Computing Smallest MUSes of Quantified Boolean Formulas*

Andreas Niskanen, Jere Mustonen, Jeremias Berg, and Matti Järvisalo

HIIT, Department of Computer Science, University of Helsinki, Finland

Abstract. Computing small (subset-minimal or smallest) explanations is a computationally challenging task for various logics and non-monotonic formalisms. Arguably the most progress in practical algorithms for computing explanations has been made for propositional logic in terms of minimal unsatisfiable subsets (MUSes) of conjunctive normal form formulas. In this work, we propose an approach to computing *smallest* MUSes of *quantified* Boolean formulas (QBFs), building on the so-called implicit hitting set approach and modern QBF solving techniques. Connecting to non-monotonic formalisms, our approach finds applications in the realm of abstract argumentation in computing smallest strong explanations of acceptance and rejection. Justifying our approach, we pinpoint the complexity of deciding the existence of small MUSes for QBFs with any fixed number of quantifier alternations. We empirically evaluate the approach on computing strong explanations in abstract argumentation frameworks as well as benchmarks from recent QBF Evaluations.

Keywords: quantified boolean formulas, minimum unsatisfiability, abstract argumentation, strong explanations

1 Introduction

Explaining inconsistency in different logics is a central problem setting with a range of applications. Finding small explanations for inconsistency is intrinsically a computationally even more challenging task than deciding satisfiability. What comes to practical algorithms for computing small explanations, arguably the most progress has been made in the realm of classical logic, in particular in propositional satisfiability where algorithms for computing minimal unsatisfiable subsets (MUSes) of conjunctive normal form formulas have been developed [7, 4, 8, 25]. Extensions to computing smallest MUSes [17, 21] and, on the other hand, to computing MUSes of quantified Boolean formulas (QBF) [16, 17] have also been proposed. Recently, it has been shown that the notion of so-called strong inconsistency [10, 36] provides for non-monotonic reasoning a natural counterpart of the inconsistency notion studied in the classical setting, satisfying the

* Work financially supported by Academy of Finland under grants 322869, 328718, 342145 and 347588. The authors thank the Finnish Computing Competence Infrastructure (FCCI) for computational and data storage resources.

well-known hitting set duality between explanations and diagnoses [30]. The general notion of strong inconsistency has already been instantiated for computing explanations in the non-monotonic formalisms of answer set programming [11, 26] and abstract argumentation [35, 34, 27, 37].

We propose an approach to computing *smallest* MUSes of *quantified* Boolean formulas (QBFs), building on the so-called implicit hitting set approach [12, 17, 31, 32] and modern QBF solving techniques [22, 23, 18]. Our approach generalizes an implicit hitting set approach [17] and a quantified MaxSAT approach [16] to computing smallest MUSes of propositional formulas to general QBFs. Justifying our approach, we pinpoint the computational complexity of deciding the existence of small MUSes for QBFs with any fixed number of quantifier alternations, generalizing and supplementing earlier complexity results related to MUSes [29, 20, 10]. While our approach is generic, a central motivation for developing the approach comes from the realm of abstract argumentation [13], in particular for explaining acceptance [15, 37, 5, 1] and rejection [33, 34, 27] of arguments. As we will detail, computation of *smallest* strong explanations [34, 27, 37] in abstract argumentation frameworks can naturally be viewed as the task of computing smallest MUSes of quantified formulas. While there is work on practical procedures for computing strong explanations for credulous *rejection* in abstract argumentation [34, 27], even the task of verifying a minimal strong explanation for credulous *acceptance* under admissible and stable semantics is by complexity arguments presumably beyond the reach of the earlier-proposed approaches [37]. The approach developed in this work hence provides a first practical approach to computing smallest strong explanations in particular for credulous acceptance and skeptical rejection in abstract argumentation frameworks.

Empirically, the approach scales favorably towards computing smallest strong explanations in abstract argumentation for ICCMA competition instances, despite the fact that this task is presumably considerably more challenging than the standard tasks of deciding acceptance considered in the ICCMA competitions. We also show that the approach allows for computing smallest MUSes for small unsatisfiable benchmarks from QBF Evaluation solver competitions.

2 Preliminaries

Quantified Boolean Formulas (QBFs). We consider closed QBFs in prenex normal form $\Phi = Q_1 X_1 \cdots Q_k X_k.\varphi$, where $Q_i \in \{\exists, \forall\}$ are alternating quantifiers, $X_1, \ldots, X_k$ are pairwise disjoint nonempty sets of variables, and φ is a Boolean formula over the variables $\bigcup_{i=1}^{k} X_i$ and the truth constants 1 and 0 (true and false). The sequence of quantifier blocks $Q_1 X_1 \cdots Q_k X_k$ is called the prefix and φ the matrix of Φ. We denote an arbitrary prefix of k alternating quantifier blocks by $\overrightarrow{Q}_k$. For a truth assignment $\tau \colon X \to \{0, 1\}$, the formula $\varphi[\tau]$ is obtained by replacing, for each $x \in X$, all occurrences of x in φ by $\tau(x)$. As convenient, we interchangeably view assignments as either sets of non-contradictory literals or as functions mapping variables to truth values. The QBF $\Phi^{\exists} = \exists X \overrightarrow{Q}_k.\varphi$ is true iff there exists a truth assignment $\tau \colon X \to \{0, 1\}$ for which

$\overrightarrow{Q}_k.\varphi\,[\tau]$ is true. In this case we call τ a solution to $\Phi^\exists$. The QBF $\Phi^\forall = \forall X \overrightarrow{Q}_k.\varphi$ is true iff for all truth assignments $\tau\colon X \to \{0,1\}$, the QBF $\overrightarrow{Q}_k.\varphi\,[\tau]$ is true. If this is not the case, that is, there is a truth assignment τ for which $\overrightarrow{Q}_k.\varphi\,[\tau]$ is false, we call τ a counterexample to $\Phi^\forall$.

