Classes of Hard Formulas for QBF Resolution

Agnes Schleitzer  
Olaf Beyersdorff

Friedrich-Schiller-Universität Jena,  
Fakultät für Mathematik und Informatik,  
Institut für Informatik,  
Ernst-Abbe-Platz 1, 07743 Jena, Deutschland

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 ($\Sigma^b_2$ 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.

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 (Cook & Reckhow, 1979) with many important connections to other fields, in particular to logic (Krajíček, 2019; Cook & Nguyen, 2010) and solving (Buss & Nordström, 2021). 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 (Buss & Nordström, 2021; Pipatsrisawat & Darwiche, 2011; Atserias, Fichte, & Thurley, 2011; Beame, Kautz, & Sabharwal, 2004) – there is a vast literature on hard formulas stemming from diverse areas such as combinatorics (e.g., Haken, 1985; Bonet, Esteban, Galesi, & Johannsen, 2000), graph theory (Urquhart, 1987), logic (Krajíček, 1995), random formulas (Beame & Pitassi, 1996), and many more (Krajíček, 2019; Segerlind, 2007).

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, Kleine Büning, Karpinski, & Flögel, 1995) and QU-Resolution (QU-Res, Van Gelder, 2012) 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.

©2023 The Authors. Published by AI Access Foundation under Creative Commons Attribution License CC BY 4.0.

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 in the QBF literature. The most prominent of these are arguably the KBKF formulas from the very first article that introduced Q-Res (Kleine Büning et al., 1995). The other ‘notorious’ QBF families are the equality formulas (Beyersdorff, Blinkhorn, & Hinde, 2019), the parity formulas (Beyersdorff, Chew, & Janota, 2019), and the CR formulas (Janota & Marques-Silva, 2015). 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 (Beyersdorff et al., 2019) 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 (Beyersdorff et al., 2019) (which actually inspired our construction), new circle, equivalence, and XOR formulas, as well as a large class of random QBFs.

---

1. A track of crafted formulas was introduced into QBFEval 2020 and a tool to generate the mentioned QBF families was presented by Beyersdorff, Pulina, Seidl, and Shukla (2021b).

All the mentioned QBF examples – KBKF, equality, and parity – are genuinely hard in this sense.
(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 (Beyersdorff et al., 2019; Egly, Lonsing, & Widl, 2013; Chew, 2017; Beyersdorff et al., 2019).

(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.

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 (Kleine Büning et al., 1995; Van Gelder, 2012; Beyersdorff & Blinkhorn, 2021). 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 (Clymo, 2021; Beyersdorff, Blinkhorn, & Mahajan, 2020).

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.
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 \( \text{vars}(l) = x \).

**QBFs.** A quantified Boolean formula (QBF) in closed prenex form \( \phi = P \cdot \varphi \) consists of a quantifier prefix \( P \) and a propositional formula \( \varphi \), called the matrix. The prefix is a series of quantifiers \( Q_i \in \{ \forall, \exists \} \), each followed by a set \( X_i \) of variables. No variable can be quantified twice, so \( X_i \cap X_j = \emptyset \) if \( i \neq j \). For a closed QBF (which we only consider here), \( P \) quantifies exactly the variables occurring in \( \varphi \). Thus, for \( P = Q_1 X_1 Q_2 X_2 \ldots Q_n X_n \), the matrix \( \varphi \) is a formula in variables \( \bigcup_{i \in [n]} X_i \) and we write \( \text{vars}(P \cdot \varphi) = \text{vars}(\varphi) = \bigcup_{i \in [n]} X_i \). As there are no free variables in a closed QBF, it is either true or false. We write \( \text{vars}_\exists(\varphi) \) for the set of existential variables in \( P \cdot \varphi \) and \( \text{vars}_\forall(\varphi) \) for those associated with \( \forall \). A QCNF is a QBF with a CNF matrix.

An assignment assigns truth values to variables. We sometimes represent an assignment as a set of pairs of variables and their associated (boolean) values. We denote by \( v^\alpha, l^\alpha \) the value of a variable \( v \) respective a literal \( l \) under an assignment \( \alpha \). We write \( \langle V \rangle \) for the set of all possible assignments to \( V \), \( \langle \chi \rangle = \langle \text{vars}(\chi) \rangle \) for the assignments of a propositional formula \( \chi \) and \( \langle \phi \rangle = \langle P \cdot \varphi \rangle = \langle \varphi \rangle \) for those of a QBF \( \phi = P \cdot \varphi \).

Closed QBFs can be viewed as a game between an existential and a universal player generating a total assignment (Sipser, 2005). 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 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. Beyersdorff et al., 2020). 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 Beyersdorff and Blinkhorn (2020):

**Definition 1** (Strategy Size \( \rho \); Beyersdorff & Blinkhorn, 2020). 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 a 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 (Kleine Buning et al., 1995) transfers Resolution from propositional logic to QBF. It uses the resolution rule (Q-Res) which only allows existential pivots and forbids
<table>
<thead>
<tr>
<th>Axiom</th>
<th>$C_{1} \cup {x} \cup {\overline{x}} \quad C_{2} \cup {\overline{x}}$</th>
<th>$C_{1} \cup C_{2}$ is non-tautologous; $x \in \text{vars}_3(\phi)$.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q-Res</td>
<td>$C_{1} \cup C_{2}$</td>
<td>$C_{1} \cup C_{2}$ is non-tautologous.</td>
</tr>
<tr>
<td>QU-Res</td>
<td>$C_{1} \cup C_{2} \cup {\overline{x}}$</td>
<td>$l^* = l \lor \overline{l}$, ${l^<em>} = {l, \overline{l}}$ for any literal $l$; $C_{1}^</em> = C_{1} \setminus (C_{1} \cap C_{2})$; $C_{2}^* = C_{2} \setminus (C_{1} \cap C_{2})$; $U^* = {u^* \mid u \in \text{vars}(C_{1} \cap C_{2})}$; $x \in \text{vars}<em>3(\phi)$; $C</em>{1} \cup C_{2}$ does not contain any existential tautologies; any $u \in \text{vars}(U^*)$ is quantified right of $x$ in $P$. $u \in \text{vars}_3(\phi)$ and quantified right of each existential variable in $C$ regarding $P$.</td>
</tr>
<tr>
<td>LDQ-Res</td>
<td>$C_{1} \cup C_{2} \cup {\overline{x}}$</td>
<td>$C_{1}^* \cup C_{2}^* \cup U^*$</td>
</tr>
<tr>
<td>∀Red</td>
<td>$\overline{u}$</td>
<td>$\overline{u}$</td>
</tr>
</tbody>
</table>

$C$ is a non-tautologous clause in the matrix $\varphi$.

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

The rules are given in Figure 2.

QU-Res (Van Gelder, 2012) 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 (Balabanov & Jiang, 2012) 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 introduced by Beyersdorff et al. (2019).

**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** (Beyersdorff et al., 2019). Let $\phi$ be a false QCNF. Then QU-Res refutations of $\phi$ have size at least $\text{cost}(\phi)$.

1459
The equality formulas from Beyersdorff et al. (2019) have exponential cost and are therefore hard for QU-Res:

**Definition 4** (Equality formulas; Beyersdorff et al., 2019). For $n \in \mathbb{N}$ we define the $n^{th}$ equality formula as

$$\text{EQ}_n = \exists x_1 \ldots x_n \forall u_1 \ldots u_n \exists t_1 \ldots t_n \cdot \left( \bigcup_{i \in [n]} \{ x_i, u_i, \overline{t_i} \}, \{ x_i, u_i, t_i \} \right) \cup \{ t_1, \ldots, t_n \}.$$ (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 $\{ t_1, \ldots, t_n \}$ and unit clauses $\{ \overline{t_1} \}, \ldots, \{ \overline{t_n} \}$, 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 $n^{th}$ 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.

We illustrate this idea in Figure 3.

### 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 \( C = \{ C_i \mid i \in [n] \} \) and by \( D = \{ D_i \mid i \in [\chi_n - n] \} \) 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). \( SC_n = \{ D_1 \} \cup \bigcup_{i \in [n]} \{ C_i \} \) with \( D_1 = \{ t_1, \ldots, t_n \} \) and \( C_i = \{ \overline{t_i} \} \) for \( i \in [n] \). Note that \( SC_n \) is minimally unsatisfiable.

In addition, we consider two further running examples.

**Example 7** (Implication Chain). \( IC_n = \bigcup_{i \in [n]} \{ C_i \} \) for \( n > 1 \) 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 \( D = \emptyset \).

**Example 8** (Equivalence Chain). \( EC_n = \left( \bigcup_{i \in [n]} \{ C_i, D_i \} \right) \cup \{ D_{n+1}, D_{n+2} \} \) 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 \( D \).

