Refined Notions of QBF Equivalences*

Peter Pfeiffer[1,2][0009−0008−8414−1463], Daniel Große[2][0000−0002−1490−6175], and Martina Seidl[1][0000−0002−3267−4494]

[1] Institute for Symbolic Artifial Intelligence, JKU Linz, Austria
[2] Institute for Complex Systems, JKU Linz, Austria
{peter.pfeiffer, daniel.grosse, martina.seidl}@jku.at

Abstract. Usually, two quantified Boolean formulas (QBFs) are said to be equivalent if they have the same truth value for every assignment to the free variables. This notion of equivalence is very coarse-grained in the sense that it considers only assignments to the free variables, but it does not take the models or counter-models of the two QBFs into account. In this paper, we investigate refined notions of equivalences on the solution level to obtain a more fine-grained comparison of two formulas. We show that the problem of checking solution equivalence is PSPACE complete.

Keywords: QBF · QBF Solutions · Equivalence Checking

1 Introduction

Quantified Boolean formulas (QBFs) [1, 3] extend propositional logic by quantifiers, enabling the compact representation of PSPACE-hard problems, which have numerous practical applications [7]. In such applications not only the truth values of the QBFs but also their solutions are of interest. These solutions encode the found plans in planning, error traces in formal verification, or a winning strategy in a two-player game. Solutions of QBFs are represented either as trees of a certain structure or, in practical solving, usually more compactly as Skolem functions for true QBFs and, dually, as Herbrand functions for false QBFs.

Classically, two QBFs are said to be equivalent if they evaluate to the same truth value for every assignment to their free variables, i.e., variables that are not bound by a quantifier [4]. Notably, this notion of equivalence neither requires the two QBFs to be defined over the same set of quantified variables nor takes the quantifier structure into account. When developing and optimizing QBF encodings, however, it is more usefull to compare not only the truth values of different formulas but also their solutions. The notion of equivalence models presented in [5] ultimately focus on the free variables as well while considering one specific solution only. Shaik et al. [6] presented a debugging tool that allows for user-guided exploration of whether two QBFs have matching solutions w.r.t. their quantified variables. However, this approach generates solutions for one encoding and checks if they are solutions for the other encoding as well.

* Supported by the LIT AI Lab and the LIT Secure Systems Lab funded by the state of Upper Austria and by the Austrian Science Fund (FWF) [10.55776/COE12].

In this paper, we investigate the notion of *solution equivalence* for checking if two given QBFs with the same quantifier prefix have the same set of models/counter-models. To this end, we first focus on true formulas and introduce the notion of *Skolem entailment* that allows to define the notion of *Skolem equivalence*. For Skolem entailment checking, we present a compact QBF encoding that can be directly employed for solution equivalence checking. Finally, we show that solution equivalence checking is PSPACE-complete.

2 Preliminaries

We consider *Boolean formulas* built from a given set of variables V, truth constants $\top$ (true) and $\bot$ (false), and the logical connectives $\{\neg, \wedge, \vee, \rightarrow, \leftrightarrow\}$. By $var(\phi)$ we denote the set of variables occuring in a formula ϕ. An *assignment* $\sigma : V' \rightarrow \{\top, \bot\}$ is a function that maps propositional variables $V' \subseteq V$ to $\top$ and $\bot$. By $[\phi]_\sigma$ we denote the formula obtained when the variables in ϕ are replaced according to σ and the resulting formula is simplified under standard semantics. If $[\phi]_\sigma = \top$, then σ is a *model* of ϕ, if $[\phi]_\sigma = \bot$, then σ is a *counter-model*.

