$ (and to the left of the existential tail variables), similarly as in [11], i.e., $\mathcal{P}^* = \mathcal{P}\exists e_1 \ldots e_n$ becomes $\mathcal{P}' = \mathcal{P}\forall v_1 \ldots v_n \exists e_1 \ldots e_n$. In addition, the occurrences of $u_i$ in the matrix are effectively doubled, i.e., $\varphi'$ contains for each clause $C \in \varphi^*$ the extended clause $C \cup \{v_i : u_i \in C\} \cup \{\overline{v_i} : \overline{u_i} \in C\}$.

▶ **Definition 35** ($\phi'$). *For any QBF $\phi^* = \mathcal{P}^* \cdot \varphi^*$ constructed from a QBF $\phi = \mathcal{P} \cdot \varphi$ following Definition 33 we define*

$$\phi' = \mathcal{P}' \cdot \varphi' = \mathcal{P}\forall v_1 \ldots v_n \exists e_1 \ldots e_n \cdot \left( \bigcup_{C \in \varphi^*} C \cup \{v_i : u_i \in C\} \cup \{\overline{v_i} : \overline{u_i} \in C\} \right).$$

Moving the universal variable copies to the right into a common universal block can only shorten QU-Res refutations, since it might enable additional universal reductions, but can never block a reduction previously possible. We then use Theorem 3 to show that $\phi'$ requires long QU-Res proofs. To do so, we first show:

▶ **Lemma 36.** *Let $\phi^*$ be a QBF constructed from $\phi$ following Definition 33 and let $\phi'$ be as described in Definition 35. Then in the assignment game for $\phi'$ the existential player can force the universal player to*

**(i)** *follow a winning strategy for $\phi$ on $u_1, \ldots, u_n$ and*
**(ii)** *assign $v_i = u_i$ for every $i \in [n]$.*

**Proof.** We first show (i). Consider the assignment game on $\mathcal{P}$. If the universal player does not use a winning strategy on $\phi$, he will lose on $\phi$. Thus the assignment $\alpha$ constructed on $\mathcal{P}$ satisfies $\varphi$ and thus all the clauses $\bigcup_{C\in\phi} \{C \cup \{\overline{e_i} : i \in [n]\} \cup \{v_i \mid u_i \in C\} \cup \{\overline{v_i} \mid \overline{u_i} \in C\}\}$, because these are just weakenings of clauses from $\varphi$. The remaining clauses are $\bigcup_{i\in[n]} \{\{u_i, v_i, e_i\}, \{\overline{u_i}, \overline{v_i}, e_i\}\}$, which can easily be satisfied by $e_i = 1$ for $i \in [n]$. Hence the existential player wins the assignment game.

For (ii) again we consider the game on $\mathcal{P}$ and assume that the existential player plays according to his strategy on $\phi$ to only lose on clauses in $F$. Since $F$ is a target set, we know that such a strategy exists. Let $\alpha$ be the assignment constructed on $\mathcal{P}$ (by both the existential and the universal player). By definition of target clauses $\alpha$ does not falsify any clause $C \in \varphi \setminus \{F\}$; these are also part of $\phi^*$. $\alpha$ also satisfies the corresponding clauses in $\phi'$, which are $\{C \cup \{v_i \mid u_i \in C\} \cup \{\overline{v_i} \mid \overline{u_i} \in C\} \mid C \in \varphi \setminus \{F\}\}$. Thus, the remaining clauses are those resulting from $C \in F$, $\bigcup_{C\in F} \{C \cup \{\overline{e_i} : i \in [n]\} \cup \{v_i \mid u_i \in C\} \cup \{\overline{v_i} \mid \overline{u_i} \in C\}\}$ and the additional clauses $\bigcup_{i\in[n]} \{\{u_i, v_i, e_i\}, \{\overline{u_i}, \overline{v_i}, e_i\}\}$. Now assume towards a contradiction that the universal player assigns $v_j \neq u_j$ for some $j \in [n]$ (let $j$ be the first index for which this applies). Then the existential player can assign $e_j = 0$ without falsifying any of these clauses. This immediately satisfies every clause originating from a clause in $F$. All the clauses $\{u_j, v_j, e_j\}, \{\overline{u_j}, \overline{v_j}, e_j\}$ with $j < i$ are already satisfied and thus only the clauses $\{u_j, v_j, e_j\}, \{\overline{u_j}, \overline{v_j}, e_j\}$ with $j > i$ remain. But now the existential player can win the assignment game by simply assigning $e_j = 1$ for each $j > i$. ◀

