UPMAIL Technical Report No. 139
April 14, 1997
ISSN 1100–0686

The Decidability of Simultaneous Rigid E-Unification with One Variable

Anatoli Degtyarev[*]
Computing Science Department
Uppsala University
Box 311, S-751 05 Uppsala
Sweden
anatoli@csd.uu.se

Yuri Gurevich[†]
EECS Department
University of Michigan
Ann Arbor, MI 48109-2122
USA
gurevich@umich.edu

Paliath Narendran[‡]
Department of Computer Science
University at Albany – SUNY
Albany, New York 12222
USA
dran@cs.albany.edu

Margus Veanes
Computing Science Department
Uppsala University
Box 311, S-751 05 Uppsala
Sweden
margus@csd.uu.se

Andrei Voronkov[§]
Computing Science Department
Uppsala University
Box 311, S-751 05 Uppsala
Sweden
voronkov@csd.uu.se

Box 311, S-751 05 Uppsala, Sweden
Phone: +48−18−18 25 00
Fax: +46−18−51 19 25

Abstract

We show that simultaneous rigid E-unification, or SREU for short, is decidable and in fact EXPTIME-complete in the case of one variable. This result implies that the $\forall^*\exists\forall^*$ fragment of intuitionistic logic with equality is decidable. Together with a previous result regarding the undecidability of the $\exists\exists$-fragment, we obtain *a complete classification of decidability of the prenex fragment of intuitionistic logic with equality, in terms of the quantifier prefix.* It is also proved that SREU with one variable and a constant bound on the number of rigid equations is P-complete.

[*]Supported by grants from the Swedish Royal Academy of Sciences, INTAS and NUTEK.

[†]Partially supported by grants from NSF, ONR and the Faculty of Science and Technology of Uppsala University.

[‡]Supported by the NSF grants CCR-9404930 and INT-9401087.

[§]Supported by a TFR grant.

1 Introduction

In Gallier, Raatz and Snyder [22] and Degtyarev, Gurevich and Voronkov [9], it is explained why simultaneous rigid E-unification, or SREU for short, plays such a fundamental role in automatic proof methods in classical logic with equality that are based on the Herbrand theorem, like semantic tableaux [18], the connection method [2] or the mating method [1], model elimination [34], and others.

It was shown recently in Degtyarev and Voronkov [12] that SREU is undecidable. The strong connections between SREU and intuitionistic logic with equality have led to new important decidability results in the latter area [13, 50]. It follows, for example, that the $\exists^*$-fragment of intuitionistic logic with equality is undecidable [14, 15]. This result is improved in Veanes [46] to the following.

The $\exists\exists$-fragment of intuitionistic logic with equality is undecidable.

The decidability of the $\exists$-fragment of intuitionistic logic with equality, or equivalenly SREU with one variable, has been an open problem which is settled in this paper. We prove the following.

SREU with one variable is decidable, in fact EXPTIME-complete.

This result is obtained by a polynomial time reduction of SREU with one variable to the intersection nonemptiness problem of finite tree automata. The latter problem is EXPTIME-complete [47]. By using an analogue of a Skolemization result for intuitionistic logic [13] we can deduce the following result.

The $\forall^*\exists\forall^*$-fragment of intuitionistic logic with equality is decidable.

The above results imply the following main contribution of this paper.

A complete classification of decidability of the prenex fragment of intuitionistic logic with equality, in terms of the quantifier prefix.

We prove also that rigid E-unification with one variable is P-complete and that SREU with one variable and a constant bound on the number of rigid equations is P-complete. One conclusion we can draw from this is that the intractability of SREU with one variable is strongly related to the *number* of rigid equations and not their *size*. With *two* variables, SREU is undecidable already with *three* rigid equations [26]. In Section 6 we summarize the current status of SREU and list some open problems.

2 Preliminaries

We will first establish some notation and terminology. We follow Chang and Keisler [4] regarding first order languages and structures. For the purposes of this paper it is enough to assume that the first order languages that we are dealing with are languages with equality and contain only function symbols and constants, so we will assume that from here on. We will in general use Σ, possibly with an index, to stand for a signature, i.e., Σ is a collection of function symbols with fixed arities. A function symbol of arity 0 is called a *constant*. We will always assume that Σ *contains at least one constant*.

2.1 Terms and Formulas

Terms and formulas are defined in the standard manner. We refer to terms and formulas collectively as *expressions*. In the following let X be an expression or a set of expressions or a sequence of such.

We write $\Sigma(X)$ for the *signature of* X, i.e., the set of all function symbols that occur in X, $\mathcal{V}(X)$ for the set of all free variables in X. We write $X(x_1, x_2, \ldots, x_n)$ to express that $\mathcal{V}(X) \subseteq \{x_1, x_2, \ldots, x_n\}$. Let $t_1, t_2, \ldots, t_n$ be terms, then $X(t_1, t_2, \ldots, t_n)$ denotes the result of replacing each (free) occurence of x_i in X by t_i for $1 \leq i \leq n$. By a *substitution* we mean a function from variables to terms. We will use θ to denote substitutions. We write $X\theta$ for $X(\theta(x_1), \theta(x_2), \ldots, \theta(x_n))$.

We say that X is *closed* or *ground* if $\mathcal{V}(X) = \emptyset$. By $\mathcal{T}_\Sigma$ or simply $\mathcal{T}$ we denote the set of all ground terms over the signature Σ. A substitution is called *ground* if its range consists of ground terms. A closed formula is called a *sentence*. Since there are no relation symbols all the atomic formulas are *equations*, i.e., of the form $t \approx s$ where t and s are terms and '$\approx$' is the formal equality sign.

2.2 First Order Structures

First order structures will (in general) be denoted by upper case gothic letters like $\mathfrak{A}$ and $\mathfrak{B}$ and their domains by corresponding capital roman letters like A and B respectively. A first order structure in a signature Σ is called a Σ-*structure*. For $F \in \Sigma$ we write $F^{\mathfrak{A}}$ for the interpretation of F in $\mathfrak{A}$.