Smallest MUSes of QBF Formulas. Consider a QBF $\Phi = \exists S \overrightarrow{Q}_k.\varphi$. A set $S^\star \subset S$ is a (an unsatisfiable) core of Φ if $\exists S \overrightarrow{Q}_k.\varphi\,[S^\star]$ is false. The smallest minimal unsatisfiable subsets (SMUSes) are smallest-cardinality cores: $S^\star$ is a SMUS iff $|S^\star| \leq |S'|$ holds for all cores S' of Φ. A set $cs \subset S$ is a correction set (CS) if the QBF $\exists S \overrightarrow{Q}_k.\varphi\,[S \setminus cs]$ is true. Note that these definitions are in line with conjunctive forms: for a QBF $\overrightarrow{Q}_k.\bigwedge_{j=1}^m \varphi_j$, where φ_j are formulas, unsatisfiable subsets (resp. correction sets) over $\{\varphi_1, \ldots, \varphi_m\}$ can be computed as cores (resp. correction sets) of $\exists S \overrightarrow{Q}_k.\bigwedge_{j=1}^m (s_j \to \varphi_j)$ with $S = \{s_1, \ldots, s_m\}$.

Important to our approach is a relationship between correction sets and SMUSes of QBFs: lower bounds on the size of SMUSes of a QBF are obtained via the minimum-cost hitting sets over *any* sets of its correction sets. A set $hs \subset S$ is a hitting set over a collection $\mathcal{C}$ of correction sets if it intersects with each $cs \in \mathcal{C}$. A hitting set hs is minimum-cost if $|hs| \leq |hs'|$ for all hitting sets hs'.

Proposition 1. *Let $\mathcal{C}$ be a set of correction sets of the QBF $\Phi = \exists S \overrightarrow{Q}_k.\varphi$, hs a minimum-cost hitting set over $\mathcal{C}$, and $S^\star$ a SMUS of Φ. Then $|hs| \leq |S^\star|$.*

Proof. (Sketch) We show that $S^\star$ is also a hitting set over $\mathcal{C}$. The claim follows by observing that hs is minimum-cost. Assume for a contradiction that $S^\star \cap cs = \emptyset$ for some $cs \in \mathcal{C}$. Then $S^\star \subset S \setminus cs$. Since $\exists S \overrightarrow{Q}_k.\varphi\,[S \setminus cs]$ is true, so is $\exists S \overrightarrow{Q}_k.\varphi\,[S^\star]$, contradicting the fact that $S^\star$ is a core. □

Abstract Argumentation. An argumentation framework (AF) [13] $F = (A, R)$ consists of a (finite) set of arguments A is and an attack relation $R \subseteq A \times A$. Argument b attacks argument a if $(b, a) \in R$. A $S \subseteq A$ is conflict-free in $F = (A, R)$ if $b \notin S$ or $a \notin S$ for each $(b, a) \in R$. The set of conflict-free sets in F is denoted by $cf(F)$. An AF semantics σ maps each AF to a collection $\sigma(F)$ of jointly acceptable subsets of arguments, i.e., extensions. A conflict-free set $S \in cf(F)$ is admissible if for each attack $(b, a) \in R$ with $a \in S$ there is an attack $(c, b) \in R$ with $c \in S$, i.e., for each attack on S there is a counterattack from S. A conflict-free set $S \in cf(F)$ is stable if for each argument $a \in A \setminus S$ there is an attack $(b, a) \in R$ with $b \in S$, i.e., all arguments outside S are attacked by S. The sets of admissible and stable extensions in F are denoted by $adm(F)$ and $stb(F)$, resp. An argument $q \in A$ is credulously accepted in F under semantics σ if there is an $E \in \sigma(F)$ with $q \in E$, and skeptically accepted in F under σ if $q \in E$ for all $E \in \sigma(F)$. Given an AF $F = (A, R)$ and $S \subseteq A$, the subframework induced by S is $F[S] = (S, R \cap (S \times S))$.

Computational complexity. We assume familiarity with standard complexity classes of the polynomial hierarchy, namely $\Sigma_0^p = \Pi_0^p = \mathrm{P}$, $\Sigma_{k+1}^p = \mathrm{NP}^{\Sigma_k^p}$, $\Pi_{k+1}^p = \mathrm{coNP}^{\Sigma_k^p}$ and $D_k^p = \{L_1 \cap L_2 \mid L_1 \in \Sigma_k^p, L_2 \in \Pi_k^p\}$, and the concepts of hardness and completeness [28].

3 Smallest Strong Explanations

Computing smallest MUSes of QBF formulas is motivated by the fact that it captures smallest strong explanations for credulous acceptance [37] under admissible and stable semantics and skeptical rejection under stable semantics in the realm of abstract argumentation. Let $F = (A, R)$ be an AF and $Q \subseteq A$ be a set of arguments. Following [37], a set $S \subseteq A$ is a strong explanation for credulously accepting Q under σ if for each $S \subseteq A' \subseteq A$ and $F' = F[A']$ there exists $E \in \sigma(F')$ with $Q \subseteq E$. Similarly, explanations for skeptically rejecting $Q \subseteq A$ under σ are sets $S \subseteq A$ for which, for all $S \subseteq A' \subseteq A$ and $F' = F[A']$, there exists $E \in \sigma(F')$ with $Q \nsubseteq E$.

