ELSEVIER Theoretical Computer Science 290 (2003) 1021–1056

Theoretical Computer Science

www.elsevier.com/locate/tcs

Explicit versus implicit representations of subsets of the Herbrand universe☆

Reinhard Pichler

Technische Universität Wien, Institut für Computersprachen, Favoritenstrasse 9/I/3, A-1040 Wien, Austria

Received 20 November 2000; received in revised form 20 March 2002; accepted 25 June 2002
Communciated by J. Tiuryn

Abstract

In Lassez and Marriott (J. Automat. Reson. 3 (3) (1987) 301–317), explicit and implicit generalizations were studied as representations of subsets of some fixed Herbrand universe H. An *explicit generalization* $E = r_1 \vee \cdots \vee r_l$ represents all ground terms that are instances of at least one of the terms t_i, whereas an *implicit generalization* $I = t/t_1 \vee \cdots \vee t_m$ represents all H-ground instances of t that are not instances of any term t_i. More generally, a disjunction $\mathscr{I} = I_1 \vee \cdots \vee I_n$ of implicit generalizations contains all ground terms that are contained in at least one of the implicit generalizations I_j.

Implicit generalizations have applications to many areas of Computer Science like machine learning, unification, specification of abstract data types, logic programming, functional programming, etc. In these areas, the so-called *finite explicit representability problem* plays an important role, i.e. given a disjunction of implicit generalizations $\mathscr{I} = I_1 \vee \cdots \vee I_n$, does there exist an explicit generalization E, s.t. $\mathscr{I}$ and E are equivalent? We shall prove the coNP-completeness of this decision problem.

Implicit generalizations can be represented as *equational formulae*, i.e., first-order formulae whose only predicate symbol is syntactic equality. Closely related to the finite explicit representability problem is the so-called *negation elimination problem* of equational formulae, i.e. given an arbitrary equational formula $\mathscr{P}$, is $\mathscr{P}$ semantically equivalent to an equational formula without universal quantifiers and negation. In this work we study the negation elimination problem of equational formulae with purely existential quantifier prefix. We prove the coNP-completeness for such formulae in DNF and the Π_2^p-hardness in case of CNF.

Keywords: Implicit generalizations; Negation elimination; Herbrand universe; Complexity

E-mail address: reini@logic.at (R. Pichler).

☆ Preliminary versions of the results in this paper appeared in Refs. [19] and [20].

PII: S0304-3975(02)00583-2

1. Introduction

In [13], explicit and implicit generalizations were studied as representations of sets of ground terms over some fixed Herbrand universe H. An *explicit generalization* $E = r_1 \vee \cdots \vee r_l$ represents all ground terms that are instances of at least one of the terms t_i. On the other hand, an *implicit generalization* $I = t/t_1 \vee \cdots \vee t_m$ represents all H-ground instances of t that are not instances of any term t_i. In the literature, the following two decision problems have received a lot of attention: The *emptiness problem* (i.e. Does a given implicit generalization $I = t/t_1 \vee \cdots \vee t_m$ contain no ground term $s \in H$?) and the *finite explicit representability problem* (i.e. Given an implicit generalization $I = t/t_1 \vee \cdots \vee t_m$, does there exist an explicit generalization $E = r_1 \vee \cdots \vee r_l$, s.t. I and E represent the same set of ground terms in H?).

The usefulness of implicit generalizations comes from their additional expressive power w.r.t. explicit ones. In particular, implicit generalizations allow us to finitely represent certain sets of ground terms which have no finite representation via explicit generalizations. The expressive power of implicit generalizations can be further increased by considering disjunctions of implicit generalizations $\mathscr{I} = I_1 \vee \cdots \vee I_n$, where $\mathscr{I}$ contains all ground terms that are contained in at least one of the implicit generalizations I_j.

The original motivation for dealing with implicit generalizations comes from the area of machine learning, where implicit generalizations can be viewed as a formal basis of learning from counter-examples. In particular, computing explicit generalizations from implicit ones corresponds to learning disjunctive concepts from counter-examples [12,13]. In logic programming, the same kind of problem has to be solved for a constructive definition of negation. In contrast to the negation-as-failure paradigm, this approach allows us the computation of answer substitutions also for *negative* non-ground goals. Likewise, the problem of eliminating ambiguity from functional programs can be easily reduced to the finite explicit representability problem (cf. [5]). Representation formalisms similar to explicit and implicit generalizations (but with a different terminology) are also used in automated model building (cf. [1,4]).

Both the emptiness problem and the finite explicit representability problem have been shown to be coNP-complete in case of a single implicit generalization (cf. [8–10], and [15], respectively). Extending the coNP-membership of the emptiness problem to disjunctions of implicit generalizations is trivial, namely: $\mathscr{I} = I_1 \vee \cdots \vee I_n$ is empty, iff all the disjuncts I_j are empty. On the other hand, the finite explicit representability problem for disjunctions of implicit generalizations is a bit more tricky. In particular, $\mathscr{I} = I_1 \vee \cdots \vee I_n$ may well have a finite explicit representation, even though some of the disjuncts I_j possibly do not. For instance, the implicit generalization $I_1 = f(x, y)/f(x, x)$ over the Herbrand universe H with signature $\Sigma = \{a, f\}$ has no finite explicit representation, while the disjunction $I_1 \vee I_2$ with $I_2 = f(x, x)$ is clearly equivalent to $E = f(x, y)$. In this paper, we present a new algorithm for the finite explicit representability problem, which will allow us to prove the coNP-membership also in case of disjunctions of implicit generalizations.

Equational formulae are first-order formulae over some Herbrand universe H whose only predicate symbol is syntactic equality. By the *negation elimination problem* we mean the problem of deciding whether a given equational formula is semantically equivalent to an equational formula without universal quantifiers and negation. Note that the finite explicit representability problem can be easily transformed into the negation elimination problem, namely: Let $I = t/t_1 \vee \cdots \vee t_m$ be an implicit generalization over H, s.t. $\vec{x}$ denotes the vector of variables in t and $\vec{y}$ denotes the variables in the terms t_i. W.l.o.g. we assume that $\vec{x}$ and $\vec{y}$ are disjoint. Moreover, let z be a fresh variable. Then an H-ground term $z\sigma$ is contained in I, iff σ is a solution of the equational formula

$$\mathscr{P} \equiv (\exists \vec{x})(\forall \vec{y})[z = t \wedge t \neq t_1 \wedge \cdots \wedge t \neq t_n].$$

Moreover, I has an equivalent explicit generalization, iff negation elimination from $\mathscr{P}$ is possible. Apart from the above mentioned applications of implicit generalizations, the negation elimination problem also has some typical applications on its own. For instance, in constrained rewriting, constraints are used to express certain rule application strategies (cf. [21]). Due to the failure of the critical pair lemma, one may eventually have to convert the constraints into equations only, which corresponds to the negation elimination problem.

In this work, we study the negation elimination problem of purely existentially quantified equational formulae. These simple formulae are the target of the transformations given in [2] and [3] for arbitrary equational formulae. Hence, an efficient negation elimination algorithm for this special case is also important for negation elimination from arbitrary equational formulae. We prove the coNP-completeness in case of purely existentially quantified equational formulae in DNF and the Π_2^p-hardness in case of CNF.

This paper is organized as follows: We start off by recalling some basic notions in Section 2. In Section 3, we shall present a new approach to deciding the finite explicit representability problem, which will allow us to prove the coNP-completeness in case of disjunctions of implicit generalizations. In Section 4, we shall develop a new algorithm for the negation elimination problem of purely existentially quantified equational formulae. A comparison with related works will be given in Section 5. Finally, in Section 6, we give a conclusion.

2. Preliminaries

2.1. Equational formulae

An *equational formula* over a Herbrand universe H is a first-order formula with syntactic equality "=" as the only predicate symbol. A disequation $s \neq t$ is a shorthand notation for a negated equation $\neg(s = t)$. The trivially true formula is denoted by $\top$ and the trivially false one by $\bot$. An interpretation is given through a ground substitution σ, which assigns a ground term from H to every free variable of the equational formula. The trivial formula $\top$ evaluates to true in every interpretation. Likewise, $\bot$ always evaluates to false. A single equation $s = t$ is validated by a ground substitution σ, if $s\sigma$ and $t\sigma$ are syntactically identical. The connectives $\wedge$, $\vee$, $\neg$, $\exists$ and $\forall$ are interpreted "as usual". A ground substitution σ which validates an equational formula $\mathscr{P}$ is called a *solution* of $\mathscr{P}$. In order to distinguish between syntactical identity and the semantic equivalence of two equational formulae, we shall use the notation

"≡" and "≈", respectively, i.e. $\mathscr{P} \equiv \mathscr{Q}$ means that the two formulae $\mathscr{P}$ and $\mathscr{Q}$ are *syntactically identical*, while $\mathscr{P} \approx \mathscr{Q}$ means that the two formulae are *semantically equivalent* (i.e. they have the same set of solutions). Moreover, by $\mathscr{P} \leqslant \mathscr{Q}$ we denote that all solutions of $\mathscr{P}$ are also solutions of $\mathscr{Q}$. We shall sometimes use term tuples as a short-hand notation for a conjunction of equations or a disjunction of disequations, respectively, i.e. for term tuples $\vec{s} = (s_1, \dots, s_k)$ and $\vec{t} = (t_1, \dots, t_k)$, we shall abbreviate "$s_1 = t_1 \wedge \cdots \wedge s_k = t_k$" and "$s_1 \neq t_1 \vee \cdots \vee s_k \neq t_k$" to "$\vec{s} = \vec{t}$" and "$\vec{s} \neq \vec{t}$", respectively.

2.2. Common ground instances of terms

A *conjunction of terms* $t_1 \wedge \cdots \wedge t_n$ over some Herbrand universe H represents the set of all ground terms $s \in H$ that are instances of every term t_i. Suppose that the terms t_i are pairwise variable disjoint. Then the set of ground terms contained in such a conjunction corresponds to the H-ground instances of the *most general instance* of the terms $t_1, \dots, t_n$. Note that the most general instance as well as the *most general unifier* are only unique up to variable renaming. However, by abuse of notation, we shall speak about *the* most general instance and *the* most general unifier, when we mean any such term or unifier, respectively. As a short-hand notation, we shall write $mgi(t_1, \dots, t_n)$ and $mgu(t_1, \dots, t_n)$. The only restriction that we impose is that the *mgu* has to be computed without introducing new variables. In [11], such an *mgu* is called *idempotent*.

A *constrained term* over some Herbrand universe H is a pair $[t : X]$ consisting of a term t and an equational formula X, s.t. an H-ground instance $t\sigma$ of t is also an instance of $[t : X]$, iff σ is a solution of X. Note that any term s can also be considered as a constrained term by adding the trivially true formula $\top$ as a constraint, i.e.: s and $[s : \top]$ are equivalent. A *conjunction of constrained terms* $[p_1 : X_1] \wedge \cdots \wedge [p_m : X_m]$ over H represents the set of all ground terms $s \in H$ that are instances of every $[p_i : X_i]$. Now suppose that the constrained terms $[p_i : X_i]$ are pairwise variable disjoint. Then the set of ground terms contained in such a conjunction is equivalent to a single constrained term $[p_1\mu : Z\mu]$, with $Z \equiv X_1 \wedge \cdots \wedge X_m$ and $\mu = mgu(p_1, \dots, p_m)$. Hence, for testing whether such a conjunction of constrained terms is non-empty, we have to check, whether $\mu = mgu(p_1, \dots, p_m)$ exists and whether $Z\mu$ has at least one solution. In this work, we shall only have to deal with constraints X_i that are either a conjunction of disequations or the trivially true constraint $\top$. Thus, also $Z\mu$ is either a conjunction of disequations or the trivially true constraint $\top$. But then $Z\mu$ has at least one solution, iff $Z\mu$ contains no trivial disequation of the form $t \neq t$ (for a proof see [2, Lemma 2]).

Recall from [11] that unification can be used to simplify conjunctions of equations and disjunctions of disequations, namely: Let $\vec{s}$ and $\vec{t}$ be k-tuples of terms and let $\vartheta = \{z_1 \leftarrow r_1, \dots, z_n \leftarrow r_n\}$ be the *mgu* of $\vec{s}$ and $\vec{t}$. Then $\vec{s} = \vec{t}$ is equivalent to $z_1 = r_1 \wedge \cdots \wedge z_n = r_n$. Likewise, $\vec{s} \neq \vec{t}$ is equivalent to $z_1 \neq r_1 \vee \cdots \vee z_n \neq r_n$. As a short-hand notation, we shall write $Equ(\vartheta)$ and $Disequ(\vartheta)$ for these simplified equations and disequations, respectively.

2.3. Linear terms

Throughout this paper, we only consider the case of an infinite Herbrand universe (i.e. a universe with a finite signature that contains at least one proper function symbol and one constant symbol), since otherwise both the finite explicit representability problem and the negation elimination problem are trivial. The property of terms which is crucial for the decision problems studied here is the so-called "linearity". We say that a term t (or a tuple $\vec{t}$ of terms) is *linear*, iff every variable in t (or in $\vec{t}$, respectively) occurs at most once. Otherwise t is *non-linear*. Let $\vec{x}$ denote the vector of variables that occur in some term t. Then we call an instance $t\vartheta$ of t a *linear instance of* t (or *linear w.r.t.* t), iff $\vec{x}\vartheta$ is linear. Otherwise $t\vartheta$ is called *non-linear w.r.t.* t. In general, the *range of a substitution* denotes a set of terms. However, in this paper, we usually have to deal with substitutions in the context of instances $t\vartheta$ of another term t. It is therefore more convenient to consider the range of a substitution as a *vector of terms*, namely: Let $\vec{x}$ denote the vector of variables in t. Then we refer to the vector $\vec{x}\vartheta$ as the *range* of ϑ, which we denote by $rg(\vartheta)$. In particular, we can then say that an instance $t\vartheta$ of t is non-linear, iff there exists a multiply occurring variable in $rg(\vartheta)$. If we want to refer to the set of variables in the range of ϑ without paying attention to multiple occurrences, we write $Var(rg(\vartheta))$. By $dom(\vartheta)$, we denote the domain of ϑ, i.e., the set of variables x for which $x \neq x\vartheta$ holds. Finally, if we want to restrict the domain of a substitution ϑ to some set V of variables, then we write $\vartheta|_V$.

2.4. The complement of terms

The implicit generalization $I = t/t\vartheta$ can be considered as the complement of $t\vartheta$ w.r.t. t, i.e. I contains all H-ground instances of t that are not instances of $t\vartheta$. By the *domain closure axiom*, every ground term in the Herbrand universe H with signature Σ is an instance of the disjunction $\bigvee_{f\in\Sigma} f(x_1,\dots,x_{\alpha(f)})$, where the x_i's are pairwise distinct variables and $\alpha(f) \geqslant 0$ denotes the arity of the function symbol f (constants are considered as function symbols of arity 0). In [13], this fact is used to provide a representation of the complement of $t\vartheta$ w.r.t. t, if $t\vartheta$ is a linear instance of t. However, for our purposes, we need a representation of the complement $t\vartheta$ w.r.t. t also in case of a non-linear instance $t\vartheta$. In [6,7], such a representation is given via constrained terms in the following way: Consider the tree representation of the range of ϑ, "deviate" from this representation at some node and close all other branches of the tree as early as possible with new, pairwise distinct variables. Depending on the label of a node, this deviation is done as follows: If a node is labelled by a function symbol, then this node has to be labelled by a different function symbol from Σ. If a node is labelled by a variable which occurs nowhere else, then no deviation at all is possible at this node. Finally, the case of variables with multiple occurrences is treated as follows: Suppose that some variable x occurs at $k \geqslant 2$ different positions $p_1,\dots,p_k$ in the range of ϑ. Then the representation of the complement of $t\vartheta$ w.r.t. t contains the following constrained terms $t_1,\dots,t_{k-1}$: let $1 \leqslant j \leqslant k-1$. Then, in order to construct t_j, we replace the occurrences of x at the positions p_j and p_{j+1} by fresh variables z_j and z_{j+1}, respectively, and we add the constraint $z_j \neq z_{j+1}$.

For instance, let $t = f(x_1, g(x_2))$ be a term over the Herbrand universe H with signature $\Sigma = \{f, g, a\}$ and let $\vartheta = \{x_1 \leftarrow f(y_1, y_2), x_2 \leftarrow g(y_1)\}$. Then we get the following representation of the complement of $t\vartheta$ w.r.t. t:

$$\begin{aligned}P = {} & f(x_1, g(x_2))\{x_1 \leftarrow a, x_2 \leftarrow v\}, f(x_1, g(x_2))\{x_1 \leftarrow g(z), x_2 \leftarrow v\},\\ & f(x_1, g(x_2))\{x_1 \leftarrow v, x_2 \leftarrow a\}, f(x_1, g(x_2))\{x_1 \leftarrow v, x_2 \leftarrow f(z_1, z_2)\},\\ & [f(x_1, g(x_2))\{x_1 \leftarrow f(z_1, v), x_2 \leftarrow g(z_2)\} : z_1 \neq z_2]\}\\ = {} & \{f(a, g(v)), f(g(z), g(v)), f(v, g(a)), f(v, g(f(z_1, z_2))),\\ & [f(f(z_1, v), g(g(z_2))) : z_1 \neq z_2]\}.\end{aligned}$$

For our purposes, only the following properties of this representation of the complement are needed (for any details and for a proof, see [7]):

Theorem 2.1 (Complement of a term). *Let t be a term over the Herbrand universe H and let $t\vartheta$ be an instance of t. Then there exists a set of constrained terms $P = \{[p_1 : X_1], \ldots, [p_n : X_n]\}$ with the following properties*:

(1) *$t/t\vartheta = [p_1 : X_1] \vee \cdots \vee [p_n : X_n]$, i.e. every H-ground instance of the complement of $t\vartheta$ w.r.t. t is an H-ground instance of some $[p_i : X_i]$ and vice versa.*
(2) *For every $i \in \{1, \ldots, n\}$, p_i is a linear instance of t and X_i is either the trivially true formula $\top$ or a (quantifier-free) disequation.*
(3) *The size of every constrained term $[p_i : X_i] \in P$ as well as the number n of such terms is linearly bounded by the number of positions in $t\vartheta$.*

Actually, Property (3) can even be strengthened as follows: The size of every constrained term $[p_i : X_i] \in P$ is linearly bounded by the size of the dag representation (directed acyclic graph) of $t\vartheta$, cf. [18]. This is important, if $t\vartheta$ itself is the result of a unification step.

