I N F S Y S

R E S E A R C H

R E P O R T

INSTITUT FÜR INFORMATIONSSYSTEME

ABTEILUNG WISSENSBASIERTE SYSTEME

ON DECIDING SUBSUMPTION PROBLEMS

Uwe EGLY Reinhard PICHLER Stefan WOLTRAN

INFSYS RESEARCH REPORT 1843-03-04

APRIL 2003

Institut für Informationssysteme
Abtg. Wissensbasierte Systeme
Technische Universität Wien
Favoritenstraße 9-11
A-1040 Wien, Austria
Tel: +43-1-58801-18405
Fax: +43-1-58801-18493
sek@kr.tuwien.ac.at
www.kr.tuwien.ac.at

TECHNISCHE UNIVERSITÄT WIEN

INFSYS RESEARCH REPORT
INFSYS RESEARCH REPORT 1843-03-04, APRIL 2003

ON DECIDING SUBSUMPTION PROBLEMS

Uwe Egly[1], Reinhard Pichler[2], and Stefan Woltran[1]

Abstract. Subsumption is an important redundancy elimination method in automated deduction. A clause D is subsumed by a set $\mathcal{C}$ of clauses if there is a clause $C \in \mathcal{C}$ and a substitution σ such that the literals of $C\sigma$ are included in D. In the field of automated model building, subsumption has been modified to an even stronger redundancy elimination method, namely the so-called clausal H-subsumption. Atomic H-subsumption emerges from clausal H-subsumption by restricting D to an atom and $\mathcal{C}$ to a set of atoms. Both clausal and atomic H-subsumption play an indispensable key role in automated model building. Moreover, problems equivalent to atomic H-subsumption have been studied with different terminologies in many areas of computer science. Both clausal and atomic H-subsumption are known to be intractable, i.e., Π_2^p-complete and NP-complete, respectively. In this paper, we present a new approach to deciding (clausal and atomic) H-subsumption that is based on a reduction to QSAT_2 and SAT, respectively.

[1]Institut für Informationssysteme, Abteilung Wissensbasierte Systeme 184/3, Technische Universität Wien, Favoritenstraße 9-11, A–1040 Wien, Austria. E-mail: [uwe, stefan]@kr.tuwien.ac.at
[2]Siemens AG Österreich, Siemensstr. 92, 1210 Vienna, AUSTRIA

Acknowledgements: This work was partially supported by the Austrian Science Fund Project under grant P15068.

A preliminary version of this paper appeared in the *Proceedings of the Fifth International Symposium on the Theory and Applications of Satisfiability Testing* (*SAT-02*).

Contents

1 Introduction

Redundancy elimination is one of the most important tasks in all kinds of deduction systems. The most prominent redundancy elimination technique is *subsumption* which is implemented in all practically successful saturation-based theorem provers. A clause D is subsumed by a set $\mathcal{C}$ of clauses, if there is a clause $C \in \mathcal{C}$ and a substitution σ such that all literals of $C\sigma$ are included in D. Although subsumption is a computationally expensive operation (see problem LO18 in [13] for the NP-completeness result), omitting it from implementations would be even worse because redundant clauses spoil the search space, preventing the theorem prover to find a proof.

In the field of automated model building [4; 3; 12], even stronger redundancy elimination techniques are required. These techniques result from ordinary subsumption by taking not only the clause D into account but all ground instances of D over a given (possibly infinite) set of terms. This set H is usually called Herbrand universe.

The important redundancy elimination techniques in the context of automated model building are *clausal H-subsumption* and *atomic H-subsumption*, both of which are described in more detail in Subsection 1.1. The latter technique emerges from the former by restricting D to an atom and $\mathcal{C}$ to a set of atoms.

It is worth noticing that, contrary to ordinary automated theorem proving, where subsumption is *not* necessary for completeness of, e.g., resolution procedures, H-subsumption is a *necessary* redundancy elimination technique for automated model building which cannot be dispensed with. Moreover, the strengthened redundancy elimination techniques have various applications in machine learning [25], logic programming [16; 24], functional programming [24], and as an extension of many familiar resolution- and paramodulation-based inference systems without destroying the refutational completeness [32].

For all of the three redundancy elimination techniques mentioned above, practically efficient algorithms are an important issue. In this paper, we focus on H-subsumption and we contribute to the search for efficient algorithms by transforming H-subsumption problems in such a way that they can be solved by theorem provers for (quantified) propositional logic, even if the Herbrand universe H is *infinite*. Such a translation approach is possible because atomic and clausal H-subsumption are known to be NP-complete and Π_2^p-complete, respectively. As a consequence, there exist polynomial translations of H-subsumption problems into prototypical problems of the corresponding complexity classes. A detailed exposition of our approach follows in the next subsection.

Our methodology benefits from the availability of efficient solvers for both the satisfiability problem of propositional logic [28; 40; 15] and the satisfiability problem of quantified propositional logic [11; 14; 26; 36]. Improvements in these solvers directly yield more efficient procedures for H-subsumption when using the presented reductions. Successful applications of reduction techniques on the basis of (quantified) propositional logic include planning problems [19; 35] and non-monotonic reasoning [9; 30; 8].

1.1 Overview of Problems and Results

In what follows, we briefly recall the definitions of the decision problems which are considered in the paper. For the description, we need the notion of an *H-ground instance* of a clause C. Such an instance results from C by consistently replacing all variables in C by elements from the given Herbrand universe H.

SUB problem (first-order subsumption).
Input: A clause set $\mathcal{C} \equiv \{C_1, \ldots, C_n\}$ and a single clause D with $C_i \equiv L_{i1} \vee \ldots \vee L_{il_i}$ and $D \equiv M_1 \vee \ldots \vee M_m$.

Question: Does $\mathcal{C}$ subsume D, i.e., does there exist a clause $C_i \in \mathcal{C}$ and a substitution ϑ on the variables occurring in C_i, such that $\{L_{i1}\vartheta, \ldots, L_{il_i}\vartheta\} \subseteq \{M_1, \ldots, M_m\}$? If this is the case, then we write $\mathcal{C} \leq_{ss} D$.

ATOM-HSUB problem (atomic H-subsumption).
Input: A tuple $(\mathcal{A}, B, H)$, where $\mathcal{A} \equiv \{A_1, \ldots, A_n\}$ is a set of atoms, B is an atom, and H is a Herbrand universe, such that $H(\mathcal{A} \cup \{B\}) \subseteq H$[1].
Question: Is every H-ground instance $B\vartheta$ of B an instance of some atom A_i? If this is the case, then we write $\mathcal{A} \leq^H_{ss} B$.

CLAUSE-HSUB problem (clausal H-subsumption).
Input: A tuple $(\mathcal{C}, D, H)$, where $\mathcal{C} \equiv \{C_1, \ldots, C_m\}$ is a set of clauses, D is a clause, and H is a Herbrand universe, such that $H(\mathcal{C} \cup \{D\}) \subseteq H$.
Question: Is every H-ground instance $D\vartheta$ of D subsumed by some clause C_i? If this is the case, then we write $\mathcal{C} \leq^H_{ss} D$.

By the definition of subsumption, $\mathcal{C} \leq_{ss} D$ implies $\mathcal{C} \leq_{ss} D\vartheta$ for any substitution ϑ over any Herbrand universe H. Hence, (first-order) subsumption implies H-subsumption for any H. But the converse is, in general, not true as the following example illustrates.

Example 1 Let $D \equiv P(u,v) \lor Q(a) \lor Q(b)$ be a clause and let $\mathcal{C} \equiv \{P(x,y) \lor Q(x) \lor Q(y)\}$ be a clause set. Then $\mathcal{C} \not\leq_{ss} D$, since there does not exist a substitution ϑ with domain $\{x, y\}$, such that all of the literals $P(x,y)\vartheta$, $Q(x)\vartheta$, and $Q(y)\vartheta$ simultaneously become literals in D. But for $H \equiv \{a, b\}$, we have $\mathcal{C} \leq^H_{ss} D$. Let $D\vartheta$ be an H-ground instance of D, i.e., $u\vartheta, v\vartheta \in \{a, b\}$. Then, by choosing $x\sigma \equiv u\vartheta$ and $y\sigma \equiv v\vartheta$, we get $\{P(x\sigma, y\sigma), Q(x\sigma), Q(y\sigma)\} \subseteq \{P(u\vartheta, v\vartheta), Q(a), Q(b)\}$. Hence $\mathcal{C} \leq_{ss} D\vartheta$ holds for any $D\vartheta$, thus yielding $\mathcal{C} \leq^H_{ss} D$.

Now let $H' \equiv \{a, b, c\}$ be a different Herbrand universe and consider the H'-ground substitution $\vartheta \equiv \{x \leftarrow c, y \leftarrow c\}$. Then the H'-ground instance $D\vartheta$ of D is not subsumed by $\mathcal{C}$. Therefore $\mathcal{C} \not\leq^{H'}_{ss} D$ holds. $\diamond$

Before we describe our reduction procedure, let us recall the computational complexity results for the H-subsumption problems because they give some insights to which classes of quantified Boolean formulas (QBFs) the different problems should be translated. Clausal H-subsumption is Π^p_2-complete (cf. [31] for the Π^p_2-hardness proof and [17] for the Π^p_2-membership proof). The co-NP-completeness of ATOM-HSUB was shown independently by several authors (cf. [21], [22], [18]).

We shall thus reduce the complementary problems co-ATOM-HSUB to SAT and co-CLAUSE-HSUB to QSAT_2[2]. From a theoretical point of view, these *polynomial* translations are optimal because they reduce an NP-complete problem to SAT and a problem complete for Σ^p_2 (which is the complementary class to Π^p_2) to QSAT_2 which is itself a Σ^p_2-complete problem [38].

In this work, *equational problems* constitute the crucial link between H-subsumption problems and (quantified) satisfiability. Consider Figure 1 for the following discussion. First of all, subsumption co-problems are naturally expressible by means of equational problems (ep). Therefore, our first step will be the translation of co-ATOM-HSUB and co-CLAUSE-HSUB (for finite as well as infinite H) to equational problems with a quantifier prefix of the form $\exists\forall$ (see Proposition 2 and Proposition 3 for the details). We then establish a polynomial mapping from equational problems of the form $\exists$ and $\exists\forall$ with *finite* universes

[1] $H(\mathcal{S})$ is the set of variable-free terms which can be constructed using all constants and function symbols occurring in $\mathcal{S}$.

[2] QSAT_2 is the set of all quantified satisfiability problems of the form $(\exists p_1 \ldots p_k)(\forall q_1 \ldots q_l)\Psi$, where Ψ is a quantifier-free formula with propositional variables in $\{p_1, \ldots, p_k, q_1, \ldots, q_l\}$.

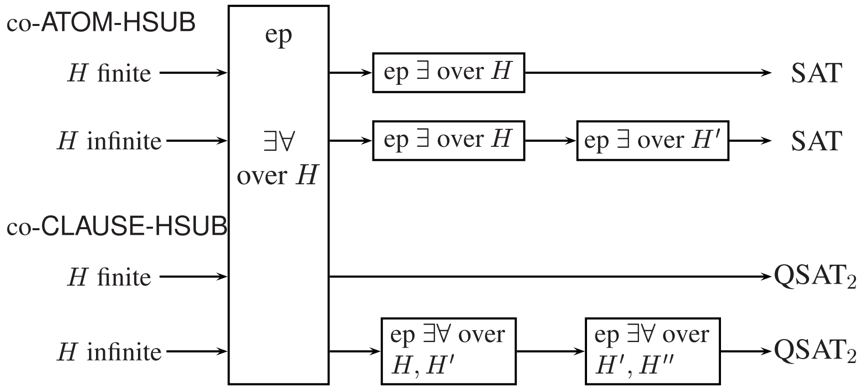

Figure 1: Overview of the translations (H' and H'' are finite).

to SAT and QSAT_2, respectively. One application of such a mapping is the translation of the co-CLAUSE-HSUB problem for finite H directly to a QBF with prefix $\exists\forall$.

For both co-ATOM-HSUB problems, we first eliminate the universal quantifiers in polynomial time. For the finite case, the resulting purely existential equational problem is translated to SAT by the mapping mentioned above. For the infinite case, we first eliminate the universal quantifiers and then derive a purely existential equational problem with *finite* H', again in polynomial time. The latter equational problem is then translated to SAT (again with the mapping mentioned above). For the infinite co-CLAUSE-HSUB problem, we have an equational problem which has two Herbrand universes. We first translate the equational problem over H into one over H and H', where H' is *finite*. Then the resulting problem is translated into an equational problem over two sorts, H' and H'', where both sorts are *finite*. Applying a generalized version of the mapping mentioned above again yields the desired QBF.

In summary, the main contributions of this paper are

- a polynomial and faithful translation of equational problems representing H-subsumption problems over *infinite* Herbrand universes into equational problems over *finite* Herbrand universes, and
- a polynomial and faithful translation of these equational problems over *finite* Herbrand universes into (quantified) propositional formulas.

The paper is organized as follows. In Section 2, we introduce basic notation, equational problems and quantified Boolean formulas. In Section 3, we discuss how H-subsumption problems over finite Herbrand universes are handled, whereas we discuss the infinite cases in Section 4. In Section 5, we show how finite equational problems can be translated to (quantified) Boolean formulas and discuss implementation issues. The appendix contains the proofs of the lemmas and theorems.

2 Preliminaries

2.1 Unification and Representation of Terms

Let V denote a set of first-order variables and let T be a set of terms. Then a *substitution* is a mapping $\lambda : V \to T$, such that $\lambda(v) \not\equiv v$ only for finitely many $v \in V$. The set of variables $v \in V$ with $\lambda(v) \not\equiv v$

is referred to as the *domain* of λ. The terms $\lambda(v)$ with $\lambda(v) \not\equiv v$ are called the *range* of λ. As a short-hand notation, we write $dom(\lambda)$ and $rg(\lambda)$, respectively. Suppose that λ is a substitution with domain $\{x_1, \dots, x_n\}$, such that $\lambda(x_i) \equiv t_i$. Then we represent λ by the set $\{x_1 \leftarrow t_1, \dots, x_n \leftarrow t_n\}$. Substitutions are applied to terms, atoms, literals and clauses in the usual way. For this purpose, we use a post-fix notation, e.g., by $t\lambda$ we denote the result of applying the substitution λ to the term t. The composition of substitutions is also denoted in post-fix notation, e.g., by $t\lambda\mu$ we denote the result of first applying λ to t and then applying μ to $t\lambda$. A substitution λ is called *ground*, if all terms in $rg(\lambda)$ are ground terms. If $rg(\lambda) \subseteq H$ for some Herbrand universe H, then λ is called an H*-ground* substitution.

Let $e_1, \dots, e_n$ be expressions, i.e., terms, atoms, literals or clauses. A substitution μ is called a *unifier* of the e_i's, if $e_1\mu \equiv \cdots \equiv e_n\mu$ holds. Moreover, μ is called *most general*, if, for any further unifier μ', there exists a substitution η, such that $x\mu' \equiv x\mu\eta$ holds for all $x \in dom(\mu)$. Note that the most general unifier is unique only up to variable renaming. However, by abuse of notation, we shall speak about *the* most general unifier, when we mean any such unifier. Moreover, as a short-hand notation, we shall write $mgu(e_1, \dots, e_n)$.

Let e be a further expression. We call e an *instance* of some e_i, iff there exists a substitution λ with $e \equiv e_i\lambda$. Moreover, e is a *common instance* of $e_1, \dots, e_n$, iff e is an instance of every e_i. Now suppose that the expressions $e_1, \dots, e_n$ are pairwise variable disjoint. Then unification can be used to represent all common instances of the e_i's as follows. If the e_i's are not unifiable, then they have no common instance. On the other hand, let μ be the mgu of the e_i's. Then we have $e_1\mu \equiv \cdots \equiv e_n\mu$ and every common instance of the e_i's is an instance of $e_1\mu$.

An algorithm is said to solve the *unification problem*, iff it decides whether two expressions are unifiable and, if so, computes the most general unifier. It is well-known that this problem can be solved in linear time (cf. [27]). An important prerequisite for an efficient unification algorithm is the appropriate representation of terms, namely as directed acyclic graphs (*dags*). Likewise, for our results in this paper, it is important to represent equational problems as dags such that each subterm of the problem occurs at most once in the dag.

Finally, we recall the notion of *positions* and *subterms* of a term t. Positions in t are words over the natural numbers, which are obtained inductively as follows:

Base case. The empty word ϵ is a position in t. It is also called the *empty position*. The subterm of t at position ϵ is the term t itself.

Induction. Let $s \equiv f(s_1, \dots, s_\alpha)$ be the subterm of t at some position p. Then, for every $i \in \{1, \dots, \alpha\}$, $p \cdot i$ (or pi, for short) is also a position in t. Moreover, the subterm of t at position $p \cdot i$ is s_i.

The length of a position is simply the length of this word. We call a position p in t *minimal* with a certain property, if p has minimal length among the positions with this property.

Now let t and t' be terms and let p be a position in both t and t'. We say that t and t' *differ* at p, iff the leading symbol of the subterm of t at p is different from the leading symbol of the subterm of t' at p. Intuitively, a position p in t describes a path in the dag-representation of a term t from the root to some node N. The subterm of t at p is simply the term corresponding to N in the dag.

2.2 Equational Problems

Equational problems are first-order formulas, whose only predicate symbol is syntactic equality "$=$". Hence, atoms of equational problems are either of the form $s = t$ for some terms s and t, or $\top$ or $\bot$. $\top$ is the trivially true problem and $\bot$ is the trivially false one, respectively. Disequations $s \neq t$ are used as a short-hand notation for negated equations $\neg(s = t)$. Moreover, it is sometimes convenient to write "double equations" $r = s = t$ as a short-hand notation for conjunctions of equations $r = s \wedge s = t$.

An equational problem $\mathcal{P}$ is usually considered over some fixed (finite or infinite) Herbrand universe H or, equivalently, over some fixed finite signature Σ consisting of function symbols of arity ≥ 0. For every $f \in \Sigma$, we denote the arity of f by $\alpha(f)$. If $\alpha(f) \equiv 0$, then f is a constant symbol. Otherwise we call f a proper function symbol.

An interpretation of $\mathcal{P}$ is given through an H-ground substitution σ, whose domain coincides with the free variables of the equational problem and whose range is (a subset of) the Herbrand universe H. The trivial problem $\top$ evaluates to "true" in every interpretation. Likewise, $\perp$ always evaluates to "false". A single equation $s = t$ is validated by an H-ground substitution σ, iff $s\sigma$ and $t\sigma$ are syntactically identical. The connectives $\wedge$, $\vee$, $\neg$, $\rightarrow$, as well as the quantifiers $\exists$ and $\forall$ are interpreted as usual. An H-ground substitution σ which validates a problem $\mathcal{P}$ is called a *solution* of $\mathcal{P}$ (over the Herbrand universe H). An equational problem is *satisfiable* over the Herbrand universe H (or, for short, H-satisfiable), iff it has at least one solution over H. Two equational problems are *equivalent* (over H) iff they possess the same solutions over H.

In what follows, we give a natural translation of the decision problems under consideration into equational problems. We start with some ancillary notion. Let P be any predicate symbol of arity n. We introduce two new function symbols f_P^+, f_P^- of the same arity. Then, L^* denotes $f_P^+(t_1, \ldots, t_n)$ if $L \equiv P(t_1, \ldots, t_n)$ and $f_P^-(t_1, \ldots, t_n)$ if $L \equiv \neg P(t_1, \ldots, t_n)$, respectively. Moreover, let $Var(\mathcal{C})$ denote the set of (free and bound) first-order variables occurring in a set $\mathcal{C}$ of formulas. For a single formula C, we write $Var(C)$ instead of $Var(\{C\})$.