We focus on smallest-cardinality strong explanations. Our approach builds on the line of work on employing propositional SMUS extractors for computing smallest strong explanations for credulous rejection [27]. We similarly declare Boolean variables x_a and y_a for each argument $a \in A$, interpreting $x_a = 1$ as argument a being included in an extension, and $y_a = 1$ as argument a being included in a subframework. We denote $Y = \{y_a \mid a \in A\}$ and $X = \{x_a \mid a \in A\}$. We define the propositional formula $\varphi_{cf}(F) = \bigwedge_{(a,b)\in R} ((y_a \wedge y_b) \rightarrow (\neg x_a \vee \neg x_b))$ for conflict-free sets, and formulas $\varphi_{adm}(F) = \varphi_{cf}(F) \wedge \bigwedge_{(b,a)\in R} \left((y_a \wedge y_b \wedge x_a) \rightarrow \bigvee_{(c,b)\in R}(y_c \wedge x_c)\right)$ and $\varphi_{stb}(F) = \varphi_{cf}(F) \wedge \bigwedge_{a\in A} \left((y_a \wedge \neg x_a) \rightarrow \bigvee_{(b,a)\in R}(y_b \wedge x_b)\right)$ encoding admissible and stable semantics. That is, for any assignment τ_Y over Y, the satisfying assignments over X of the formula $\varphi_\sigma(F)[\tau_Y]$ correspond exactly to $\sigma(F[A'])$ with $A' = \{a \in A \mid \tau(y_a) = 1\}$, since $\varphi_\sigma(F)[\tau_Y]$ reduces to a standard SAT encoding of semantics σ for which this result is well-known [9].

For extracting a smallest strong explanation for credulous acceptance and skeptical rejection, it suffices to compute a SMUS of a 2-QBF formula.[1]

Proposition 2. *Given an AF $F = (A, R)$, $Q \subseteq A$, and semantics $\sigma \in \{adm, stb\}$. Let $S^* \subseteq A$. It holds that $Y[S^*] = \{y_a \mid a \in S\}$ is a SMUS of*

a) $\Phi^{CA}_\sigma(F, Q) = \exists Y \forall X(\varphi_\sigma(F) \rightarrow \bigvee_{q\in Q} \neg x_q)$ if and only if S^* is a smallest strong explanation for credulously accepting Q in F under σ,

b) $\Phi^{SR}_{stb}(F, Q) = \exists Y \forall X(\varphi_{stb}(F) \rightarrow \bigwedge_{q\in Q} x_q)$ if and only if S^* is a smallest strong explanation for skeptically rejecting Q in F under stb.

Proof. Case a): Suppose S^* is a strong explanation for credulously accepting Q in F under σ, that is, for all $S^* \subseteq A' \subseteq A$ there is an extension $E \in \sigma(F[A'])$ containing Q. Equivalently, for all assignments τ_Y over Y which set $\tau_Y(y_a) = 1$ for $a \in S^*$, there is an assignment τ_X over X which satisfies $\varphi_\sigma(F)[\tau_Y]$ and sets $\tau_X(x_q) = 1$ for all $q \in Q$. This means that the QBF $\forall Y \exists X \left(\varphi_\sigma(F)[Y[S^*]] \wedge \bigwedge_{q\in Q} x_q\right)$ is true, which in turn means that

[1] The proposition also holds when considering subset-minimal strong explanations and MUSes of the corresponding 2-QBF.

$\Phi_{\sigma}^{CA}(F,Q)\,[Y\,[S^*]]$ is false. That is, $Y\,[S^*]$ is a core of $\Phi_{\sigma}^{CA}(F,Q)$. By applying the same steps in the other direction, we obtain a one-to-one mapping between strong explanations for credulous acceptance in F and cores of $\Phi_{\sigma}^{CA}(F,Q)$. The reasoning is similar for case b) and skeptical rejection. The claims follow. □

The SMUSes of a 2-QBF $\Phi_{com}^{SR}(F,Q)$ with a subformula $\varphi_{com}(F)$ for complete semantics (see e.g. [6]) capture strong explanations for skeptical rejection under complete, which in turn coincides with (credulous and skeptical) rejection under grounded semantics. Further, QBF encodings for second-level-complete argumentation semantics [14, 2] allow for similarly capturing strong explanations for, e.g., skeptical rejection under preferred semantics as SMUSes of 3-QBFs. In terms of computational complexity, *verifying* that a given subset of arguments is a *minimal* strong explanation for credulous acceptance is already D_2^p-complete under admissible and stable semantics [37]. In contrast, for credulous rejection this task is D_1^p-complete, and deciding whether a small strong explanation exists is Σ_2^p-complete [27]. By Proposition 2, the complexity of computing a smallest strong explanation for credulous acceptance and skeptical rejection is bounded by the complexity of computing a SMUS of a given 2-QBF formula. We find it likely that for credulous acceptance this task is complete for the third level of the polynomial hierarchy, namely, that deciding whether a small strong explanation exists is Σ_3^p-complete. This would be in line with the complexity of deciding whether a 2-QBF has a small unsatisfiable subset, detailed next.

4 On Complexity of Computing Smallest MUSes of QBFs

In the context of propositional logic, verification of a MUS is D_1^p-complete [29], and deciding the existence of MUS of small size is Σ_2^p-complete [20]. Further, verifying whether a given QBF with $k \geq 2$ alternating quantifiers is minimally unsatisfiable is D_k^p-complete [10]. However, to the best of our knowledge, the complexity of deciding whether a QBF has a small–of size at most a given integer–unsatisfiable subset has not been established so far. We show that the problem is Σ_{k+1}^p-complete for k-QBFs when the leading quantified is existential.

Theorem 1. *Consider a QBF $\exists X_1 \forall X_2 \cdots Q_k X_k. \bigwedge_{j=1}^{m} \varphi_j$, where φ_j are propositional formulas over $\bigcup_{i=1}^{k} X_i$. Deciding whether there is an unsatisfiable subset $\varphi^* \subseteq \{\varphi_j \mid j = 1,\ldots,m\}$ with $|\varphi^*| \leq p$ is Σ_{k+1}^p-complete.*

Proof. (Sketch) For membership, guess a subset φ^* and verify using a Π_k^p-oracle that $\exists X_1 \forall X_2 \cdots Q_k X_k.\varphi^*$ is false. For hardness, we reduce from the Σ_{k+1}^p-complete problem of deciding whether a QBF $\Psi = \exists X_1 \forall X_2 \cdots Q_{k+1} X_{k+1}.\psi$ is true. We may assume w.l.o.g. that ψ is in conjunctive normal form (CNF) if k is even ($Q_{k+1} = \exists$), and in disjunctive normal form (DNF) if k is odd ($Q_{k+1} = \forall$). For our reduction, we adapt (and simplify) the reduction for the propositional case [20]. Let $X_1 = \{x_1,\ldots,x_n\}$ and declare variables $P = \{p_i \mid i = 1,\ldots,n\}$ and $N = \{n_i \mid i = 1,\ldots,n\}$. Let ψ' be the formula obtained from ψ by replacing each literal x_i with p_i and $\neg x_i$ with n_i. Finally, consider $\varphi = \bigwedge_{i=1}^{n}(p_i \vee n_i) \rightarrow \neg\psi'$.