For X a sentence or a set of sentences, $\mathfrak{A} \models X$ means that the structure $\mathfrak{A}$ is a *model of* or *satisfies* X according to Tarski's truth definition. A set of sentences is called *satisfiable* if it has a model. If X and Y are (sets of) sentences then $X \models Y$ means that Y is a *logical consequence* of X, i.e., that every model of X is a model of Y. We write $X \equiv Y$ when $X \models Y$ and $Y \models X$. We write $\models X$ to say that X is *valid*, i.e., true in all models.

By the *free algebra over* Σ we mean the Σ-structure $\mathfrak{A}$, with domain $\mathcal{T}_\Sigma$, such that for each n-ary function symbol $f \in \Sigma$ and $t_1, \ldots, t_n \in \mathcal{T}_\Sigma$, $f^{\mathfrak{A}}(t_1, \ldots, t_n) = f(t_1, \ldots, t_n)$. We let $\mathcal{T}_\Sigma$ also stand for the free algebra over

Σ.

Let E be a set of ground equations. Define the equivalence relation $=_E$ on $\mathcal{T}$ by $s =_E t$ iff $E \models s \approx t$. By $\mathcal{T}_{\Sigma/E}$ (or simply $\mathcal{T}_{/E}$) we denote the quotient of $\mathcal{T}_\Sigma$ over $=_E$. Thus, for all $s, t \in \mathcal{T}$,

$$\mathcal{T}_{/E} \models s \approx t \quad \Leftrightarrow \quad E \models s \approx t.$$

We call $\mathcal{T}_{/E}$ the *canonical model* of E. Structures that are isomorphic with the canonical model of a finite set of ground equations are sometimes called *finitely presented algebras*. Various problems that are related to finitely presented algebras, and their computational complexity, have been studied in Kozen [28, 29]. Below, we will make use of some of those results.

2.3 Simultaneous Rigid E-Unification

A *rigid equation* is an expression of the form $E \models_\forall s \approx t$ where E is a finite set of equations, called the *left-hand side* of the rigid equation, and s and t are arbitrary terms. A *system* of rigid equations is a finite set of rigid equations. A substitution θ is a *solution of* or *solves* a rigid equation $E \models_\forall s \approx t$ if

$$\models \quad (\bigwedge_{e \in E} e\theta) \Rightarrow s\theta \approx t\theta,$$

and θ is a *solution of* or *solves* a system of rigid equations if it solves each member of that system. The problem of solvability of systems of rigid equations is called *simultaneous rigid E-unification* or SREU for short. Solvability of a single rigid equation is called *rigid E-unification*. Rigid E-unification is known to be decidable, in fact NP-complete [21].

2.4 Term Rewriting

In some cases it is convenient to consider a system of ground equations as a rewrite system. We will assume that the reader is familiar with basic notions regarding ground term rewrite systems [16]. We will only use very elementary properties. In particular, we will use the following property of canonical (or convergent) rewrite systems. Let R be a ground and canonical rewrite system and consider it also as as a set of equations. For any ground term t, let $t{\downarrow_R}$ denote the normal form of t with respect to R. Then, for all ground terms t and s, (cf [16, Section 2.4])

$$R \models t \approx s \quad \Leftrightarrow \quad t{\downarrow_R} = s{\downarrow_R}.$$

A *reduced* set of rules R is such that for each rule $l \to r$ in R, l is irreducible with respect to $R \setminus \{l \to r\}$ and r is irreducible with respect to R. In the case of ground rules, a reduced set of rules is also canonical [43]. It is always possible to find a reduced set of ground rewite rules that is equivalent to a given finite set of ground equations [32]. Moreover, this can be done in $O(n \log n)$ time [43].

2.5 Finite Tree Automata

Finite tree automata, or simply tree automata from here on, is a generalization of classical automata. Tree automata were introduced, independently, in Doner [17] and Thatcher and Wright [45]. The main motivation was to obtain decidability results for the weak monadic second-order logic of the binary tree. Here we adopt the following definition of tree automata, that is based on rewrite rules [5, 7].

▶ A *tree automaton* or *TA* A is a quadruple (Q, Σ, R, F) where

- Q is a finite set of constants called *states*,
- Σ is a *signature* that is disjoint from Q,
- R is a set of *rules* of the form $f(q_1, \ldots, q_n) \to q$, where $f \in \Sigma$ has arity $n \geq 0$ and $q, q_1, \ldots, q_n \in Q$,
- $F \subseteq Q$ is the set of *final states.*

A is called a *deterministic* TA or DTA if there are no two different rules in R with the same left-hand side.

It is also assumed that Q and Σ are disjoint. Note that if A is deterministic then R is a reduced set of ground rewrite rules and thus canonical [43]. Tree automata as defined above are usually also called *bottom-up* tree automata. Acceptance for tree automata or recognizability is defined as follows.

▶ The set of terms *recognized* by a TA $A = (Q, \Sigma, R, F)$ is the set

$$T(A) = \{\, \tau \in \mathcal{T}_\Sigma \mid (\exists q \in F)\ \tau \xrightarrow{*}_R q \,\}.$$

A set of terms is called *recognizable* if it is recognized by some TA.

Two tree automata are *equivalent* if they recognize the same set of terms. It is well known that the nondeterministic and the deterministic versions of TAs have the same expressive power [17, 23, 45], i.e., for any TA there is an equivalent DTA. For an overview of the notion of recognizability in general algebraic structures see Courcelle [6] and the fundamental paper by Mezei and Wright [36].

3 Decidability of SREU with One Variable

In this section we will formally establish the decidability of SREU with one variable. The proof has two parts.

1. First we prove that rigid E-unification with one variable can be reduced to the problem of testing membership in a finite union of congruence classes.