▶ **Lemma 37.** *Let $\phi$, $\phi^*$, and $\phi'$ be as in Lemma 36. Then QU-Res proof size of $\phi'$ is at least $\rho(\phi)$.*

**Proof.** According to Lemma 36 the universal player has to assign $u_1, \ldots, u_n$ according to a $\phi$-strategy and $v_i = u_i$ for $i \in [n]$. Thus the cost of $\phi'$ is at least $\rho(\phi)$, because the whole strategy is pooled in the last universal block. Now we can use the cost/size argument (Theorem 3) and obtain that proof size of $\phi'$ in QU-Res is at least $\rho(\phi)$. ◀

We can now prove the lower bound for $\phi^*$, following an approach described in [11].

▶ **Lemma 38.** *Let $\phi^* = \mathcal{P}^* \cdot \varphi^*$ be a QBF constructed from $\phi = \mathcal{P} \cdot \varphi$ according to Definition 33. Then proof size of $\phi^*$ in Q-Res is at least $\frac{1}{2}\rho(\phi)$.*

**Proof sketch.** This follows from Lemma 37, since we can extend a Q-Res refutation for $\phi^*$ by doubling the reduction steps (reduce $v_i$ as soon as $u_i$ is reduced) to a Q-Res (or QU-Res) refutation for $\phi'$ of twice the size. Note that we have to start from a Q-Res proof and cannot do the same transformation with a QU-Res proof. ◀

Lemma 38 in combination with the conditions from Theorem 34 (i.e., exponential strategy size of $\phi_n$) implies Q-Res-hardness of $\phi_n^*$:

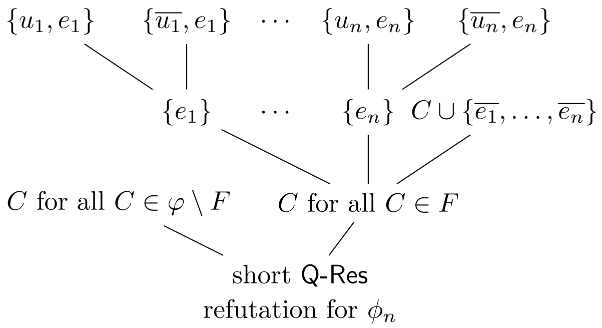

$\{u_1, e_1\}$
$\{\overline{u_1}, e_1\}$
$\cdots$
$\{u_n, e_n\}$
$\{\overline{u_n}, e_n\}$
$\{e_1\}$
$\cdots$
$\{e_n\}$
$C \cup \{\overline{e_1}, \dots, \overline{e_n}\}$
$C$ for all $C \in \varphi \setminus F$
$C$ for all $C \in F$
short Q-Res
refutation for $\phi_n$

**Figure 5** Polynomial-size QU-Res refutations for $\phi^*$.

▶ **Corollary 39** ($\phi^*_n$ is Hard for Q-Res). *Let $\phi^*_n$ be tailed versions constructed from $\phi_n$ following the rules and conditions from Theorem 34. Then $\phi^*_n$ is hard for Q-Res.*

Let us now prove the upper bound stated in Theorem 34:

▶ **Lemma 40** ($\phi^*_n$ has Short QU-Res Refutations). *If $\phi^*_n$ are QBFs constructed from $\phi_n$ following the rules and conditions from Theorem 34, then $\phi^*_n$ has short QU-Res refutations.*