Proposition 2 *Let $(\mathcal{A}, B, H)$ be an instance of the co-*ATOM-HSUB *problem with $\mathcal{A} \equiv \{A_1, \ldots, A_n\}$, $\vec{y} \equiv Var(\mathcal{A})$ and $\vec{x} \equiv Var(B)$, where $\vec{x} \cap \vec{y} \equiv \emptyset$. Then $\mathcal{A} \not\leq_{ss}^{H} B$ holds, iff the following equational problem over H is satisfiable:*

$$(\exists \vec{x}\,)(\forall \vec{y}\,) \bigwedge_{i=1}^{n} (A_i^* \neq B^*) \tag{1}$$

The correctness of this problem reduction is immediate from the definition of atomic H-subsumption, namely that $\{A_1, \ldots, A_n\} \not\leq_{ss}^{H} B$ holds, iff there exists an H-ground substitution ϑ for the variables $\vec{x}$, such that $B\vartheta$ is not an instance of any A_i, i.e., for all possible variable bindings σ of the variables $\vec{y}$ and for all $i \in \{1, \ldots, n\}$, the disequation $A_i^*\sigma \neq B^*\vartheta$ holds. Actually, a similar reduction to equational problems can be found in [4]. For clausal H-subsumption, we have

Proposition 3 *Let $(\mathcal{C}, D, H)$ be an instance of the co-*CLAUSE-HSUB *problem with $\mathcal{C} \equiv \{C_1, \ldots, C_n\}$, $C_i \equiv L_{i1} \vee \ldots \vee L_{il_i}$, and D a clause with $D \equiv M_1 \vee \ldots \vee M_m$. Moreover, let $\vec{y} \equiv Var(\mathcal{C})$ and $\vec{x} \equiv Var(D)$, where $\vec{x} \cap \vec{y} \equiv \emptyset$. Then $\mathcal{C} \not\leq_{ss}^{H} D$ holds, iff the following equational problem over H is satisfiable:*

$$(\exists \vec{x}\,)(\forall \vec{y}\,) \bigwedge_{i=1}^{n} \bigvee_{j=1}^{l_i} \bigwedge_{k=1}^{m} (L_{ij}^* \neq M_k^*) \tag{2}$$

Clearly, systems of the form (2) coincide with (1) iff $m \equiv 1$, $l_i \equiv 1$ for all $i \equiv 1 \ldots n$, and all L_{ij} and M_k are atoms.

In what follows we shall apply a straightforward simplification to such equational problems via unification (cf. [23]).

- Suppose that two terms L_{ij}^* and M_k^* are not unifiable. Then the equation $e \equiv (L_{ij}^* = M_k^*)$ is clearly "false" for any interpretation of the variables in e, i.e., $(L_{ij}^* \neq M_k^*)$ is always "true".

- On the other hand, suppose that the mgu μ of L_{ij}^* and M_k^* with

  $$\mu \equiv \{x_1 \leftarrow s_1, \ldots, x_k \leftarrow s_k, y_1 \leftarrow t_1, \ldots, y_l \leftarrow t_l\}$$

  exists. Then the disequation $d \equiv (L_{ij}^* \neq M_k^*)$ is equivalent to $x_1 \neq s_1 \vee \ldots \vee x_k \neq s_k \vee y_1 \neq t_1 \vee \ldots \vee y_l \neq t_l$.

This leads us to the following abbreviations.

Definition 4 *Let L and M be literals. We define $MGU(L, M) \equiv \top$ if L^* and M^* are not unifiable. Otherwise, if the mgu of L^* and M^* is given by $\{x_1 \leftarrow s_1, \ldots, x_k \leftarrow s_k, y_1 \leftarrow t_1, \ldots, y_l \leftarrow t_l\}$, then we set*

$$MGU(L, M) \equiv x_1 \neq s_1 \vee \ldots \vee x_k \neq s_k \vee y_1 \neq t_1 \vee \ldots \vee y_l \neq t_l.$$

Proposition 5 *Let $(\mathcal{C}, D, H)$, $\vec{x}$, and $\vec{y}$ be as in Proposition 3. Then $\mathcal{C} \not\leq_{ss}^{H} D$ holds, iff the equational problem*

$$\mathcal{P}_{\mathcal{C},D} \equiv (\exists \vec{x})(\forall \vec{y}) \bigwedge_{i=1}^{n} \bigvee_{j=1}^{l_i} \bigwedge_{k=1}^{m} MGU(L_{ij}, M_k) \tag{3}$$

is satisfiable over H.

This simplification is also applicable to the co-ATOM-HSUB problem. Observe that it is computationally inexpensive and, in general, leads to reductions with a simpler structure (and possibly with fewer variables). Later we shall show further simplifications.

Example 6 Recall the clause set $\mathcal{C}$ and the clause D from Example 1 with $\mathcal{C} \equiv \{P(x,y) \vee Q(x) \vee Q(y)\}$ and $D \equiv P(u,v) \vee Q(a) \vee Q(b)$. With Proposition 3, we get

$$\begin{aligned} \exists uv \forall xy \; \Big[& \Big(f_P^+(x,y) \neq f_P^+(u,v) \wedge f_P^+(x,y) \neq f_Q^+(a) \wedge f_P^+(x,y) \neq f_Q^+(b) \Big) \\ & \vee \Big(f_Q^+(x) \neq f_P^+(u,v) \wedge f_Q^+(x) \neq f_Q^+(a) \wedge f_Q^+(x) \neq f_Q^+(b) \Big) \\ & \vee \Big(f_Q^+(y) \neq f_P^+(u,v) \wedge f_Q^+(y) \neq f_Q^+(a) \wedge f_Q^+(y) \neq f_Q^+(b) \Big) \Big]. \end{aligned}$$

We can delete true disequations and derive the unifiers yielding

$$\mathcal{P}_{\mathcal{C},D} \equiv \exists uv \forall xy \Big((u \neq x \vee v \neq y) \vee (x \neq a \wedge x \neq b) \vee (y \neq a \wedge y \neq b) \Big).$$

Observe that, for any H, the first disjunct $u \neq x \vee v \neq y$ never evaluates to true. The other disjuncts are unsatisfiable for $H \equiv \{a, b\}$, but satisfiable, e.g., for $H' \equiv \{a, b, c\}$. Hence, for $H \equiv \{a, b\}$, we have $\mathcal{C} \leq_{ss}^{H} D$, whereas, for $H' \equiv \{a, b, c\}$, we have $\mathcal{C} \not\leq_{ss}^{H'} D$. ◇

Finally let us mention that equational problems have also been considered with terms (and, in particular, variables) of different sorts (see e.g., [6]). In Section 4.2, we shall encounter such problems over two Herbrand universes H_1 and H_2. Actually, for our purposes, the following restricted form of equational problems with terms over n sorts suffices.

Definition 7 *Let $H_1, \ldots, H_n$ be Herbrand universes over the pairwise disjoint signatures $\Sigma_1, \ldots, \Sigma_n$. Moreover, let $X_1, \ldots, X_n$ be pairwise disjoint sets of first-order variables. We say that a variable x is* associated to H_i*, iff $x \in X_i$. Likewise, a term t is associated to H_i, iff every variable in t is associated to H_i and every function symbol occurring in t is from Σ_i. An equation $s = t$ is associated to H_i, iff both terms s and t are associated to H_i. Finally, we say that an equational problem $\mathcal{P}$ is given over n sorts $H_1, \ldots, H_n$, iff every equation in $\mathcal{P}$ is associated to exactly one of the H_i's.*

Consequently, an interpretation of an equational problem $\mathcal{P}$ over different sorts is given through H_i-ground substitutions σ_i on the free variables associated to H_i for each $1 \leq i \leq n$. For bound variables, quantifications $\exists x$ and $\forall x$ have to be understood to range over that H_i which is associated to x. However, the notions of "solution" and "satisfiability" are not affected by the use of several sorts.

2.3 Quantified Boolean Formulas

Quantified Boolean formulas generalize ordinary propositional formulas by the admission of quantifications over propositional atoms. Informally, a QBF of the form $\forall p \, \exists q \, \Phi$ means that, for all truth assignments of p, there is a truth assignment of q such that Φ is true.

As usual, an occurrence of a propositional atom p in a QBF Φ is *free* iff it does not appear in the scope of a quantifier $\mathsf{Q}p$ ($\mathsf{Q} \in \{\forall, \exists\}$). By, $\Phi^p_\top$, resp. $\Phi^p_\bot$, we denote the result of uniformly substituting the free occurrences of an atom p in Φ by $\top$, resp. $\bot$.

An *interpretation* is understood as a set of propositional atoms. Given an interpretation I, the truth value, $\nu_I(\Phi)$, of a QBF Φ under I is recursively defined as follows:

1. if $\Phi \equiv \top$, then $\nu_I(\Phi) \equiv 1$;
2. if $\Phi \equiv p$ is an atom, then $\nu_I(\Phi) \equiv 1$ if $p \in I$, and $\nu_I(\Phi) \equiv 0$ otherwise;
3. if $\Phi \equiv \neg\Psi$, then $\nu_I(\Phi) \equiv 1 - \nu_I(\Psi)$;
4. if $\Phi \equiv (\Phi_1 \wedge \Phi_2)$, then $\nu_I(\Phi) \equiv min(\{\nu_I(\Phi_1), \nu_I(\Phi_2)\})$;
5. if $\Phi \equiv \forall p \, \Psi$, then $\nu_I(\Phi) \equiv \nu_I(\Psi^p_\top \wedge \Psi^p_\bot)$.

The truth conditions for $\bot$, $\vee$, $\rightarrow$, $\leftrightarrow$, and $\exists$ follow from the given ones in the usual way. Note that $\exists$ is defined similarly as in first-order logic, i.e., $\exists p \, \Psi \equiv \neg\forall p \, \neg\Psi$, for each formula Ψ. Hence,

$$\nu_I(\exists p \, \Psi) \equiv \nu_I(\Psi^p_\top \vee \Psi^p_\bot).$$

We say that Φ is *true under I* iff $\nu_I(\Phi)$ evaluates to 1, otherwise Φ is *false under I*. If Φ is true under an interpretation I, then I is called a *model* of Φ. If there exists at least one model of Φ, then Φ is said to be *satisfiable*. Observe that *closed* QBFs, i.e., QBFs with no free occurrences of atoms, are either true under every interpretation or false under every interpretation.

QBFs in prenex form lead to the prototypical decision problems for the polynomial hierarchy (cf. [29; 38]). Here, only the first two levels of this hierarchy with the problems SAT and QSAT_2, respectively, are of interest. For the SAT problem, we have to decide whether a given quantifier-free boolean formula Φ is satisfiable. For the QSAT_2 problem, we have to decide whether a QBF of the form $(\exists p_1 \ldots p_k)(\forall q_1 \ldots q_l)\Psi$ is satisfiable, where Ψ is a quantifier-free formula with propositional variables in $\{p_1, \ldots, p_k, q_1, \ldots, q_l\}$.

(R_1)	$z = t \wedge P$	$\rightarrow$	$z = t \wedge P\{z \leftarrow t\}$
(R_2)	$z \neq t \vee P$	$\rightarrow$	$z \neq t \vee P\{z \leftarrow t\}$
(U_2)	$(\forall \vec{y})[P \wedge (y \neq t \vee R)]$	$\rightarrow$	$(\forall \vec{y})[P \wedge R\{y \leftarrow t\}]$
	if $y \in \vec{y}$ and $y \notin Var(t)$.		
(E)	$(\forall \vec{y})P$	$\rightarrow$	$(\exists \vec{w})(\forall \vec{y})\,[P \wedge \bigvee_{f\in\Sigma} x_j = f(w_1, \ldots, w_{\alpha(f)})]$
	where the w_i's are fresh and pairwise distinct variables.		

Figure 2: Slightly modified rule system of Comon and Lescanne.

3 H-Subsumption over a Finite Universe

Our ultimate goal is to reduce the problems co-ATOM-HSUB and co-CLAUSE-HSUB to (quantified) propositional logic. More specifically, we shall transform the NP-complete problem co-ATOM-HSUB into the SAT problem and the Σ_2^p-complete problem co-CLAUSE-HSUB into the QSAT_2 problem. In Section 2.2, we discussed a transformation of these problems into equational problems of an appropriate form.

In this section, we restrict ourselves to the case where the universe H, over which co-ATOM-HSUB and co-CLAUSE-HSUB are considered, is *finite*. Actually, as far as the co-CLAUSE-HSUB problem is concerned, we are essentially contented with the equational problems of the form (3) in Proposition 5. Referring to Figure 1, equational problems of this form are *directly* translated into the QSAT_2 problem. This translation is discussed in Section 5.

Recall that the co-ATOM-HSUB problem is initially translated into an equational problem as follows. Let $A_1, \ldots, A_n, B$ be atoms, let $\mathcal{A} \equiv \{A_1, \ldots, A_n\}$, and let $\vec{x} \equiv Var(B)$ and $\vec{y} \equiv Var(\mathcal{A})$, where $\vec{x} \cap \vec{y} \equiv \emptyset$. Then $\mathcal{A} \not\leq_{ss}^H B$ holds, iff the equational problem

$$\mathcal{P} \equiv (\exists \vec{x})(\forall \vec{y}) \bigwedge_{i=1}^{n} MGU(A_i, B)$$

is satisfiable. Unfortunately, the form of $\mathcal{P}$ is not fully satisfactory due to the quantifier prefix $\exists\forall$. We shall thus provide an easy transformation of $\mathcal{P}$ into an equivalent problem with a purely existential prefix.

This transformation as well as the one in Section 4 will be essentially based on several transformation rules of [7], which are recalled in Figure 2. Note that many more rules of [7] (like the decomposition rule, the clash rule, the occur check, etc.), which are not mentioned explicitly here, are "hidden" in the unification steps. Additionally, we use a slightly modified form of the explosion rule which is derivable from the original one by quantifier rules and distributivity. We remark that we sometimes apply the rules R_1 and R_2 in the reverse direction. This is correct because the left-hand side and the right-hand side of these rules are equivalent.

Eliminating the universal quantifiers from an equational problem $\mathcal{P} \equiv (\exists \vec{x})(\forall \vec{y}) \bigwedge_{i=1}^{n} MGU(A_i, B)$ can be done as follows. First we shift the universal quantifiers inside the conjunction. We thus get as result $(\exists \vec{x}) \bigwedge_{i=1}^{n} (\forall \vec{y}) MGU(A_i, B)$. Now recall that each conjunct

$$MGU(A_i, B) \equiv x_1 \neq s_1 \vee \ldots \vee x_k \neq s_k \vee y_1 \neq t_1 \vee \ldots \vee y_l \neq t_l \tag{4}$$

is derived from an *mgu*

$$mu \equiv \{x_1 \leftarrow s_1, \ldots, x_k \leftarrow s_k, y_1 \leftarrow t_1, \ldots, y_l \leftarrow t_l\}$$

for A_i and B. Since we deal here with finite H, the atoms A_i and B do not contain function symbols. Moreover, w.l.o.g., we may assume that none of the terms s_m is a universally quantified variable (since, otherwise, we can simply choose the variable binding $s_m \leftarrow x_m$ rather than $x_m \leftarrow s_m$ when constructing μ). But then, by applying the U_2-rule l-times, we may replace $MGU(A_i, B)$ by

$$MGU^+(A_i, B) \equiv x_1 \neq s_1 \vee \ldots \vee x_k \neq s_k. \tag{5}$$

Observe that y_m occurs only once in $MGU(A_i, B)$ (this is guaranteed by unification) and therefore, an application of a substitution of the form $y_m \leftarrow t_m$ has no effect. Finally, the universal quantifiers may of course be deleted, since the variables y_j do not occur in $MGU^+(A_i, B)$ any more. We thus get

Theorem 8 *Let $(\mathcal{A}, B, H)$ be an instance of the co-*ATOM-HSUB *problem where H is finite, with $\mathcal{A} \equiv \{A_1, \ldots, A_n\}$ and $\vec{y} \equiv Var(\mathcal{A})$, $\vec{x} \equiv Var(B)$, and $\vec{x} \cap \vec{y} \equiv \emptyset$. Moreover, let $MGU^+(A_i, B)$ (for $i \in \{1, \ldots, n\}$) be defined as in (5). Then, $\{A_1, \ldots, A_n\} \not\leq_{ss}^H B$ holds, iff the following equational problem over H is satisfiable:*

$$\mathcal{P}^+_{\mathcal{A},B} \equiv (\exists \vec{x}\,) \bigwedge_{i=1}^{n} MGU^+(A_i, B). \tag{6}$$

We illustrate the above transformation by the following example:

Example 9 Let $(\mathcal{A}, B, H)$ be an instance of the co-ATOM-HSUB problem with $\mathcal{A} \equiv \{A_1, A_2, A_3\} \equiv \{P(y_1, a, y_2), Q(y_1, b), P(b, y_1, y_2)\}$, $B \equiv P(x_1, x_2, x_2)$ and $H \equiv \{a, b, c\}$. Hence, $\mathcal{A} \not\leq_{ss}^H B$ holds, iff the equational problem $\mathcal{P} \equiv (\exists \vec{x}\,) \bigwedge_{i=1}^{3} (\forall \vec{y}) MGU(A_i, B)$ is satisfiable with

$$\begin{aligned} MGU(A_1, B) &\equiv \Big(y_1 \neq x_1 \vee x_2 \neq a \vee y_2 \neq a\Big) \\ MGU(A_2, B) &\equiv \top \\ MGU(A_3, B) &\equiv \Big(x_1 \neq b \vee y_1 \neq x_2 \vee y_2 \neq x_2\Big). \end{aligned}$$

By applying the U_2-rule twice to $MGU(A_1, B)$, we may delete the disjuncts $y_1 \neq x_1$ and $y_2 \neq a$ involving universally quantified variables. Likewise, $y_1 \neq x_2 \vee y_2 \neq x_2$ may be deleted in $MGU(A_3, B)$. Finally, the trivially true conjunct $MGU(A_2, B)$ may of course be ignored. Hence, by Theorem 8, $\{A_1, \ldots, A_n\} \not\leq_{ss}^H B$ holds, iff the equational problem

$$\mathcal{P}^+_{\mathcal{A},B} \equiv (\exists \vec{x}\,)\Big((x_2 \neq a) \wedge (x_1 \neq b)\Big)$$

over H is satisfiable. $\diamond$

4 H-Subsumption over an Infinite Universe

4.1 Atomic H-Subsumption over an Infinite Universe

In this section, we present a reduction of the co-ATOM-HSUB problem over an *infinite* Herbrand universe to an equational problem over a *finite* universe. Together with the transformation to be presented in Section 5, we thus get a (polynomial time) problem reduction from the co-ATOM-HSUB problem to the SAT problem.

Now let $(\mathcal{A}, B, H)$ be an instance of the co-ATOM-HSUB problem where H is infinite, $\mathcal{A}$ is given by $\{A_1, \ldots, A_n\}$ and $\vec{y} \equiv Var(\mathcal{A})$, $\vec{x} \equiv Var(B)$, and $\vec{x} \cap \vec{y} \equiv \emptyset$. Analogously to the case of a finite universe, we know that $\{A_1, \ldots, A_n\} \not\leq^H_{ss} B$ holds, iff the following equational problem over H is satisfiable:

$$\mathcal{P} \equiv (\exists \vec{x}\,)(\forall \vec{y}\,) \bigwedge_{i=1}^{n} MGU(A_i, B).$$

This form is the starting point for our problem reduction to an equivalent equational problem $\mathcal{P}'$ over a finite universe H'. The reduction consists of two principal steps:

1. First we transform the equational problem $\mathcal{P}$ over H into an equivalent problem $\mathcal{Q}$ over the same H but with a much simpler form. We perform this transformation for each $MGU(A_i, B)$ separately, and combine the results conjunctively. As a result, the original quantifier prefix $\exists\forall$ is replaced by a purely existential quantifier prefix.