2. By using the property that any finite union of congruence classes is recognizable, we then reduce SREU with one variable to the intersection nonemptiness problem of finite tree automata.

The decidability of SREU with one variable follows then from the fact that recognizable sets are closed under boolean operations and that the nonemptiness problem of finite tree automata is decidable. In Section 4 we will address the computational complexity of this reduction.

3.1 Reduction to Membership in a Union of Congruence Classes

We start by proving two lemmas. Roughly, these lemmas allow us to reduce an arbitrary rigid equation $S(x)$ with one variable to a finite collection of rigid equations $\{\, S_i(x) \mid i < n \,\}$ such that, for all substitutions θ, θ solves S iff θ solves some S_i. Furthermore, each of the S_i's has the form $E \models_{\forall} x = t_i$ where E is ground and t_i is some ground term. The set E is common to all the S_i's.

Let E be a set of ground equations and t a ground term. We denote by $[t]_E$ the interpretation of t in $\mathcal{T}_{/E}$, in other words $[t]_E$ is the congruence class induced by $=_E$ on $\mathcal{T}$ that includes t. For a set T of ground terms we will write $[T]_E$ for $\{\, [t]_E \mid t \in T \,\}$. We write $\mathit{Terms}(E)$ for the set of all terms that occur in E, in particular $\mathit{Terms}(E)$ is closed under the subterm relation. We will use the following lemma. Lemma 1 follows also from a more general statement in de Kogel [8, Theorem 5.11].

Lemma 1 *Let t be a ground term, c a constant, E a finite set of ground equations and e a ground equation. Let $T = \mathit{Terms}(E \cup \{e\})$. If $[t]_E \notin [T]_E$ and $E \cup \{t \approx c\} \models e$ then $E \models e$.*

Proof. Assume that $[t]_E \notin [T]_E$ and that $E \cup \{t \approx c\} \models e$. Let E' be a reduced set of rules equivalent to E, such that $c{\downarrow}_{E'} = c$. Let $t' = t{\downarrow}_{E'}$. If $t' = c$ then

$$E \cup \{t \approx c\} \equiv E' \cup \{t \approx c\} \equiv E' \cup \{t' \approx c\} \equiv E$$

and the statement follows immediately. So assume that $t' \neq c$. Let $R = E' \cup \{t' \to c\}$. Let $l \to r$ be a rule in E'. Neither l nor r can be reduced with the rule $t' \to c$ because $[t']_E = [t]_E \notin [T]_E$. Hence R is reduced, and thus canonical [43]. Also, $R \equiv E \cup \{t \approx c\}$. (Note that $t' \in [t]_E$ and $[T]_E = [T]_{E'}$.)

Let $e = t_0 \approx s_0$ and let $u = t_0{\downarrow}_R = s_0{\downarrow}_R$. We have that

$$t_0 \xrightarrow{*}_R u, \quad s_0 \xrightarrow{*}_R u.$$

Consider the reduction $t_0 \xrightarrow{*}_R u$ and let $t_i \longrightarrow t_{i+1}$ be any rewrite step in that reduction. Obviously, if each subterm of t_i is in some congruence class in $[T]_E$ then the rule $t' \to c$ is not applicable since $[t']_E \notin [T]_E$ and it follows also that each subterm of t_{i+1} is in some congruence class in $[T]_E$. It follows by induction on i that the rule $t' \to c$ is not used in the reduction. The same argument holds for $s_0 \xrightarrow{*}_R u$. Hence

$$t_0 \xrightarrow{*}_{E'} u, \quad s_0 \xrightarrow{*}_{E'} u,$$

and thus $E' \models t_0 \approx s_0$. Hence $E \models e$. ⊠

Consider a system S of rigid equations. There is an extreme case of rigid equations that are easy to handle from the point of view of solvability of S, namely the redundant ones:

- A rigid equation is *redundant* if all substitutions solve it.

To decide if a rigid equation $E(x) \models_\forall s(x) \approx t(x)$ is redundant, it is enough to decide if $E(c) \models s(c) \approx t(c)$ where c is a new constant.

- The *uniform word problem for ground equations* is the following decision problem. Given a set of ground equations E and a ground equation e, is e a logical consequence of E?

We will use the following complexity result [28, 29].

Theorem 1 (Kozen) *The uniform word problem for ground equations is P-complete.*

So redundancy of rigid equations is decidable in polynomial time.

Lemma 2 *Let $E(x) \models_\forall e(x)$ be a rigid equation, c be a new constant and t be a ground term not containing c. Then*

$$E(c) \cup \{t \approx c\} \models e(c) \quad \Leftrightarrow \quad E(t) \models e(t).$$

Proof. The only non-obvious direction is '$\Rightarrow$'. Since t does not include c, $E(c) \cup \{t \approx c\} \models e(c)$ holds with c replaced by t, but then the equation $t \approx t$ is simply superfluous. ⊠

Clearly, S is solvable iff the set of rigid equtions in S that are not redundant, is solvable. We will use the following lemma.

Lemma 3 *Let $E(x) \models_\forall s_0(x) \approx t_0(x)$ be a non-redundant rigid equation of one variable x and let c be a new constant. There exists a finite set of ground terms T such that, for any ground term t not containing c the following holds:*

$$E(t) \models s_0(t) \approx t_0(t) \quad \Leftrightarrow \quad E(c) \models t \approx s \textit{ for some } s \in T.$$

Furthermore, T can be obtained in polynomial time.

Proof. Let T' be the set $\mathit{Terms}(E(c) \cup \{s_0(c) \approx t_0(c)\})$. Let

$$T = \{\, s \in T' \mid E(c) \cup \{s \approx c\} \models s_0(c) \approx t_0(c) \,\}.$$

Note that T may be empty. Let t be any ground term that does not contain c. By using Lemma 2, it is enough to prove that the following statements are equivalent:

1. $E(c) \cup \{t \approx c\} \models s_0(c) \approx t_0(c)$,
2. $E(c) \models t \approx s$ for some $s \in T$.