**Proof.** $\phi_n = \mathcal{P} \cdot \varphi_n$ has by assumption short Q-Res proofs. $\phi^*_n$ additionally contains the clauses $\{u_i, e_i\}$ and $\{\overline{u_i}, e_i\}$ for all $i \in [n]$, from which we can get all the unit clauses $\{e_i\}, i \in [n]$ in only $n$ universal resolution steps (available in QU-Res). We then remove all the $\overline{e_i}$ literals from the clauses originated from $F$ in $|F| \cdot n$ resolution steps. Together with the unchanged clauses from $\varphi_n \setminus F$ we now have all clauses from $\varphi_n$ and can proceed with the short Q-Res refutation of $\phi_n$. The proof of $\phi_n$ is extended by $(|F| + 1) \cdot n \leq (|\varphi_n| + 1) \cdot n$ steps. Therefore we get a polynomial-size QU-Res refutation of $\phi^*_n$. The composition of the proof is shown in Figure 5. ◀

**Proof of Theorem 34.** The theorem follows from Corollary 39 and Lemma 40. ◀

## 5.3 Examples

We illustrate our construction on the interleaved equality formulas from [11], which we already discussed in Section 5.1:

▶ **Example 41** (Tailed Equality). We first need suitable formulas, on which we can use the tail construction:

$$\phi_n = (\exists x_1 \forall u_1 \exists t_1) \dots (\exists x_n \forall u_n \exists t_n) \cdot \left( \bigcup_{i \in [n]} \{\{x_i, u_i, \overline{t_i}\}, \{\overline{x_i}, \overline{u_i}, \overline{t_i}\}\} \right) \cup \{\{t_1, \dots, t_n\}\} .$$

As mentioned in Section 5.1, these are exactly the ${}^{\text{il}}\text{SC}^{\text{EQ}}_n$-formulas, i.e., they have exponential strategy size and short Q-Res refutations. Thus, $(\phi_i)_{i \in \mathbb{N}}$ meets the requirements for constructing separating formulas according to the above method. The existential player has a strategy to satisfy all clauses except for $\{x_n, u_n, \overline{t_n}\}$, $\{\overline{x_n}, \overline{u_n}, \overline{t_n}\}$ and $\{t_1, \dots, t_n\}$ in any game (by just setting $t_i = 0$ for $i < n$). With $u_n = x_n$ we get the following possible assignments:

- $x_n = u_n = 1, t_n = 1$ falsifies $\{\overline{x_n}, \overline{u_n}, \overline{t_n}\}$,
- $x_n = u_n = 0, t_n = 1$ falsifies $\{x_n, u_n, \overline{t_n}\}$ and
- $x_n = u_n, t_n = 0$ falsifies $\{t_1, \ldots, t_n\}$.

The remaining two clauses are satisfied in each case. Thus there are three possibilities for a minimal set $F$ of target clauses, containing one of these three clauses. The most intuitive choice for $F$ is $F = \{\{t_1, \ldots, t_n\}\}$. The tail construction then leads to the following formulas, separating Q-Res and QU-Res:

$$\begin{aligned}\phi_n^* =^{\mathrm{tl}} \mathrm{SC}_n^{\mathrm{EQ}} =& (\exists x_1 \forall u_1 \exists t_1) \ldots (\exists x_n \forall u_n \exists t_n) \exists e_1 \ldots e_n \cdot \\ & \left( \bigcup_{i \in [n]} \{\{x_i, u_i, \overline{t_i}\}, \{\overline{x_i}, \overline{u_i}, \overline{t_i}\}, \{u_i, e_i\}, \{\overline{u_i}, e_i\}\} \right) \\ & \cup \{\{t_1, \ldots, t_n, \overline{e_1}, \ldots, \overline{e_n}\}\}.\end{aligned}$$

Interestingly, the KBKF formulas [31] correspond to the tail construction (they actually inspired our construction).