The equational constraints are only needed in order to finitely express the complement of a *non-linear* instance $t\vartheta$ of t. Hence, the two approaches from [13] and [6] are quite similar, when only linear instances are considered. Obviously, the representation of the complement of a term from Theorem 2.1 can be easily extended to implicit generalizations, namely: Let $I = t/t_1 \vee \cdots \vee t_m$ be an implicit generalization and let $P = \{[p_1 : X_1], \ldots, [p_n : X_n]\}$ be the complement of t w.r.t. some variable x, then $P \cup \{t_1, \ldots, t_m\}$ is a representation of the complement of I. By writing the terms t_i as constrained terms $[t_i : \top]$, we end up again with a set of constrained terms.

Remark. Note that we are using the word "complement" with two different meanings, namely: On the one hand, the complement of a term $t\vartheta$ w.r.t. t refers to the set of all ground terms contained in t but not in $t\vartheta$. In this case, the "complement" is simply a synonym for an implicit generalization $t/t\vartheta$. On the other hand, the complement of a term t (or of an implicit generalization I) as such consists of all terms in the Herbrand universe H that are not contained in t (or in I, respectively). By abuse of notation, we shall sometimes omit the extension "w.r.t. t". This slight inaccuracy is somehow

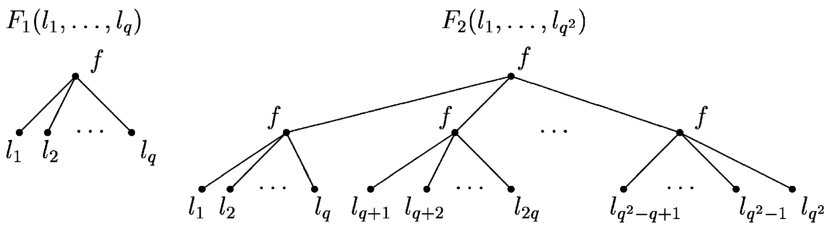

Fig. 1. $F_1(l_1,\dots,l_q)$ and $F_2(l_1,\dots,l_{q^2})$.

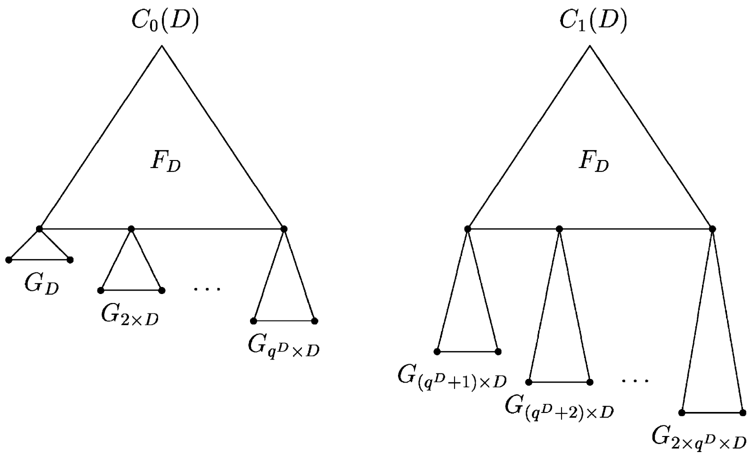

Fig. 2. $C_0(D)$ and $C_1(D)$.

justified by the way we are working with the complement, namely: When considering an implicit generalization of the form $I = t/t\vartheta_1 \vee \cdots \vee t\vartheta_m$, we shall try to restrict the ground instances of some term $t\vartheta_i$ to those terms which are also contained in the "complement of $t\vartheta_j$" for some $j \neq i$. Of course, in this case, it makes no difference whether we really mean the "complement of $t\vartheta_j$" or the "complement of $t\vartheta_j$ w.r.t. t", since all instances of $t\vartheta_i$ are instances of t anyway.

2.5. The terms $C_0(D), C_1(D), \dots$

In [13], the construction of certain terms $C_0(D), C_1(D), \dots$ plays a crucial role. These terms, which will also be made use of in our proofs, are defined as follows: Let f be a function symbol with arity q and let a be a constant. Then $F_D(l_1,\dots,l_{q^D})$ denotes the term whose tree representation has the label f at all nodes down to depth $D-1$ and whose nodes at depth D are labelled with l_1 through l_{q^D}. The first two such terms are depicted in Fig. 1.

G_D corresponds to the special case where $l_i = a$ holds for every i, i.e. $G_D = F_D(a,\dots,a)$, e.g., for a binary function symbol f, $G_2 = f(f(a,a), f(a,a))$ holds. Then $C_i(D)$ is defined as $C_i(D) = F_D(G_{(q^D\times i+1)\times D}, G_{(q^D\times i+2)\times D}, \dots, G_{(i+1)\times q^D\times D})$ for $i \geqslant 0$. The terms $C_0(D)$ and $C_1(D)$ are sketched in Fig. 2.

The usefulness of these terms comes from the following property: *If u is a subterm of $C_i(D)$, s.t. the root of u is at a depth smaller than D, then u occurs only once in $C_i(D)$ and u does not occur at all in any other term $C_j(D)$ with $i \neq j$*. From this, another property follows easily, namely: *If there exists an index i, s.t. $C_i(D)$ is an instance of a term t of depth smaller than D, then t contains no multiple variable occurrences and, moreover, $C_j(D)$ is also an instance of t for every index j*. For any details, refer to [13].

3. Explicit representation of implicit generalizations

Note that we can assume w.l.o.g. that all terms t_i on the right-hand side of an implicit generalization $I = t/t_1 \vee \cdots \vee t_m$, are instances of t, since otherwise we would replace t_i by the most general instance $mgi(t, t_i)$. If all the terms t_i are linear w.r.t. t, then an equivalent explicit representation can be immediately obtained via the complement representation of linear instances from [13] (cf. also Section 2.4). The following result is taken from [16, Corollary 3.5].

Lemma 3.1. *Let $I = t/t\vartheta_1 \vee \cdots \vee t\vartheta_m$ be an implicit generalization over some Herbrand universe H, s.t. all terms $t\vartheta_i$ on the right-hand side of I are linear instances of t. Moreover, for every $i \in \{1,\dots,m\}$, let $P_i = \{p_{i1},\dots,p_{iM_i}\}$ be a representation of the complement of $t\vartheta_i$ w.r.t. t and let the p_{ij}'s be pairwise variable disjoint. Then I is equivalent to the explicit generalization*

$$E = \bigvee_{\alpha_1=1}^{M_1} \cdots \bigvee_{\alpha_m=1}^{M_m} mgi(p_{1\alpha_1},\dots,p_{m\alpha_m}).$$

In [13], it is shown that an implicit generalization $I = t/t\vartheta_1 \vee \cdots \vee t\vartheta_m$ has an equivalent explicit representation, iff every non-linear instance $t\vartheta_i$ of t can be replaced by a finite number of linear instances of t. A non-linear instance $t\vartheta_i$ of t which cannot be replaced by a finite number of linear ones will be referred to as "*essentially non-linear*" in this paper. This notion is illustrated in the following example:

Example 3.2. Let $I = f(x, y)/f(x,x)$ be an implicit generalization over the Herbrand universe H with signature $\Sigma = \{f, a\}$. It is shown in [13], that I has no equivalent explicit representation. In particular, the non-linear term $f(x,x)$ cannot be replaced by finitely many linear ones. Hence, $f(x,x)$ is *essentially non-linear*. On the other hand, consider the implicit generalization $J = f(y_1, y_2)/[f(x,x) \vee f(f(x_1,x_2),x_3)]$. Then $J' = f(y_1, y_2)/[f(a,a) \vee f(f(x_1,x_2),x_3)]$ is equivalent to J, i.e. the term $f(x,x)$ is *inessentially non-linear*, since it can be replaced by the linear term $f(a,a)$.

In this section, we present a new approach to deciding the finite explicit representability problem by formalizing the notion of *essential non-linearity*. In Section 3.1, we outline the basic ideas by considering single implicit generalizations. These ideas will then be extended to disjunctions of implicit generalizations in Section 3.2. Finally, in

Section 3.3, we shall slightly modify these ideas in order to prove the coNP-membership of the finite explicit representability problem of disjunctions of implicit generalizations.

3.1. The basic algorithm

We start our study of the finite explicit representability problem by considering a single implicit generalization. In Definition 3.3 below, we define a transformation rule T which allows us to replace a non-linear instance on the right-hand side of an implicit generalization by a disjunction of terms.

Definition 3.3 (Transformation rule T). Let $I = t/t\vartheta_1 \vee \cdots \vee t\vartheta_m$ be an implicit generalization over some Herbrand universe H, s.t. $t\vartheta_1$ is a non-linear instance of t. Moreover, let $P_i = \{[p_{i1} : X_{i1}], \ldots, [p_{iM_i} : X_{iM_i}]\}$ represent the complement of $t\vartheta_i$ w.r.t. t and let all terms in $\{t\vartheta_1\} \cup \{p_{i\alpha_i} \mid 2 \leqslant i \leqslant m \text{ and } 1 \leqslant \alpha_i \leqslant M_i\}$ be pairwise variable disjoint.

Then $t\vartheta_1$ may be replaced in I by the disjunction $\bigvee_{\mu \in M} t\vartheta_1\mu$, where M is the following set of substitutions:

$$M = \{\mu \mid \exists\alpha_2 \ldots \exists\alpha_m, \text{ s.t. } \mu = mgu(t\vartheta_1, p_{2\alpha_2}, \ldots, p_{m\alpha_m}) \text{ and}$$
$$(X_{2\alpha_2} \wedge \cdots \wedge X_{m\alpha_m})\mu \text{ contains no trivial disequation}\}.$$

The idea of the transformation rule T is that we may restrict the term $t\vartheta_i$ on the right-hand side of I to those instances, which are in the complement of the remaining terms $t\vartheta_j$ with $j \neq i$. The correctness of this step is proven below. Intuitively, it is due to the equality $A - [B \cup C] = A - [(B - C) \cup C]$, which holds for arbitrary sets A, B and C.

Theorem 3.4 (Correctness of T). *Let $I = t/t\vartheta_1 \vee \cdots \vee t\vartheta_m$ be an implicit generalization over some Herbrand universe H and let*

$$I' = t \Bigg/ \left(\bigvee_{\mu \in M} t\vartheta_1\mu \right) \vee t\vartheta_2 \vee \cdots \vee t\vartheta_m$$

be the result of applying the transformation rule T from Definition 3.3 to I. Then I and I' are equivalent.

Proof. All terms $t\vartheta_1\mu$ with $\mu \in M$ are instances of $t\vartheta_1$. Hence, $I \subseteq I'$ trivially holds. We only have to prove the opposite subset relation. Suppose conversely that there exists a term $u \in (I' - I)$. Then u is not in $t\vartheta_2 \vee \cdots \vee t\vartheta_m$. Moreover, u is an instance of $t\vartheta_1$ but not of any term $t\vartheta_1\mu$. For every i, $P_i = \{[p_{i1} : X_{i1}], \ldots, [p_{iM_i} : X_{iM_i}]\}$ completely covers the complement of $t\vartheta_i$ w.r.t. t. Hence, there exist indices $\alpha_2, \ldots, \alpha_m$, s.t. $u \in t\vartheta_1 \wedge [p_{2\alpha_2} : X_{2\alpha_2}] \wedge \cdots \wedge [p_{m\alpha_m} : X_{m\alpha_m}]$. By the considerations from Section 2.2, this conjunction is equivalent to $[t\vartheta_1\mu : Z\mu]$ with $\mu = mgu(t\vartheta_1, p_{2\alpha_2}, \ldots, p_{m\alpha_m})$ and $Z \equiv X_{2\alpha_2} \wedge \cdots \wedge X_{m\alpha_m}$. Moreover, $[t\vartheta_1\mu : Z\mu] \subseteq t\vartheta_1\mu$ holds. But then u is not an instance of I', which contradicts our original assumption. $\square$

In order to arrive at a decision procedure for the explicit representability problem, it remains to show that a given implicit generalization I has no finite explicit representation, if the transformation rule T does not allow us to remove all non-linearities from I.

Theorem 3.5 (Essential non-linearity). *Let $I = t/t\vartheta_1 \vee \cdots \vee t\vartheta_m$ be an implicit generalization over some Herbrand universe H, s.t. $t\vartheta_1$ is a non-linear instance of t and let*

$$I' = t \Bigg/ \left(\bigvee_{\mu \in M} t\vartheta_1\mu \right) \vee t\vartheta_2 \vee \cdots \vee t\vartheta_m$$

be the result of applying the transformation rule T from Definition 3.3 to I. Moreover, suppose that there exists a substitution $\mu \in M$, s.t. $t\vartheta_1\mu$ is a non-linear instance of t. Then I has no finite explicit representation.

Proof. Let $P_i = \{[p_{i1} : X_{i1}], \ldots, [p_{iM_i} : X_{iM_i}]\}$ represent the complement of $t\vartheta_i$ w.r.t. t and let all terms in $\{t\vartheta_1\} \cup \{p_{i\alpha_i} \mid 2 \leqslant i \leqslant m$ and $1 \leqslant \alpha_i \leqslant M_i\}$ be pairwise variable disjoint. By assumption, there exists a term $t\vartheta_1\mu$ that is a non-linear instance of t. In other words, there exist indices $\alpha_2, \ldots, \alpha_m$ with $1 \leqslant \alpha_i \leqslant M_i$ for all i, s.t. the following conditions hold:

(1) $\mu = mgu(t\vartheta_1, p_{2\alpha_2}, \ldots, p_{m\alpha_m})$ exists.
(2) $Z\mu$ with $Z \equiv X_{2\alpha_2} \wedge \cdots \wedge X_{m\alpha_m}$ contains no trivial disequation.
(3) There exists a multiply occurring variable y in the range of ϑ_1, s.t. $y\mu$ is a non-ground term, i.e. $y\mu = f[z]$, where $f[z]$ is a term containing some variable z.

The proof is indirect, i.e., we assume that I is equivalent to the explicit generalization $E = r_1 \vee \cdots \vee r_M$. Recall that the variable y occurs at least twice in the range of ϑ_1. Now let us modify ϑ_1 to ϑ_1' by replacing one occurrence of y in the range of ϑ_1 by a fresh variable y' and extend μ to μ', s.t. $y'\mu' = f[z']$ for another fresh variable z'. Then we derive a contradiction as follows:

1. *$t\vartheta_1'\mu'$ is an instance of $p_{2\alpha_2}\pi$ with $\pi = mgu(p_{2\alpha_2}, \ldots, p_{m\alpha_m})$*: By Theorem 2.1, all terms $p_{i\alpha_i}$ are linear w.r.t. t and, therefore, also $p_{2\alpha_2}\pi$ is linear w.r.t. t. Hence, $p_{2\alpha_2}\pi = t\varphi$ for some substitution φ which has no multiple variable occurrences in its range. Moreover, $t\vartheta_1\mu$ is an instance of $[p_{2\alpha_2}\pi : Z\pi]$ with $Z \equiv X_{2\alpha_2} \wedge \cdots \wedge X_{m\alpha_m}$. Thus, there exists a substitution λ, s.t. $\vartheta_1\mu = \varphi\lambda$. Now consider the occurrence of $f[z]$ in $\vartheta_1\mu$ which is replaced by $f[z']$ in $\vartheta_1'\mu'$. There must be some variable x in the range of φ which is instantiated by λ to some term containing this occurrence of the variable z. But then, since all variables in the range of φ occur only once, λ can be modified to λ' s.t. this occurrence of z in $x\lambda$ is replaced by z' and λ' coincides with λ everywhere else. Thus $\varphi\lambda' = \vartheta_1'\mu'$ with $\lambda = \lambda' \circ \{z' \leftarrow z\}$ holds. Hence, in particular, $t\vartheta_1'\mu'$ is an instance of $p_{2\alpha_2}\pi$ with $p_{2\alpha_2}\pi\lambda' = t\vartheta_1'\mu'$ and $p_{2\alpha_2}\pi\lambda' \circ \{z' \leftarrow z\} = t\vartheta_1\mu$. Moreover, since the equational problems $X_{i\alpha_i}$ only contain variables from $p_{i\alpha_i}$, the equivalence $Z\mu \equiv Z\pi\lambda' \circ \{z' \leftarrow z\}$ also holds.