2. Then we define a finite Herbrand universe H' and an equational problem $\mathcal{P}'$ over H', such that $\mathcal{Q}$ (and, equivalently $\mathcal{P}$) over H is satisfiable, iff $\mathcal{P}'$ over H' is satisfiable.

The transformation of $\mathcal{P}$ into $\mathcal{Q}$ consists of four transformation rules (T0) – (T3) which are described below:

(T0) *(Initial simplifications)*
We start off like in Section 3. Recall that the disjuncts $MGU(A_i, B)$ are either of the form $\top$ (and are thus deleted) or they are of the form $x_1 \neq s_1 \vee \ldots \vee x_k \neq s_k \vee y_1 \neq t_1 \vee \ldots \vee y_l \neq t_l$, where $\mu \equiv mgu(A_i, B)$ is of the form $\{x_1 \leftarrow s_1, \ldots, x_k \leftarrow s_k, y_1 \leftarrow t_1, \ldots, y_l \leftarrow t_l\}$. In the latter case, we delete the disequations $y_1 \neq t_1 \vee \ldots \vee y_l \neq t_l$ from $MGU(A_i, B)$ by the U_2-rule. It is not necessary to apply a substitution because, whenever there exists $y_m \leftarrow t_m$ in μ, y_m does not occur in any s_j (this is guaranteed by the unification procedure). Therefore, an application of $\{y_m \leftarrow t_m\}$ to s_j has no effect. If some term s_j in the remaining disequations is not a variable from $\vec{x}$ (i.e., s_j is either a variable from $\vec{y}$ different from $y_1, \ldots, y_l$ or a non-variable term), then we replace all further occurrences of s_j in $MGU(A_i, B)$ by x_j. Moreover, if s_j is a variable from $\vec{y}$, then we delete the disequation $x_j \neq s_j$. These simplifications correspond to applications of the R_2-rule (in reverse direction) and the U_2-rule (with $y_j \neq x_j$), respectively.
After (T0), $\mathcal{Q}'$ is of the form $(\exists \vec{x}\,)(\forall \vec{y}\,) \bigwedge_{i=1}^{n} \mathcal{Q}_i$ with

$$\mathcal{Q}_i \equiv \mathcal{E}_1 \vee \ldots \vee \mathcal{E}_N \vee [\mathcal{E}_0 \wedge (d_1 \vee \ldots \vee d_M)],$$

where all $\mathcal{E}_j$'s are initially empty. The $\mathcal{E}_j$'s are conjunctions of equations without universal variables and the d_i's are disequations. We now apply the following transformation rules (T1) – (T3) exhaustively until all disequations in all $\mathcal{Q}_i$ have the simple form $x_\alpha \neq x_\beta$ for two existentially quantified variables x_α and x_β. During the iteration, some $\mathcal{E}_j$'s become non-empty and the disjunction of disequations changes.

(T1) *(Explosion rule)*
Let $\mathcal{R} \equiv x_1 \neq s'_1 \vee \ldots \vee x_M \neq s'_M$ be a disjunction of disequations and suppose that s'_j is a term of the form $f(r_1, \ldots, r_{\alpha(f)})$, such that $f \in \Sigma$ with arity $\alpha(f) \geq 0$. Then we replace $\mathcal{R}$ by

$$[\bigvee_{g \in \Sigma} x_j = g(z_{(g,1)}, \ldots, z_{(g,\alpha(g))})] \wedge [x_1 \neq s'_1 \vee \ldots \vee x_j \neq s'_j \vee \ldots \vee x_M \neq s'_M],$$

where the z's are fresh, existentially quantified variables. So far, this is simply an application of the explosion rule E to $(\forall \vec{y}\,)\, \mathcal{R}$ with a subsequent shifting of quantifiers to maintain the prenex form. The formula $(\forall \vec{y}\,)\, \mathcal{R}$

is obtained by shifting the $\forall$-quantifiers. This is possible because $\mathcal{E}_0, \ldots, \mathcal{E}_N$ do not contain universal variables.

(T2) *(Decomposition)*

By making use of the distributivity of $\wedge$ and $\vee$ and by observing that, for all symbols $g \neq f$ in Σ, the equation $x_j = g(z_{(g,1)}, \ldots, z_{(g,\alpha(g))})$ implies the disequation $x_j \neq f(r_1, \ldots, r_{\alpha(f)})$, we get

$$[\bigvee_{g \neq f} x_j = g(z_{(g,1)}, \ldots, z_{(g,\alpha(g))})] \vee [x_j = f(z_{(f,1)}, \ldots, z_{(f,\alpha(f))}) \wedge \\ (x_1 \neq s'_1 \vee \ldots \vee x_j \neq s'_j \vee \ldots \vee x_M \neq s'_M)].$$

Note that $x_j \neq s'_j \equiv x_j \neq f(r_1, \ldots, r_{\alpha(f)})$ is replaced by the disequation $f(z_{(f,1)}, \ldots, z_{(f,\alpha(f))}) \neq f(r_1, \ldots, r_{\alpha(f)})$ via R_1. Observe that we distribute the equation to the disequation, apply R_1 and distribute the equation back in order to retain the overall structure of the formula. If $\alpha(f) \equiv 0$, then we get the trivially false disequation $f \neq f$, which may simply be deleted. Otherwise, if $\alpha(f) \geq 1$, then this disequation is decomposed via unification into $z_{(f,1)} \neq r_1 \vee \ldots \vee z_{(f,\alpha(f))} \neq r_{\alpha(f)}$. Hence, in total, we transform the disjunction $\mathcal{R}$ into

$$\begin{aligned}&[\bigvee_{g \neq f} x_j = g(z_{(g,1)}, \ldots, z_{g,\alpha(g)})] \vee \\ &[x_j = f(z_{(f,1)}, \ldots, z_{(f,\alpha(f))}) \wedge (x_1 \neq s'_1 \vee \ldots \vee x_{j-1} \neq s'_{j-1} \vee \\ &\qquad z_{(f,1)} \neq r_1 \vee \ldots \vee z_{(f,\alpha(f))} \neq r_{\alpha(f)} \vee \\ &\qquad x_{j+1} \neq s'_{j+1} \vee \ldots \vee x_M \neq s'_M)].\end{aligned}$$

(T3) *(R_2-rule and U_2-rule)*

For every term r_γ with $\gamma \in \{1, \ldots, \alpha(f)\}$, we have to check whether this term occurs as a subterm in one of the remaining terms $s'_1, \ldots, s'_{j-1}, s'_{j+1}, \ldots, s'_M$ on the right-hand side of the disequations. If this is the case, then we replace this occurrence of r_γ by the variable $z_{(f,\gamma)}$. In other words, we apply the R_2-rule in reverse order. Of course, r_γ may possibly be a universal variable from $\vec{y}$. In this case, we not only replace all other occurrences of r_γ by the existential variable $z_{(f,\gamma)}$ but we also delete the disequation $z_{(f,\gamma)} \neq r_\gamma$ by the U_2-rule. Finally, we use distributivity and associativity to retain the structure of $\mathcal{Q}_i$.

Iteration of (T1) – (T3): We apply the rules (T1) – (T3) to all non-variable terms s'_j on the right-hand side of a disequation and also recursively to all new terms r_k until all non-variable terms are eliminated from the right-hand side of the disequations.

We use the following notation for the variables introduced by the explosion rule in (T1). At the beginning, we apply the rule E to the variables x_j. But then, we have to apply E also to the new variables $z_{(f,\gamma)}$. Hence, it is convenient to denote the variables resulting from E applied to x_j as $x_j[(g,1)], \ldots, x_j[(g,\alpha(g))]$ for every symbol $g \in \Sigma$, i.e., applying the explosion rule on the "top level", yields as result the disjunction $\bigvee_{g \in \Sigma} x_j = g(x_j[(g,1)], \ldots, x_j[(g,\alpha(g))])$. Applying the rule E recursively to the new variables will then produce variables of the form $x_j[w]$, where w is a word of the form $w \equiv (g_1, \gamma_1), (g_2, \gamma_2), \ldots, (g_m, \gamma_m)$, such that each g_l is a symbol from Σ and $\gamma_l \in \{1, \ldots, \alpha(g_l)\}$. Hence, whenever we apply the explosion rule, we produce a disjunction of the form $\bigvee_{g \in \Sigma} x_j[w] = g(x_j[w,(g,1)], \ldots, x_j[w,(g,\alpha(g))])$. Moreover, we assume that $\vec{x}$ is chosen "big enough", such that it also contains all the new variables of the form $x_j[w]$.

Lemma 10 *Let $\mathcal{P} \equiv (\exists\vec{x})(\forall\vec{y}) \bigwedge_{i=1}^{n} MGU(A_i, B)$ be an equational problem resulting from a co-*ATOM-HSUB *problem and suppose that $\mathcal{P}$ is transformed into the equational problem $\mathcal{Q}$ by first applying the transformation rule (T0) and then applying exhaustively the rules (T1), (T2), and (T3) to every subproblem $MGU(A_i, B)$. Then $\mathcal{Q}$ is equivalent to $\mathcal{P}$. Moreover, $\mathcal{Q}$ is of the form $\mathcal{Q} \equiv (\exists\vec{x}) \bigwedge_{i=1}^{n} \mathcal{Q}_i$ with*

$$\mathcal{Q}_i \equiv \mathcal{E}_1 \vee \ldots \vee \mathcal{E}_N \vee [\mathcal{E}_0 \wedge (d_1 \vee \ldots \vee d_M)],$$

where the d_i's are disequations $x_j[w] \neq x_{j'}[w']$ and the $\mathcal{E}_i$'s are conjunctions of equations $x_j[w] = g(x_j[w,(g,1)], \ldots, x_j[(w,(g,\alpha(g))])$.

A proof is given in the appendix. This transformation is illustrated by

Example 11 Let $\mathcal{P} \equiv (\exists\vec{x})(\forall\vec{y})\, A^* \neq B^*$ be an equational problem over H with signature $\Sigma \equiv \{a, f, g\}$ and with

$$\begin{array}{lcl} A & \equiv & P(f(f(y_3,y_1),y_1), f(y_1,y_2)) \\ B & \equiv & P(f(x_1,g(x_2)), f(x_3,g(a))). \end{array}$$

The disequation $A^* \neq B^*$ can be simplified by unification to

$$x_1 \neq f(y_3,g(x_2)) \vee x_3 \neq g(x_2) \vee y_2 \neq g(a) \vee y_1 \neq g(x_2).$$

Transformation rule (T0). All disequations of the form $y \neq t$, where y is a universally quantified variable, are deleted by the U_2-rule. Hence, we are left with the disjunction $x_1 \neq f(y_3,g(x_2)) \vee x_3 \neq g(x_2)$. Finally, we replace $g(x_2)$ in $f(y_3,g(x_2))$ by x_3 because $x_3 \neq g(x_2)$ is present. Therefore, we get

$$(\exists\vec{x})(\forall\vec{y})\, [x_1 \neq f(y_3,x_3) \vee x_3 \neq g(x_2)].$$

Transformation rule (T1). By applying the explosion rule E for the variable x_1, we get

$$\begin{array}{l} (\exists\vec{x})(\forall\vec{y})\; [x_1 = a \vee x_1 = g(x_1[(g,1)]) \vee x_1 = f(x_1[(f,1)], x_1[(f,2)])] \wedge \\ \qquad\qquad [x_1 \neq f(y_3,x_3) \vee x_3 \neq g(x_2)]. \end{array}$$

Recall that $x_1[(g,1)]$ denotes the existential variable that is used as argument of $g(x_1[(g,1)])$ when applying the explosion rule E for x_1. Likewise, the variables $x_1[(f,1)]$ and $x_1[(f,2)]$ are introduced by this application of E. Actually, in this easy example, it is convenient to replace these variable names by x_4, x_5, and x_6, respectively. We get

$$\begin{array}{l} (\exists\vec{x})(\forall\vec{y})\; [x_1 = a \vee x_1 = g(x_4) \vee x_1 = f(x_5,x_6)] \wedge \\ \qquad\qquad [x_1 \neq f(y_3,x_3) \vee x_3 \neq g(x_2)]. \end{array}$$

Transformation rule (T2). Each of the equations $x_1 = a$ and $x_1 = g(x_4)$ implies the disequation $x_1 \neq f(y_3,x_3)$ and hence the whole disjunction. We thus further transform the above problem into

$$\begin{array}{l} (\exists\vec{x})(\forall\vec{y})\Big(x_1 = a \vee x_1 = g(x_4) \vee \\ \qquad\qquad [x_1 = f(x_5,x_6) \wedge (x_1 \neq f(y_3,x_3) \vee x_3 \neq g(x_2))]\Big). \end{array}$$

With the conjunction $x_1 = f(x_5,x_6)$, we transform $x_1 \neq f(y_3,x_3)$ into $f(x_5,x_6) \neq f(y_3,x_3)$, which is equivalent to $x_5 \neq y_3 \vee x_6 \neq x_3$.

Transformation rule (T3). The disequation $x_5 \neq y_3$ has an isolated universal variable and may therefore be deleted by the U_2-rule. The equational problem after the first round is

$$\begin{array}{l} (\exists\vec{x})(\forall\vec{y})\Big(x_1 = a \vee x_1 = g(x_4) \vee \\ \qquad\qquad [x_1 = f(x_5,x_6) \wedge (x_6 \neq x_3 \vee x_3 \neq g(x_2))]\Big). \end{array}$$

Iteration of (T1)–(T3). By repeating the above steps once again, we can also simplify the disequation $x_3 \neq g(x_2)$. We apply (T1) to the disjunction $\mathcal{R} \equiv x_6 \neq x_3 \vee x_3 \neq g(x_2)$ and replace $\mathcal{R}$ by

$$\Big(x_3 = a \vee x_3 = f(x_8,x_9) \vee x_3 = g(x_7)\Big) \wedge \Big(x_6 \neq x_3 \vee x_3 \neq g(x_2)\Big). \tag{7}$$

Note that in the above formula, we are again using short variable names of the form x_7, x_8, and x_9 rather than the long form $x_3[(g,1)]$, $x_3[(f,1)]$, and $x_3[(f,2)]$, respectively.

Then (T2) is applied first resulting in a replacement of (7) by

$$\Big(x_3 = a \vee x_3 = f(x_8,x_9)\Big) \vee \Big(x_3 = g(x_7) \wedge (x_6 \neq x_3 \vee x_3 \neq g(x_2))\Big)$$

which is further changed to

$$\Big(x_3 = a \vee x_3 = f(x_8,x_9)\Big) \vee \Big(x_3 = g(x_7) \wedge (x_6 \neq x_3 \vee x_7 \neq x_2)\Big).$$

After applying distributivity and associativity laws, we obtain

$$\begin{aligned} \mathcal{Q} \equiv (\exists \vec{x}\,)\Big(& x_1 = a \vee x_1 = g(x_4) \vee (x_1 = f(x_5,x_6) \wedge x_3 = a) \vee \\ & (x_1 = f(x_5,x_6) \wedge x_3 = f(x_8,x_9)) \vee \\ & [(x_1 = f(x_5,x_6) \wedge x_3 = g(x_7)) \wedge (x_6 \neq x_3 \vee x_7 \neq x_2)]\Big). \end{aligned} \qquad \diamond$$

We now proceed to the definition of the equational problem $\mathcal{P}'$ over the *finite* Herbrand universe H'.

Definition 12 *Let* $\mathcal{Q} \equiv (\exists \vec{x}\,) \bigwedge_{i=1}^{n} \mathcal{Q}_i$ *with*

$$\mathcal{Q}_i \equiv \mathcal{E}_1 \vee \ldots \vee \mathcal{E}_N \vee [\mathcal{E}_0 \wedge (d_1 \vee \ldots \vee d_M)]$$

be an equational problem over the infinite *Herbrand universe* H *with signature* Σ*, such that* $\mathcal{Q}$ *results from the above transformation. Then we define* H' *and* $\mathcal{P}'$ *as follows.*

We set $H' \equiv \{a_\gamma \mid g_\gamma \in \Sigma\} \cup \{c_j[w] \mid x_j[w]$ *is a variable in* $\mathcal{Q}\}$*, i.e.,* H' *contains a constant symbol* a_γ *for every* $g_\gamma \in \Sigma$ *and a new constant symbol* $c_j[w]$ *for every variable* $x_j[w]$ *occurring in* $\mathcal{Q}$*.*

We set $\mathcal{P}' \equiv (\exists \vec{x}\,) \bigwedge_{l=1}^{n} \mathcal{P}'_l$ *where every* $\mathcal{P}'_l$ *is of the form*

$$\mathcal{P}'_l \equiv \mathcal{E}'_1 \vee \ldots \vee \mathcal{E}'_N \vee [\mathcal{E}'_0 \wedge (\mathcal{D}_1 \vee \ldots \vee \mathcal{D}_M)]$$

and the subproblems $\mathcal{E}'_j$ *and* $\mathcal{D}_j$ *are defined as follows.*

Each equation $x_j[w] = g_\gamma(x_j[w,(g_\gamma,1)],\ldots,x_j[w,(g_\gamma,\alpha(g_\gamma))])$ *in* $\mathcal{E}_i$ *(with* $g_\gamma \in \Sigma$*) is encoded by the equation* $x_j[w] = a_\gamma$ *in the corresponding conjunction* $\mathcal{E}'_i$ *of equations.*

For each disequation $d_i \equiv x_j[w] \neq x_{j'}[w']$*, we proceed as follows. Let* V *denote the set of all words* v*, such that there exists both a variable* $x_j[w,v]$ *and a variable* $x_{j'}[w',v]$ *in* $\mathcal{Q}$*. Of course,* V *always contains the empty word* ϵ*. Then* d_i *is encoded by the disjunction* $\bigvee_{v \in V} \mathcal{D}(v)$*, where* $\mathcal{D}(v)$ *is defined as follows. If* $v \equiv \epsilon$*, then* $\mathcal{D}(\epsilon) \equiv x_j[w] \neq x_{j'}[w']$*. If* v *is of the form* $(g_1,\gamma_1),\ldots,(g_k,\gamma_k)$*, then*

$$\begin{aligned} \mathcal{D}(v) \equiv\ & x_j[w] = x_{j'}[w'] = a_1 \wedge \\ & x_j[w,(g_1,\gamma_1)] = x_{j'}[w',(g_1,\gamma_1)] = a_2 \wedge \ldots \wedge \\ & x_j[w,(g_1,\gamma_1),\ldots,(g_{k-1},\gamma_{k-1})] = \\ & \quad x_{j'}[w',(g_1,\gamma_1),\ldots,(g_{k-1},\gamma_{k-1})] = a_k \wedge \\ & x_j[w,(g_1,\gamma_1),\ldots,(g_k,\gamma_k)] \neq x_{j'}[w',(g_1,\gamma_1),\ldots,(g_k,\gamma_k)]. \end{aligned}$$

We explain the basic idea of the elements in the finite universe H' in the following. The constants a_i are used to encode the equations $x_j[w] = g_i(x_j[w,(g_i,1)],\dots,x_j[(w,(g_i,\alpha(g_i))])$. The purpose of the constants $c_j[w]$ (which do *not* occur in the translated formula) is to make sure that every conjunction of non-trivial disequations over H with variables in $\vec{x}$ is indeed satisfiable. Note that (in contrast to an infinite Herbrand universe) this condition is in general not true for a finite universe. The definition of the conjunctions $\mathcal{E}_i'$ of equations is obvious. The idea of the encoding of the disequations d_i is as follows. Suppose that we compute the mgu μ of any conjunctions of equations occurring in $\mathcal{Q}$. Then the words v for which the variable $x_j[w,v]$ indeed occurs in $\mathcal{Q}$ correspond to positions in the terms that may possibly be substituted for the variable $x_j[w]$ by μ. Likewise, the words v, such that $x_{j'}[w',v]$ occurs in $\mathcal{Q}$, correspond to possible positions in $x_{j'}[w']\mu$. Then the disjunction $\bigvee_{v\in V}\mathcal{D}(v)$ encodes all possibilities to validate the disequation $x_j[w]\mu \neq x_{j'}[w']\mu$, i.e., there exists a common position p (corresponding to such a word v) in $x_j[w]\mu$ and $x_{j'}[w']\mu$, such that these two terms have distinct leading symbols, whereas at all positions above p, these two terms have the same leading symbol from Σ.