A *quantified Boolean formula* (QBF) $\Phi = P.\phi$ consists of a *quantifier prefix* $P = Q_1 v_1 \dots Q_n v_n$ ($Q_i \in \{\forall, \exists\}$, $v_i \in V$, $v_i \neq v_j$ for $i \neq j$) and a Boolean formula ϕ, which is also called *matrix*. With $var(P) = \{v_i \mid Q_i v_i \text{ occurs in } P\}$ we denote the set of variables bound in prefix P. *Free variables* $free(\Phi) = var(\phi) \setminus var(P)$ are not bound by a quantifier. If $free(\Phi) = \emptyset$ then Φ is a *closed formula*. We sometimes write successive quantifiers $Qx_1 \dots Qx_n$ of the same type more compactly as QX where $X = \{x_1, \dots, x_n\}$. If the variables in X do not occur in a prefix P, $QX : P$ denotes the prefix obtained by prepending QX to P. For a QBF $\Phi = P.\phi$, the negation $\neg\Phi$ is the QBF $P'.(\neg\phi)$ where P' is obtained from P by flipping the quantifiers. The semantics of QBFs is follows: A QBF $\forall v P.\phi$ is true iff $(P.[\phi]_{\{v=\top\}})$ and $(P.[\phi]_{\{v=\bot\}})$ are true. Dually, a QBF $\exists v P.\phi$ is true iff $(P.[\phi]_{\{v=\top\}})$ or $(P.[\phi]_{\{v=\bot\}})$ is true. A QBF $\Phi = P.\phi$ with free variables V' is true iff there exists an assignment $\sigma : V' \rightarrow \{\top, \bot\}$ such that the closed QBF $[\Phi]_\sigma = P.[\phi]_\sigma$ is true. Models and counter-models of closed QBFs are expressed in terms of binary trees of a certain structure. Consider a true QBF $\Phi = Q_1 v_1 \dots Q_n v_n.\phi$ with n variables. Then a model S of Φ is a tree of height $n + 1$ such that each leaf node is labeled with $\top$ and each node at level i of the tree corresponds to variable v_i of Φ. Each node at level i has two children if $\mathcal{Q}_i = \forall$ and one child otherwise. One edge from a node with a universal variable is labeled with $\bot$, the other edge is labeled with $\top$. For existential nodes, the label on the edge to the child has to be set in such a manner that the full variable assignment on each path from the root to a leaf over the respective edge satisfies ϕ. We write $\sigma \in S$, if there is a path in S that corresponds to assignment σ. The set of all models of a closed QBF Φ is denoted by $\mathbb{S}_\exists(\Phi)$. Obviously, if Φ is false, then $\mathbb{S}_\exists(\Phi) = \emptyset$. Counter-models of false QBFs $\Phi = Q_1 v_1 \dots Q_n v_n.\phi$ are defined dually: nodes with existential variables have two children, nodes with universal variables have one child, and the leaves of the tree contain $\bot$. Each full assignment on a path from the root to a leaf is a counter-model of ϕ. The set of all counter-models of a closed QBF Φ is denoted by $\mathbb{S}_\forall(\Phi)$. Obviously, if Φ is true, then $\mathbb{S}_\forall(\Phi) = \emptyset$. Given a closed QBF Φ, by $\mathbb{S}(\Phi)$ we denote the set of all models and counter-models, i.e., $\mathbb{S}(\Phi) = \mathbb{S}_\exists(\Phi) \cup \mathbb{S}_\forall(\Phi)$. QBF models can also be represented as Skolem functions and QBF counter-models can be represented as Herbrand functions, but for this work the tree-representation is more convenient.

3 Notions of Equivalences

In propositional logic, two Boolean formulas ϕ and ψ which are defined over the same variables are said to be *equivalent* (written as $\phi \Leftrightarrow \psi$) if for every assignment $\sigma : V \to \{\top, \bot\}$ it holds that $[\phi]_\sigma = [\psi]_\sigma$. The more relaxed notion of *satisfiability equivalence* only requires that both formulas are satisfiable or that both formulas are unsatisfiable. For many purposes like efficient normal form transformation [8] or formula simplification through preprocessing [2], preserving satisfiability equivalence is sufficient. The notions of (satisfiability) equivalence have also been transfered to QBFs with free variables [4, 5].

Definition 1. *Let Φ and Ψ be QBFs with free variables V'. Then Φ and Ψ are* equivalent *(resp.* satisfiability equivalent*) iff $[\Phi]_\sigma$ and $[\Psi]_\sigma$ have the same truth values for all assignments (resp. for some assignment) $\sigma : V' \to \{\top, \bot\}$.*

In this definition, equivalence and satisfiability equivalence have the free variables as discriminator. Quantified variables are not considered at all, hence equivalence and satisfiability equivalence are the same for closed QBFs.