2. *Construction of two ground terms s' and s, s.t. s' is in I and s is outside*: We shall now make use of the terms $C_0(D)$, $C_1(D), \dots$ recalled in Section 2.5. By assumption, $Z\mu \equiv Z\pi\lambda' \circ \{z' \leftarrow z\}$ contains no trivial disequation. Hence, $Z\pi\lambda'$ contains no trivial disequation either and, therefore, there exists a solution σ' of $Z\pi\lambda'$, i.e. $Z\pi\lambda'\sigma'$ is yet another conjunction of disequations with no trivial disequation. Now let D be an integer, s.t. D is greater than the depth of any term occurring in $Z\pi\lambda'\sigma'$ and greater than the depth of $mgi(t\vartheta_1'\mu', r_\gamma)$ for all terms r_γ from the explicit representation E of I. Then we can modify σ' to τ', s.t. both substitutions coincide on all variables except for z' and z, where we define $z\tau' = C_0(D)$ and $z'\tau' = C_1(D)$. By the definition of $C_0(D)$ and $C_1(D)$, no trivial disequation can be introduced into $Z\pi\lambda'\tau'$ by this transformation from $Z\pi\lambda'\sigma'$. Let the term s' be defined as $s' = p_{2\alpha_2}\pi\lambda'\tau'$. Then, on the one hand, s' is an instance of $[p_{2\alpha_2}\pi : Z\pi]$ and, on the other hand, s' is not an instance of $t\vartheta_1\mu = p_{2\alpha_2}\pi\lambda' \circ \{z' \leftarrow z\}$, since τ' assigns different values to z and z'. However, if we define a new substitution τ, s.t. τ instantiates both variables z and z' to $C_0(D)$ and τ coincides with σ' everywhere else, then $s = p_{2\alpha_2}\pi\lambda'\tau$ is an instance of $t\vartheta_1\mu = p_{2\alpha_2}\pi\lambda' \circ \{z' \leftarrow z\}$ and, in particular, of $t\vartheta_1$. Thus, s' is an instance of $I = t/t\vartheta_1 \vee \cdots \vee t\vartheta_m$ whereas s is not.

3. *If s' is an instance of r_γ, then s is also an instance of r_γ*: By assumption, I is equivalent to $E = r_1 \vee \cdots \vee r_M$. Hence, there exists a term r_γ, s.t. s' is an instance of r_γ and, therefore, also of $mgi(t\vartheta_1'\mu', r_\gamma)$. Thus, there exist substitutions ρ' and η', s.t. $t\vartheta_1'\mu'\rho' = mgi(t\vartheta_1'\mu', r_\gamma)$ and $t\vartheta_1'\mu'\rho'\eta' = s'$. By construction, $C_1(D)$ is an instance of $z'\rho'$ and the term depth of $z'\rho'$ is smaller than D. Moreover, all subterms of $C_1(D)$ with root at depth smaller than D occur nowhere else in the range of $\vartheta_1'\mu'\rho'\eta'$. Hence, the variables in $z'\rho'$ occur nowhere else in the range of ρ' and, by the properties of the terms $C_i(D)$ recalled in Section 2.5, $C_0(D)$ is also an instance of $z'\rho'$. We can thus modify η' to η, s.t. $z'\rho'\eta = C_0(D)$ holds and η coincides with η' on all variables not occurring in $z'\rho'$. Thus, $t\vartheta_1'\mu'\rho'\eta = s$ holds. But then, s is an instance of $t\vartheta_1'\mu'\rho' = mgi(t\vartheta_1'\mu', r_\gamma)$ and, therefore, also of r_γ, which contradicts the assumption that r_γ is a disjunct from the explicit representation E of I. $\square$

It is now clear, what our new algorithm for deciding the finite explicit representability problem of a single implicit generalization looks like: Let $I = t/t\vartheta_1 \vee \cdots \vee t\vartheta_m$ be an implicit generalization and suppose that some term $t\vartheta_i$ is a non-linear instance of t. Of course, the order of the terms on the right-hand side of an implicit generalization does not matter. Hence, we may assume w.l.o.g. that $i = 1$. Now we can apply our transformation rule T from Definition 3.3. If at least one of the terms $t\vartheta_1\mu$ thus introduced is non-linear w.r.t. t, then we may conclude by Theorem 3.5 that I has no finite explicit representation. Otherwise, the number of non-linear instances of t on the right-hand side of I is strictly decreased. After repeating this step at most m times, we either detect failure according to Theorem 3.5 or we manage to replace all non-linear instances of t from the right-hand side of I by linear ones. In the latter case, it is an easy task to compute an equivalent explicit representation via Lemma 3.1. This algorithm is put to work in the following example:

Example 3.6. Let $I = t/t\vartheta_1 \vee \cdots \vee t\vartheta_4$ be an implicit generalization over the Herbrand universe H with signature $\Sigma = \{a, f, g\}$, s.t. $t = f(f(x_1, x_2), x_3))$ and the ϑ_i's are defined as follows:

$$\vartheta_1 = \{x_1 \leftarrow f(y_{11}, y_{12}), x_2 \leftarrow y_{13}, x_3 \leftarrow y_{13}\},$$

$$\vartheta_2 = \{x_1 \leftarrow y_{21}, x_2 \leftarrow y_{21}, x_3 \leftarrow g(y_{22})\},$$

$$\vartheta_3 = \{x_1 \leftarrow y_{31}, x_2 \leftarrow f(y_{32}, y_{33}), x_3 \leftarrow y_{34}\},$$

$$\vartheta_4 = \{x_1 \leftarrow y_{41}, x_2 \leftarrow g(y_{42}), x_3 \leftarrow y_{43}\}.$$

Note that any instance of t is of the form $t\sigma$ with $\sigma = \{x_1 \leftarrow s_1, x_2 \leftarrow s_2, x_3 \leftarrow s_3\}$. In order to keep the notation simple, we denote such an instance by $t(s_1, s_2, s_3)$. Then the complement of the terms $t\vartheta_2$, $t\vartheta_3$ and $t\vartheta_4$ w.r.t. t is represented by the following sets P_2, P_3 and P_4, respectively:

$$P_2 = \{[t(z_{21}, z_{22}, a) : \top], [t(z_{21}, z_{22}, f(z_{23}, z_{24})) : \top], [t(z_{21}, z_{22}, z_{23}) : z_{21} \neq z_{22}]\},$$

$$P_3 = \{[t(z_{31}, a, z_{32}) : \top], [t(z_{31}, g(z_{32}), z_{33}) : \top]\},$$

$$P_4 = \{[t(z_{41}, a, z_{42}) : \top], [t(z_{41}, f(z_{42}, z_{43}), z_{44}) : \top]\}.$$

In order to apply the transformation rule T from Definition 3.3, we have to compute the set M of certain unifiers. As a short-hand notation, we shall write $\mu_{(\alpha_2, \alpha_3, \alpha_4)}$ to denote the *mgu* of $t\vartheta_1$ with the terms $p_{2\alpha_2}$, $p_{3\alpha_3}$ and $p_{4\alpha_4}$ from the sets P_2, P_3 and P_4. Then M consists of a single element, namely $\mu_{(1,1,1)} = \mu_{(3,1,1)} = \{y_{13} \leftarrow a\}$. Hence, I may be transformed into the equivalent generalization $I' = t/t\vartheta_2 \vee t\vartheta_3 \vee t\vartheta_4 \vee t(f(y_{11}, y_{12}), a, a)$.

We already know the representations $P_2' = P_3$ and $P_3' = P_4$ of the complement of $t\vartheta_3$ and $t\vartheta_4$, respectively. Hence, there is only the complement P_4' of $t(f(y_{11}, y_{12}), a, a)$ w.r.t. t missing, namely:

$$P_4' = [t(a, z_{51}, z_{52}) : \top], [t(g(z_{51}), z_{52}, z_{53}) : \top], [t(z_{51}, f(z_{52}, z_{53}), z_{54}) : \top],$$

$$[t(z_{51}, g(z_{52}), z_{53}) : \top], [t(z_{51}, z_{52}, f(z_{53}, z_{54})) : \top],$$

$$[t(z_{51}, z_{52}, g(z_{53})) : \top]\}.$$

Analogously to the short-hand notation above, we write $\mu_{(\alpha_2, \alpha_3, \alpha_4)}$ to denote the *mgu* of $t\vartheta_2$ with the terms $p_{2\alpha_2}$, $p_{3\alpha_3}$ and $p_{4\alpha_4}$ from the sets P_2', $P_{3'}$ and P_4'. Then the set M (for I') again consists of a single element, namely: $\mu_{(1,1,1)} = \mu_{(1,1,6)} = \{y_{21} \leftarrow a\}$. The non-linear instance $t\vartheta_2 = t(y_{21}, y_{21}, g(y_{22}))$ in I' may therefore be replaced by the linear instance $t\vartheta_2\mu_{(1,1,1)} = t(a, a, g(y_{22}))$. We have thus transformed I into $I'' = t/t(y_{31}, f(y_{32}, y_{33}), y_{34}) \vee t(y_{41}, g(y_{42}), y_{43}) \vee t(f(y_{11}, y_{12}), a, a) \vee t(a, a, g(y_{22}))$, which has only linear instances of t on the right-hand side. Hence, I has a finite explicit representation E. In order to actually compute E, we need a representation of the complement of all terms on the right-hand side. $P_1'' = P_3$, $P_2'' = P_4$ and $P_3'' = P_4'$ have already been computed. The complement of $t(a, a, g(y_{22}))$ w.r.t. t can be represented by

$$P_4'' = \{t(g(z_{61}), z_{62}, z_{63}), t(f(z_{61}, z_{62}), z_{63}), t(z_{61}, g(z_{62}), z_{63}),$$

$$t(z_{61}, f(z_{62}, z_{63}), z_{64}), t(z_{61}, z_{62}, a), t(z_{61}, z_{62}, f(z_{63}, z_{64}))\}.$$

Hence, the explicit representation E of I is of the form

$$
\begin{aligned}
E &= \left(\bigvee_{p_1 \in P_1''} p_1\right) \wedge \left(\bigvee_{p_2 \in P_2''} p_2\right) \wedge \left(\bigvee_{p_3 \in P_3''} p_3\right) \wedge \left(\bigvee_{p_4 \in P_4''} p_4\right)\\
&= \bigvee_{p_1 \in P_1''} \bigvee_{p_2 \in P_2''} \bigvee_{p_3 \in P_3''} \bigvee_{p_4 \in P_4''} (p_1 \wedge p_2 \wedge p_3 \wedge p_4).
\end{aligned}
$$

By computing all possible conjunctions $p_1 \wedge p_2 \wedge p_3 \wedge p_4$ and deleting those terms which are a proper instance of another conjunction, we get

$$
\begin{aligned}
E &= t(a,a,a) \vee t(f(y_1,y_2),a,g(y_3)) \vee t(y_1,a,f(y_2,y_3)) \vee t(g(y_1),a,y_2)\\
&= f(f(a,a),a)) \vee f(f(f(y_1,y_2),a),g(y_3))) \vee f(f(y_1,a),f(y_2,y_3))\\
&\quad \vee f(f(g(y_1),a),y_2)).
\end{aligned}
$$

3.2. Disjunctions of implicit generalizations

We shall now extend the notion of *essential non-linearity* to disjunctions of implicit generalizations. In particular, we have to extend the transformation rule T from Definition 3.3.

Definition 3.7 (Transformation rule T'). Let $\mathscr{I} = I_1 \vee \cdots \vee I_n$ with $I_j = t_j/t_j\vartheta_{j1} \vee \cdots \vee t_j\vartheta_{jm_j}$ for $j \in \{1,\ldots,n\}$ be a disjunction of implicit generalizations over some Herbrand universe H and suppose that the term $t_1\vartheta_{11}$ is non-linear w.r.t. t_1. Moreover, let $P_i = \{[p_{i1} : X_{i1}],\ldots,[p_{iM_i} : X_{iM_i}]\}$ represent the complement of $t_1\vartheta_{1i}$ w.r.t. t_1 and let $Q_j = \{[q_{j1} : Y_{j1}],\ldots,[q_{jN_j} : Y_{jN_j}]\}$ represent the complement of I_j. Finally, let all terms in $\{t_1\vartheta_{11}\} \cup \{p_{i\alpha_i} \mid 2 \leqslant i \leqslant m_1 \text{ and } 1 \leqslant \alpha_i \leqslant M_i\} \cup \{q_{j\beta_j} \mid 2 \leqslant j \leqslant n \text{ and } 1 \leqslant \beta_j \leqslant N_j\}$ be pairwise variable disjoint. By $\vec{y}$ we denote the vector of variables with multiple occurrences in the range of ϑ_{11}.

Then $t_1\vartheta_{11}$ may be replaced in I_1 by the disjunction $\bigvee_{\mu \in M} t_1\vartheta_{11}\mu$, where M is the following set of substitutions:

$$
\begin{aligned}
M = \{\mu \mid\ & \exists\alpha_2 \cdots \exists\alpha_{m_1} \exists\beta_2 \cdots \exists\beta_n, \text{ s.t.}\\
& v = mgu(t_1\vartheta_{11}, p_{2\alpha_2},\ldots, p_{m\alpha_{m_1}}, q_{2\beta_2},\ldots\ldots, q_{n\beta_n}),\\
& (X_{2\alpha_2} \wedge \cdots \wedge X_{m_1\alpha_{m_1}} \wedge Y_{2\beta_2} \wedge \cdots \wedge Y_{n\beta_n})v \text{ contains no trivial}\\
& \text{disequation and } \mu = v|_{\vec{y}}\}.
\end{aligned}
$$

The idea of the transformation rule T' is twofold: First, we may restrict the term $t_1\vartheta_{11}$ to those instances, which are in the complement of the remaining terms $t_1\vartheta_{1j}$ on the right-hand side of I_1 with $j \neq 1$. Then we may further restrict the term $t_1\vartheta_{11}$

to those instances, which are in the complement of the other implicit generalizations I_k with $k \neq 1$. The correctness of these two replacement steps is due to the equalities $A - [B \cup C] = A - [(B - C) \cup C]$ and $[A - B] \cup C = [A - (B - C)] \cup C$, respectively, which hold for arbitrary sets A, B and C. The correctness of T' is formally proven below.

Theorem 3.8 (Correctness of T'). *Let* $\mathscr{I} = I_1 \vee \cdots \vee I_n$ *with* $I_j = t_j / t_j\vartheta_{j1} \vee \cdots \vee t_j\vartheta_{jm_j}$ *for* $j \in \{1, \dots, n\}$ *be a disjunction of implicit generalizations over some Herbrand universe* H. *Moreover, let* $\mathscr{I}' = I_1' \vee I_2 \vee \cdots \vee I_n$ *with*

$$I_1' = t_1 \bigg/ \left(\bigvee_{\mu \in M} t_1\vartheta_{11}\mu \right) \vee t_1\vartheta_{12} \vee \cdots \vee t\vartheta_{1m_1}$$

be the result of applying the transformation rule T' *of Definition* 3.7 *to* $\mathscr{I}$. *Then* $\mathscr{I}$ *and* $\mathscr{I}'$ *are equivalent.*

Proof. All terms $t_1\vartheta_{11}\mu$ with $\mu \in M$ are instances of $t_1\vartheta_{11}$. Hence, $I_1 \subseteq I_1'$ and, therefore, also $\mathscr{I} \subseteq \mathscr{I}'$ trivially holds. So we only have to prove the opposite subset relation: Let $u \in \mathscr{I}'$. If $u \in I_2 \vee \cdots \vee I_n$, then u is of course also contained in $\mathscr{I}$. Thus, the only interesting case to consider is that $u \in I_1'$ and $u \notin I_2 \vee \cdots \vee I_n$. Hence, in particular, u is in the complement of every $t_1\vartheta_{1i}$ with $2 \leqslant i \leqslant m_1$ and in the complement of every I_j with $2 \leqslant j \leqslant n$. But for every i, $P_i = \{[p_{i1} : X_{i1}], \dots, [p_{iM_i} : X_{iM_i}]\}$ completely covers the complement of $t_1\vartheta_{1i}$ w.r.t. t_1. Likewise, for every j, $Q_j = \{[q_{j1} : Y_{j1}], \dots, [q_{jN_j} : Y_{jN_j}]\}$ completely covers the complement of I_j. Hence, there exist indices $\alpha_2, \dots, \alpha_{m_1}$ and $\beta_2, \dots, \beta_n$, s.t. $u \in \bigwedge_{i=2}^{m_1} [p_{i\alpha_i} : X_{i\alpha_i}] \wedge \bigwedge_{j=2}^{n} [q_{j\beta_j} : Y_{j\beta_j}]$. Then, analogously to the proof of Theorem 3.4, it can be shown that if u is not an instance of any term $t_1\vartheta_{11}\mu$ on the right-hand side of I_1', then u is not an instance of $t_1\vartheta_{11}$ either. In other words, if u is an instance of I_1' but not of any I_i with $i \geqslant 2$, then u is actually an instance of I_1. $\square$

Analogously to Theorem 3.5, we have to show that a disjunction of implicit generalizations $\mathscr{I}$ has no finite explicit representation, if the transformation rule T' does not remove all non-linearities.