It holds that $\Phi = \exists(P \cup N \cup X_2)\forall X_3 \cdots Q_{k+1}X_{k+1}.\varphi \wedge \bigwedge_{i=1}^{n} p_i \wedge \bigwedge_{i=1}^{n} n_i$ has an unsatisfiable subset of size at most $n+1$ iff Ψ is true. Intuitively, a solution of Ψ gives rise to an unsatisfiable subset of Φ containing φ and exactly one p_i or n_i for each $i = 1, \dots, n$. On the other hand, all unsatisfiable subsets of Φ must contain φ, and due to the bound $n+1$, exactly one p_i or n_i for each $i = 1, \dots, n$, which simulates a truth assignment which is a solution of Ψ. $\square$

Note that for even (resp. odd) k, the reduction gives hardness for k-QBFs of form $\exists X \overrightarrow{Q}_{k-1}.\varphi \wedge S$ where $S \subset X$ and φ is in DNF (resp. CNF—to see this, consider additional variables for each disjunct $p_i \vee n_i$ in φ). This is in line with SMUSes of QBFs being subsets of the first existential quantifier block. Interestingly, there is a difference between the complexity of computing a SMUS in the case $Q_1 = \forall$ and in the case $Q_1 = \exists$. For $Q_1 = \forall$ the problem turns out to be merely Σ_k^p-complete. This is because a nondeterministic guess may contain both an unsatisfiable subset candidate and a counterexample assignment.

Proposition 3. *Consider a QBF* $\forall X_1 \exists X_2 \cdots Q_k X_k. \bigwedge_{j=1}^{m} \varphi_j$*, where* φ_j *are propositional formulas over* $\bigcup_{i=1}^{k} X_i$*. Deciding whether there is an unsatisfiable subset* $\varphi^* \subseteq \{\varphi_j \mid j = 1, \dots, m\}$ *with* $|\varphi^*| \leq p$ *is* Σ_k^p*-complete.*

Proof. For membership, guess a subset φ^* and a counterexample τ to the QBF. Verify using a Π_{k-1}^p oracle that $\exists X_2 \cdots Q_k X_k.\varphi^*[\tau]$ is false. Hardness follows by a reduction from the Σ_k^p-complete problem of deciding whether a QBF $\exists X_1 \forall X_2 \cdots Q_k X_k.\varphi$ is true (consider the negation). $\square$

5 Computing Smallest MUSes via Implicit Hitting Sets

`SMUS-IHS`, the implicit-hitting set based approach for computing a SMUS of a given QBF $\Phi = \exists S \overrightarrow{Q}_k.\varphi$ is detailed in Algorithm 1. The algorithm works by

`SMUS-IHS`
- **Input:** A QBF $\Phi = \exists S \overrightarrow{Q}_k.\varphi$
- **Output:** A SMUS $S^\star \subset S$ of Φ
- $(\tau, \mathit{true?}) \leftarrow$ `QBF-Solve`(Φ, S);
- **if** *true?* **then**
  - **return** "no cores";
- $UB \leftarrow |S|$; $LB \leftarrow 0$;
- $S^\star \leftarrow S$; $\mathcal{C} \leftarrow \emptyset$;
- **while** TRUE **do**
  - $(hs, \mathit{opt?}) \leftarrow$ `Min-Hs`$(\mathcal{C}, S, UB)$;
  - **if** *opt?* **then** $LB \leftarrow |hs|$;
  - **if** $LB = UB$ **then break**;
  - $\mathcal{C} \leftarrow \mathcal{C} \cup$ `Extract-MCS`$(S^\star, UB, \Phi, S)$;
  - **if** $LB = UB$ **then break**;
- **return** $S^\star$;

Algorithm 1: Computing a QBF SMUS

`Min-Hs`$(\mathcal{C}, S, UB)$:

minimize: $\sum_{s \in S} s$

subject to:

$$\sum_{s \in cs} s \geq 1 \quad \forall cs \in \mathcal{C}$$
$$s \in \{0,1\} \quad \forall s \in S$$

return:

$\{s \mid s \text{ set to 1 in opt. soln}\}$

Fig. 1. Hitting set IP

```
Extract-MCS(S*, hs, UB, Φ, S)
    A = hs, C_n = ∅;
    while TRUE do
        (τ, true?) ←
          QBF-Solve(Φ, A);
        if not true? then
            if |A| < UB then
                UB ← |A|; S* ← A;
            return C_n;
        else
            cs ←
              MinCS(τ, UB, S*, Φ, S);
            C_n ← C_n ∪ {cs};
            A ← A ∪ {cs};
```

```
MinCS(τ, UB, S*, Φ, S)
    A = {s ∈ S | τ(s) = 1};
    for s ∈ S \ A do
        if s ∈ A then continue;
        (τ, true?) ←
          QBF-Solve(Φ, A ∪ {s});
        if not true? then
            if |A| + 1 < UB then
                UB ← |A| + 1;
                S* ← A ∪ {s};
        else
            A ← {s ∈ S | τ(s) = 1};
    return S \ A;
```

Fig. 2. Extracting (left) and minimizing (right) correction sets of a QBF.