## 6 Conclusion and Open Problems

While our construction of hard formulas in Section 3 yields a large class of hard QBFs, it does not allow to generate all hard QBFs. One apparent limitation is that we only produce $\Sigma_3^b$ formulas. While this is arguably the most interesting case, it would be worthwhile to explore systematically how to construct hard QBFs with higher quantifier complexity. While it is easy to derive such formulas from $\Sigma_3^b$ QBFs by just adding further dummy quantifiers, "more natural" constructions appear of interest.

A related question is which exact class of formulas can be generated by our construction. As we always import hardness via the size-cost method, one might aim for a construction that yields all such formulas. We do not achieve this yet, as one can even find $\Sigma_3^b$-formulas with high costs that do not stem from our method. Of course there are also further sources of hardness. E.g. the parity formulas [15] are hard for QU-Res, but have small cost. Finding general constructions for other QBF families, where hardness does not originate from cost, also appears interesting for future work.

## References

1 Hoda Abbasizanjani. *The combinatorics of minimal unsatisfiability: connecting to graph theory.* dissertation, Department of Computer Science, Swansea University, 2021.

2 Hoda Abbasizanjani and Oliver Kullmann. Minimal unsatisfiability and minimal strongly connected digraphs. In Olaf Beyersdorff and Christoph M. Wintersteiger, editors, *Theory and Applications of Satisfiability Testing (SAT)*, volume 10929 of *Lecture Notes in Computer Science*, pages 329–345. Springer, 2018.

3 Hoda Abbasizanjani and Oliver Kullmann. Classification of minimally unsatisfiable 2-cnfs. *CoRR*, abs/2003.03639, 2020. `arXiv:2003.03639`.

4 Albert Atserias, Johannes Klaus Fichte, and Marc Thurley. Clause-learning algorithms with many restarts and bounded-width resolution. *J. Artif. Intell. Res.*, 40:353–373, 2011.

5 Valeriy Balabanov and Jie-Hong R. Jiang. Unified QBF certification and its applications. *Formal Methods in System Design*, 41(1):45–65, 2012. `doi:10.1007/s10703-012-0152-6`.

6 Valeriy Balabanov, Magdalena Widl, and Jie-Hong R. Jiang. QBF resolution systems and their proof complexities. In *Proc. Theory and Applications of Satisfiability Testing (SAT)*, pages 154–169, 2014.

7 P. Beame and T. Pitassi. Simplified and improved resolution lower bounds. In *Proc. 37th IEEE Symposium on the Foundations of Computer Science*, pages 274–281. IEEE Computer Society Press, 1996.

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

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

10 Olaf Beyersdorff and Joshua Blinkhorn. A simple proof of QBF hardness. *Information Processing Letters*, 168, 2021.

11 Olaf Beyersdorff, Joshua Blinkhorn, and Luke Hinde. Size, cost, and capacity: A semantic technique for hard random QBFs. *Logical Methods in Computer Science*, 15(1), 2019.

12 Olaf Beyersdorff, Joshua Blinkhorn, and Meena Mahajan. Hardness characterisations and size-width lower bounds for QBF resolution. In *Proc. ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 209–223. ACM, 2020.

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

14 Olaf Beyersdorff and Benjamin Böhm. Understanding the Relative Strength of QBF CDCL Solvers and QBF Resolution. In *12th Innovations in Theoretical Computer Science Conference (ITCS 2021)*, volume 185 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 12:1–12:20, 2021.

15 Olaf Beyersdorff, Leroy Chew, and Mikolás Janota. New resolution-based QBF calculi and their proof complexity. *ACM Transactions on Computation Theory*, 11(4):26:1–26:42, 2019.

16 Olaf Beyersdorff and Luke Hinde. Characterising tree-like Frege proofs for QBF. *Inf. Comput.*, 268, 2019.

17 Olaf Beyersdorff, Luke Hinde, and Ján Pich. Reasons for hardness in QBF proof systems. *ACM Transactions on Computation Theory*, 12(2), 2020.