Theorem 3.9 (Essential non-linearity). *Let* $\mathscr{I} = I_1 \vee \cdots \vee I_n$ *with* $I_j = t_j / t_j\vartheta_{j1} \vee \cdots \vee t_j\vartheta_{jm_j}$ *for* $j \in \{1, \dots, n\}$ *be a disjunction of implicit generalizations, s.t.* $t_1\vartheta_{11}$ *is non-linear w.r.t.* t_1 *and let* $\mathscr{I}' = I_1' \vee I_2 \vee \cdots \vee I_n$ *with*

$$I_1' = t_1 \bigg/ \left(\bigvee_{\mu \in M} t_1\vartheta_{11}\mu \right) \vee t_1\vartheta_{12} \vee \cdots \vee t\vartheta_{1m_1}$$

be the result of applying the transformation rule T' *of Definition* 3.7 *to* $\mathscr{I}$. *Moreover, suppose that there exists a substitution* $\mu \in M$, *s.t.* $t_1\vartheta_{11}\mu$ *is a non-linear instance of* t. *Then* $\mathscr{I}$ *has no finite explicit representation.*

Proof (Rough sketch). Similarly to the proof of Theorem 3.5, we can use the terms $C_0(D)$ and $C_1(D)$ for constructing ground terms s and s', s.t. s' is inside $\mathscr{I}$ and s

is outside. Now suppose that $\mathscr{I}$ has an explicit representation $r_1 \vee \cdots \vee r_l$. Then, in particular, s' is an instance of some r_j. Analogously to the proof of Theorem 3.5, we can derive a contradiction by showing that then also s is an instance of r_j. The details are omitted here, since, in Theorem 3.12, we shall prove a slightly stronger result than Theorem 3.9 anyway. □

The construction of a new algorithm for the finite explicit representability problem of disjunctions of implicit generalizations is now straightforward, namely: Let $\mathscr{I} = I_1 \vee \cdots \vee I_n$ with $I_j = t_j / t_j\vartheta_{j1} \vee \cdots \vee t_j\vartheta_{jm_j}$ for $j \in \{1,\dots,n\}$ be a disjunction of implicit generalizations. Suppose that there exists a term $t_j\vartheta_{jk}$ that is a non-linear instance of t_j. Of course, we may arrange the disjuncts of $\mathscr{I}$ and the terms on the right-hand side of each implicit generalization in such a way that $j=1$ and $k=1$ hold. Then we can either replace $t_1\vartheta_{11}$ by a disjunction of linear instances of t_1 via the transformation rule T' or we know by Theorem 3.9 that $\mathscr{I}$ has no equivalent explicit representation. This algorithm clearly terminates, since the total number of terms on the right-hand side of the implicit generalizations that are non-linear w.r.t. the corresponding left-hand side is strictly decreased, whenever we apply the transformation rule T'. Hence, eventually, we either detect failure by Theorem 3.9 or we manage to transform $\mathscr{I}$ into an equivalent disjunction of implicit generalizations $\mathscr{I}' = I_1' \vee \cdots \vee I_n'$, s.t. *every term* $t_j\vartheta_{jk}'$ on the right-hand side of *every implicit generalization* I_j' is *linear* w.r.t. the term t_j on the corresponding left-hand side. In the latter case, $\mathscr{I}$ clearly has an equivalent explicit representation E. Moreover, E is simply the disjunction of the explicit representations of the implicit generalizations involved, which in turn can be easily computed via Lemma 3.1. The following example will help to illustrate these ideas:

Example 3.10. Let $\mathscr{I} = I_1 \vee I_2$ be a disjunction of implicit generalizations over the Herbrand universe H with signature $\Sigma = \{a, f\}$, where I_1 and I_2 are defined as follows:

$$I_1 = f(f(x_1,x_2), f(x_3,x_4))/[f(f(y_1,y_1), f(y_2,a))$$

$$\vee f(f(y_1,y_1), f(a,y_2)) \vee f(f(y_1,y_2), f(f(y_3,y_4),y_5))],$$

$$I_2 = f(x_1, f(x_2,x_3))/f(y_1, f(y_2,a)).$$

We first apply the transformation rule T' from Definition 3.7 to the non-linear instance $f(f(y_1,y_1), f(y_2,a))$ of $f(f(x_1,x_2), f(x_3,x_4))$ in I_1. Analogously to Example 3.6, we write $t(s_1,s_2,s_3,s_4)$ to denote an instance $t\sigma$ of $t = f(f(x_1,x_2), f(x_3,x_4))$ with $\sigma = \{x_1 \leftarrow s_1, x_2 \leftarrow s_2, x_3 \leftarrow s_3, x_4 \leftarrow s_4\}$. Then the complement representations P_{12}, P_{13}, and Q_2 have the following form:

$$P_{12} = \{[t(z_{11},z_{12},z_{13},z_{14}) : z_{11} \neq z_{12}],\ t(z_{21},z_{22},f(z_{23},z_{24}),z_{25})\},$$

$$P_{13} = \{t(z_{31},z_{32},a,z_{33})\},$$

$$Q_2 = \{a, f(z_{41},a), f(z_{41}, f(z_{42},a))\}.$$

In order to apply T', we have to compute certain unifiers of $t(y_1, y_1, y_2, a)$ with the terms in P_{12}, P_{13}, and Q_2. As a short-hand notation, we shall write $v_{(\alpha_2,\alpha_3,\beta_2)}$ to denote the *mgu* of $t(y_1, y_1, y_2, a)$ with the terms $p_{2\alpha_2}$, $p_{3\alpha_3}$ and $q_{2\beta_2}$. Then $v_{(1,1,3)} = \{y_2 \leftarrow a, z_{11} \leftarrow y_1, z_{12} \leftarrow y_1, z_{13} \leftarrow a, \ldots\}$ is the only element in M. Applying this substitution to the constraint $z_{11} \neq z_{12}$ yields the trivially false disequation $(z_{11} \neq z_{12})v_{(1,1,3)} \equiv (y_1 \neq y_1)$. Hence, T' allows us to delete the term $t(y_1, y_1, y_2, a)$ from the right-hand side of I_1. We thus get $I_1' = f(f(x_1,x_2), f(x_3,x_4)) / f(f(y_1, y_1), f(a, y_2)) \vee f(f(y_1, y_2), f(f(y_3, y_4), y_5))]$.

Now we want to apply T' to $t_1\vartheta_{11}' = f(f(y_1, y_1), f(a, y_2))$. The complement representations $P_{12}' = P_{13}$ and Q_2 have already been computed above. Again, we use the short-hand notation $v_{(\alpha_2,\beta_2)}$ for the *mgu* of $f(f(y_1, y_1), f(a, y_2))$ with the terms $p_{2\alpha_2}$ in P_{12}' and $q_{2\beta_2}$ in Q_2. The only such substitution is $v_{(1,3)} = \{y_2 \leftarrow a, z_{31} \leftarrow y_1, z_{32} \leftarrow y_1, \ldots\}$. There exist no constraints to which $v_{(1,3)}$ has to be applied. Moreover, the only multiply occurring variable in the range of ϑ_{11}' is y_1 and $y_1 v_{(1,3)} = y_1$. Hence, the non-linear instance $t_1\vartheta_{11}'$ of t_1 remains unchanged, when we apply the transformation rule T' to it. Thus, by Theorem 3.9, $\mathscr{I}$ has no finite explicit representation.

3.3. *coNP-completeness*

Note that the transformation rules T and T' introduced in the previous sections may lead to hyper-exponentiality in the worst case. This can be seen as follows: A disjunction of implicit generalizations is a set represented as $\mathscr{I} = I_1 \cup \cdots \cup I_n$, where each I_i is basically given in the form $I_i = S_i - (S_{i1} \cup \cdots \cup S_{im_i})$. We write S^c to denote the complement of a set S. Then the transformation rule T' allows us to replace $\mathscr{I}$ by $\mathscr{I}' = I_1' \cup I_2 \cup \cdots \cup I_n$ with $I_1' = S_1 - (S_{11}' \cup S_{12} \cup \cdots \cup S_{1m_1})$ and $S_{11}' = S_{11} \cap S_{12}^c \cap \cdots \cap S_{1m_1}^c \cap I_2^c \cap \cdots \cap I_n^c$. Unfortunately, in our case of implicit generalizations, the set S_{11}' is represented by a disjunction of the form $\bigvee_{\mu \in M} t_1\vartheta_{11}\mu$, where $|M|$ may be exponentially big. Now if we apply the rule T' also to S_{12}, then we have to reduce S_{12} to the complement of S_{11}'. Hence, we end up with a disjunction of the form $\bigvee_{v \in N} t_1\vartheta_{12}v$, where $|N|$ is, in general, exponentially big w.r.t. $|M|$. Of course, the situation will get increasingly bad with every further application of the rule T'.

The goal of this section is to construct a much more efficient algorithm, which never reduces a set $S_{\alpha\beta}$ w.r.t. sets $S'\gamma\delta$ and I_ε', which are themselves the result of such a reduction step. Instead, we do this reduction w.r.t. the original sets $S\gamma\delta$ and I_ε only. To this end, we briefly recall the two ideas which the transformation rule T' is based upon:

First, let I_1 and I_2 be arbitrary sets of the form $I_1 = (S_1 - T_1)$ and $I_2 = (S_2 - T_2)$. Then the equality $I_1 \cup I_2 = [S_1 - (T_1 \cap I_2^c)] \cup I_2$ holds. In fact, even the following equality holds:

$$I_1 \cup I_2 = [S_1 - (T_1 \cap I_2^c)] \cup [S_2 - (T_2 \cap I_1^c)].$$

In other words, after having transformed I_1 into I_1' by reducing T_1 to the complement of I_2, we may also transform I_2 into I_2' by reducing T_2 to the complement of I_1 rather than to the complement of I_1'.

Now let J be a set of the form $J = S - (S_1 \cup S_2)$. Then the equality $J = S - [(S_1 \cap S_2^c) \cup S_2]$ clearly holds. On the other hand, in general, we have

$$J \neq S - [(S_1 \cap S_2^c) \cup (S_2 \cap S_1^c)].$$

To see this, just consider the case where $S = S_1 = S_2$ holds. In other words, after having transformed S_1 into $S_1' = (S_1 \cap S_2^c)$, we may not reduce S_2 to the complement of the original set S_1. Instead, we are only allowed to reduce S_2 w.r.t. S_1'.

In Fig. 3 we give a new algorithm which will ultimately allow us to prove the coNP-membership of the finite explicit representability problem of disjunctions of implicit generalizations. The key to this algorithm is the equality

$$S - [S_1 \cup \dots \cup S_n] = S - [S_1 \cup (S_2 \cap S_1^c) \cup (S_3 \cap S_1^c \cap S_2^c) \cup \cdots \cup (S_n \cap S_1^c \cap \dots \cap S_{n-1}^c)$$

which holds for any sets $S, S_1, \dots, S_n$. The procedure STEPWISE_REDUCTION takes a Herbrand universe H and a disjunction of implicit generalizations $\mathscr{I} = I_1 \vee \dots \vee I_n$ with $I_i = t_i / t_i\vartheta_{i1} \vee \dots \vee t_i\vartheta_{im_i}$ over H as an input and returns an n-tuple $(R_1, \dots, R_n)$ of term sets, s.t. for all i, all terms in R_i are linear instances of t_i and $\mathscr{I}$ is equivalent to $\mathscr{I}' = I_1' \vee \dots \vee I_n'$ with $I_i' = t_i / \bigvee_{r \in R_i} r$.

For the termination of the procedure STEPWISE_REDUCTION, we only have to check the while-loop in Step 2.2. Actually, whenever this loop is executed, we either halt with failure or we replace K by the strictly smaller set $K - \{k\}$. Hence, the termination of this loop and, therefore, of the whole procedure is obvious. In the Theorems 3.11 and 3.12, we prove the correctness and completeness of this procedure.

Theorem 3.11 (Correctness). *Let $\mathscr{I} = I_1 \vee \dots \vee I_n$ be a disjunction of implicit generalizations over some Herbrand universe H and suppose that, on input H and $\mathscr{I}$, the procedure* STEPWISE_REDUCTION *returns the tuple $(R_1, \dots, R_n)$ of sets of terms.*

Then $\mathscr{I}$ is equivalent to $\mathscr{I}' = I_1' \vee \dots \vee I_n'$ with $I_i' = t_i / \bigvee_{r \in \mathscr{R}_i} r$.

Proof. By the construction of $\mathscr{R}_i$, every term $r \in \mathscr{R}_i$ is an instance of some term $t_i\vartheta_{ij}$. Hence, $I_i \subseteq I_i'$ and, therefore, also $\mathscr{I} \subseteq \mathscr{I}'$ clearly holds. In order to prove also the opposite subset relation, we choose an arbitrary term $s \in \mathscr{I}'$ and show that $s \in \mathscr{I}$ also holds. W.l.o.g. we may assume that s is an instance of I_1'. Moreover, the terms on the right-hand side of I_1 can be arranged in such a way that the terms $t_1\vartheta_{ik}$ are non-linear instances of t_1 for all $k \in \{1, \dots, \lambda\}$ and linear instances of t_1 in case of $k \in \{\lambda+1, \dots, m_1\}$. By $s \in I_1'$, we know that s is an instance of t_1. Now if s is an instance of $I_2 \vee \dots \vee I_n$ then, of course, $s \in \mathscr{I}$ holds and we are done. So suppose that s is in the complement of $I_2, \dots, I_n$, i.e. there exist indices $\beta_2, \dots, \beta_n$, s.t. $s \in \bigwedge_{j=2}^{n} [q_{j\beta_j} : Y_{j\beta_j}]$. Note that by the initialization of $\mathscr{R}_1$ in Step 2.1, all linear instances $t_1\vartheta_{1(\lambda+1)}, \dots, t_1\vartheta_{1m_1}$ of t_1 from the right-hand side of I_1 also appear on the right-hand side of I_1'. By the condition $s \in I_1'$ we thus know that s is in the complement of the terms $t_1\vartheta_{1(\lambda+1)}, \dots, t_1\vartheta_{1m_1}$, i.e. thus, there exist indices $\alpha_{\lambda+1}, \dots, \alpha_{m_1}$, s.t. $s \in \bigwedge_{j=\lambda+1}^{m_1} [p_{(1j),\alpha_j} : X_{(1j),\alpha_j}]$.

Now we have a look at the nested loops over all possible values of i, $(\beta_1, \dots, \beta_{i-1}, \beta_{i+1}, \dots, \beta_n)$, and $(\alpha_1, \dots, \alpha_{m_i})$. We consider the following combination of values: $i = 1$,

STEPWISE REDUCTION VIA THE ORIGINAL TERMS
input: H and $\mathcal{I} = I_1 \vee \ldots \vee I_n$ with $I_i = t_i / t_i\vartheta_{i1} \vee \ldots \vee t_i\vartheta_{im_i}$
output: n-tuple $(R_1, \ldots, R_n)$ of sets of terms

begin

/* Step 1: compute the complement representations P_{ij} and Q_k */
  for all i and j, let $P_{ij} = \{[p_{(ij),1} : X_{(ij),1}], \ldots, [p_{(ij),M_{ij}} : X_{(ij),M_{ij}}]\}$
    represent the complement of $t_i\vartheta_{ij}$ w.r.t. t_i;
  for all k, let $Q_k = \{[q_{k1} : Y_{k1}], \ldots, [q_{kN_k} : Y_{kN_k}]\}$
    represent the complement of I_k;
  let the terms in $[\bigcup_{i=1}^{n} \bigcup_{j=1}^{m_i} \{t_i\vartheta_{ij}\}] \cup [\bigcup_{i=1}^{n} \bigcup_{j=1}^{m_i} \bigcup_{\alpha=1}^{M_{ij}} \{p_{(ij),\alpha}\}] \cup$
    $[\bigcup_{k=1}^{n} \bigcup_{\beta=1}^{N_k} \{q_{k\beta}\}]$ be pairwise variable disjoint.

/* Step 2: compute the sets $R_1, \ldots, R_n$ of terms */
  for $i := 1$ **to** n **do begin**
  /* Step 2.1: deal with the linear terms on the right-hand side of I_i */
    let $\lambda \in \{0, \ldots, m_i\}$, s.t. $\forall k > \lambda$, $t_i\vartheta_{ik}$ is a *linear* instance of t_i
      and $\forall k \leq \lambda$, $t_i\vartheta_{ik}$ is a *non-linear* one;
    $R_i := \{t_i\vartheta_{i(\lambda+1)}, \ldots, t_i\vartheta_{im_i}\}$;
    $K := \{1, \ldots, \lambda\}$;

  /* Step 2.2: deal with the non-linear terms on the right-hand side of I_i */
    for all $(\beta_1, .., \beta_{i-1}, \beta_{i+1}, .., \beta_n) \in \{1, .., N_1\} \times .. \times \{1, .., N_n\}$ **do begin**
      for all $(\alpha_1, .., \alpha_{m_i}) \in \{1, .., M_{i1}\} \times \ldots \times \{1, .., M_{im_i}\}$ **do begin**
        while $K \neq \emptyset$ **do begin**
          if $\forall k \in K$ the following conditions hold:
            1. $\nu_k = mgu(\{t_i\vartheta_{ik}\} \cup \{p_{(ij),\alpha_j} \mid j \in (\{1, \ldots, m_i\} - K)\}$
              $\cup \{q_{j\beta_j} \mid j \in \{1, \ldots, i-1, i+1, \ldots, n\}\})$ exists.
            2. $Z\nu_k$ contains no trivial disequation with
              $Z \equiv [\bigwedge_{j \in (\{1,\ldots,m_i\}-K)} X_{(ij),\alpha_j}] \wedge [\bigwedge_{j \in \{1,\ldots,i-1,i+1,\ldots,n\}} Y_{j\beta_j}]$.
            3. There exists a multiply occurring variable y_k
              in the range of ϑ_{ik}, s.t. $y_k\nu_k$ is a non-ground term.
          then halt with failure;
          fi;

          let $k := \min(\{\gamma \in K \mid$ one of the Conditions 1 - 3 is violated$\})$;
          $K := K - \{k\}$;
          if only Condition 3 is violated for k **then**
            let $\vec{y} =$ multiply occurring variables in the range of ϑ_{ik};
            $R_i := R_i \cup \{\vartheta_{ik}\mu\}$ with $\mu = \nu_k|_{\vec{y}}$;
          fi;
        end; /* while $K \neq \emptyset$ */
      end; /* for all $(\alpha_1, \ldots)$ */
    end; /* for all $(\beta_1, \ldots)$ */
  end; /* for i */
  return $(R_1, \ldots, R_n)$;
end.

Fig. 3. Procedure STEPWISE_REDUCTION.

$(\beta_2,\dots,\beta_n)$ is chosen exactly as described above, i.e. $s\in\bigwedge_{j=2}^{n}[q_{j\beta_j}:Y_{j\beta_j}]$. Likewise, the values of $(\alpha_{\lambda+1},\dots,\alpha_{m_1})$ are the ones mentioned above, s.t. $s\in\bigwedge_{j=\lambda+1}^{m_1}[p_{(1j),\alpha_j}:X_{(1j),\alpha_j}]$. Finally, $(\alpha_1,\dots,\alpha_\lambda)$ can be chosen arbitrarily, e.g., $\alpha_1=\cdots=\alpha_\lambda=1$. When the while-loop is entered for the first time, then $K=\{1,\dots,\lambda\}$ holds by Step 2.1 of the procedure STEPWISE_REDUCTION.