### 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 \( \Sigma \) \( \Sigma \) QBF \( \phi = \Phi \cdot \varphi \) 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 \( \Sigma \) \( \Sigma \) 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). The equality gadget forces the universal player to assign \( u = x \): \( EQ = \exists x \forall u \cdot \{ \{ x, u \}, \{ \overline{x}, \overline{u} \} \} \).

Note that the gadget is equivalent to \( \exists x \forall u \cdot x \equiv u \), 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 \( SC_n \), we could restate (1) as

\[
\exists x_1 \cdot x_n \forall u_1 \cdot u_n \exists t_1 \cdot t_n \cdot \left( \bigwedge_{i=1}^{n} ((x_i \leftrightarrow u_i) \rightarrow \overline{t_i}) \right) \land \left( \bigvee_{i=1}^{n} t_i \right). \tag{2}
\]

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 \( SC_n \).

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 \( \exists x^1 x^2 \forall u \cdot (x^1 \oplus x^2) \neq u \), which has the unique winning strategy \( u = x^1 \oplus x^2 \).
Example 11 (XOR Gadget). The XOR gadget forces the universal player to assign \( u = x_1 \oplus x_2 \): XOR = \( \exists x^1 \forall x^2 \forall u \cdot \{\{x^1, x^2, u\}, \{x^1, \overline{x^2}, \overline{u}\}, \{x^1, \overline{x^2}, u\}, \{x^1, x^2, \overline{u}\}\} \).

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 CNF \( \varphi \) and a clause \( C \) as \( \varphi \times C := \{D \cup C \mid D \in \varphi\} \).

We illustrate the construction in Figure 4.

Let us first show the following:

Lemma 13. 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 \( C = \{C_1, \ldots, C_n\} \) of critical clauses and a set \( D \) of remaining clauses. Set \( T_n = \text{vars}(\chi_n) \) and let \( \chi_n \) have no common variables with \( \bigcup_{i \in [n]} (X_i \cup U_i) \). Then

\[
\chi_n = \exists X_1 \ldots X_n \forall U_1 \ldots U_n \exists T_n \cdot \left( \bigcup_{i \in [n]} \{\varphi_i \times C_i\} \right) \cup D
\]

requires QU-Res refutations of size at least \( 2^n \).

Let us first show the following:

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. Obviously, \( \chi_n^\Phi \) is false: It is sufficient to combine the winning strategies of the gadgets (these are variable-disjoint and false). The existential player then has to satisfy the formula \( \chi_n \) by assigning the variables in \( T_n \), but he cannot succeed because \( \chi_n \) is unsatisfiable.

We now consider an arbitrary winning strategy \( S \) for \( \chi_n^\Phi \). We first argue that \( S \) must falsify each gadget: If it would satisfy the matrix \( \varphi_i \) of a gadget \( \phi_i \), it would also satisfy all clauses of \( \varphi_i \times C_i \) in \( \chi_n^\Phi \) stemming from \( \varphi_i \). This relieves the existential player from the burden of having to satisfy all the clauses in \( C \). By not satisfying \( C_i \) (because the concerned clauses are already satisfied), he can find a satisfying assignment for the remaining clauses in \( \chi_n \), since \( C_i \) is critical. Since all variables from \( \chi_n \) are quantified in the last block, the existential player can react accordingly. Thus, he succeeds in satisfying the matrix of \( \chi_n^\Phi \), which means that \( S \) is not a winning strategy.
So let us assume that $S$ falsifies the matrix of each gadget. Then $S$ contains a winning strategy for each gadget contained in $\chi^\Phi_n$, which, due to their variable disjointness, implies the claim of the lemma.

Proof of Theorem 12. We know from Lemma 13 that any winning strategy $S$ for $\chi^\Phi_n$ is composed of winning strategies for the single gadgets. As the $n$ gadgets in $\chi^\Phi_n$ do not have constant winning strategies and are variable disjoint, the combination of winning strategies must have range at least $2^n$, i.e., $\chi^\Phi_n$ 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 $\Sigma^b_3$ prefix, which is the result of using $\Sigma^b_3$ gadgets. The $\Sigma^b_3$ case is probably also the most natural setting as the size-cost technique from Theorem 3 essentially works for $\Sigma^b_3$ formulas. However, as mentioned, the restriction to $\Sigma^b_2$-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; Beyersdorff et al., 2019). The equality formulas (Definition 4) arise from applying equality gadgets to simple contradiction formulas: \( \text{EQ}_n = \text{SC}_{n}^{\text{EQ}} \). The construction is illustrated in Figure 5.

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

\[
\text{IC}_{n}^{\text{EQ}} = \exists x_1 \ldots x_n \forall u_1 \ldots u_n \exists t_0 \ldots t_{n-2} \cdot 
\left( \bigcup_{i=1}^{n-2} \{ \{ u_i, x_i, t_{i-1}, \overline{t_i} \}, \{ \overline{u_i}, x_i, t_{i-1}, t_i \} \} \right)
\cup \{ \{ u_{n-1}, x_{n-1}, \overline{t_0} \}, \{ \overline{u_{n-1}}, x_{n-1}, t_0 \}, \{ u_n, x_n, t_{n-2} \}, \{ \overline{u_n}, x_n, t_{n-2} \} \}.
\]

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

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

with clauses \( C_{i,1} = \{ x_i, u_i, t_{i-1}, \overline{t_i} \}, C_{i,2} = \{ \overline{x_i}, u_i, t_{i-1}, t_i \}, D_i = \{ \overline{t_{i-1}}, t_i \} \) 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:

\[
\text{SC}^\text{XOR}_n = \exists x_1 x_2 \ldots x_n \exists u_1 \ldots u_n \exists t_1 \ldots t_n.
\]

\[
\bigcup_{i \in [n]} \left\{ \{x_i, x_i^2, u_i, t_i\}, \{x_i, x_i^2, \overline{u_i}, \overline{t_i}\}, \{\overline{x_i}, x_i^2, u_i, \overline{t_i}\}, \{\overline{x_i}, x_i^2, \overline{u_i}, t_i\} \right\}
\]

\[
\cup \{t_1, \ldots, t_n\}.
\]

**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 \(X = \{x_1, \ldots, x_a\}, U = \{u_1, \ldots, u_b\}\) and a function \(F : \langle X \rangle \rightarrow \langle U \rangle\) we call an assignment \(\alpha \in \langle X \cup U \rangle\) F-satisfying iff \(F(x_1^\alpha \ldots x_a^\alpha) = u_1^\alpha \ldots u_b^\alpha\).

**Definition 19 (F\(_{a,b}\)-Gadget).** An \(F_{a,b}\)-gadget is built from a non-constant Boolean function \(F : \{0,1\}^a \rightarrow \{0,1\}^b\) as follows: We introduce sets of variables \(X = \{x_1, \ldots, x_a\}\) and \(U = \{u_1, \ldots, u_b\}\). 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:

\[
\text{RG}^F_{a,b} = \exists x_1 \ldots x_a \forall u_1 \ldots u_b \cdot \{\{v \mid v^\alpha = 0\} \cup \{\overline{v} \mid v^\alpha = 1\} \mid \alpha \text{ is } F\text{-satisfying}\}.
\]

We check that \(F_{a,b}\)-gadgets satisfy the required properties:

**Lemma 20.** Let \(\text{RG}^F_{a,b}\) be an \(F_{a,b}\)-gadget based on a Boolean function \(F : \{0,1\}^a \rightarrow \{0,1\}^b\) as described in Definition 19. Then \(\text{RG}^F_{a,b}\) 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., \(F(x_1^\alpha \ldots x_a^\alpha) = u_1^\alpha \ldots u_b^\alpha\). 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 \(\{l_1^u, \ldots, l_b^u\}\) was its negation pattern on \(\{u_1, \ldots, u_b\}\) (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 \(\{\overline{l_1^u}, \ldots, \overline{l_b^u}\}\). 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 \(F: \{0, 1\}^{a_i} \rightarrow \{0, 1\}^{b_i}\) (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 by Abbasizanjani and Kullmann (2020) (see also Abbasizanjani, 2021; Abbasizanjani & Kullmann, 2018). We can use this characterization to obtain the propositional part of our formulas. A full characterization of minimally unsatisfiable 2-CNFs was recently given by Abbasizanjani and Kullmann (2020) (see also Abbasizanjani, 2021; Abbasizanjani & Kullmann, 2018).

Abbasizanjani (2021) 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 \(i = 1, \ldots, n\):

- Choose a clause \(C \in F_{i-1}\) at random (set \(C := \emptyset\) 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 \(v \notin \text{vars}(F_{i-1})\).