iteratively refining a lower and upper bound LB and UB on the size of the SMUSes of Φ. The lower bounds are obtained by extracting an increasing collection $\mathcal{C}$ of correction sets of Φ with a QBF oracle in the `Extract-MCS` subroutine, and computing hitting sets hs over them with an integer programming solver, in the `Min-Hs` subroutine. The correction set extraction subroutine also obtains unsatisfiable cores of Φ, the smallest core found at any point is stored in $S^\star$ and the upper bound UB set to $UB = |S^\star|$. The search terminates when $UB = LB$ and returns $S^\star$ which at that point is known to be a SMUS.[2]

We abstract the use of a QBF oracle into the function `QBF-Solve`. Given a subset $S_s \subset S$, the call `QBF-Solve`(Φ, S_s) returns a tuple $(\tau, \mathit{true?})$ where *true?* is true iff $\exists S \overrightarrow{Q}_k.\varphi\,[S_s]$ is true. In the affirmative case the oracle returns a solution τ to Φ that sets $\tau(s) = 1$ for all $s \in S_s$. A useful intuition here is that if the QBF oracle returns true, then the set $cs = (S \setminus \{s \mid \tau(s) = 1\}) \supset (S \setminus S_s)$ is a correction set of Φ. Similarly, if the result is false, then S_s is a core of Φ and $|S_s|$ an upper bound on the size of the SMUSes.

More specifically, given an input QBF $\Phi = \exists S \overrightarrow{Q}_k.\varphi$, `SMUS-IHS` begins by checking that the QBF has no solutions by invoking `QBF-Solve`(Φ, S) on Line 2. If the result is true, then there are no cores (and as such no SMUSes) of Φ so the search terminates on Line 4. Otherwise, S is a core of Φ, so the upper bound UB is set to $|S|$, the smallest known core $S^\star$ to S, the set $\mathcal{C}$ of correction sets to $\emptyset$, and the lower bound LB on the size of the SMUSes to 0 (Lines 5 and 6).

Each iteration of the main search loop (Lines 7-12) starts by computing a hitting set hs over the collection $\mathcal{C}$ of correction sets extracted so far. The procedure `Min-Hs` on Line 8 computes an incumbent solution hs to the integer program representation of the hitting set problem detailed in Figure 1. The solution either (a) is optimal, i.e., represents a minimum-cost hitting set or

[2] Note that by employing integer programming our approach also allows for computing weighted SMUSes, i.e., cores with smallest total weight over their elements.

(b) has $|hs| < UB$. In addition to hs, the procedure returns an indicator *opt?* on whether hs is minimum-cost. If it is, then by Proposition 1 $|hs|$ is a lower bound on the size of the SMUSes, so the LB is updated on Line 9 and the termination criterion ($UB = LB$) checked on Line 10. If $UB > LB$, the procedure `Extract-MCS` next extracts correction sets of Φ that do not intersect with hs. In addition to new correction sets, the procedure will also compute new unsatisfiable cores of the instance, thereby potentially tightening the upper bound UB, which is why the termination criterion is checked again on Line 12 before the loop is reiterated. An important note here is that no correction sets are ever removed from $\mathcal{C}$ so the sequence of LB values will be increasing.

The procedure `Extract-MCS` (Figure 2, left) computes MCSes that do not intersect with hs by using a QBF oracle. The procedure maintains a subset $\mathcal{A} \subset S$ (initialized to hs) and iteratively invokes the QBF oracle by calling `QBF-Solve`$(\Phi, \mathcal{A})$. If the result is false, the set $\mathcal{A}$ is a core of Φ, so the procedure checks whether the upper bound can be improved before terminating and returning the set $\mathcal{C}_n$ of new corrections sets extracted. Otherwise (i.e., if the result is true) the oracle also returns a solution τ to Φ that sets $\tau(s) = 1$ for each $s \in \mathcal{A}$. Since $hs \subset \mathcal{A}$ holds in each iteration of `Extract-MCS`, the set $S \setminus \{s \mid \tau(s) = 1\}$ is a correction set of Φ that does not intersect with hs. The correction set is then minimized in the `MinCS` procedure (Figure 2, right) by repeated queries to the QBF oracle, each asking for a solution that sets at least one more variable in $S \setminus \{s \mid \tau(s) = 1\}$ to true. The minimization procedure ends when the oracle reports false. Then a new core of Φ is also obtained, potentially allowing the upper bound to be tightened. The minimized cs is added to the set $\mathcal{C}_n$ of new correction sets and to $\mathcal{A}$ to prevent it from being rediscovered.

The following proposition establishes the correctness of `SMUS-IHS`.

Proposition 4. *On input $\Phi = \exists S \overrightarrow{Q}_k . \varphi$,* `SMUS-IHS` *terminates and returns a SMUS $S^\star$ of Φ.*

Proof. Subject to termination, $S^\star$ is a subset of S for which $\exists S \overrightarrow{Q}_k . \varphi\,[S^\star]$ is false (since the set $S^\star$ is only updated after the QBF oracle reports false) and $|S^\star| = |hs|$ for some minimum-cost hitting set hs over a set of correction sets of Φ. Termination follows by the finite number of correction sets of Φ and the fact that each hitting set hs is computed at most twice during the execution of the algorithm. More precisely, consider a hitting set hs returned by `Min-Hs`. In the next invocation of `Extract-MCS` either (i) a new correction set cs for which $cs \cap hs = \emptyset$ is computed, or (ii) the set hs is shown to be a core of Φ. In case (i) cs is added to $\mathcal{C}$, preventing hs from being recomputed in subsequent iterations. In case (ii) `SMUS-IHS` will either terminate on Line 12 if $LB = |hs|$ (i.e., we know hs is minimum-cost), or compute a new hitting set hs' that is either a minimum-cost hitting set over $\mathcal{C}$ or has $|hs'| < UB \leq |hs|$. That is, the only way in which hs can be recomputed in subsequent iterations is if it was of minimum cost in which case the algorithm terminates after computing hs for a second time. □

The proof of Proposition 4 is similar to a correctness proof of IHS for MaxSAT [3]. Note that the correctness of `SMUS-IHS` does not rely on correction sets being minimal or the extraction of all disjoint correction sets at each iteration; as long as the loop on Lines 3-12 is executed at least once on each invocation of `Extract-MCS`, the algorithm will either compute a previously unseen correction set, or be able to determine that the input hitting set is a SMUS. Similarly, the minimization of the correction sets need not be exhaustive. The set $S \setminus \mathcal{A}$ will be a correction set of Φ after each iteration of the loop in `MinCS`.