Lemma 13 *Let $\mathcal{Q}$ over H be an equational problem that results from the transformation based on the rules (T0) – (T3) and let $\mathcal{P}'$ over H' be the corresponding equational problem according to Definition 12. Then the following equivalence holds:*

$\mathcal{Q}$ is satisfiable (over H), iff $\mathcal{P}'$ is satisfiable (over H').

A proof is given in the appendix. Below we continue the Example 11.

Example 14 Let $\mathcal{Q}$ be the formula from Example 11 with

$$\begin{aligned}\mathcal{Q} \equiv (\exists\vec{x})\Big(&x_1 = a \vee x_1 = g(x_4) \vee (x_1 = f(x_5,x_6) \wedge x_3 = a) \vee\\ &(x_1 = f(x_5,x_6) \wedge x_3 = f(x_8,x_9)) \vee\\ &[(x_1 = f(x_5,x_6) \wedge x_3 = g(x_7)) \wedge (x_6 \neq x_3 \vee x_7 \neq x_2)]\Big).\end{aligned}$$

From this we define the desired equational problem $\mathcal{P}'$ over the finite universe H' as follows. Recall that H' contains one constant a_i for each symbol in $\Sigma \equiv \{a,f,g\}$ and one constant c_l for each variable in $\mathcal{Q}$. Hence, $H' \equiv \{a_1,a_2,a_3\} \cup \{c_1,\dots,c_9\}$. Moreover, $\mathcal{P}'$ is defined as

$$\begin{aligned}(\exists\vec{x})\ \Big(&x_1 = a_1 \vee x_1 = a_3 \vee (x_1 = a_2 \wedge x_3 = a_1) \vee (x_1 = a_2 \wedge x_3 = a_2) \vee\\ &[(x_1 = a_2 \wedge x_3 = a_3) \wedge (x_6 \neq x_3 \vee x_7 \neq x_2)]\Big).\end{aligned}$$

Recall that the constants a_1, a_2, and a_3 are used to encode terms of the form a, $f(x_i,x_j)$ and $g(x_k)$, which are introduced by the explosion rule. The idea of the constants c_l is to guarantee that any conjunction of disequations over H' is satisfiable, provided that none of the disequations is of the trivially false form $t \neq t$. Of course, for this simple example, we could have chosen a much smaller universe H'. $\diamond$

By combining the two Lemmas 10 and 13, we immediately get

Theorem 15 *Let an instance of the co-*ATOM-HSUB *problem over the* infinite *Herbrand universe H be given through the atoms $A_1,\dots,A_n$, B. Moreover, let $\mathcal{P}'$ be the equational problem over H' according to Definition 12. Then $\{A_1,\dots,A_n\} \not\leq_{ss}^{H} B$ holds, iff $\mathcal{P}'$ over the* finite *Herbrand universe H' is satisfiable.*

In order to give upper bounds on the complexity of this transformation, it is convenient to introduce the following notation: We write $|\cdot|$ to denote the number of elements in a set and $||\cdot||$ to denote the size of a dag representation, e.g., $|\mathcal{A}|$ denotes the number of atoms in $\mathcal{A}$ and $||\mathcal{A}||$ denotes the total size of the representation of $\mathcal{A}$. Let Σ be the signature of the input Herbrand universe H. We write $arity(\Sigma)$ for the sum of the arities of the symbols in Σ, i.e., $arity(\Sigma) \equiv \sum_{f \in \Sigma} \alpha(f)$, where $\alpha(f)$ denotes the arity of f. Finally, we define $size(\Sigma) \equiv arity(\Sigma) + |\Sigma|$.

Theorem 16 *Let $(\mathcal{A}, B, H)$ be an instance of the co-*ATOM-HSUB *problem with infinite H over some signature Σ and let $\mathcal{P}'$ and H' be as in Definition 12. Then the following upper bounds hold:*

$$\begin{aligned} |Var(\mathcal{P}')| &\leq |\mathcal{A}| * (||\mathcal{A}|| + ||B||) * size(\Sigma) \\ ||\mathcal{P}'|| &\equiv O\Big(|\mathcal{A}|^3 * (||\mathcal{A}|| + ||B||)^3 * size(\Sigma)^3\Big) \\ |H'| &\equiv O\Big(|\mathcal{A}| * (||\mathcal{A}|| + ||B||) * size(\Sigma)\Big) \end{aligned}$$

The time required to compute $(\mathcal{P}', H')$ is also bounded by

$$O\Big(|\mathcal{A}|^3 * (||\mathcal{A}|| + ||B||)^3 * size(\Sigma)^3\Big).$$

A proof sketch of these upper bounds is given in the appendix.

4.2 Clausal H-Subsumption over an Infinite Universe

We shall now reduce the co-CLAUSE-HSUB problem to an equational problem with two sorts of variables, such that each sort is interpreted over a finite Herbrand universe. Together with the transformation to be presented in Section 5, we thus get a (polynomial time) problem reduction from the co-CLAUSE-HSUB problem to the QSAT_2 problem. This reduction consists of several steps, which will be worked out in detail for the case of H-subsumption by a single clause, i.e., we have to test $C \not\leq^H_{ss} D$ for a single clause C. The extension to testing $\{C_1, \dots, C_n\} \not\leq^H_{ss} D$ will then be straightforward.

1. We shall start with a new criterion for testing $C \not\leq^H_{ss} D$, which leads to an equational problem with variables of two sorts. One sort is already interpreted over a finite universe, whereas the other one is considered over an infinite universe.
2. In order to get rid of the infinite universe, we apply the problem reduction from Section 4.1.
3. Finally, some minor transformations of the equational problem are required, in order to produce the desired quantifier prefix $\exists\forall$.

Let us first introduce some useful notation. It is convenient to consider clauses $C \equiv L_1 \vee \ldots \vee L_l$ and $D \equiv M_1 \vee \ldots \vee M_m$ as sets of literals. In particular, the order of literals and multiple occurrences of a literal are irrelevant. On the other hand, if the order of literals and multiple occurrences of literals do matter, then we have to switch to tuples of literals, which are denoted as $\langle L_{i_1}, \dots, L_{i_k}\rangle$ and $\langle M_{j_1}, \dots, M_{j_k}\rangle$, respectively. Moreover, we write $\langle L_{i_1}, \dots, L_{i_k}\rangle \leq_s \langle M_{j_1}, \dots, M_{j_k}\rangle$ to denote that $\langle M_{j_1}, \dots, M_{j_k}\rangle$ is an instance of $\langle L_{i_1}, \dots, L_{i_k}\rangle$.

Step 1 *(Definition of a new H-subsumption criterion)*
(Clausal) H-subsumption can be tested via the following criterion. A proof will be given in the appendix.

Lemma 17 *Let $C \equiv L_1 \vee \ldots \vee L_l$ and $D \equiv M_1 \vee \ldots \vee M_m$ be clauses over an arbitrary Herbrand universe H. Then $C \not\leq^H_{ss} D$ holds, iff there exists a substitution ϑ on $Var(D)$, such that, for every tuple $\langle j_1, \dots, j_l\rangle \in \{1, \dots, m\}^l$, there exist two indices α, β satisfying $1 \leq \alpha < \beta \leq l$ and $\langle L_\alpha, L_\beta\rangle \not\leq_s \langle M_{j_\alpha}\vartheta, M_{j_\beta}\vartheta\rangle$.*

Step 2 *(Getting rid of the infinite Herbrand universe)*
Let $\langle \alpha, \beta, \gamma, \delta \rangle$ be an arbitrary quadruple of indices for which the relation $\langle L_\alpha, L_\beta \rangle \not\leq_s \langle M_\gamma \vartheta, M_\delta \vartheta \rangle$ holds. Then the following equational problem over the infinite Herbrand universe H is satisfiable:

$$(\exists \vec{x})(\forall \vec{y}) \langle L^*_\alpha, L^*_\beta \rangle \neq \langle M^*_\gamma, M^*_\delta \rangle$$

where $\vec{x} \equiv Var(D)$ and $\vec{y} \equiv Var(C)$. By Lemma 10, we know how to transform such an equational problem $\mathcal{P}$ into a purely existentially quantified problem $\mathcal{P}'$ over the *finite* Herbrand universe H'. For a given quadruple $\langle \alpha, \beta, \gamma, \delta \rangle$, we shall write

$$(\exists \vec{x}) P(\alpha, \beta, \gamma, \delta)$$

to denote this equational problem $\mathcal{P}'$. Together with the H-subsumption criterion in Lemma 17, we claim that $C \not\leq^H_{ss} D$ holds, iff the following "equational problem" is satisfiable (The correctness of this step is by no means trivial and will be formally proven in the appendix):

$$(\exists \vec{x})(\forall j_1, \ldots, j_l)(\exists \alpha, \beta)\Big((\alpha < \beta) \wedge P(\alpha, \beta, j_\alpha, j_\beta)\Big). \tag{8}$$

Step 3 *(Producing the desired quantifier prefix)*
Actually, we have slightly simplified the formulation of the equational problem (8) since the equational problems $P(\alpha, \beta, j_\alpha, j_\beta)$ are somehow parameterized by the variables α, β, j_α, and j_β. Moreover, we would like to get rid of the quantification $(\exists \alpha, \beta)$ (and also the inequality $\alpha < \beta$), in order to arrive at the desired quantifier prefix $\exists\forall$. The latter problem can be solved by replacing $(\exists \alpha, \beta)\Big[(\alpha < \beta) \wedge P(\alpha, \beta, j_\alpha, j_\beta)\Big]$ by the following disjunction of $\frac{l*(l-1)}{2}$ disjuncts:

$P(1, 2, j_1, j_2) \vee \ldots \vee P(1, l, j_1, j_l) \vee P(2, 3, j_2, j_3) \vee \ldots \vee$
$\quad P(2, l, j_2, j_l) \vee \ldots \vee P(l-1, l, j_{l-1}, j_l)$

The former defect can be fixed by explicitly listing all m^2 possible values for j_α and j_β in all of the disjuncts above, i.e., each of the problems $P(1, 2, j_1, j_2), P(1, 3, j_1, j_3), \ldots, P(l-1, l, j_{l-1}, j_l)$ is replaced by a conjunction consisting of m^2 conjuncts in the following way:

$$\begin{array}{lll}
P(1, 2, j_1, j_2) & \Rightarrow & [(j_1 = 1 \wedge j_2 = 1) \rightarrow P(1, 2, 1, 1)] \wedge \ldots \wedge \\
 & & [(j_1 = m \wedge j_2 = m) \rightarrow P(1, 2, m, m)] \\
P(1, 3, j_1, j_3) & \Rightarrow & [(j_1 = 1 \wedge j_3 = 1) \rightarrow P(1, 3, 1, 1)] \wedge \ldots \wedge \\
 & & [(j_1 = m \wedge j_3 = m) \rightarrow P(1, 3, m, m)] \\
 & \vdots & \\
P(l-1, l, j_{l-1}, j_l) & \Rightarrow & [(j_{l-1} = 1 \wedge j_l = 1) \rightarrow P(l-1, l, 1, 1)] \wedge \ldots \wedge \\
 & & [(j_{l-1} = m \wedge j_l = m) \rightarrow P(l-1, l, m, m)]
\end{array}$$

In total, we get an equational problem $(\exists \vec{x})(\forall j_1, \ldots, j_l)\mathcal{Q}$ with variables $\vec{x}$ over the finite Herbrand universe H_1 and variables $j_1, \ldots, j_l$ over the finite universe $H_2 \equiv \{1, \ldots, m\}$, where H_1 is obtained as the union of all universes H' that we get by reducing $(\exists \vec{x})(\forall \vec{y}) \langle L^*_\alpha, L^*_\beta \rangle \neq \langle M^*_\gamma, M^*_\delta \rangle$ over H to the equivalent problem $(\exists \vec{x}) P(\alpha, \beta, \gamma, \delta)$ over H' via Lemma 10 for any quadruple $\langle \alpha, \beta, \gamma, \delta \rangle$ of indices. Moreover, $\mathcal{Q}$ is a disjunction consisting of $\frac{l*(l-1)}{2}$ disjuncts, each of which is a conjunction consisting of m^2 conjuncts. In the theorem below, we claim that this problem reduction is correct. For a proof, see the appendix.

Theorem 18 *Let $(\{C\}, D, H)$ be an instance of the co-*CLAUSE-HSUB *problem where H is infinite and C and D are clauses. Moreover, let the equational problem $(\exists\vec{x})(\forall j_1, \ldots, j_l)\mathcal{Q}$ over the* finite *Herbrand universes H_1 and H_2 be obtained by the above problem reduction. Then $(\exists\vec{x})(\forall j_1, \ldots, j_l)\mathcal{Q}$ is satisfiable, iff $C \not\leq_{ss}^{H} D$ holds.*

This transformation is illustrated in the following example.

Example 19 Let $(\{C\}, D, H)$ be an instance of the co-CLAUSE-HSUB problem where the clauses $C \equiv P(y_1, y_2) \vee Q(y_1) \vee Q(y_2)$ and $D \equiv P(x_1, x_2) \vee Q(a) \vee Q(b)$ are from Example 1 (up to some variable renamings). Moreover, let H be the infinite Herbrand universe with signature $\Sigma \equiv \{a, b, g\}$, such that a and b are constants and g is a function symbol of arity 1. Below, we shall prove that $C \not\leq_{ss}^{H} D$ holds via the above reduction to an equational problem over two finite universes.

Recall from (8) that $C \not\leq_{ss}^{H} D$ holds, iff the formula

$$(\exists\vec{x})(\forall j_1, \ldots, j_l)(\exists\alpha, \beta)\Big((\alpha < \beta) \wedge P(\alpha, \beta, j_\alpha, j_\beta)\Big)$$

is satisfiable. Here, we have $l \equiv 3$. Hence, (8) can be transformed into

$$(\exists\vec{x})(\forall j_1, j_2, j_3)\Big(P(1, 2, j_1, j_2) \vee P(1, 3, j_1, j_3) \vee P(2, 3, j_2, j_3)\Big) \tag{9}$$

We now have to compute $P(\alpha, \beta, \gamma, \delta)$ for all possible values of $\alpha, \beta, \gamma, \delta \in \{1, 2, 3\}$ with $\alpha < \beta$. We first summarize the results and explain the computation below.

$$P(1,2,\gamma,\delta) \equiv \begin{cases} x_1 \neq a_1 & \text{if } (\gamma,\delta) \equiv (1,2) \\ x_1 \neq a_2 & \text{if } (\gamma,\delta) \equiv (1,3) \\ \top & \text{otherwise} \end{cases}$$

$$P(1,3,\gamma,\delta) \equiv \begin{cases} x_2 \neq a_1 & \text{if } (\gamma,\delta) \equiv (1,2) \\ x_2 \neq a_2 & \text{if } (\gamma,\delta) \equiv (1,3) \\ \top & \text{otherwise} \end{cases}$$

$$P(2,3,\gamma,\delta) \equiv \begin{cases} \bot & \text{if } (\gamma,\delta) \in \{(2,2),(2,3),(3,2),(3,3)\} \\ \top & \text{otherwise} \end{cases}$$

In all of the above cases, x_1 and x_2 are interpreted over the universe $H_1 \equiv \{a_1, a_2, a_3, c_1, c_2\}$ where a_1 encodes $a \in \Sigma$, a_2 encodes $b \in \Sigma$, and a_3 encodes $g \in \Sigma$. Moreover, c_1, c_2 are needed since we are dealing with two existential variables x_1 and x_2. Of course, the j_i's are interpreted over $H_2 \equiv \{1, 2, 3\}$.

We shall now briefly sketch how the above properties are derived. To this end, we first consider the case $(\alpha, \beta) \equiv (1, 2)$:

For $(\gamma, \delta) \equiv (1, 2)$, we transform $(\exists\vec{x})(\forall\vec{y})\langle L_1^*, L_2^*\rangle \neq \langle M_1^*, M_2^*\rangle$ into

$$(\exists\vec{x})(\forall\vec{y})\langle f_P^+(y_1, y_2), f_Q^+(y_1)\rangle \neq \langle f_P^+(x_1, x_2), f_Q^+(a)\rangle$$

which is equivalent to $(\exists\vec{x})(\forall\vec{y})(y_1 \neq x_1 \vee y_2 \neq x_2 \vee x_1 \neq a)$. By two applications of the U_2-rule, we derive $(\exists\vec{x})x_1 \neq a$. Hence, for the finite universe $H_1 \equiv \{a_1, a_2, a_3, c_1, c_2\}$, we indeed get $P(1, 2, 1, 2) \equiv x_1 \neq a_1$.

Analogously, for $(\gamma, \delta) \equiv (1, 3)$, we get

$$(\exists\vec{x})(\forall\vec{y})\langle f_P^+(y_1, y_2), f_Q^+(y_1)\rangle \neq \langle f_P^+(x_1, x_2), f_Q^+(b)\rangle$$

which can be simplified to $(\exists\vec{x})x_1 \neq b$. Hence, we indeed end up with $P(1,2,1,3) \equiv x_1 \neq a_2$.

It remains to show that for all other values of (γ,δ), the equational problem $(\exists\vec{x})(\forall\vec{y})\langle L^*_\alpha, L^*_\beta\rangle \neq \langle M^*_\gamma, M^*_\delta\rangle$ is equivalent to $\top$. It suffices to show that in all these cases, the left-hand side $\langle L^*_\alpha, L^*_\beta\rangle$ of this disequation is *not* unifiable with the right-hand side $\langle M^*_\gamma, M^*_\delta\rangle$. But this can be easily verified. For instance, for $(\gamma,\delta) \equiv (2,1)$, we have $(\exists\vec{x})(\forall\vec{y})\langle f^+_P(y_1,y_2), f^+_Q(y_1)\rangle \neq \langle f^+_Q(a), f^+_P(x_1,x_2)\rangle$ which is $\top$.

It is now also obvious how $P(\alpha,\beta,\gamma,\delta)$ is computed for $(\alpha,\beta) \equiv (1,3)$. We thus immediately proceed to the case $(\alpha,\beta) \equiv (2,3)$.

Let $(\gamma,\delta) \equiv (2,2)$. Then we have that the equational problem $(\exists\vec{x})(\forall\vec{y})\langle L^*_2, L^*_3\rangle \neq \langle M^*_2, M^*_2\rangle$ is of the form $(\exists\vec{x})(\forall\vec{y})\langle f^+_Q(y_1), f^+_Q(y_2)\rangle \neq \langle f^+_Q(a), f^+_Q(a)\rangle$ which is equivalent to $(\exists\vec{x})(\forall\vec{y})(y_1 \neq a \vee y_2 \neq a)$. By applying the U_2-rule twice, the latter problem can be simplified to the empty disjunction which is equivalent to $\perp$. In fact, if $\gamma \not\equiv 1$ and $\delta \not\equiv 1$ holds, then we have $P(2,3,\gamma,\delta)$ is equivalent to $\perp$. On the other hand, if $\gamma \equiv 1$ or $\delta \equiv 1$ holds, then $P(2,3,\gamma,\delta)$ is equivalent to $\top$ follows immediately from the non-unifiability of $f^+_P(x_1,x_2)$ with both $f^+_Q(y_1)$ and $f^+_Q(y_2)$.