The formulas constructed according to this method are minimally unsatisfiable.

**Proof.** We show this by induction: Clearly, \(F_0 = \{\bot\}\) is minimally unsatisfiable. No we consider \(F_{i+1}\). To get \(F_{i+1}\) from \(F_i\) we choose a new variable \(v\), a clause \(C \in F_i\) (or \(C = \emptyset\) for \(F_1\)) and a decomposition \(C_1 \cup C_2 = C\). Now we replace \(C\) with \(C_{1} \cup \{v\}\) and \(C_{2} \cup \{\overline{v}\}\). At this point it is very easy to modify a proof of resolution for \(F_i\) to one for \(F_{i+1}\): We just have to replace any axiom \(C\) by the resolution from \(C_1 \cup \{v\}\) and \(C_2 \cup \{\overline{v}\}\) to \(C\). Thus we already know that \(F_{i+1}\) is unsatisfiable.

Now, to show minimality, have a look at the single clauses. We distinguish two cases: Suppose first, we omit a clause \(D \in F_i \setminus \{C\}\) from \(F_{i+1}\). We know from induction that \(F_i\) is minimally unsatisfiable, thus \(F_i \setminus \{D\}\) is satisfiable. A satisfying assignment to \(F_i \setminus \{D\}\) satisfies \(C = C_1 \lor C_2\), i.e. it satisfies at least one of \(C_1\) and \(C_2\) resp. \(C_1 \lor \{v\}\) and \(C_2 \lor \{\overline{v}\}\). The second can easily be satisfied by extending the assignment to \(v\) (with the appropriate value). The resulting assignment satisfies \(F_{i+1} \setminus \{D\}\).

For the second case, suppose we omit w.l.o.g \(C_1 \cup \{v\}\) (the case of omitting \(C_2 \cup \{\overline{v}\}\) is analogous). We know by induction that there is a satisfying assignment to \(F_i \setminus \{C\}\). Extending this assignment by \(v = 0\) satisfies \(C_2 \lor \{\overline{v}\}\) and thus \(F_{i+1} \setminus \{C_1 \cup \{v\}\}\).

Now \(SC_n\) 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 \(\text{QU-Res}\) by Theorem 12:
**Proposition 22.** Let \( \Phi_n = (\phi_i)_{i \in [n]} \) be a sequence of variable disjoint random \((a_i, b_i)\)-gadgets, \( \chi_n \) a random minimally unsatisfiable formula with \( n \) clauses and \( T_n = \text{vars}(\chi_n) \). 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 by Beyersdorff et al. (2019). The formulas shown by Beyersdorff et al. resemble our formulas, but with one major difference: the QBFs from Beyersdorff et al. 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 Beyersdorff et al. 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 \( \text{SC}_n^{\text{RG}} \) 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 \( \text{RG}_n = (\text{RG}_{2,1}^F)_{i \in [n]} \) the sequence of the associated gadgets in variables \( x_i^1, x_i^2 \) and \( u_i \) each, i.e. \( \text{RG}_{2,1}^F = \exists x_i^1 x_i^2 \forall u_i \cdot \varphi_i \). We build

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

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

Their hardness follows directly from Proposition 22 and the construction of \( \text{SC}_n^{\text{RG}} \):

**Corollary 24.** Any QU-Res refutation of \( \text{SC}_n^{\text{RG}} \) 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 (Bjørner, Janota, & Klieber, 2015; Peitl, Slivovsky, & Szeider, 2019; Beyersdorff, Blinkhorn, & Mahajan, 2021), which is a strict fragment of LD-Q-Res (Beyersdorff et al., 2021). 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 variable disjoint 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 \( \chi_n = C \cup D \) with critical clauses...
\[
\begin{array}{cccc}
\varphi_1 \times C_1 & \cdots & \varphi_n \times C_n \\
S_1^* & \cdots & S_n^* \\
C_1 \cup U_1^* & \cdots & C_n \cup U_n^* \\
\bigcup_{i \in [n]} U_i^* \\
\forall \text{red} & \{\}
\end{array}
\]

Figure 6: Polynomial-size LD-Q-Res refutations for \(\chi_n^\Phi\).

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 shown by Beyersdorff and Hinde (2019):

**Theorem 27** (Beyersdorff & Hinde, 2019). For any QBF \(\phi\), if \(\pi\) is a treelike \(P+\forall\text{red}\) 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.
This implies that all the formulas we have constructed so far, including the random QBFs, are hard for all tree-like \(P^+\forall\text{red}\) systems.

**Corollary 28.** If \(\chi_n^\Phi\) is a QBF as described in Theorem 12, then any refutation of \(\chi_n^\Phi\) in treelike \(P^+\forall\text{red}\) systems has length at least \(2^n\).

This leads to an interesting fact:

**Proposition 29.** Treelike reductionless \(LD-Q-\text{Res}\) is not simulated by treelike QBF extended Frege systems \((EF^+\forall\text{red})\).

**Proof.** The polynomial-size reductionless \(LD-Q-\text{Res}\) refutations shown in the proof of Theorem 25 are treelike, as long as the resolution refutation of the propositional formula and the reductionless \(LD-Q-\text{Res}\) refutation of the gadgets are (it is easy to find examples for both). Since \(EF^+\forall\text{red}\) 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 \(LD-Q-\text{Res}\) itself is not a very strong proof system; certainly the treelike variant is not either. Reductionless \(LD-Q-\text{Res}\) does not even simulate \(Q-\text{Res}\) (the two systems are in fact incomparable, see Peitl et al., 2019). This is interesting to contrast with the recent simulation of \(LD-Q-\text{Res}\) (and even stronger systems) by extended QBF Frege (Chew & Slivovský, 2022). 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 \(Q-\text{Res}\) and \(QU-\text{Res}\)**

We now want to construct QBFs that separate \(Q-\text{Res}\) and \(QU-\text{Res}\). As an intermediate step, we will build QBFs that are easy for \(Q-\text{Res}\) but have exponential strategy size. In Section 5.1 we will use the examples from Section 3.4 to explain the construction, which, in fact, only changes the prefix in these examples. In Section 5.2 we formalize the general construction, adding some conditions on the underlying propositional formulas. In Section 5.3 we will then use such false QBFs with exponential strategy size and short \(Q-\text{Res}\) refutations to construct a large class of formulas separating \(Q-\text{Res}\) from \(QU-\text{Res}\).

5.1 **Some Formulas with Exponential Strategy Size and Short \(Q-\text{Res}\) Refutations**

First we will look at Examples 14, 16 and 17 from Section 3.4 and show how to obtain formulas from them that are easy for \(Q-\text{Res}\) but still have exponential strategy size. The key point here is the prefix – while we leave the matrix unchanged, we re-sort the \(\Sigma^3_3\) prefix into an unrestricted 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

\(^5\) They are in fact the pivots of certain resolution steps in special resolution refutations of the propositional formula.
the appropriate order of the critical clauses (i.e., of their variables in the prefix), which is not arbitrary. We therefore initially only verify the desired properties for Examples 14, 16 and 17 from Section 3.4, the general construction in all details follows in Section 5.2.

We start with the equality formulas. These were already modified in the desired way to the *interleaved equality formulas* (Beyersdorff et al., 2019), which have the same matrix as the equality formulas, but with an interleaved prefix (this also inspired our general construction). We adopt the name ‘interleaved’ also for our other examples and denote the interleaved variant of a \( \Sigma_3 \text{-QBF} \chi^\Phi_n \) by \( \text{il} \chi^\Phi_n \). We will give short Q-Res refutations for each example.

**Example 30** (Interleaved Equality; Beyersdorff et al., 2019). We build \( \text{il} \text{SC}^\text{EQ}_n \) from \( \text{SC}^\text{EQ}_n \) by reordering the prefix in a natural way according to the indices:

\[
\text{SC}^\text{EQ}_n = \exists x_1 \ldots x_n \forall u_1 \ldots u_n \exists t_1 \ldots t_n \cdot \psi \\
\text{il} \text{SC}^\text{EQ}_n = (\exists x_1 \forall u_1 \exists t_1) \ldots (\exists x_n \forall u_n \exists t_n) \cdot \psi \\
\psi = \bigcup_{i \in [n]} \{\{\overline{t_i}, x_i, u_i\}, \{\overline{t_i}, \overline{x_i}, \overline{u_i}\}\} \cup \{t_1, \ldots, t_n\}.
\]

The Q-Res refutation shown in Figure 7 follows closely the resolution proof of \( \text{SC}_n \).