Actually, if $K=\emptyset$, then $\lambda=0$ and all terms on the right-hand side of I_1 are linear instances of t_1. Hence, in this case, s is indeed an instance of I_1 since we are assuming that s is an instance of t_1 and that s is in the complement of $t_1\vartheta_{1(\lambda+1)},\dots,t_1\vartheta_{1m_1}$.

So let $K\neq\emptyset$. By assumption, the procedure STEPWISE_REDUCTION does not halt with failure. Hence, the three conditions in the while-loop do not hold for all $k\in K$. Now let k denote the minimum in K, for which at least one such condition is violated. We claim that then s is not an instance of $t_1\vartheta_{1k}$. Suppose on the contrary that s is an instance of $t_1\vartheta_{1k}$. Then s is also contained in

$$t_1\vartheta_{1k}\wedge\bigwedge_{j=\lambda+1}^{m_1}[p_{(1j),\alpha_j}:X_{(1j),\alpha_j}]\wedge\bigwedge_{j=2}^{n}[q_{j\beta_j}:Y_{j\beta_j}].$$

Hence, s is an instance of $t_1\vartheta_{1k}\mu_k$, where μ_k is the restriction of the substitution

$$v_k=mgu(\{t_1\vartheta_{1k}\}\cup\{p_{(1j),\alpha_j}\mid\lambda+1\leqslant j\leqslant m_1\}\cup\{q_{j\beta_j}\mid 2\leqslant j\leqslant n\})$$

to the variables $\vec{y}$ with multiple occurrences in the range of ϑ_{1k}. However, $t_1\vartheta_{1k}\mu_k$ with $\mu_k=v_k|_{\vec{y}}$ is added to $\mathscr{R}_1$ in Step 2.2 of our algorithm and, therefore, $t_1\vartheta_{1k}\mu_k$ occurs on the right-hand side of I_1'. But then s is not an instance of I_1', which is a contradiction.

Thus s is in the complement of $t_1\vartheta_{1k}$ and, therefore, there exists an index $\alpha_k\in\{1,\dots,M_{1k}\}$, s.t. $s\in[p_{(1k),\alpha_k}:X_{(1k),\alpha_k}]$. As far as the nested loops over all possible values of i, $(\beta_1,\dots,\beta_{i-1},\beta_{i+1},\dots,\beta_n)$, and $(\alpha_1,\dots,\alpha_{m_i})$ are concerned, we consider now the following combination of values: $i=1$, $(\beta_2,\dots,\beta_n)$, and $(\alpha_{\lambda+1},\dots,\alpha_{m_1})$ are chosen as before. Moreover, α_k is the index with $s\in[p_{(1k),\alpha_k}:X_{(1k),\alpha_k}]$ that we have just determined. The values of the remaining α_j's can again be chosen arbitrarily, e.g., $\alpha_1=\cdots=\alpha_{k-1}=\alpha_{k+1}=\cdots=\alpha_\lambda=1$. When the while-loop is entered for the first time, then $K=\{1,\dots,\lambda\}$ holds by Step 2.1 of the procedure STEPWISE_REDUCTION. By the above considerations, K will be modified to $K'=K-\{k\}$ in the first execution of the while-loop. But then we can repeat the same argument as above also for K'. Let k' denote the minimum in K' for which at least one of the three conditions in the while-loop is violated. We claim that then s is not an instance of $t_1\vartheta_{1k'}$. For suppose on the contrary that s is an instance of $t_1\vartheta_{1k'}$. Then s is also in

$$t_1\vartheta_{1k'}\wedge[p_{(1k),\alpha_k}:X_{(1k),\alpha_k}]\wedge\bigwedge_{j=\lambda+1}^{m_1}[p_{(1j),\alpha_j}:X_{(1j),\alpha_j}]\wedge\bigwedge_{j=2}^{n}[q_{j\beta_j}:Y_{j\beta_j}].$$

Hence, s is an instance of $t_1\vartheta_{1k'}\mu_{k'}$, where $\mu_{k'}$ is defined as the restriction of

$$v_{k'}=mgu(\{t_1\vartheta_{1k'}\}\cup\{p_{(1k),\alpha_k}\}\cup\{p_{(1j),\alpha_j}\mid\lambda+1\leqslant j\leqslant m_1\}$$
$$\cup\{q_{j\beta_j}|2\leqslant j\leqslant n\})$$

to the variables with multiple occurrences in the range of $\vartheta_{1k'}$. This is again impossible, since $t_1\vartheta_{1k'}\mu_{k'}$ is added to $\mathscr{R}_1$ in Step 2.2 of our algorithm and, therefore, $t_1\vartheta_{1k'}\mu_{k'}$ occurs on the right-hand side of I_1'. But then s is not an instance of I_1', which is a contradiction.

By iterating this argument at most λ-times, we can show that s is not an instance of any term $t_1\vartheta_{1j}$ with $j \in \{1,\dots,\lambda\}$. Moreover, recall that we are considering the case where s is an instance of t_1 and s is in the complement of $t_1\vartheta_{1(\lambda+1)},\dots,t_1\vartheta_{1m_1}$. Hence, s is indeed an instance of I_1 and, therefore, also of $\mathscr{I}$. □

Theorem 3.12 (Completeness). *Let $\mathscr{I} = I_1 \vee \cdots \vee I_n$ be a disjunction of implicit generalizations over some Herbrand universe H and suppose that, on input H and $\mathscr{I}$, the procedure* STEPWISE_REDUCTION *halts with failure.*

Then $\mathscr{I}$ does not have a finite explicit representation.

Proof. Let the sets $P_{ij} = \{[p_{(ij),1} : X_{(ij),1}],\dots,[p_{(ij),M_{ij}} : X_{(ij),M_{ij}}]\}$ and $Q_k = \{[q_{k1} : Y_{k1}],\dots,[q_{kN_k} : Y_{kN_k}]\}$ be the complement representations according to Step 1 in procedure STEPWISE_REDUCTION. Moreover, w.l.o.g., suppose that there exists a $\lambda \in \{0,\dots,m_i\}$, s.t. $\forall k \leqslant \lambda$, $t_i\vartheta_{ik}$ is a linear instance of t_i and $\forall k > \lambda$, $t_i\vartheta_{ik}$ is a non-linear one. Then, by the failure of this procedure in Step 2.2, we know that

(1) $\exists i \in \{1,\dots,n\}$
(2) $\exists K \subseteq \{1,\dots,\lambda_i\}$ with $K \neq \emptyset$
(3) $\exists(\beta_1,\dots,\beta_{i-1},\beta_{i+1},\dots,\beta_n) \in \{1,\dots,N_1\} \times \cdots \times \{1,\dots,N_n\}$
(4) $\exists(\alpha_1,\dots,\alpha_{m_i}) \in \{1,\dots,M_{i1}\} \times \cdots \times \{1,\dots,M_{im_i}\}$

s.t. the following conditions hold for all $k \in K$:

(1) $v_k = mgu(\{t_i\vartheta_{ik}\} \cup \{p_{(ij),\alpha_j} \mid j \in (\{1,\dots,m_i\} - K)\} \cup \{q_{j\beta_j} \mid j \in \{1,\dots,i-1,i+1,\dots,n\}\})$ exists.
(2) Zv_k with $Z \equiv [\bigwedge_{j\in(\{1,\dots,m_i\}-K)} X_{(ij),\alpha_j}] \wedge [\bigwedge_{j\in\{1,\dots,i-1,i+1,\dots,n\}} Y_{j\beta_j}]$ contains no trivial disequation.
(3) There exists a multiply occurring variable y_k in the range of ϑ_{ik}, s.t. $y_k v_k$ is a non-ground term.

We may assume w.l.o.g. that the implicit generalizations in the disjunction $\mathscr{I}$ and the terms on the right-hand side of each implicit generalization have been arranged in such a way, that $i = 1$ and $K = \{1,\dots,\kappa\}$ hold for some κ with $1 \leqslant \kappa \leqslant \lambda_1$. Moreover, of all the sets P_{ij}, only the P_{1j}'s will play a role in the sequel. Hence, we may simplify our notation by writing $P_{1j} = \{[p_{j1} : X_{j1}],\dots,[p_{jM_j} : X_{jM_j}]\}$ for the complement of $t_1\vartheta_{1j}$ w.r.t. t_1. Then the conditions for the failure of the procedure can be rephrased as follows:

For all $k \in \{1,\dots,\kappa\}$, the following three conditions hold:

(1) $v_k = mgu(\{t_1\vartheta_{1k}\} \cup \{p_{j\alpha_j} \mid \kappa+1 \leqslant j \leqslant m_1\} \cup \{q_{j\beta_j} \mid 2 \leqslant j \leqslant n\})$ exists.
(2) Zv_k with $Z \equiv [\bigwedge_{j=\kappa+1}^{m_1} X_{j\alpha_j}] \wedge [\bigwedge_{j=2}^{n} Y_{j\beta_j}]$ contains no trivial disequation.
(3) There exists a multiply occurring variable y_k in the range of ϑ_{1k}, s.t. $y_k v_k$ is a non-ground term.

Similarly to Theorem 3.5, we are now going to show that $\mathcal{I}$ has no finite explicit representation. For every $k \in \{1,\dots,\kappa\}$, let $y_k v_k = f_k[z_k]$, where $f_k[z_k]$ is some term containing the variable z_k. Then we can again modify every substitution ϑ_{1k} to ϑ'_{1k} by replacing one occurrence of y_k in the range of ϑ_{1k} by a fresh variable y'_k and extend v_k to v'_k, s.t. $y'_k v'_k = f_k[z'_k]$ for another fresh variable z'_k. Now suppose that $\mathcal{I}$ is equivalent to the explicit generalization $E = r_1 \vee \cdots \vee r_l$. Then we derive a contradiction in the following way:

1. *Every* $t_k \vartheta'_{1k} v'_k$ *with* $k \in \{1,\dots,\kappa\}$ *is an instance of* $p_{m_1\alpha_{m_1}}\pi$, where π is defined as $\pi = mgu(p_{(\kappa+1)\alpha_{(\kappa+1)}},\dots,p_{m_1\alpha_{m_1}})$. This proof goes exactly like the first part of the proof of Theorem 3.5. In particular, there exist substitutions φ_k and φ'_k with $\varphi_k = \varphi'_k \circ \{z'_k \leftarrow z_k\}$, s.t. the relations $t_1\vartheta'_{1k}v'_k = p_{m_1\alpha_{m_1}}\pi\varphi'_k$, $t_1\vartheta_{1k}v_k = p_{m_1\alpha_{m_1}}\pi\varphi'_k \circ \{z'_k \leftarrow z_k\}$ and $Zv_k \equiv Z\pi\varphi'_k \circ \{z'_k \leftarrow z_k\}$ hold.

2. *Construction of* s_1 *and* s'_1: Analogously to the second part of the proof of Theorem 3.5, we can construct substitutions τ'_1 and τ_1 from a solution σ'_1 of $Z\pi\varphi'_1$, s.t. $z_1\tau'_1 = C_0(D_1)$, $z'_1\tau'_1 = C_1(D_1)$ and $z_1\tau_1 = z'_1\tau_1 = C_0(D_1)$ hold, where D_1 is greater than the depth of any term occurring in $Z\pi\varphi'_1\sigma'_1$, any term $p_{m_1\alpha_{m_1}}\pi\varphi_k$ with $k \in \{1,\dots,\kappa\}$ and, finally, greater than the depth of $mgi(t_1\vartheta'_{11}v'_1, r_\gamma)$ for all terms r_γ from the explicit representation E of $\mathcal{I}$. Then $s_1 = p_{m_1\alpha_{m_1}}\pi\varphi'_1\tau_1$ is an instance of $t_1\vartheta_{11}$ but not of any implicit generalization $I_2,\dots,I_n$. On the other hand, $s'_1 = p_{m_1\alpha_{m_1}}\pi\varphi'_1\tau'_1$ is an instance of $t_1/t_1\vartheta_{11} \vee t_1\vartheta_{1(\kappa+1)} \vee t_1\vartheta_{(\kappa+2)} \vee \cdots \vee t_1\vartheta_{1m_1}$.

3. *Iteration of the construction of* s_k *and* s'_k *for* $k \in \{2,\dots,\kappa\}$: If s'_1 is not an instance of any term $t_1\vartheta_{1k}$ with $k \in \{1,\dots,\kappa\}$, then our construction is finished and we can proceed with Step 4 below. So suppose that s'_1 is an instance of some term $t_1\vartheta_{1k}$. W.l.o.g. we assume $k=2$, i.e. s'_1 is an instance of $t_1\vartheta_{12}$. Moreover, by construction, s'_1 is an instance of every $[p_{j\alpha_j} : X_{j\alpha_j}]$ with $j \in \{\kappa+1,\dots,m_1\}$ and of every $[q_{j\beta_j} : Y_{j\beta_j}]$ with $j \in \{2,\dots,n\}$. Hence, there exists a solution σ'_2 of $Z\pi\varphi'_2$, s.t. $s'_1 = p_{m_1\alpha_{m_1}}\pi\varphi'_2\sigma'_2$. But then we can also modify this substitution σ'_2 to the substitutions τ'_2 and τ_2 with $z_2\tau'_2 = C_0(D_2)$, $z'_2\tau'_2 = C_1(D_2)$ and $z_2\tau_2 = z'_2\tau_2 = C_0(D_2)$, respectively, where D_2 is greater than the depth of any term occurring in $Z\pi\varphi'_2\sigma'_2$, any term $p_{m_1\alpha_{m_1}}\pi\varphi_k$ with $k \in \{1,\dots,\kappa\}$ and, finally, greater than the depth of $mgi(t_1\vartheta'_{12}v'_2, r_\gamma)$ for all terms r_γ from the explicit representation E of $\mathcal{I}$. Then $s_2 = p_{m_1\alpha_{m_1}}\pi\varphi'_2\tau_2$ is an instance of $t_1\vartheta_{12}$, while $s'_2 = p_{m_1\alpha_{m_1}}\pi\varphi'_2\tau'_2$ is not. Moreover it can be shown that s'_2 is not an instance of $t_1\vartheta_{11}$ in the following way:

By construction, s'_1 is not an instance of $t_1\vartheta_{11}v_1$, since there is one occurrence of the variable z_1 in $t_1\vartheta_{11}v_1$, which would have to be instantiated to $C_1(D_1)$, while all other occurrences of z_1 are instantiated to $C_0(D_1)$. Moreover, s'_1 is an instance of $t_1\vartheta_{12}v_2$, i.e. $s'_1 = t_1\vartheta_{12}v_2\eta_2$ for some substitution η_2. Hence, the multiply occurring variable z_2 from the range of $\vartheta_{12}v_2$ cannot be instantiated to a subterm of $C_1(D_1)$. For suppose on the contrary that $z_2\eta_2$ is a subterm of $C_1(D_1)$ then, by construction, this subterm has its root in $C_1(D_1)$ at depth lower than D_1. But then, this subterm occurs at most once in s'_1, whereas it must occur more than once in $t_1\vartheta_{12}v_2\eta_2$. since z_2 has more than 1 occurrence in $rg(\vartheta_2\mu_2)$. On the other hand, s'_2 is constructed from s'_1 by replacing one occurrence of $z_2\eta_2$ in s'_1 by $C_1(D_2)$ and all other occurrences by $C_0(D_2)$. Now suppose that s'_2 is an instance of $t_1\vartheta_{11}$. Note that by the above considerations, the subterm $C_1(D_1)$ from s'_1 occurs in the same place also in s'_2. But then, in order to be an instance of $t_1\vartheta_{11}$, all occurrences of the subterm $C_0(D_1)$ in s'_1 must be transformed into $C_1(D_1)$, when we construct s'_2 from s'_1. Of course, this cannot be the case by the definition of D_2. We have thus proven that s'_2 is an instance of $t_1/t_1\vartheta_{11} \vee t_1\vartheta_{12} \vee t_1\vartheta_{1(\kappa+1)} \vee t_1\vartheta_{1(\kappa+2)} \vee \cdots \vee t_1\vartheta_{1m_1}$. By iterating this construction at most κ times, we finally end up with terms s_k and s'_k for some $k \in \{1,\dots,\kappa\}$, s.t. neither $s_k = p_{m_1\alpha_{m_1}}\pi\varphi'_k\tau_k$ nor $s'_k = p_{m_1\alpha_{m_1}}\pi\varphi'_k\tau'_k$ is contained in $I_2 \vee \cdots \vee I_n$ and s_k is an instance of $t_1\vartheta_{1k}$ while s'_k is an instance of $t_1/t_1\vartheta_{11} \vee \cdots \vee t_1\vartheta_{1m_1}$.

4. *If s'_k is an instance of r_γ, then s_k is also an instance of r_γ*: Exactly like in the last part of the proof of Theorem 3.5, we can derive a contradiction by showing that a term r_γ from the explicit representation of $\mathscr{I}$ contains the ground instance s_k, if s'_k is an instance of r_γ. □

Below, we revisit the disjunction $\mathscr{I}$ of implicit generalizations from Example 3.10 and apply the procedure STEPWISE_REDUCTION to it.

Example 3.13. Let $\mathscr{I} = I_1 \vee I_2$ be a disjunction of implicit generalizations over the Herbrand universe H with signature $\Sigma = \{a, f\}$, where I_1 and I_2 are defined as follows:

$$I_1 = f(f(x_1,x_2), f(x_3,x_4))/[f(f(y_1,y_1), f(y_2,a))$$

$$\vee f(f(y_1,y_1), f(a,y_2)) \vee f(f(y_1,y_2), f(f(y_3,y_4),y_5))],$$

$$I_2 = f(x_1, f(x_2,x_3))/f(y_1, f(y_2,a)).$$

The complement representations P_{12}, P_{13}, and Q_2 have already been computed in Example 3.10. It will turn out that the remaining sets P_{11}, P_{21}, and Q_1 are not needed here. Hence, their computation is omitted.