Assume first that $[t]_{E(c)} \notin [T']_{E(c)}$. In particular $[t]_{E(c)} \notin [T]_{E(c)}$, so statement 2 is trivially false. Suppose (by contradiction) that statement 1 holds. But then $E(c) \models s_0(c) \approx t_0(c)$ by Lemma 1, which contradicts the assumption that the rigid equation is not redundant.

Assume now that $[t]_{E(c)} = [s]_{E(c)}$ for some $s \in T'$. Thus

$$E(c) \cup \{s \approx c\} \equiv E(c) \cup \{t \approx c\}. \tag{1}$$

So, if $s \in T$ then statement 2 is trivially true and statement 1 is true by (1) and the definition of T. If on the other hand $s \notin T$ then statement 2 is trivially false and statement 1 is false by (1) and the definition of T.

Observe that the size of T' is proportional to the size of the rigid equation, and to decide if some term in T' belongs to T takes polynomial time by Kozen's result. So the construction of T takes polynomial time. ⊠

From Lemma 3 we get the following result.

Theorem 2 *Rigid E-unification with one variable is P-complete.*

Proof. P-hardness of rigid E-unification with one variable follows immediately from P-hardness of the uniform word problem for ground equations. Inclusion in P is proved as follows. Let $S(x) = E(x) \models_{\forall} e(x)$ be a rigid equation. Test first that $S(x)$ is not redundant. If so, use Lemma 3 to obtain T. Now, $S(x)$ is solvable iff T is nonempty. ⊠

This P-completeness result is extended in Section 4.3 to SREU with one variable and a constant bound on the number of rigid equations.

4 Computational Complexity of SREU with One Variable

In the previous section we showed that SREU with one variable is decidable. We paid little or no attention to the actual computational complexity of this decision problem. Here we take a closer look at the reduction and show that SREU with one variable is in fact EXPTIME-complete. We first introduce the following definition.

- ▶ The *intersection nonemptiness problem of DTAs* or *DTAI* is the following decision problem. Given a collection $\{ A_i \mid 1 \leq i \leq n \}$ of DTAs, is $\bigcap_{i=1}^{n} T(A_i)$ nonempty?

We will use the following result that has been observed by other authors [19, 24, 41] and strictly proved in Veanes [47].

Theorem 3 (Veanes) *DTAI is EXPTIME-complete.*

We will first show that SREU with one variable reduces to DTAI in polynomial time. This establishes the inclusion of SREU with one variable in EXPTIME. We then show that DTAI reduces to SREU with one variable, which shows the hardness part. The construction that we will use is in fact based on a construction in de Kogel [8, Theorems 4.1 and 4.2] that is based on Shostak's congruence closure algorithm [42].[1] A similar construction is used also in Gurevich and Voronkov [27].

4.1 SREU with one variable is in EXPTIME

In the following we will assume that none of the rigid equations are redundant. Lemma 3 tells us that the set of solutions of a rigid equation $E(x) \models_\forall e(x)$ with one variable is given by the union of a finite number of congruence classes

$$\bigcup_{s \in T} \{ t \mid E(c) \models s \approx t \},$$

where $T \subseteq \mathit{Terms}(E(c) \cup \{e(c)\})$ and c is a new constant. We will now give a polynomial time construction of a DTA that recognizes the above set of terms. Our considerations lead naturally to the following definition. Let E be a set of ground equations and T a subset of $\mathit{Terms}(E)$.

- ▶ A DTA $A = (Q, \Sigma, R, F)$ is *presented by* (E, T) if A has the following form (modulo renaming of states). First, let q_C be a new state for each $C \in [\mathit{Terms}(E)]_E$.

$$\begin{array}{rcl} Q &=& \{ q_C \mid C \in [\mathit{Terms}(E)]_E \}, \\ \Sigma &=& \Sigma(E), \\ F &=& \{ q_C \mid C \in [T]_E \}, \\ R &=& \{ f(q_{[t_1]_E}, \ldots, q_{[t_n]_E}) \to q_{[t]_E} \mid t = f(t_1, \ldots, t_n) \in \mathit{Terms}(E) \}. \end{array}$$

It is clear that the above definition is well defined. It follows from elementary properties of congruence relations that A is deterministic and thus R is reduced. Note that for each constant c in $\Sigma(E)$, there is a rule $c \to q_{[c]_E}$ in R. Note also that for any equation $s \approx t$ in E, both s and t reduce to the same normal form $q_{[s]_E} = q_{[t]_E}$ with respect to R, since they belong to $\mathit{Terms}(E)$. We will use the following lemma.

[1] De Kogel does not use tree automata but the main idea is the same.

Lemma 4 *Let E be a set of ground equations and $T \subseteq Terms(E)$. Let A be a DTA presented by (E,T). Then*

1. $T(A) = \{ t \in \mathcal{T}_{\Sigma(E)} \mid (\exists s \in T)\, E \models t \approx s \}$,
2. *A can be constructed in polynomial time from E and T.*

Proof. To prove the first statement, consider a Σ-structure $\mathfrak{A}$ with the universe $\{ t{\downarrow_R} \mid t \in \mathcal{T}_{\Sigma\cup\Gamma} \}$ and the interpretation function such that $t^{\mathfrak{A}} = t{\downarrow_R}$ for all $t \in \mathcal{T}_\Sigma$. Clearly, it is enough to prove that, for all $t, s \in \mathcal{T}_\Sigma$,

$$E \models t \approx s \quad \Leftrightarrow \quad \mathfrak{A} \models t \approx s.$$

For a proof of this statement see de Kogel [8].