**Example 31** (Interleaved Equivalence). The prefix of \( \text{il} \text{EC}^\text{EQ}_n \) equals the one of interleaved equality, additionally quantifying \( t_0 \) existentially in the leftmost block,

\[
\text{EC}^\text{EQ}_n = \exists x_1 \ldots x_n \forall u_1 \ldots u_n \exists t_0 \ldots t_n \cdot \psi \\
\text{il} \text{EC}^\text{EQ}_n = \exists t_0 (\exists x_1 \forall u_1 \exists t_1) \ldots (\exists x_n \forall u_n \exists t_n) \cdot \psi \\
\psi = \bigcup_{i \in [n]} \{C_{i,1}, C_{i,2}, D_i\} \cup \{D_{n+1}, D_{n+2}\}
\]

with clauses

\[
C_{i,1} = \{x_i, u_i, t_{i-1}, \overline{t_i}\} \\
C_{i,2} = \{\overline{x_i}, \overline{u_i}, t_{i-1}, \overline{t_i}\} \\
D_i = \{\overline{t_{i-1}}, t_i\} \quad i \in [n] \\
D_{n+1} = \{t_0, t_n\} \\
D_{n+2} = \{\overline{t_0}, \overline{t_n}\}.
\]
Again, the Q-Res refutation (see Figure 8) is structurally similar to the resolution proof for EC\(_n\), although it can be seen quite clearly that only one side of the proof increases by the refutations of the gadgets, due to the choice of the critical clauses.

We now consider using XOR gadgets:

**Example 32 (Interleaved XOR).** For \(\Sigma^{\pi}_{b} \text{SC}_{n}^{\pi XOR}\), the existential blocks in the prefix each comprise two existential variables, as specified by the XOR gadget. The matrix remains the same as for \(\Sigma^{\pi}_{b} \text{SC}_{n}^{\pi XOR}\):

\[
\text{SC}_{n}^{\pi XOR} = \exists x_1^1 x_1^2 \ldots x_n^1 x_n^2 \forall u_1 \ldots u_n \exists t_1 \ldots t_n \cdot \psi
\]

\[
\text{iiSC}_{n}^{\pi XOR} = (\exists x_1^1 x_1^2 \forall u_1 \exists t_1) \ldots (\exists x_n^1 x_n^2 \forall u_n \exists t_n) \cdot \psi
\]

\[
\psi = \bigcup_{i \in [n]} \left\{ \{x_i^1, x_i^2, u_i, t_i\}, \{\overline{x_i^1}, \overline{x_i^2}, u_i, \overline{t_i}\}, \{ \overline{x_i^1}, x_i^2, u_i, t_i\}, \{ \overline{x_i^1}, \overline{x_i^2}, u_i, \overline{t_i}\} \right\}
\]

The Q-Res refutations are made slightly more complex by the gadgets, but even here the structure of the resolution proof of SC shines through, as can be seen in Figure 9.

Note, that all the universal reductions in the Q-Res refutations shown in Figures 7 to 9 comply with the rules thanks to the variable order in the prefixes.

It is readily verified that the interleaved formulas inherit exponential strategy size from their \(\Sigma^{\pi}_{b}\) 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 \(\Sigma^{\pi}_{b}\) formulas, i.e., \(u_i = x_i\) for all \(i \in [n]\) in the case of equality gadgets and \(u_i = x_i^1 \oplus x_i^1\) for all \(i \in [n]\) in the case of XOR gadgets. Thus, the interleaved formulas retain exponential strategy size.
Note that the circle formulas $\text{IC}^\text{EQ}_n$ from Example 15 cannot be modified this way – there are not even enough propositional $t$ variables to build the prefix accordingly.\(^6\)

5.2 General Construction of Formulas as in Section 5.1

While we show in Section 5.1 that certain variants of the previously introduced examples satisfy the required conditions, in the following we will give a general construction for such formulas that are easy for Q-Res but have exponential strategy size. We will use the same ingredients as in Section 3. In fact, we only have to change the prefix and some requirements to the underlying propositional formulas and QBF gadgets. This approach is consistent with the relationship between the examples in Section 3.4 and those in Section 5.1 (e.g. the original equality formulas from Beyersdorff et al., 2019, and interleaved equality).

Building on this construction we will later carry out the full construction in Section 5.3 and thus find further separating formulas between Q-Res and QU-Res.

We recall the exponential strategy size from Section 3, and we will reuse the procedure described there, refining our requirements to the propositional base formula as well as to the QBF gadgets and reordering the prefix. To get short Q-Res refutations of the constructed formulas, in addition to gadgets with short proofs, of course we need to use propositional base formulas with short resolution refutations. In fact, the condition is more complex. In the following we denote by $C|_\alpha$ the clause $C$, where any literal $l \in C$ is replaced by the value $l^\alpha$ of $l$ under $\alpha$ (literals of variables not assigned by $\alpha$ remain unaffected). We call $C|_\alpha$ the restriction of $C$ by $\alpha$. We further define the restriction of a CNF $\chi$ by $\alpha$ as $\chi|_\alpha := \{C|_\alpha \mid C \in \chi\}$ and the restriction of an assignment $\alpha$ by a variable set $V$ as $\alpha|_V := \{(v,b) \mid v \in V \land (v,b) \in \alpha\}$. We denote by $R = (\{C,D\}, p)$ a resolution step with parent clauses $C$ and $D$ over the pivot variable $p$.

---

\(^6\) The modification becomes straightforward if we choose $D = \{(\overline{t_0}), \{t_0\}\}$ and $C_i = \{t_{i-1}, \overline{t_i}\}$ as clauses of $\text{IC}^\text{EQ}_n$ for $i \in [n]$ instead of the definition from Example 7 (note that the formula family remains the same, only the indices of the formulas shift and the partition in $C$- and $D$-clauses changes).
We define some terminology regarding (a suitable assignment for EC Example 36 that assigns \( \alpha \) shown in Example 34 and adopt the terminology used there. Consider as EC from \( C \) \( \pi \)-critical clauses and a refutation (a suitable refutation for EC Example 34 Definition 33 (suitable refutations). Let \( \chi \) be a propositional formula with a set \( C \) of critical clauses and a refutation \( \pi \) of \( \chi \). We call a resolution step in \( \pi \) involving a clause from \( C \) a \( C \)-step and we call \( \pi \) \( C \)-suitable, if it satisfies the following properties:

(i) For any \( C \in \pi \) there is exactly one \( C \)-step in \( \pi \) using \( C \) as axiom.

(ii) Every \( C \)-step resolves a clause from \( C \) with a clause from \( D = \chi \setminus C \) or a derived clause.

(iii) The pivots of the \( C \)-steps are pairwise different.

(iv) Any resolvent of a \( C \)-step contains no pivot which is used in an earlier \( C \)-step.

We define some terminology regarding \( \pi \): Let \( R_1, \ldots, R_n \) be the sequence of \( C \)-steps in \( \pi \) in reverse order and \( C_1, \ldots, C_n \) resp. \( t_1, \ldots, t_n \) the according sequences of parent clauses from \( C \) respective pivot variables. Let further \( T = \text{vars}(\chi) \) and \( T_i = T \setminus \{t_{i+1}, \ldots, t_n\} \), i.e. \( T_0 = T \setminus \{t_1, \ldots, t_n\} \) and \( T_i = T_{i-1} \cup \{t_i\} \) for \( i \in [n] \).

Example 34 (a suitable refutation for \( EC_n \)). Recall the formulas \( EC_n \) from Example 8:

\[
EC_n = \left( \bigcup_{i \in [n]} \{C_i, D_i\} \right) \cup \{D_{n+1}, D_{n+2}\} \text{ with } C_i = \{t_{i-1}, \overline{t_i}\}, \ D_i = \{\overline{t_{i-1}}, t_i\} \text{ for } i \in [n], \text{ and } D_{n+1} = \{t_0, t_n\}, \ D_{n+2} = \{\overline{t_0}, \overline{t_n}\}.
\]

Figure 10 shows a suitable refutation \( \pi \) for \( EC_n \). \( R_1, \ldots, R_n \) is the sequence of \( C \)-steps (in reverse order), \( C_1, \ldots, C_n \) and \( t_1, \ldots, t_n \) are the corresponding sequences of \( C \)-parent clauses and pivot variables, respectively. Accordingly, \( T = \{t_0, \ldots, t_n\} \), \( T_0 = \{t_0\} \) and \( T_i = \{t_0, \ldots, t_i\} \) for \( i \in [n] \). Let us briefly review the individual elements of the definition:

(i) Since the derivation contains exactly one outgoing edge per axiom and there is no resolution step with two parent clauses from \( C \), this condition is obviously satisfied.