In summary, the formula (9) can be transformed into

$$
\begin{aligned}
&\mathcal{Q} \equiv (\exists\vec{x})(\forall j_1,j_2,j_3)\Big(\\
&\quad [(j_1 = 1 \wedge j_2 = 2) \rightarrow x_1 \neq a_1) \wedge (j_1 = 1 \wedge j_2 = 3) \rightarrow x_1 \neq a_2)] \vee\\
&\quad [(j_1 = 1 \wedge j_3 = 2) \rightarrow x_2 \neq a_1) \wedge (j_1 = 1 \wedge j_3 = 3) \rightarrow x_2 \neq a_2)] \vee\\
&\quad [(j_2 = 2 \wedge j_3 = 2) \rightarrow \perp) \wedge \ldots \wedge (j_2 = 3 \wedge j_3 = 3) \rightarrow \perp)]\Big).
\end{aligned}
$$

In Section 5, we shall show how the satisfiability of such a formula can be tested via a transformation to a QBF. However, due to the simple form of the above formula $\mathcal{Q}$, our satisfiability test here can be easily done in an ad hoc manner. Note that $\mathcal{Q}$ is satisfiable, if there exists a ground substitution ϑ on $\vec{x}$, such that, for any value of j_1, j_2, j_3, the above disjunction is trivially true. In fact, $\vartheta \equiv \{x_1 \leftarrow a_3, x_2 \leftarrow a_3\}$ is such a substitution since, in this case, the first disjunct $P(1,2,j_1,j_2)\vartheta$ becomes $[(j_1 = 1 \wedge j_2 = 2) \rightarrow a_3 \neq a_1) \wedge (j_1 = 1 \wedge j_2 = 3) \rightarrow a_3 \neq a_2)]$, which is equivalent to $\top$ for any choice of $\langle j_1, j_2, j_3\rangle$. $\diamond$

The extension of this construction to the general co-CLAUSE-HSUB problem is straightforward. Let an instance of the co-CLAUSE-HSUB problem over some infinite Herbrand universe H be given through the clause set $\mathcal{C} \equiv \{C_1,\ldots,C_n\}$ with $C_i \equiv L_{i1} \vee \ldots \vee L_{il_i}$ and the clause $D \equiv M_1 \vee \ldots \vee M_m$. Moreover, let $\vec{y} \equiv Var(\mathcal{C})$ and $\vec{x} \equiv Var(D)$. Analogously to the considerations on H-subsumption criteria above, $\mathcal{C} \not\leq^H_{ss} D$ holds, iff there exists a substitution ϑ on $\vec{x}$, such that, for every $i \in \{1,\ldots,n\}$ and for every tuple $\langle j_1,\ldots,j_{l_i}\rangle \in \{1,\ldots,m\}^{l_i}$, there exist two indices α, β with $1 \leq \alpha < \beta \leq l$ for which the condition $\langle L_{i\alpha}, L_{i\beta}\rangle \not\leq_s \langle M_{j_\alpha}\vartheta, M_{j_\beta}\vartheta\rangle$ holds.

Let $i \in \{1,\ldots,n\}$ and let $\langle\alpha,\beta,\gamma,\delta\rangle$ be a quadruple of indices for which $\langle L_{i\alpha}, L_{i\beta}\rangle \not\leq_s \langle M_\gamma, M_\delta\vartheta\rangle$ holds. Then the equational problem

$$(\exists\vec{x})(\forall\vec{y})\langle L^*_{i\alpha}, L^*_{i\beta}\rangle \neq \langle M^*_\gamma, M^*_\delta\rangle$$

over the infinite Herbrand universe H is satisfiable. Each such equational problem over H can be transformed into an equivalent equational problem $\mathcal{P}'$ over the finite universe H'. For any index $i \in \{1,\ldots,n\}$ and any quadruple $\langle\alpha,\beta,\gamma,\delta\rangle$, we shall write $(\exists\vec{x})P(i,\alpha,\beta,\gamma,\delta)$ to denote this equational problem $\mathcal{P}'$. Analogously to the case of $i \equiv 1$, we thus get the following criterion for $\mathcal{C} \not\leq^H_{ss} D$:

$$(\exists\vec{x})(\forall j_1,\ldots,j_l)(\forall i)(\exists\alpha_i,\beta_i)(\alpha_i < \beta_i) \wedge P(i,\alpha_i,\beta_i,j_{\alpha_i},j_{\beta_i}).$$

We get rid of the quantification $(\forall i)$ by replacing this formula by

$$(\exists \vec{x}\,)(\forall j_1, \ldots, j_l)\,(\exists \alpha_1, \beta_1)(\alpha_1 < \beta_1) \wedge P(1, \alpha_1, \beta_1, j_{\alpha_1}, j_{\beta_1}) \wedge \ldots \wedge$$
$$(\exists \alpha_n, \beta_n)(\alpha_n < \beta_n) \wedge P(n, \alpha_n, \beta_n, j_{\alpha_n}, j_{\beta_n})$$

Getting rid of the quantifications $(\exists \alpha_i, \beta_i)$ and the conjuncts $(\alpha_i < \beta_i)$ as well as replacing the "parameters" $\alpha_i, \beta_i, j_{\alpha_i}, j_{\beta_i}$ of the subformulas $P(i, \alpha_i, \beta_i, j_{\alpha_i}, j_{\beta_i})$ by concrete numbers can then be done in exactly the same way as for the special case $i \equiv 1$. Hence, we end up with an equational problem

$$(\exists \vec{x}\,)(\forall j_1, \ldots, j_l)\mathcal{Q}_1 \wedge \ldots \wedge \mathcal{Q}_n$$

with variables $\vec{x}$ over the finite Herbrand universe H_1 and variables $j_1, \ldots, j_l$ over the finite universe $H_2 \equiv \{1, \ldots, m\}$, where H_1 is again the union of all possible universes H' that we get by applying the problem transformation from Lemma 10. Moreover, for every $i \in \{1, \ldots, n\}$, $\mathcal{Q}_i$ is a disjunction consisting of $\frac{l_i*(l_i-1)}{2}$ disjuncts, each of which is a conjunction consisting of m^2 conjuncts. In the theorem below, we claim that this problem reduction is correct. The proof is omitted since it goes analogously to the proof of Theorem 18 in the appendix.

Theorem 20 *Let an instance of the co-*CLAUSE-HSUB *problem over the infinite Herbrand universe H be given through the clause set $\mathcal{C} \equiv \{C_1, \ldots, C_n\}$ and the clause D. Moreover, let the equational problem $\mathcal{Q} \equiv (\exists \vec{x}\,)(\forall j_1, \ldots, j_l)\mathcal{Q}_1 \wedge \ldots \wedge \mathcal{Q}_n$ over the* finite *Herbrand universes H_1 and H_2 be obtained from $\mathcal{C}$ and D by the above problem reduction. Then $\mathcal{C} \not\leq^H_{ss} D$ holds, iff $\mathcal{Q}$ is satisfiable.*

In order to formulate complexity upper bounds, we again use the notation $|\cdot|$, $||\cdot||$, $arity(\Sigma)$, and $size(\Sigma)$ as in Section 4.1. Moreover, we write l to denote the maximum number of literals of any clause $C_i \in \mathcal{C}$ and we write m for the number of literals in D. Finally, let $n \equiv |\mathcal{C}|$ and $||C_{\max}|| \equiv \max\Big(\{||C_i|| : C_i \in \mathcal{C}\}\Big)$. Then we get

Theorem 21 *Let $(\mathcal{C}, D, H)$ be an instance of the co-*CLAUSE-HSUB *problem with infinite H over some signature Σ and let $\mathcal{Q}$ and the universes H_1 and H_2 be defined as described above. Then the following upper bounds hold:*

$$\begin{aligned}
|\,Var(\mathcal{Q})| &\equiv O\Big(n * l^2 * m^2 * (||C_{\max}|| + ||D||) * size(\Sigma)\Big) \\
||\mathcal{Q}|| &\equiv O\Big(n^3 * l^6 * m^6 * (||C_{\max}|| + ||D||)^3 * size(\Sigma)^3\Big) \\
|H_1| &\equiv O\Big(n * l^2 * m^2 * (||C_{\max}|| + ||D||) * size(\Sigma)\Big) \\
|H_2| &\equiv m
\end{aligned}$$

The time required to compute $(\mathcal{Q}, H_1, H_2)$ is also bounded by

$$O\Big(n^3 * l^6 * m^6 * (||C_{\max}|| + ||D||)^3 * size(\Sigma)^3\Big).$$

A proof sketch of these upper bounds is given in the appendix.

5 Transformation into QBFs

This section outlines how equational problems over several finite Herbrand universes can be expressed in terms of (quantified) propositional logic. As already argued, this class of problems contains all forms of equational problems, which have been derived in the previous sections to encode the considered H-subsumption problems. In order to obtain a *polynomial* mapping from equational problems into a propositional language, we have to use a well-known generalization of propositional logic, namely quantified Boolean formulas. However, for equational problems representing an *atomic* H-subsumption problem, it turns out that the resulting QBFs are in fact propositional formulas.

5.1 Main Construction

Definition 22 *Let $\mathcal{P}$ be an equational problem over $H_1, \ldots, H_r$. We define an underlying alphabet of propositional atoms $At_{\mathcal{P}}$, containing for each first-order variable x which is associated to H_i in $\mathcal{P}$ a vector of mutually disjoint propositional atoms $\vec{x} \equiv \langle x_1, \ldots, x_{N_i} \rangle$ with $N_i \equiv \lceil log_2\,(|H_i|) \rceil$.*

Moreover, for each H_i ($1 \leq i \leq r$), we have a set of mutually disjoint vectors $\vec{a} \equiv \langle a_1, \ldots, a_{N_i} \rangle$ with $a_j \in \{\top, \bot\}$, representing binary encodings for each constant symbol $a \in H_i$.

The basic idea for the subsequent translation is to express equations $x = y$ associated to a Herbrand universe H within the language of propositional logic by a conjunction of equivalences

$$x \Leftrightarrow_H y \equiv (x_1 \leftrightarrow y_1) \wedge \ldots \wedge (x_N \leftrightarrow y_N) \tag{10}$$

where $N \equiv \lceil log_2\,(|H|) \rceil$ and x is either a first-order variable x or an element $a \in H$. For the first case, $x_1, \ldots, x_N$ are the *propositional variables* from the corresponding vector $\vec{x}$. For the second case, $x_1, \ldots, x_N$ are the *propositional constants* $a_1, \ldots, a_N$ from the corresponding vector $\vec{a}$. Analogously, y is encoded by the corresponding vector of propositional variables or constants.

In addition, we have to be careful, whether assignments to propositional atoms from $At_{\mathcal{P}}$ have corresponding first-order assignments. For instance, consider a Herbrand universe H given by $\{a, b, c\}$, the corresponding vectors $\vec{a} \equiv \langle \bot, \bot \rangle$, $\vec{b} \equiv \langle \bot, \top \rangle$, and $\vec{c} \equiv \langle \top, \bot \rangle$ and a conjunction of disequations

$$x \neq a \wedge x \neq b \wedge x \neq c.$$

Obviously, the above equational problem is unsatisfiable over H. A straightforward encoding of this problem is given by

$$\neg D_{\bot,\bot} \wedge \neg D_{\bot,\top} \wedge \neg D_{\top,\bot} \tag{11}$$

where $D_{c,d} \equiv (x_1 \leftrightarrow c) \wedge (x_2 \leftrightarrow d)$ for $c, d \in \{\top, \bot\}$. But formula (11) is satisfiable under an interpretation which sets both x_1 and x_2 to true. Therefore, we should exclude such satisfying assignments, since there is no corresponding element in H to $\langle \top, \top \rangle$.

In other words, we have to define some form of *domain closure axiom* (DCA) which restricts truth-assignments to propositional atoms $x_1, \ldots, x_{N_i}$ in such a way that they represent only elements from that H_i which the first-order variable x is assigned to in a given equational problem $\mathcal{P}$. Straightforwardly, we define the propositional modules

$$DCA_H[x] \equiv \bigvee_{a \in H} (x \Leftrightarrow_H a), \tag{12}$$

for any Herbrand universe H. In certain cases, $DCA_H[x]$ can be optimized. For instance, if $|H| \equiv 2^N$ then $DCA[x]$ reduces to $\top$ for any variable x.

Definition 23 *Let $\mathcal{P}$ be an equational problem over a sequence $\mathcal{H} \equiv H_1, \ldots, H_r$ of finite Herbrand universes, and let $N_i \equiv \lceil log_2(|H_i|) \rceil$ for each $1 \leq i \leq r$. Then, $\mathcal{T}_{\mathcal{H}}[\mathcal{P}]$ is recursively defined using abbreviations (10) and (12) as follows:*

1. *$\mathcal{T}_{\mathcal{H}}[\top] \equiv \top$; $\mathcal{T}_{\mathcal{H}}[\bot] \equiv \bot$.*

2. *For $s = t$ associated to H_i, we define $\mathcal{T}_{\mathcal{H}}[s = t] \equiv s \Leftrightarrow_{H_i} t$.*

3. *For an equational problem $\mathcal{P} \equiv \neg\mathcal{P}_1$, we define $\mathcal{T}_{\mathcal{H}}[\mathcal{P}] \equiv \neg\mathcal{T}_{\mathcal{H}}[\mathcal{P}_1]$.*

4. *For equational problems $\mathcal{P}_1$, $\mathcal{P}_2$ and $\circ \in \{\wedge, \vee, \rightarrow\}$, we define*

$$\mathcal{T}_{\mathcal{H}}[\mathcal{P}_1 \circ \mathcal{P}_2] \equiv \mathcal{T}_{\mathcal{H}}[\mathcal{P}_1] \circ \mathcal{T}_{\mathcal{H}}[\mathcal{P}_2].$$

5. *For an equational problem $\mathsf{Q}x\,\mathcal{P}$, $\mathsf{Q} \in \{\exists, \forall\}$, and x a first-order variable associated to H_i in $\mathcal{P}$, we have*

$$\mathcal{T}_{\mathcal{H}}[\mathsf{Q}x\,\mathcal{P}] \equiv \mathsf{Q}x_1 \ldots \mathsf{Q}x_{N_i}\Big(DCA_{H_i}[x] \circ_{\mathsf{Q}} \mathcal{T}_{\mathcal{H}}[\mathcal{P}]\Big),$$

where $\circ_\exists \equiv \wedge$, and $\circ_\forall \equiv \rightarrow$.

Obviously, $\mathcal{T}_{\mathcal{H}}[\mathcal{P}]$ can be constructed in polynomial time (with respect to the size of $\mathcal{P}$). More precisely, we have the following upper bound. As in Section 4.1, we use $||\cdot||$ to denote the size of a dag representation of an equational problem as well as of a QBF. Moreover, recall that $Var(\mathcal{P})$ is used to refer to *all* first order variables in $\mathcal{P}$. Finally, $At(\Phi)$ denotes all propositional variables occurring in QBF Φ. The following theorem can be easily verified.

Theorem 24 *Let $\mathcal{P}$ be an equational problem over a sequence of finite Herbrand universes $\mathcal{H}$ and let $N \equiv \max\Big(\{|H| : H \in \mathcal{H}\}\Big)$. Then,*

$$\begin{aligned} ||\mathcal{T}_{\mathcal{H}}[\mathcal{P}]|| &\equiv O\Big(\lceil log_2(N) \rceil * ||\mathcal{P}|| + N * |Var(\mathcal{P})|\Big) \\ |At(\mathcal{T}_{\mathcal{H}}[\mathcal{P}])| &\equiv O\Big(\lceil log_2(N) \rceil * |Var(\mathcal{P})|\Big) \end{aligned}$$

The time required to compute $\mathcal{T}_{\mathcal{H}}[\mathcal{P}]$ is also bounded by

$$\lceil log_2(N) \rceil * ||\mathcal{P}|| + N * |Var(\mathcal{P})|.$$

The following result states the correctness and completeness of the transformation. Observe that $\mathcal{T}_{\mathcal{H}}[\mathcal{P}]$ yields a closed QBF whenever the given equational problem $\mathcal{P}$ is closed.

Theorem 25 *Let $\mathcal{P}$ be a closed equational problem over Herbrand universes $\mathcal{H}$. Then $\mathcal{P}$ is satisfiable over $\mathcal{H}$ iff $\mathcal{T}_{\mathcal{H}}[\mathcal{P}]$ evaluates to true.*

The proof is given in the appendix.

5.2 H-Subsumption via QBFs

This section summarizes all the encodings under consideration, and thus recapitulates the main results of the paper. Recall that our overall procedure is depicted in Figure 1.

We start with atomic H-subsumption over finite H. By Theorem 8, we already have a purely existential equational problem for a given co-ATOM-HSUB problem. Thus, putting things together leads to

Theorem 26 *Let $(\mathcal{A}, B, H)$ be an instance of the co-*ATOM-HSUB *problem where H is finite, and let $\mathcal{P}^+_{\mathcal{A},B}$ be of form (6) as defined in Theorem 8. Then $\mathcal{A} \not\leq^H_{ss} B$ holds, iff $\mathcal{T}_H[\mathcal{P}^+_{\mathcal{A},B}]$ evaluates to true.*

Note that $\mathcal{T}_H[\mathcal{P}^+_{\mathcal{A},B}]$ just contains positive occurrences of existential quantifiers over mutually disjoint propositional variables. Hence, all these quantifiers can be removed, yielding a traditional SAT-problem.

For clausal H-subsumption over finite H, there was no such simplification possible (which is, of course, witnessed by the computational complexity of this problem).

Theorem 27 *Let $(\mathcal{C}, D, H)$ be an instance of the co-*CLAUSE-HSUB *problem where H is finite, and let $\mathcal{P}_{\mathcal{C},D}$ be of form (3) as defined in Proposition 5. Then $\mathcal{C} \not\leq^H_{ss} D$ holds, iff $\mathcal{T}_H[\mathcal{P}_{\mathcal{C},D}]$ evaluates to true.*

Concerning the size of the resulting QBFs observe that, for a co-ATOM-HSUB problem $(\mathcal{A}, B, H)$, we have that $\mathcal{P}^+_{\mathcal{A},B}$ is linear in the size of $\mathcal{A}$ and B. Also the number of first-order variables in $\mathcal{P}^+_{\mathcal{A},B}$ is at most the number of first-order variables in $\mathcal{A}$ and B. Together with the results from Theorem 24 we thus get compact encodings. For clausal H-subsumption observe that we use $\mathcal{P}_{\mathcal{C},D}$ as defined in Proposition 5. Obviously, this equational problem is quadratic in the size of the given problem. Consequently, the size of $\mathcal{T}_H[\mathcal{P}_{\mathcal{C},D}]$ depends on $|H|$ and is quadratic in the size of the given problem.

We now proceed with those problems having an infinite Herbrand universe. Again, we first give the result for atomic H-subsumption.

Theorem 28 *Let $(\mathcal{A}, B, H)$ be an instance of the co-*ATOM-HSUB *problem where H is infinite, and let $\mathcal{P}'$ be the equational problem over the finite Herbrand universe H' as defined in Theorem 15. Then $\mathcal{A} \not\leq^H_{ss} B$ holds, iff $\mathcal{T}_{H'}[\mathcal{P}']$ evaluates to true.*

Similar as in in the finite case, the QBF $\mathcal{T}_{H'}[\mathcal{P}']$ just contains existential quantifiers which can be removed. We get a polynomial-time computable SAT-encoding for the NP-complete problem of atomic H-subsumption over infinite H.

Finally, we consider the clausal H-subsumption problem over an infinite Herbrand universe. Here, as discussed in Section 4.2, we derive a corresponding equational problem over two different sorts. This gives the justification why $\mathcal{T}[\cdot]$ was defined in such a general way, i.e., why it is capable of dealing with equational problems over different sorts as introduced in Definition 7.