The second part is proved as follows. The number of terms in *Terms*(E) is proportional to the size of E. It follows by Theorem 1 that the time complexity of the construction of Q, i.e., the time complexity to partition *Terms*(E) into congruence classes, is polynomial. The rest is obvious. ⊠

We prove now that SREU with one variable is in EXPTIME.

Lemma 5 *SREU with one variable is in EXPTIME.*

Proof. Let $S(x) = \{ S_i(x) \mid 1 \le i \le n \}$ be a system of rigid equations. Assume, without loss of generality, that none of the rigid equations is redundant. Let $S_i(x) = E_i(x) \models_\forall e_i(x)$. Let Σ be the signature of S. Use Lemma 3 to obtain, for each i, $1 \le i \le n$, a set of ground terms T_i in polynomial time such that, for all t in $\mathcal{T}_\Sigma$,

$$E_i(t) \models e_i(t) \quad \Leftrightarrow \quad E_i(c) \models t \approx s \text{ for some } s \in T_i.$$

Use now Lemma 4 to obtain (in polynomial time) a DTA A_i that presents $(E_i(c), T_i)$, for $1 \le i \le n$. It follows by Lemma 3 and the first part of Lemma 4 that

$$T(A_i) = \{ t \in \mathcal{T}_\Sigma \mid E_i(t) \models e_i(t) \} \quad (\text{for } 1 \le i \le n).$$

Thus, θ is a solution to $S(x)$ iff $x\theta$ is recognizable by all $T(A_i)$. Consequently, $S(x)$ is solvable iff $\bigcap_{i=1}^n T(A_i)$ is nonempty. The lemma follows, since DTAI is in EXPTIME. ⊠

Remark Decidability of SREU with one variable can also be proved by combining Lemma 3 with a result by Brainerd [3] that states that, given a set R of a ground rewrite rules and a set T of ground terms, then the set $\{ t \mid (\exists s \in T)\; t \xrightarrow{*}_R s \}$ is recognizable. This proof would not give us the computational complexity result, however.

4.2 SREU with one variable is EXPTIME-complete

We will reduce DTAI to SREU with one variable to establish the hardness part. First, let us state some simple but useful facts.

Lemma 6 *Let $A = (Q, \Sigma, R, F)$ be a DTA, f be a unary function symbol not in Σ, and c be a constant not in Q or Σ. Let*

$$S(x) = (R \cup \{ f(q) \to c \mid q \in F \} \models_{\forall} x \approx c).$$

Then, for all θ such that $x\theta \in \mathcal{T}_{\Sigma \cup \{f\}}$,

$$\theta \text{ solves } S(x) \quad \Leftrightarrow \quad x\theta = f(t) \text{ for some } t \in T(A).$$

Proof. Let $E = R \cup \{ f(q) \to c \mid q \in F \}$. From the fact that R is reduced and that $f(q)$ is irreducible in E and c is irreducible in R, follows that E is reduced and thus canonical. So, for any $x\theta \in \mathcal{T}_{\Sigma \cup \{f\}}$, $E \models x\theta \approx c$ iff $x\theta \xrightarrow{*}_E c$. But

$$\begin{aligned} x\theta \xrightarrow{*}_E c \quad &\Leftrightarrow \quad x\theta \xrightarrow{*}_E f(q) \longrightarrow c \text{ for some } q \in F \\ &\Leftrightarrow \quad x\theta = f(t) \text{ for some } t \in \mathcal{T}_\Sigma \text{ and } t \xrightarrow{*}_R q \\ &\Leftrightarrow \quad x\theta = f(t) \text{ for some } t \in T(A). \end{aligned}$$

⊠

For a given signature Σ, and some constant c in it, let us denote by $S_\Sigma(x)$ the following rigid equation:

$$S_\Sigma(x) = (\{ \sigma(c, \ldots, c) \approx c \mid \sigma \in \Sigma \} \models_{\forall} x \approx c).$$

The following lemma is elementary [15].

Lemma 7 *For all θ, θ solves $S_\Sigma(x)$ iff $x\theta \in \mathcal{T}_\Sigma$.*

We have now reached the point where we can state and easily prove the following result.

Theorem 4 *SREU with one variable is EXPTIME-complete.*

Proof. Inclusion in EXPTIME follows by Lemma 5. Let $\{ A_i \mid 1 \le i \le n \}$ be a collection of DTAs with a signature Σ. Let f be a new unary function symbol and $\Sigma' = \Sigma \cup \{f\}$. For each A_i, let $S_i(x)$ be the rigid equation given by Lemma 6. So, for all θ such that $x\theta \in \mathcal{T}_{\Sigma'}$,

$$\theta \text{ solves } S_i(x) \quad \Leftrightarrow \quad x\theta = f(t) \text{ for some } t \in T(A_i).$$

Let

$$S(x) = \{\, S_i(x) \mid 1 \leq i \leq n \,\} \cup \{S_{\Sigma'}(x)\}.$$

It follows by Lemma 7 that for any θ that solves $S(x)$, $x\theta$ is in $\mathcal{T}_{\Sigma'}$. Hence, by Lemma 6, $S(x)$ is solvable iff $\bigcap_{i=1}^{n} T(A_i)$ is nonempty. Obviously, $S(x)$ has been constructed in polynomial time. The statement follows, since DTAI is EXPTIME-hard. ⊠

So in the general case, SREU is already intractable with one variable. It should be noted however that the exponential behaviour is strongly related to the unboundedness of the number of rigid equations. (See Section 4.3.)

4.3 Bounded SREU with One Variable

The exponential worst case behaviour of SREU with one variable is strongly related to the unboundedness of the number of rigid equations, and not to the size or other parameters of the rigid equations. This behaviour is explained by the fact that the intersection nonemptiness problem of a family of DTAs is in fact the nonemptiness problem of the corresponding direct product of the family. The size of a direct product of a family of DTAs is proportional to the product of the sizes of the members of the family, and the time complexity of the nonemptiness problem of a DTA is polynomial.