Dual IHS for leading universal quantifier. A way of employing `SMUS-IHS` for computing a SMUS of $\forall X_1 \overrightarrow{Q}_{k-1}. \bigwedge_{j=1}^{m} \varphi_j$ over $\{\varphi_1, \ldots, \varphi_m\}$ is to give the $(k+1)$-QBF $\Phi_{k+1} = \exists S \forall X_1 \overrightarrow{Q}_{k-1}. \bigwedge_{j=1}^{m} (s_j \to \varphi_j)$ as input. For an alternative—more inline with the complexity results of Proposition 3—approach we can instead consider the k-QBF $\Phi_k = \forall S \forall X_1 \overrightarrow{Q}_{k-1}. \bigwedge_{j=1}^{m} (s_j \to \varphi_j)$. For any $S^* \subset \neg S = \{\neg s \mid s \in S\}$, $\Phi_k [S^*]$ consists of the same formulas as Φ_k except for the ones corresponding to S^*, which are essentially deactivated. Thus we may define for a QBF $\Phi = \forall S \overrightarrow{Q}_k.\varphi$ that a core is a set $S^* \subset \neg S$ for which $\forall S \overrightarrow{Q}_k.\varphi [(\neg S) \setminus S^*]$ is false, and that a correction set $cs \subset \neg S$ makes $\forall S \overrightarrow{Q}_k.\varphi [cs]$ true.

This leads to a dual IHS algorithm. First, check for the existence of a core by a call to `QBF-Solve`$(\Phi_k, \emptyset)$. If the oracle reports true, exit. Else we obtain a counterexample assignment to S, giving an upper bound as the number of variables in S set to true. Similarly to `SMUS-IHS`, we obtain lower bounds by computing minimum-cost hitting sets over collections of (now the dual notion of) correction sets. A correction set is now extracted by calling `QBF-Solve`$(\Phi_k, \neg(S \setminus hs))$. A true result implies that $\neg(S \setminus hs)$ is a correction set which is then minimized similarly as in `SMUS-IHS`. Further, some modern QBF oracles are able to provide a subset $\mathcal{A}' \subset \neg(S \setminus hs)$ used to prove the absence of a counterexample. Such $\mathcal{A}'$ can directly be used as a correction set. Upper bounds on the size of SMUSes are obtained via the oracle reporting false and providing a counterexample assignment. Note that the dual algorithm can be applied for QBFs of form $\exists X_1 \overrightarrow{Q}_{k-1} \bigwedge_{j=1}^{m} \varphi_j$ by giving $\forall S \exists X_1 \overrightarrow{Q}_{k-1}. \bigwedge_{j=1}^{m} (s_j \to \varphi_j)$ as input.

6 Empirical Evaluation

We implemented the `SMUS-IHS` algorithm; the implementation is available in open source at https://bitbucket.org/coreo-group/qbf-smuser. Since no direct competitors are available, we demonstrate the feasibility of the approach for computing smallest explanations in abstract argumentation, as well as in the more general context of extracting clausal SMUSes from QBF instances. We use CPLEX as the minimum-cost hitting set problem IP solver. As choices for the QBF solver, we consider DepQBF (version 6.0.3) [24] and RAReQS (version 1.1) [18]. DepQBF is a search-based QDPLL solver with conflict-driven clause learning and solution-driven cube learning, providing an incremental interface for extracting assignments and unsatisfiable cores and solving under user-provided assumption literals [22, 23]. RAReQS is an expansion-based CEGAR solver, iteratively SAT solving and refining a propositional abstraction. We modified RAReQS to extract unsatisfiable cores from the top-level SAT solver. We consider the following variants of the `SMUS-IHS` algorithm.

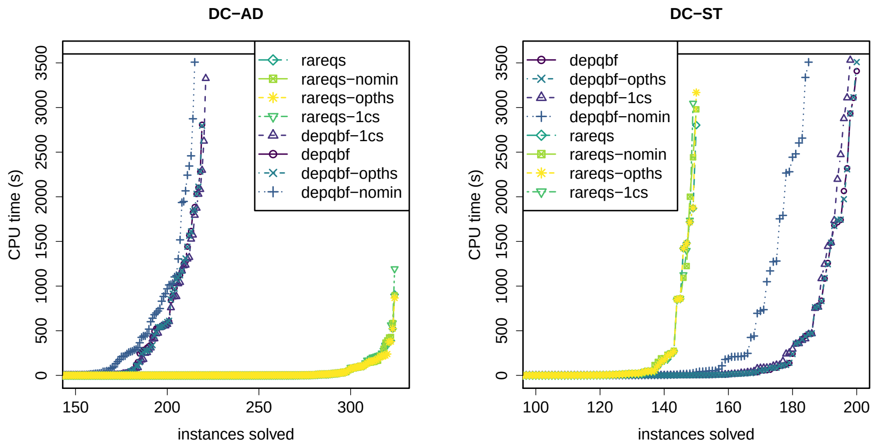


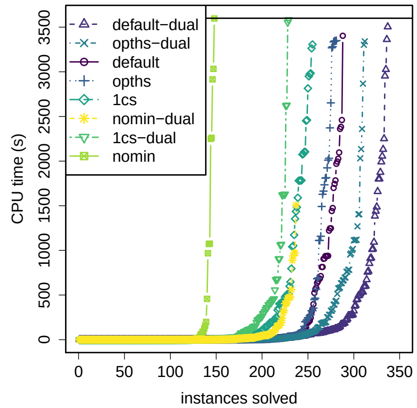
Fig. 3. Number of solved instances: strong explanations for credulous acceptance under admissible (left) and stable (right).