(ii) The first \( C \)-step resolves \( C_n \in C \) with \( D_{n+1} \in D \), any other \( C \)-step resolves a clause from \( C \) with a derived clause.

(iii) The pivot of step \( R_i \) is \( t_i \).

(iv) The resolvent of step \( R_i \) is \( \{t_0, t_{i-1}\} \), so it obviously does not contain any \( t_j \) with \( j > i \) (which are the pivots of earlier \( C \)-steps).

Definition 35 (suitable assignments). Now let \( \chi \) be a propositional formula, \( C \) a set of critical clauses and \( \pi \) a \( C \)-suitable resolution refutation with \( T \), \( T_i \) as described above.

Let \( \alpha \in \langle T \rangle \) be an assignment to \( T \) and \( \alpha_i := \alpha \upharpoonright T_i \) for \( i \in [0, n] \). We call \( \alpha \ \pi \)-suitable, if \( C_i \upharpoonright \alpha_{i-1} \) is critical in \( \chi \upharpoonright \alpha_{i-1} \) for any \( i \in [n] \).

Example 36 (a suitable assignment for \( EC_n \)). We build on the suitable refutation for \( EC_n \) shown in Example 34 and adopt the terminology used there. Consider as \( \alpha \) the assignment that assigns 0 to any \( t_i \), \( i \in [0, n] \) and let \( \alpha_i, i \in [0, n] \) be as described in Definition 35. We
will show that $\alpha$ is $\pi$-suitable (for $\pi$ from Example 34). Let us look at the corresponding restrictions of the formula:

$$\text{EC}_n \upharpoonright_{\alpha_{i-1}} = \{\{t_i\}, \{t_n\}\} \cup \{\{t_j-1, t_j\} \mid j \in [i+1, n]\} \cup \{\{\overline{t_j-1}, t_j\} \mid j \in [i+1, n]\}$$

for $i \in [n]$, where $C_i \upharpoonright_{\alpha_{i-1}} = \{\overline{t_i}\}$ should be critical. To verify the property of criticality, it is sufficient to check the satisfiability of $\text{EC}_n \upharpoonright_{\alpha_{i-1}} \setminus \{C_i \upharpoonright_{\alpha_{i-1}}\}$:

$$\text{EC}_n \upharpoonright_{\alpha_{i-1}} \setminus \{C_i \upharpoonright_{\alpha_{i-1}}\} = \text{EC}_n \upharpoonright_{\alpha_{i-1}} \setminus \{\overline{t_i}\}$$

$$= \{\{t_n\}\} \cup \{\{t_j-1, \overline{t_j}\} \mid j \in [i+1, n]\} \cup \{\{t_{j-1}, t_j\} \mid j \in [i+1, n]\}.$$ 

It is easy to see that assigning 1 to all $t_j, j \in [i, n]$ leads to a satisfying assignment for this formula, since all clauses contain a positive literal. Thus, $C_i \upharpoonright_{\alpha_{i-1}}$ is critical in $\text{EC}_n \upharpoonright_{\alpha_{i-1}}$ and $\alpha$ is a $\pi$-suitable assignment as desired.

**Definition 37** ($\overline{\text{id}}\chi^\Phi$). Let $\chi$ be a propositional formula with a set $C$ of $n$ critical clauses, $D = \chi \setminus C$ and a $C$-suitable refutation $\pi$. Let $t_1, \ldots, t_n$ and $C_1, \ldots, C_n$ be the sequences of pivot variables and $C$-parent clauses of $C$-steps in $\pi$ (in reverse order) and $T_0 = \text{vars}(\chi) \setminus \{t_1, \ldots, t_n\}$ as described in Definition 33. Let further $\Phi_n = (\phi_i)_{i \in [n]} = (P_i \cdot \varphi_i)_{i \in [n]}$ be a sequence of variable disjoint QBF gadgets. We define

$$\overline{\text{id}}\chi^\Phi = \exists_{T_0}(P_1 \exists t_1) \ldots (P_n \exists t_n) \cup \bigcup_{i \in [n]} [\varphi_i \times C_i] \cup D.$$
Figure 11: Construction principle of interleaved formulas from unsatisfiable propositional formulas with critical clauses $\mathcal{C}$ and a $\mathcal{C}$-suitable refutation $\pi$. 

$\chi_n$: $D \ C \ D \ C \ D \ \cdots \ D \ C \ D$

$n$ Critical Clauses within an unsatisfiable Propositional Formula in variables $T$

QBF-Gadgets with variable disjoint prefixes $\exists X_i \forall U_i$

Other Clauses

QBF-Gadgets with variable disjoint prefixes $\exists X_i \forall U_i$

Critical Clauses

$\vdash \begin{array}{l}
\exists T_0 \exists X_1 \forall U_1 \exists t_1 \cdots \exists X_n \forall U_n \exists t_n : \\
\bigwedge_{i=1}^{n} \left( \neg \varphi_i \rightarrow C_i \right) \wedge \mathcal{D}
\end{array}$

$\equiv \begin{array}{l}
\exists T_0 \exists X_1 \forall U_1 \exists t_1 \cdots \exists X_n \forall U_n \exists t_n : \\
\bigwedge_{i=1}^{n} \left( \varphi_i \times C_i \right) \wedge \mathcal{D}
\end{array}$
The construction principle of these interleaved formulas is illustrated in Figure 11.

**Lemma 38.** For \( n \in \mathbb{N} \) let \( \Phi_n \) be a sequence of \( n \) variable disjoint QBF gadgets with polynomial-size Q-Res refutations and \( \chi_n \) a propositional formula with polynomial-size resolution refutations. Let \( \mathcal{C} \) with \( |\mathcal{C}| = n \) be a set of critical clauses for \( \chi_n \) and \( \pi \) a short \( \mathcal{C} \)-suitable resolution refutation of \( \chi_n \). Then \( \exists^{\Phi_n} \chi_n, \ n \in \mathbb{N} \) has polynomial-size Q-Res refutations.

**Proof.** Let \( \Phi_n = (\phi_i)_{i \in [n]} = (P_i \cdot \varphi_i)_{i \in [n]} \) be a sequence of variable disjoint QBF gadgets with polynomial-size Q-Res refutations, \( \chi_n \) a propositional formula with polynomial-size resolution refutations where \( \text{vars}(\chi_n) \cup \text{vars}(\Phi_n) = \emptyset \) and \( \mathcal{C}, \mathcal{D}, \pi \) as described above. Let \( C_1, \ldots, C_n \) and \( t_1, \ldots, t_n \) be the sequences of axioms and pivots as described in Definition 33. We consider the \( \mathcal{C} \)-steps performed in \( \pi \) and show, that the resolvents can be derived in only a few more steps using axioms from \( \exists^{\Phi_n} \chi_n \) and Q-Res.

Let \( C_i \) be an axiom from \( \mathcal{C} \) and \( D \) an axiom from \( \mathcal{D} \) or a derived clause, where \( C_i \) and \( D \) are resolved with each other in \( \pi \) to the resolvent \( E \). \( t_i \) is the pivot element to this resolution step. \( \exists^{\Phi_n} \chi_n \) contains \( \varphi_i \times C_i \) instead of \( C_i \). By first resolving all clauses of \( \varphi_i \times C_i \) with \( D \), we obtain \( \varphi_i \times E \), thereby eliminating the pivot \( t_i \). Since \( \phi_i \) and \( \chi_n \) are variable disjoint and all the \( T \)-variables are existential, this is easily possible. Since \( \pi \) is \( \mathcal{C} \)-suitable, \( E \) does not contain any variable \( t_j \) with \( j > i \) (thanks to point (iv) of Definition 33). Now we can use the short refutation of \( \varphi_i \) by extending each clause in it by \( E \). Since \( \varphi_i \times E \) only contains variables from \( P_i \) and \( T \), reduction steps within the derivation could – corresponding to the prefix – only be blocked by variables \( t_j \), \( j \geq i \). However, these are not contained in \( \varphi_i \times E \). So at the end of the derivation we get the clause \( E \) instead of the empty clause – as desired.

Figure 12 illustrates the procedure using an equality gadget \( \text{EQ} = P_i \cdot \varphi_i = \exists x_i \forall u_i \cdot \{\{x_i, u_i\}, \{\overline{x}_i, \overline{u}_i\}\} \).

In this way we can replace all resolution steps that use an axiom from \( \mathcal{C} \). We get the same resolvents with only a few more steps (since the \( \varphi_i \) have short Q-Res refutations) and can connect the rest of \( \pi \). Overall, we get a Q-Res refutation for \( \exists^{\Phi_n} \chi_n \) of the same order of magnitude (as \( \pi \)). This method can be found in the Q-Res refutations of all examples from Section 5.1.