Example 1. The two quantifier-free formulas $\phi_1 = (((a \leftrightarrow b) \vee c) \wedge d)$ and $\phi_2 = (((a \leftrightarrow b) \vee \neg c) \wedge d)$ are not equivalent, but satisfiability equivalent. The closed QBFs $\Phi_1 = P.\phi_1$ and $\Phi_2 = P.\phi_2$ with prefix $P = \forall a \exists b \forall c \exists d$ are equivalent in the sense of Definition 1 and so are Φ_1 and $\Phi_3 = P.d$, but $P.\phi_1$ and $P.(\phi_1 \wedge c)$ are not, because the former formula is true and the latter is false.

In the example above, the equivalent QBFs Φ_1 and Φ_2 even have the same solutions, i.e., $\mathbb{S}(\Phi_1) = \mathbb{S}(\Phi_2)$. Note that their matrices ϕ_1 and ϕ_2 are not equivalent. In contrast, Φ_1 and Φ_3 have different solutions, i.e., $\mathbb{S}(\Phi_1) \neq \mathbb{S}(\Phi_3)$ despite being equivalent according to Definition 1: any assignment in which variable d is set to $\top$ satisfies the matrix of Φ_3. However, setting d to $\top$ is not enough for satisfying the matrix of Φ_1. In the following, we present a more fine-grained notion of QBF equivalence that also takes the solutions of the formulas into account. As indicated by the example above, requiring that the matrices of the two QBFs are equivalent is not a necessary creterion. For simplicity, we consider only closed QBFs, but the introduced notions naturally extend to QBFs with free variables. To relate two QBFs, we first start with the notion of Skolem entailment.

Definition 2 (Skolem Entailment). *Let $\Phi = P.\phi$ and $\Psi = P.\psi$ be two closed QBFs. Then Φ* Skolem entails *Ψ (written as $\Phi \models_{\mathsf{Sk}} \Psi$) iff $\mathbb{S}_\exists(\Phi) \subseteq \mathbb{S}_\exists(\Psi)$.*

In the definition of Skolem entailment, not only the truth values of the given QBFs are considered, but also their models. The definition requires that the two QBFs have the same quantifier prefix to ensure that the tree models have the same structure. A false formula trivially Skolem entails every formula, while a false formula is only Skolem entailed by a false formula. This is consistent with the definition of entailment as commonly found in the literature (e.g., [4]).

Example 2. Consider the QBFs $\Phi_1 = P.(((a \leftrightarrow b) \vee c) \wedge d)$, $\Phi_2 = P.(((a \leftrightarrow b) \vee \neg c) \wedge d)$, and $\Phi_3 = P.d$ with prefix $P = \forall a \exists b \forall c \exists d$ from the previous example. It holds that $\Phi_1 \models_{\mathsf{Sk}} \Phi_2$ and $\Phi_2 \models_{\mathsf{Sk}} \Phi_1$. Further, $\Phi_1 \models_{\mathsf{Sk}} \Phi_3$, but $\Phi_3 \not\models_{\mathsf{Sk}} \Phi_1$.

Next, we use Skolem entailment to define the notion of Skolem equivalence which holds if two formulas with the same prefix have the same set of models.

Definition 3 (Skolem Equivalence). *Two closed QBFs Φ and Ψ are* Skolem equivalent *(written as $\Phi \Leftrightarrow_{\mathsf{Sk}} \Psi$) iff $\Phi \models_{\mathsf{Sk}} \Psi$ and $\Psi \models_{\mathsf{Sk}} \Phi$, i.e., $\mathbb{S}_\exists(\Phi) = \mathbb{S}_\exists(\Psi)$.*

Example 3. The QBFs Φ_1 and Φ_2 from the previous example are Skolem equivalent ($\Phi_1 \Leftrightarrow_{\mathsf{Sk}} \Phi_2$), but Φ_1 and Φ_3 are not ($\Phi_1 \not\Leftrightarrow_{\mathsf{Sk}} \Phi_3$).

Two false formulas Φ and Ψ with the same prefix are always Skolem equivalent, because $\mathbb{S}_\exists(\Phi) = \mathbb{S}_\exists(\Psi) = \emptyset$. To consider counter-models as well, we introduce the notion of *solution equivalence* as follows.