- ▶ *Bounded SREU* is SREU with a number of rigid equations that is bounded by some fixed positive integer.

We will use the following definition.

- ▶ The *nonemptiness* problem of TAs is the following decision problem. Given a TA A, is $T(A)$ nonempty?

The nonemptiness problem of DTAs is basically the problem of "generability" of finitely presented algebras. The latter problem is P-complete [29] and thus, by a very simple reduction, also the former problem [47].[2] In general the following holds.

Theorem 5 (Veanes) *The nonemptiness problem of TAs is in P and P-hard already for DTAs.*

For bounded SREU with one variable we get the following result.

Theorem 6 *Bounded SREU with one variable is P-complete.*

[2] The book of Greenlaw, Hoover and Ruzzo [25] includes an excellent up-to-date survey of around 150 P-complete problems, including generability.

Proof. Let the number of rigid equtions be bounded by some fixed positive integer n. P-hardness follows immediately from Theorem 2. Without loss of generality consider a system

$$S(x) = \{ S_i(x) \mid 1 \le i \le n \}$$

of exactly n rigid equations. For each S_i construct a DTA A_i in polynomial time, like in Lemma 5. Let A be the DTA that recognizes $\bigcap_{i=1}^{n} T(A_i)$. For example, A can be the direct product of $\{ A_i \mid 1 \le i \le n \}$ (Gécseg and Steinby [23]). It is straightforward to construct A in time that is proportional to the product of the sizes of the A_i's. Hence A is obtained in polynomial time (because n is fixed) and $T(A)$ is nonempty iff $S(x)$ is solvable. ⊠

4.4 Monadic SREU with One Variable

When we restrict the signature to consist of function symbols of arity ≤ 1, i.e., when we consider the so-called *monadic* SREU then the complexity bounds are different. We can note that DTAs restricted to signatures with just unary function symbols correspond to classical deterministic finite automata or DFAs. It was proved by Kozen that the computational complexity of the intersection nonemptiness problem of DFAs is PSPACE-complete [30]. So, by using this fact we can see that Theorem 4 proves that monadic SREU with one variable is PSPACE-complete.

Monadic SREU is studied in detail elsewhere [27]. We can note that, in general, the decidability of monadic SREU is still an open problem. There is also a very close connection between monadic SREU and the prenex fragment of intuitionistic logic with equality restricted to function symbols of arity ≤ 1 [13].

5 Implications to the Prenex Fragment of Intuitionistic Logic

The *prenex fragment* of intuitionistic logic is the collection of all intuitionistically provable prenex formulas. Many new decidability results about the prenex fragment have been obtained quite recently by Degtyarev and Voronkov [13, 14, 15] and Voronkov [49]. Some of these results are:

1. Decidability, and in particular PSPACE-completeness, of the prenex fragment of intuitionistic logic *without* equality [49].

2. Prenex fragment of intuitionistic logic *with* equality but *without* function symbols is PSPACE-complete [13]. Decidability of this fragment was proved in Orevkov [39].

3. Prenex fragment of intuitionistic logic with equality in the language with one unary function symbol is decidable [13].

4. $\exists^*$-fragment of intuitionistic logic with equality is undecidable [14, 15].

In some of the above results, the corresponding result has first been obtained for a fragment of SREU with similar restrictions. For example, the proof of the last statement is based on the undecidability of SREU. The undecidability of the $\exists^*$-fragment is improved in Veanes [46] where it is proved that, already the

5. $\exists\exists$-fragment of intuitionistic logic with equality is undecidable.

With the following result we obtain a complete characterization of decidability of the prenex fragment of intutionistic logic with equality with respect to quantifier prefix.

Theorem 7 *The $\forall^*\exists\forall^*$-fragment of intuitionistic logic with equality is decidable and EXPTIME-hard.*

Proof. Intuitionistic provability of any formula in the $\forall^*\exists\forall^*$-fragment can be reduced to solvability of SREU with one variable [13]. Conversely, solvability of a system of rigid equations with one variable reduces trivially to provability of a corresponding formula in the $\exists$-fragment [13]. The statement follows by Theorem 4. ⊠

Remark The undecidability of the $\exists\exists$-fragment holds if there is one binary function symbol in the signature. The reduction in Theorem 7 from a $\forall^*\exists\forall^*$-formula to SREU with one variable may take exponential time, so the precise computational complexity for this fragment is unkown at this moment.

Other fragments Decidability problems for other fragments of intuitionistic logic have been studied by Orevkov [38, 39], Mints [37], Statman [44] and Lifschitz [33]. Orevkov proves that the $\neg\neg\forall\exists$-fragment of intuitionistic logic with function symbols is undecidable [38]. Lifschitz proves that intuitionistic logic with equality and without function symbols is undecidable, i.e., that the pure constructive theory of equality is undecidable [33]. Orevkov shows decidability of some fragments (that are close to the prenex fragment) of intuitionistic logic with equality [39]. Statman proves that the intuitionistic propositional logic is PSPACE-complete [44].