**Lemma 39.** For \( n \in \mathbb{N} \) let \( \Phi_n \) be a sequence of \( n \) variable disjoint QBF gadgets and \( \chi_n \) a propositional formula. Let \( \mathcal{C} \) with \( |\mathcal{C}| = n \) be a set of critical clauses for \( \chi_n \), \( \pi \) a short
$C$-suitable resolution refutation of $\chi_n$ and $\alpha$ a $\pi$-suitable assignment to the variables in $T = \text{vars}(\chi_n)$. Then $^{il}_T\Phi^n$, $n \in \mathbb{N}$ has exponential strategy size.

Proof. We can use a similar argumentation as in Theorem 12 to show that any winning strategy for $^{il}_T\Phi^n$ is based on a combination of winning strategies for the $\phi_i$ formulas (but we have to take into account that the prefix does not collect the $T$-variables at the end and therefore need the $\pi$-suitability of $\alpha$).

Let $\Phi_n = (\phi_i)_{i \in [n]} = (P_i \cdot \varphi_i)_{i \in [n]}$ be a sequence of variable disjoint QBF gadgets, $\chi_n$ a propositional formula and $C$, $D$, $\pi$, $\alpha$ as described above. Let further $T$, $T_i$ for $i \in [0, n]$ be the variable sets and $\alpha_i$ for $i \in [0, n]$ the restrictions of $\alpha$ as introduced in Definitions 33 and 35. It is obvious, that assigning $u_i$ according to a winning strategy for $\phi_i$ for each $i \in [n]$ is a universal winning strategy on $^{il}_T\Phi^n$ with exponential size (since the gadgets are non-constant). We assume (for contradiction), there is a winning strategy $S$ assigning $u_i$ different from any winning strategy for $\phi_i$ for some $i \in [n]$ (we consider the smallest $i$ with this property). Then all clauses from $\varphi_i \times C_i$ for some $i \in [n]$ (we consider the smallest $i$ with this property). Then all clauses from $\varphi_i \times C_i$ for some $i \in [n]$ are satisfied (since $\varphi$ is satisfied). We assume that the existential player has followed $\alpha_i$ on $T_0$ and $t_1, \ldots, t_{i-1}$. But since $\alpha$ is $\pi$-suitable, we know $C_i |_{\alpha_{i-1}}$ is critical in $\chi_n |_{\alpha_{i-1}}$. That means $\chi_n |_{\alpha_{i-1}} \setminus \{C_i |_{\alpha_{i-1}}\}$ is satisfiable with some assignment $\alpha'$ to the remaining variables $t_i, \ldots, t_n$. Since the clauses $\varphi_i \times C_i$ are already satisfied by the universal assignment, the existential player wins the assignment game with $\alpha'$ and an arbitrary assignment to the remaining existential variables. Thus $S$ is not a winning strategy.

The formulas for simple contradiction and equivalence chain from Section 3 satisfy the required properties, where the naming of the clauses identifies $C$ and $D$ and the related $\pi$ and $\alpha$ should be obvious. While interleaved equality formulas (Beyersdorff et al., 2019) are an instantiation of $^{il}_T\Phi^n$ formulas already known from literature, we present some new examples in Section 5.1.

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

**Theorem 40.** For $n \in \mathbb{N}$ let $\Phi_n$ be a sequence of $n$ variable disjoint QBF gadgets with polynomial-size $\text{Q-Res}$ refutations and $\chi_n$ a propositional formula with polynomial-size resolution refutations. Let $C$ with $|C| = n$ be a set of critical clauses for $\chi_n$ and $\pi$ a short $C$-suitable resolution refutation of $\chi_n$. Let additionally a $\pi$-suitable assignment to the variables in $T = \text{vars}(\chi_n)$ exist. Then $^{il}_T\Phi^n$, $n \in \mathbb{N}$ separate tree-like from dag-like $\text{Q-Res}$.

Proof. This follows immediately from Lemmas 38 and 39 and Theorem 27.

The formulas from Examples 30 to 32 shown in Section 5.1 satisfy the conditions of Theorem 40, thus they separate treelike from dag-like $\text{Q-Res}$.

\[7. \text{Note that the formulas for implication chain (IC_n) can’t be used within this pattern. As already mentioned in Section 5.1 this is due to the indexing policy - by simply shifting the index by two and renaming the clauses of index } n \text{ to } D = \{(T_0, \{t_n\}\} \text{ and } C_i = \{t_{i-1}, T_i\} \text{ for } i \in [n] \text{ they get usable.} \]
5.3 Separating Formulas

In a 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 (Kleine Büning et al., 1995). We first define the concept of target clauses.

**Definition 41 (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 (it is illustrated in Figure 13):

**Definition 42 (Tail Construction).** Let \( \phi = \mathcal{P} \cdot \varphi \) be a false QBF with universal variables \( \text{vars}_\varphi(\phi) = \{u_1, \ldots, u_n\} \) and \( \psi \) an unsatisfiable propositional formula with \( n + 1 \) critical clauses \( C = \{C_1, \ldots, C_{n+1}\} \), \( D = \psi \setminus C \) and variables \( \text{vars}(\psi) = \{e_1, \ldots, e_m\} \) where \( \text{vars}(\phi) \cap \text{vars}(\psi) = \emptyset \). Let further \( F \) be a set of target clauses for \( \phi \). Then we call

\[
\varphi_{\psi}^\text{tail} = \mathcal{P}^\text{tail} \cdot \varphi_{\psi}^\text{tail} = \mathcal{P} \exists e_1 \ldots e_m \cdot (\varphi \setminus F) \cup (F \times C_{n+1}) \cup \left( \bigcup_{i \in [n]} \{u_i\}, \{\pi_i\} \times C_i \right) \cup D
\]
the \( \psi \)-tailed version \( \phi_{\psi}^{\text{tail}} \) of \( \phi \).

Although the choice of \( F = \varphi \) will not significantly increase the size of the resulting formula, i.e., we always have \( |\phi_{\psi}^{\text{tail}}| \in O(|\phi|) \), 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 43.** For \( n \in \mathbb{N} \) let \( \phi_{\psi,n}^{\text{tail}} \) be tailed versions of formulas \( \phi_n \) as described in Definition 42, where \( \phi_n \) requires super-polynomial strategy size, but has polynomial-size \( Q\text{-Res} \) refutations and \( \psi_n \) has polynomial-size resolution refutations. Then \( \phi_{\psi,n}^{\text{tail}} \) separates \( Q\text{-Res} \), i.e., \( \phi_{\psi,n}^{\text{tail}} \) requires super-polynomial size \( Q\text{-Res} \) refutations, but has polynomial-size \( Q\text{-Res} \) refutations.

We will split the proof of Theorem 43 into two parts, first showing hardness for \( Q\text{-Res} \) of the constructed formulas and afterwards constructing short \( Q\text{U}-\text{Res} \) proofs.

To show hardness of \( \phi_{\psi,n}^{\text{tail}} \) for \( Q\text{-Res} \), we modify \( \phi_{\psi,n}^{\text{tail}} \) once more, similarly as described by Balabanov, Widl, and Jiang (2014) 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 et al. (2014) 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 shown by Beyersdorff et al. (2019), i.e., \( \mathcal{P}_{\psi}^{\text{tail}} = \mathcal{P} \exists e_1 \ldots e_m \) becomes \( \mathcal{P}_{\psi}^{\text{double}} = \mathcal{P} \forall v_1 \ldots v_n \exists e_1 \ldots e_m \). In addition, the occurrences of \( u_i \) in the matrix are effectively doubled, i.e., \( \varphi_{\psi}^{\text{double}} \) contains for each clause \( C \in \varphi_{\psi}^{\text{tail}} \) the extended clause \( \text{dupl} (C) := C \cup \{ v_i : u_i \in C \} \cup \{ \overline{v_i} : \overline{u_i} \in C \} \). We extend the \( \text{dupl} () \) function to clause sets \( S \) via \( \text{dupl} (S) := \{ \text{dupl} (C) | C \in S \} \).