Definition 4 (Solution Equivalence). *Two closed QBFs Φ and Ψ are* solution equivalent *(written as $\Phi \Leftrightarrow_{\mathsf{Sol}} \Psi$) iff $\mathbb{S}(\Phi) = \mathbb{S}(\Psi)$.*

Obviously, a true QBF Φ and a false QBF Ψ can never be solution equivalent, because $\mathbb{S}_\forall(\Phi) = \emptyset$, $\mathbb{S}_\exists(\Psi) = \emptyset$, and the intersection of the non-empty sets $\mathbb{S}_\exists(\Phi)$ and $\mathbb{S}_\forall(\Psi)$ is empty. Since the models of a true QBF Φ are the counter-models of the false QBF $\neg\Phi$ and, dually, since the counter-models of a false QBF Ψ are the models of the true QBF $\neg\Psi$, solution equivalence can be expressed in terms of Skolem equivalence.

Lemma 1. *Two closed QBFs Φ and Ψ which have the same prefix are solution equivalent ($\Phi \Leftrightarrow_{\mathsf{Sol}} \Psi$) iff $\Phi \Leftrightarrow_{\mathsf{Sk}} \Psi$ and $\neg\Phi \Leftrightarrow_{\mathsf{Sk}} \neg\Psi$.*

This lemma allows us to express solution equivalence checking in terms of Skolem equivalence checking, which, in turn, can be expressed in terms of Skolem entailment checking. Next, we present a QBF encoding for this reasoning task.

4 Equivalence Checking

Two propositional formulas ϕ and ψ over variables V are equivalent if the QBF $\forall V.(\phi \leftrightarrow \psi)$ is true. In the previous section, an example showed that the equivalence of their propositional matrices is not a criterion for solution equivalence of QBFs. To obtain a QBF encoding for solution equivalence checking, we first introduce a QBF $\Delta(\Phi, \Psi)$ which has the same prefix as Φ and Ψ plus some prepended existential variables and is false iff $\Phi \models_{\mathsf{Sk}} \Psi$.

Lemma 2. *Let $\Phi = P.\phi$ and $\Psi = P.\psi$ be two true closed QBFs with the same prefix P. Then $\Phi \models_{\mathsf{Sk}} \Psi$ iff the QBF*

$$\Delta(\Phi, \Psi) = \exists X' : P.(\phi \wedge ((X' \leftrightarrow X) \rightarrow \neg\psi))$$

is false where $X' = \{u_x \mid \forall x \text{ occurs in } P\}$ and $(X' \leftrightarrow X) := \bigwedge_{u_x \in X'}(u_x \leftrightarrow x)$.

Proof. $\Rightarrow$: Assume that $\Phi \models_{\mathsf{Sk}} \Psi$. By definition, every model $S \in \mathbb{S}(\Phi)$ is also a model of Ψ. Further assume that $\Delta(\Phi, \Psi)$ is true. As the QBFs Φ and $[\Delta(\Phi, \Psi)]_\sigma$ have the same prefix P, and as the matrix of $\Delta(\Phi, \Psi)$ strengthens the matrix ϕ of Φ, it holds that $\mathbb{S}([\Delta(\Phi, \Psi)]_\sigma) \subseteq \mathbb{S}(\Phi)$ for any assignment $\sigma : X' \rightarrow \{\top, \bot\}$.