As far as Step 2 of the procedure STEPWISE_REDUCTION is concerned, we start the execution of the outermost loop with $i = 1$. In this case, we may add the only linear instance $f(f(y_1,y_2), f(f(y_3,y_4),y_5))$ on the right-hand side of I_1 to R_1 and we set $K := \{1,2\}$ in Step 2.1. In the nested loops of Step 2.2, we have to inspect all possible values of $(\beta_1,\dots,\beta_{i-1},\beta_{i+1},\dots,\beta_n)$ and $(\alpha_1,\dots,\alpha_{m_i})$. Actually, $(\beta_1,\dots,\beta_{i-1},\beta_{i+1},\dots,\beta_n)$ is simply a single index β_2 of a term $q_{2\beta_2}$ in Q_2. Hence, β_2 can take one of the values in $\{1,2,3\}$. The cases $\beta_2 = 1$ and $\beta_2 = 2$ are left as an exercise. We move straight away to $\beta_2 = 3$. Then $q_{2\beta_2} = q_{23} = f(z_{41}, f(z_{42},a))$ holds.

Now we consider the inner loop over all values of $(\alpha_1,\dots,\alpha_{m_i})$. We start off with $\alpha_1 = 1$, $\alpha_2 = 1$, and $\alpha_3 = 1$. In the while-loop over K we have to check whether the Conditions 1–3 hold. In fact, it turns out that they do hold in this case, namely: For $k \in K = \{1,2\}$, the following *mgu*'s v_k exist.

$$v_1 = mgu(f(f(y_1,y_1), f(y_2,a)), f(f(z_{31},z_{32}), f(a,z_{33})), f(z_{41}, f(z_{42},a)))$$

$$= \{y_2 \leftarrow a, z_{31} \leftarrow y_1, z_{32} \leftarrow y_1, z_{33} \leftarrow a, \dots\},$$

$$v_2 = mgu(f(f(y_1,y_1), f(a,y_2)), f(f(z_{31},z_{32}), f(a,z_{33})), f(z_{41}, f(z_{42},a)))$$

$$= \{y_2 \leftarrow a, z_{31} \leftarrow y_1, z_{32} \leftarrow y_1, z_{33} \leftarrow a, \dots\}.$$

Both for $k=1$ and $k=2$, there are no constraints to be considered. Finally, also Condition 3 holds for both values of k. This can be seen as follows: Let $t_1\vartheta_{1k}$ denote the kth term on the right-hand side of I_1. Then the variable y_1 occurs twice both in the range of ϑ_{11} and of ϑ_{12}. However, y_1 is neither bound to a ground term by v_1 nor by v_2. But then the procedure STEPWISE_REDUCTION halts with failure. Hence, by Theorem 3.12, $\mathscr{I}$ has no finite explicit representation.

We are now ready to prove the main result of this paper:

Theorem 3.14 (coNP-completeness). *The finite explicit representability problem of disjunctions of implicit generalizations is coNP-complete.*

Proof. By the coNP-hardness result from [16], we only have to prove the coNP-membership. Let $\mathscr{I}=I_1\vee\cdots\vee I_n$ with $I_i=t_i/t_i\vartheta_{i1}\vee\cdots\vee t_i\vartheta_{im_i}$ be a disjunction of implicit generalizations and let the sets P_{ij} and Q_k be defined according to Step 1 of the procedure STEPWISE_REDUCTION. Moreover, w.l.o.g., suppose that there exists a $\lambda\in\{0,\dots,m_i\}$, s.t. $\forall k\leqslant\lambda$, $t_i\vartheta_{ik}$ is a linear instance of t_i and $\forall k>\lambda$, $t_i\vartheta_{ik}$ is a non-linear one. Then we can check via the following non-deterministic algorithm that $\mathscr{I}$ has no finite explicit representation:

(1) Guess values of i, $(\alpha_1,\dots,\alpha_{m_i})$, $(\beta_1,\dots,\beta_{i-1},\beta_{i+1},\dots,\beta_n)$, and $K\subseteq\{1,\dots,\lambda_i\}$ with $K\neq\emptyset$.
(2) Check for all $k\in K$, that the Conditions 1–3 in the while-loop of the procedure STEPWISE_REDUCTION hold.

This algorithm is basically a non-deterministic version of the procedure STEPWISE_REDUCTION. It clearly works in non-deterministically polynomial time, provided that an efficient unification algorithm is used (cf. [18]). Moreover, its correctness follows immediately from the Theorems 3.11 and 3.12 on the deterministic version of this algorithm. □

4. Negation elimination from simple equational formulae

In [23], implicit generalizations of terms were extended in the obvious way to implicit generalizations of tuples of terms, i.e. let $\vec{t},\vec{t}_1,\dots,\vec{t}_m$ be k-tuples of terms over some Herbrand universe H. Then the implicit generalization $I=\vec{t}/(\vec{t}_1\vee\cdots\vee\vec{t}_m)$ represents all k-tuples of ground terms $\vec{s}\in H^k$, s.t. $\vec{s}$ is an instance of $\vec{t}$ but not an instance of any tuple $\vec{t}_i$. For the sake of readability, we have only considered implicit generalizations of *terms* in Section 3. However, it is easy to verify that all the results in Section 3 as well as in [13] (for single implicit generalizations) and in [23] (for disjunctions of implicit generalizations) still hold, when we consider *tuples of terms* instead. In particular, it is trivial to modify the procedure STEPWISE_REDUCTION from Section 3.3 to a new procedure STEPWISE_REDUCTION' that takes a Herbrand universe H and a disjunction $\mathscr{I}$ of implicit generalizations of *k-tuples of terms* as an input and returns an n-tuple $(R_1, \dots, R_n)$ of sets of *k-tuples of terms*. In fact, in order to obtain this new procedure, we basically just have to replace every occurrence of the string "terms" in our formulation of the procedure STEPWISE_REDUCTION by the string "k-tuples of terms". The same thing applies to the proofs of the correctness and completeness in the Theorems 3.11 and 3.12. Hence, Theorem 3.14 can be rephrased as follows:

Theorem 4.1. *The finite explicit representability problem of disjunctions of implicit generalizations of k-tuples of terms with $k \geqslant 1$ is coNP-complete.*

The proof goes via exactly the same non-deterministic algorithm as in the proof of Theorem 3.14 and is therefore omitted here.

In this section, we shall investigate the negation elimination problem of purely existentially quantified equational formulae. At the heart of our algorithm for this decision problem will be a transformation of existentially quantified equational formulae into a specific form, where the correspondence with implicit generalizations (of tuples of terms) is obvious. Together with the results from Section 3, we thus get the following negation elimination procedure for existentially quantified equational formulae: First we transform a given existentially quantified equational formula $\mathscr{P}$ into a simpler form $\mathscr{P}'$ and then we apply a decision procedure for the finite explicit representability problem to the corresponding implicit generalization (or disjunction of implicit generalizations, respectively). In Section 4.1, we provide the desired transformation of existentially quantified equational formulae for the special case that this formula is simply a conjunction of equations and disequations. The ideas developed for this special case will then be extended in the Sections 4.2 and 4.3 to existentially quantified equational formulae in DNF and CNF, respectively. As far as the complexity is concerned, we shall show the coNP-completeness of the negation elimination problem in case of DNF and the Π_2^p-hardness in case of CNF.

4.1. Conjunctions of equations and disequations

Recall from Section 2.2 that we write $Disequ(\vartheta)$ as a short-hand notation for a disjunction of disequations corresponding to some substitution ϑ, i.e. let $\vartheta = \{v_1 \leftarrow r_1, \dots, v_n \leftarrow r_n\}$, then $Disequ(\vartheta)$ denotes the equational formula $Disequ(\vartheta) \equiv v_1 \neq r_1 \vee \cdots \vee v_n \neq r_n$. Now suppose that an existentially quantified equational formula has the simple form

$$\mathscr{P} \equiv (\exists \vec{x}) \left[\vec{z} = \vec{t} \wedge \bigwedge_{i=1}^{l} Disequ(\vartheta_i) \right],$$

where $\vec{z} = (z_1, \dots, z_k)$ denotes the vector of free variables in $\mathscr{P}$ and these free variables neither occur on the right-hand side $\vec{t}$ of the equation nor anywhere in the subformulae $Disequ(\vartheta_i)$. Moreover, assume that all variables involved in the substitutions ϑ_i also occur in $\vec{t}$, i.e. $dom(\vartheta_i) \cup Var(rg(\vartheta_i)) \subseteq Var(\vec{t})$. Then an H-ground substitution σ is a solution of $\mathscr{P}$, iff the ground term tuple $\vec{z}\sigma$ is contained in the implicit generalization $I = \vec{t}/(\vec{t}\vartheta_1 \vee \cdots \vee \vec{t}\vartheta_l)$. Moreover, negation elimination from $\mathscr{P}$ corresponds to the conversion of I into an explicit generalization. Likewise, if in an equational formula $\mathcal{Q} \equiv \bigvee_{i=1}^{n} \mathcal{Q}_i$, every disjunct $\mathcal{Q}_i$ has such a simple form, then negation elimination from $\mathcal{Q}$ is equivalent to the conversion of the corresponding disjunction of implicit generalizations $\mathcal{I}$ into an explicit generalization.

In this section, we construct a negation elimination procedure for existentially quantified conjunctions of equations and disequations by showing how they can be transformed into equivalent formulae of the above mentioned simple form. This transformation can be done efficiently with the procedure SIMPLIFY_CONJUNCTIONS given in Fig. 4. The target of the first 3 steps of this procedure is to transform the equations into the form $\vec{z} = \vec{t}$, s.t. the free variables $\vec{z}$ neither occur in $\vec{t}$ nor in the remaining formula. Then, in Step 4, the disequations are transformed appropriately, s.t. they contain variables from $Var(\vec{t})$ only. Of course, in the resulting equational formula $\mathcal{P}'$, we may delete all conjuncts d_i''' that are trivially true. Hence, the procedure SIMPLIFY_CONJUNCTIONS indeed transforms any equational formula $\mathcal{P}$ of the form $\mathcal{P} \equiv (\exists \vec{x})(e_1 \wedge \cdots \wedge e_k \wedge d_1 \wedge \cdots \wedge d_l)$ either into the trivially false problem $\mathcal{P}' \equiv \bot$ or into the form $\mathcal{P}' \equiv (\exists \vec{x})(\exists \vec{u})[\vec{z} = \vec{t} \wedge \bigwedge_{i=1}^{l} Disequ(\vartheta_i)]$ described above. It only remains to show that in either case, $\mathcal{P}$ and $\mathcal{P}'$ are equivalent.

Lemma 4.2 (Correctness of procedure SIMPLIFY_CONJUNCTIONS). *Let $\mathcal{P} \equiv (\exists \vec{x})(e_1 \wedge \cdots \wedge e_k \wedge d_1 \wedge \cdots \wedge d_l)$ be an equational formula over some Herbrand universe H, where the e_i's are equations and the d_i's are disequations. Moreover, let $\mathcal{P}'$ denote the equational formula returned by* SIMPLIFY_CONJUNCTIONS *on input $\mathcal{P}$. Then $\mathcal{P}$ and $\mathcal{P}'$ are equivalent.*

Proof. We first prove that the formula $\mathcal{R} \equiv (\exists \vec{x})(\exists \vec{u})[\vec{z} = \vec{t} \wedge \bigwedge_{i=1}^{l} d_i'']$, which we get after the Steps 1–3, is equivalent to $\mathcal{P}$. Actually, if the equations $e_1, \ldots, e_k$ are not unifiable, then the conjunction $e_1 \wedge \cdots \wedge e_k$ and, therefore, also $\mathcal{P}$ is unsatisfiable. On the other hand, if $\mu = mgu(e_1, \ldots, e_k) = \{z_1 \leftarrow t_1, \ldots, z_\alpha \leftarrow t_\alpha, x_1 \leftarrow s_1, \ldots, x_\beta \leftarrow s_\beta\}$ exists, then we may replace the equations $e_1 \wedge \cdots \wedge e_k$ by $\bigwedge_{i=1}^{\alpha}(z_i = t_i) \wedge \bigwedge_{i=1}^{\beta}(x_i = s_i)$. Moreover, each d_i may be replaced by $d_i' \equiv d_i\mu$. So far, we have basically followed Theorem 4.8 in [11]. Now note that, for any equational formula $\mathcal{Q}$ and any variable u not occurring in $\mathcal{Q}$, we have $\mathcal{Q} \approx (\exists u)[z = u \wedge \mathcal{Q}\{z \leftarrow u\}]$. Hence, $\mathcal{P}$ is equivalent to

$$\mathcal{R}' \equiv (\exists \vec{x})(\exists \vec{u}) \left[\bigwedge_{i=\alpha+1}^{n} (z_i = u_i) \wedge \bigwedge_{i=1}^{\alpha} (z_i = t_i v) \wedge \bigwedge_{i=1}^{\beta} (x_i = s_i v) \wedge \bigwedge_{i=1}^{l} d_i \mu v \right].$$

Note that the formula $\mathcal{R} \equiv (\exists \vec{x})(\exists \vec{u})[\vec{z} = \vec{t} \wedge \bigwedge_{i=1}^{l} d_i'']$ is obtained from $\mathcal{R}'$ by deleting the equations $x_i = s_i v$ for all $i \in \{1, \ldots, \beta\}$. We have to prove that no solutions are added to $\mathcal{R}'$, if we delete these equations. So suppose that $\sigma = \{z_1 \leftarrow v_1, \ldots, z_n \leftarrow v_n\}$ is a solution of $\mathcal{R}$. Then there exists a ground substitution τ with $dom(\tau) = \{x_{\beta+1}, \ldots, x_m, u_{\alpha+1}, \ldots, u_n\}$, s.t. $\bigwedge_{i=1}^{\alpha}(z_i\sigma = t_i v\tau) \wedge \bigwedge_{i=\alpha+1}^{n}(z_i\sigma = u_i\tau) \wedge \bigwedge_{i=1}^{l}(d_i\mu v\tau) \approx \top$ holds. By the definition of the mgu μ, the variables $x_1, \ldots, x_\beta$ from the domain of μ cannot occur in the range of μ. Moreover, by applying μ to the disequations, the variables $x_1, \ldots, x_\beta$ do not occur any more in the disequations in $\mathcal{R}'$. In other words, the only occurrence of the

SIMPLIFICATION OF CONJUNCTIONS

<u>input</u>: equational formula $\mathcal{P} \equiv (\exists \vec{x})(e_1 \wedge \ldots \wedge e_k \wedge d_1 \wedge \ldots \wedge d_l)$
with existentially quantified variables $\vec{x} = (x_1, \ldots, x_m\}$
and free variables $\vec{z} = (z_1, \ldots, z_n)$,
s.t. the e_i's are equations and the d_i's are disequations.

<u>output</u>: equivalent formula $\mathcal{P}'$ with "simpler" form.

begin

/* Step 1 */
  if the equations $e_1, \ldots, e_k$ are not unifiable **then return** $\perp$; **fi**;

/* Step 2 */
  $\mu := mgu(e_1, \ldots, e_k)$;
  for $i := 1$ **to** l **do begin**
    $d_i' := d_i\mu$;
    if d_i' has the form $t \neq t$ for some term t **then return** $\perp$; **fi**;
  end;

/* Step 3 */
  let $\mu = \{z_1 \leftarrow t_1, \ldots, z_\alpha \leftarrow t_\alpha, x_1 \leftarrow s_1, \ldots, x_\beta \leftarrow s_\beta\}$
    for some $\alpha \leq n$ and $\beta \leq m$;
  let $\vec{u} = (u_{\alpha+1}, \ldots, u_n)$ be fresh, pairwise distinct variables;
  $\nu := \{z_{\alpha+1} \leftarrow u_{\alpha+1}, \ldots, z_n \leftarrow u_n\}$;
  $\vec{t} := (z_1\mu\nu, \ldots, z_\alpha\mu\nu, u_{\alpha+1}, \ldots, u_n)$;
  for $i := 1$ **to** l **do** $d_i'' := d_i'\nu$;
  /* Now $\mathcal{R} \equiv (\exists \vec{x})(\exists \vec{u})[\vec{z} = \vec{t} \wedge \bigwedge_{i=1}^{l} d_i'']$ is equivalent to $\mathcal{P}$ */

/* Step 4 */
  for $i := 1$ **to** l **do begin**
  /* Step 4.1 */
    if the equation $\neg d_i''$ is not unifiable **then**
      $d_i''' := \top$;
    else
  /* Step 4.2 */
      $\vartheta_i := mgu(\neg d_i'')$;
      let $\vartheta_i = \{y_1 \leftarrow r_1, \ldots, y_\gamma \leftarrow r_\gamma\}$, s.t. for all j,
        $y_j \in \{x_{\beta+1}, \ldots, x_m, u_{\alpha+1}, \ldots, u_n\}$ holds;
      if $dom(\vartheta_i) \cup Var(rg(\vartheta_i)) \subseteq Var(\vec{t})$ **then** $d_i''' := Disequ(\vartheta_i)$;
      else $d_i''' := \top$;
      fi;
    fi;
  end; /* for i */
  return $(\exists \vec{x})(\exists \vec{u})[\vec{z} = \vec{t} \wedge \bigwedge_{i=1}^{l} d_i''']$;

end.

Fig. 4. Procedure SIMPLIFY_CONJUNCTIONS.

variables $x_1,\dots,x_\beta$ in $\mathscr{R}'$ is on the left-hand side of the equations $\bigwedge_{i=1}^{\beta}(x_i = s_i v)$. But then, we can extend τ to $\tau' = \tau \cup \{x_1 \leftarrow s_1 v\tau,\dots,x_\beta \leftarrow s_\beta v\tau\}$, s.t. also $(x_i = s_i v)\tau' \approx \top$ holds for all $i \in \{1,\dots,\beta\}$.