Theorem 29 *Let $(\mathcal{C}, D, H)$ be an instance of the co-*CLAUSE-HSUB *problem where H is infinite, and let $\mathcal{Q}$ be the equational problem over the finite sorts H_1 and H_2 with the corresponding variables associations as defined in Theorem 18. Then $\mathcal{C} \not\leq^H_{ss} D$ holds, iff $\mathcal{T}_{H_1,H_2}[\mathcal{Q}]$ evaluates to true.*

For the size of the QBFs representing H-subsumption problems over an infinite Herbrand universe, recall the complexity bounds for the underlying equational problems (cf. Theorems 16 and 21). It is clear that, in these cases, the resulting QBFs get quite complicated.

5.3 Implementation Issues

Our methodology for expressing H-subsumption problems in terms of propositional logic and quantified Boolean formulas, respectively, is motivated by the availability of several practicably efficient solvers for these logics.

We already mentioned that removing the existential quantifiers from $\mathcal{T}_H[\mathcal{P}]$ where the equational problem $\mathcal{P}$ over H represents an encoding of an atomic H-subsumption problem, immediately gives us a traditional SAT-problem. Hence, invoking state-of-the-art SAT-solvers like Chaff [28], SATO [40], or Berkmin [15], to mention just a few, suffices to solve these problems.

For the clausal H-subsumption problems, which are computationally more complex, we actually employ QBF-solvers. Among the different tools available, there is a propositional theorem prover, `boole`[3], based on *binary decision diagrams*, a system using a generalized resolution principle [20], several provers implementing an extended Davis-Putnam procedure [2; 11; 14; 26; 36], as well as a distributed algorithm running on a PC-cluster [11]. With the exception of `boole`, these tools do not accept arbitrary QBFs, but require the input formula to be in *prenex conjunctive normal form*. To avoid an exponential increase of formula size in the translation of an arbitrary formula to (prenex) conjunctive normal form, *structure-preserving normal-form translations* [39; 34] can be used to translate a general QBF into the required normal form. In contrast to the usual normal-form translations based on distributivity laws, structure-preserving normal-form translations introduce new labels for sub-formula occurrences and are polynomial in the length of the input formula.

In the context of QBFs, such translations are usually achieved in two steps. First, one has to translate a QBF into prenex form, that are QBFs of form $\mathsf{Q}_1 P_1 \ldots \mathsf{Q}_n P_n \psi$ where ψ is a propositional formula without quantifier occurrences. The second step then deals with the usual translation of the purely propositional part ψ into conjunctive normal form.

Let us briefly illustrate the prenexing of QBFs $\mathcal{T}_{\mathcal{H}}[\mathcal{P}]$ where $\mathcal{P}$ represents a clausal H-subsumption problem over finite H. In Proposition 5, $\mathcal{P}$ is in general of the form $(\exists\vec{x})(\forall\vec{y})\mathcal{P}'$ with $\mathcal{P}'$ quantifier-free. Hence,

$$\begin{aligned}\mathcal{T}_{\mathcal{H}}[\mathcal{P}] \;\equiv\;& \exists X_1(DCA_H[x_1] \wedge \exists X_2(DCA_H[x_1] \wedge \ldots \wedge \\ &\wedge \exists X_n(DCA_H[x_n] \wedge \forall Y_1(DCA_H[y_1] \rightarrow \forall Y_2(DCA_H[y_2] \rightarrow \\ &\rightarrow \ldots \rightarrow \forall Y_m(DCA_H[y_m] \rightarrow \mathcal{T}_{\mathcal{H}}[\mathcal{P}'])\ldots)))\ldots)),\end{aligned}$$

where the X_i's and Y_j's denote the sequence of the propositional variables from the corresponding vectors $\vec{x}_i$ and $\vec{y}_j$, respectively. Applying typical shifting-rules [10] for QBFs, an equivalent prenex form of this QBF is constructed resulting in

$$\exists X_1 \ldots X_n \forall Y_1 \ldots Y_m \Big((\bigwedge_{i=1}^{n} DCA_H[x_i]) \wedge ((\bigwedge_{j=1}^{m} DCA_H[y_j]) \rightarrow \mathcal{T}_{\mathcal{H}}[\mathcal{P}'])\Big).$$

This translated encoding also shows the adequacy of our reduction since prenex QBFs with one quantifier alternation are known as the representative problems for the second level of the polynomial hierarchy. Recalling the complexity results for clausal H-subsumption this exactly matches the hardness of the original problem.

Similar results hold also for QBFs dealing with the infinite case of clausal H-subsumption.

[3]The system, together with its source code, can be downloaded from URL http://www.cs.cmu.edu/∼modelcheck/bdd.html.

6 Concluding Remarks

As already mentioned, H-subsumption is an indispensable redundancy elimination technique in automated model building. In this paper, we have presented a new approach to deciding (clausal and atomic) H-subsumption via a reduction to the well studied $QSAT_2$ and SAT problem, respectively. As a byproduct, we got a polynomial time transformation of a specific kind of equational problems with $\exists\forall$ prefix into a purely existentially quantified form.

Especially clausal H-subsumption has so far suffered from the lack of efficient algorithms, which is a genuine obstacle to its wider use. With the new method presented here, all the progress that has been made in the construction of efficient SAT and QBF solvers over the past years is now also accessible to the H-subsumption problem. This is a first step towards more efficient H-subsumption algorithms, to which further steps like the optimization of the translations or even a "real" implementation of the H-subsumption algorithms could be added.

Since H-subsumption does not destroy the refutational completeness of many familiar deduction systems, it is interesting to actually integrate (clausal) H-subsumption into an automated theorem prover. Practical experimentation will show in which cases the additional cost of H-subsumption (compared to ordinary subsumption) is outweighed by the gained power of redundancy elimination.

References

[1] Baader, F. and J. H. Siekmann: 1994, 'Unification Theory'. In: D. M. Gabbay, C. J. Hogger, and J. A. Robinson (eds.): *Handbook of Logic in Artificial Intelligence and Logic Programming*. pp. 41–125.

[2] Cadoli, M., A. Giovanardi, and M. Schaerf: 1998, 'An Algorithm to Evaluate Quantified Boolean Formulae'. In: *Proceedings of the 15th National Conference on Artificial Intelligence (AAAI'98)*. pp. 262–267.

[3] Caferra, R. and N. Peltier: 1996, 'Decision Procedures using Model Building Techniques'. In: H. K. Büning (ed.): *Proceedings of the 9th International Workshop for Computer Science Logic (CSL'95)*, Vol. 1092 of *LNCS*. pp. 130–144.

[4] Caferra, R. and N. Zabel: 1991, 'Extending Resolution for Model Construction'. In: J. van Eijck (ed.): *Logics in Artificial Intelligence (JELIA'90), Proceedings*, Vol. 478 of *LNCS*. pp. 153–169.

[5] Colmerauer, A.: 1984, 'Equations and Inequations on Finite and Infinite Trees'. In: *Proceedings of the International Conference on Fifth Generation Computer Systems (FGCS'84)*. pp. 85–99.

[6] Comon, H. and C. Delor: 1994, 'Equational Formulae with Membership Constraints'. *Information and Computation* **112**(2), 167–216.

[7] Comon, H. and P. Lescanne: 1989, 'Equational Problems and Disunification'. *Journal of Symbolic Computation* **7**(3/4), 371–425.

[8] Delgrande, J., T. Schaub, H. Tompits, and S. Woltran: 2001, 'On Computing Solutions to Belief Change Scenarios'. In: S. Benferhat and P. Besnard (eds.): *Proceedings of the 6th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU'01)*, Vol. 2143 of *LNCS*. pp. 510–521.

[9] Egly, U., T. Eiter, H. Tompits, and S. Woltran: 2000, 'Solving Advanced Reasoning Tasks Using Quantified Boolean Formulas'. In: *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI'00)*. pp. 417–422.

[10] Egly, U., H. Tompits, and S. Woltran: 2002, 'On Quantifier Shifting for Quantified Boolean Formulas'. In: *Proceedings of the SAT-02 Workshop on Theory and Applications of Quantified Boolean Formulas (QBF-02)*. pp. 48–61.

[11] Feldmann, R., B. Monien, and S. Schamberger: 2000, 'A Distributed Algorithm to Evaluate Quantified Boolean Formulas'. In: *Proceedings of the 17th National Conference on Artificial Intelligence (AAAI'00)*. pp. 285–290.

[12] Fermüller, C. and A. Leitsch: 1996, 'Hyperresolution and Automated Model Building'. *Journal of Logic and Computation* **6**(2), 173–203.

[13] Garey, M. R. and D. S. Johnson: 1979, *Computers and Intractability, A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company.

[14] Giunchiglia, E., M. Narizzano, and A. Tacchella: 2001, 'QuBE: A System for Deciding Quantified Boolean Formulas Satisfiability'. In: R. Goré, A. Leitsch, and T. Nipkow (eds.): *Proceedings of the International Joint Conference on Automated Reasoning (IJCAR'01)*. pp. 364–369.

[15] Goldberg, E. and Y. Novikov: 2002, 'BerkMin: A Fast and Robust Sat-Solver'. In: *Proceedings of the Design, Automation and Test in Europe Conference (DATE'02)*. pp. 142–149.

[16] Gottlob, G., S. Marcus, A. Nerode, G. Salzer, and V. S. Subrahmanian: 1996, 'A Non-Ground Realization of the Stable and Well-Founded Semantics'. *Theoretical Computer Science* **166**(1&2), 221–262.

[17] Gottlob, G. and R. Pichler: 2001, 'Working with ARMs: Complexity Results on Atomic Representations of Herbrand Models'. *Information and Computation* **165**, 183–207.

[18] Kapur, D., P. Narendran, D. Rosenkrantz, and H. Zhang: 1991, 'Sufficient-Completeness, Ground-Reducibility and their Complexity'. *Acta Informatica* **28**(4), 311–350.

[19] Kautz, H. and B. Selman: 1992, 'Planning as Satisfiability'. In: B. Neumann (ed.): *Proceedings of the 10th European Conference on Artificial Intelligence (ECAI'92)*. pp. 359–363.

[20] Kleine-Büning, H., M. Karpinski, and A. Flögel: 1995, 'Resolution for Quantified Boolean Formulas'. *Information and Computation* **117**(1), 12–18.

[21] Kunen, K.: 1987, 'Answer Sets and Negation as Failure'. In: J.-L. Lassez (ed.): *Proceedings of the 4th International Confernence on Logic Programming (ICLP'87)*. pp. 219–228.

[22] Kuper, G., K. McAloon, K. Palem, and K. Perry: 1988, 'Efficient Parallel Algorithms for Anti-Unification and Relative Complement'. In: *Proceedings of the 3rd Annual Symposium on Logic in Computer Science (LICS'88)*. pp. 112–120.

[23] Lassez, J.-L., M. Maher, and K. Marriott: 1986, 'Unification Revisited'. In: *Foundations of Logic and Functional Programming, Proceedings*. pp. 67–113.

[24] Lassez, J.-L., M. Maher, and K. Marriott: 1991, 'Elimination of Negation in Term Algebras'. In: A. Tarlecki (ed.): *Proceedings of the 16th International Symposium on Mathematical Foundations of Computer Science (MFCS'91)*. pp. 1–16.

[25] Lassez, J.-L. and K. Marriott: 1987, 'Explicit Representation of Terms Defined by Counter Examples'. *Journal of Automated Reasoning* **3**(3), 301–317.

[26] Letz, R.: 2002, 'Lemma and Model Caching in Decision Procedures for Quantified Boolean Formulas'. In: U. Egly and C. G. Fermüller (eds.): *Automated Reasoning with Analytic Tableaux and Related Methods, International Conference, TABLEAUX 2002*, Vol. 2381 of *LNCS*. pp. 160–175.

[27] Martelli, A. and U. Montanari: 1982, 'An Efficient Unification Algorithm'. *ACM Transactions on Programming Languages and Systems* **4**(2), 258–282.

[28] Moskewicz, M. W., C. F. Madigan, Y. Zhao, L. Zhang, and S. Malik: 2001, 'Chaff: Engineering an Efficient SAT Solver'. In: *Proceedings of the 38th Design Automation Conference (DAC'01)*.

[29] Papadimitriou, C. H.: 1994, *Computational Complexity*. Addison-Wesley.

[30] Pearce, D., H. Tompits, and S. Woltran.: 2001, 'Encodings for Equilibrium Logic and Logic Programs with Nested Expressions'. In: P. Brazdil and A. Jorge (eds.): *Progress in Artificial Intelligence, Knowledge Extraction, Multi-agent Systems, Logic Programming and Constraint Solving, 10th Portuguese Conference on Artificial Intelligence (EPIA'01)*, Vol. 2258 of *LNCS*. pp. 306–320.

[31] Pichler, R.: 1998, 'On the Complexity of H-Subsumption'. In: G. Gottlob, E. Grandjean, and K. Seyr (eds.): *Proceedings of the 12th International Workshop for Computer Science Logic (CSL'98)*, Vol. 1584 of *LNCS*. pp. 355–371.

[32] Pichler, R.: 2000a, 'Completeness and Redundancy in Constrained Clause Logic'. In: R. Caferra and G. Salzer (eds.): *Automated Deduction in Classical and Non-Classical Logics, Selected Papers*, Vol. 1761 of *LNCS*. pp. 221–235.

[33] Pichler, R.: 2000b, 'The Explicit Representability of Implicit Generalizations'. In: L. Bachmair (ed.): *Proceedings of the 11th International Confernece on Rewriting Techniques and Applications (RTA'00)*. pp. 187–202.

[34] Plaisted, D. A. and S. Greenbaum: 1986, 'A Structure Preserving Clause Form Translation'. *Journal of Symbolic Computation* **2**(3), 293–304.

[35] Rintanen, J.: 1999a, 'Constructing Conditional Plans by a Theorem Prover'. *Journal of Artificial Intelligence Research* **10**, 323–352.

[36] Rintanen, J.: 1999b, 'Improvements to the Evaluation of Quantified Boolean Formulae'. In: T. Dean (ed.): *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI'99)*. pp. 1192–1197.

[37] Siekmann, J. and G. Wrightson (eds.): 1983, *Automation of Reasoning: Classical Papers in Computational Logic 1967–1970*, Vol. 2. Springer Verlag.

[38] Stockmeyer, L.: 1977, 'The Polynomial-Time Hierarchy'. *Theoretical Computer Science* **3**, 1–22.

[39] Tseitin, G.: 1970, 'On the Complexity of Proofs in Propositional Logics'. *Seminars in Mathematics* **8**. Reprinted in [37].

[40] Zhang, H.: 1997, 'SATO: An Efficient Propositional Prover'. In: W. McCune (ed.): *Proceedings of the 17th Internationonal Confernence on Automated Deduction (CADE-17)*. pp. 272–275.

A Proofs

Proof of Lemma 10.
As for the *termination*, we only have to consider the iterations of the rules (T1) – (T3). Note that by an application of the transformation rule (T1) followed by (T2) and (T3), the disequation $x_j \neq s_j$ with the non-variable term s_j on the right-hand side is either deleted or it is replaced by a disjunction of disequations, such that the terms $r_1, \ldots, r_{\alpha(f)}$ on the right-hand side have strictly smaller term-depth. Hence, eventually, this iteration must indeed terminate.

The *completeness* of the transformation (and, hence, the desired form of the subproblems $\mathcal{Q}_i$) is clear by our comments in Section 4.1.

The *correctness* of the transformation rules (T0) – (T3) follows immediately from the correctness of the rules E, U_2, R_1 and R_2 from Figure 2. We just have to be a bit careful with our notation for the new variables. Suppose that the explosion rule is applied to some variable x_j in two different subproblems $MGU(A_i, B)$ and $MGU(A_{i'}, B)$ of $\mathcal{P}$. Then we introduce the same new variables $x_j[(g,1)], \ldots, x_j[(g,\alpha(g))]$ in both subproblems. Likewise, when applying the explosion rule recursively to some new variable $x_j[w]$ in two different subproblems, then we again introduce the same new variables $x_j[w,(g,1)]$, …, $x_j[w,(g,\alpha(g))]$ in both subproblems. It can be easily checked that this is allowed. To see this, just suppose that we add the trivially true disjunction $\bigvee_{g \in \Sigma} x_j[w] = g(x_j[w,(g,1)], \ldots, x_j[w,(g,\alpha(g))])$ as a new conjunct on the outermost level of the equational problem $\mathcal{P}$. $\diamond$

Proof of Lemma 13.
Before we start with the actual proof, we recall an important fact concerning the satisfiability of (purely existentially quantified) equational problems over an *infinite* Herbrand universe, namely the so-called *independence of inequations* of [5]:

Proposition 30 *Every purely existentially quantified conjunction of disequations over an infinite Herbrand universe has at least one solution, if and only if each of the conjuncts has a solution.*

From this, the following criterion for testing the satisfiability of an existentially quantified conjunction of equations and disequations over an *infinite* Herbrand universe can be easily derived (e.g., see [33]):

Proposition 31 *Let* $\mathcal{P} \equiv (\exists \vec{x})[e_1 \wedge \ldots \wedge e_k \wedge d_1 \wedge \ldots \wedge d_l]$ *be a conjunction of equations* e_i *and disequations* d_j *over some infinite universe* H. *Then the satisfiability of* $\mathcal{P}$ *can be tested as follows:*

- Case 1*: If* $e_1 \wedge \ldots \wedge e_k$ *is unsatisfiable, then* $\mathcal{P}$ *is also unsatisfiable.*
- Case 2*: Let* $e_1 \wedge \ldots \wedge e_k$ *be satisfiable and let* ϑ *be the* mgu *of* $e_1, \ldots, e_k$. *Then* $\mathcal{P}$ *is satisfiable, iff* $d_1\vartheta \wedge \ldots \wedge d_l\vartheta$ *contains no trivial disequation of the form* $t \neq t$.

We are now prepared to prove the equivalence between the equational problems $\mathcal{Q} \equiv (\exists \vec{x}) \bigwedge_{i=1}^n \mathcal{Q}_i$ and $\mathcal{P}' \equiv (\exists \vec{x}) \bigwedge_{i=1}^n \mathcal{P}'_i$. Recall that

$$\begin{aligned}
\mathcal{Q}_i &\equiv \mathcal{E}_{i1} \vee \ldots \vee \mathcal{E}_{iN_i} \vee (\mathcal{E}_{i0} \wedge d_{i1}) \vee \ldots \vee (\mathcal{E}_{i0} \wedge d_{iM_i}) \\
\mathcal{P}'_i &\equiv \mathcal{E}'_{i1} \vee \ldots \vee \mathcal{E}'_{iN_i} \vee (\mathcal{E}'_{i0} \wedge \mathcal{D}_{i1}) \vee \ldots \vee (\mathcal{E}'_{i0} \wedge \mathcal{D}_{iM_i})
\end{aligned}$$

where each $\mathcal{D}_{ij}$ is a disjunction of the form $\bigvee_{v \in V} \mathcal{D}(v)$ with

$$\begin{aligned}\mathcal{D}(v) \equiv\ & x_j[w] = x_{j'}[w'] = a_1 \wedge \\ & x_j[w,(g_1,\gamma_1)] = x_{j'}[w',(g_1,\gamma_1)] = a_2 \wedge \ldots \wedge \\ & x_j[w,(g_1,\gamma_1),\ldots,(g_{k-1},\gamma_{k-1})] = \\ & \quad x_{j'}[w',(g_1,\gamma_1),\ldots,(g_{k-1},\gamma_{k-1})] = a_k \wedge \\ & x_j[w,(g_1,\gamma_1),\ldots,(g_k,\gamma_k)] \neq x_{j'}[w',(g_1,\gamma_1),\ldots,(g_k,\gamma_k)].\end{aligned}$$

Note that we have applied the distributivity law of $\wedge$ and $\vee$ in order to turn every subproblem $\mathcal{Q}_i$ and $\mathcal{P}'_i$ into a disjunction of conjunctions. Of course, this slight modification of the formulas from Section 4.1 is correct. We have to prove that $\mathcal{Q}$ is satisfiable iff $\mathcal{P}'$ is satisfiable.