6 Current Status of SREU

Here we briefly summarize the current status of SREU. The first decidability proof of rigid E-unification is given in Gallier, Narendran, Plaisted and Snyder [21]. Recently a simpler proof, without computational complexity considerations, has been given by de Kogel [8]. We start with the **solved cases**:

- Rigid E-unification with ground left-hand side is NP-complete [31]. Rigid E-unification in general is NP-complete and there exist finite complete sets of unifiers [21, 20].
- Rigid E-unification with one variable is P-complete (Theorem 2). Or, more generally, SREU with one variable and a bounded number of rigid equations is P-complete (Theorem 6).
- If all function symbols have arity ≤ 1 (the *monadic* case) then it follows that SREU is PSPACE-hard [24]. If only one unary function symbol is allowed then the problem is decidable [11, 10]. If only constants are allowed then the problem is NP-complete [11] if there are at least two constants.
- About the monadic case it is known that SREU with more than two unary function symbols is decidable iff it is decidable with just two unary function symbols [11].
- If the left-hand sides are ground then the monadic case is decidable [27]. Monadic SREU with one variable is PSPACE-complete [27].
- The word equation solving [35] (unification under associativity), which is an extremely hard problem with no interesting known computational complexity bounds, can be reduced to monadic SREU [10].
- Monadic SREU is equivalent to a non-trivial extension of word equations [27].
- Monadic SREU is equivalent to the provability problem of the prenex fragment of intuitionistic logic with equality with function symbols of arity ≤ 1 [13].
- In general SREU is undecidable [12]. Moreover, it is undecidable with ground left-hand sides [40]. Furthermore, SREU is undecidable with three rigid equations with ground left-hand sides and two variables [48, 26].
- SREU with one variable is decidable, in fact EXPTIME-complete (Theorem 4).

Note also that SREU is decidable when there are no variables, since each rigid equation can be decided for example by using any congruence closure algorithm or ground term rewriting technique. Actually, the problem is then P-complete because the uniform word problem for ground equations is P-complete [29]. The **unsolved cases** are:

? Decidability of monadic SREU [27].

? Decidability of SREU with *two* rigid equations.

Both problems are highly non-trivial.

References

[1] P.B. Andrews. Theorem proving via general matings. *Journal of the Association for Computing Machinery*, 28(2):193–214, 1981.

[2] W. Bibel. *Deduction. Automated Logic.* Academic Press, 1993.

[3] W.S. Brainerd. Tree generating regular systems. *Information and Control*, 14:217–231, 1969.

[4] C.C. Chang and H.J. Keisler. *Model Theory.* North-Holland, Amsterdam, third edition, 1990.

[5] J.L. Coquidé, M. Dauchet, R. Gilleron, and S. Vágvölgyi. Bottom-up tree pushdown automata: classification and connection with rewrite systems. *Theoretical Computer Science*, 127:69–98, 1994.

[6] B. Courcelle. On recognizable sets and tree automata. In M. Nivat and H. Ait-Kaci, editors, *Resolution of Equations in Algebraic Structures.* Academic Press, 1989.

[7] M. Dauchet. Rewriting and tree automata. In H. Comon and J.P. Jouannaud, editors, *Term Rewriting (French Spring School of Theoretical Computer Science)*, volume 909 of *Lecture Notes in Computer Science*, pages 95–113. Springer Verlag, Font Romeux, France, 1993.

[8] E. De Kogel. Rigid E-unification simplified. In P. Baumgartner, R. Hähnle, and J. Posegga, editors, *Theorem Proving with Analytic Tableaux and Related Methods*, number 918 in Lecture Notes in Artificial Intelligence, pages 17–30, Schloß Rheinfels, St. Goar, Germany, May 1995.

[9] A. Degtyarev, Yu. Gurevich, and A. Voronkov. Herbrand's theorem and equational reasoning: Problems and solutions. In *Bulletin of the European Association for Theoretical Computer Science*, volume 60. October 1996. The "Logic in Computer Science" column.

[10] A. Degtyarev, Yu. Matiyasevich, and A. Voronkov. Simultaneous rigid E-unification is not so simple. UPMAIL Technical Report 104, Uppsala University, Computing Science Department, April 1995.

[11] A. Degtyarev, Yu. Matiyasevich, and A. Voronkov. Simultaneous rigid E-unification and related algorithmic problems. In *Eleventh Annual IEEE Symposium on Logic in Computer Science (LICS'96)*, pages 494–502, New Brunswick, NJ, July 1996. IEEE Computer Society Press.

[12] A. Degtyarev and A. Voronkov. Simultaneous rigid E-unification is undecidable. UPMAIL Technical Report 105, Uppsala University, Computing Science Department, May 1995.

[13] A. Degtyarev and A. Voronkov. Decidability problems for the prenex fragment of intuitionistic logic. In *Eleventh Annual IEEE Symposium on Logic in Computer Science (LICS'96)*, pages 503–512, New Brunswick, NJ, July 1996. IEEE Computer Society Press.

[14] A. Degtyarev and A. Voronkov. Simultaneous rigid E-unification is undecidable. In H. Kleine Büning, editor, *Computer Science Logic. 9th International Workshop, CSL'95*, volume 1092 of *Lecture Notes in Computer Science*, pages 178–190, Paderborn, Germany, September 1995, 1996.

[15] A. Degtyarev and A. Voronkov. The undecidability of simultaneous rigid E-unification. *Theoretical Computer Science*, 166(1–2):291–300, 1996.

[16] N. Dershowitz and J.-P. Jouannaud. Rewrite systems. In J. Van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B: Formal Methods and Semantics, chapter 6, pages 243–309. North Holland, Amsterdam, 1990.

[17] J. Doner. Tree acceptors and some of their applications. *Journal of Computer and System Sciences*, 4:406–451, 1970.

[18] M. Fitting. First-order modal tableaux. *Journal of Automated Reasoning*, 4:191–213, 1988.

[19] T. Frühwirth, E. Shapiro, M. Vardi, and E. Yardeni. Logic programs as types of logic programs. In *Proc. 6th Symposium on Logics in Computer Science (LICS)*, pages 300–309, 1991.

[20] J. Gallier, P. Narendran, D. Plaisted, and W. Snyder. Rigid E-unification: NP-completeness and applications to equational matings. *Information and Computation*, 87(1/2):129–195, 1990.

[21] J.H. Gallier, P. Narendran, D. Plaisted, and W. Snyder. Rigid E-unification is NP-complete. In *Proc. IEEE Conference on Logic in Computer Science (LICS)*, pages 338–346. IEEE Computer Society Press, July 1988.