It remains to show that the transformation of the disequations d_i'' in $\mathscr{R}$ to d_i''' via Step 4 of our algorithm is correct. The replacement of d_i'' by $\top$ in Step 4.1 is correct by the definition of *mgu*'s. On the other hand, if $\neg d_i''$ is unifiable with *mgu* ϑ_i, then d_i'' may be replaced by the disjunction $Disequ(\vartheta_i)$ of disequations. These two cases are basically the negated form of Theorem 4.8 in [11], which we already made use of above. Now we have to show that, if $dom(\vartheta_i) \cup Var(rg(\vartheta_i)) \not\subseteq Var(\vec{t})$, then $Disequ(\vartheta_i)$ may be replaced by $\top$ or, equivalently, $Disequ(\vartheta_i)$ may be deleted. W.l.o.g., we assume that the disequations $d_1,\dots,d_\delta$ for some $\delta \leqslant l$ are the ones that are replaced by $\top$ in Step 4.2. Now let $\vec{v}$ denote the vector of those variables from $\vec{x}$ which do not occur in $\vec{t}$ and, for every $i \in \{1,\dots,\delta\}$, let $y_{j_i} \neq s_{j_i}$ denote a disequation in $Disequ(\vartheta_i)$ which contains a variable from $\vec{v}$. Then the equivalences $(\exists\vec{v})\mathscr{Q} \approx \mathscr{Q} \approx (\exists\vec{v})[\mathscr{Q} \wedge \bigwedge_{i=1}^{\delta} y_{j_i} \neq s_{j_i}]$ hold for any equational formula $\mathscr{Q}$ that contains no variable from $\vec{v}$. Moreover, $y_{j_i} \neq s_{j_i} \leqslant d_i$ holds for every $i \in \{1,\dots,\delta\}$. We thus have

$$\mathscr{R} \leqslant (\exists\vec{x})(\exists\vec{u})\left(\vec{z} = \vec{t} \wedge \bigwedge_{i=\delta+1}^{l} d_i'''\right)$$

$$\approx (\exists\vec{x})(\exists\vec{u})\left(\vec{z} = \vec{t} \wedge \bigwedge_{i=1}^{\delta} y_{j_i} \neq s_{j_i} \wedge \bigwedge_{i=\delta+1}^{l} d_i'''\right) \leqslant \mathscr{R}.$$

But then, all the above "$\leqslant$"-relations can actually be replaced by "$\approx$". Hence, $\mathscr{R}$ and $\mathscr{P}' \equiv (\exists\vec{x})(\exists\vec{u})(\vec{z} = \vec{t} \wedge \bigwedge_{i=\delta+1}^{l} d_i''')$ are indeed equivalent. $\square$

Together with the decision procedure for the explicit representability problem from Section 3.1 or [13], we immediately get a negation elimination procedure for existentially quantified conjunctions of equations and disequations. These ideas are illustrated in the following example.

Example 4.3. Let H be the Herbrand universe with signature $\Sigma = \{a, f, g\}$ and let the equational formula $\mathscr{P}$ over H be defined as follows:

$$\mathscr{P} \equiv (\exists x_1, x_2, x_3)[g(z_1, x_2) = g(f(x_1), f(x_3))$$

$$\wedge z_2 \neq z_1 \wedge f(z_2) \neq f(a) \wedge x_3 \neq z_1].$$

Then the *mgu* μ of the equations has the form $\mu = \{z_1 \leftarrow f(x_1), x_2 \leftarrow f(x_3)\}$ and, therefore, $\mathscr{P}$ is equivalent to

$$\mathscr{Q} \equiv (\exists x_1, x_2, x_3)[(z_1, x_2) = (f(x_1), f(x_3))$$

$$\wedge z_2 \neq f(x_1) \wedge f(z_2) \neq f(a) \wedge x_3 \neq f(x_1)].$$

Of course, neither in Step 1 nor in Step 2, the algorithm SIMPLIFY_CONJUNCTIONS will return the trivially false problem $\perp$. Now we come to Step 3 of our algorithm. In order to bring the free variable z_2 to the left-hand side of the equations and to remove z_2 from the disequations, we define the substitution $v = \{z_2 \leftarrow u_2\}$. Then $\mathscr{Q}$ is equivalent to

$$\mathscr{Q}' \equiv (\exists x_1, x_2, x_3, u_2)[(z_1, z_2, x_2) = (f(x_1), u_2, f(x_3)) \\ \wedge u_2 \neq f(x_1) \wedge f(u_2) \neq f(a) \wedge x_3 \neq f(x_1)].$$

Finally, again by Step 3 of procedure SIMPLIFY_CONJUNCTIONS, we may delete the equation $x_2 = f(x_3)$. We thus get

$$\mathscr{R} \equiv (\exists x_1, x_2, x_3, u_2)[(z_1, z_2) = (f(x_1), u_2) \\ \wedge u_2 \neq f(x_1) \wedge f(u_2) \neq f(a) \wedge x_3 \neq f(x_1)].$$

By Step 4 of procedure SIMPLIFY_CONJUNCTIONS, we may transform the disequations in the following way: $u_2 \neq f(x_1)$ is left unchanged. $f(u_2) \neq f(a)$ may be simplified to $u_2 \neq a$. Finally, $x_3 \neq f(x_1)$ may be deleted (due to the presence of the variable x_3, which does not occur any more in the equations). Finally, Hence, the original formula $\mathscr{P}$ is equivalent to

$$\mathscr{P}' \equiv (\exists x_1, x_2, u_2)[(z_1, z_2) = (f(x_1), u_2) \wedge u_2 \neq f(x_1) \wedge u_2 \neq a].$$

Negation elimination from $\mathscr{P}'$ is equivalent to the conversion of the implicit generalization of term tuples

$$I = (f(x_1), u_2)/((f(x_1), f(x_1)) \vee (f(x_1), a))$$

into an explicit generalization. Note that the term $(f(x_1), u_2)\{u_2 \leftarrow f(x_1)\} = (f(x_1), f(x_1))$ is non-linear w.r.t. $(f(x_1), u_2)$. In order to apply the transformation rule T from Definition 3.3, we have to compute the complement representation P_2 of the second tuple $(f(x_1), a) = (f(x_1), u_2)\{u_2 \leftarrow a\}$ w.r.t. $(f(x_1), u_2)$. We thus have $P_2 = \{(f(x_1), u_2)\lambda_1, (f(x_1), u_2)\lambda_2\}$ with $\lambda_1 = \{u_2 \leftarrow f(y)\}$ and $\lambda_2 = \{u_2 \leftarrow g(z)\}$. Now restricting the non-linear instance $(f(x_1), f(x_1))$ of $(f(x_1), u_2)$ to the instances of $(f(x_1), u_2)\lambda_1$ leaves $(f(x_1), f(x_1))$ unchanged. Hence, by Theorem 3.5, the implicit generalization I does not have a finite explicit representation. Consequently, negation elimination from $\mathscr{P}'$ and, therefore, also from the original formula $\mathscr{P}$ is impossible.

Remark. Let I be an implicit generalization that corresponds to an equational formula $\mathscr{P}'$ resulting from our procedure SIMPLIFY_CONJUNCTIONS. Moreover, let $\vec{t}\vartheta_i$ be a term tuple on the right-hand side of I. Then ϑ_i is either a ground substitution or it gives rise to a non-linear instance $\vec{t}\vartheta_i$ of $\vec{t}$. Recall from the above example that the non-linear instance $(f(x_1), f(x_1)) = (f(x_1), u_2)\{u_2 \leftarrow f(x_1)\}$ of $(f(x_1), u_2)$ is in fact *essentially* non-linear. Actually, by a very technical proof, it can be shown that any such non-linear instance $\vec{t}\vartheta_i$ on the right-hand side of I is *essentially* non-linear (cf. [20]). However, we are not going to make use of this property here, since it is not needed for deriving the complexity results in the subsequent subsections anyway.

4.2. Equational formulae in DNF

The algorithms in [2] and [3] for solving equational formulae result in the transformation of an arbitrary equational formula into the so-called "definition with constraints", which is basically an existentially quantified equational formula $\mathscr{D}$ in DNF. Then $\mathscr{D} \equiv (\exists \vec{x})[\mathscr{D}_1 \vee \cdots \vee \mathscr{D}_n]$, where each $\mathscr{D}_i$ is a conjunction of equations and disequations. Of course, $(\exists \vec{x})[\mathscr{D}_1 \vee \cdots \vee \mathscr{D}_n]$ is equivalent to $[(\exists \vec{x})\mathscr{D}_1] \vee \cdots \vee [(\exists \vec{x})\mathscr{D}_n]$. Hence, we can apply our procedure SIMPLIFY_CONJUNCTIONS from the previous section to each disjunct $(\exists \vec{x})\mathscr{D}_i$ separately in order to get simplified disjuncts of the form

$$(\exists \vec{x})(\exists \vec{u})\mathscr{D}'_i \quad \text{with } \mathscr{D}'_i \equiv \left[\vec{z} = \vec{t}_i \wedge \bigwedge_{j=1}^{l_i} Disequ(\vartheta_{ij})\right],$$

where $\vec{z} = (z_1, \dots, z_k)$ denotes the free variables in $\mathscr{D}$. Again, the one-to-one correspondence with the disjunction $\mathscr{I} = I_1 \vee \cdots \vee I_n$ of implicit generalizations, where $I_i = \vec{t}_i/(\vec{t}_i\vartheta_{i1} \vee \cdots \vee \vec{t}_i\vartheta_{il_i})$ holds, is obvious. Hence, by combining the procedure SIMPLIFY_CONJUNCTIONS from Section 4.1 with the procedure STEPWISE_REDUCTION from Section 3.3, we get a negation elimination procedure for existentially quantified equational formulae in DNF. Clearly, the transformation in procedure SIMPLIFY_CONJUNCTIONS can be done in polynomial time, since it only consists of several unification steps followed by some cheap operations like applying a unifier to another subformula of $\mathscr{P}$ and checking for the existence of certain variables in the resulting terms. Hence, together with Theorem 3.14, we immediately get the coNP-membership of the negation elimination problem of existentially quantified equational formulae in DNF. In this section we show that this bound is tight. The following Proposition will be helpful for the coNP-hardness proof.

Proposition 4.4. *Let $\mathscr{P}$ be an equational formula over some Herbrand universe H with free variables in $\vec{z} = (z_1, \dots, z_{k+2})$ and suppose that there exist ground terms $s_1, \dots, s_k$ in H, s.t. for all ground terms s_{k+1} and s_{k+2} in H, the following equivalence holds*:

$$\sigma = \{z_1 \leftarrow s_1, \dots, z_k \leftarrow s_k, z_{k+1} \leftarrow s_{k+1}, z_{k+2} \leftarrow s_{k+2}\} \text{ is a solution of } \mathscr{P},$$

$$\text{iff } s_{k+1} \text{ and } s_{k+2} \text{ are distinct.}$$

Then the negation elimination from $\mathscr{P}$ is impossible.

Proof. (Indirect) Let the set $\mathscr{T}$ of ground term tuples in H^{k+2} be defined as

$$\mathscr{T} = \{(t_1, \dots, t_{k+2}) \mid \sigma = \{z_1 \leftarrow t_1, \dots, z_{k+2} \leftarrow t_{k+2}\} \text{ is a solution of } \mathscr{P}\}.$$

Moreover suppose that negation elimination from $\mathscr{P}$ is possible. Then there exists a finite set $\mathscr{R} = \{\vec{r}_1, \dots, \vec{r}_n\}$ of $(k+2)$-tuples of terms over H, s.t. $\mathscr{T}$ and the H-ground instances of $\mathscr{R}$ coincide. Now consider the $(k+2)$-tuple $(s_1,\dots,s_k,C_0(D),C_1(D))$, where $C_0(D)$ and $C_1(D)$ are defined as in Section 2.5 and D is chosen sufficiently big. Then $\sigma' = \{z_1 \leftarrow s_1,\dots,z_k \leftarrow s_k, z_{k+1} \leftarrow C_0(D), z_{k+2} \leftarrow C_1(D)\}$ is a solution of $\mathscr{P}$ and, therefore, $(s_1,\dots,s_k,C_0(D),\ C_1(D))$ is contained in $\mathscr{T}$. Hence, $(s_1,\dots,s_k,C_0(D),$ $C_1(D))$ is an instance of some term tuple $\vec{r}_j \in \mathscr{R}$. Analogously to the third part of the proof of Theorem 3.5, it can be shown that then also $(s_1,\dots,s_k,C_0(D),C_0(D))$ is an instance of $\vec{r}_j$. However, $\sigma = \{z_1 \leftarrow s_1,\dots,z_k \leftarrow s_k, z_{k+1} \leftarrow C_0(D), z_{k+2} \leftarrow C_0(D)\}$ is not a solution of $\mathscr{P}$ and, therefore, $(s_1,\dots,s_k,C_0(D),C_0(D))$ is not contained in $\mathscr{T}$. But this is a contradiction to the assumption that $\mathscr{T}$ and the H-ground instances of $\mathscr{R}$ coincide. □

Theorem 4.5 (coNP-completeness). *The negation elimination problem of purely existentially quantified equational formulae in DNF is coNP-complete.*

Proof. The *coNP-membership* is clear by our comments above and by the coNP-membership result in Theorem 3.14. The *coNP-hardness* can be shown by a reduction from the emptiness problem of implicit generalizations, which is a well-known coNP-complete problem (cf. [8–10,16]), namely: Let $I = t/s_1 \vee \cdots \vee s_n$ be an implicit generalization, s.t. every term s_i is an instance of t. Moreover, let $\vec{x}$ denote a vector of variables, s.t. $Var(\vec{t}) \subseteq \vec{x}$, $Var(\vec{s}_i) \subseteq \vec{x}$, and $Var(\vec{t}) \cap Var(\vec{s}_i) = \emptyset$ holds for every i. Finally let u, v, z_1, z_2, z_3 be fresh, pairwise distinct variables and let $\vec{z}$ be defined as $\vec{z} = (z_1, z_2, z_3)$. Then we define the formula $\mathscr{P}$ in DNF as follows:

$$\mathscr{P} \equiv (\exists \vec{x})(\exists u, v)\left[(\vec{z} = (t,u,v) \wedge u \neq v) \vee \bigvee_{i=1}^{n} \vec{z} = (s_i,u,u)\right].$$

Of course this transformation can be done in polynomial time. It only remains to show that the implicit generalization $I = t/s_1 \vee \cdots \vee s_n$ is empty, iff negation elimination from $\mathscr{P}$ is possible. For the "only if"-direction, suppose that I is empty. Then t is equivalent to $s_1 \vee \cdots \vee s_n$. Hence, $\mathscr{P}$ is equivalent to $\mathscr{P}' \equiv (\exists \vec{x})(\exists u,v)\vec{z} = (t,u,v)$. For the "if"-direction, suppose that there exists a ground instance t' of t that is not contained in any term s_i. Thus, for every ground substitution of the form $\sigma = \{z_1 \leftarrow t', z_2 \leftarrow t_2, z_3 \leftarrow t_3\}$ we know that σ is a solution of $\mathscr{P}$, iff t_2 and t_3 are distinct ground terms from H. But then, by Proposition 4.4, negation elimination from $\mathscr{P}$ is impossible. □

4.3. Equational formulae in CNF

A straightforward negation elimination algorithm for purely existentially quantified equational formulae in CNF consists of a transformation from CNF into DNF followed by the algorithm outlined in the previous section. Of course, in the worst case, this transformation into DNF leads to an exponential blow-up. However, by the Π_2^p-hardness to be shown in Theorem 4.6 below, we cannot expect to do much better than this anyway.