Let $\tau_1 : X' \rightarrow \{\top, \bot\}$ be an assignment such that $[\Delta(\Phi, \Psi)]_{\tau_1} = \top$ and let $S \in \mathbb{S}([\Delta(\Phi, \Psi)]_{\tau_1})$. Now we pick a path from the root to a leaf in S such that for the corresponding variable assignment $\tau_2 \in S$ it holds that $\tau_2(x) = \tau_1(u_x)$ for all $u_x \in X', x \in X$. Let $\tau = \tau_1 \cup \tau_2$. Then $[\phi]_\tau = \top$ and $[X \leftrightarrow X']_\tau = \top$. Because of $\Phi \models_{\mathsf{Sk}} \Psi$, $[\neg\psi]_\tau = \bot$, hence $[\delta]_\tau = \bot$ where δ is the matrix of $\Delta(\Phi, \Psi)$. This contradicts the assumption that S is a model of $[\Delta(\Phi, \Psi)]_{\tau_1}$.
$\Leftarrow$: Assume that $\Delta(\Phi, \Psi)$ is false, but $\Phi \models_{\mathsf{Sk}} \Psi$ does not hold, i. e., there is a model $S \in \mathbb{S}(\Phi)$ of Φ that is not a model of Ψ, i. e., $S \notin \mathbb{S}(\Psi)$. Hence, there is an assignment $\sigma \in S$ with $[\phi]_\sigma = \top$, but $[\psi]_\sigma = \bot$. Let $\tau : X' \rightarrow \{\top, \bot\}$ be an assignment such that $\tau(u_x) = \sigma(x)$ for all $u_x \in X', x \in X$. For all assignments $\tau' \in S$ with $\tau' \neq \sigma$, (1) $[\phi]_{\tau'} = \top$, because S is a model of ϕ and (2) $[(X' \leftrightarrow X) \rightarrow \neg\psi]_{\tau \cup \tau'} = \top$. The latter holds, because $\tau(u_X) \neq \tau'(x)$ for some $u_x \in X', x \in X$ and the left-hand side of the implication is false. But also $[(X' \leftrightarrow X) \rightarrow \neg\psi]_{\tau \cup \sigma} = \top$, because $\tau(x') = \sigma(x)$ for all $x' \in X', x \in X$ and $[\neg\psi]_\sigma = \top$. Hence, S is a model for $P.[\delta]\tau$ with δ being the matrix of $\Delta(\Phi, \Psi)$. This means that $\Delta(\Phi, \Psi)$ has a model, contradicting the assumption that $\Delta(\Phi, \Psi)$ is false. It follows that $\Phi \models_{\mathsf{Sk}} \Psi$.

□

The Δ formula introduces a set of existentially quantified variables X' at the beginning of its prefix such that X' contains one variable for each universal variable of P. Then Δ is true if there is an assignment to the universal variables such that there is an assignment to the existential variables satisfying matrix ϕ, but falsifying matrix ψ. This is illustrated in the following example.

Example 4. Consider the QBFs $\Phi_3 = P.d$ and $\Phi_2 = P.(((a \leftrightarrow b) \vee \neg c) \wedge d)$ with $P = \forall a \exists b \forall c \exists d$ from the previous examples. Then

$$\Delta(\Phi_3, \Phi_2) = \exists a' \exists c' \forall a \exists b \forall c \exists d : d \wedge (((a' \leftrightarrow a) \wedge (c' \leftrightarrow c)) \rightarrow \neg(((a \leftrightarrow b) \vee \neg c) \wedge d))$$

is true if a' is set to $\bot$ and c' is set to $\top$. The reason can be seen in the assignment tree shown in Figure 1. The yellow subtree is a model of Φ_3, but not of Φ_2, because of the assignment on the red path on which a and c have the same values as a' and c' that make $\Delta(\Phi_3, \Phi_2)$ true.

Since Skolem entailment can be reduced to a QBF, Skolem equivalence checking can also be reduced to a QBF. Consequently, by Lemma 1, solution equivalence checking can be reduced to a QBF as well. It follows that all three problems

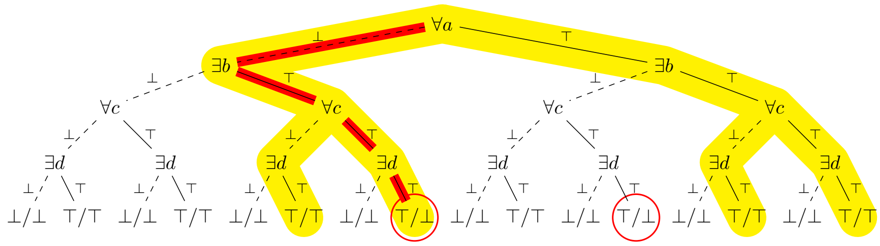

Fig. 1. Full assignment tree for QBFs Φ_3/Φ_2 from Example 4. The leaves contain truth values of ϕ_3/ϕ_2 under the assignment on the path to the root (different results are in red circles). The yellow subtree is a model of Φ_3, but not of Φ_2 because of the red path.

lie in PSPACE. Furthermore, PSPACE-hardness is derived from the following lemma.