[22] J.H. Gallier, S. Raatz, and W. Snyder. Theorem proving using rigid E-unification: Equational matings. In *Proc. IEEE Conference on Logic in Computer Science (LICS)*, pages 338–346. IEEE Computer Society Press, 1987.

[23] F. Gécseg and M. Steinby. *Tree Automata*. Akadémiai Kiodó, Budapest, 1984.

[24] J. Goubault. Rigid $\vec{E}$-unifiability is DEXPTIME-complete. In *Proc. IEEE Conference on Logic in Computer Science (LICS)*. IEEE Computer Society Press, 1994.

[25] R. Greenlaw, H.J. Hoover, and W.L. Ruzzo. *Limits to Parallel Computation: P-Completeness Theory*. Oxford University Press, 1995.

[26] Y. Gurevich and M. Veanes. Some undecidable problems related to the Herbrand theorem. UPMAIL Technical Report 138, Uppsala University, Computing Science Department, March 1997.

[27] Y. Gurevich and A. Voronkov. The monadic case of simultaneous rigid E-unification. UPMAIL Technical Report 137, Uppsala University, Computing Science Department, 1997. To appear in Proc. of *ICALP'97*.

[28] D. Kozen. Complexity of finitely presented algebras. Technical Report TR 76-294, Cornell University, Ithaca, N.Y., 1976.

[29] D. Kozen. Complexity of finitely presented algebras. In *Proc. of the 9th Annual Symposium on Theory of Computing*, pages 164–177, New York, 1977. ACM.

[30] D. Kozen. Lower bounds for natural proof systems. In *Proc. 18th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 254–266, 1977.

[31] D. Kozen. Positive first-order logic is NP-complete. *IBM J. of Research and Development*, 25(4):327–332, 1981.

[32] D.S. Lankford. Canonical inference. Technical report, Department of Mathematics, South-Western University, Georgetown, Texas, 1975.

[33] V. Lifschitz. Problem of decidability for some constructive theories of equalities (in Russian). *Zapiski Nauchnyh Seminarov LOMI*, 4:78–85, 1967. English Translation in: Seminars in Mathematics: Steklov Math. Inst. 4, Consultants Bureau, NY-London, 1969, p.29–31.

[34] D.W. Loveland. Mechanical theorem proving by model elimination. *Journal of the Association for Computing Machinery*, 15:236–251, 1968.

[35] G.S. Makanin. The problem of solvability of equations in free semigroups. *Mat. Sbornik (in Russian)*, 103(2):147–236, 1977. English Translation in American Mathematical Soc. Translations (2), vol. 117, 1981.

[36] J. Mezei and J.B. Wright. Algebraic automata and context-free sets. *Information and Control*, 11:3–29, 1967.

[37] G.E. Mints. Choice of terms in quantifier rules of constructive predicate calculus (in Russian). *Zapiski Nauchnyh Seminarov LOMI*, 4:78–85, 1967. English Translation in: Seminars in Mathematics: Steklov Math. Inst. 4, Consultants Bureau, NY-London, 1969, p.43–46.

[38] V.P. Orevkov. Unsolvability in the constructive predicate calculus of the class of the formulas of the type $\neg\neg\forall\exists$ (in Russian). *Soviet Mathematical Doklady*, 163(3):581–583, 1965.

[39] V.P. Orevkov. Solvable classes of pseudo-prenex formulas (in Russian). *Zapiski Nauchnyh Seminarov LOMI*, 60:109–170, 1976. English translation in: Journal of Soviet Mathematics.

[40] D.A. Plaisted. Special cases and substitutes for rigid E-unification. Technical Report MPI-I-95-2-010, Max-Planck-Institut für Informatik, November 1995.

[41] H. Seidl. Haskell overloading is DEXPTIME-complete. *Information Processing Letters*, 52(2):57–60, 1994.

[42] R. Shostak. An algorithm for reasoning about equality. *Communications of the ACM*, 21:583–585, July 1978.

[43] W. Snyder. Efficient ground completion: An $O(n\log n)$ algorithm for generating reduced sets of ground rewrite rules equivalent to a set of ground equations E. In G. Goos and J. Hartmanis, editors, *Rewriting Techniques and Applications*, volume 355 of *Lecture Notes in Computer Science*, pages 419–433. Springer-Verlag, 1989.

[44] R. Statman. Lower bounds on Herbrand's theorem. *Proc. American Mathematical Society*, 75(1):104–107, 1979.

[45] J.W. Thatcher and J.B. Wright. Generalized finite automata theory with an application to a decision problem of second-order logic. *Mathematical Systems Theory*, 2(1):57–81, 1968.

[46] M. Veanes. Uniform representation of recursively enumerable sets with simultaneous rigid E-unification. UPMAIL Technical Report 126, Uppsala University, Computing Science Department, July 1996.

[47] M. Veanes. On computational complexity of basic decision problems of finite tree automata. UPMAIL Technical Report 133, Uppsala University, Computing Science Department, January 1997.

[48] M. Veanes. The undecidability of simultaneous rigid E-unification with two variables. In *Proc. Kurt Gödel Colloquium KGC'97*, 1997. To appear in *LNCS*.

[49] A. Voronkov. Proof search in intuitionistic logic based on constraint satisfaction. In P. Miglioli, U. Moscato, D. Mundici, and M. Ornaghi, editors, *Theorem Proving with Analytic Tableaux and Related Methods. 5th International Workshop, TABLEAUX '96*, volume 1071 of *Lecture Notes in Artificial Intelligence*, pages 312–329, Terrasini, Palermo Italy, May 1996.

[50] A. Voronkov. Proof search in intuitionistic logic with equality, or back to simultaneous rigid E-unification. In M.A. McRobbie and J.K. Slaney, editors, *Automated Deduction — CADE-13*, volume 1104 of *Lecture Notes in Computer Science*, pages 32–46, New Brunswick, NJ, USA, 1996.