Theorem 4.6 (Π_2^p-hardness). *The negation elimination problem of existentially quantified equational formulae in CNF is Π_2^p-hard.*

Proof. Recall the well-known Σ_2^p-hard problem $3\mathrm{QSAT}_2$ (=quantified satisfiability with two alternating blocks of quantifiers cf. [22]), i.e. let $P = \{p_1, \dots, p_k\}$ and $R = \{r_1, \dots, r_l\}$ be sets of propositional variables and let $E = (l_{11} \wedge l_{12} \wedge l_{13}) \vee \dots \vee (l_{n1} \wedge l_{n2} \wedge l_{n3})$ be a Boolean formula, s.t. the literals l_{ij} in E are unnegated or negated propositional variables from $P \cup R$. Then we have to decide, whether the quantified Boolean sentence $(\exists P)(\forall R)E$ is satisfiable. Now let H denote some Herbrand universe with signature Σ and let $a \in \Sigma$ denote an arbitrary constant symbol. Then we reduce such an instance of the $3\mathrm{QSAT}_2$ problem to the complementary problem of the negation elimination problem in the following way:

$\mathscr{P} \equiv (\exists \vec{x})[(d_{11} \vee d_{12} \vee d_{13} \vee z_{k+1} \neq z_{k+2}) \wedge \dots \wedge (d_{n1} \vee d_{n2} \vee d_{n3} \vee z_{k+1} \neq z_{k+2})]$, where $\vec{z} = (z_1, \dots, z_{k+2})$ denotes the free variables in $\mathscr{P}$, $\vec{x}$ is of the form $\vec{x} = (x_1, \dots, x_l)$ and the d_{ij}'s are defined as follows:

$$d_{ij} \equiv \begin{cases} z_\gamma \neq a \text{ if } l_{ij} \text{ is an unnegated propositional variable } p_\gamma \in P, \\ z_\gamma = a \text{ if } l_{ij} \text{ is of the form } \neg p_\gamma \text{ for some } p_\gamma \in P, \\ x_\gamma \neq a \text{ if } l_{ij} \text{ is an unnegated propositional variable } r_\gamma \in R, \\ x_\gamma = a \text{ if } l_{ij} \text{ is of the form } \neg r_\gamma \text{ for some } r_\gamma \in R. \end{cases}$$

It is easy to check that the Boolean sentence $(\exists P)(\forall R)E$ is satisfiable, iff the equational formula $(\exists \vec{z})(\forall \vec{x})\neg[(d_{11} \vee d_{12} \vee d_{13}) \wedge \dots \wedge (d_{n1} \vee d_{n2} \vee d_{n3})]$ is satisfiable. We claim that then also the following equivalence holds: Negation elimination from $\mathscr{P}$ is impossible, iff $(\exists P)(\forall R)E$ is satisfiable:

"*if*"-*direction*: Suppose that $(\exists P)(\forall R)E$ is satisfiable. Then $(\exists \vec{z})(\forall \vec{x})\neg[(d_{11} \vee d_{12} \vee d_{13}) \wedge \dots \wedge (d_{n1} \vee d_{n2} \vee d_{n3})]$ is also satisfiable. Hence, there exists a substitution $\sigma = \{z_1 \leftarrow s_1, \dots, z_k \leftarrow s_k\}$, s.t. $(\forall \vec{x})\neg[(d_{11} \vee d_{12} \vee d_{13}) \wedge \dots \wedge (d_{n1} \vee d_{n2} \vee d_{n3})]\sigma \approx \top$. This is equivalent to the condition that $(\exists \vec{x})[(d_{11} \vee d_{12} \vee d_{13}) \wedge \dots \wedge (d_{n1} \vee d_{n2} \vee d_{n3})]\sigma \approx \bot$ holds. On the other hand, $\mathscr{P} \equiv (\exists \vec{x})[(d_{11} \vee d_{12} \vee d_{13} \vee z_{k+1} \neq z_{k+2}) \wedge \dots \wedge (d_{n1} \vee d_{n2} \vee d_{n3} \vee z_{k+1} \neq z_{k+2})]$ can of course be transformed into $\mathscr{P}' \equiv z_{k+1} \neq z_{k+2} \vee (\exists \vec{x})[(d_{11} \vee d_{12} \vee d_{13}) \wedge \dots \wedge (d_{n1} \vee d_{n2} \vee d_{n3})]$. Hence, for every extension σ' of σ with $\sigma' = \{z_1 \leftarrow s_1, \dots, z_k \leftarrow s_k, z_{k+1} \leftarrow t_{k+1}, z_{k+2} \leftarrow t_{k+2}\}$ we know that σ' is a solution of $\mathscr{P}'$ (and, therefore of $\mathscr{P}$), iff t_{k+1} and t_{k+2} are distinct ground terms from H. But then, by Proposition 4.4, negation elimination from $\mathscr{P}$ is impossible.

"*only if*"-*direction*: Suppose that $(\exists P)(\forall R)E$ is unsatisfiable. Then the equational formula $(\exists \vec{z})(\forall \vec{x})\neg[(d_{11} \vee d_{12} \vee d_{13}) \wedge \dots \wedge (d_{n1} \vee d_{n2} \vee d_{n3})]$ is also unsatisfiable. In other words, $(\forall \vec{z})(\exists \vec{x})(d_{11} \vee d_{12} \vee d_{13}) \wedge \dots \wedge (d_{n1} \vee d_{n2} \vee d_{n3}) \approx \top$. But then $\mathscr{P} \approx \top$ holds and, therefore, negation elimination from $\mathscr{P}$ is clearly possible. □

Unfortunately, it is not clear whether the Π_2^p-membership also holds. The obvious upper bound on the negation elimination problem of equational formulae in CNF is coNEXPTIME, since we can of course transform the CNF into DNF (possibly at the expense of an exponential blow-up) and then apply the coNP-membership result from the previous section. An exact complexity classification of the negation elimination problem in case of CNF has to be left for future research.

5. Related works

The notion of *linear terms*, which is essential to our approach, was introduced in [24] in the area of formal specification. It was later rediscovered in the area of automated deduction (e.g. in [13]). A decision procedure for the finite explicit representability problem of a single implicit generalization $I = t/t_1 \vee \cdots \vee t_m$ was first presented in [13]. Its basic idea is to split the original problem into subproblems via the complement of a *linear* instance t_i on the right-hand side of I, i.e. let $P = \{p_1, \ldots, p_M\}$ represent the complement of t_i w.r.t. t and suppose that all terms t_i, p_j and t are pairwise variable disjoint. Then I is equivalent to the disjunction $\bigvee_{j=1}^{M} I_j$ with

$$I_j = p_j/mgi(t_1, p_j) \vee \cdots \vee mgi(t_{i-1}, p_j) \vee mgi(t_{i+1}, p_j) \vee \cdots \vee mgi(t_n, p_j).$$

The subproblems I_j thus produced have strictly fewer terms on the right-hand side. Hence, by applying this splitting step recursively to each subproblem, we either manage to remove all terms from the right-hand side of all subproblems or we eventually encounter a subproblem with only non-linear instances on the right-hand side. For the latter case, it has been shown in [13], that I has no finite explicit representation.

In Section 3.1, we have provided a new algorithm for deciding the finite explicit representability problem of a single implicit generalization. Note that this algorithm starts from the "opposite direction", i.e. rather than removing the linear terms from the right-hand side until finally only non-linear ones are left, our algorithm tries to replace the non-linear terms by linear ones until it finally detects a non-linear term which cannot be replaced by linear ones. If an implicit generalization has only few non-linear terms on the right-hand side, then our algorithm from Section 3.1 may possibly be advantageous. However, in general, the algorithm from [13] will by far outperform our algorithm from Section 3.1 due to the hyper-exponential behaviour discussed in Section 3.3, i.e. suppose that there are several non-linear terms on the right-hand side of an implicit generalization I and that we may replace one such non-linear term t_1 by linear ones $u_1, \ldots, u_M$ via the transformation rule T from Definition 3.3. Then this replacement step, in general, yields *exponentially many* terms $u_1, \ldots, u_M$ (w.r.t. the number m of terms and also w.r.t. the size of these terms). Suppose that we next apply the transformation rule T to another non-linear term t_2 on the right-hand side of I and that we may replace t_2 by the linear terms $v_1, \ldots, v_N$. Then N is actually exponential w.r.t. M. But then N is doubly exponential w.r.t. the size of the original implicit generalization. Of course, having to restrict yet another non-linear term t_3 to the complement of the terms $v_1, \ldots, v_N$ makes the situation even worse, etc.

In [23], the algorithm from [13] is extended to disjunctions of implicit generalizations. This algorithm consists of two rewrite rules: One is exactly the splitting rule from [13]. The other one basically allows one to restrict the instances of a non-linear term t_{ij} on the right-hand side of an implicit generalization I_i to the complement of another implicit generalization $I_k = t_k/t_{k1} \vee \cdots \vee t_{km_k}$, provided that $mgi(t_{ij}, t_k)$ is a linear instance of t_{ij}. Hence, when a disjunction of implicit generalizations contains several disjuncts with non-linear terms on the right-hand side, then the algorithm of [23] also suffers from the above mentioned exponential blow-up whenever a term is restricted to the complement of terms which are themselves the result of such a restriction step. Of course, our algorithm from Section 3.2 has this problem as well. However, the algorithm STEPWISE_REDUCTION from Section 3.3 is clearly better than this, i.e. the terms that we add to the sets $\mathcal{R}_i$ in Step 2.2 of this algorithm result from restricting a non-linear term t_{ij} on the right-hand side of I_i to the complement of some other terms $t_{ij'}$ on the right-hand side of I_i and to the complement of the other implicit generalizations $I_{i'}$. However, we never restrict a term w.r.t. terms which are themselves the result of a previous restriction step.

In [16] an upper bound on the time and space complexity of the algorithm from [13] is given, which is basically exponential w.r.t. the size of an input implicit generalization (cf. [11, Theorem 5.5]). In fact, our algorithm from Section 3.3 also has an upper bound with a single exponentiality, no matter whether we apply this algorithm to a single implicit generalization or to a disjunction of implicit generalizations: In order to see this, recall from Theorem 2.1, that the number of constrained terms in the complement of a term $t\vartheta$ is linearly bounded by the size of this term. Hence, also the number M_{ij} of constrained terms in the representation of the complement of $t_i\vartheta_{ij}$ w.r.t. t_i as well as the number N_k of constrained terms in the complement representation of I_k is linearly bounded by the size of $t_i\vartheta_{ij}$ and I_k, respectively. But then the number of possible values of i, $(\alpha_1,\dots,\alpha_{m_i})$, $(\beta_1,\dots,\beta_{i-1},\beta_{i+1},\dots,\beta_n)$, and $K\subseteq\{1,\dots,m_i\}$ that our algorithm STEPWISE_REDUCTION from Section 3.3 has to inspect, is clearly exponentially bounded in the size of the original disjunction $\mathcal{I}$ of implicit generalizations.

A decision procedure for the negation elimination problem of arbitrary equational formulae was first presented in [23], which comprised two components: On the one hand, Tajine was the first to provide an algorithm for the finite explicit representability problem of disjunctions of implicit generalizations. On the other hand, it was shown in [23], that only a slight extension of the transformation given in [14] is required so as to transform any equational formula into an equivalent formula, where the correspondence with disjunctions of implicit generalizations (of term tuples) is immediately clear. In [5], a different decision procedure for the negation elimination problem of arbitrary equational formulae is given by appropriately extending the reduction system from [3]. Note that even if we restrict this algorithm to purely existentially quantified equational formulae or to equational formulae that correspond to a disjunction of implicit generalizations, then this approach differs significantly from the method of [23] and from our algorithms from the Sections 3 and 4. At any rate, analogously to the algorithm of [23], it does not seem as though there exists a singly exponential upper bound on the complexity of the algorithm from [5] even if it is only applied to equational formulae of this restricted form. This is due to the fact that, similarly to Tajine's algorithm, also the algorithm from [5] requires that certain terms have to be transformed w.r.t. terms that are themselves the result of a previous transformation step.

Several publications deal with the complexity of the emptiness problem and of the finite explicit representability problem, respectively. In [8,9], and [10], the coNP-completeness of the emptiness problem was proven. Independently, the coNP-hardness of the emptiness problem was also shown in [16], where the coNP-hardness of the finite explicit representability problem was then proven by reducing the emptiness problem to it. A coNP-membership proof of the finite explicit representability problem of a single implicit generalization was given in [15]. The coNP-membership of the latter problem in case of disjunctions of implicit generalizations has been an open question so far.

The negation elimination problem can be shown to be non-elementary recursive if arbitrary equational formulae are considered. This follows immediately from the results in [25] and [16]: In the former paper, the non-elementary complexity of the satisfiability problem of equational formulae is established. In the latter work, the explicit representability problem is shown to be at least as hard as the emptiness problem of implicit generalizations. Equivalently, the negation elimination problem is at least as hard as the non-satisfiability problem of equational formulae. Not much has been known so far about the complexity of the negation elimination problem of subclasses of equational formulae (apart from equational formulae corresponding to implicit generalizations and very simple forms like quantifier-free conjunctions of equations and disequations, cf. [16]).

6. Conclusion

In this paper, we have revisited the finite explicit representability problem of implicit generalizations. We have provided an alternative algorithm for this decision problem which allowed us to prove the coNP-completeness in case of disjunctions of implicit generalizations. Similar ideas were applied to the negation elimination problem of purely existentially quantified equational formulae. Together with the transformations from [2] and [3] of arbitrary equational formulae into existentially quantified ones in DNF, our algorithm from Section 4.2 can also be seen as a step towards a more efficient negation elimination procedure for the general case.

For existentially quantified formulae in DNF, we have provided an exact complexity classification of the negation elimination problem by proving its coNP-completeness. In case of CNF, we have left a gap between the Π_2^p lower bound and the coNEXPTIME upper bound for future research. The most important aim for future research is clearly the search for further improvements of our algorithms both for the finite explicit representability problem and for the negation elimination problem of simple equational formulae. Moreover, one should try to extend the complexity investigations and the algorithms presented here to more general subclasses of equational formulae than the ones considered here.

Acknowledgements

I am very grateful to the anonymous referees for their detailed comments.

References

[1] R. Caferra, N. Zabel, Extending resolution for model construction, in: J. van Eijck (Ed.), Proc. Logics in AI, European Workshop (JELIA'90), Lecture Notes in Artificial Intelligence, vol. 478, Springer, Berlin, 1991, pp. 153–169.

[2] H. Comon, C. Delor, Equational formulae with membership constraints, Inform. and Comput. 112 (2) (1994) 167–216.
[3] H. Comon, P. Lescanne, Equational problems and disunification, J. Symbolic Comput. 7 (3/4) (1989) 371–425.
[4] Ch. Fermüller, A. Leitsch, Hyperresolution and automated model building, J. Logic Comput. 6 (2) (1996) 173–203.
[5] M. Fernández, Negation elimination in empty or permutative theories, J. Symbolic Comput. 26 (1) (1998) 97–133.
[6] G. Gottlob, R. Pichler, Working with ARMs: complexity results on atomic representations of Herbrand models, Proc. 14th Ann. IEEE Symp. on Logic in Computer Science (LICS'99), Trento, Italy, IEEE Computer Society, Silver Spring, MD, 1999, pp. 306–315.
[7] G. Gottlob, R. Pichler, Working with ARMs: complexity results on atomic representations of Herbrand models, Inform. and Comput. 165 (2001) 183–207.
[8] D. Kapur, P. Narendran, D. Rosenkrantz, H. Zhang, Sufficient-completeness, ground-reducibility and their complexity, Acta Inform. 28 (4) (1991) 311–350.
[9] K. Kunen, Answer sets and negation as failure, in: J.-L. Lassez (Ed.), Proc. 4th Internat. Conf. on Logic Programming (ICLP'87), Melbourne, Victoria, Australia, MIT Press, Cambridge, MA, 1987, pp. 219–228.
[10] G. Kuper, K. McAloon, K. Palem, K. Perry, Efficient parallel algorithms for anti-unification and relative complement, Proc. 3rd Ann. Symp. on Logic in Computer Science (LICS'88), Edinburgh, Scotland, UK, IEEE Computer Society, Silver Spring, MD, 1988, pp. 112–120.
[11] J.-L. Lassez, M. Maher, K. Marriott, Unification revisited, in: M. Boscarol, L. Carlucci Aielli, G. Levi (Eds.), Proc. Workshop on Foundations of Logic and Functional Programming, Lecture Notes in Computer Science, vol. 306, Trento, Italy, Springer, Berlin, 1986, pp. 67–113.
[12] J.-L. Lassez, M. Maher, K. Marriott, Elimination of negation in term algebras, in: A. Tarlecki (Ed.), Proc. 16th Internat. Symp. on Mathematical Foundations of Computer Science (MFCS'91), Lecture Notes in Computer Science, vol. 520, Kazimierz Dolny, Poland, Springer, Berlin, 1991, pp. 1–16.
[13] J.-L. Lassez, K. Marriott, Explicit representation of terms defined by counter examples, J. Automat. Reason. 3 (3) (1987) 301–317.
[14] M. Maher, Complete axiomatizations of the algebras of finite, rational and infinite trees, Proc. 3rd Ann. Symp. on Logic in Computer Science (LICS'88), Edinburgh, Scotland, UK, IEEE Computer Society, Silver Spring, MD, 1988, pp. 348–357.
[15] M. Maher, P. Stuckey, On inductive inference of cyclic structures, Ann. Math. Artificial Intelligence 15 (2) (1995) 167–208.
[16] K. Marriott, Finding explicit representations for subsets of the Herbrand Universe, Ph.D. thesis, University of Melbourne, Australia, 1988.
[17] A. Martelli, U. Montanari, An efficient unification algorithm, ACM Trans. Programming Languages and Systems 4 (2) (1982) 258–282.
[18] R. Pichler, Solving equational problems efficiently, in: H. Ganzinger (Ed.), Proc. 16th Internat. Conf. on Automated Deduction (CADE-16), Lecture Notes in Artificial Intelligence, vol. 1632, Trento, Italy, Springer, Berlin, 1999, pp. 97–111.
[19] R. Pichler, The explicit representability of implicit generalizations, in: L. Bachmai (Ed.), Proc. 11th Internat. Conf. on Rewriting Techniques and Applications (RTA 2000), Lecture Notes in Computer Science, vol. 1833, Norwich, UK, Springer, Berlin, 2000, pp. 187–202.
[20] R. Pichler, Negation elimination from simple equational formulae, in: U. Montanari, J.D.P. Rolim, E. Welzl (Eds.), Proc. 27th Internat. Coll. on Automata, Languages and Programming (ICALP 2000), Lecture Notes in Computer Science, vol. 1553, Geneva, Switzerland, Springer, Berlin, 2000, pp. 612–623.
[21] M. Rusinowitch, C. Kirchner, H. Kirchner, Deduction with symbolic constraints. Revue Française d'Intelligence Artificielle 4 (3) (1990) 9–52 (Special issue on Automatic Deduction).
[22] L.J. Stockmeyer, The polynomial time hierarchy, Theoret. Comput. Sci. 3 (1976) 1–12.
[23] M. Tajine, The negation elimination from syntactic equational formulas is decidable, in: C. Kirchner (Ed.), Proc. 5th Internat. Conf. on Rewriting Techniques and Applications (RTA'93), Lecture Notes in Computer Science, vol. 690, Montreal, Canada, Springer, Berlin, 1993, pp. 316–327.

[24] J.-J. Thiel, Stop losing sleep over incomplete data type specifications, Proc. 11th Ann. ACM Symp. on Principles of Programming Languages (POPL'84), Salt Lake City, Utah, ACM Press, New York, 1984, pp. 76–82.
[25] S. Vorobyov, An improved lower bound for the elementary theories of trees, in: M.A. McRobbie, K. Slaney (Eds.), Proc. 13th Internat. Conf. on Automated Deduction (CADE-13), Lecture Notes in Artificial Intelligence, vol. 690, New Brunswick, NJ, USA, Springer, Berlin, 1996, pp. 316–327.