**Definition 44 \( (\varphi_{\psi}^{\text{double}}) \).** For any QBF \( \phi_{\psi}^{\text{tail}} = \mathcal{P}_{\psi}^{\text{tail}} \cdot \varphi_{\psi}^{\text{tail}} \) constructed from a QBF \( \phi = \mathcal{P} \cdot \varphi \) following Definition 42 we define

\[
\varphi_{\psi}^{\text{double}} = \mathcal{P}_{\psi}^{\text{double}} \cdot \varphi_{\psi}^{\text{double}} = \mathcal{P} \forall v_1 \ldots v_n \exists e_1 \ldots e_m \cdot \text{dupl} \left( \varphi_{\psi}^{\text{tail}} \right)
= \mathcal{P} \forall v_1 \ldots v_n \exists e_1 \ldots e_m \cdot \text{dupl} \left( \varphi \setminus F \right) \cup \text{dupl} \left( F \times C_{n+1} \right)
\cup \left( \bigcup_{i \in [n]} \{ u_i, v_i \} \times C_i \right) \cup \mathcal{D}.
\]

Moving the universal variable copies to the right into a common universal block can only shorten \( Q\text{U}-\text{Res} \) refutations, since it might enable additional universal reductions, but cannot block a reduction previously possible. We then use Theorem 3 to show that \( \varphi_{\psi}^{\text{double}} \) requires long \( Q\text{U}-\text{Res} \) proofs. To do so, we first show:

**Lemma 45.** Let \( \phi_{\psi}^{\text{tail}} \) be a QBF constructed from \( \phi \) and \( \psi \) following Definition 42 and let \( \varphi_{\psi}^{\text{double}} \) be as described in Definition 44. Then in the assignment game for \( \varphi_{\psi}^{\text{double}} \) 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 \( i \in [n] \).

1479
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 $\mathcal{P}$ will satisfy $\varphi$ and thus all the clauses in $\text{dupl}(\varphi \setminus F)$ and $\text{dupl}(F \times C_{n+1})$, because these are just weakenings of clauses from $\varphi$. The remaining clauses are $\bigcup_{i \in [n]} \{\{u_i, v_i\}, \{\overline{w_i}, \overline{v_i}\}\} \times C_i$ and $D$. Since $C_1, \ldots, C_{n+1}$ are critical clauses in $\psi$, $\psi \setminus \{C_{n+1}\} = \{C_i \mid i \in [n]\} \cup D$ is satisfiable. All variables occurring in this clause set are quantified existentially in the last (i.e. right most) block. Therefore, the existential player can choose an assignment to this variables, which satisfies $\psi \setminus \{C_{n+1}\}$ and thus as well $\left(\bigcup_{i \in [n]} \{\{u_i, v_i\}, \{\overline{w_i}, \overline{v_i}\}\} \times C_i \right) \cup D$. This immediately makes the existential player the winner.

For (ii) again we consider the game on $\mathcal{P}$. Now we 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 $\varphi^{\text{tail}}$. $\alpha$ also satisfies the corresponding clauses $\text{dupl}(\varphi \setminus F)$ in $\phi^{\text{double}}$. Thus, the remaining clauses are $\text{dupl}(F \times C_{n+1})$, the additional clauses $\bigcup_{i \in [n]} \{\{u_i, v_i\}, \{\overline{w_i}, \overline{v_i}\}\} \times C_i$ and $D$. Now assume towards a contradiction that the universal player assigns $v_j \neq u_j$ for some $j \in [n]$. Since $C_1, \ldots, C_{n-1}$ are critical clauses of $\psi$, there is an assignment to $\{e_1, \ldots, e_n\}$ that satisfies $\psi \setminus \{C_j\}$. Let the existential player choose this assignment. Then $C_j$ is falsified, but since $u_j \neq v_j$ each of the clauses $C_j \cup \{u_j, v_j\}$, $C_j \cup \{\overline{u_j}, \overline{v_j}\}$ is still satisfied, as is any other clause mentioned above. \hfill $\square$

Lemma 46. Let $\phi$, $\psi$, $\phi^{\text{tail}}$, and $\phi^{\text{double}}$ be as in Lemma 45. Then QU-Res proof size of $\phi^{\text{double}}$ is at least $\rho(\phi)$.

Proof. According to Lemma 45 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 $\phi^{\text{double}}$ 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 $\phi^{\text{double}}$ in QU-Res is at least $\rho(\phi)$. \hfill $\square$

We can now prove the lower bound for $\phi^{\text{tail}}$, following an approach described by Beyersdorff et al. (2019).

Lemma 47. Let $\phi^{\text{tail}} = P^{\text{tail}} \cdot \varphi^{\text{tail}}$ be a QBF constructed from $\phi = P \cdot \varphi$ and $\psi$ according to Definition 42. Then proof size of $\phi^{\text{tail}}$ in Q-Res is at least $\frac{1}{2}\rho(\phi)$.

Proof. Suppose that proof size of $\phi^{\text{tail}}$ in Q-Res was smaller than $\frac{1}{2}\rho(\phi)$ and let $\pi$ be such a short Q-Res refutation. To obtain the empty clause all universal variables must be reduced by universal reduction in $\pi$ (there is no other option, which is the decisive difference to QU-Res). But then we can construct a Q-Res proof $\pi'$ for $\phi^{\text{double}}$ by just doubling all reduction steps in $\pi$ in the sense of introducing an additional reduction step for $v_i$ as soon as $u_i$ is reduced. That is always possible, because $v_i$ is never quantified left from $u_i$. The remainder of the proof can be left unchanged, since the variable copies $(v_i, \overline{v_i})$ cannot cause any tautologies that would not also be caused by the originals $(u_i, \overline{u_i})$. The proof constructed in this way remains in the same order of magnitude as the original one, more precisely $|\pi'| \leq 2|\pi| < \rho(\phi)$ in contradiction to the above observation of Lemma 46. Thus any Q-Res refutation for $\phi^{\text{tail}}$ has size at least $\frac{1}{2}\rho(\phi)$.
Classes of Hard Formulas for QBF Resolution

\[
\{u_1 \cup C_1 \{\overline{u_1}\} \cup C_1 \cdots \{u_n \cup C_n \{\overline{u_n}\} \cup C_n \}
\]

\[
C_1 \cdots C_n \cup C \cup C_{n+1}
\]

\[
C \text{ for all } C \in \varphi \setminus F \quad C \text{ for all } C \in F
\]

\[
\text{short Q-Res}
\]

refutation for \(\phi_n\)

Figure 14: Polynomial-size QU-Res refutations for \(\phi_{\psi,n}^{\text{tail}}\).

Lemma 47 in combination with the conditions from Theorem 43 (i.e., exponential strategy size of \(\phi_n\)) implies Q-Res-hardness of \(\phi_{\psi}^{\text{tail}}\):

Corollary 48 (\(\phi_{\psi,n}^{\text{tail}}\) is Hard for Q-Res). For \(n \in \mathbb{N}\) let \(\phi_{\psi,n}^{\text{tail}}\) be tailed versions constructed from \(\phi_n\) and \(\psi_n\) following the rules and conditions from Theorem 43. Then \(\phi_{\psi,n}^{\text{tail}}\) requires exponential-size Q-Res refutations.

Let us now prove the upper bound stated in Theorem 43:

Lemma 49 (\(\phi_{\psi,n}^{\text{tail}}\) has Short QU-Res Refutations). If for \(n \in \mathbb{N}\) \(\phi_{\psi,n}^{\text{tail}}\) are QBFs constructed from \(\phi_n\) and \(\psi_n\) following the rules and conditions from Theorem 43, then \(\phi_{\psi,n}^{\text{tail}}\) has polynomial-size QU-Res refutations.

Proof. \(\phi_n = \mathcal{P} \cdot \varphi_n\) has by assumption short Q-Res proofs. \(\phi_{\psi,n}^{\text{tail}}\) additionally contains the clauses \(\{u_i \cup C_i \} \cup \{\overline{u_i} \cup C_i \}\) for all \(i \in [n]\), from which we can get all the clauses \(C_i, i \in [n]\) in only \(n\) universal resolution steps (available in QU-Res). We then remove all the additional literals from the clauses \(F \times C_{n+1}\) originated from \(F\), which needs \(|F|\) times as many resolution steps as the short refutation of \(\psi_n\). Together with the unchanged clauses from \(\varphi_n \setminus F\) we now derived all clauses from \(\phi_n\) and can proceed with the short Q-Res refutation of \(\phi_n\). Overall the proof of \(\phi_n\) is extended by polynomially many steps. Therefore we get a polynomial-size QU-Res refutation of \(\phi_{\psi,n}^{\text{tail}}\). The composition of the proof is shown in Figure 14.

Proof of Theorem 43. The theorem follows from Corollary 48 and Lemma 49.