18 Olaf Beyersdorff, Mikolás Janota, Florian Lonsing, and Martina Seidl. Quantified boolean formulas. In Armin Biere, Marijn Heule, Hans van Maaren, and Toby Walsh, editors, *Handbook of Satisfiability*, Frontiers in Artificial Intelligence and Applications, pages 1177–1221. IOS Press, 2021.

19 Olaf Beyersdorff, Luca Pulina, Martina Seidl, and Ankit Shukla. Qbffam: A tool for generating QBF families from proof complexity. In Chu-Min Li and Felip Manyà, editors, *Theory and Applications of Satisfiability Testing (SAT)*, volume 12831 of *Lecture Notes in Computer Science*, pages 21–29. Springer, 2021.

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

21 Maria Luisa Bonet, Juan Luis Esteban, Nicola Galesi, and Jan Johannsen. On the relative complexity of resolution refinements and cutting planes proof systems. *SIAM J. Comput.*, 30(5):1462–1484, 2000. doi:10.1137/S0097539799352474.

22 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*, Frontiers in Artificial Intelligence and Applications, pages 233–350. IOS Press, 2021.

23 Leroy Chew. *QBF proof complexity*. PhD thesis, University of Leeds, Leeds, 2017.

24 Leroy Chew and Friedrich Slivovsky. Towards uniform certification in QBF. *Electron. Colloquium Comput. Complex.*, 2021. To appear at STACS 2022. URL: https://eccc.weizmann.ac.il/report/2021/144.

25 Judith Clymo. *Proof Complexity for Quantified Boolean Formulas*. PhD thesis, School of Computing, University of Leeds, 2021.

26 Stephen A. Cook and Phuong Nguyen. *Logical Foundations of Proof Complexity.* Cambridge University Press, 2010.

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

28 Uwe Egly, Florian Lonsing, and Magdalena Widl. Long-distance resolution: Proof generation and strategy extraction in search-based QBF solving. In *Proc. Logic for Programming, Artificial Intelligence, and Reasoning (LPAR)*, pages 291–308, 2013.

29 Amin Haken. The intractability of resolution. *Theoretical Computer Science*, 39:297–308, 1985.

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

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

32 Jan Krajíček. *Bounded Arithmetic, Propositional Logic, and Complexity Theory*, volume 60 of *Encyclopedia of Mathematics and Its Applications.* Cambridge University Press, Cambridge, 1995.

33 Jan Krajíček. *Proof complexity*, volume 170 of *Encyclopedia of Mathematics and Its Applications.* Cambridge University Press, 2019.

34 Florian Lonsing, Uwe Egly, and Allen Van Gelder. Efficient clause learning for quantified Boolean formulas via QBF pseudo unit propagation. In *Proc. International Conference on Theory and Applications of Satisfiability Testing (SAT)*, pages 100–115, 2013.

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

36 Knot Pipatsrisawat and Adnan Darwiche. On the power of clause-learning SAT solvers as resolution engines. *Artif. Intell.*, 175(2):512–525, 2011. doi:10.1016/j.artint.2010.10.002.

37 Nathan Segerlind. The complexity of propositional proofs. *Bulletin of Symbolic Logic*, 13(4):417–481, 2007.

38 M. Sipser. *Introduction to the Theory of Computation.* Course Technology, 2nd edition, February 2005.

39 Alasdair Urquhart. Hard examples for resolution. *J. ACM*, 34(1):209–219, 1987.

40 Allen Van Gelder. Contributions to the theory of practical quantified Boolean formula solving. In *Proc. Principles and Practice of Constraint Programming (CP)*, pages 647–663, 2012.

41 Lintao Zhang and Sharad Malik. Conflict driven learning in a quantified Boolean satisfiability solver. In *Proc. IEEE/ACM International Conference on Computer-aided Design (ICCAD)*, pages 442–449, 2002.