Proof of "$\Rightarrow$".
If $\mathcal{Q}$ is satisfiable, then we can select from every subproblem $\mathcal{Q}_i$ a disjunct $\mathcal{E}_{ij}$ or $(\mathcal{E}_{i0} \wedge d_{ij})$, such that the conjunction of these selected disjuncts is satisfiable. Note that we are thus dealing with a conjunction of equations and disequations. Hence, by Proposition 31 above, this means that the selected equations are unifiable, with mgu μ say, and the application of μ to the selected disequations does not produce a trivially false disequation of the form $t \neq t$.

We claim that then it is possible to select from every subproblem $\mathcal{P}'_i$ a disjunct, such that the conjunction of these selected disjuncts is also satisfiable. In fact, if, in the i-th subproblem $\mathcal{Q}_i$, $\mathcal{E}_{ij}$ was selected, then we select the corresponding $\mathcal{E}'_{ij}$ in $\mathcal{P}'_i$. Likewise, if, in $\mathcal{Q}_i$, $(\mathcal{E}_{i0} \wedge d_{ij})$ was selected, then we select the corresponding $(\mathcal{E}'_{i0} \wedge \mathcal{D}_{ij})$ in $\mathcal{P}'_i$. Moreover, from the substitution μ on the variables $\vec{x}$ in $\mathcal{Q}$, we define the following substitution σ on the variables $\vec{x}$ in $\mathcal{P}'$:

$$x_j[w]\sigma \equiv \begin{cases} a_\gamma & \text{if } x_j[w]\mu \text{ has leading symbol } g_\gamma \in \Sigma \\ c_j[w] & \text{otherwise} \end{cases}$$

It remains to show that σ is indeed a solution of each selected formula $\mathcal{E}'_{ij}$ and $(\mathcal{E}'_{i0} \wedge \mathcal{D}_{ij})$ in $\mathcal{P}'$, respectively.

(1) σ and the formulas $\mathcal{E}'_{ij}$.
Recall that every $\mathcal{E}'_{ij}$ is a conjunction of equations of the form $x_j[w] = a_\gamma$ for some $a_\gamma \in H'$. By the construction of $\mathcal{E}'_{ij}$, we know that, for every such equation $x_j[w] = a_\gamma$, the corresponding formula $\mathcal{E}_{ij}$ in $\mathcal{Q}$ contains an equation of the form $x_j[w] = g_\gamma(x_j[w,(g_\gamma,1)],\ldots,x_j[w,(g_\gamma,\alpha(g_\gamma))])$. Hence, the mgu μ of the selected equations in $\mathcal{Q}$ clearly binds the variable $x_j[w]$ to a term with leading symbol g_γ. But then, by the definition of σ, we may conclude that $x_j[w]\sigma$ is indeed set to a_γ.

(2) σ and the formulas $(\mathcal{E}'_{i0} \wedge \mathcal{D}_{ij})$.
We know from (1) that σ indeed validates all equations in $\mathcal{E}'_{i0}$. Hence, it only remains to show that σ also validates $\mathcal{D}_{ij}$. Recall that the disequation d_{ij} is of the form $x_j[w] \neq x_{j'}[w']$. Moreover, by assumption, $d_{ij}\mu$ is not a trivially false disequation. In other words, $x_j[w]\mu$ and $x_{j'}[w']\mu$ are syntactically different terms. Hence, there exists a position p in both terms $x_j[w]\mu$ and $x_{j'}[w']\mu$, such that the corresponding subterms at this position have different leading symbols. Let p be minimal with this property. We have to distinguish two cases depending on whether p is the empty position or not.

CASE 1: Suppose that p is the empty position.
In other words, $x_j[w]\mu$ and $x_{j'}[w']\mu$ have different leading symbols. Recall that the formula $\mathcal{D}_{ij}$ contains a disjunct of the form $\mathcal{D}(\epsilon) \equiv x_j[w] \neq x_{j'}[w']$. In order to show that $\mathcal{D}(\epsilon)$ (and, hence, $\mathcal{D}_{ij}$) is validated by σ, we distinguish three subcases as to whether 0, 1 or 2 of the variables $x_j[w]$ and $x_{j'}[w']$ are unbound under μ.

CASE 1.1: If both of them are bound to some term with leading symbol g_γ and $g_{\gamma'}$, respectively, then these two symbols must be distinct. Hence, by the definition of σ, the variables $x_j[w]$ and $x_{j'}[w']$ are bound to two distinct constants $a_\gamma \in H'$ and $a_{\gamma'} \in H'$, respectively, under σ.

CASE 1.2: Now suppose that exactly one of the variables $x_j[w]$ and $x_{j'}[w']$ is bound by μ to some term with leading symbol g_γ, say the first one. Then we have $x_j[w]\sigma = a_\gamma$ and $x_{j'}[w']\sigma = c_{j'}[w']$, where a_γ and $c_{j'}[w']$ are distinct by the construction of H'.

CASE 1.3: Finally, if both $x_j[w]$ and $x_{j'}[w']$ are unbound variables, then we have $x_j[w]\sigma = c_j[w]$ and $x_{j'}[w']\sigma = c_{j'}[w']$ for two distinct constants $c_j[w]$ and $c_{j'}[w']$.

CASE 2: Suppose that p is not the empty position.
In this case, $p \equiv i_1 \dots i_t$ for some $t \geq 1$. By the minimality of p, we know that, for every s with $s < t$, the subterms in $x_j[w]\mu$ and $x_{j'}[w']\mu$ at the position $i_1 \dots i_s$ have the same leading symbol from Σ, say g_s. By the form of the equations in $\mathcal{Q}$ and by the fact that μ is the mgu of such equations, the following equations must be among the selected ones:

$$
\begin{aligned}
&x_j[w] = g_1(x_j[w,(g_1,1)],\dots,x_j[w,(g_1,\alpha(g_1))]) \\
&x_{j'}[w'] = g_1(x_{j'}[w',(g_1,1)],\dots,x_{j'}[w',(g_1,\alpha(g_1))]) \\
&x_j[w,(g_1,i_1)] = g_2(x_j[w,(g_1,i_1),(g_2,2)],\dots,x_j[w,(g_1,i_1),(g_2,\alpha(g_2))]) \\
&x_{j'}[w',(g_1,i_1)] = g_2(x_{j'}[w',(g_1,i_1),(g_2,2)],\dots,x_{j'}[w',(g_1,i_1),(g_2,\alpha(g_2))]) \\
&\qquad\vdots \\
&x_j[w,(g_1,i_1),\dots,(g_{t-1},i_{t-1})] = g_t(x_j[w,(g_1,i_1),\dots,(g_{t-1},i_{t-1}), \\
&\qquad (g_t,1)],\dots,x_j[w,(g_1,i_1),\dots,(g_{t-1},i_{t-1}),(g_t,\alpha(g_t))]) \\
&x_{j'}[w',(g_1,i_1),\dots,(g_{t-1},i_{t-1})] = g_t(x_{j'}[w',(g_1,i_1),\dots,(g_{t-1},i_{t-1}), \\
&\qquad (g_t,1)],\dots,x_{j'}[w',(g_1,i_1),\dots,(g_{t-1},i_{t-1}),(g_t,\alpha(g_t))]).
\end{aligned}
$$

In particular, both variables $x_j[w,v]$ and $x_{j'}[w',v]$ with $v \equiv (g_1,i_1),\dots,(g_t,i_t)$ actually exist in $\mathcal{Q}$. Hence, $\mathcal{D}_{ij}$ contains the disjunct $\mathcal{D}(v)$ with

$$
\begin{aligned}
\mathcal{D}(v) \equiv\ & x_j[w] = x_{j'}[w'] = a_1 \wedge \\
& x_j[w,(g_1,i_1)] = x_{j'}[w',(g_1,i_1)] = a_2 \wedge \dots \wedge \\
& x_j[w,(g_1,i_1),\dots,(g_{t-1},i_{t-1})] = \\
& \quad x_{j'}[w',(g_1,i_1),\dots,(g_{t-1},i_{t-1})] = a_k \wedge \\
& x_j[w,(g_1,i_1),\dots,(g_t,i_t)] \neq x_{j'}[w',(g_1,i_1),\dots,(g_t,i_t)].
\end{aligned}
$$

It is easy to check that σ indeed validates the equations in $\mathcal{D}(v)$. In order to show that also the disequation $x_j[w,v] \neq x_{j'}[w',v]$ with $v \equiv (g_1,i_1),\dots,(g_t,i_t)$ is validated by σ, we have to distinguish three subcases as to whether 0, 1 or 2 of the variables $x_j[w,v]$ and $x_{j'}[w',v]$ are unbound under μ. This case distinction goes analogously to the one in Case 1 above and is therefore omitted.

Proof of "$\Leftarrow$".
If $\mathcal{P}'$ is satisfiable, then we can select from every subproblem $\mathcal{P}'_i$ a disjunct $\mathcal{E}'_{ij}$ or $(\mathcal{E}_{i0} \wedge \mathcal{D}_{ij})$, such that the conjunction of these selected disjuncts is satisfiable. Let σ be a solution of this conjunction.

Analogously to the "$\Rightarrow$"-direction of the proof, we select the corresponding disjuncts $\mathcal{E}_{ij}$ and d_{ij} from $\mathcal{Q}$ and show that their conjunction is also satisfiable. By Proposition 31, we have to show that the selected equations are unifiable, and that the application of the mgu to the selected disequations does not produce a trivially false disequation. Actually, it suffices to show that there exists any unifier μ (i.e., not necessarily the mgu) of the equations with this property. We define the substitution μ as follows:

$$x_j[w]\mu = \begin{cases} g_\gamma(x_j[w,(g_\gamma,1)],\ldots,x_j[w,(g_\gamma,\alpha(g_\gamma))]) & \text{if } x_j[w]\sigma = a_\gamma \\ x_j[w] & \text{otherwise} \end{cases}$$

We claim that μ is a substitution with the desired property.

It is easy to check that μ is indeed a unifier of the equations in the selected formulas $\mathcal{E}_{ij}$ in $\mathcal{Q}$. Now consider the disequations d_{ij}, which are of the form $x_j[w] \neq x_{j'}[w']$. Let us assume that $x_j[w]\mu$ and $x_{j'}[w']\mu$ are syntactically identical terms. We have to show that this assumption leads to a contradiction. Recall that $\mathcal{D}_{ij}$ is a disjunction of the form $\bigvee_{v\in V}\mathcal{D}(v)$. By assumption, σ validates this disjunction. Hence, there exists a word $v \equiv (g_1,i_1),\ldots,(g_t,i_t)$, such that the formula $\mathcal{D}(v)$ is validated by σ. Note that, by the definition of $\mathcal{D}(v)$ in Section 4.1, the symbols $g_1,\ldots,g_t$ must be proper function symbols in Σ. Hence, by the definition of μ, we know that $i_1\ldots i_s$ is indeed a position in the terms $x_j[w]\mu$ and $x_{j'}[w']\mu$, respectively. Moreover, since g_t is a proper function symbol, also the position $i_1\ldots i_t$ indeed exists in the terms $x_j[w]\mu$ and $x_{j'}[w']\mu$. Now we are ready to show that the above assumption $x_j[w]\mu \equiv x_{j'}[w']\mu$ leads to a contradiction. Since $x_j[w]\mu \equiv x_{j'}[w']\mu$ are identical, the subterms at every joint position of these two terms must be identical. Hence, since $v \equiv (g_1,i_1),\ldots,(g_t,i_t)$ is a joint position in these two terms, we know that $x_j[w,(g_1,i_1),\ldots,(g_t,i_t)]\mu$ and $x_{j'}[w',(g_1,i_1),\ldots,(g_t,i_t)]\mu$ are identical terms. Actually, the variables $x_j[w,(g_1,i_1),\ldots,(g_t,i_t)]$ and $x_{j'}[w',(g_1,i_1),\ldots,(g_t,i_t)]$ as such are distinct. Hence, in order to become identical under μ, they have to be bound to the same non-variable term. Let g_γ denote the leading symbol of this terms. Then, by the definition of μ, we have $x_j[w,(g_1,i_1),\ldots,(g_t,i_t)]\sigma = a_\gamma$ and $x_{j'}[w',(g_1,i_1),\ldots,(g_t,i_t)]\sigma = a_\gamma$. But this contradicts the fact that σ validates $\mathcal{D}(v)$, which contains the disequation $x_j[w,(g_1,i_1),\ldots,(g_t,i_t)] \neq x_{j'}[w',(g_1,i_1),\ldots,(g_t,i_t)]$. ◇

Proof Sketch of Theorem 16.
Let $\mathcal{A} \equiv \{A_1,\ldots,A_n\}$. After the transformation (T0), which is essentially a unification step, we have an equational formula of n conjuncts such that the size of their dag-representation is bounded by $||\mathcal{A}|| + ||B||$ each. Moreover, the number of existentially quantified variables in each conjunct is bounded by $|\mathit{Var}(B)|$. Now consider $\mathcal{Q} \equiv (\exists\vec{x})\bigwedge_{i=1}^{n}\mathcal{Q}_i$ obtained by exhaustive application of the transformation rules (T1) – (T3). We get the following upper bounds for each $\mathcal{Q}_i$.

First of all note that the number of iterations of these rules to get $\mathcal{Q}_i$ is bounded by $||\mathcal{A}|| + ||B||$. This is due to the fact that the (reverse) R_2-rule guarantees that no subterm on the right-hand side of any disequation is decomposed more than once. Hence, the rule (T1) is never applied twice to any subterm occurring in any disequation. Note that without these R_2-rule applications, we would only get an upper bound that corresponds to the *number of positions* in $s_1,\ldots,s_k$. However, these terms, which result from unification of the atoms A_i and B, may have exponentially many positions (cf. [1], [27]).

Each application of (T1) introduces $\mathit{arity}(\Sigma)$ new variables and increases the formula by $O\Big(\mathit{size}(\Sigma)\Big)$. On each application of (T2) and (T3), no new variables are introduced. The size of the formula is mainly increased by the application of the distributivity. A bound on this increase is obtained as follows. Recall that on each iteration, we transform a formula of the form $\mathcal{E}_1 \vee \ldots \vee \mathcal{E}_N \vee [\mathcal{E}_0 \wedge (d_1 \vee \ldots \vee d_M)]$ into

$$\mathcal{E}_1 \vee \ldots \vee \mathcal{E}_N \vee [\bigvee_{g\neq f}\mathcal{E}_0 \wedge (x = g(.))] \vee [\mathcal{E}_0 \wedge x = f(.) \wedge (d_1 \vee \ldots \vee d_M)].$$

For the next iteration, the formula $\mathcal{E}_0' \equiv \mathcal{E}_0 \wedge x = f(.)$ plays the role of $\mathcal{E}_0$. This transition from $\mathcal{E}_0$ to $\mathcal{E}_0 \wedge x = f(.)$ means an increase by at most $O\Big(\mathit{size}(\Sigma)\Big)$. Hence, by the bound $||\mathcal{A}|| + ||B||$ on the number of iterations, $||\mathcal{E}_0||$ is bounded by $O\Big(||(\mathcal{A}|| + ||B||) * size(\Sigma)\Big)$. Note that the above sketched application of the distributivity basically replaces the subformula $\mathcal{E}_0$ by the disjunction $[\bigvee_{g\in\Sigma} \mathcal{E}_0 \wedge (x = g(.))]$ Hence, the resulting increase of the formula (on each iteration) is bounded by $O\Big(||\mathcal{E}_0|| * size(\Sigma)\Big)$. Putting these pieces together, we get

$$\begin{array}{rcl} |Var(\mathcal{Q}_i)| & \leq & |Var(B)| + (||\mathcal{A}|| + ||B||) * arity(\Sigma) \\ & \leq & (||\mathcal{A}|| + ||B||) * size(\Sigma) \\ ||\mathcal{Q}_i|| & \equiv & O\Big((||\mathcal{A}|| + ||B||)^2 * size(\Sigma)^2\Big). \end{array}$$

The upper bounds for $|Var(\mathcal{Q})|$ and $||\mathcal{Q}||$ are simply obtained by multiplying the above ones by $n \equiv |\mathcal{A}|$. For the equational problem $\mathcal{P}' \equiv (\exists\vec{x}\,) \bigwedge_{i=1}^{n} \mathcal{P}'_i$ over H', we clearly have $H' \leq |Var(Q)| + |\Sigma|$. For the conjuncts $\mathcal{P}'_i$ of $\mathcal{P}'$, we get the following bounds:

The total size of all the equations $\mathcal{E}'_j$ in $\mathcal{P}'_i$ is bounded by $O\Big(||\mathcal{Q}_i||\Big)$. As for the total size of the disjunctions $\bigvee_{v\in V} \mathcal{D}(v)$ encoding the disequations from $\mathcal{Q}_i$, we get the following bound: Recall that each iteration of the rules (T1) – (T3) decomposes one disequation into at most $arity(\Sigma)$ simpler ones. Hence, the number of disequations after $||\mathcal{A}|| + ||B||$ iterations is bounded by $(||\mathcal{A}|| + ||B||) * size(\Sigma)$. For any disequation $x_j[w] \neq x_{j'}[w']$, there exist at most $|Var(\mathcal{Q})|$ words v, s.t. both $x_j[w, v]$ and $x_{j'}[w', v]$ are variables in $\mathcal{Q}$. Finally, the length of each word v is bounded by $O\Big(|Var(\mathcal{Q})|\Big)$. In summary, we get the upper bound $O\Big((||\mathcal{A}|| + ||B||) * size(\Sigma) * |Var(\mathcal{Q})|^2\Big)$ on the total size of all disjuncts $\bigvee_{v\in V} \mathcal{D}(v)$ in $\mathcal{P}'_i$ and the overall bound $O\Big(n * (||\mathcal{A}|| + ||B||) * size(\Sigma) * |Var(\mathcal{Q})|^2\Big)$ on the size of $\mathcal{P}'$. Moreover, it is easy to check that the time required for the whole transformation also fits into the latter bound. $\diamond$

Proof Sketch of Lemma 17.
We consider the following criterion for first-order subsumption. The relation $C \not\leq_{ss} D$ holds, iff, for every tuple $\langle j_1, \ldots, j_l\rangle \in \{1, \ldots, m\}^l$, $\langle L_1, \ldots, L_l\rangle \not\leq_s \langle M_{j_1}, \ldots, M_{j_l}\rangle$ holds. Hence, for H-subsumption, we get that $C \not\leq^H_{ss} D$ is equivalent to the condition that there exists a substitution ϑ on the variables in $Var(D)$, such that for every tuple $\langle j_1, \ldots, j_l\rangle \in \{1, \ldots, m\}^l$, $\langle L_1, \ldots, L_l\rangle \not\leq_s \langle M_{j_1}\vartheta, \ldots, M_{j_l}\vartheta\rangle$ holds.