5.4 Examples

We illustrate our construction on the interleaved equality formulas from Beyersdorff et al. (2019), which we already discussed in Section 5.1:

Example 50 (SC-Tailed Equality). We first need suitable formulas, on which we can use the tail construction:

\[
\phi_n = (\exists x_1 \forall u_1 \exists t_1) \cdots (\exists x_n \forall u_n \exists t_n) \cdot \left( \bigcup_{i \in [n]} \{x_i, u_i, \overline{t_i}\}, \{x_i, \overline{u_i}, \overline{t_i}\} \right) \cup \{t_1, \ldots, t_n\}.
\]
As mentioned in Section 5.1, these are exactly the $\text{ilSC}^\text{EQ}_n$-formulas, i.e., they have exponential strategy size and short $Q$-Res refutations. Thus, they meet the requirements for constructing separating formulas according to the above method. The existential player has a strategy to satisfy all clauses except for $\{x_n, u_n, \bar{t}_n\}, \{\bar{x}_n, \bar{u}_n, \bar{t}_n\}$ and $\{t_1, \ldots, t_n\}$ 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 $\{\bar{x}_n, \bar{u}_n, \bar{t}_n\}$,
- $x_n = u_n = 0, t_n = 1$ falsifies $\{x_n, u_n, \bar{t}_n\}$ 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 $F = \{\{t_1, \ldots, t_n\}\}$. As a template for the tail, we use the simple contradiction formulas from Section 3.1 (note that we slightly change the naming of the clauses: $C_i = \{\bar{e}_i\}$ for $i \in [n]$ and $C_{n+1} = \{e_1, \ldots, e_n\}$). The tail construction then leads to the following formulas, separating $Q$-Res and QU-Res:

$$\phi_{\text{tail}}^{\text{tail}}_{\text{SC}, n} = \left(\text{ilSC}^\text{EQ}\right)_{\text{SC}, n} = (\exists x_1 \forall u_1 \exists t_1) \ldots (\exists x_n \forall u_n \exists t_n) \exists e_1 \ldots e_n \cdot \bigcup_{i \in [n]} \{\{x_i, u_i, \bar{t}_i\}, \{\bar{x}_i, \bar{u}_i, \bar{t}_i\}, \{u_i, e_i\}, \{\bar{u}_i, \bar{e}_i\}\} \cup \{\{t_1, \ldots, t_n, e_1, \ldots, e_n\}\}.$$

**Example 51 (IC-Tailed Equality).** We could also use another propositional template to tail the $\text{ilSC}^\text{EQ}_n$-formulas, e.g. implication chain formulas from Section 3.1. With $F = \{\{t_1, \ldots, t_n\}\}$ again and $\psi = \text{IC}_{n+1} = \bigcup_{i \in [n+1]} \{C_i\}$ with $C_i = \{e_{i-1}, \bar{e}_i\}$ for $i \in [1, n-1]$ and $C_n = \{\bar{e}_0\}, C_{n+1} = \{e_{n-1}\}$ we get the following formulas, separating $Q$-Res and QU-Res:

$$\phi_{\text{IC}, n}^{\text{tail}} = \left(\text{ilSC}^\text{EQ}\right)_{\text{IC}, n} = (\exists x_1 \forall u_1 \exists t_1) \ldots (\exists x_n \forall u_n \exists t_n) \exists e_0 \ldots e_n \cdot \bigcup_{i \in [1, n]} \{\{x_i, u_i, \bar{t}_i\}, \{\bar{x}_i, \bar{u}_i, \bar{t}_i\}\} \bigcup \left(\bigcup_{i \in [1, n-1]} \{\{u_i, e_{i-1}, \bar{e}_i\}, \{\bar{u}_i, e_{i-1}, \bar{e}_i\}\} \bigcup \{\{u_n, \bar{e}_0\}, \{\bar{u}_n, e_0\}\} \bigcup \{\{t_1, \ldots, t_n, e_{n-1}\}\}.$$

Interestingly, the KBKF formulas (Kleine Büning et al., 1995) correspond to the tail construction with simple contradiction formulas, where we just flip positive and negative literals ($\text{SC}'$ formulas). They actually inspired our construction:
Example 52 (KBKF). The KBKF formulas presented by Kleine Büning et al. (1995) are defined as

$$\phi_{SC', n}^{tail} = KBKF_n = \exists y_0(\exists y_1 y'_1 \forall u_1) \ldots (\exists y_n y'_n \forall u_n) \exists y_{n+1} \ldots y_{n+t} \cdot \bigcup_{i \in [0, 2n]} \{ C_i, C_i' \}$$

where the matrix clauses are defined as follows:

- $$C_0 = \{ \overline{y_0} \}$$
- $$C'_0 = \{ y_0, \overline{y'_1}, \overline{y'_1} \}$$
- $$C_k = \{ y_k, \overline{u_k}, \overline{y_{k+1}}, \overline{y'_k}, \overline{y'_{k+1}} \}$$
- $$C'_k = \{ y'_k, u_k, \overline{y_{k+1}}, \overline{y'_{k+1}} \}$$
- $$C_n = \{ y_n, \overline{u_n}, \overline{y_{n+1}}, \ldots, \overline{y_{n+n}} \}$$
- $$C'_n = \{ y'_n, u_n, y_{n+1}, \ldots, y_{n+n} \}$$
- $$C_{n+t} = \{ u_t, y_{n+t} \}$$

with $$1 \leq k < n$$ and $$1 \leq t \leq n$$. We now immediately see, that some parts of the formula equal those constructed in Section 5. Especially the variables $$y_{n+1}, \ldots, y_{n+n}$$ correspond to those called $$e_1, \ldots, e_m$$ in Section 5, which make up the tail. We examine the basic formula, whose modification according to the tail construction leads to the KBKF formulas:

$$\phi_n = \exists y_0(\exists y_1 y'_1 \forall u_1) \ldots (\exists y_n y'_n \forall u_n) \cdot \bigcup_{i \in [0 \ldots n]} \{ D_i, D'_i \}$$

with

- $$D_0 = C_0 = \{ \overline{y_0} \}$$
- $$D'_0 = C'_0 = \{ y_0, \overline{y'_1}, \overline{y'_1} \}$$
- $$D_k = C_k = \{ y_k, \overline{u_k}, \overline{y_{k+1}}, \overline{y'_k}, \overline{y'_{k+1}} \}$$
- $$D'_k = C'_k = \{ y'_k, u_k, \overline{y_{k+1}}, \overline{y'_{k+1}} \}$$
- $$D_n = \{ y_n, \overline{u_n} \}$$
- $$D'_n = \{ y'_n, u_n \}$$

for $$1 \leq k < n$$.

$$\phi_n$$ is also a false QBF and the existential player can force the universal player to follow the same strategy as in KBKF: setting $$u_k = y'_k$$ for each $$k \in [n]$$. (Note that this is not a unique winning strategy, since the existential player could leave the universal player a wide range of freedom in assigning the universal variables.) To force the universal player assigning variables according to the KBKF-strategy, the existential player will assign $$y_k = y'_k$$ in every round $$k \in [n]$$. The last remaining clauses are $$D_n = \{ y_n, \overline{u_n} \}$$ and $$D'_n = \{ y'_n, u_n \}$$, and every so constructed assignment falsifies exactly one of them: $$y_n = 0$$, $$y'_n = 1 = u_n$$ falsifies $$D_n$$ and $$y_n = 1$$, $$y'_n = 0 = u_n$$ falsifies $$D'_n$$; in each case the other clause is satisfied. Thus it is sufficient for the set $$F$$ of target clauses to contain one of the two clauses. For KBKF $$F = \{ D_n, D'_n \}$$ was chosen (which is not minimal), which makes the tail construction with SC' as propositional tail pattern formula generating just the KBKF formulas. Polynomial-size Q-Res refutations of $$\phi_n$$ are shown in Figure 15.

Hence $$\phi_n$$, $$n \in \mathbb{N}$$ has exponential strategy size and short Q-Res refutations, thus satisfying the conditions of the tail construction. It follows immediately, that the KBKF formulas separate Q-Res from QU-Res.

As an aside we see that $$F$$ can be minimized, i.e., the negative literals $$\overline{y_{n+1}}, \ldots, \overline{y_{n+n}}$$ can be removed from one of the clauses $$C_n$$ or $$C'_n$$ without affecting the separation property.
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 \( \Sigma^b_3 \) 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 \( \Sigma^b_3 \) 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 \( \Sigma^b_3 \)-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 (Beyersdorff et al., 2019) 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.

7. Acknowledgments

This research was supported by the Carl-Zeiss Foundation and DFG grant BE 4209/3-1. We would like to express our appreciation to the anonymous reviewers for their helpful comments and constructive feedback, which contributed to the improvement of Section 5.2 and hopefully increased its comprehensibility. An extended abstract of this paper has been published in the proceedings of SAT (Schleitzer & Beyersdorff, 2022).

References


Beyersdorff, O., Pulina, L., Seidl, M., & Shukla, A. (2021b). Qbf fam: A tool for generating QBF families from proof complexity. In Li, C., & Manyà, F. (Eds.), *Theory and