– S (default): S as the QBF solver, extracting all MCSes at each iteration, i.e., executing `Extract-MCS` until unsatisfiability, and calling `MinCS` in `Extract-MCS`.
– S-1cs: S, extracting at most one MCS per invocation of `Extract-MCS`.
– S-noMin: S without correction set minimization.
– S-optHS: S, computing minimum-cost hitting sets at each iteration.

Experiments were run under per-instance 3600-s time and 16-GB memory limit with Intel Xeon E5-2670 CPUs, 57-GB memory, RHEL 8.5 and GCC 8.5.0.

To obtain benchmarks for computing strong explanations in argumentation frameworks, we extended the implementation from [27] to output the negations of encodings described in Section 3 in QDIMACS format. The updated version is available at https://bitbucket.org/andreasniskanen/selitae/. As input AFs, we used the set of 326 AFs from ICCMA'19 (http://argumentationcompetition.org/2019/). We consider three tasks: computing smallest strong explanations for credulous acceptance under admissible and stable semantics and for skeptical rejection under stable semantics. For each AF, a query argument was picked uniformly at random from the set of credulously accepted arguments or skeptically rejected arguments. This gave 324 AF-query pairs for admissible semantics and 312 AF-query pairs for stable semantics (there were 2 AFs which have no non-empty admissible extensions and 14 AFs without a stable extension).

The runtime results for computing smallest explanations for credulous acceptance under admissible and stable are summarized in Figure 3. On credulous admissible (left), RAReQS as the QBF solver yields clearly the best results: all algorithmic variants except for 1cs solve 324 instances under 1000 seconds. In contrast, using DepQBF results in solving only 221 instances using the configuration 1cs. On credulous stable (right), using RAReQS results in solving 150



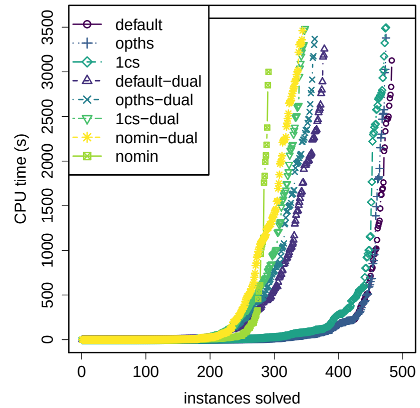


Fig. 4. Number of solved instances for different `SMUS-IHS` variants on 2-QBFs (left) and 3-QBFs (right) using RAReQS as the QBF oracle.

instances for each `SMUS-IHS` variant except 1cs. DepQBF results in clearly better performance, allowing for solving 200 instances using the default and optHS configurations. Interestingly, correction set minimization is an important factor for runtime efficiency when using DepQBF on these instances. The results for skeptical rejection under stable are similar: using DepQBF results in better performance, but the difference due to the choice of the QBF solver is not as drastic (with 181 solved instances using DepQBF and 156 using RAReQS).

To demonstrate more general applicability of `SMUS-IHS`, we also consider computing SMUSes of QBFs in CNF form, for the relatively small unsatisfiable QBFLIB (http://www.qbflib.org/) benchmarks encoding reduction finding [19] using RAReQS which has exhibited good performance for deciding satisfiability in this domain. We discarded instances for which RAReQS on its own took more than one second to decide unsatisfiability, leaving 719 2-QBF and 905 3-QBF instances. For each instance, each clause C in the matrix φ was replaced by $s_C \rightarrow C$, where s_C is a fresh variable. Finally, the quantifier QS with $S = \{s_C \mid C \in \varphi\}$ was appended as the outermost quantifier in the prefix either with $Q = \exists$ for the `SMUS-IHS` algorithm or with $Q = \forall$ for dual `SMUS-IHS`.

The results are shown in Figure 4. For 2-QBF instances (left) we observe that the dual algorithm outperforms other solver variants if several MCSes are extracted at each iteration. The default configuration solves more instances than optHS-dual which computes minimum-cost hitting sets. Disabling either minimization (noMin-dual) or exhaustive MCS extraction (1cs-dual) leads to a noticeable loss in performance. The non-dual variants are not as effective, which is in line with the fact that their input is a 3-QBF. For 3-QBF instances (right) the default, optHS and 1cs configurations clearly outperform all other configurations, with slight performance improvements obtained by using non-optimal hitting sets and exhaustive MCS extraction. Here the dual variants are less competitive; their input is a 4-QBF, since the original 3-QBF has an $\exists\forall\exists$ prefix.

7 Conclusions

We proposed an approach to computing smallest unsatisfiable subsets of quantified Boolean formulas, and pinpointed the complexity of deciding if a k-QBF (for arbitrary k) has a small unsatisfiable subset. While the approach is generally applicable to computing SMUSes of QBFs, we detailed an application in computing smallest strong explanations for credulous acceptance and skeptical rejection in abstract argumentation. Our implementation allows for computing smallest strong explanations of standard ICCMA argumentation competition benchmarks in practice. This suggests studying further applications of the approach to other non-monotonic formalisms admitting QBF encodings. The exact complexity of computing smallest strong explanations remains a further open question.