Now suppose that we check $\langle L_1, \ldots, L_l\rangle \not\leq_s \langle M_{j_1}\vartheta, \ldots, M_{j_l}\vartheta\rangle$ by a straightforward matching algorithm. Then we either search for a non-variable position p in $\langle L_1, \ldots, L_l\rangle$, such that the leading symbol of the subterm of $\langle L_1, \ldots, L_l\rangle$ at position p is different form the leading symbol of the corresponding subterm at position p in $\langle M_{j_1}\vartheta, \ldots, M_{j_l}\vartheta\rangle$. Or we try to find two variable positions p_1 and p_2 in $\langle L_1, \ldots, L_l\rangle$ with the same variable x, such that these two occurrences of x in $\langle L_1, \ldots, L_l\rangle$ are matched against two distinct terms at the corresponding positions in $\langle M_{j_1}\vartheta, \ldots, M_{j_l}\vartheta\rangle$. In order to detect that $\langle L_1, \ldots, L_l\rangle$ does not match $\langle M_{j_1}\vartheta, \ldots, M_{j_l}\vartheta\rangle$, it therefore suffices to check whether the following condition holds: There exists an index $\alpha \in \{1, \ldots, l\}$, such that $L_\alpha \not\leq_s M_{j_\alpha}$ or there exist indices $\alpha, \beta \in \{1, \ldots, l\}$, such that $\langle L_\alpha, L_\beta\rangle \not\leq_s \langle M_{j_\alpha}\vartheta, M_{j_\beta}\vartheta\rangle$ holds. Actually, $\langle L_\alpha, L_\beta\rangle \not\leq_s \langle M_{j_\alpha}\vartheta, M_{j_\beta}\vartheta\rangle$ holds, iff $\langle L_\beta, L_\alpha\rangle \not\leq_s \langle M_{j_\beta}\vartheta, M_{j_\alpha}\vartheta\rangle$ holds, Moreover, if $L_\alpha \not\leq_s M_{j_\alpha}\vartheta$, then $\langle L_\alpha, L_\beta\rangle \not\leq_s \langle M_{j_\alpha}\vartheta, M_{j_\beta}\vartheta\rangle$ holds for any β. But then the new H-subsumption criterion follows immediately. $\diamond$

Proof of Theorem 18.
Let $C \equiv L_1 \vee \ldots \vee L_l$ and $D \equiv M_1 \vee \ldots \vee M_m$ be clauses over the infinite Herbrand universe H and let $(\exists\vec{x}\,)(\forall j_1, \ldots, j_l)\mathcal{Q}$ be the equational problem over the finite universes H_1 and H_2 from Section 4.2. We have to prove that $C \not\leq^H_{ss} D$ holds iff $(\exists\vec{x}\,)(\forall j_1, \ldots, j_l)\mathcal{Q}$ is satisfiable.

Proof of "$\Rightarrow$".
Suppose that $C \not\leq_{ss}^{H} D$ holds. By Lemma 17, there exists a substitution ϑ on $Var(D)$, such that, for every tuple $\langle j_1, \ldots, j_l \rangle \in \{1, \ldots, m\}^l$, there exist two indices α, β with $1 \leq \alpha < \beta \leq l$ and $\langle L_\alpha, L_\beta \rangle \not\leq_s \langle M_{j_\alpha}\vartheta, M_{j_\beta}\vartheta \rangle$.

Let Ω be defined as the set of all quadruples $\langle \alpha, \beta, \gamma, \delta \rangle$, such that $\langle L_\alpha, L_\beta \rangle \not\leq_s \langle M_\gamma\vartheta, M_\delta\vartheta \rangle$ holds. For every $\langle j_1, \ldots, j_l \rangle \in \{1, \ldots, m\}^l$, Ω contains at least one quadruple $\langle \alpha, \beta, \gamma, \delta \rangle$ with $j_\alpha \equiv \gamma$ and $j_\beta \equiv \delta$. Now let us consider the following equational problem

$$\mathcal{R} \equiv (\exists \vec{x}\,)(\forall \vec{y}\,) \bigwedge_{(\alpha,\beta,\gamma,\delta)\in\Omega} \langle L^*_\alpha, L^*_\beta \rangle \neq \langle M^*_\gamma, M^*_\delta \rangle$$

over the infinite Herbrand universe H. $\mathcal{R}$ is satisfiable by the construction of Ω. Therefore, analogously to Lemma 13, it can be shown that also the equational problem

$$\mathcal{R}' \equiv (\exists \vec{x}\,) \bigwedge_{(\alpha,\beta,\gamma,\delta)\in\Omega} P(\alpha, \beta, \gamma, \delta)$$

(over H_1) is satisfiable. In other words, there exists a substitution σ on the variables $\vec{x}$, such that, for every quadruple $\langle \alpha, \beta, \gamma, \delta \rangle \in \Omega$, the conjunct $P(\alpha, \beta, \gamma, \delta)$ in $\mathcal{R}'$ is validated by σ.

Recall that, for the equational problem $(\exists \vec{x}\,)(\forall j_1, \ldots, j_l)\,\mathcal{Q}$ over the finite universes H_1 and H_2, we used the short-hand notation

$$(\exists \vec{x}\,)(\forall j_1, \ldots, j_l)(\exists \alpha, \beta)(\alpha < \beta) \wedge P(\alpha, \beta, j_\alpha, j_\beta)$$

in Section 4.2, i.e., $(\exists \vec{x}\,)(\forall j_1, \ldots, j_l)\mathcal{Q}$ is satisfiable, iff there exists a substitution τ on the variables $\vec{x}$, such that, for every tuple $\langle j_1, \ldots, j_l \rangle$, there exist two indices α, β with $1 \leq \alpha < \beta \leq l$ for which τ validates $P(\alpha, \beta, j_\alpha, j_\beta)$. In fact, by the definition of the set Ω, we know that, for every tuple $\langle j_1, \ldots, j_l \rangle \in \{1, \ldots, m\}^l$, Ω contains at least one quadruple $\langle \alpha, \beta, \gamma, \delta \rangle$ with $j_\alpha \equiv \gamma$ and $j_\beta \equiv \delta$. Thus, the solution σ of $\mathcal{R}'$ is the desired substitution τ.

Proof of "$\Leftarrow$".
Suppose that $(\exists \vec{x}\,)(\forall j_1, \ldots, j_l)\mathcal{Q}$ is satisfiable. Hence, there exists a substitution σ on the variables $\vec{x}$, such that, for all tuples $\langle j_1, \ldots, j_l \rangle$, there exist indices $\alpha < \beta$ for which $P(\alpha, \beta, j_\alpha, j_\beta)$ is validated by σ. Let Ω denote the set of all quadruples $\langle \alpha, \beta, j_\alpha, j_\beta \rangle$, such that $P(\alpha, \beta, j_\alpha, j_\beta)$ is validated by σ. Analogously to the proof of the "$\Rightarrow$"-direction, Ω has the following property: For every tuple $\langle j_1, \ldots, j_l \rangle \in \{1, \ldots, m\}^l$, Ω contains at least one quadruple $\langle \alpha, \beta, \gamma, \delta \rangle$ with $j_\alpha \equiv \gamma$ and $j_\beta \equiv \delta$.

Moreover, for this Ω, we again consider the equational problem

$$\mathcal{R}' \equiv (\exists \vec{x}\,) \bigwedge_{(\alpha,\beta,\gamma,\delta)\in\Omega} P(\alpha, \beta, \gamma, \delta)$$

(over the finite Herbrand universe H_1) and the problem

$$\mathcal{R} \equiv (\exists \vec{x}\,)(\forall \vec{y}\,) \bigwedge_{(\alpha,\beta,\gamma,\delta)\in\Omega} \langle L^*_\alpha, L^*_\beta \rangle \neq \langle M^*_\gamma, M^*_\delta \rangle$$

(over the infinite Herbrand universe H). By construction, $\mathcal{R}'$ is satisfiable, since the substitution σ on $\vec{x}$ validates every conjunct $P(\alpha, \beta, \gamma, \delta)$ in $\mathcal{R}'$. Thus, analogously to Lemma 13, it can be shown that also $\mathcal{R}$ is satisfiable. In other words, there exists a substitution ϑ on $\vec{x}$, such that, for all H-ground substitutions η on $\vec{y}$, $\langle L^*_\alpha\eta, L^*_\beta\eta \rangle \neq \langle M^*_\gamma\vartheta, M^*_\delta\vartheta \rangle$ holds for every $(\alpha, \beta, \gamma, \delta) \in \Omega$. Equivalently, $\langle L_\alpha, L_\beta \rangle \not\leq_s \langle M_\gamma\vartheta, M_\delta\vartheta \rangle$

holds for every $(\alpha, \beta, \gamma, \delta) \in \Omega$. Thus, by the definition of Ω, the H-subsumption criterion from Lemma 17 is fulfilled. Thus, $C \not\leq_{ss}^{H} D$ indeed holds. $\diamond$

Proof Sketch of Theorem 21.
The equational problem $\mathcal{Q}$ is basically made up of $n * \frac{l*(l-1)}{2} * m^2$ sub-formulas $P(i, \alpha, \beta, \gamma, \delta)$. Each $P(i, \alpha, \beta, \gamma, \delta)$ is obtained from an equational problem $(\exists \vec{x})(\forall \vec{y})\langle L^*_{i\alpha}, L^*_{i\beta}\rangle \neq \langle M^*_\gamma, M^*_\delta\rangle$ whose size is bounded by $O(||C_{\max}|| + ||D||)$. By Theorem 16, we thus get

$$\begin{aligned} |\mathit{Var}(P(i, \alpha, \beta, \gamma, \delta))| &\equiv O\Big((||C_{\max}|| + ||D||) * \mathit{size}(\Sigma)\Big) \\ |\mathit{Var}(\mathcal{Q})| &\equiv O\Big(n * l^2 * m^2 * (||C_{\max}|| + ||D||) * \mathit{size}(\Sigma)\Big). \end{aligned}$$

Hence, for the total size, we get (by considerations analogous to the proof sketch of Theorem 16)

$$||P(i, \alpha, \beta, \gamma, \delta)|| \equiv O\Big((||C_{\max}|| + ||D||) * \mathit{size}(\Sigma) * |\mathit{Var}(\mathcal{Q})|^2\Big)$$

and, therefore,

$$\begin{aligned} ||\mathcal{Q}|| &\equiv O\Big(n * l^2 * m^2 * (||C_{\max}|| + ||D||) * \mathit{size}(\Sigma) * |\mathit{Var}(\mathcal{Q})|^2\Big) \\ &\equiv O\Big(n^3 * l^6 * m^6 * (||C_{\max}|| + ||D||)^3 * \mathit{size}(\Sigma)^3\Big). \end{aligned}$$

Moreover, the size of the universe H_1 has the same bound as $|\mathit{Var}(\mathcal{Q})|$. The bound m on $|H_2|$ is obvious. Finally, it is easy to check that the time required for the whole transformation fits into the bound

$$O\Big(n^3 * l^6 * m^6 * (||C_{\max}|| + ||D||)^3 * \mathit{size}(\Sigma)^3\Big)$$

of $||\mathcal{Q}||$. $\diamond$

Proof of Theorem 25.
In what follows, we assume w.l.o.g. that each propositional variable (in a closed QBF) is bound by exactly one quantifier. Such *cleansed* forms are obtained by renaming of bound variables. Observe that the original QBF and its cleansed forms are logically equivalent.

We now proceed with the proof of Theorem 25. Recall that we have to prove that for each closed equational problem $\mathcal{P}$ over finite Herbrand universes $\mathcal{H} \equiv H_1, \ldots, H_r$, $\mathcal{P}$ is satisfiable iff $\mathcal{T}_\mathcal{H}[\mathcal{P}]$ evaluates to true.

Let us consider equational problems over primitive connectives $\neg$, $\wedge$ as well as existential quantifiers. Obviously, every equational problem is easily translated into such a simpler one by the usual techniques. The proof proceeds by induction on the logical complexity of $\mathcal{P}$, $lc(\mathcal{P})$, which is the number of occurrences of symbols $\neg$, $\wedge$, and $\exists$ in $\mathcal{P}$.

INDUCTION BASE. Assume $lc(\mathcal{P}) \equiv 0$. We deal with closed equational problems over finite sorts. Thus, $\mathcal{P}$ is of form $a = b$ where a and b are elements from some $H \in \mathcal{H}$, or $\mathcal{P}$ is of the form $\top$ or $\bot$, respectively. In the former case, $\mathcal{P}$ is satisfiable iff a and b are syntactically identical. Since, by assumption, we assign different vectors of truth values to $\vec{a} \equiv \langle a_1, \ldots, a_N\rangle$ and $\vec{b} \equiv \langle b_1, \ldots, b_N\rangle$ (with $N \equiv \lceil ld(|H|)\rceil$), whenever a and b are different, it follows that

$$\mathcal{T}_\mathcal{H}[a = b] \equiv (a \Leftrightarrow_H b) \equiv (a_1 \leftrightarrow b_1) \wedge \ldots \wedge (a_N \leftrightarrow b_N)$$

is true, iff a and b are the same element from H. In the latter case, i.e., $\mathcal{P}$ is either $\top$ or $\bot$, $\mathcal{T}_\mathcal{H}[\top] \equiv \top$ or $\mathcal{T}_\mathcal{H}[\bot] \equiv \bot$ by item 1. in Definition 23.

$$\begin{tikzcd}
\mathcal{P} \arrow[r, "\text{remove } \exists x"] \arrow[d, "\mathcal{T}_{\mathcal{H}}[\cdot]"'] & \mathcal{P}_1 \arrow[r, "\sigma \equiv \{x \leftarrow a\}"] \arrow[d, "\mathcal{T}_{\mathcal{H}}[\cdot]"'] & \mathcal{P}_1\sigma \arrow[d, "\mathcal{T}_{\mathcal{H}}[\cdot]"] \\
\mathcal{T}_{\mathcal{H}}[\mathcal{P}] & \mathcal{T}_{\mathcal{H}}[\mathcal{P}_1] \arrow[l, "\text{add } DCA + \exists"'] & \mathcal{T}_{\mathcal{H}}[\mathcal{P}_1\sigma] \arrow[l, "\text{construct } M"']
\end{tikzcd}$$

Figure 3: The steps for the only-if part in case 3 in the proof of Theorem 25.

INDUCTION STEP. Assume $lc(\mathcal{P}) \equiv n > 0$, and let the theorem hold for all closed equational problems $\mathcal{P}'$ over $\mathcal{H}$ with $lc(\mathcal{P}') < n$. Since we deal only with primitive connectives, it is sufficient to consider the following three cases, depending on the principal symbol of $\mathcal{P}$.

CASE 1: Assume that $\mathcal{P} \equiv \mathcal{P}_1 \wedge \mathcal{P}_2$. $\mathcal{P}$ is satisfiable iff both $\mathcal{P}_1$ and $\mathcal{P}_2$ are satisfiable. Since $lc(\mathcal{P}_i) < n$ holds for both $i \in \{1, 2\}$, the QBFs $\mathcal{T}_{\mathcal{H}}[\mathcal{P}_1]$ and $\mathcal{T}_{\mathcal{H}}[\mathcal{P}_2]$ are true by the induction hypothesis. Consequently, $\mathcal{T}_{\mathcal{H}}[\mathcal{P}] \equiv \mathcal{T}_{\mathcal{H}}[\mathcal{P}_1] \wedge \mathcal{T}_{\mathcal{H}}[\mathcal{P}_2]$ evaluates to true as well.

CASE 2: Assume that $\mathcal{P} \equiv \neg\mathcal{P}_1$. $\mathcal{P}$ is satisfiable over $\mathcal{H}$ iff $\mathcal{P}_1$ is not satisfiable over $\mathcal{H}$. Since $lc(\mathcal{P}_1) \equiv n - 1$, the induction hypothesis provides that $\mathcal{P}_1$ is unsatisfiable over $\mathcal{H}$ iff $\mathcal{T}_{\mathcal{H}}[\mathcal{P}_1]$ evaluates to false. But this holds iff $\mathcal{T}_{\mathcal{H}}[\mathcal{P}] \equiv \neg\mathcal{T}_{\mathcal{H}}[\mathcal{P}_1]$ evaluates to true.

CASE 3: Finally, assume $\mathcal{P} \equiv \exists x \mathcal{P}_1$, where x is associated to some $H \in \mathcal{H}$.

For the only-if direction, the relevant transformation steps are depicted in Figure 3. Assume that $\mathcal{P}$ is satisfiable over $\mathcal{H}$. Satisfiability of $\mathcal{P}$ (over $\mathcal{H}$) together with the semantics of first-order logic imply the existence of an $a \in H$ and a substitution $\sigma \equiv \{x \leftarrow a\}$ such that the closed equational problem $\mathcal{P}_1\sigma$ is satisfiable over $\mathcal{H}$. Since $lc(\mathcal{P}_1\sigma) \equiv n - 1$, the induction hypothesis gives us the satisfiability of $\mathcal{T}_{\mathcal{H}}[\mathcal{P}_1\sigma]$.

Before we continue, let us compare the formula trees of $\mathcal{T}_{\mathcal{H}}[\mathcal{P}_1\sigma]$ and $\mathcal{T}_{\mathcal{H}}[\mathcal{P}_1]$. Due to the translation $\mathcal{T}_{\mathcal{H}}[\cdot]$ and the encoding of x by $\langle x_1, \ldots, x_N\rangle$ and a by $\langle a_1, \ldots, a_N\rangle$ (where $N \equiv \lceil \mathrm{ld}(|H|)\rceil$), both formula trees share the same structure but have different labels at some positions. Nodes which are labeled with x_i in the latter tree are labeled with a_i in the former. Let occurrences of x_i and a_i be called *corresponding* if they occur at the same position, whereby x_i occurs in the latter tree and a_i occurs in the former tree.

The latter tree can be reconstructed from the former tree by replacing all occurrences of a_i's in the former tree by x_i's in case the occurrence of a_i corresponds to x_i. Since $a_i \in \{\bot, \top\}$ and $\mathcal{T}_{\mathcal{H}}[\mathcal{P}_1\sigma]$ is true, $\mathcal{T}_{\mathcal{H}}[\mathcal{P}_1]$ is satisfiable under an interpretation $M \subseteq \{x_1, \ldots, x_N\}$ with $x_i \in M$ iff $a_i \equiv \top$, and $x_i \notin M$ otherwise. By definition, M is a model of $DCA_H(x)$ as well. Hence, $DCA_H(x) \wedge \mathcal{T}_{\mathcal{H}}[\mathcal{P}_1]$ is satisfiable and, by the semantics of QBFs, the closed formula

$$\mathcal{T}_{\mathcal{H}}[\mathcal{P}] \equiv \exists x_1 \ldots x_N (DCA_H(x) \wedge \mathcal{T}_{\mathcal{H}}[\mathcal{P}_1])$$

evaluates to true.

For the if-direction, assume that the QBF

$$\mathcal{T}_{\mathcal{H}}[\mathcal{P}] \equiv \exists x_1 \ldots x_N (DCA_H(x) \wedge \mathcal{T}_{\mathcal{H}}[\mathcal{P}_1]) \equiv \exists x_1 \ldots x_N K(x_1, \ldots, x_N)$$

evaluates to true. Satisfiability of $\mathcal{T}_{\mathcal{H}}[\mathcal{P}]$ together with the semantics of QBFs imply that there are truth values $a_1, \ldots, a_N$ such that the closed QBF $K(a_1, \ldots, a_N)$ is true. Then $M \subseteq \{x_1, \ldots, x_N\}$ is a model of $K(x_1, \ldots, x_N)$, where $x_i \in M$ iff $a_i \equiv \top$, and $x_i \notin M$ otherwise. Since M is a model for $DCA_H(x) \wedge \mathcal{T}_{\mathcal{H}}[\mathcal{P}_1]$, $DCA_H(x)$ evaluates to true under M. This implies that there is an element $a \in H$ such that

$\vec{a} \equiv \langle a_1, \ldots, a_N \rangle$. Moreover, $\mathcal{T}_{\mathcal{H}}[\mathcal{P}_1]$ evaluates to true under M, the closed QBF $(\cdots(\mathcal{T}_{\mathcal{H}}[\mathcal{P}_1]^{x_1}_{a_1})\cdots)^{x_N}_{a_N}$ is true and the logical complexity for $\mathcal{P}_1$ and $\mathcal{P}_1\sigma$ is $n-1$, where $\sigma \equiv \{x \leftarrow a\}$. The induction hypothesis yields a closed equational problem $\mathcal{P}_1\sigma$ satisfiable over $\mathcal{H}$. By the semantics of first-order logic, the closed equational problem $\exists x \mathcal{P}_1$ is satisfiable. $\diamond$