Lemma 3. *A closed QBF Φ is true iff $\Phi \models_{\mathsf{Sk}} \bot$ does not hold.*

Proof. On the one hand, if Φ is true, then $\mathbb{S}_\exists(\Phi) \neq \emptyset$, but $\mathbb{S}_\exists(\bot) = \emptyset$. Hence, $\mathbb{S}_\exists(\Phi) \not\subseteq \mathbb{S}_\exists(\top)$ and $\Phi \models_{\mathsf{Sk}} \bot$ cannot hold. On the other hand, if $\Phi \models_{\mathsf{Sk}} \bot$ does not hold, then $\mathbb{S}_\exists(\Phi) \neq \emptyset$. Hence, Φ is true. $\square$

Proposition 1. *The problems of Skolem entailment checking, Skolem equivalence checking, and solution equivalence checking are PSPACE complete.*

The notions and the results discussed above can be directly extended to formulas with free variables. For example, solution equivalence checking with free variables is defined as follows.

Definition 5. *Let Φ and Ψ be QBFs with free variables V'. Then Φ and Ψ are* solution equivalent *iff $[\Phi]_\sigma \Leftrightarrow_{\mathsf{Sol}} [\Psi]_\sigma$ for all assignments $\sigma : V' \to \{\top, \bot\}$.*

It follows that solution equivalence checking for QBFs with free variables is PSPACE complete as well.

5 Conclusion

In this work, we introduced the notion of solution equivalences for QBFs. Solution equivalence does not consider only truth values and assignments of free variables to compare two formulas, but models and counter-models. We showed that solution equivalence checking is PSPACE-complete.

The QBF encoding we presented in this paper has the potential for practical applications in future work. On the basis of this encoding, a QBF solver can be used to check solution equivalence of two formulas. The current approach is limited to compare QBFs with the same prefix. In the future, we plan to explore more general notions of equivance that are more relaxed with respect to prefix structures.

References

1. Beyersdorff, O., Janota, M., Lonsing, F., Seidl, M.: Quantified boolean formulas. In: Handbook of Satisfiability - Second Edition, Frontiers in Artificial Intelligence and Applications, vol. 336, pp. 1177–1221. IOS Press (2021)
2. Heule, M., Järvisalo, M., Lonsing, F., Seidl, M., Biere, A.: Clause elimination for SAT and QSAT. J. Artif. Intell. Res. **53**, 127–168 (2015)
3. Kleine Büning, H., Bubeck, U.: Theory of quantified boolean formulas. In: Handbook of Satisfiability - Second Edition, Frontiers in Artificial Intelligence and Applications, vol. 336, pp. 1131–1156. IOS Press (2021)
4. Kleine Büning, H., Lettmann, T.: Aussagenlogik – Deduktion und Algorithmen. Teubner (1994)
5. Kleine Büning, H., Zhao, X.: Equivalence models for quantified boolean formulas. In: Proc. of the 7th Int. Conf. on Theory and Applications of Satisfiability Testing (SAT). Lecture Notes in Computer Science, vol. 3542, pp. 224–234. Springer (2004)
6. Shaik, I., Heisinger, M., Seidl, M., van de Pol, J.: Validation of QBF Encodings with Winning Strategies. In: Proc. of the 26th Int. Conf. on Theory and Applications of Satisfiability Testing (SAT). Leibniz International Proceedings in Informatics (LIPIcs), vol. 271, pp. 24:1–24:10. Schloss Dagstuhl – Leibniz-Zentrum für Informatik (2023)
7. Shukla, A., Biere, A., Pulina, L., Seidl, M.: A Survey on Applications of Quantified Boolean Formulas. In: Proc. of the 2019 IEEE 31st Int. Conf. on Tools with Artificial Intelligence (ICTAI). pp. 78–84. IEEE (2019)
8. Tseitin, G.S.: On the Complexity of Derivation in Propositional Calculus, pp. 466–483. Springer Berlin Heidelberg, Berlin, Heidelberg (1983)