References

1. Alfano, G., Calautti, M., Greco, S., Parisi, F., Trubitsyna, I.: Explainable acceptance in probabilistic abstract argumentation: Complexity and approximation. In: KR, pp. 33–43 (2020)
2. Arieli, O., Caminada, M.W.A.: A QBF-based formalization of abstract argumentation semantics. J. Appl. Log. **11**(2), 229–252 (2013)
3. Bacchus, F., Hyttinen, A., Järvisalo, M., Saikko, P.: Reduced cost fixing in MaxSAT. In: CP, *LNCS*, vol. 10416, pp. 641–651. Springer (2017)
4. Bacchus, F., Katsirelos, G.: Using minimal correction sets to more efficiently compute minimal unsatisfiable sets. In: CAV, *LNCS*, vol. 9207, pp. 70–86. Springer (2015)
5. Baumann, R., Ulbricht, M.: Choices and their consequences - explaining acceptable sets in abstract argumentation frameworks. In: KR, pp. 110–119 (2021)
6. Baumeister, D., Järvisalo, M., Neugebauer, D., Niskanen, A., Rothe, J.: Acceptance in incomplete argumentation frameworks. Artif. Intell. **295**, 103,470 (2021)
7. Belov, A., Marques-Silva, J.: Accelerating MUS extraction with recursive model rotation. In: FMCAD, pp. 37–40. FMCAD Inc. (2011)
8. Bendík, J., Cerná, I.: Replication-guided enumeration of minimal unsatisfiable subsets. In: CP, *LNCS*, vol. 12333, pp. 37–54. Springer (2020)
9. Besnard, P., Doutre, S.: Checking the acceptability of a set of arguments. In: NMR, pp. 59–64 (2004)
10. Brewka, G., Thimm, M., Ulbricht, M.: Strong inconsistency. Artif. Intell. **267**, 78–117 (2019)
11. Brewka, G., Ulbricht, M.: Strong explanations for nonmonotonic reasoning. In: Description Logic, Theory Combination, and All That, *LNCS*, vol. 11560, pp. 135–146. Springer (2019)
12. Delisle, E., Bacchus, F.: Solving weighted CSPs by successive relaxations. In: CP, *LNCS*, vol. 8124, pp. 273–281. Springer (2013)
13. Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artif. Intell. **77**(2), 321–358 (1995)
14. Egly, U., Woltran, S.: Reasoning in argumentation frameworks using quantified boolean formulas. In: COMMA, *FAIA*, vol. 144, pp. 133–144. IOS Press (2006)
15. Fan, X., Toni, F.: On computing explanations in argumentation. In: AAAI, pp. 1496–1502. AAAI Press (2015)

16. Ignatiev, A., Janota, M., Marques-Silva, J.: Quantified maximum satisfiability. Constraints **21**(2), 277–302 (2016)
17. Ignatiev, A., Previti, A., Liffiton, M.H., Marques-Silva, J.: Smallest MUS extraction with minimal hitting set dualization. In: CP, *LNCS*, vol. 9255, pp. 173–182. Springer (2015)
18. Janota, M., Klieber, W., Marques-Silva, J., Clarke, E.M.: Solving QBF with counterexample guided refinement. Artif. Intell. **234**, 1–25 (2016)
19. Jordan, C., Kaiser, L.: Experiments with reduction finding. In: SAT, *LNCS*, vol. 7962, pp. 192–207. Springer (2013)
20. Liberatore, P.: Redundancy in logic I: CNF propositional formulae. Artif. Intell. **163**(2), 203–232 (2005)
21. Liffiton, M.H., Mneimneh, M.N., Lynce, I., Andraus, Z.S., Marques-Silva, J., Sakallah, K.A.: A branch and bound algorithm for extracting smallest minimal unsatisfiable subformulas. Constraints **14**(4), 415–442 (2009)
22. Lonsing, F., Egly, U.: Incremental QBF solving. In: CP, *LNCS*, vol. 8656, pp. 514–530. Springer (2014)
23. Lonsing, F., Egly, U.: Incrementally computing minimal unsatisfiable cores of QBFs via a clause group solver API. In: SAT, *LNCS*, vol. 9340, pp. 191–198. Springer (2015)
24. Lonsing, F., Egly, U.: DepQBF 6.0: A search-based QBF solver beyond traditional QCDCL. In: CADE, *LNCS*, vol. 10395, pp. 371–384. Springer (2017)
25. Marques-Silva, J., Mencía, C.: Reasoning about inconsistent formulas. In: IJCAI, pp. 4899–4906. ijcai.org (2020)
26. Mencía, C., Marques-Silva, J.: Reasoning about strong inconsistency in ASP. In: SAT, *LNCS*, vol. 12178, pp. 332–342. Springer (2020)
27. Niskanen, A., Järvisalo, M.: Smallest explanations and diagnoses of rejection in abstract argumentation. In: KR, pp. 667–671 (2020)
28. Papadimitriou, C.H.: Computational complexity. Addison-Wesley (1994)
29. Papadimitriou, C.H., Wolfe, D.: The complexity of facets resolved. J. Comput. Syst. Sci. **37**(1), 2–13 (1988)
30. Reiter, R.: A theory of diagnosis from first principles. Artif. Intell. **32**(1), 57–95 (1987)
31. Saikko, P., Dodaro, C., Alviano, M., Järvisalo, M.: A hybrid approach to optimization in answer set programming. In: KR, pp. 32–41. AAAI Press (2018)
32. Saikko, P., Wallner, J.P., Järvisalo, M.: Implicit hitting set algorithms for reasoning beyond NP. In: KR, pp. 104–113. AAAI Press (2016)
33. Sakama, C.: Abduction in argumentation frameworks. J. Appl. Non Class. Logics **28**(2-3), 218–239 (2018)
34. Saribatur, Z.G., Wallner, J.P., Woltran, S.: Explaining non-acceptability in abstract argumentation. In: ECAI, *FAIA*, vol. 325, pp. 881–888. IOS Press (2020)
35. Ulbricht, M., Baumann, R.: If nothing is accepted - repairing argumentation frameworks. J. Artif. Intell. Res. **66**, 1099–1145 (2019)
36. Ulbricht, M., Thimm, M., Brewka, G.: Handling and measuring inconsistency in non-monotonic logics. Artif. Intell. **286**, 103,344 (2020)
37. Ulbricht, M., Wallner, J.P.: Strong explanations in abstract argumentation. In: AAAI, pp. 6496–6504. AAAI Press (2021)